

Making Everything Easier!™

2nd Edition

Building Web Sites

ALL-IN-ONE

FOR
DUMMIES®

**9 BOOKS
IN 1**

- Preparations
- Site Design
- Site Construction
- Web Graphics
- Multimedia
- Audience Interaction
- E-Commerce
- Site Management
- Case Studies

Doug Sahlin
Claudia Snell



Building Web Sites

ALL-IN-ONE

FOR

DUMMIES[®]

2ND EDITION

Building Web Sites

ALL-IN-ONE

FOR

DUMMIES[®]

2ND EDITION

by Doug Sahlin and Claudia Snell



WILEY

Wiley Publishing, Inc.

Building Web Sites All-in-One For Dummies®, 2nd Edition

Published by

Wiley Publishing, Inc.

111 River Street

Hoboken, NJ 07030-5774

www.wiley.com

Copyright © 2009 by Wiley Publishing, Inc., Indianapolis, Indiana

Published by Wiley Publishing, Inc., Indianapolis, Indiana

Published simultaneously in Canada

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Trademarks: Wiley, the Wiley Publishing logo, For Dummies, the Dummies Man logo, A Reference for the Rest of Us!, The Dummies Way, Dummies Daily, The Fun and Easy Way, Dummies.com, Making Everything Easier, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates in the United States and other countries, and may not be used without written permission. All other trademarks are the property of their respective owners. Wiley Publishing, Inc., is not associated with any product or vendor mentioned in this book.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, please contact our Customer Care Department within the U.S. at 800-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002.

For technical support, please visit www.wiley.com/techsupport.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Library of Congress Control Number: 2008941621

ISBN: 978-0-470-38541-8

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1



About the Authors

Doug Sahlin is an author, photographer, and Web designer living in central Florida. His clients include attorneys, authors, artists, doctors, and musicians. He has written and coauthored over 16 books on office applications, Web design applications, and digital photography. His books have been translated into five foreign languages. When he's not busy writing, photographing clients, or designing Web sites, he enjoys playing the guitar and dabbling in watercolor painting.

Claudia Snell is Webmaster at Nichols College, Dudley, MA and a digital media producer based in Worcester, MA. She has worked as a digital media producer at Boston's CBS Television Stations, wbztv.com and tv38.com and also as an online producer at the Worcester Telegram & Gazette, a wholly owned subsidiary of The New York Times Company.

During her career, Claudia has been a freelance web and multimedia designer for marketing agencies, non-profits, and small business owners. She has also been a contractor on large corporate projects and sites. Claudia has also been a guest speaker at industry user groups, in class, and for professional groups.

Dedication

From Doug: Dedicated to the memory of my dear mother Inez. I miss you, kiddo.

From Claudia: This book is dedicated to my family: my husband, Jason Snell, and my kids, Sarah and Damon Bousquet.

Acknowledgments

From Doug: Thanks for Steve Hayes for making this project possible. Kudos to project editor Nicole Sholly for her sense of humor and dedication to the task at hand. Thanks to Teresa Artman for rendering squeaky clean text and providing insightful questions. Special thanks for fellow authors Bonnie Blake, Joyce Evans, and Ken Milburn for their friendship and inspiration. Thanks to literary agent extraordinaire, Margot Maley Hutchison, who shall go forward in time with the nickname Ollie. As always, thanks for my friends, mentors, and relatives, especially you Karen and Ted. Special thanks to my social secretary, an 11-year-old calico cat named Niki, who is also known as “Queen of the Universe.”

From Claudia: I would like to acknowledge the great support I have gotten from the faculty at Assumption College and most especially: Dr. Robert Fry, Scott Glushien, Brian Bercier, and Dr. Rockie Blunt. Their support and guidance has made it possible for me to pursue my career.

I would also like to acknowledge the Adobe User Group program, the team at Adobe, Ed Sullivan, and Amy Brooks, and all the managers. Participating in the User Group program gave me tremendous opportunities to learn. The user group managers are a fantastic resource. I always appreciated their thoughts and advice on Web design and development.

I'd also like to acknowledge the people at Wiley publishing, including Steve Hayes and Nicole Sholly who patiently guided me through this project. Thanks also to the team of editors, reviewers, and production staff who have worked hard to make this book better. And finally, a special thanks to my coauthor, Doug Sahlin.

Publisher's Acknowledgments

We're proud of this book; please send us your comments through our online registration form located at www.dummies.com/register/.

Some of the people who helped bring this book to market include the following:

Acquisitions, Editorial

Project Editor: Nicole Sholly

Executive Editor: Steve Hayes

Senior Copy Editor: Teresa Artman

Technical Editors: Joyce Nielsen,
Chris McCulloh

Editorial Manager: Kevin Kirschner

Editorial Assistant: Amanda Foxworth

Sr. Editorial Assistant: Cherie Case

Cartoons: Rich Tennant (www.the5thwave.com)

Composition Services

Project Coordinator: Katherine Key

Layout and Graphics: Stacie Brooks,
Reuben W. Davis, Sarah Philippart,
Christine Williams

Proofreaders: Laura Albert, Laura L. Bowman

Indexer: Slivoskey Indexing Services

Special Help: Brian Walls

Publishing and Editorial for Technology Dummies

Richard Swadley, Vice President and Executive Group Publisher

Andy Cummings, Vice President and Publisher

Mary Bednarek, Executive Acquisitions Director

Mary C. Corder, Editorial Director

Publishing for Consumer Dummies

Diane Graves Steele, Vice President and Publisher

Composition Services

Gerry Fahey, Vice President of Production Services

Debbie Stailey, Director of Composition Services

Contents at a Glance

<i>Introduction</i>	1
<i>Book I: Preparations</i>	7
Chapter 1: Planning a Web Project	9
Chapter 2: Build a Web Team or Go It Alone.....	21
Chapter 3: Developing the Content	35
Chapter 4: Creating Relevant Site Content	43
<i>Book II: Site Design</i>	55
Chapter 1: Conceptualizing Your Site	57
Chapter 2: Creating Effective Layouts	71
Chapter 3: Organizing Your Content	91
Chapter 4: Ensuring Visual Appeal	103
<i>Book III: Site Construction</i>	123
Chapter 1: Pulling Together Tools and Materials	125
Chapter 2: Making a Web Page with HTML	145
Chapter 3: Creating a Web Page Layout with HTML and CSS	161
Chapter 4: Introducing Web and Graphics Software.....	179
Chapter 5: Getting Started with Dreamweaver	201
Chapter 6: Using Dreamweaver: Advanced Techniques.....	229
Chapter 7: Creating a Web Page from a Photoshop File.....	253
Chapter 8: Meeting HTML's Powerful Friends	273
Chapter 9: Web Site Usability and Accessibility	287
Chapter 10: Publishing the Web Site	299
<i>Book IV: Web Graphics</i>	313
Chapter 1: Finding Inspiration.....	315
Chapter 2: Bringing Your Vision to Life	329
Chapter 3: Workhorse Graphics	345
Chapter 4: Creating Compelling Graphics	367

<i>Book V: Multimedia</i>	391
Chapter 1: Incorporating Flash Creations	393
Chapter 2: Using Sound	423
Chapter 3: Using Web Video	437
Chapter 4: Active Slideshow Pro.....	455
<i>Book VI: Audience Interaction</i>	471
Chapter 1: Adding Basic Interactivity with HTML Forms and JavaScript	473
Chapter 2: An Introduction to ASP	499
Chapter 3: Introduction to PHP	519
Chapter 4: MySQL and PHP	531
Chapter 5: Additional Site Interactivity	543
<i>Book VII: E-Commerce</i>	563
Chapter 1: An E-Commerce Primer	565
Chapter 2: Building an E-Commerce Site	585
Chapter 3: Maintaining an E-Commerce Site	609
<i>Book VIII: Site Management</i>	627
Chapter 1: Helping a Site Succeed	629
Chapter 2: Maintaining a Web Site.....	649
Chapter 3: Expanding a Site.....	679
<i>Book IX: Case Studies</i>	691
Chapter 1: Blog Site	693
Chapter 2: Podcasting Site.....	711
Chapter 3: Photographer/Portfolio Site	721
Chapter 4: Foundation Site	731
<i>Index</i>	737

Table of Contents

Introduction 1

About This Book	1
Foolish Assumptions	1
Conventions Used in This Book	2
What You Don't Have to Read	2
How This Book Is Organized	3
Book I: Preparations	3
Book II: Site Design	3
Book III: Site Construction	3
Book IV: Web Graphics	3
Book V: Multimedia	4
Book VI: Audience Interaction	4
Book VII: E-Commerce	4
Book VIII: Site Management	4
Book IX: Case Studies	4
Icons Used in This Book	5
Where to Go from Here	5

Book I: Preparations 7

Chapter 1: Planning a Web Project 9

Defining the Audience	9
Internal stakeholders	9
External audience	10
Setting Goals	12
Creating a Scope Document	12
Preparing to Get Started	13
Defining why you're doing the project	13
Brainstorming and evaluating your ideas	14
Looking at budget and timelines	14
Considering collaborations	15
Selling the Idea	15
Holding a Kick-Off Meeting	15
Revising Your Original Plans: Using Feedback to Improve	16
Receiving feedback	17
Giving feedback	18
Preparing to Redesign an Existing Site	19

Chapter 2: Build a Web Team or Go It Alone	21
Flying Solo: Skills You Need to Go It Alone	21
Managing the project	21
Serving your customer	22
Dealing with HTML, CSS, and other scripting	23
Using Photoshop, Fireworks, and other graphics applications	24
Developing content	25
Considering basic computer and Internet skills	26
Incorporating multimedia in your project	26
Handling a solo project	26
Hiring Members of the Team	27
Project manager	28
Web designer/new media designer	28
Web developer/programmer	28
Content developer/writer	29
Webmaster/host	30
Other professional help	30
Managing the team	31
Handing Off a Project to a Client	32
Chapter 3: Developing the Content	35
Knowing What to Put on Your Site	35
Finding out what your client's visitors need	36
Dealing with copyright issues	37
Finding out what your client needs from site visitors	38
Reconciling the content with the goals	40
Keeping It Fresh: An Ongoing Process	41
Assigning content development tasks	41
Keeping the graphics fresh	41
Chapter 4: Creating Relevant Site Content	43
Portraying Your Client's Company	43
Defining your client's voice	43
Developing your client's look	45
Selling Goods or Services	46
Emphasizing key points	46
Highlighting the product	47
Providing Information	48
Handling large amounts of content	48
Presenting information	49
Including e-learning materials	52
Using Personas to Develop Content	53
Defining your client's customers	54
Delivering what they want	54

Book 11: Site Design..... 55

Chapter 1: Conceptualizing Your Site57

- Deciding What Types of Pages You Need..... 57
 - Considering static Web pages 59
 - Freshening the content with dynamic Web pages..... 59
 - Evaluating multimedia element choices 61
- Choosing the Right Delivery Method 64
 - Knowing when static pages suit your purpose 64
 - Using dynamic pages to manage complex information 65
 - Considering multimedia..... 66
- Handling Printable Materials 67

Chapter 2: Creating Effective Layouts.71

- Content Is Key: Making Sure It’s All Included 72
- Creating Order from Chaos: Consistency Is Your Friend 73
- Including Things That Every Page Needs 75
 - Banner 76
 - Page headline 76
 - Global navigation 76
 - Page title 77
 - Page description and keywords..... 77
 - Page footer 80
- Planning How to Emphasize Important Information..... 81
- Understanding the Basics of Usable and Accessible Design 83
- Creating a Layout Wire Frame 85

Chapter 3: Organizing Your Content91

- Starting with the Skeleton: The Structure 91
- Considering File Management..... 92
- Keeping It Tidy..... 94
- Implementing Information Architecture..... 96
 - Developing section-level information architecture 97
 - Arranging page-level information 98
- Avoiding Content Problems 99

Chapter 4: Ensuring Visual Appeal103

- Colors on the Web 103
 - CMYK color..... 104
 - RGB color 104
 - Hexadecimal colors 104
 - Making nice with established color schemes 105
 - Web-safe colors..... 106
- Getting Familiar with Graphics File Formats..... 108

Using Fonts Effectively on Web Sites	111
Implementing Stock Images and Other Elements	113
Facelifting an Existing Site	115
Choosing a new color scheme	116
Figuring out which graphics to replace	117
Creating a new prototype	117
Applying the new colors to your CSS file	121

Book III: Site Construction 123

Chapter 1: Pulling Together Tools and Materials 125

Matching the Tools to Your Needs	125
Hauling Out the Hardware	126
A computer	126
Computer monitor	127
Backups and storage	128
Printers and scanners	129
Cameras and camcorders	130
Graphics tablet	132
Setting Up Your Web-Design Studio	132
Desperately Seeking Software	133
Researching software before you buy	134
Graphics software	134
Multimedia software	137
Web-design software	140
Browser software	142

Chapter 2: Making a Web Page with HTML 145

Getting Acquainted with the Basic Parts of an HTML Document	145
Starting Your HTML Document Right	147
The document type definition	148
The head tag and what goes in it	148
Now, for some body!	151
The anatomy of a whole page	153
Controlling layouts with CSS	158
Helpful Development Practices	159

Chapter 3: Creating a Web Page Layout with HTML and CSS 161

Understanding How CSS Can Help Your Site (And You)	161
Checking the source	163
Using CSS with other tools	164
Ways of Working with CSS	165
Inline styles	166
Styles embedded in the head of the document	166
External style sheets	167
Commenting your code	168

Creating a CSS Document	168
Setting default selectors	169
Setting class and ID selectors.....	170
Special Effects with CSS.....	175
Styling your links	175
Cool headlines.....	176
Custom padding and margins	177
Custom borders	178

Chapter 4: Introducing Web and Graphics Software.179

Choosing Graphics Software	179
Adobe Photoshop.....	180
Adobe Illustrator.....	180
Other professional-grade graphics software.....	181
Considering Web-Design Software	183
Deciding on Multimedia Software.....	185
Touring Photoshop CS4.....	186
The default workspace.....	186
The work area.....	187
The Advanced button.....	188
Layers and the Layers panel	188
Tools and the tool panel.....	189

Chapter 5: Getting Started with Dreamweaver.201

Exploring the Dreamweaver Interface	201
The Welcome Screen.....	202
The New Document dialog box.....	205
The Files panel	208
The Document window	212
The Preferences dialog box.....	217
The Property inspector (panel).....	218
The CSS Styles panel	219
The Insert panel	220
Creating a Web Site with Dreamweaver	221
Testing Your Pages and Validating the Code.....	225
Publishing Your Web Site with Dreamweaver	227

Chapter 6: Using Dreamweaver: Advanced Techniques229

Creating a Site with Prebuilt Page Designs.....	230
Creating Templates to Ease Site Maintenance.....	234
Making a template.....	234
Including editable regions	235
Creating a page from a template.....	236
Updating a template	236
Exploring the Assets Panel and Library Items	238
Touring the Assets panel	238
Discovering library items.....	240

Adding Functionality with Dreamweaver Behaviors	243
Including the Open Browser Window behavior	246
Editing a behavior via the Behaviors panel.....	250
Creating a pop-up message.....	250
Chapter 7: Creating a Web Page from a Photoshop File	253
Taking a Wire Frame to a Finished Design	253
Finishing the Artwork.....	259
Placing a photo in your banner.....	260
Creating a clipping mask.....	260
Implementing filters, adjustment layers, and blend modes	262
Slicing a Photoshop Document.....	264
Optimizing Graphics for the Web.....	267
Making a Graphic Transparent	270
Chapter 8: Meeting HTML's Powerful Friends	273
Web Technologies Defined.....	273
Vary your content with variables	274
Conditional statements	275
Here we go loop-de-loop	276
Creating functional functions	277
Looking on the Server Side.....	278
ASP/ASP.NET	278
.NET Framework.....	279
JavaScript.....	280
PHP	281
CGI/Perl	282
Checking Out the Client Side.....	284
Java	284
VBScript	285
Chapter 9: Web Site Usability and Accessibility	287
Ensuring Your Site Is Easy to Use.....	287
Helping Users Access Your Site.....	289
Images need alternate content.....	290
Use more than one method to convey your information	291
Pictures are important	292
Orientation tools are helpful	293
Creating Accessible Web Pages	293
Using Dreamweaver to insert accessible elements	293
Using Dreamweaver to check your site's accessibility.....	295
Chapter 10: Publishing the Web Site	299
Look before You Leap: What to Do Before Launch.....	299
Develop a checklist.....	299
Get opinions	302

Going Live	304
Using Dreamweaver to upload content.....	305
Uploading pages with an FTP client	308
What Next? The Launch Is Not the End of the Project	310
Submitting the site to search engines.....	310
Implementing marketing.....	311
Requesting link exchanges	312

***Book IV: Web Graphics* 313**

Chapter 1: Finding Inspiration 315

Finding Fresh Ideas	315
Respecting the Copyrights of Others.....	316
Protecting Your Own Copyrights	318
Working with Colors (Web Safe versus Not-Web Safe).....	319
Creativity Tools — Color Charts and More.....	322
Font/Type Issues on Web Sites	323
Print to Web: Making Your Web Site Work with Existing Materials.....	324
Understanding Image Formats for Web Design	325
Using the GIF image file format	325
Using the JPEG image file format	326
Using the PNG image file format	328

Chapter 2: Bringing Your Vision to Life 329

Introducing Fireworks CS4	329
Vector and raster images.....	329
The PNG file format	331
Fireworks interface mini-tour.....	332
Creating Art with Other Tools	340
Adding Photoshop CS4 to your graphics toolbox.....	340
Getting Images ready for the Web with Photoshop	341
Creating painterly images in Corel Painter.....	342

Chapter 3: Workhorse Graphics 345

Creating Buttons in Fireworks	345
Introducing the Button Editor.....	345
Creating a simple two-state button	346
Creating buttons with pizzazz.....	350
Creating a Navigation Menu	351
Creating a Pop-Up Menu	354
Recycling: Reuse Everything.....	359
Making a reusable graphic template	359
Organizing a site	360
A Doodle to a Working Page: Concept to Completion	361
Mind mapping.....	361
Creating a client mockup	361

Optimizing Artwork in Fireworks	362
Optimizing GIF artwork.....	362
Optimizing JPEG artwork.....	365
Exporting Artwork from Fireworks	366

Chapter 4: Creating Compelling Graphics 367

Creating Banner Graphics	367
Using Photoshop to create a banner.....	367
Including all the important information.....	369
Working with an existing logo.....	370
Create Promotional Graphics	371
Emphasize your point	371
Complement your style.....	373
Creating Fast-Loading, Beautiful Photos	374
Resizing your photos for the Web	374
Using professional optimization techniques.....	377
Using the Photoshop Save for Web & Devices command	377
Creating a Photo Gallery.....	385
Choosing your photos.....	385
Using Dreamweaver to create a photo gallery.....	387

Book V: Multimedia 391

Chapter 1: Incorporating Flash Creations 393

Introducing Flash CS4	393
Touring the interface.....	394
Getting comfortable with Flash.....	400
Setting up your workspace	406
Building Flashy Navigation.....	409
Creating the navigation menu document	409
Creating buttons in Flash.....	410
Assembling your menu.....	414
Adding functionality to the buttons	416
Publishing your file and adding it to your page.....	419
Flash as an Animation Tool.....	420
Create a Flash Presentation	421
Exploring the Many Tricks of Flash.....	422

Chapter 2: Using Sound 423

Exploring Audio Formats for the Web	423
Adding Flash Audio to a Page	424
Creating Flash audio.....	424
Adding sound to a Web page.....	427
Embedding a QuickTime Player in a Page.....	429

Adding the RealMedia Player to a Page 431
 Adding the Windows Media Player to a Web Page..... 433
 Delivering Your Message 436

Chapter 3: Using Web Video. 437

Exploring Web Video Formats 437
 Working with Digital Video..... 438
 Capturing video..... 439
 Encoding video..... 440
 Creating and Encoding Flash Video 442
 Adding Flash Video to a Web Page..... 447
 Adding QuickTime Player to a Page..... 449
 Adding Windows Media Player to a Page..... 451
 Avoiding Digital Video Pitfalls..... 453

Chapter 4: Active Slideshow Pro. 455

About Active Slideshow Pro..... 455
 Installing the Extension 456
 Creating a Slide Show..... 457
 Launching Active Slideshow Pro 458
 Adding slides to your show 461
 Adding text and sound to a slide 465
 Editing a slide show..... 468

***Book VI: Audience Interaction* 471**

Chapter 1: Adding Basic Interactivity with HTML Forms and JavaScript 473

Understanding Interactivity 473
 Getting Ready to Code 474
 Building an HTML Form 476
 Form tags: <form> </form> 477
 Text fields: <input type="text"> 477
 Submit buttons: <input type="submit"> 477
 Creating a simple form in Dreamweaver..... 478
 Testing the form..... 483
 Adding Elements to Your Form 483
 Password fields 483
 Hidden fields..... 484
 Textarea form objects 485
 Multiple choice types (check box and radio buttons)..... 486
 Drop-down lists 491
 Event Handlers and JavaScript 494
 Making Dynamic Pages with Server Side Technology 496

Chapter 2: An Introduction to ASP	499
Understanding ASP	499
Creating ASP Pages and Testing Them on Your Local System	501
Working with IIS	501
Setting up the Web site	502
Defining the site in Dreamweaver	502
Creating ASP Pages in Dreamweaver	506
Using variables	507
Introducing VBScript functions	511
Discovering Cool ASP Tricks	513
Displaying a random image	513
Create a rotating ad banner	515
Chapter 3: Introduction to PHP	519
Defining PHP	519
Fulfilling PHP Server Side Requirements	520
Creating PHP Pages	520
Declaring and Using Variables	521
Working with Conditional Statements	523
Putting conditional statements into action	523
Using comparison operators	525
Using logical operators	526
Repeating Lines of Code, Using Loops	526
Using a while loop	527
Using a for loop	527
Generating a Random Image	528
Chapter 4: MySQL and PHP	531
Creating a Mailing List Form	532
Creating the Database	534
Creating the PHP Code	537
Retrieving Information from a Database	539
Chapter 5: Additional Site Interactivity	543
Engaging Visitors with an Online Journal (Blog)	543
Initiating Online Conversations: Forums and Message Boards	545
Establishing an Online Community: A Wiki	548
Feeding Them Your Content: RSS	550
Using an online feed reader	551
Creating an RSS feed	552
Validating your feed	556
Getting your feed going	558
Branding through E-Mail	560
Keeping Them Up to Date: Newsletters	561

Book VII: E-Commerce 563

Chapter 1: An E-Commerce Primer 565

- Nailing Down E-Commerce Concepts..... 565
 - Establishing an online identity..... 566
 - Researching domain names..... 566
 - Purchasing domain names..... 568
 - Choosing an e-commerce-friendly server 569
 - Planning a user-friendly site 572
 - Brainstorming the site..... 573
- Adding Basic E-Commerce with PayPal..... 575
- Using PayPal Website Payments Pro 578
- Integrating with Credit Card Authorization Packages 579
 - Solving the online payment maze 580
 - Internet fraud: An e-commerce merchant’s worst nightmare..... 581
- SSL — What Is It? 581
 - Using a secure server 582
 - Sharing an SSL certificate 582
- E-Commerce Do’s and Don’ts..... 582

Chapter 2: Building an E-Commerce Site. 585

- Technological Considerations 585
- Usability: Thinking Like a Customer 586
- Legal Considerations: Call a Lawyer 588
 - Cookies that don’t crumble 588
 - Spam, spam, spam, spam!..... 588
- Creating an E-Commerce Site..... 590
 - Setting up the e-mail addresses 590
 - Selling items with a PayPal account 592
- Using a Shopping Cart..... 598
 - Creating a mailing list form 599
 - Forwarding the information to a database 602
 - Test, test, and then test again 603
- Selling Online 604
 - If you build it they won’t care 604
 - Security concerns 605
 - Following through on every sale..... 606

Chapter 3: Maintaining an E-Commerce Site 609

- When Technology Breaks Down..... 609
 - When client-side technology runs amuck..... 609
 - Server-side technology, or code sleight of hand 610

Handling Customers	611
Getting it in writing.....	611
Documenting everything.....	613
Handling e-commerce customers	614
Search Engine Optimization (SEO) and Marketing.....	614
Optimizing the site for search engines	614
Finding sites to link to the e-commerce site.....	616
Danger, Will Robinson! These do not compute!.....	618
Using search engine software	619
Marketing the site to the world.....	620
Figuring Out When to Upgrade	624
Upgrading to a more robust Web hosting package.....	624
Upgrading to another server	625

Book VIII: Site Management..... 627

Chapter 1: Helping a Site Succeed 629

Planning and Incorporating Search Engine Optimization (SEO)	629
SEO defined.....	630
Seeing why SEO is essential.....	630
Understanding the mechanisms: Meta tags, keywords, descriptions, and alt text	630
Improving site searchability.....	632
Hiring the Right Professional Help	636
Deciding between SEO and paid advertising.....	637
Submitting your site to search engines	638
Using a service to submit your site	639
Determining Whether Your SEO Is Working	640
Reading your Web stats	641
Defining Web stat terms	642
Adjusting the site and driving traffic.....	643
Using Reciprocal Links to Boost the Site's Visibility	644
Promoting a Site with Traditional Marketing.....	644
Keeping Them Coming Back	645
Ongoing content development.....	645
Creating content to keep them coming back	646

Chapter 2: Maintaining a Web Site 649

Updating Pages without Destroying Them.....	649
Sharing the Work with Contribute	651
Creating a connection	652
Administering the Web site	656
Sending a connection key	660
Editing Web pages	662

Managing a Site with Dreamweaver	665
Editing new pages	665
Archiving pages.....	668
Tips for Managing Tasks.....	668
Simplifying repetitive tasks with Snippets	669
Taking advantage of your assets	672
Using the Dreamweaver Library	673

Chapter 3: Expanding a Site 679

Use It or Lose It: Should You Add On or Redesign?	679
Adding on to a site.....	680
Redesigning the site from scratch.....	680
Headache-Free Expansion	683
Things to consider when redesigning.....	683
Trauma-free site expansion	687
After Launching the Expanded Site.....	689

Book IX: Case Studies 691

Chapter 1: Blog Site 693

Examining Blog Software and Services	693
b2evolution.....	694
WordPress	697
Exploring other options: Blogger.com.....	706
Creating a Web Site from a Blog	707
Managing spam	707
Extending WordPress.....	708
Photoblogs.....	710

Chapter 2: Podcasting Site 711

About Podcasts.....	711
Creating the Podcast Site	712
Recording Hardware	714
Zoom H2.....	714
Blue Microphones Snowball.....	715
Blue Microphones Snowflake	716
Sound-Editing Software.....	717
Adobe Audition and Sound Booth.....	718
Sony Sound Forge	719
Sony Acid Music Studio (Windows only).....	719

Chapter 3: Photographer/Portfolio Site 721

Defining Goals for the Site	721
-----------------------------------	-----

Designing the Site	722
Creating the navigation menu in Fireworks.....	722
Optimizing images in Fireworks.....	724
Creating thumbnails in Fireworks.....	724
Creating the other assets.....	726
Putting it together in Dreamweaver	726
Creating the portfolio pages.....	727
Showing off without snootiness.....	728

Chapter 4: Foundation Site 731

Meeting with the Client.....	731
Meeting the Client's Needs.....	732
Creating the Assets	733
Putting It All Together in Dreamweaver	735

***Index*** **737**

Introduction

Web sites can be very complex or very simple. When you're building a site, you have many decisions to make, all dependent on the needs of a particular project. This can be overwhelming, but don't fret — you have *Building Web Sites All-in-One For Dummies*, 2nd Edition, to help you. It's so comprehensive and so flexible that it will help you in all aspects of Web design, from the initial planning phases to testing and publishing your masterpiece.

This book can help you take on the role of project manager, graphic designer, developer, or multimedia designer, and it has information about interacting with different specialists on larger or more complex projects. So whether you're undertaking your first Web design project or are a veteran taking on a Web design team, this book is for you.

About This Book

Here are some of the things you can do using this book:

- ◆ Plan your Web site project.
- ◆ Assemble and manage a Web team.
- ◆ Create layouts, graphics, navigation menus, and Web pages from scratch.
- ◆ Optimize graphic elements for your Web pages.
- ◆ Work with HTML and CSS to create and maintain your pages.
- ◆ Create HTML forms.
- ◆ Add multimedia content such as Flash movies and video clips.
- ◆ Integrate e-commerce into your Web site project.
- ◆ Maintain and promote a Web site.
- ◆ Explore server-side and client-side code.

Foolish Assumptions

We have, perhaps foolishly, made a few assumptions about our readers. We expect that you have basic computer skills (either Windows or Mac) and a basic understanding of how to use a browser and the Internet. The authors and publisher of this book assume you're a bright, intelligent person who

wants to learn but doesn't have the time to read a book from cover to cover. We assume you'll find the information you need by perusing the index and then you'll cut to the chase and read that section.

If you don't have any prior knowledge of HTML, CSS, or graphics software, that's okay; we give you the basics here. We do assume that you have more than a passing interest in Web design. In fact, this book is geared for Web designers, or anyone with aspirations of becoming a Web designer. If you don't have any prior experience with Web design or managing Web site projects, that's okay, too. This book starts at the beginning before moving into more advanced topics.

Conventions Used in This Book

By *conventions*, we simply mean a set of rules we've employed in this book to present information to you consistently. When you see a term *italicized*, look for its definition, which we've included so that you know what things mean in the context of Web site construction. Sometimes, we give you information to enter on-screen; in this case, we've **bolded** what you need to type. We've placed Web site addresses and e-mail addresses in `monofont` so that they stand out from regular text. Code appears in its own font, set off from the rest of the text, like this:

```
Never mind the furthermore, the plea is self-defense.
```

When we present complex information with copious amounts of code, we present examples and then dissect the code. Blocks of code are neatly bundled into a listing, which looks like this:

Listing Intro-1: Everything You Wanted to Know about Code, but Were Afraid to Ask

Throughout the book, you'll find icons such as Technical Stuff, Tips, Warnings, and Remember. These little tidbits are cold, hard facts we found out the hard way. We sprinkle this information liberally so you won't fall into the same chuckholes we did, or worse, end up with egg on your face.

What You Don't Have to Read

We've structured this book *modularly* — that is, it's designed so that you can easily find just the information you need — so you don't have to read whatever doesn't pertain to your task at hand. We include sidebars here and there throughout the book that contain interesting information that isn't necessarily integral to the discussion at hand; feel free to skip over these.

You also don't have to read the paragraphs next to the Technical Stuff icons, which parse out uber-techie tidbits (which might or might not be your cup of tea).

How This Book Is Organized

Building Web Sites All-in-One For Dummies, 2nd Edition, is split into nine minibooks. You don't have to read them sequentially, you don't have to look at every minibook, you don't have to review each chapter, and you don't even have to read all the sections in any particular chapter. (Of course, you can if you want to; it's a good read.) The Table of Contents and the Index can help you quickly find whatever information you need. In this section, we briefly describe what each minibook contains.

Book I: Preparations

This minibook covers all the things you need to do before you start actually creating a Web site. The topics of planning, managing your project, flying solo on a project, and assembling and managing a Web design team are all covered here. We've also included information about creating a site that portrays the client and her organization in a flattering light while providing visitors with content that keeps them coming back for more.

Book II: Site Design

Site design is about the nuts and bolts of how a site works and about making the visual aspects work within the technical needs — this minibook helps you make these two worlds come together. In short, if you're looking for information about creating layouts and planning site structure, implementing appropriate design, and collecting and using feedback, you'll want to read this minibook.

Book III: Site Construction

Book III takes you into the hands-on creation of your site. The minibook begins with an overview of the tools and materials necessary for Web design; other chapters introduce you to graphics and Web design software before delving into the details of preparing a site to go live. Book II is about being an architect and designer; Book III is about being the crew that gets the job done.

Book IV: Web Graphics

A Web site without graphics is text, which won't keep a visitor interested for very long. Book IV begins with a section on finding inspiration. After you're inspired, it's off to the races with information on how to create graphics in Photoshop and Fireworks.

Book V: Multimedia

Book V shows you how to add multimedia content to your designs. We start out by showing you how to incorporate Flash elements into your designs. If your client wants music or other joyful noise on her Web site, read Chapter 2 of this minibook. If it's full-motion video your client is after, we show you how to add it in Chapter 3 of this minibook. If your client has copious amounts of images to display in a short amount of space, check out Chapter 4, where we show you how to add a *tres-cool* slide show to a Web site.

Book VI: Audience Interaction

Web sites come in many flavors. If your client has the need to give and receive information via the World Wide Web, this minibook is your Rx for Web interactivity. If your client's site needs forms, see Chapter 1. In the latter chapters of this book we also show you how to incorporate additional interactivity, such as databases and pages that change depending on the needs of the visitors. We end this book with an introduction to other forms of interactivity, such as blogs, forums, and other such delights.

Book VII: E-Commerce

If your client wants to take his local, bricks-and-mortar business worldwide, we show you how to accomplish this feat in Book VII. E-commerce is indeed complex, but we do our best to simplify it for you. First, we explore basic concepts such as credit card packages, secure Web sites, and PayPal. Then, we move on to the technological and legal considerations of an e-commerce Web site. Last but not least, we show you what you need to consider when building and maintaining an e-commerce site.

Book VIII: Site Management

If you build it, they will come. *Not*. The only way to get people to flock to a Web site in droves is to promote it. We begin this book by exploring methods you can use to promote a site and get it recognized by the search engines. Of course, after the site has a steady stream of visitors, you or your client will need to maintain the site. And if the site really catches on, your client will probably need you to revise or redesign the site. We show you how to maintain and expand a Web site in Chapters 2 and 3 of this minibook.

Book IX: Case Studies

The final minibook of this lofty tome is four chapters of case studies. We learn a lot by surfing the Net and dissecting what's good, bad, and downright ugly about what's out there. You can too. To give you an idea of what goes into planning and then creating a site, we explore the needs of four different clients and the resulting Web sites.

Icons Used in This Book

For Dummies books are known for those helpful icons that point you in the direction of really great information. In this section, we briefly describe each icon used in this book.



The Tip icon points out helpful information that is likely to make your job easier.



This icon marks an interesting and useful fact — something that you might want to remember for later use.



The Warning icon highlights lurking danger. With this icon, we're telling you to pay attention and proceed with caution.



When you see this icon, you know that there's techie stuff nearby. If you're not feeling very techie, you can skip this info.

Where to Go from Here

While the book is written so that more experienced Web designers can skip around to the parts they need, novice users probably need to start with Book I, which gives a good foundation of building Web sites, before proceeding to the other books. If you're one of those experienced designers, scour the index for the material you need and then read those sections.

Experience is the best teacher. After immersing yourself in those topics you want to know more about, launch your favorite HTML editor and begin noodling with designs you've previously created, and tweak them, using the information from this book. And if — like the plumber with leaky faucets — your personal Web site was designed around the turn of the century, by all means spiff it up using the techniques we show you.

Book I

Preparations

The 5th Wave

By Rich Tennant



"I can't really explain it, but every time I animate someone swinging a golf club, a little divot of code comes up missing on the home page."

Contents at a Glance

Chapter 1: Planning a Web Project	9
Defining the Audience	9
Setting Goals	12
Creating a Scope Document	12
Preparing to Get Started	13
Selling the Idea	15
Holding a Kick-Off Meeting	15
Revising Your Original Plans: Using Feedback to Improve	16
Preparing to Redesign an Existing Site	19
Chapter 2: Build a Web Team or Go It Alone	21
Flying Solo: Skills You Need to Go It Alone	21
Hiring Members of the Team	27
Handing Off a Project to a Client	32
Chapter 3: Developing the Content	35
Knowing What to Put on Your Site	35
Keeping It Fresh: An Ongoing Process	41
Chapter 4: Creating Relevant Site Content	43
Portraying Your Client's Company	43
Selling Goods or Services	46
Providing Information	48
Using Personas to Develop Content	53

Chapter 1: Planning a Web Project

In This Chapter

- ✓ Clarifying your needs
- ✓ Defining the scope
- ✓ Getting the project started
- ✓ Getting comfortable with the iterative process

Like with any project, planning and gaining support are crucial to your success. Web site projects are sometimes deceptive. Even if it looks as if you can just toss one together without too much fuss, don't be fooled by first impressions because you just can't skimp on the planning process. You need assistance regardless of the size of the project, so make sure you've got everyone you need on board: department managers or IT personnel, for example. Although the players might be different when looking at fully inhouse productions as opposed to a collaboration between inhouse and hired teams (the topic of Chapter 2 of this minibook), the need to research and plan still exists. It's easy to get excited when you start a Web project, but don't let that get you into trouble later.

Something to keep in mind while you work is that Web projects are *iterative*. That is, a Web project requires many cycles of review and revision while you move along. You must develop a flow to move through the project, and you need a clear vision of the project, too — just don't get hung up on working out every single detail beforehand. Being flexible to adjustments while not deviating from the plan is a difficult but important balancing act. In this chapter, we lay out the information that will help you stay balanced.

Defining the Audience

Your Web site has multiple audiences — internal stakeholders and external Web site visitors — and each set has its own needs. You must consider everyone when defining your audience and goals.

Internal stakeholders

Internal stakeholders are the people who work for the company that hired you to create its Web site — managers, public relations people, IT and marketing professionals, customer service reps, salespeople, and so on. All these people have needs to consider when planning and creating the site.

You want their support. And the best way to gain their support is to do a little upfront legwork. As much as possible, talk to key players and ask them what they want from the Web site. Doing this helps you make sure that you address their needs and concerns while you prepare to make your formal presentation. It also ensures that when you make the formal presentation of your project, you can speak to their needs and anticipate their questions. Web builders commonly make the mistakes of not letting other people have input, thus wasting stakeholders' time by presenting information that doesn't help them decide how to support your project.

Some helpful questions to ask before you start your project are

What are your expectations for this Web site?

What do you want the results to be and what sort of functionality do you want?

Do you have an established site or printed materials?

Do you have a corporate brand, logos, and other look-and-feel materials that need to be incorporated?

Will people within your organization help support the site? If so, what are their roles?

Do you have a project point person?

If your client doesn't know what she wants, ask her to spend some time online to peruse existing Web sites. Tell her to look at competitors' sites and also at sites that have similar purposes (informational, e-commerce, or whatever) to hers. Ask the client to make a list of things she likes and doesn't like. Set a time for a follow-up meeting to discuss what she finds. When you review the sites, ask her to explain what she likes or dislikes — and why. Analyzing these other sites will help you and your client develop a better vision of what the client needs.

External audience

Oddly, Web site users are often the most overlooked part of the equation. Getting caught up in all the other details of planning and deciding how the site will support the goals of the organization can easily take you away from considering the real needs of your external audience. Often, the visitors are referred to as a *target audience* and are described in very broad terms. Unfortunately, the discussion of a target audience is generally a short one and not very detailed. Failing to think carefully about the needs of visitors will result in a site that looks great to the site owner but probably won't be what visitors need.

To effectively lead a Web site project, you must be careful to dig a little deeper into what kinds of people make up the target audience as well as what they want; otherwise, the site will try to be everything to everybody instead of being just what it needs to be for that audience. Luckily, certain techniques, including creating profiles, can help you avoid the common pitfalls of designing for your users. *Profiles* (also known as *personas*) are detailed descriptions of your users as individuals. While you're developing your content and design, don't forget that your users are individuals. Generalizing your audience makes it much harder to deliver truly helpful and engaging content. Profiles help you think about your users' needs.

Here are some easy steps to help you create a profile:

1. Start with a general target audience description and then imagine one of those people standing out in the crowd.
2. Create a detailed description of that individual, even giving the person a name or using photos that fit the description to help you envision this person.
3. Choose several more individuals from that target audience crowd.

If your site will service multiple crowds, create several profiles for each crowd. For instance, if you're making a site that helps kids with homework, you need to think of a group of teachers, a group of parents, and a group of kids.

4. Select just a few individuals to represent the group.

Starting with several individuals and then narrowing it down to only a few gives you more information at first, which you can focus on later. If you've really thought about your individuals, you should start to notice patterns that can help you focus on what their needs are.

With profiles in hand, develop a list of how to meet the needs of those individuals. Try to come up with specific ideas of how you will meet their needs. For example, if you're building an e-commerce site for people who are not so technologically savvy, you need a plan for how to help them use your site. You could plan to include an online tutorial and informational minisite that explains to them the process and addresses the concerns they may have. Consider a glossary or other materials that would help your users learn about e-commerce. Doing so can help customers feel more confident about doing business with the company.



Paying careful attention to your users helps you create a site that delivers what they need and expect. You should be able to develop a good idea of what works well for them. Doing this upfront work helps you decide everything from look and feel to voice and functionality. Making sure your site is what your visitors need and want will help your site be a success.

Setting Goals

Each Web site needs a purpose. Even if you're designing a small site about your hobby, you need to have a reason for building the site. You also need to decide what you will use to measure the success of the site. People often talk about a "successful" Web site, but what does that really mean? Is the goal to raise awareness or to increase sales? To cut down on customer service calls or perhaps obtain a large volume of traffic? If a site doesn't have a goal or a definition of success, you can't develop or maintain it. You determine these goals in part by talking with the internal stakeholders of the site (what are their expectations?) and also by determining what it is that the external audience needs. (We discuss these two groups in the preceding section.)

Usually, a site has more than one goal, so make a list and prioritize your goals. Decide which goals and features are must-haves and which are simply nice to have. Concentrate your efforts on the must-haves first and then create a development game plan to add the nice-to-have stuff. The great thing about Web sites is that if you plan them well, adding things can be fairly simple.

Another important reason to have goals is that it helps you set project *milestones* — short-range goals that help you measure your project's progress while keeping everyone on track. When working for a client, having milestones is important because it enables the client to sign off on the progress, showing approval of the work to that point. If the client doesn't approve of the work, you need to get specific feedback about what the concerns are, refer to the original plans for the site, and determine whether the requests fit within the scope of the project. If they do, make the changes and then request a review/sign-off of those changes. This process ensures that there is no confusion about whether a client approves of the work and that the client agrees that the product has been delivered as expected. For more about checking your progress at milestones, see the "Revising Your Original Plans: Using Feedback to Improve" section, later in this chapter.

Creating a Scope Document

Define your Web projects in terms of what features and content you intend to include. Having a general idea of what the site will include isn't enough because everyone has their own vision of that. A well-run project needs good communication right from the start. The success of a project depends on everyone agreeing on what the project includes. This project definition — *scope* — should be clearly written in a *scope document* and distributed to all team members. Any changes to the scope of the project need to be recorded in the scope document and then redistributed to team members.



Creating the scope document involves defining what the project is, but don't forget to also define what a project is *not*. For example, if the site will use Flash but not video, the scope document needs to say so. Define each element clearly. Simply including "Flash element," for example, in a list in the scope document isn't enough. With each element, spell out what the project functionality *is* and *is not* in very definite terms. So, to properly reflect the "Flash element" in your document, you must be specific, like this:

Flash document to include animated bulleted list as provided by client and supporting graph from client's PowerPoint presentation. This Flash element will not include audio or video.

By being specific, you protect yourself and your client from being unpleasantly surprised when you produce the Flash piece.

All projects suffer to some degree from something called *scope creep*, which happens when people start throwing in little extras that weren't part of the original scope. Keep these things to a minimum. Make sure you discuss the impact of the changes on the timeline and budget with the client immediately. If you need to revise a portion of a project in any substantial way, make sure you amend the original scope document, adjust the timelines as necessary, adjust the budget to reflect the major changes, and then get all the key players to sign off. Failing to rein in scope creep can kill your project, ruin the budget, and ultimately damage your business if you are an independent Web designer.

Preparing to Get Started

Your preparation doesn't have to be a giant project. Smaller or less-complicated sites require just a little time — you can accomplish a lot with a couple of hours, a cup of coffee, and a pad of paper. Just sit down and start focusing on the details of the project, such as defining your purpose, coming up with specific ideas, considering budget and timeline, and so on. Of course, you'll need more time and more people on the team if you're planning a large or complicated project — so make that a whole *pot* of coffee (and maybe some donuts, too).

Defining why you're doing the project

This might sound obvious, but you should define why you're going to build or redesign a site in the first place. Companies commonly start a Web project because their colleagues or competitors have Web sites. While "keeping up with the other guy" is a reason, it can't be the only reason. Without a strong message or clear direction, your site can end up being a bland imitation of other sites, which is usually counter-productive. A Web site can be a great tool for an organization if you focus on why you're building it and what can make your site better than everyone else's.

Brainstorming and evaluating your ideas

Like with any type of a project, the first thing you can do after you gather basic information about the needs of your internal stakeholders and external audience is have a brainstorming session. Do this alone or with the core team to get the ideas flowing with minimal complications. You can (and in many cases, should) have additional brainstorming sessions with the team. The important thing at this point is to write down everything that pops to mind.

After your brainstorming session, consider the ideas from that session in a more practical way. Compare what you have with your defined goals and reasons and start discarding things that just don't fit. Again, this part of the Web project is similar to many other types of projects. Web project managers can easily fall into the trap of thinking that because it's a Web project, all the work will be done on a computer, and old-school techniques don't apply. However, resist the urge to fall into that trap. Getting away from the computer and technology can help you focus on the purpose of your project and the content you plan to deliver without the distraction of the computer and technology.

Looking at budget and timelines

Even an inhouse project has a budget and a timeline. These things can change during the course of a project: Sometimes, deadlines can't be met or need to be pushed forward. Time and money, as in any project, are tied.

While you work on your first Web projects, you'll probably find that budgeting time and money is difficult. Projects usually are more complicated than they seem; even small projects need input from multiple individuals, and those individuals will need to agree and collaborate. Although we can't give you a magic formula for calculating how these changes and collaborations will impact your project, keep in mind the following guidelines when you're planning the budget and deadlines:

- ◆ **Clearly establish deadlines up front, specifying what elements team members will deliver on those deadlines and what resources team members will need from stakeholders for the project to continue on track.** Include information about what will happen if stakeholders or clients delay the project. When a client is responsible for delivering materials (such as photos or text) and those materials don't arrive on time, you can't proceed. Standard practice is to add the number of days the materials were late to the timeline. That is, if clients are two days late with text, the deadline pushes out two days.
- ◆ **Whether you're working as a freelance Web designer or as part of an internal team, put everything in writing.** This way, you avoid the "he said/she said" scenario that causes frustration and is counterproductive.

- ◆ **Use a *rush fee*.** This is extra money that you charge if the client wants you to deliver the project faster than originally agreed. You can also use a rush fee when a client asks for a project on extremely short notice — for example, a client calls and asks you to create a minisite in only two days.

Considering collaborations

While you complete your planning, consider the possible reactions to what you're planning. For example, you might encounter opposition from individuals who don't think a Web site is a good thing. On the other hand, some overly enthusiastic people might want to pitch in and help. Try to think about the people who will be impacted by the project and be prepared to address concerns, using all the information you gathered to this point. Thinking of how (or if) you'll collaborate with those who want to help is a good idea, which can help you with the next steps — selling the idea and having a meeting to officially get things rolling.

Selling the Idea

Whether you work in a large, corporate environment or you're a freelancer/design firm, you need to sell your plan to the stakeholders. Remember, *selling* isn't a dirty word. You're not trying to get people to agree to buy something they don't need. In fact, if you've done your preplanning, your project will actually help them solve problems.

Be prepared to address the stakeholders' concerns while you point out how your plans will solve problems for them. Don't avoid discussing the negative points or other impact your project might have. You can gain respect and important feedback if you show that you're open to discussion and knowledgeable enough to know that the project is not all about fun and glamour.

In short, present your idea, answer their questions — and be prepared for their concerns.

Holding a Kick-Off Meeting

Another form of selling the idea, a *kick-off* is a meeting to get all the hands-on people involved. The main purpose for this meeting is to explain the project, set expectations among the members of the team, and give them copies of the scope document so they can review and understand fully what is expected. Additionally, open a discussion among team members, giving them an opportunity for sharing ideas and honing the plan.

When presenting your idea and defining the project, ask your team questions regarding feasibility and capabilities. Also, be prepared for their questions: After all, production people and IT folks need details to do

their jobs correctly. Don't misinterpret their questioning as them "being difficult." Also, try to understand any issues that are raised. Sometimes, features or functionality are possible but just not practical to create or support, so you might need to suggest a compromise. Work with your team to come up with the best solutions.

Make brainstorming ideas a part of the process. Allow everyone to give input about big-picture concepts on features and functionality. However, when the actual work begins, respect people's expertise. Writers should be the ones responsible for the written content; designers create the designs; and developers work with the code. It's great to share ideas, but team members doing each other's jobs is counterproductive. As a project manager, you should establish that collaboration is good, but second-guessing expertise creates friction and generally hurts the finished project. Make sure you hire the right people and then define the roles and build a good environment for teamwork.

The final task for a kick-off meeting is setting the next steps. Make sure all team members understand what they need to do after the meeting. Be clear about what you expect from each team member and give deadlines. A good way to start the project off right is to follow up with an e-mail that includes a summary of the meeting, a list of tasks, and an outline of expectations.

Revising Your Original Plans: Using Feedback to Improve

As a Web project progresses, it moves through a cycle of review, feedback, and revision. Each iteration of the project (hopefully) helps the finished product become better. Establishing project milestones (which we discuss in the earlier section, "Setting Goals") is an important part of the initial planning phases. The milestones provide points along the way when all stakeholders can have a look at the site and give feedback about the progress.

Whether formal or informal, *usability testing* (asking people to play with the site to see what works and what doesn't) at key milestone points can provide useful feedback. To conduct a quick, informal usability test, select some people who are representative of your target audience (preferably, people not involved in the project) and ask them to try to use your site; then have them provide feedback about their experiences. In Book III, Chapters 9 and 10 contain more information about usability testing and getting feedback.



When you're asking others to preview the site — whether they're usability testers or internal stakeholders — make sure you label placeholder text and graphics clearly to let people know that placeholders are not part of the finished project. *Placeholder* items do just that in a layout — they hold the

place while finished text and artwork are being created. A large, red, “For Placement Only” statement across a graphic often helps people stay on track while they review a project in progress.

Make sure you show your work to stakeholders whenever you make significant progress or hit a project milestone. They need to know that you’re staying on track. Also, keep in mind that some of your stakeholders might not be able to envision the finished product. When you notice someone getting bogged down on a temporary item (such as a placeholder graphic), thank them for their feedback and then try to redirect their attention to items you do need feedback on.

Receiving feedback

If you’re looking for a particular type of feedback, make sure you ask specific questions that prompt users to comment on the elements or issues you want to focus on. In general, it’s not very productive to just send out a link with a note that says, “What do you think?” You’ll get vague responses like, “Looks good!,” which is great for a final okay before launching a project but is not so good when you’re midproject and looking for something more solid. In some cases, you might even want to direct their attention to a particular piece of functionality, such as the subnavigation or a new Flash presentation, and ask them to comment specifically about just that piece. Here are some tips that can help you get the information you need:

- ◆ **Don’t ask for general feedback unless that’s what you really want.** The best way to get a group of random comments and personal opinions (“I like it”) is to just send a link without any explanation, or with a vague direction (“Check this out”).
- ◆ **Make sure you ask for specific feedback from individuals based on their expertise.** Any Web project involves many details and many different disciplines working together. Make sure you have experts to help keep you on track. In other words, ask writers to help you proofread your content; trust designers to make sure your colors are working for you. Rely on experts you trust for detailed feedback on details specific to what they know.
- ◆ **Never assume that a person has nothing useful to contribute.** Although the finer details should be picked over by an expert, a fresh set of eyes is very helpful when looking at the project as a whole. Remember, your actual visitors don’t have inside knowledge or expertise and will also be looking at your site from a fresh perspective. For example, we’ve even gotten great feedback from an 8-year-old child about some icons that weren’t working. You never know who will have a useful tip.
- ◆ **Include a list of what’s new since the last time you sent out a link for review.** Expecting people to play compare-and-contrast to figure out what you’ve been up to is neither polite nor productive. Keep in mind that most of your usability testers and sources of feedback are trying

to look at your project and return comments between working on their own projects. They won't take the time to help you if you don't take the time to direct their attention to the important issues.

- ◆ **Make sure you don't ask for feedback if you can't use it.** If you know that you're locked into a particular piece of functionality or presentation, don't ask people to comment on whether it should be there. It wastes their time, and they might not want to help you the next time. Let people know up front about situations that are beyond your control. For instance, if you must display a particular logo in a specific place, include that information in your note requesting feedback.
- ◆ **Ask open-ended questions.** Try to come up with questions that will make people interact with your site and really think about what they are experiencing. You need honest input from people even if it's not a bunch of compliments. If you collect useful information and act on it, you will get plenty of compliments when you launch a great Web site.
- ◆ **Thank reviewers for their input.** Make sure you thank them for their time because you'll need to call on them again as your project progresses. Keep them interested in helping you. It's easy to forget this little detail when you're wrapped up in your project, but people want to know that their time was well spent. Make sure you send out a follow-up after you collect feedback, including a summary of the feedback and what you intend to do as a result of the comments given.

Giving feedback

Giving feedback can be trickier than getting it, so follow these pointers that can help you give feedback without stepping on any toes:

- ◆ **Take your time.** When you're evaluating a project to give feedback, take your time and look at the site. Your feedback isn't helpful if you immediately start reacting without taking a few moments to look at it and consider what you're going to say.
- ◆ **Stay polite, and don't get personal in a negative way.** Being polite goes a long way when giving feedback. People often forget that someone has put a lot of time and effort into her work, and no one likes to be criticized. Make sure that when you give feedback, you take that into consideration. Blurting out comments shuts down communications pretty quickly. Ultimately, it's the project that suffers for it.
- ◆ **Balance positive and negative comments.** Launching into a laundry list of everything that is wrong with the site is a bad idea. Remember, a human being did the work — not a machine. The best way to have your suggestions ignored is to sound like you're launching a nitpicky attack. Try to balance your negative comments with positive ones. For instance, instead of saying, "I don't like where the logo is, it's crunched up in the corner," try, "I see that you've put a lot of work into this. It's looking good, but I think I'd like to see the logo with a little more space

around it. It seems a bit crowded.” The second approach takes a few seconds longer but helps build and maintain a good working relationship. Web projects take a long time and can be difficult and frustrating. To avoid creating problems, take a few moments to consider delivery of comments.

- ◆ **If something doesn’t look right, ask questions.** Web sites go through a lot of changes throughout the process. If you think something looks wrong, ask what’s going on. Sometimes, there is a good explanation for why something looks strange. For example, if the logo is missing, don’t just say, “The logo is missing.” Instead, try something like, “I noticed the logo is missing, why is that?” It could be an oversight; it could be that a new logo is being developed. Again, delivery of critical comments makes the difference between a healthy collaboration and a confrontation.
- ◆ **Keep your feedback mostly objective.** Remember that the project isn’t your personal, artistic statement. If you don’t like the color but the colors have already been decided, accept the decision and move on to other issues. It’s okay to have some personal reactions to the site and comment on them, but don’t be offended if you’re overruled. Everyone has something to contribute, but not every idea can be included — or the site will look like a crazy quilt!
- ◆ **When giving feedback on an interactive piece, be specific!** It’s not helpful to look at the functionality of a piece and respond to the developers with, “It’s broken.” Designers and developers that are working on interactive pieces need specific information about what went wrong. They need to know what you did (for example, “I clicked the Shop Now button”), what you expected to have happen (“I thought it would take me to the shopping cart page”), and what actually happened (“I got a page that said, ‘404 error – Page Not Found’”). This tells the developer or designer exactly what to look at. “It’s broken” doesn’t tell them anything. If you encounter an error message or error code, tell the designer/developer what it is — *specifically* — and what action you took right before it occurred. The more information you give, the better. If you don’t give specific information up front, you can count on playing a game of 50 questions later as designers and developers try to wrestle the details from you.

Preparing to Redesign an Existing Site

Most of the preparations that you need to do for a new site also apply to an existing site. Some differences that you should take into consideration as you prepare to redesign a site include these:

- ◆ **Evaluate your current Web site.** The first task is to look at the current site and evaluate how well it aligns with your needs. Look at the content, functionality, and look and feel as separate elements. Take each aspect

of your site into consideration — technologies used, coding techniques, site structure, colors, style of writing, and so on. List them all and rate them based on whether they can be used on the new site.

- ◆ **Have experts look at what you have.** Large or complex sites need to be evaluated by selected experts or consultants. In particular, coding and technologies need to be evaluated to make sure that you don't reuse old, outdated technologies instead of using a redesign as an opportunity to make important upgrades.



Don't get caught up in the "newer is always better" line of thinking because it isn't. You know the old saying — "If it isn't broken, don't fix it." Sometimes that's the best course to take. Make sure you weigh the pros and cons of keeping or replacing code carefully before you dive into a big project.

- ◆ **Include all stakeholders in the initial evaluation process.** This can help you verify that the information on your site is current and accurate. A site redesign is a good opportunity to involve all interested parties in looking at and updating materials that have been posted for a while. It's common for certain types of information to be posted to a site and then forgotten. Be sure to look at contact information and directions pages because they often harbor out-of-date information. Make sure you include appropriate professionals from the client organization in the evaluation. For example, sites that deal with financial, legal, or medical information will need to be reviewed for compliance with laws specific to those industries (Sarbanes-Oxley, HIPAA, ADA, and so on). It is good practice to ask your client about possible compliance issues and have them identify individuals within their organization that will be responsible for reviewing site content in that context.
- ◆ **Check the front-end code.** Look at the code that handles the display of your interface — HTML and Cascading Style Sheets (CSS) (if your site has them). These technologies have undergone many rapid changes over the past few years, and many sites could benefit from recoding the pages. In Book III, Chapters 2 and 3 have information about current HTML and CSS coding.
- ◆ **Gather the data and make decisions.** When the analysis and evaluation are done and you have collected all the feedback from interested parties and *content owners* (people responsible for the content of part or all of a site), start the planning process. Compare what you have with what you need your site to be and then decide what parts of the current site can be used as is, repurposed and used, or thrown out. When you're deciding what to do, keep in mind that it's often better to put a little more effort into recoding or reworking an existing item than it is to roll a cumbersome or badly developed piece of functionality into a new site. One of your project goals is to make the site more efficient. With the analysis in hand, you're ready to start working on meeting with stakeholders and your team to plan your approach.

Chapter 2: Build a Web Team or Go It Alone

In This Chapter

- ✓ **Evaluating what you need to do it yourself**
- ✓ **Knowing who can help**
- ✓ **Giving the reins to the client**

Web design projects require a blend of skills. The unique blend of creative and technological requirements means that an aspect of Web design and development draws on skills you've used in other positions — or on skills you don't necessarily consider your strengths. The most effective Web professionals are those who know how to leverage their own personal skills and talents and find others who can fill in any skill gaps. Specializing in what you're good at and aligning yourself with others who share your work philosophies enables you to build long-term relationships with those who can work with you to get the job done — whether you go it alone or are part of a team. This chapter covers both scenarios to help you decide which route is the best for you.

Flying Solo: Skills You Need to Go It Alone

You need many skills to single-handedly undertake a Web design project. Some of these skills are tangible; for example, you must know some HTML and CSS (Cascading Style Sheet) skills as well as how to effectively use Photoshop and other graphics programs. Other skills are intangible: You must be able to provide good service while managing the project and your time and keeping the budget under control. This section lists and describes some of the necessary skills you need to fly solo on a Web-design project.

Managing the project

If you're doing most or all of the work yourself, you must be able to work on several aspects of the project simultaneously — requiring your project, time, and money-management skills — in addition to communicating all that to your client. You can make these tasks easier by

- ◆ **Keeping notes:** Get a notebook and write down what you've done, why you have done it, and what you'll do next.

- ◆ **Making a special e-mail folder for project-related e-mails:** If you can, set up your e-mail to direct all mail pertaining to the project to that folder. (Many e-mail programs, including Microsoft Outlook, Entourage, and Lotus Notes, have this capability.)
- ◆ **Drawing up a budget:** Your budget should include your fees for doing the work (Hourly Rate x Time = Cost of Work), fees for any contractors you hire, and also fees for project management. You should also include fees for extra services (say, image scanning or writing content) if you want to make those available to the client. Another possible thing to add is special software or equipment. If your client *requires* that you buy something, build the cost of it into the budget.
- ◆ **Establishing a timeline:** Clients often don't understand how much work and time a Web project requires — they just know that you make it look simple and easy. Developing a quality site isn't quick, but creating a to-do list for each week (or day) and also marking deadlines on a calendar helps you track what you need to do and when.
- ◆ **Devising a troubleshooting plan:** Technological issues are inevitable. For instance, multimedia elements sometimes don't download fast enough, certain functions don't work as expected, or layouts have CSS issues. These sorts of problems can take some time to troubleshoot and fix. Your plan for the troubleshooting process should include staying task oriented and *not* participating in finger-pointing. During troubleshooting, work with the server administrators and others who are involved with the project to find a solution. Communicate with your client; explain the problem and what you're doing to fix it. Stay calm and confident so that your client feels reassured that you are in control and dealing fairly with them. If you need to bring in help, tell your client who you're bringing in and why.



The bottom line here is that if you don't figure out how much time you have to complete the project and how much money is in the budget *very early in the process* — and communicate that to your client — you could end up wasting a lot of time and energy planning a project only to find out that the client is unrealistic in her expectations.

Serving your customer

Part of a Web project manager's job is customer service. Providing good customer service can help ensure that your clients are happy, and that can help you build a solid reputation. The following list describes four important aspects of customer service:

- ◆ **Communicate often and minimize technical jargon.** You must communicate often about the status of the project so that your client knows what's going on. Communicating with less-than-Web-savvy clients,

however, can be a little awkward. Don't talk down to them, but avoid using a lot of jargon (which can make them feel stupid). Try to ease into the techno-talk gently unless you're sure they speak geek, too.

- ◆ **Stay professional.** Web jobs can be a lot of fun for everyone if they're well run and everyone has a good attitude. Unfortunately, sometimes you won't mesh well with a client. If that happens, you must keep a professional attitude, do the work, treat the client with respect, and just suffer through it. That's business. However, in rare situations — for instance, if a client becomes abusive — you might find it impossible to continue working with that client. In that case, you must decide how to wrap things up with the client; you can either finish the job or hand it off to another designer. Either way, you must carefully explain to the client that they would be better off working with someone else. For those rare occasions when a working relationship goes sour, be sure that your contract allows you to get out of an abusive situation.
- ◆ **Know when to say “no” to a project.** Accepting every job that comes your way might seem like a good idea. It isn't. Some clients don't have the money or game plan in place to make it worth your time to work with them. If you waste time on someone who can't make a commitment, you could be missing out on a client who is ready and able to start a project. If a client isn't ready right now, stay in touch with him. He will appreciate your interest in his project and might just give you the job when it's time.
- ◆ **Take only projects that you can execute well.** Your portfolio and reputation are important. Delivering a good product is a great thing, and your client will recommend you to their colleagues — that's free advertising. Delivering a bad product can have the opposite effect, though, because you might lose that client and any prospective clients who ask them for advice on hiring a Web designer. This doesn't mean that you should never take a project unless you can do every part of it. If a project has some components you can't do on your own, call in a specialist — just make sure you let the client know you're working with a team. The fewer surprises to your client, the better off you'll be.

Dealing with HTML, CSS, and other scripting

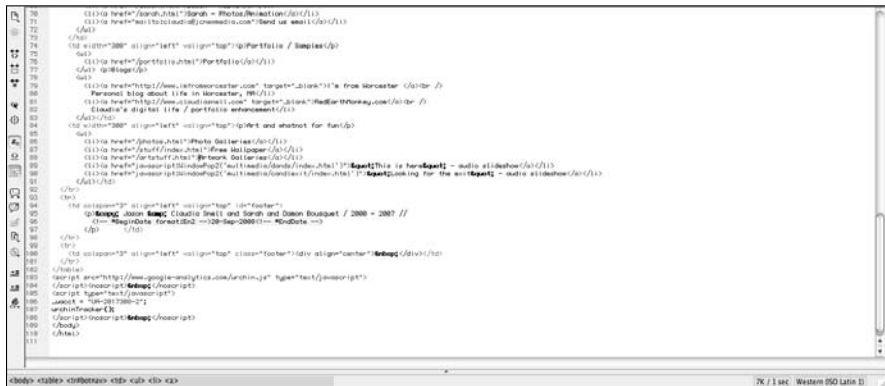
Web pages are made of code, so no matter how you look at it, you can't avoid code. Luckily, basic Web code — HTML and CSS — is fairly easy to learn. It might seem complicated at first, but with practice and patience, you'll be hand-coding pages pretty quickly. Many tools can help you generate code, but you still have to understand the code because sometimes you have to roll up your sleeves and get in there.

Just bear in mind the following.

- ◆ Even the most sophisticated software package is still just software and can make mistakes.
- ◆ Many advanced techniques require a deeper understanding of the underlying code and how it works.
- ◆ Taking on an existing site to redesign or maintain requires that you know how to analyze the code that's there. If you can't, you might find that you can't work with the site.

Book III covers coding by hand, using Dreamweaver, and techniques to help you take advantage of both. Figure 2-1 shows an example of HTML code. The first few times you look at the code, it might seem confusing, but it will quickly become familiar.

Figure 2-1:
This is an
example of
HTML code.



```
70 <!--#include virtual="/about.htm" -->
71 </!--#include virtual="/includes/footer.htm" -->
72 </!--#include virtual="/includes/header.htm" -->
73 </!--#include virtual="/includes/navigation.htm" -->
74 </!--#include virtual="/includes/sidebar.htm" -->
75 </!--#include virtual="/includes/content.htm" -->
76 </!--#include virtual="/includes/footer.htm" -->
77 </!--#include virtual="/includes/header.htm" -->
78 </!--#include virtual="/includes/navigation.htm" -->
79 </!--#include virtual="/includes/sidebar.htm" -->
80 </!--#include virtual="/includes/content.htm" -->
81 </!--#include virtual="/includes/footer.htm" -->
82 </!--#include virtual="/includes/header.htm" -->
83 </!--#include virtual="/includes/navigation.htm" -->
84 </!--#include virtual="/includes/sidebar.htm" -->
85 </!--#include virtual="/includes/content.htm" -->
86 </!--#include virtual="/includes/footer.htm" -->
87 </!--#include virtual="/includes/header.htm" -->
88 </!--#include virtual="/includes/navigation.htm" -->
89 </!--#include virtual="/includes/sidebar.htm" -->
90 </!--#include virtual="/includes/content.htm" -->
91 </!--#include virtual="/includes/footer.htm" -->
92 </!--#include virtual="/includes/header.htm" -->
93 </!--#include virtual="/includes/navigation.htm" -->
94 </!--#include virtual="/includes/sidebar.htm" -->
95 </!--#include virtual="/includes/content.htm" -->
96 </!--#include virtual="/includes/footer.htm" -->
97 </!--#include virtual="/includes/header.htm" -->
98 </!--#include virtual="/includes/navigation.htm" -->
99 </!--#include virtual="/includes/sidebar.htm" -->
100 </!--#include virtual="/includes/content.htm" -->
101 </!--#include virtual="/includes/footer.htm" -->
102 </!--#include virtual="/includes/header.htm" -->
103 </!--#include virtual="/includes/navigation.htm" -->
104 </!--#include virtual="/includes/sidebar.htm" -->
105 </!--#include virtual="/includes/content.htm" -->
106 </!--#include virtual="/includes/footer.htm" -->
107 </!--#include virtual="/includes/header.htm" -->
108 </!--#include virtual="/includes/navigation.htm" -->
109 </!--#include virtual="/includes/sidebar.htm" -->
110 </!--#include virtual="/includes/content.htm" -->
111 </!--#include virtual="/includes/footer.htm" -->
```

Using Photoshop, Fireworks, and other graphics applications

If you're going to do design work, you need some design skills. In addition to understanding how to use color, fonts, and images to support your content and how to use layout effectively, you need to know some basics in a variety of graphics programs, including these three:

- ◆ **Adobe Photoshop CS4:** This is a great tool for doing practically everything you need to do with Web graphics. Photoshop is the industry standard for working with graphics. With Photoshop, you can work with photos and create supporting graphics (such as banners and buttons) and any other graphics you need. If you want to pursue a professional career in design, you need Photoshop skills. See Book III, Chapter 4 for a brief introduction to the Photoshop CS4 interface and Tools panel.
- ◆ **Fireworks:** Use this Web graphics creation tool to manipulate photos and create other graphics. Its strength is in being tightly integrated with Dreamweaver and Flash. (You can launch Fireworks easily from within

either program via a handy icon.) Fireworks also has great optimization tools and a helpful and easy-to-use batch-processing tool (which is good for resizing a large number of photos at once, among other things).

- ◆ **Adobe Illustrator:** With Illustrator, you can create and edit vector graphics. A *vector graphic* is one that is made up of mathematical information defining points and lines that make up shapes. A discussion of bitmap versus vector graphics is in Book III, Chapter 4. Vectors are great for building graphics with hard lines, such as logos.

Note: You have to convert vectors into bitmaps to use them on a Web site. Photoshop and Fireworks are primarily bitmap editing programs; bitmaps are the choice for photographs. Figure 2-2 shows a vector graphic (on the left) and a bitmap graphic side by side. Notice the jagged edges created by pixilation in the bitmap.

Information about how to create, use, and prepare graphics for use on a site is included in Book IV.

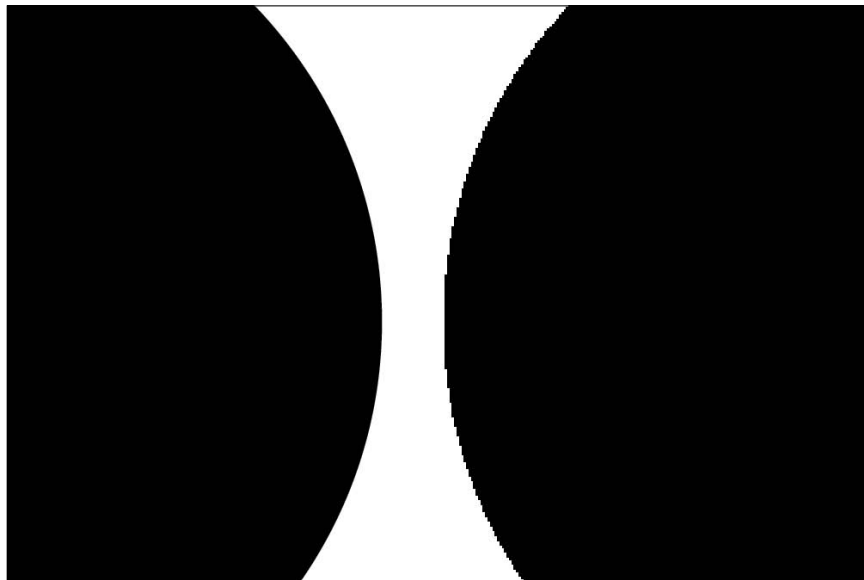


Figure 2-2:
A vector
graphic
(left) and
a bitmap
graphic
(right).

Developing content

Good Web writing skills are essential for creating a successful Web site because a site without good content isn't useful to anyone. Whoever creates the site's content — whether you or a partner — must understand how people use Web sites and why they go to the site in the first place.

Good writing skills also come in handy when you're preparing presentations and proposals. Almost all projects require some degree of writing skill for preparing contracts, proposals, scope documents, and other communications. Short paragraphs and bullet points help readers find what they need quickly.

Considering basic computer and Internet skills

Not that we want to state the obvious, but you need some computer skills if you are going to design Web sites.

- ◆ **Word processing:** Whether you're using Microsoft Office, iWork (Mac only), or another word-processing program, the ability to prepare documents efficiently ensures that you can prepare contracts and proposals and keep track of the business side of things.
- ◆ **E-mail/IM (instant messaging):** Communicating with clients is important for your success. With e-mail and instant messenger, staying in contact with clients has never been easier.
- ◆ **Browsers:** Having some familiarity with and testing your site on the popular Web browsers (Internet Explorer, Safari, and Firefox) ensures that the site will work no matter how visitors are viewing it.
- ◆ **Windows/Macintosh platforms:** Developing skills on both Macintosh and Windows computers broadens your range and eases testing of your Web pages on both platforms. Professional Web designers work to make their sites function under a wide range of conditions. Book III has more details about things you need to look for and how to deal with platform issues.

Incorporating multimedia in your project

If you plan to work with Flash, audio, or video, you need some multimedia skills. You have a choice of many types of multimedia, and what you use depends on what you're trying to accomplish. Book V has information about technologies and techniques for using multimedia elements in your project.

Handling a solo project

If you decide to take on a project by yourself, here's a brief list that can help keep you on the right track:

- ◆ **Prioritize your tasks and develop a workflow.** Some parts of the project are more enjoyable than others, but you still have to complete them all. After you do a few projects, you start to develop a workflow that enables you to work effectively through all the parts of the project, even the ones that aren't as enjoyable. Prioritizing your tasks and breaking up creative and analytical tasks (so you don't burn out on one aspect of the project) can help you meet your deadlines.

- ◆ **Establish a workflow with your client.** Do this *at the beginning* of the project. Working with one contact person can help reduce misunderstandings.
- ◆ **Notify your client of personnel changes.** If you need to bring in some help, make sure you let the client know.
- ◆ **Treat your home office like a “real” office.** If you’re freelancing from your home, make sure you have a good workspace with all the equipment you need. Treat it like a regular job. Make regular hours for yourself. It’s also a good idea to save some days strictly for production and others for meeting days.
- ◆ **Network and market yourself.** Build time into your schedule to look for your next project. When you’re writing proposals and discussing projects with clients, don’t forget to build “lost” days into the timeline. *Lost* days are those spent going to meetings or other events. If you know that you’ll need 40 hours to complete a job, don’t tell the client that you can do it in five business days. You won’t have it done. That 40 hours is how long you’ll spend on the project *specifically*; however, you’ll have other tasks that take up your time. A “40-hour-job” can be more like a month-long project.

Hiring Members of the Team

You might hire help for your projects for many different reasons. Some projects might have components that you don’t have time to complete yourself. The client might need things that are beyond the scope of your capabilities. Most folks are good at some aspects of the work but not so good at others. Maybe you’re excellent at developing code, but colors and design issues mystify you. Individuals who are great at everything aren’t the norm. Even if you’re one of those who can do it all, working with a team is often more cost and time effective.

An effective way of working is to establish relationships with individuals or companies that provide the services you need. After you find people you can work with, you can quickly build project teams that are tailored to the needs of your client. Your “regulars” can develop a smooth workflow. You will also expand the size and scope of projects you take on.

Note: Job titles in the Web-design industry vary greatly and even overlap. For instance, a Web/new media designer and project manager might both be dubbed Producer. When interviewing people, ask them what their actual experience is and don’t rely solely on their job title. Ask them whether they’ve worked on teams and whether they like a team environment. Assembling people who are team players helps ensure that the team will collaborate.

Project manager

A *project manager* for a Web project — like the project manager for any project — keeps everyone on the same path and makes sure issues are dealt with as quickly and smoothly as possible. The project manager should be organized and detail-oriented enough to responsibly do the following: Make sure members of the team are hitting deadlines, keep track of hours spent on the project, and inform team members of situations that might affect them.

A Web project manager needs to know about the workflow and time traps that can accompany the type of project you're planning. Experience working with creative individuals is also important because a project manager must understand each individual's needs concerning workflow and how materials are to be delivered (both to and from team members). The project manager also has to estimate the time needed to complete different types of projects. Familiarity with industry standards and terms can help the project manager when dealing with professional designers. In some environments, the Web project manager is sometimes referred to as a *producer*.

Web designer/new media designer

A true Web designer knows how to hand-code HTML and CSS and might know some other scripting technologies, such as JavaScript. Web designers are responsible for the visual design of the interface, creation of graphics, and *optimization* (resizing and compressing them for Web use) of photos. (Book III, Chapter 7 has more information about optimizing graphics.)



Watch out for Web designers that “don't do” code. Web designers who downplay the importance of strong HTML and CSS skills aren't really Web designers. They're graphic designers who make graphics that coders can use to create Web pages. It will save you time, money, and aggravation if you shop around for someone who can take care of the whole job, as opposed to someone who needs other members of the team to complete the tasks of creating the interfaces.

Web developer/programmer

Web developers and programmers design and create the back-end systems that make your site do more than just deliver static information. The term *back-end systems* refers to databases and programming that are stored on the server and are used by the Web site. If you need or want your Web site to interact with your users, you need databases and the supporting programming to be in place.

Like with Web designers, the importance of a good developer is often misunderstood. The availability of point-and-click development tools gives some people the impression that it's simple and easy to create databases and Web pages that will work with them. These easy-to-use tools are generally fairly

limited in what they can do, though. If the site will have a lot of people interacting with it or if people will be performing complicated functions on the site, hire developers to build your system for you. If you want to add shopping functionality, you need developers who understand how to make a safe, secure, and easy-to-use site, or users won't buy the advertised products.

Point-and-click tools should not be confused with the open source content management systems (CMSes) that have evolved in recent years. Some of the most popular open source solutions include Joomla, Drupal, WordPress, and XOOPS. The full functionality available through these systems varies and is beyond the scope of this book. However, they are often a good option to consider. Additional information about each is available on its respective Web site.

CMSes all come with robust features and can be used to get a dynamic site up and running very rapidly. Still, we recommend working with a developer who is familiar with working with these systems and the development environment (CSS, PHP, and MySQL, all of which are platform independent). To get the best experience from an open source CMS, your developer needs to understand the strengths and weaknesses of these systems and be able to manage the customization and technical support involved.

Content developer/writer

A good content developer or Web writer understands that writing for the Web is different than writing for other media because visitors interact with Web sites differently than they do with traditional, printed materials.

When looking for Web writers, you want to hire someone who can deliver the following:

- ◆ **Easy-to-understand information:** Look for a writer who can deliver concise text that has a good tone for the Web. Most successful Web sites have a more conversational tone.
- ◆ **A good call to action:** A good Web writer can get a site's users to act. Every site has a goal, but without a good call to action, users might not perform the tasks you want them to while on the site. A good call to action is irresistible. A savvy Web writer knows how to craft the content so it drives people to the pages and actions that you want them to see and do.
- ◆ **Organization:** Content developers must understand how to create a flow of information and leverage the nature of the Web to provide users and site owners with the best results.
- ◆ **Search engine optimization (SEO):** Content developers must know how to create text and meta tags that enable users to find the site through search engines. More information about meta tags is in Book III, Chapter 2. Book VIII, Chapter 1 covers SEO.

Webmaster/host

Hosting is a service usually provided by a third party. Web hosts own and care for Web servers, which are computers with server software installed and that are connected to the Internet. For a fee, the owner of the server — the *host* — allows individuals to copy their Web site files to the Web server so that they're accessible to everyone with an Internet connection.

In addition to owning servers and renting space, Web hosts take care of the servers. They might also provide additional software that tracks the number of visitors your site has. Many Web hosts also offer *URL* purchasing services. A URL is a Web site address — for example, `www.yourname.com`. Most Web hosts sell a complete package, including space for your site, e-mail, *traffic reports* (number of visitors to your site), and URL purchasing.

If your site will include multimedia or dynamic content, consult with your developers and designers before you find and commit to a host. The server software must be compatible with the technologies the team will use to build your site. Many developers and designers will assist their clients with hosting choices or actually offer the services directly or through partnerships.

Other professional help

You should consult with other professionals as you start to work on Web site projects. Designing Web sites is just like any other business, and it's important to get some help so that everything runs smoothly. We suggest that you contact the following types of professionals:

- ◆ **Lawyer:** If you plan to work for clients, you need a lawyer. Make sure you hire someone who specializes in technology and creative industries. A knowledgeable lawyer can prepare contracts that spell out copyrights, address deliverables, and specify timelines, making sure everyone gets a fair deal. You need an intellectual property specialist.

A good contract makes sure that everyone understands the scope of the project, the responsibilities of the parties, and when the project will be delivered. It also spells out how the fees for services work (an hourly rate or a flat fee), which everyone needs to understand and agree upon. Also make sure your contract specifies whether the client incurs the cost if they request or require you to purchase stock photography, extra software, or fonts.

- ◆ **Accountant:** Like with any business, you need an accountant. Seek the advice of an accountant before you start to accept fees or hire others to work for you. She can advise you how to set up your business so that everything runs smoothly.

Managing the team

After you assess the needs of a project and call in the necessary team members, have a kick-off meeting. Chapter 1 of this minibook covers kick-off meetings and how to run them successfully. In short, everyone needs to get together and discuss the project, timelines, expectations, and next steps. These two steps are particularly important in keeping the project moving smoothly:

- ◆ **Establish the workflow during the planning phase.** This is especially important if you're working with more than one service provider. Everyone needs to know how to communicate with one another. Will you be using Instant Messenger to stay in touch? Make sure everyone has contact information for each other. Find out what the daily schedules will be to facilitate communications.
- ◆ **Set up weekly production meetings.** To facilitate the meeting, the project manager should prepare a job grid (generally just in Excel) specifying all the tasks, who is responsible for what, expected delivery dates, status, priority, and any contingencies. All team members should update the team on their progress, any issues, and next steps. Make sure that you get good feedback from everyone — statements like, “I'm working on it,” aren't really helpful. At the very least, find out when each individual expects to finish assigned tasks. After the meeting, send out a follow-up e-mail that outlines what was agreed. Include a new job grid reflecting progress and next steps.

Giving feedback that helps

Web projects have a lot of details to be taken care of. Don't forget to establish a process of asking for and receiving feedback. The project manager should inform members of the team that he is going to send materials to the client for review and await confirmation that everyone is ready for the client to see the work. If some pieces won't be ready on time, don't hold up a scheduled review. Inform the client about the status as soon as you're aware of an issue. Proceed with the scheduled review and be prepared with adjusted timelines.

Establish a contact person within the client organization and communicate with that person only. It might sound unfriendly, but it isn't. Having only one contact person ensures that there won't be confusion as multiple people give feedback. The contact person should be responsible for asking people in their organization for feedback, getting *signoffs* (formal acceptance of the work as complete), obtaining materials, and communicating with you or your project manager. The project manager is responsible for communicating with the team, presenting materials to the client for review, and making sure the project flows smoothly.

For more information about communicating feedback with the team (giving and receiving), see Chapter 1 of this minibook.

Keeping the team on track

One of the hardest things to control is *scope creep*. This is what happens when someone — a client or development team — decides to add “little extras” to make the project better. It is the job of the project manager to keep track of those little extras and make sure that they don’t add up to a whole bunch of extra functionality that wasn’t in the original agreement or budget.

If the client asks for things outside the scope of the project, you have two options for moving forward:

- ◆ **Incorporate the extra request into the current plan.** In this case, you have to tell the client that the extra request is beyond the scope of the agreed project and that amending the current scope document (and timeline and budget) is necessary.
- ◆ **Discuss the additional functionality as a future project.** If the client agrees to hold off on the new idea for later, you can proceed with the project as planned. The good news is that you now have a future job already lined up.

In either case, let the team know about the requests so they can adjust accordingly.



If your client decides to amend the scope of the current project, you must prepare a new scope document, timeline, and budget. You and the client will have to agree to and sign off on it.

Handing Off a Project to a Client

Whether you complete the whole project on your own or with a team, you might need to hand off the project to someone else. When you’re planning a project, you need to think about the ongoing maintenance of the Web site and either include a maintenance agreement in your proposal or outline how you’ll hand off the project.

A *maintenance agreement* should outline how much it will cost to maintain the site and what services you’ll provide. Clients must know that requesting large additions to the site will require a new proposal, scope document, and contract.

If you're planning to hand off the Web site to the client, you and the client must agree on the following:

- ◆ **What will you hand off?** If you're turning over development materials, how much and in what format?

The choice of what to do with production graphics is up to you. Some designers keep their original, editable versions of their graphics. Others hand over all the files. Whichever you decide, make sure that the client understands what they're going to get and how they'll get it.

Whatever you decide as far as the deliverable materials to the client, you need to make sure you keep copies for your own records. Burn the files to discs, collect the site notes, and gather the electronic documents that you've used (e-mails, word-processing documents, contracts, invoices, and so on). Put all the materials together and keep them for your records. Sometimes clients that take on a project come back to you for follow-up work.

- ◆ **How much transitional support will you give at handoff?** You might want to offer some training if the client doesn't have inhouse staff with skills to maintain the site. Make sure you figure any training or transitional support work into your budget.
- ◆ **How will you transport the material to the client?** It's standard to deliver the site itself to the client's Web server via FTP (File Transfer Protocol). This technique is covered in Books III and IV. Or you might deliver the site files — and other files, too — on a CD or DVD.



Whatever you decide, make sure you get it all in writing. Make sure that everyone understands what you're delivering and how. Consult your lawyer regarding your copyrights and how to protect against your materials being used in a way that you don't intend.

Chapter 3: Developing the Content

In This Chapter

- ✓ **Researching site content**
- ✓ **Defining goals for the site**
- ✓ **Keeping the site relevant**

So you have a client who signed on the dotted line and wants you to create a drop-dead gorgeous Web site with all the bells and whistles that the law allows. When you brainstorm with the client and ask him what he wants on the site, you get an answer like, “You know, the usual.” Argggggggh. Wrong answer. The problem with many Web sites is that they don’t have content that’s relevant to what’s out there. So instead of trying to read the client’s mind and put together a site that will fail and inevitably taint your reputation as a Web designer, your best course of action is to do some research with the client to steer him in the right direction. Then, when he’s developing the text content for the site, you can do what you do best: Design an aesthetically pleasing site that keeps visitors returning time and again. In this chapter, we show you how to guide your client in the right direction.

Knowing What to Put on Your Site

When you go to a bookstore and open a book on cooking, you find recipes and information on how to prepare them. When your client’s customers visit her Web site, they expect to find content that relates to the title of the site. You know, truth in advertising and all that bunk? In most cases, the client is responsible for the text content. If your client has created content for the Web, or is experienced in marketing, you’re home free. If not, you need to act as the voice of reason and steer your client in the right direction.

As the Web designer, you’re responsible for the look and feel of the site. This includes elements such as navigation menus; the colors used for the background, buttons, and text; fonts used for the text; and so on. Your choices are driven by your personal taste, experience, and your client’s vision. However, factor in two additional parameters: your client’s intended audience and competing sites. After all, people don’t expect to find a picture of a girl in a bikini on the cover of that Bam guy’s cookbook, do they? In the upcoming sections, we show you how to guide yourself and your client in the right direction.

Finding out what your client's visitors need

During the initial stages of the client/designer relationship, figure out what your client's grand vision for the site is. If your client doesn't have a grand vision but wants a Web site because everyone else has one, you might have a problem. If, however, your client does have a viable product, service, or cause, there are probably several hundred sites devoted to the same product, service, or cause. The following list shows some methods of ascertaining what visitors will require from your client's site:

- ◆ **Visit the Web sites of your client's fiercest competitors.** Imitation is the sincerest form of flattery. If your client's competitors have successful Web sites, explore the sites in depth. Make sure you do your exploration with the client so she knows what type of material you'll require from her to complete your design. Bookmark the sites and refer to them when creating your design. Of course, creating a blatant copy of the competitor's site is not good practice. Put your own spin on what you consider are the most successful elements of the sites you visit.
- ◆ **Find out which elements are considered standard for a Web site in your client's industry.** For example, all photographers have galleries of portfolios, and most e-commerce sites have some sort of catalog and online shopping cart. Savvy Web site visitors expect to see these elements when they visit sites. If your client's site doesn't have these elements, visitors might go elsewhere for their needs.
- ◆ **Poll existing customers.** If your client has an established bricks-and-mortar business, he has another excellent resource for determining content for the site. Ask existing customers which Web sites they frequent that offer services similar to your client's. See Book I, Chapter 4 for more information about creating relevant content.
- ◆ **Decide whether the site should be interactive.** Many Web site owners have customers fill out questionnaires, and other Web sites entertain customers with interactive games or quizzes that relate to the product or service being offered. Find out whether interactive elements are usually associated with Web sites that sell products or services similar to your client's. Book I, Chapter 4 contains more information about creating relevant content and Web elements.
- ◆ **Find out what technology your client's intended audience uses to access the Internet.** All Web site visitors want a fast-loading site, and this information can help you provide that. Your goal as a Web designer is to make an attractive, fast-loading site. The definition of *fast-loading* varies greatly depending on whether your client's intended audience uses dialup modems, DSL modems, or cable modems.

If your client has an existing Web site, you can garner some insight about what types of technology an audience is using by looking at the site's statistics tools. Ask your client what statistics package is in use and request a report — or better yet, request access to the tool itself.

You can also research online for general benchmarks and acceptable ranges for different types of audiences. While the latter will be less specific than the first information source, it is better than making assumptions.

- ◆ **Find out whether visitors of Web sites of businesses similar to your client's expect bells and whistles, such as Flash, videos, slide shows, RSS feeds, blogs, downloadable documents, interactive content, and so on — the list could go on.** If they do, make sure that your content is backward compatible. For example, if you create Flash content that works with only the latest version of Flash Player, you're potentially alienating a large part of your client's potential customers.

Check the Adobe Web site for information about the latest versions of its players and penetration of the various versions of its players. Include alternate versions of content for visitors who do not have access to the latest versions of plug-ins, and also include links to the manufacturer's Web site where they can download a newer version. For example, include a link to the Flash Player download page if you use Flash on a site; include a link to the Apple QuickTime page if you use QuickTime content on the site.

- ◆ **Find out whether visitors like to personalize their experience on Web sites similar to your client's.** For instance, if your visitors are accustomed to being able to sign in and maintain a profile or shopping preferences, add that type of functionality to your site. If you don't, your audience will leave your site in favor of one that delivers the experience they want.

Dealing with copyright issues

Copyright laws protect creators of original content — such as writing, art, photographs, and so on — from people using unauthorized copies of their work. Copyright laws also apply to Web designers. When you create a Web site, you're using content supplied by your client. If your client indeed created the text and images you're using on the site, he owns the copyright to this material. If, however, you use material that was not created by the client (such as photographs and music), you must license the right to use this material as part of your design. If you purchase a collection of clip art or purchased stock images from one of the stock art houses *and your license allows you to use the image as part of a Web design*, you're covered under the copyright laws. Notice the caveat we include regarding your license? That's right. Even if you bought it, that doesn't mean you can use it. Some licenses are very rigid and allow a limited number of uses for an item. Many licenses also prohibit significantly altering clip art.

The best defense here is to read the fine print before using any item you purchased, or are contemplating purchasing, for use in a Web design. If you purchase a collection of images or music for use in your designs, make sure

that they're royalty free. If not, you're responsible for paying royalties to the copyright owner of the work. Copyright laws also protect logos. If your client sells a product line and requests that you use the product logo on the Web site, make sure this is permissible by the company that manufactures the product. In most instances, your client has to agree to certain terms in order to display the licensed version of the logo on the Web site.

The written word is also copyrighted. If your client provides you with verbatim descriptions from a product catalog or another Web site, he might be in violation of the copyright law. Certain items can be copied and used under the Fair Use Doctrine (see the U.S. Copyright Office site on Fair Use at www.copyright.gov/fls/fl1102.html). Even though you're using material that might be copyrighted by others, the manner in which the material on your site is presented is unique and should be copyrighted by the owner of the site you're designing.

Another copyright issue is the completed site. You can copyright the site by adding the following at the bottom of each page: Copyright 2009 by *your client*. All rights reserved.

In the end, the best defense is a good offense, which, in this case, means that you and the client should create as much of the content as possible. If your client presents any material that might be questionable, strongly suggest that he contact legal counsel. Attorney's fees are a lot cheaper than paying for duking it out in court.

Finding out what your client needs from site visitors

Before you launch your favorite HTML editor, do your homework and find out as much as possible about your client and the type of Web site she envisions. If you already designed sites for similar clients, your knowledge can help guide the client when she's at a loss for answers. Ask your client what her goals are for the site. Get your client to go into detail concerning site goals. Learn as much as you can about her business, and make sure to take copious notes. The following are a few questions you can ask your client to clarify her goals:

- ◆ **Does your client want to sell goods to visitors?** If so, will the client rely on the site as an online catalog and have personnel fill orders via phone, or will the site be a full-blown, e-commerce site on a secure server? (See Book VII for more information about e-commerce Web sites.)
- ◆ **Does your client want to inform visitors?** If the client has an established bricks-and-mortar business, she can use a Web site to cut down on overhead. For example, instead of giving out catalogs, the client can provide product specifications on the Web site, which eliminates mailing and printing costs and also cuts down on the personnel needed to staff the phones.

- ◆ **Does your client want a service-oriented site?** Clients use service-oriented sites to answer frequently asked questions from customers, handle service issues, and so on.
- ◆ **Does your client want repeat visitors?** If so, you have to design a site that gives visitors a reason to return. You also have to tell your client that the site must be updated frequently in order for visitors to return. The frequency of updates varies depending on the goals and audience of the site. Some sites need daily updates and new content; other sites can be effective with seasonal updates. For instance, a site for a vacation destination may not need to be updated daily; a seasonal change or monthly update could be all that's necessary to serve the audience. The most important thing to remember is to set expectations with visitors that there's a reason to return to the site, and then deliver on that promise.
- ◆ **Does your client want to collect contact information to keep visitors apprised of new information about his products or services?** Some clients may want to offer a newsletter to their visitors. You will need to know this and address the need either through creating a system to handle a newsletter or by finding a newsletter service for your client to use.
- ◆ **Does your client want to frequently update the site?** If so, negotiate a separate contract for ongoing updates. Some clients will opt for doing their own site maintenance. You will need to know what their plan is before you start the project because it will affect both the budget and how you build the site.

For a client that wants to maintain her own site, you have options for how to proceed. You can build the site with Dreamweaver templates and advise her to use Contribute (Adobe's easy-to-use and inexpensive WYSIWYG editor/site management tool). You can also opt for building the site with an open source content management system, such as Joomla, Drupal, or even WordPress. There are several good options if you choose this route. The "right" one will depend on your personal preference as the designer/developer and on the project's needs.

- ◆ **Does your client want to maintain ongoing communication with site visitors?** If so, what sort of communication would they like to have? Would the client benefit from using a blog, newsletter, message board, forum, or other social media? The answers will depend on how much time the client wants to spend interacting, how interested their target audience is in interacting, and what their favorite method of interacting is. Some sites will have visitors that will like a message board but won't care about using MySpace. Other sites can leverage a site like MySpace very effectively. As a site designer/developer, you can help figure out what (if any) communication will be best for the site.



Tell your client when his goals require technology or design elements that exceed his budget. If your client insists on using these elements, give him the price for the additional elements and be prepared to write an addendum to your contract.

Reconciling the content with the goals

After you know what visitors expect from sites similar to your client's and you know what your client's goals are, you're ready to create the site. Your client should provide you with a good deal of the content. In an ideal world, the client submits all images to you in electronic format, optimized and ready to pop into your design. However, it's been our experience that the client is most likely to send you images that you need to scan and optimize, or images in electronic format that you need to resize and optimize for the Web. Your client should also provide text for the site. In the Zen tradition of less is more, make sure that your client doesn't go over the top by providing too much information or by providing information that isn't relevant to site visitors. When reviewing text for the site, think like the site visitor and ask the question, "What's in it for me?" If some of the information doesn't address a customer question or need, suggest that your client delete it.

In addition to what your client supplies, you might need to create content for the site. The content that you create should be based on customer needs and your client's goals. Based on this information, you might need to add the following to the site:

- ◆ **Secure server:** A secure server and an online shopping cart are necessities if your client wants to sell merchandise from the site.
- ◆ **Online catalog:** Create an online catalog if your client wants to inform visitors about his goods and services.
- ◆ **E-mail newsletter:** Include an e-mail newsletter if one of your client's goals is to keep customers informed. An in-depth discussion of e-mail newsletters and the techniques for using them are beyond the scope of this book, but you can read about the topic of effectively using e-mail campaigns in *E-Mail Marketing For Dummies*, by John Arnold (Wiley).
- ◆ **Blog:** A *blog* (a Web log) is an easy way for your client to maintain ongoing communication with site visitors.
- ◆ **Forms and databases:** The Web site needs forms and databases if the client wants to collect contact information from people visiting the site.
- ◆ **Analytics/statistics tools:** Many hosting companies include Web site analytics/statistics tools in a hosting package. Some free products — such as Google Analytics — are very good, too. An in-depth discussion of these products and the techniques for using them are beyond the scope of this book, but the topic of collecting information about your site's traffic is covered in *Web Analytics For Dummies*, by Pedro Sostre and Jennifer LeClaire (Wiley).

- ◆ **Cookies:** Use cookies if customers want to personalize their experience while visiting the site. *Cookies* are small text files that are downloaded to the user's computer. When the visitor next visits the site, the information from the cookie is recognized by the HTML code. See Book VII, Chapter 2 for more information about cookies.
- ◆ **Templates:** Use templates in your layout of sections of the site your client wants to update frequently. Using templates saves time and trouble. Creating templates enables you to delegate the work to a less-experienced designer. If the client wants his staff to update the site, templates are imperative. Templates prevent inexperienced editors from corrupting navigation menus and so on. When you create templates, lock the areas you don't want the client or other designers to modify.



If the client's staff will update the site, design your site so that it can be edited with Adobe Contribute — and include the price of the application as part of your Web design fee. Also, make sure to include training sessions in your contract. Cover additional training and consultation in a separate contract or separate clause of your Web design contract.

Keeping It Fresh: An Ongoing Process

If you've done your best to create a compelling site for your client and you've optimized the site 27 ways to Sunday, people will visit the site. Like anything else, though, the bloom is off the rose rather quickly. People might watch a good movie many times, but people rarely visit a Web site with the same content over and over again. To keep visitors returning, you or your client need to give visitors a reason to return. If interested visitors know that the content is updated on a regular basis, they will return.

Assigning content development tasks

If your client decides that he wants the site updated frequently, either you or your client's staff is responsible for updating the pages and uploading them to the server. If your client contracts you to do the updates, you and your staff can schedule updates at a convenient time. If your client does the updates, he can coordinate the task with his staff. If your client's staff is doing the updates, make sure you design the site with templates, locking the areas with which you don't want the client to tamper. Make sure your client's staff has the knowledge and software to update the site without destroying your hard work. For personnel with little or no Web experience, Adobe Contribute is an ideal application to update a site.

Keeping the graphics fresh

When people visit your Web site for the first time, it's a shiny new face. But like the man in the mirror, the shiny, new face grows old after a while. In this regard, the graphics on the Web site should be updated on a regular

basis. Anything with the client's logo, such as the site banner, should not be updated. After all, you do want some continuity in the site. The graphics you do want to update are any Flash movies and graphics used for news, information, and the like. If your client will update the site, get her to supply you with several images that will be rotated in certain parts of the site — for example, the home page and the What's New page, if the client has one. Size the graphics to fit the spots on the page where they will be displayed, optimize them for Web viewing, and then return the optimized images to your client. If your client uses Flash movies on the Web site, create a few alternatives that are the same size and show your client's staff how to swap out the movies when needed.

Chapter 4: Creating Relevant Site Content

In This Chapter

- ✓ **Putting the company in a favorable light**
- ✓ **Offering information**
- ✓ **Developing content**

A Web site is a very important thing. At least, it better be. When you're hired to create a Web site, you're creating a worldwide presence for your client. As a Web designer, you do your best to create a site that portrays your client's business or service at its best. It's a two-way street, though. You can be the best designer in the world, but if your client doesn't know what he wants or needs for his Web site, your best design efforts are in vain. A Web site is all about marketing a company, product, or service. It's an extension of the client's bricks-and-mortar business (if he has one). If not, the Web site serves as the identity for your client's business. Your job as a designer is to work with the client in order to portray his company in the best possible light. This chapter gives you some information that you can use to guide your client in the right direction.

Portraying Your Client's Company

When you develop a Web site for a client, you're presenting your customer's professional image for the world to see. Your client might sell goods or services or provide information for customers. A Web site works 24/7 and has the potential for drawing a worldwide audience. The organization portrayed on the Web site might or might not have a bricks-and-mortar location where it does business. Even if it does have a bricks-and-mortar business, that won't matter to an audience whose opinion of the company is defined by its Web site. Therefore, it's paramount that you paint the best portrait possible when you create a Web site for a client.

Defining your client's voice

Your client's voice encompasses many things: the manner in which your client does business, your client's mission, the image your client portrays to his customers, and so on. When you define your client's voice through a

Web site, you're answering the question: Do I want to do business with this company or individual? To gain the trust of Web site visitors, the site must leave a positive impression. But more than that, the site should be unique enough to stand out against the client's competition.

Most of your client's serious competitors probably have Web sites. And it's almost a given that Web sites within a certain industry will have a common look. Therefore, the only way you can make your client's site rise above the competition is to portray your client's unique assets — the client's voice, if you will. When defining the voice of a company, the message you portray on the Web should be consistent with the client's other communications, such as printed ads, television commercials, and audio advertisements. If your client has a startup company, the Web site helps define the client's essence. To define this intangible, you need to brainstorm with your client and consider the following:

- ◆ **Audience:** Who is the client's intended audience? What's unique about your client's intended audience? How do you portray your client as part of this unique group of individuals? What type of message does the Web site need to deliver to set your client apart from his competition, in the eyes of his intended audience?
- ◆ **Mission:** What is your client's intended mission? How does your client plan to serve the intended audience? When defining the client's mission, steer clear of the usual clichés, such as highest quality, superior service, satisfied customers, and so on. Choose more sophisticated wording to make your client stand out from his competition. In addition to instilling confidence in your client's intended audience, his mission statement must portray his unique core values. You can augment the mission statement with testimonials from your client's satisfied customers.
- ◆ **Style:** What is your client's style? Is she an entrepreneur? Where does the client fall in the broad spectrum of similar businesses? Is she smack-dab in the middle? Is she conservative, or bold and brash?
- ◆ **Customers:** How do your client's customers perceive her? The competitors? Do the answers to these questions match your client's perception of herself and her business?
- ◆ **Goals:** Where does your client see herself in five or ten years? Will the message your client portrays now be viable in five or ten years?

Armed with this information, you and your client can begin making some decisions about how you'll portray the company on the Web. While you're at it, have the client distill the information into a few paragraphs that describe the company, its core values, and future vision. You can incorporate this information into an effective mission statement. If the client already has a mission statement for his business, use it on the Web site. There's no need to reinvent the wheel.

Developing your client's look

After you and the client define her voice, think of how you'll incorporate this with the Web site you create. In essence, you're defining your client's look on the Web. The amount of creativity you can employ in defining your client's look on the Web depends on whether the client has an established, bricks-and-mortar business, and here's why.

If your client has an established bricks-and-mortar business, you need to create a Web site with a similar look and feel. The site has to incorporate the client's logo and images used in corporate brochures and advertising. Your client's presence on the Web is defined by how well you assimilate your client's bricks-and-mortar look with your Web design. When you're designing a Web site for a client who has already established a look, ask yourself these questions:

- ◆ **Does the Web site have the look and feel of the client's bricks-and-mortar business?** When established customers visit the Web site, the site should seem like an extension of the client's bricks-and-mortar business. The design should feel right to the client as well as to his customers. In addition, you should incorporate your client's logo in the design. The site colors should be harmonious with the colors used in the customer's logo.
- ◆ **Does the design effectively portray the client's voice?** The design you create needs to portray the client's manner of doing business, his mission statement, and his style of doing business. Your design must also be consistent with the expectations of the client's current customers and intended audience.
- ◆ **Can the design be modified to fit the client's goals and needs?** If your client's long-term goals are substantially different than current goals, you'll probably end up doing a complete redesign at some point in the future. However, if your client's goals are to grow and increase his customer base with the same or similar product offerings, your initial design has to be flexible enough to incorporate future expansion, such as the addition of new sections, changes in product lines, and so on.

If you're designing a site for a client that will do business only on the Web, you have considerably more leeway when developing that client's look. With the client's help, you can establish the client's Web presence. If the client already has a logo, you use this as a starting point. The colors in the client's logo help define the color palette you use to design the Web site. After deciding on the color palette, your next step is to assimilate graphics into the design. If the client is a *solopreneur* (runs a one-person business) or has a small business, you can make the business seem larger than life with the creative use of clip art. If your client is an aggressive businessperson, you can depict your client's style of doing business with the creative use of

color, fonts, and bold graphics. The finished site must effectively communicate your client's mission, goals, and style of doing business in the most positive manner. The design in Figure 4-1 incorporates the colors from the client's logo.



Figure 4-1: Create a site that portrays the client's voice.

Selling Goods or Services

If you're creating a site that sells goods or services, your goal is to create a compelling site that piques customer curiosity and instills customer confidence. A lot of snake oil salesmen lurk on the Web, and buyers definitely err on the side of caution. Therefore, your job as a designer is to instill trust in your client's potential customer base while also bringing the product or service to the forefront of the visitor's mind, hopefully resulting in a click of one or more Buy Now buttons. It's a shame that Buy Now buttons don't make a "cha-ching" noise your client can hear, as this would be music to his ears.

Emphasizing key points

When you sell a product or service online, design your pages in such a manner as to instill customer confidence while emphasizing the key points of your client's product or service. The old, Zen-like maxim of "Less is more" definitely applies here. Instead of barraging the Web site visitor with each and every subtle nuance about the product, point out major features of a product without engaging the visitor in several paragraphs of text. You can easily sum up a product line by showing the key points about a product in a succinct paragraph or two. Better yet, use bullet lists to emphasize the key points of a product line. Your goal is to pique the buyer's curiosity and get her to take action.

You can also emphasize key points by using different colors for text, using formatting (such as bold or italic), and so on. If you're using bullet points, consider boldfacing the word(s) that define the bullet point and perhaps using a different color text. If you use a different color text to emphasize a key point, choose a color that is harmonious with the design. (See Figure 4-2.)

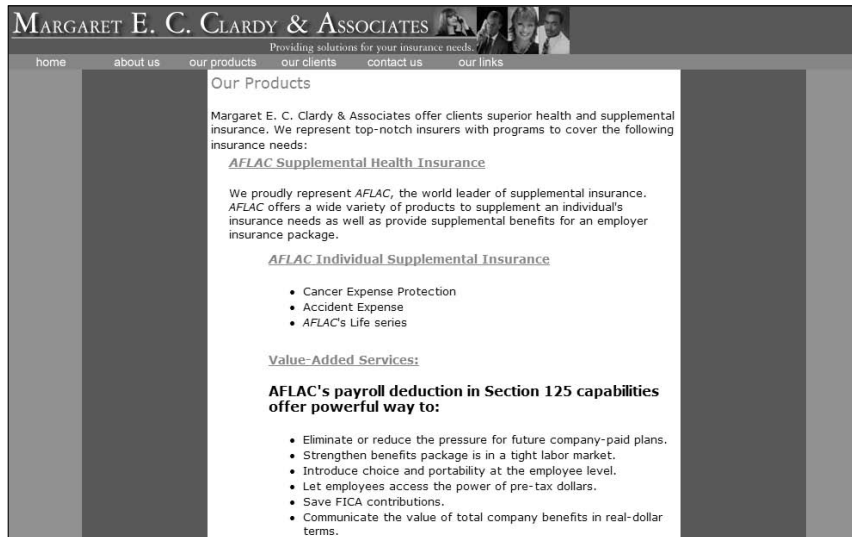


Figure 4-2:
List major points to pique viewer curiosity.

The same rules apply when you're presenting information about a service. Emphasize the service with bold text and an introductory sentence followed by bullet points. To emphasize the key points of the service, use a different color for the text of the key points. This directs the viewer's eye to the most important information.

Highlighting the product

Whether the goal of a Web site is to sell products or present information about a product, your design should make that readily apparent to the viewer. When working with products, obtain high-quality images from your client. If your client sends you pixelated images that have been severely compressed, ask the customer for the originals and then optimize them in Photoshop or Fireworks. Armed with good pictures, you can then employ other methods to highlight the product.

If you're dealing with a complex product line, devote a single page to each product. Including too many products on a single page dilutes the effectiveness of the message. Figure 4-3 shows a product page. The product image is the highlight of the page, followed by a paragraph of concise text that tells visitors key information about the product.



Figure 4-3:
Highlight
a single
product on a
Web page.

Providing Information

If your client's goal is to provide only information about his services, you'll create a site that provides information about the client's products or services but with (understandably) no option to purchase the product or service online. When you create an information-driven Web site, you use text or brochures supplied by the client to create the content for the site. You might also assimilate information created by others, such as articles or multimedia content. And, of course, you need high-quality images of your client, his services, and his products. As far as accuracy of the material goes, the client is responsible for that. You, however, are responsible for providing the information via your Web design in an easy-to-digest and understand manner.

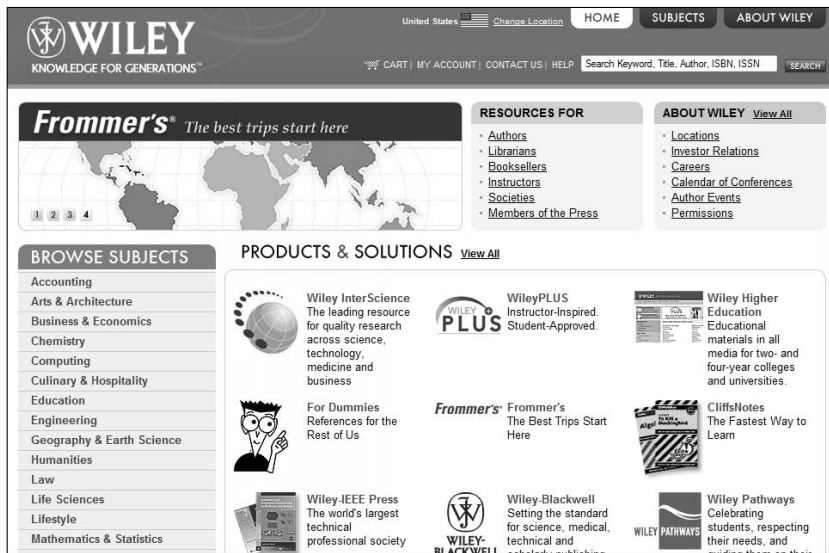
Handling large amounts of content

If your client has copious amounts of information to dispense through his Web site, you have a challenge on your hands. The attention span of most Web site visitors is less than a New York minute — and we all know how short that is — which means visitors aren't going to read pages with a lot of text. It's your job as a designer to break the information down into easily digestible, bite-size pieces. Here are a few techniques you can use to achieve that goal:

- ◆ **Break large amounts of text into several pages.**
- ◆ **Intersperse images with the text to break up the content.**

- ◆ **Break up large amounts of text with headlines that visitors can use to ascertain whether they want to read the content.** Savvy Web surfers also use headlines to get the gist of what information is presented on a page.
- ◆ **Create a newspaper-style page where the information is presented in columns.** Each article has a headline. Instead of including all of the information on a page, include the first paragraph or two and then include a More link at the end of the last sentence. When clicked, the More link opens the full article in another window.
- ◆ **Create a home page that contains links to the information you need to present.** (See Figure 4-4.)
- ◆ **Create a site map.** The site map has a text description of every page on the site. Each text description doubles as a hyperlink, which, when clicked, reveals the applicable page.

Figure 4-4: Links enable you to present a large amount of information in a small space.



Presenting information

Presenting information on a Web page is an art unto itself. The information you present depends on the type of site you design and what your client gives you to work with. The typical Web site is a mixture of images and text, with, perhaps, some multimedia content added for grins and giggles.

- ◆ **Break major ideas into bullet points.** As we mention earlier, bullet points make a page look less cluttered and help visitors find information more easily. See Figure 4-5.

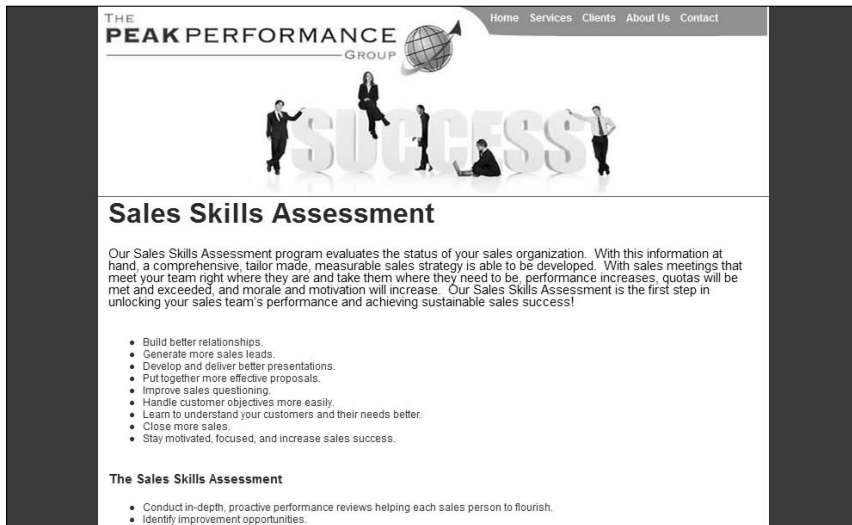
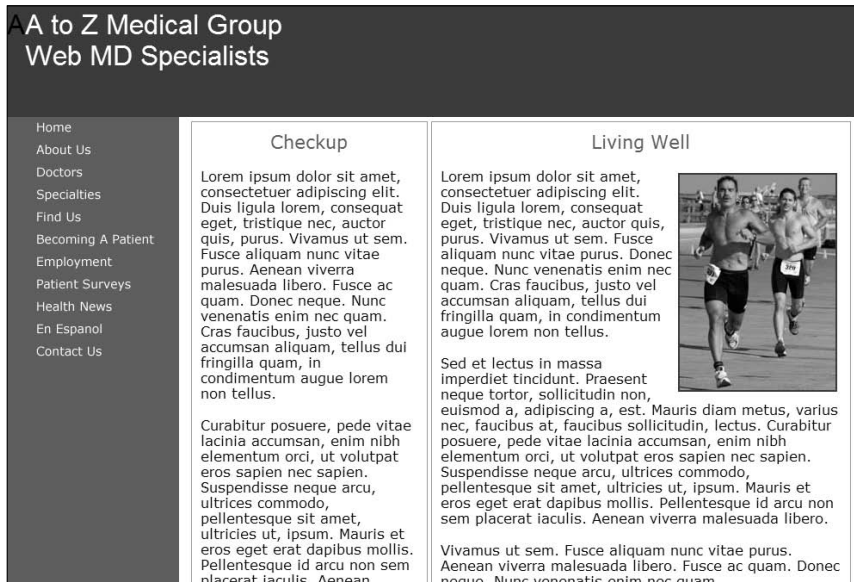


Figure 4-5:
Break
information
into bullet
points.

- ◆ **Arrange the page so that the information has a logical flow.** You can break up big blocks of information using header styles. Readers should be able to sum up the information presented on a page by glancing at the headlines and then deciding what information they want to read.
- ◆ **Break up text with pictures.** No one wants to read a term paper. If applicable, add a caption to each picture on a page. Busy site visitors can use captions to sum up the information presented on a Web page at a glance.
- ◆ **If you're using a newsletter style with multiple columns for your Web page, make the columns different widths.** Also, don't exceed two columns, especially if your audience uses a desktop size smaller than 1024 x 768. (See Figure 4-6.) If you're using a horizontal menu, you can increase the size to three columns.
- ◆ **Make sure that the structure and the layout of the pages are consistent.** Pages might have to vary, depending on the content, but the look and style need to be consistent.
- ◆ **Don't use color combinations that are hard to read.** For example, pink text on a dark-colored background is difficult to read. The old tried-and-true black text on a white background is always legible.
- ◆ **Don't use a busy background for your pages.** The busy-ness makes text hard to read. If your client insists on a *tiling* background (a small image is repeated on the page, which looks like a single image when the page loads), choose a simple design and lower the opacity in an image-editing application (such as Fireworks or Photoshop).

Figure 4-6:
Create a
newsletter-
style page
that's easy
to read.



- ◆ **Don't use small fonts.** A text size smaller than 10 points (pt; the height of a character) is almost impossible for visitors to read unless they have extremely acute vision. And if your intended audience is on the presbyopic side (um, getting on in years, vision-wise), text shouldn't be smaller than 12 pt.
- ◆ **Use bold-faced text to direct the viewer's eye to important information.**
- ◆ **Don't use nonstandard font faces when designing your pages.** Your intended audience might not have the font installed on their machines, which causes a browser to use its default system font. This might cause usability issues and make the page difficult to read. When in doubt, stick to the big six: Arial, Times New Roman, Courier New, Courier Mono, Helvetica, and Verdana.
- ◆ **Don't use too many font faces.** If you use too many font faces, the end result doesn't look professional and is distracting to the viewer. If you need to draw attention to a headline, you can create a style using the same font you use for text: Boldface the text, increase the size and use a harmonious color from the site to draw the viewer's attention to the headline.
- ◆ **Use hyperlinked text to draw the viewer's attention to pertinent content.** Just make sure you don't disable hyperlink decoration with a Cascading Style Sheet (CSS) setting. Your goal is to make the hyperlink easy to see when the page loads. If you don't like underlined hyperlinks, use the style sheet to modify the color of the hyperlink text.

- ◆ **If several authors write the Web site content, make sure the style and voice is similar.** A Web site should read like a book. Each page is like a chapter of the book, and if the styles are jarringly different, the visitors may think they're not on the same site. Consistency is important.
- ◆ **Make sure the terminology and spelling are consistent sitewide.** For example, don't use *Website* (one word) in one section, and *Web site* (two words) in another. At the risk of being redundant, consistency is important.
- ◆ **When you're designing your site, make sure you view it using the smallest desktop size of your intended audience.** As of this writing, 91 percent of Web site visitors are surfing the Net with a desktop size of 1024 x 768. Make sure your design flows logically and is easy to read.
- ◆ **If you're using multimedia content (such as movies and sound files), make sure they're all the same file format.** For example, don't use the Windows Media Video (WMV) format for some of your movies, Flash video for other movies, and Apple QuickTime for the rest. Choose a file format that the majority of your client's intended audience can view, and stick with that standard.



Read the information provided by the client before adding it to the page. If (in your opinion) the client is presenting too much information (also known as stuffing 50 pounds of text into a 30-pound bag), ask the client to condense the text to be more concise. When all else fails, refer your client to a reliable copy editor.

Including e-learning materials

If your client is creating a Web site that instructs end users on how to use a particular application or accomplish a specific task, include material that can get them there. Many methods of accomplishing this are available. In fact, specific software packages are available to create content for e-learning Web sites. The following is a list of some of the items currently used on e-learning Web sites:

- ◆ **Instructional video:** Nothing gets the point across like video. If the site shows users how to master a software application, a video capture of the application in action is the ideal teaching tool. In conjunction with the teacher's audio instructions, video is a powerful e-learning tool. You can capture a video of an application using Camtasia or HyperCam software. The video you post online can vary depending on the type of Internet connection used by your client's intended audience. You can post streaming video using the Windows Media Video format, Apple QuickTime format, or Flash video.

- ◆ **Flash interactive content:** Flash is a wonderful tool for creating animations. You can also use Flash to create quizzes, interactive learning exercises, and so on. If you're adept with ActionScript, you can get Flash to do some wonderful things, such as tally the score of a quiz or direct a user to a different part of the Flash movie based on the response to a question.
- ◆ **PDF documents:** Adobe Acrobat is another wonderful tool that you can use to create tutorials. Many people think that PDF documents can only show text and images. This is not true. You can create a PDF document in Acrobat 8 Professional or later that can include multimedia content, such as audio and video. Acrobat PDF documents can also be interactive. Acrobat has built-in actions that you can use to link to other documents, play video or audio, and so on. An Acrobat document can also use JavaScript to perform tasks.

You include links to your e-learning materials in your HTML documents. You can set the learning material to open in the same browser window or in a new window. You can embed e-learning material, such as video (complete with a player), in the document. This is the preferred method because many people rely on the Back button to recall viewed content. When a document is opened in a new window, this navigation isn't available.

Using Personas to Develop Content

When you create a Web site for a client, part of your job is to identify the client's intended audience. After you find out who the client's intended audience is, find out as much about the audience as possible. Your client should be able to provide you with the majority of the information. You might be able to find other information, such as demographics, online. Armed with this information, you can tailor the content to the intended audience. In other words, the site you design connects with viewers on a personal level, a task that's easier said than done.

If your client's goal is to connect with viewers on a personal level, you can create one or more personas to define your client's archetypical visitors. A *persona* is a hypothetical person whose characteristics and demographics fit your client's intended audience to a tee — and, therefore, has all the information you need to define the audience. When considering what content to use on the site, use the persona to guide you. In other words, tailor your content to the persona, and your client's intended audience will feel as if the content were written personally for them. Depending on the scope of your client's intended audience, you might have to create multiple personas. For example, if the intended audience is young males between the ages of 24

and 35 with a college degree who live in the U.S. and make between \$65,000 and \$95,000 per year, you can get by with one persona. However, if your intended audience is male and female with varying degrees of education and from varied socioeconomic groups, you'll have to create multiple personas.

Defining your client's customers

When creating personas, you have to rely on your client to provide the information. After all, you are a Web designer, not a marketing guru. If your client is not familiar with creating personas, here are a few guidelines:

- ◆ Create a one- or two-page description of each persona. Include information such as gender, age, marital status, ethnicity, education, profession, income, and so on.
- ◆ Include information about the habits and daily routine of each persona.
- ◆ What are the hobbies and pastimes of each persona?
- ◆ What are the buying patterns of each persona?
- ◆ What computer skills does the persona have?

Your client will probably end up with between three and six target personas. If the number gets larger than that, you'll have a hard time tailoring content for such a diverse group. The whole idea behind personas is focusing on the type of individuals who will use your client's product or service, not the general public.

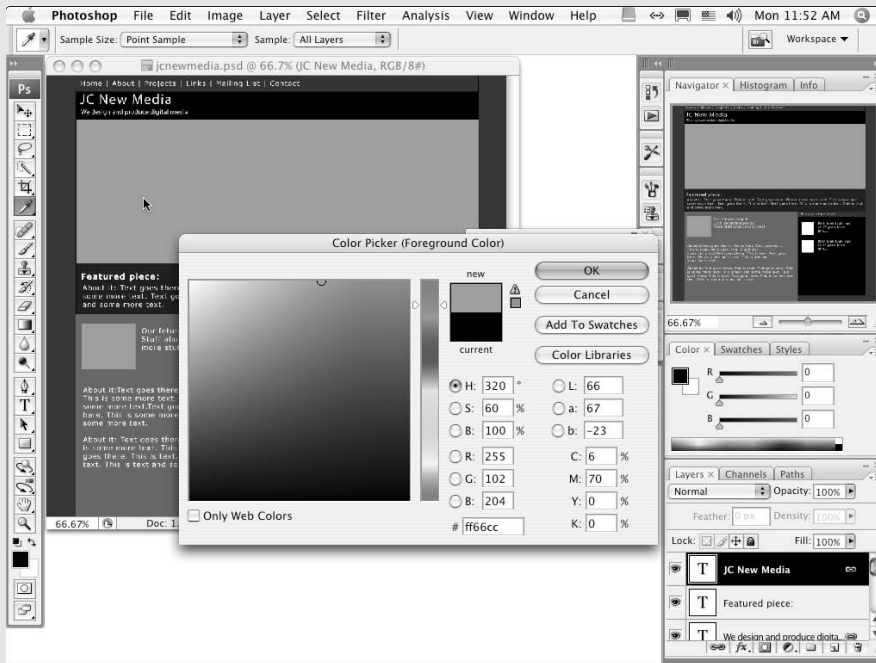
Delivering what they want

With multiple personas defined, your client can begin creating content for the site. Of course, the content your client provides will be text. The authors will create text that is appropriate for the age, gender, and education level of the personas. You'll create a navigation menu and design concurrent with the likes and computer skills of the personas defined by your client. When you have the initial design nailed, you start adding graphic elements, such as banners and images. If you're using multimedia elements, such as background music, the personas defined by your client will guide you to the proper choice.

The content created for the Web site is also governed by the area in which the personas live. If the personas reside locally, you'll want to include information about your client's local events. If the personas live in all areas of the country or world, you can include information about worldwide events pertaining to the client's business or service. If your client hosts *Webinars* (Web-based seminars), make sure that information is prominently displayed on your client's site.

Book II

Site Design



Choosing a color system.

Contents at a Glance

Chapter 1: Conceptualizing Your Site	57
Deciding What Types of Pages You Need.....	57
Choosing the Right Delivery Method	64
Handling Printable Materials	67
Chapter 2: Creating Effective Layouts	71
Content Is Key: Making Sure It's All Included	72
Creating Order from Chaos: Consistency Is Your Friend.....	73
Including Things That Every Page Needs	75
Planning How to Emphasize Important Information.....	81
Understanding the Basics of Usable and Accessible Design	83
Creating a Layout Wire Frame	85
Chapter 3: Organizing Your Content	91
Starting with the Skeleton: The Structure	91
Considering File Management.....	92
Keeping It Tidy.....	94
Implementing Information Architecture.....	96
Avoiding Content Problems	99
Chapter 4: Ensuring Visual Appeal	103
Colors on the Web.....	103
Getting Familiar with Graphics File Formats.....	108
Using Fonts Effectively on Web Sites	111
Implementing Stock Images and Other Elements.....	113
Facelifting an Existing Site.....	115

Chapter 1: Conceptualizing Your Site

In This Chapter

- ✓ Deciding what types of pages you need
- ✓ Choosing the right delivery methods
- ✓ Creating printer-friendly materials

An upcoming chapter in this minibook (Book II, Chapter 3) covers developing content. This chapter discusses how to present content to your visitors. You must keep some content in a database; you store other content as plain HTML pages. And you can consider several multimedia methods of delivery. Not every method is good for every type of information. You must determine what types of pages you need, how to deliver those pages to your viewers, and whether to include printable pages.

Deciding What Types of Pages You Need

Depending on your need, you choose between *static* Web pages (plain HTML pages) and *dynamic* Web pages (pages that can react to site visitors; more on that in a bit). Different situations require these different types of pages. Either type of page can include multimedia elements. When determining what types of pages you need, consider the following:

- ◆ **How much information you have and how you want to organize it:** Some sites have only a few pages with some pictures and a little text. Other sites have massive amounts of information and need a database (or more than one database) to manage it all. Most sites fall somewhere in between.
- ◆ **How you want visitors to use your information:** Do you want them to passively read, or would you like them to interact with the content? Do you want them to be able to download anything?
- ◆ **Users' expectations:** To get an idea of what users want from your site, look at similar sites to get an idea of what sorts of features and functionality they have in common. Visitors to your site will most likely have an idea of what is typical, expecting that your site have similar features. For instance, if you build a site that sells children's stories, the target

audience will probably expect some downloadable coloring pages and other free fun things for kids. If your site doesn't have those types of features, your audience might abandon your site for one that does.

- ◆ **Layout considerations:** Some sites have a unique home page layout and then two or three subordinate page designs. Other sites have a consistent layout for every page. Which method you choose depends on the needs of your site. Having a unique home page layout helps unify a site that has several sections that have different needs. For instance, for sections of your site that primarily display text, you need a layout that works for that type of information. The same site can include a catalog section for products. The layout for that section is different because you must display the products. The home page for a site like that works as a jumping-off point that unifies the two areas of the site. The home page also is a good place to highlight information or drive traffic to a specific feature of your site. A unique home page layout can effectively support that functionality.

Other page layout considerations are whether the site will have charts and graphs or tables, which need to be delivered as graphics or multimedia elements. In some situations, the best way to handle this type of information is to embed the chart/graph directly into the HTML document. This works best if you can display it effectively along with any supporting text on the monitor without having to scroll — even on monitors with lower resolutions (800 x 600, for instance). If you need the chart or graph to be larger, you can have it launch in a second browser window. That way, it can use more space while keeping the supporting text information easily available.

- ◆ **Multimedia considerations:** How do you want multimedia elements to display? The considerations are the same for charts and graphs. Look at the content to decide whether it will work best embedded in the page or displayed in a second browser window. The second window can be smaller, which allows visitors to see and use the original page. Figure 1-1 shows a Flash element launched in a smaller browser window.

Figure 1-1:
Using a smaller, second browser window can help visitors access your information.



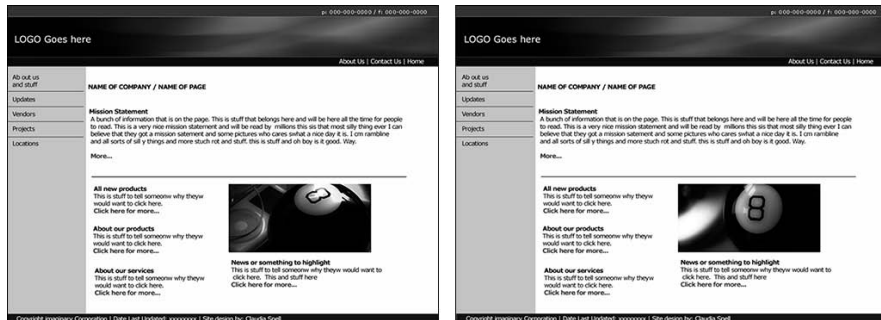
Considering static Web pages

Static Web pages are plain HTML pages. They can include some JavaScript and should include some CSS (Cascading Style Sheets) for formatting. *Static* pages contain elements that don't change (that is, unless the Webmaster makes changes). Each time the page loads, it's pretty much the same. The content remains the same regardless of who views it.

When a visitor tries to view a static Web page, a request to view the page is sent to the server that holds the HTML document. The document and any supporting files are delivered to the visitor's browser. The HTML document that's stored on the server is a whole document with all its elements included in it. The only external pieces are CSS, images, and some JavaScript. The same HTML document is delivered every time, regardless of who is trying to view it or what they did previously.

Static Web pages, however, can include some JavaScript that can give the illusion of dynamic content. An example of this is script that displays different images in a specific place each time the page loads. If you've browsed a few sites, you've probably seen this effect. You can use it to switch out CSS files, also. Figure 1-2 shows an example of a Web page that has a bit of JavaScript randomizing an image. This might appear to be changing content, but it really isn't. The JavaScript has a list of image files to choose from, and it randomly chooses from that list each time the page loads.

Figure 1-2:
A simple bit of JavaScript can give the illusion of changing content.



Freshening the content with dynamic Web pages

Dynamic Web pages are built when a visitor comes to your site. That might sound a little strange, but it's true. When a visitor tries to view a page that is dynamic, requests for information are sent to the server. The server reacts to the request and assembles the pieces of the page as requested and delivers them to the visitor's browser as an HTML document. An example of this is when you search for an item on a shopping site. The server finds and delivers to your screen what you've asked for. This is dynamic content.

The big difference between dynamic pages and static ones is that dynamic ones can react to your visitors. You can include scripts in parts of your code that help your site remember visitors when they return. The scripts assemble customized pages based on those visitors. A good example of this is the Amazon.com Web site. Each visitor's experience is customized to that person based on previous visits. This is accomplished by using a series of communications between the server and the user's browser.

Dynamic pages allow users to send you information via forms that can be stored in a database. And you can allow your visitors to request information and perform searches on your site content.

Dynamic pages comprise pieces of information as well as pieces of pages. Where the code for a static page is included in only one document, dynamic pages are made up of headers, footers, and include files. Each of these is a mini-HTML document that isn't complete on its own. When assembled by the server and delivered to a browser, though, dynamic pages become complete. The content in these pages is typically pulled from a database. It's easy to spot pages that are dynamic while you're browsing. A page with an `.asp`, `.jsp`, or `.cfm` file extension — rather than `.htm` or `.html` — is a dynamic page.

You can create dynamic pages using any of several technologies. You're likely to hear about ASP (Active Server Pages), JSP (JavaServer Pages), and ColdFusion when you start looking into dynamic Web sites. (You can find more information about using ASP to create dynamic Web pages in Book VI, Chapter 2.) If you want a page to really interact and react to your visitors, you need dynamic functionality. The pages can be the same for each visitor, and you don't need to pull out the big guns.

When talking about dynamic Web sites, you'll hear certain terms, including these:

- ◆ **Front end:** This is code — the CSS and HTML wrapped around the information that's being delivered from the database — that deals with how the information is displayed.
- ◆ **Back end:** This refers to the database and supporting structures.
- ◆ **Middleware:** This is the collection of code that communicates with the server, the database, and the front-end code. It's just what it sounds like — the middleman between the front and back.



Each area — front, back, and middle — requires different skill sets to create. If you'll be collecting personal information or allowing purchases on your site, you need to take special care in building the middle and back-end pieces to avoid any problems.

Evaluating multimedia element choices

Multimedia is a broad term that includes video, sound, animations, and other presentations. Multimedia isn't a series of HTML pages that are navigated in a linear fashion. Although the effect is similar to a presentation, it's not multimedia unless it's actually a self-contained piece embedded in a Web page.

Multimedia elements can be dynamic, depending on what type of multimedia you use. For instance, Flash can communicate with databases and serve dynamic content — *dynamic multimedia*. Flash can also collect information from your visitors by using forms, and can then deliver the information to your server. When you use Flash this way, the Flash element is said to be the *front end*. You can use Flash to create a form (see Figure 1-3) that passes information to a database.

Figure 1-3:
You can use Flash to collect information to be passed to a database.

Customer Feedback		
Name	Comments	
<input type="text"/>	<input type="text"/>	
Street		
<input type="text"/>		
City		
<input type="text"/>		
State		Zip Code
<input type="text"/>		<input type="text"/>
Phone Number	eMail	
<input type="text"/>	<input type="text"/>	
<input type="submit" value="submit"/>		

You can deliver multimedia content in many different ways, and which way you choose to use depends on many factors. The most important things to consider are what the multimedia content is and what you're trying to accomplish. If you're trying to let your visitors hear some music, you need to look into formats that let you get the music to your visitors easily and well. Figure 1-4 shows a Flash-based jukebox used to deliver MP3 files to users.

Figure 1-4:
A Flash
jukebox
delivering
music files.



Making a decision about what types of multimedia to use (if any) is an involved process. You must think about the pros and cons: Essentially, you're weighing functionality against the technological constraints of your audience and your own ability to build and maintain the piece. You have to consider your individual situation on each project. To help you decide, consider the following:

- ◆ **Your current resources:** Do you have software to create multimedia elements? Can you purchase the software? After you purchase software, do you, or anyone on your team, know how to use the software well enough to create your project? Is it feasible to call in a specialist if you don't have the software and skill set available to you?
- ◆ **Target audience:** You must know whether the site's visitors appreciate multimedia elements or find them annoying. Are you targeting individuals with newer machines and good connections to the Web, or are you trying to target a more general audience? Targeting broader audiences generally means that you should stay more conservative in your multimedia methods and deliveries and pay special attention to file sizes and download speeds.
- ◆ **User preferences:** Everyone benefits from different types of experiences — differently. Some people get a lot out of charts. Others prefer text. Some users appreciate graphics or video to help them understand the content. Not all users are the same. Sometimes your best bet is to make your information available through a variety of methods. For example, post a video for those who appreciate visual and audio information, but also include a text version for those who prefer written text or aren't able to use the video content.



In general, users appreciate your site more if you create navigation that allows them to move freely through the content as much as possible. The Web offers users the unique opportunity to experience your content their own way. No two users will take the same path, and any user can choose different paths on return visits.

- ◆ **The speed of multimedia evolution:** Make sure that if you're trying to target an audience that generally has older machines, you create multimedia that uses older versions of browser plug-ins. This also helps you make sure that the piece won't require too much processor power or provide alternate content for those visitors like a nice graphic and some text. It won't be as fancy, but at least your visitors won't be left out entirely. Just remember, if you use the latest and greatest in multimedia on your site, older machines may have a difficult time with it or won't be able to display your multimedia at all. That results in a bad experience for visitors and ultimately is not good for your site either.
- ◆ **Number of versions of the project you can create:** You must decide whether the benefits of offering pieces with different specifications are worth the extra time and effort required to build and maintain sites that have more than one version of multimedia elements. You go through the initial process of creating the project only once, but then you have to *export* (publish) different versions of the project. Creating the different versions can be time consuming because creating each version is a process of trial and error, to a degree. You can develop some good starting points (as far as settings to make), but each project is a little different and the results are a little different. You have to work at it until you get the right balance between quality and performance.



To provide different versions, export versions with different plug-in and compression settings. The basics of compression are *the more you compress a file, the smaller it gets and the faster it downloads*. The down side of that is that you also lose quality in the file. Compression is always a game of balancing quality and performance.

- ◆ **Server space:** Multimedia files are typically larger than other files. Make sure that the server you're storing them on has enough room. If you're hosting your site remotely, make sure you have enough monthly bandwidth to support the amount of traffic you expect. Each time users view your files, they use up some of your bandwidth. Hosting plans typically allow a certain amount of bandwidth per month. If you exceed the amount of bandwidth, you can incur extra charges. Make sure you have enough bandwidth to handle the amount of traffic you expect, and find out how much it will cost if you go over.

If you opt for multiple versions of multimedia files, you have to upload those different versions to the server. That uses up more server space. You'll also have to provide ways to navigate to the alternate versions of the multimedia elements. Each version needs an HTML document so that it can play in visitors' browsers. You want to make the user's experience as seamless as possible. Plan navigation elements in advance, not as an afterthought.



Making multimedia elements available to all users

A target audience using older machines or slower connections doesn't mean that you can't use multimedia elements. It does mean, though, that you have to consider how you can make your content work for them. If you need to make your multimedia more efficient, consider these options:

- ✓ **Use a photograph instead of video.** You can slowly move the photograph to give the impression of a video element (the Ken Burns effect).
- ✓ **Use vector graphics in your animations.** If you use Flash to create animated elements, make sure you take advantage of vector elements and their ability to reuse elements. See Book V, Chapters 1–3 for more information about using Flash.
- ✓ **Download only parts of your multimedia as they're needed.** You can use Flash to download only the pieces of your multimedia

that are needed. You can also plan your files so that they play parts of the presentation while more is still downloading (*progressively downloading*).

- ✓ **Publish your files to an older version of the player.** For example, instead of using the latest features of QuickTime, opt for fewer bells and whistles and publish the file so that older versions of the player can use it. This also works in Flash; publish to Flash Player version 6 or 7 to accommodate users that might not have updated machines.

If you know that a larger number of your audience has updated machines and better connections, feel free to create content that plays best using the latest technologies. Another option is to make multiple versions available. As always, weigh the benefits against how much work it will take to create and maintain multiple versions.

Choosing the Right Delivery Method

You can create Web sites made up of all static pages, a few templates created to work with dynamic information, or a combination of the two. You can use multimedia elements throughout the site on either static or dynamic pages. Figure 1-5 shows a page that has multimedia elements on a static page.

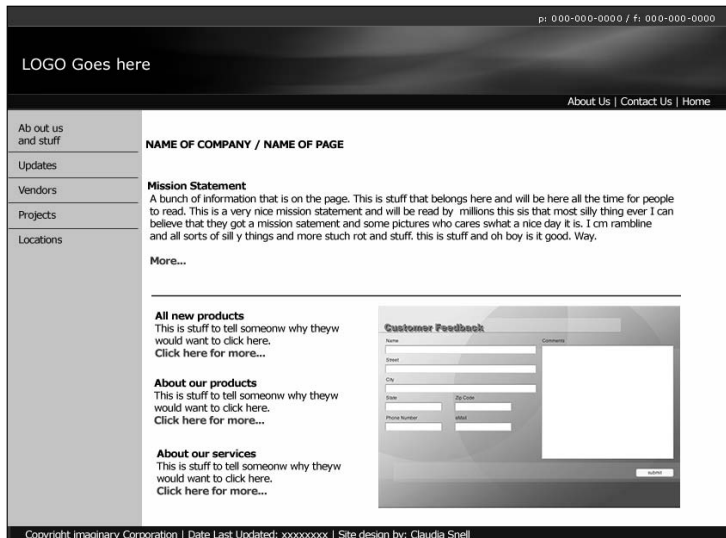
Knowing when static pages suit your purpose

Static HTML works well for information such as directions to your facility and general contact information. If you intend to post a large number of contact pages, consider a dynamic solution.

Pages with content that doesn't need to change very often are also good candidates for static HTML. Make sure that the content appears in only a couple of places. Because the content is embedded directly in the HTML, you have

to update each file separately to make sure they're consistent. If you use a *WYSIWYG* (what you see is what you get) editor to create pages, you can use a Find and Replace feature.

Figure 1-5: Static HTML and multimedia work together to create a total experience for your users.



Using dynamic pages to manage complex information

Manage large amounts of information that the Web team or visitors will need to search by using databases and dynamic pages. Sites with content that changes often also are good candidates for dynamic pages.

Using a database and dynamic pages to deliver your content also makes content easier to repurpose. With static pages, the content is embedded directly in the code. Dynamic sites hold all their content in databases; therefore, more than one location can use the same content at the same time. You can have more than one site using the database, or you can enable more than one section of your site to display the same information. Because the information is all coming from one source, it's consistent. Static pages run the risk of becoming out of synch with each other if information is updated on one page but not on another.

You have several choices for delivering multimedia elements on your site. The most common ones are Flash, QuickTime, Windows Media Player, and Shockwave. The right choice depends on the content you're delivering and the target audience. Each option can deliver video, audio, and animations to your users. We discuss things to consider when choosing what type of multimedia to use in the next section.

Using Find and Replace in Dreamweaver

The Dreamweaver Find and Replace feature is a very useful tool when updating pages on your site. This tool allows you to simply locate instances of a specific piece of text or code and replace it with something else. (In Book III, Chapters 5 and 6 contain information about using Dreamweaver.) To use the Find and Replace feature in Dreamweaver, follow these steps:

- 1. In the Code view window, select the piece of text or code you want to find in another place in the document.**
- 2. Choose Edit⇨Find and Replace.**

The Find and Replace dialog box opens. It has your selected text and/or code in the Find field.

You can decide to search just the selected text, the current document, the whole site, all open documents, selected files, or a selected folder of the site. You have the choice of doing a *Find* (Dreamweaver searches for the term) or a *Find and Replace* (Dreamweaver replaces the found term with the term you enter in the Replace field).

- 3. Enter the text and/or code that you'd like to replace the current selection with in the Replace field.**
- 4. Click the button to the right that represents the function you'd like to perform.**

Your choices are

Find Next: Moves the cursor to the next instance of the term you're looking for.

Find All: Finds all instances of the term you're looking for and displays a list in the Results box, which launches automatically when needed.

Replace: Moves the cursor to the next instance of the term you're looking for, deletes it, and replaces it with the term you typed in the Replace field. Then it hops to the next instance and awaits your next instruction.

Replace All: Replaces all instances of the Find term with the Replace term.

Close: Closes the dialog box when you're done.

Warning: Before you do a Find and Replace All, you should be very sure that you really want to make the change. This is especially true if you choose to do a Find and Replace that involves files that aren't open. You can't undo the changes you make to those files.

Remember that global Find and Replace operations can sometimes create undesirable results. In cases where the selected text to be replaced occurs in locations that you hadn't thought of, you can damage your content or your code.

If you're unsure about the impact of a Find and Replace action, use Find All first, and then individually select the instances from the list that you want to change.

Considering multimedia

Multimedia can be a good way to make very boring information more interesting. You can use animations to establish information or to clarify your point; animations can enforce your message and clarify your information.

When deciding whether to use multimedia on your site, however, you must consider your time, your resources, and whether there will be a good return on the investment. Also, make sure that the content itself is better presented as multimedia. Some content is not well suited for multimedia treatment. Think about how using multimedia enhances your information; see the “Evaluating multimedia element choices” section earlier in this chapter.

Handling Printable Materials

Sometimes you want to allow your visitors to print or download materials from your site. You have different ways to handle this task.

For smaller sites, it’s fine to create PDFs from your printable documents, post those PDFs to your server, and provide links to them. This method is effective if you have only a few printables that don’t change very often. We don’t recommend posting Word documents. They’re too easy to download and change. PDFs allow you to control the content of the documents.

You can write CSS that instructs the browser how to display the page on-screen and also instructs the browser/printer how to print the document. CSS is generally thought of as a tool to control your layout on the screen, but that’s just one thing that CSS can do. You can specify different media types for CSS to control. One of the options is Print, which controls how a page prints. For more about working with CSS, see Book III, Chapter 3.

Remember all your visitors while building

In addition to common browsers and printers, some other types of media that CSS can work with are

- ✓ **Screen readers:** This is software used by visually impaired users; it reads aloud the pages.
- ✓ **Braille devices:** These devices translate the pages into Braille.
- ✓ **TTY & Handheld:** This is for devices with limited display capabilities, such as terminals or Web-enabled handheld devices.

- ✓ **TVs and projectors:** This makes pages usable for television monitors or on-screen projectors where scrolling and resolution may be issues.

Information and a more complete listing of media types are at the World Wide Web Consortium (W3C) Web site (www.w3.org/TR/CSS21/media.html). The W3C is an organization that develops recommended standards for Web coding technologies.

To create a style sheet for print, follow these steps:

1. Create a screen CSS file.

A *screen CSS* is the same thing as CSS without a media type specified.

2. Create a print CSS file.

Make sure that the names of your selectors are the same in both files. There will be different values for some of the selectors depending on the intended use of the CSS, but they both need to have the same names to apply styles to. Otherwise, your code can become very messy with code that is intended for use by different style sheets.

3. Link to the CSS files and give them the correct media types, as follows:

```
<LINK href="myPrint.css" rel="stylesheet" type="text/css"
media="print">
<LINK href="myScreen.css" rel="stylesheet" type="text/css"
media="screen">
```

Using the `media` type tells the browser which style sheet to use for printing and which one to use for displaying the page in a browser. There are many different values that are recognized for media type. For your purposes, you need to know only about `print` and `screen`.

4. In the HTML, create the layers and name them.

More information about how to create HTML documents and apply styles to elements is in Book III, Chapters 2 and 3.

To use the print and screen style sheets together in a document, follow these steps:

1. Type the following three lines of code for the CSS for screen display:

```
body{color: #fff; background-color:#000;}
```

This creates a black page with white text

```
h1 {color: #ccc; background-color:#999; }
```

This makes the `h1` headings display as gray on gray.

```
#banner{background-color: #009; color: #ccc;}
```

This code instructs the browser to display the area of the page marked as a banner with a dark blue background with gray text.

2. In the HTML, include the following code:

```
<div id="banner">This is my banner area</div>
```

3. In the CSS document, include this code:

```
body{color: #000; background-color:#fff;}
```

This creates a white page with black text

```
h1 {color: #000; background-color:#ccc; }
```

This makes the h1 headings display as black text over a light gray background.

```
#banner{display: none;}
```

This last bit makes the banner area in the HTML document disappear. This also works for elements that you want to display in the printed document but not on the Web page. Just use the technique in reverse.



This method lets you make sure that the content on your site prints nicely for your visitors. It also ensures that the content in the HTML document matches what your visitor downloads. With PDFs, there is the possibility that the content on your site won't match what's in your PDF. Depending on what your content is, that might be all right — or it might not be.

Creating printable style sheets

Printable style sheets are cascading style sheets that are created for controlling the layout of Web pages when users print them from their browsers. If the owners of the site haven't included instructions for printers, the user gets everything that's on the page delivered how the printer decides to interpret it. A *printable style sheet* allows you to instruct the printer to print your page how you want it to be printed. You can even instruct the printer to ignore or replace areas of your page with more printer friendly versions, such as the banner graphics. You can create a special header that is invisible in the browser but replaces the banner when the visitor prints the page.

Printable style sheets are relatively easy to develop and implement. From a user perspective, printable style sheets make it easier to print a page. Visitors can choose File⇒Print in their browser to skip clicking a special button and awaiting a new version of the page. Here are two good reasons to create printable style sheets:

- ✓ **Printable style sheets make things much easier on your visitors.** And they show consideration for their ink cartridges. Those are things that add to the overall good (or bad, if you don't do them) impression that visitors have of a site. Not all sites need printable CSS because not all of them have content that people will want to print.
- ✓ **Creating printable style sheets means you won't have to create or maintain separate documents for your visitors to download and print.** Webmasters use various methods for producing printable versions of their pages. Some use coding or scripts to strip out the elements that aren't printer friendly (such as navigation). That process requires a visitor to click a Print This Page button that sends a request to the server to create and deliver a page that has the important content but strips things such as navigation, advertisements, or graphics the user won't want to print.

Chapter 2: Creating Effective Layouts

In This Chapter

- ✓ Identifying and organizing page contents
- ✓ Including essential Web page elements
- ✓ Emphasizing important information
- ✓ Making pages usable and accessible
- ✓ Creating wire frames

This chapter helps you begin designing the pages for your Web site. Before you proceed with the page-level design process, though, you need a clear idea of what your pages should include. (If you want help defining your project, check out the information about this important step in Book I, Chapter 3.) You also need a *site map* that shows how the pages you're designing relate to one another within the site.

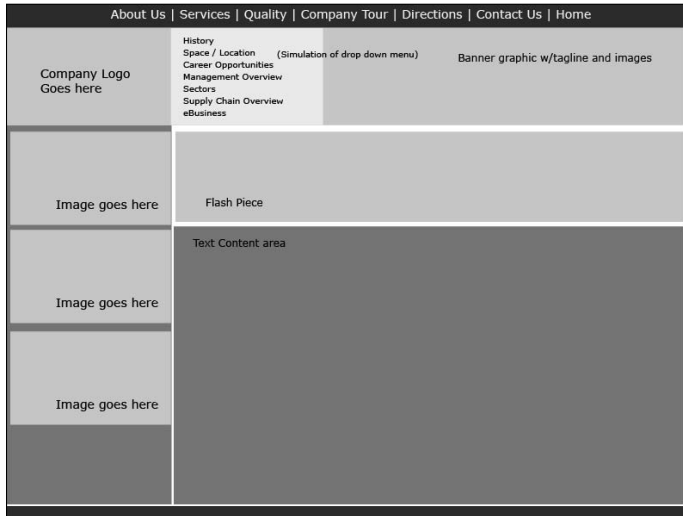
Creating wire frames for page-level layouts is an important and helpful step in the process of building a Web site. *Wire frames* are working sketches of the page-level layouts for your site. The process of creating these sketches gives you an opportunity to think about how the elements of your site (the pages) work together regardless of details such as specific colors and photos.

In addition to showing you how to create wire frames, this chapter gives you basic design rules, tells you what essential elements to include, and advises you how to fashion an accessible design.



Wire frames shouldn't include finished art. The point of creating wire frames is to make sure that the elements you want lay out on your pages in a useful and aesthetically pleasing way. Figure 2-1 shows a sample wire frame; notice that the sample represents page elements but not completed artwork.

Figure 2-1:
This wire frame shows where we plan to put the logo, text, and other elements we want to include on the page.



Content Is Key: Making Sure It's All Included

In Book I, we discuss a few important steps that make your project go more smoothly: listing goals, evaluating potential users, and so on. Here, we help you decide what content you need to include. Planning a Web page layout requires that you consider all the types of information the pages need to accommodate. You might want to include tables, lists, headers, paragraphs, images, and multimedia elements — just make a list of them all. Look at all the content you collected and determine the best way to present each piece. Also, you can make a list of content that you want to add later and include these elements in your list. Figuring out now how you'll eventually handle everything saves time later.



Consider what would happen if you were building a house. If halfway through the project, you figure out that not all the rooms are big enough, you're stuck with a lot more work when you try to rework what has already been done to accommodate the new requirements. The same thing applies to a Web site. If you know that you would eventually like to add some graphs, for instance, figure out during this phase how you would like to display them and where they will fit into the site. That way, when the time comes, you can quickly add the new content instead of having to shoehorn it in somewhere.

Creating Order from Chaos: Consistency Is Your Friend

Always remember that consistency is your friend. Being creative doesn't mean that each page of a site becomes a freestanding document with its own design. Apply creativity on a global scale with the entire site being treated as a whole, and consider both sides of your site. You must include both the *front end* (the stuff your user sees) and the *back end* (all the files that make it work) during this phase and the site-map-building phase. (See Figure 2-2 for a sample site map.)

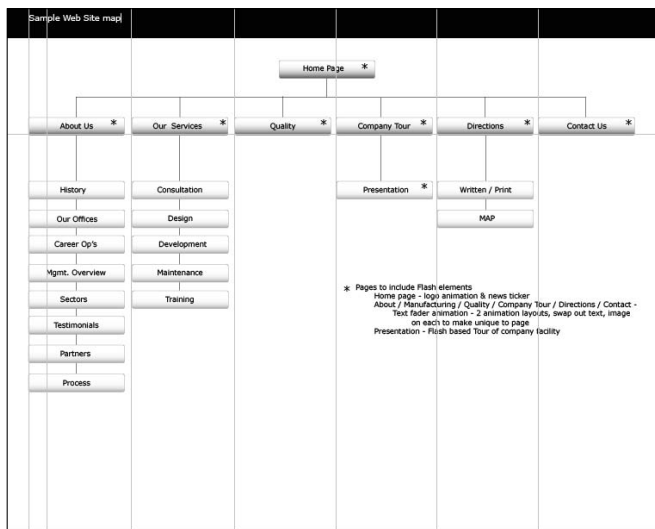


Figure 2-2: A site map shows the structure of a Web site.

- ◆ **On the front end, consistency involves a cohesive and uniform layout for all pages.** Banner, global navigation, and main content areas need to be located in the same places on every page. (See the later section, “Including Things That Every Page Needs,” for more about the banner and other necessary elements.) Using the Web site shouldn't feel like a task in itself; rather, the site should be a transparent delivery system for your information. If a layout doesn't working for all your content, rethink your layout.



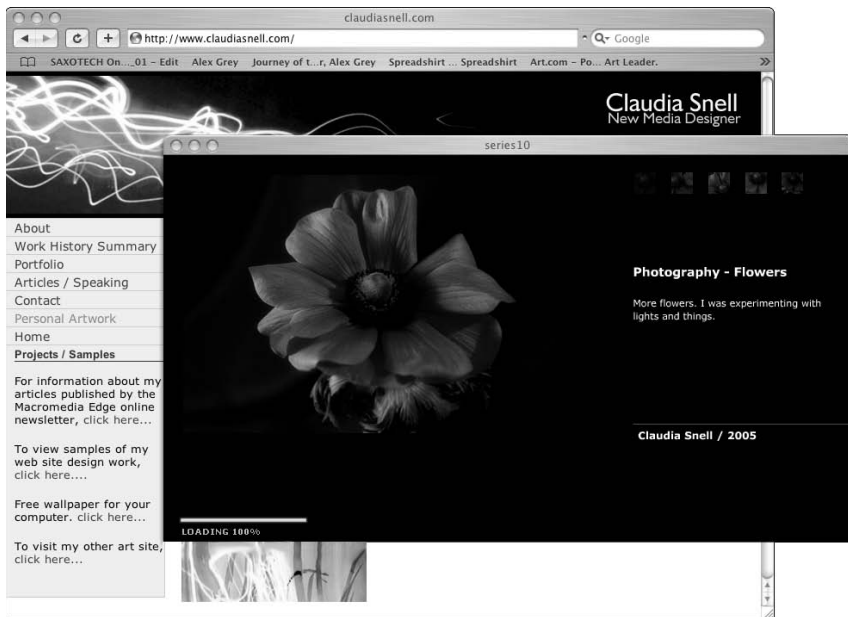
Like with anything, here are a couple of exceptions to the rule — in this case, the “always be consistent” rule:

- *Creating a home page that has a different layout:* If you choose to do this, all the elements should have a common look and feel to the rest of the site. Navigation elements should have the same style

throughout the site so that users understand that they are used for navigation. The names on buttons, colors used, and style in general must be consistent throughout the site.

- *Launching multimedia elements into a separate, smaller window:* If you do this, make sure the content has a consistent look and feel with the rest of the site. Figure 2-3 shows an example of launching a multimedia element in a new window. Note how the elements in both windows share a common look and feel. This signals to users that, while different, these pieces are part of a whole.

Figure 2-3: Sample of a Flash piece launched into a secondary browser window showing a different layout but a consistent style with the rest of the site.



- ◆ **On the back end, consistency involves employing file organization.** The back end of your Web site comprises the collection of documents, images, media, and other elements that make it all work. Any project, no matter what the size, should have a planned file structure. Keep images together in a folder; group content documents in a logical way. Develop a standard way of naming files, explain the naming convention to other team members, and stick to it. Such tasks might take you more time in the beginning, but as the site grows, you'll benefit from huge time savings in maintenance. Sharing the work with others, if you need to, is also easier if file organization is implemented and adhered to.

Staying Web legal

The term *Web legal* has nothing to do with the law. It simply refers to filenames that work on the Web. Page names with spaces and characters in them might cause problems or not

work at all. Names that are too long also can cause trouble. In other words, avoid calling a file "company "picnic" with lots of great food & drinks.jpg".

Whatever naming conventions you create, make sure that you and your team members stick to them, and especially make sure that your system is logical, easy to follow, expandable, and *Web legal* (that is, use filenames that work on the Web). Avoid vague names, such as `IMG001.jpg` and `pagetwo.html`. Such names end up causing confusion because they're not descriptive. Maintenance can become difficult when filenames give no clue about what they are.



One common way of labeling folders for Web documents is to match them to the major navigation areas. For example, if your global navigation includes About Us, Our Products, and Services links, your file structure can include folders named About, Products, and Services. All Web documents pertaining to those areas are then organized by how they logically appear. In a similar fashion, include information on the type of graphic (such as a button, icon, thumbnail, and so on) and, sometimes, the size dimensions (for instance, 50 x 20) in a graphic filename. For example, you can name the file for your About Us button `about_but_50x20.gif`.

Including Things That Every Page Needs

You want to include some elements on every page of your Web site. These things should be located in the same place on each page (except in the cases we mention earlier: an alternate home page or multimedia layouts), and should have the same look and feel on every page. It isn't helpful to users of your site, or to your organization, to express your creativity by changing things on each page. First-time Web designers often make this mistake.

The short list of must-haves we include in this section might seem to you like common sense. Surprisingly, though, many sites fail to include the most basic page elements on every page. Don't let it happen to you. In Figure 2-4, all the important page elements are represented. Note how the site's name and organization are clearly displayed. Navigation is obvious, as are highlighted features. Including these items helps your users to understand your Web site.

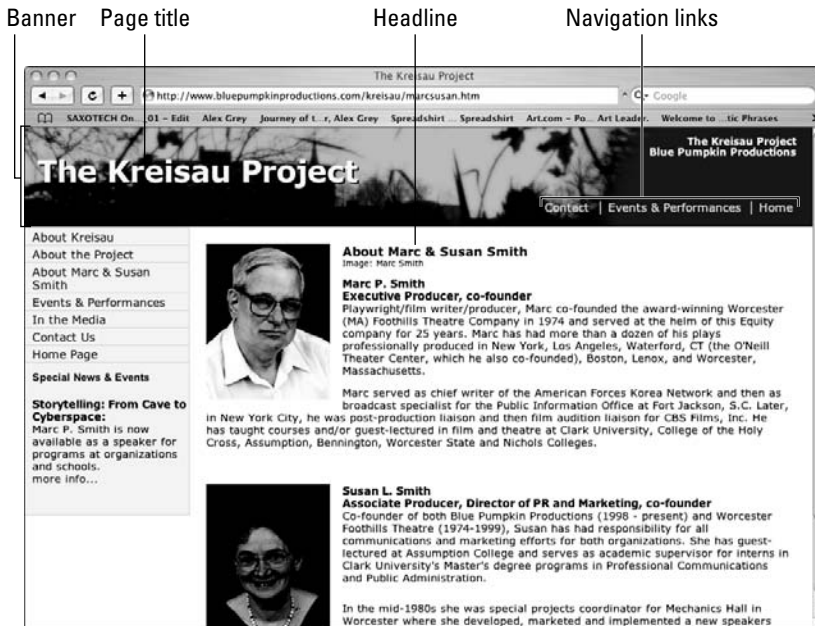


Figure 2-4: This Web page has all the necessary stuff, including user-friendly navigation, a banner, and a page headline.

Banner

Adding a *banner* (a headline graphic that identifies a site and establishes a look and feel) to each page lets users know that they are on the same site, even though they've navigated to different pages within it. For example, in Figure 2-4, the graphics, logo, and links at the top of the page appear on every page of the site that you navigate to. The banner conveys such information as branding and the name of the organization and/or site name. It gives a page a finished look and grounds the design. It's also a good place to put some contact information, a link to a site map, or a search function.

Page headline

The *page headline* sets user expectations of what they'll find on the page. It also helps users orient themselves on your site. If they're looking for information about your products, for example, a page headline with the name or type of product lets them know that they're in the right place. In Figure 2-4, the headline "About Marc & Susan Smith" lets users know that they can find biographical information on that page.

Global navigation

Every site has a collection of major sections. Whether those sections are made up of one page each or many, *global navigation* is how users access

those areas. In Figure 2-4, clicking any of the links on the left side of the page takes you to one of the primary areas of the site. This same global navigation feature appears in the same place on every page of the site.

Larger sites certainly need *supplementary navigation*, and so a larger site structure can have multiple levels. For example, the user starts on the home page, and clicks About Us from the global navigation, which takes them to the main page of that section. On that page, the user finds some basic information and a second navigational area that allows her to dig deeper for more specific information: *secondary navigation*. You might even have to provide subnavigation within those pages: *tertiary navigation*. If you're designing a site that requires supplementary navigation, make sure you place that secondary (and tertiary, if you need it) navigation in consistent place(s) on each page. The secondary and tertiary navigation should also share a common look and feel throughout the site.

Page title

The page title isn't the same thing as the headline. The *page title* is information that needs to be included in the head section of the HTML. You can find information about how to create and implement these in Book III, Chapter 2. This text shows up in the browser window and also serves as the link text in a search engine, among other things. Make sure you include a page title on each page of your site. In Figure 2-4, you see that the page title `The Kreisau Project` appears at the top edge of the browser above the Web address field.

Page description and keywords

Other information that you should add to each page include page descriptions and keywords. Descriptions and keywords are important to your site, although it's easy to overlook them because they aren't visible on the actual page.

The *page description* is just what it sounds like: a description of the page's content; it's what shows up in search engine listings. Page descriptions give users a quick summary of what your site is about. Keep the description short but informative. Short is important because of space constraints in search engine listings; only the first sentence or two will be visible. You can't see the page description in Figure 2-4, but here's what it looks like in the HTML:

```
<meta name="Description" content="The Kreisau Project:  
Production to result  
in a book and a TV series telling the remarkable history  
of Germany&#8217;s  
von Moltke family and its family estate at Kreisau." />
```



Be considerate of your users' needs. If your description isn't accurate, users will leave your site to find one that delivers what they need. Also, avoid marketing rhetoric. You can find more information about adding page descriptions and keywords to your documents in Book III, Chapter 2.

Keywords are the terms that you expect your users to search for your site with. They are the words that your users think of to describe your product, service, or organization. If your users would call your organization a *daycare* or *preschool*, put that in the code, even if the correct industry term is *child care center*. Keywords are most effective when they reflect accurately the content of the page. If, for example, one of your keywords is *puppies* but the page is actually about *tractors*, that keyword isn't effective because a Web surfer is directed to a page they're not looking for. Search engines use a combination of keywords, meta tags, content, and other factors to determine the relevance and placement of a page in the listings and to generate useful links to the information people need.

Here is an example of keywords:

```
<meta name="Keywords" content="kreisau, krzyzowa, vonmoltke,
    von moltke, freya, germany, anti-nazi resistance, marc
    smith, blue pumpkin productions " />
```

Here is some code you can use to implement some sample page information:

◆ **Page title:** Claudia Snell:: New Media Designer

```
<title>Claudia Snell:: New Media Designer</title>
```

◆ **Keywords:** Web design, flash designers, Website, new media, streaming, presentations, online demonstrations, video, worcester, massachusetts

```
<meta name="Keywords" content="Claudia Snell, Web
    Design, Web Development, Flash, Flash Developers,
    Website, new media, streaming, e-learning,
    presentations, online demonstrations, video,
    worcester, massachusetts, claudia, bousquet, jason,
    snell" />
```

◆ **Description:** Claudia Snell is a new media designer based in Worcester, MA. You may notice that each of these bits of code include "meta name=" and then a property. This bit is telling the browser that this is *metadata* (information about the document, not content) and what sort of metadata it is. In the following example, it's a description of the page content.

```
<meta name="Description" content="Jason and Claudia
    Snell: Digital media production and design.
    We create web sites, video, motion graphics &
    interactive media." />
```

Put the code for your page information in the head section of the HTML code. See Book III, Chapter 2 for more information about the parts of an HTML document. Figure 2-5 shows how the content is included in the code.

Notice that the description is in the form of a sentence, the keywords are single words or phrases separated by commas, and the title is the name of the site or page.

Figure 2-5:
This is how meta tag information looks in a block of code.



```

Source of http://www.jcnewmedia.com/
<?xml version="1.0" encoding="iso-8859-1" ?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Jason and claudia snell :: new media designers</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<link rel="shortcut icon" href="..//icon4.ico" />
<meta name="Keywords" content="Web Design,Web Development,Flash,Flash Developers,Website,new media,video,streaming,e-commerce, e-learning, presentations,online demonstrations, video,production,post production, animation,worcester, massachusetts,claudia, bousquet,jason, snell" />
<meta name="Description" content="Portfolio site of Jason Snell, motion graphics/video production and Claudia Snell, new media designer." />
<meta name="revisit-after" content="14 days" />
<meta name="ROBOTS" content="index,follow" />
<meta name="copyright" content="©2008-2004, Jason and claudia snell" />
<meta name="publisher" content="Jason and claudia snell" />
<meta name="author" content="claudia snell" />
<script language="JavaScript" type="text/javascript" src="..//functions.js"></script>
<link href="siteStyle.css" rel="stylesheet" type="text/css" />
</head>
<body>
<table border="0" align="center" cellpadding="0" cellspacing="0">

```

Note: Descriptions and keywords are just one way that search engines find your site. You can also submit your site for inclusion into search engines. There are services that you can pay for and also free submission tools, such as the Open Directory Project at <http://dmoz.org>. When you submit your site, you are given an opportunity to include information, descriptions, and keywords. When dealing with search engines, remember that they don't update instantly. If you change your keywords or descriptions, expect for some time to pass before the databases catch up.

Figure 2-6 shows how some of this information is used by a browser.

Figure 2-6:
The browser uses meta tag information to display content on-screen.

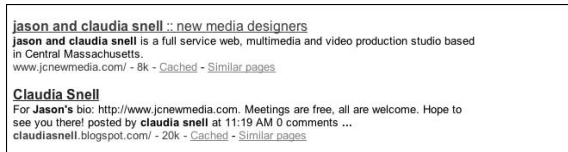




The combined effect of submitting your site with alternate descriptions or changing your site description can result in your listing being different from your actual descriptions as they are shown on your site.

Figure 2-7 shows how some of this information is used by search engines. The page title and description from the meta tags of the page show up in the search engine listings.

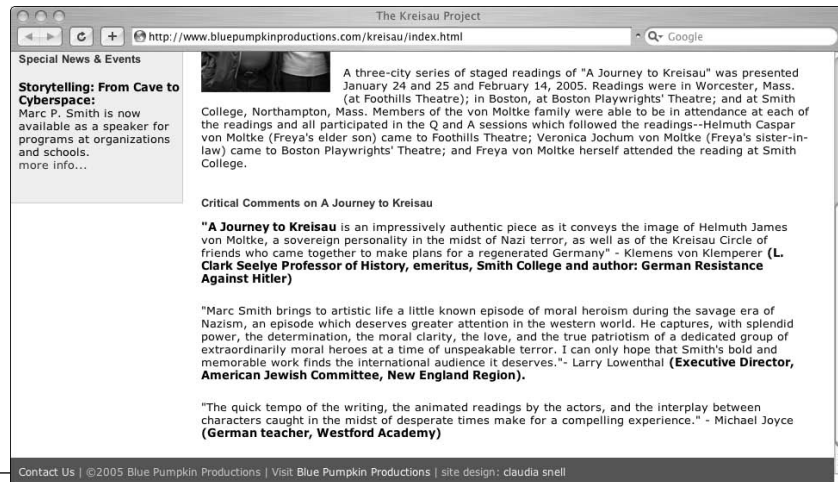
Figure 2-7:
Meta tags show up in search engine listings.



Page footer

You should put a footer at the bottom of every page, where the footer belongs. Important elements to include in the footer are copyright information, links to any legal or privacy policies you include on your site, contact information for your organization, and so on. It's also a good spot to put contact information for your Webmaster, the date the page was last modified, the global navigation mirrored as text links, and Web production credits. In Figure 2-8, you can see the footer that appears on the Web page for The Kreisau Project.

Figure 2-8:
Copyright information, the name of the site designer, and a Contact Us link appear in this footer.



Footer

Planning How to Emphasize Important Information

A well-designed Web page draws the user's attention to one key piece of information without that element looking so far out of place that it looks wrong. Planning for where and how you'll emphasize an element on your site saves you time later. You'll already know how to handle that special announcement before you have to add it. A little planning now also prevents strange or awkward-looking retrofitted elements, so keep these points in mind:

- ◆ **Emphasis is good, but your page can have too much of a good thing.** Good Web designers tend to break a page into three levels of emphasis. Use one major piece of information on the page, with the second and third levels being lesser pieces of information. Convey the emphasis to the visitor by using text size (larger equals more important) and text position (higher and in the middle equals more important). You can also use color to help convey emphasis.



When using color to show emphasis, be careful! Too many colors can make a page look garish. Another consideration is that many people might not be able to see the colors as you see them. For instance, a user might be color blind or visually impaired, so make sure you incorporate more than one cue to bring the visitor's attention to your major point of focus. We discuss usability and accessibility further in the next section.

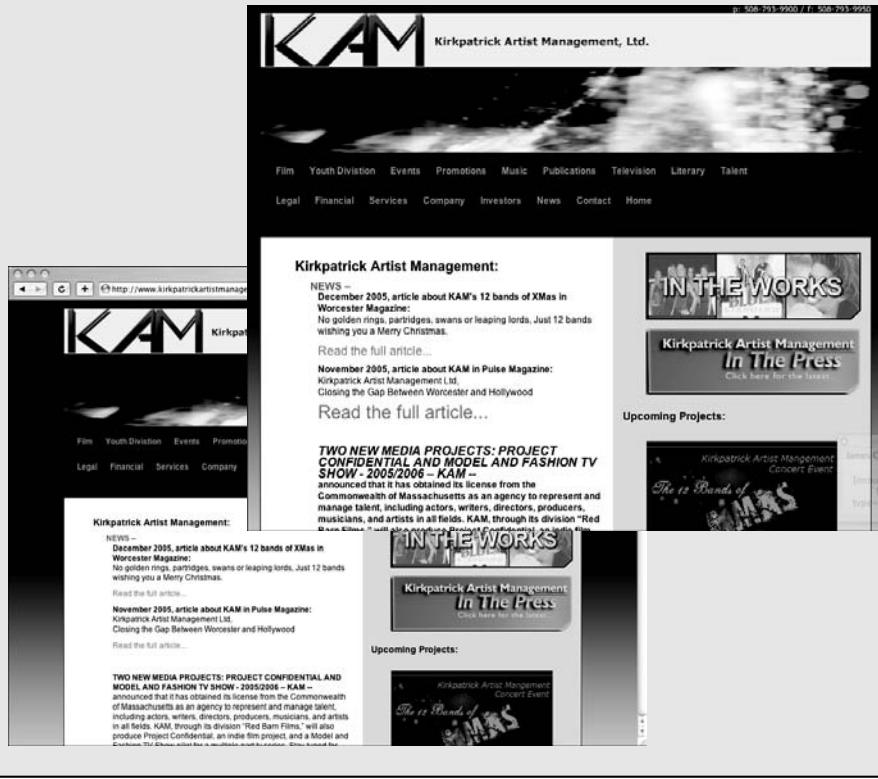
- ◆ **Be sparing in your use of special font treatments to create emphasis.** Keep the number of treatments to a minimum to avoid visual confusion and amateurish-looking pages. Keep it simple. Plan areas for headings and for content text for any special promotional items you need to include on your site. Make sure that these elements match the look and feel of your site. Don't create elements that clash with the rest of the site; it creates a bad impression for your site and organization and also confuses visitors. Remember, no one will be standing there to explain your design choices to your users. The site must convey its message all by itself.
- ◆ **Remember that nothing is emphasized if everything is emphasized.** If too many elements are fighting for attention, the site is difficult to use, and visitors are confused by too many messages that demand equal amounts of attention. This sort of thing can make users feel bombarded, and they'll miss out on the information you're working hard to make available to them — or worse, they'll leave your site and go to one that doesn't confuse or assault them.



About fonts

When choosing fonts, each color and size of text used counts as a separate font. For instance, if you have Arial text but it's in three different colors and five different sizes, count that as eight different font treatments. In a case like that, rethink your design and eliminate some of the treatments. A good rule to live by is to limit the number of fonts on a page to three. Notice how the fonts in the first image

shown here compete for attention; this site uses too many fonts, which creates a confusing layout. The second image shows the actual page, which uses only a limited number of fonts; using fewer fonts helps bring emphasis without confusing the visitors. For more information about working with fonts, see Book II, Chapter 4.



Understanding the Basics of Usable and Accessible Design

If you spend any time around Web site design projects, usability and accessibility are words that you hear a lot. *Usability* refers to how easily your visitors can use your site, and *accessibility* refers to how well your site performs on technologies other than browsers and/or for visitors with disabilities. As often happens, these are two more buzzwords thrown into a presentation to sound impressive. Unfortunately, in practice, these concepts are generally misunderstood or overlooked.

Everyone involved in a Web project knows that people view Web pages using a computer and browser. Often overlooked are the increasing numbers of other devices, such as handhelds and cellphones, that are Web enabled. And don't forget about screen readers and screen magnifiers, which enable users who are blind or have limited vision to use the Web.



From screen readers for visually impaired users to handheld devices, people are visiting sites in many different ways other than traditional browsers. You want your site to be easy for them to use.

Usability and accessibility are vital to the success of a site. If visitors can't understand or use your site, they'll leave and not return. To prevent that scenario — and to ensure usability — consider these points when planning your site:

- ◆ **All clickable or otherwise interactive elements need to work, and they all need to work as expected.** This means that if a button reads `Click Here to Search the Site`, clicking that button activates a search function — or at least brings the user to a site search function. It should not, for example, launch a photo gallery — or worse, do nothing at all.
- ◆ **A site should download quickly, and the content should be easy to get to.** In general, navigation of a site should be easy to figure out and easy to remember. Visitors don't like frustrating navigation or confusing content structures. Try to keep everything within just one or two clicks away.
- ◆ **Make sure you plan ways for your site to react to human errors.** Your visitors sometimes make mistakes, especially if you have forms or other types of interactivity. Try to make correcting an error easy for your visitors. If making a mistake throws a visitor completely out of a transaction, he won't appreciate having to start all over again; he might, instead, abandon the task and not come back.

- ◆ **Your image counts.** People do assess an organization by what its site looks like. Try to remember that visitors might have only one impression of an organization — its Web site. If it looks amateurish or inappropriate, they're less likely to trust the site. You could lose valuable visitors/customers to a competitor that has a better-looking site.

Accessibility shares many of the same considerations as usability but also brings a few more items to keep in mind:

- ◆ **Make sure that images, especially navigational ones, are properly tagged for screen readers.** The following code is a sample of an HTML image tag. For example, a screen reader that detects the code that follows reads `artwork sample` to the user. Using descriptive tags like this is vital for visually impaired users. Without them, a page can become confusing or tedious to visit, depending on how their reader deals with untagged images.

```

```

- ◆ **In addition to color coding, use a secondary method of emphasizing information.** Color coding is good for some people but not so good for color blind and visually impaired people. Make sure you group content by using positioning, headings, and other visual cues (icons) to separate content.

Common usability testing traps

Often, Web designers fall into a common trap. They think, "I find my site easy to use, and I'm a *normal* person, so the site is very usable." Or worse, they ask their friends or others that are closely involved in the project to check the site. Generally, these so-called testers don't give accurate feedback. No one wants to hurt their friends' feelings, so they simply say that they like the site. Also, people who are involved with the project are too close to it to see its flaws; they have a lot of inside information that helps them understand the site. In most cases, your target audience isn't as intimately involved with your content as you are, so your site might not make as much sense to them as it does to the team that created it.

Testing the usability of your site is an ongoing process. You should look for feedback throughout the process of planning and building your site. Correcting usability issues along the way is easier than having to rework things later.

Give testers a list of tasks like, "Find information about *Product X*" or "Locate our company profile." Observe while the tester tries to accomplish these tasks. After the user completes the tasks, ask her for feedback about whether she found the tasks easy or hard. Also, ask her why a task was difficult and solicit comments that can help you solve the issues. The point is to see your site from other people's points of view and make it as easy for them to use the site as you can. See Book III, Chapter 9 for more about usability testing.

- ◆ **Make sure that the colors you choose have enough contrast so that people can read the text.** We're surprised by how many sites sport gray backgrounds with gray text. They might look cool, but for anyone over 35, it might be hard to read.
- ◆ **Make sure your site includes site maps, breadcrumbs, and other orientation tools.** *Breadcrumbs* refer to a list of links, generally displayed at the top of a page, that shows the path to where the user is — think Hansel and Gretel. Figure 2-9 shows an example of breadcrumbs.

We cover these important topics in more depth in Chapters 1 and 3 of this minibook and also in Book III, Chapter 9.



Figure 2-9:
This breadcrumb trail shows that the user went from the Home page to the About page.

Creating a Layout Wire Frame

A *wire frame* is a page-level layout sketch — like a blueprint of the main areas where you intend to put images, text, headings, links, and so on, on a page; see Figure 2-10. You can create wire frames with traditional drawing tools (pen, pencil, paper, or crayons, if that's what you like) or with a graphics program such as Photoshop, Illustrator, or Fireworks. Claudia generally starts to work out her ideas on paper and then re-creates the wire frames in Illustrator.

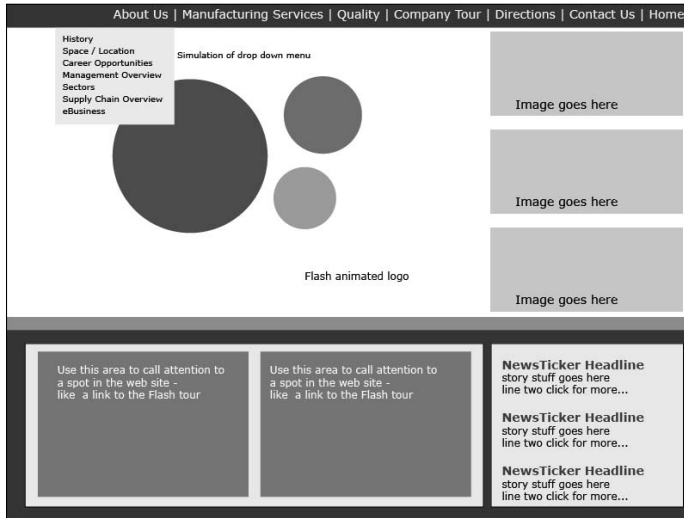


Figure 2-10: This wire frame is a sketch of a site Claudia is planning.

The number of wire frames you need to create depends on the individual needs of the site you're creating. Some sites work well with one layout for all the pages, and other sites need multiple layouts. Remember, these are just layouts at this point, not finished designs.

To create a layout wire frame using Photoshop CS4, just follow these steps:

1. Open Photoshop and choose File → New to create a new document.

The New dialog box opens. Select Web from the Preset drop-down menu and then 1024 x 768 (which is the default) as the size. (See Figure 2-11.)

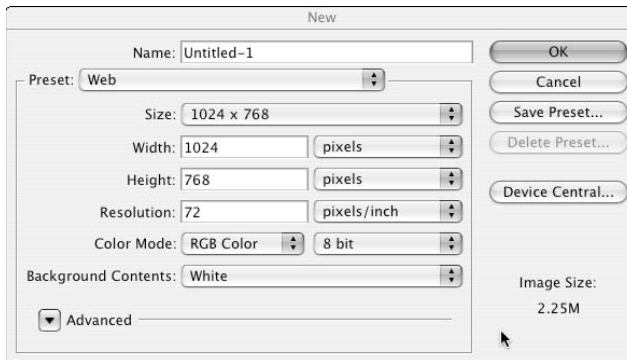


Figure 2-11: Start your wire frame here.

2. Fill in the following information and then click OK.

- *Name*: Set the name of the file to `myDocument` (or something else you like better). Because this becomes the name of the file, create a name you'll remember.
- *Width*: Set at 1024 pixels.
- *Height*: Set at 768 pixels.

The current standard Web screen resolution is 1024 x 768. You can set this number to any size you like, but by starting with 1024 x 768 pixels, you can see how your layout works at the standard Web size.

- *Background contents*: Set to white or any other background color. (You can change this later if you change your mind.) This just starts your work area off with a background color.

After you click OK, the dialog box disappears, and the new document appears in its place.

3. In the new document (see Figure 2-12), create the main element areas (banner, content, navigation, and so on):

- a. *Click the Rectangle tool on the Tools panel.*

It's the fifth tool from the bottom. Your cursor now looks like a plus sign.

- b. *Click and drag to create rectangles to represent all the areas of your page — banner, navigation, content, and so on.*

Each time you click and drag, a new layer is created.

- c. *On the Layers panel, double-click the name of the new layer and give it a descriptive name.*

If you don't, you'll end up with a bunch of layers with the names Layer 1, Layer 2 (or Shape 1, if you used the Shape tool), and so on. Unless you can remember what each layer number represents, it's hard to change your layout later.

- d. *Position the elements by selecting the layer you wish to move, clicking the Move tool, and dragging each element.*

- e. *Resize an element by selecting the layer, choosing `Edit`→`Transform`→`Scale`, and then clicking and dragging in the main window.*

- f. *Change the color of elements by double-clicking the layer thumbnail on the Layers panel.*

- g. *In the Color Picker that launches, choose a color and click OK.*

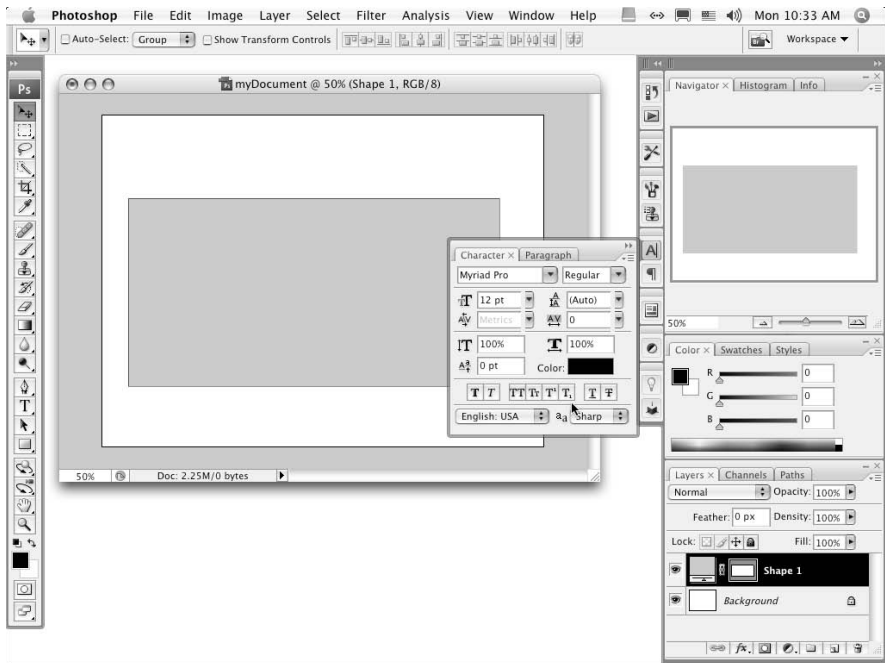


Figure 2-12:
Create
the main
element
areas in
your new
document.

4. In the main window, create placeholder text. (See Figure 2-13.)

- a. Select the Text tool.
- b. Click a location in the main window and type some text.
- c. Format the text by double-clicking the large T on the Layers panel and making changes via the Options panel or on the Character panel.

You can move text the same way you moved the layout elements: by dragging with the Move tool.

When you're finished, your wire frame should be a layout grid that will drive the graphic design and page development process.

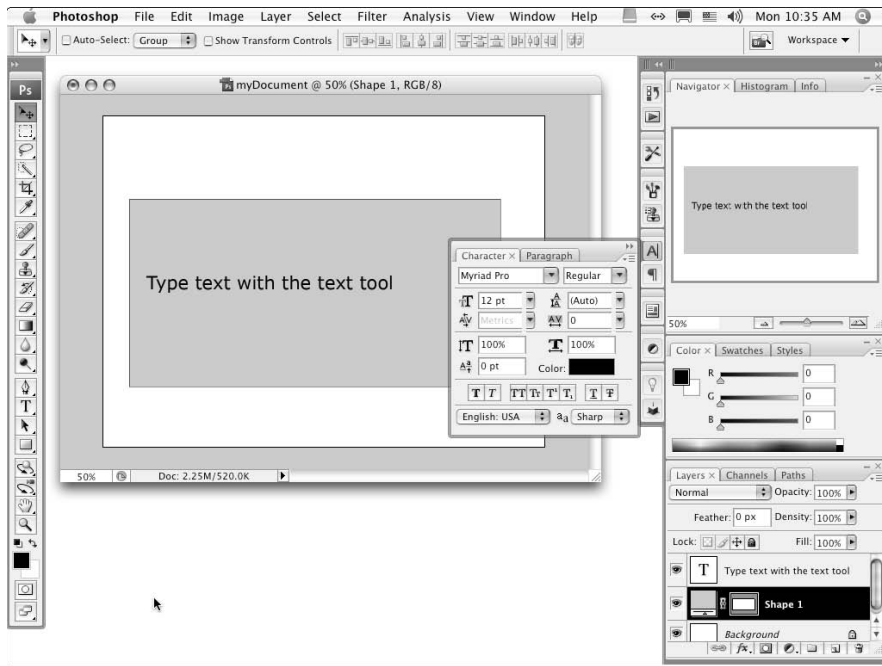


Figure 2-13: Accessing the Text tool and text editing via the Options panel.

Chapter 3: Organizing Your Content

In This Chapter

- ✓ Establishing and sharing file management conventions
- ✓ Developing Web site content
- ✓ Keeping best practices in mind

Book II, Chapter 2 helps you figure out what content to include on your site. Although, by now, you might have collected much of it, your content will evolve while you develop it. Web site content actually continues to develop throughout the life of the site. For this reason, planning and building a good organizational structure is very important, and that's why we devote this chapter to that topic. While you collect information, it's tempting to just dive right in to the artwork part of designing your site. Resist the temptation. This chapter shows you how.

Starting with the Skeleton: The Structure

The following pointers can help you organize the content (and save yourself some headaches in the process):

- ◆ **Keep all site files in one set of folders.** Typically, you have pictures, text documents, and other elements (such as PowerPoint files and Excel spreadsheets). Some of these files end up directly on your site, and others just provide information. In either case, keep the original files together in a set of folders. Keep this set of folders in the local copy of your site. (A *local copy* refers to the version of the site that's on your computer's hard drive.) You work on these files and then publish them to a Web server.
- ◆ **Do first things first.** Start with designing the structure and maintenance of your site. If you start creating artwork first, you're likely to design something that looks great but doesn't work for the purpose of the site.
- ◆ **Keep site maintenance manageable.** A good way to make sure that maintenance doesn't get done is to make it difficult for people to work on the site. For instance, if you create content that needs to be updated often, you'll need to assign the editing to people who probably already have other jobs. Or the task might fall to someone who doesn't know

how to do it, which means that you need to train that person. So if you don't think through all aspects of the site upfront, you risk ending up with a hard-to-manage site and overworked people.

One small update shouldn't be a big deal, but several can accumulate into a large project. Consider looking for ways to automate part of the maintenance or create content that doesn't need to be updated as often. The decision of what is best depends on your needs and your resources.

- ◆ **Remember that design includes the whole project, not just the pretty graphics and layout.** The *whole project* includes functionality, maintenance, usability, accessibility, graphics, layout, and expandability. You need to consider all users of the site, from content owners to the people who maintain the site to potential visitors. A beautiful-looking site that doesn't work for the intended purpose is a failed design. The life of such a design is not as long as the life of a design that you've thought through and designed well.

The rest of this chapter elaborates on these ideas and also explores the issues of file management, content placement, and best practices in terms of what's best for the site user.

Considering File Management

Even if you think file management is boring and tedious, it's necessary. Whether you're building a site that multiple people maintain or you're working alone, make sure you have a plan for managing files. When working on a Web project, you'll accumulate a lot of files — Word documents, Excel spreadsheets, images, audio, HTML, sketches or notes on paper, and so on — so developing a logical method of organizing these documents is essential to the project. Part of the file management process includes keeping copies of original files, for these (and other) reasons:

- ◆ **Reference:** Suppose you need to reconstruct a page or section of a Web site. If you back up your entire site periodically (a good practice in case of a major problem with your Web servers), you can easily backtrack and redo that part.
- ◆ **Repurposing content:** In many cases, you can repurpose existing print content, such as logos, fliers, product catalogs, and so on, to the Web site. Naturally, you want to keep copies of all those resources. With these resources at hand, you can
 - Pull up the original content quickly and make changes.
 - Keep track of the original content.
 - Build a source of information for other parts of your project.

While, for the most part, the information stays the same from print to Web, the layout changes. Maybe you need only parts of the material on a particular page. For instance, if you have a presentation comprising five slides with bullet points, you might want to condense the information to three bullet points for the Web. Two months after you launch your site, if you decide to freshen the content and give your visitors even more information, you can refer back to the original document and add those other bullet points.

- ◆ **Revision of graphics:** With most images, whether they're photos or art, you edit them, apply effects, or add them to other elements and use the finished version of the images on the Web. If you crop and prepare your photos and other graphics for use on your Web site without saving the originals, you can't revise the images as easily. Keeping a copy of the originals ensures that you can reuse the graphics with the most flexibility. To keep a copy, just choose File⇨Save As and name the file something descriptive.



When revising a file, choose File⇨Save As and change the name *immediately* after opening the file so you don't accidentally ruin your original. This technique works no matter what software package you're using — such as Fireworks, Photoshop, or Illustrator. Saving the original with a name that's different from the names of the files in-progress and the finished file ensures that you can go back to your starting point if you need to. Make sure you include something in the filenames that will help you keep track of versions. For example, append a date or version number to the filename. Create a standard for yourself and stick to it.



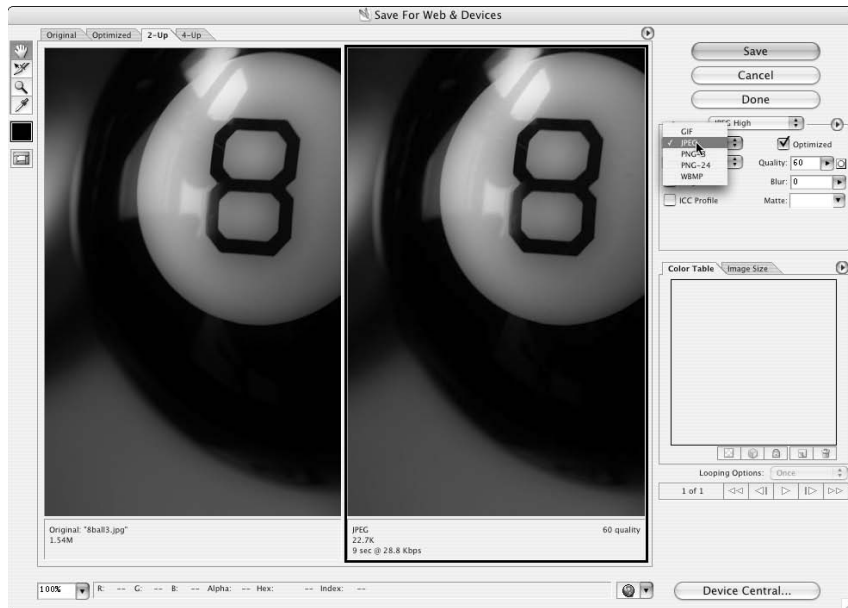
With graphics files, you typically have a working file and a final version. The *working file* is the version that has all the editable pieces still intact. The important thing to know here is that as you work in graphics software, you generate a series of layers with pieces of your graphic on each layer. One image, actually, is made up of multiple layers that you can edit. When you generate the final copy of the graphic for your site, the resulting file is *flattened* — in other words, the layers no longer exist in that file. Because the layers don't exist anymore, you can't easily make changes to the file, and some changes become impossible. In most cases, you have to open the working copy and edit the layers.

Keeping copies of these working files — along with the other support and content files — in a Production Files folder means that you don't have to reconstruct the project every time you want to make a change to the graphics. (Book IV has more information about how to use graphics software.) And keeping a Production Files folder within your local site means that you can easily find the files you want to edit.

The *final version* of a graphics or multimedia file is the JPEG, GIF, or SWF file (to name a few) that you use on your actual Web page. Put these files in your Images folder with the other site graphics. Here's an example of how you

take a working file to its final version: When you work with Flash, it generates an editable FLA file — your working file. When you're ready to publish the Flash project to your live site, you save it as a SWF file — the file that a browser with the Flash player plug-in can run. You store the FLA file in your Production Files folder so that you can edit it in the future if you need to, and you save the SWF file in your Images folder to publish on the Web site. Figure 3-1 shows the Save for Web & Devices dialog box in Photoshop; in this example, a photograph is being optimized for use on a page. By choosing File⇨Save for Web & Devices, you can prepare a graphic to use on your site by choosing a format, optimizing the graphic for the Web, and even resizing it if you need to.

Figure 3-1:
Use the Save for Web & Devices dialog box to turn a layered Photoshop document into an optimized Web graphic.



Keeping It Tidy

You need to keep your production files somewhere, and if you're working alone, it's all right to keep them within your local copy of the site. If you're working on a team, you might want to find a place that all team members can use, such as a shared network drive. That way, they can do what they need to do and keep the site consistent. It also ensures that the live server doesn't become cluttered in the process. Regardless, whether you're using your local copy of the site or a shared drive, the following points can help you continue good file management after initially implementing it:

- ◆ **Create a system for naming your files before you start building the site.** This applies whether you're working alone or with a team. If you're working as a team, make sure everyone on the team understands the naming conventions and uses them correctly. Create a document explaining the site structure and naming conventions and make sure that everyone has access to it. This document of standards helps everyone understand how to work on the site without creating a mess.
- ◆ **Write down maintenance procedures as you create them.** Attempting to figure out how you need to update something is frustrating, even for seasoned Web professionals. The task of producing consistent results becomes much more difficult when you make your teammates guess how they should do things. The techniques of keeping documents and notes aren't only for large, complex sites with lots of people working on them. Documented maintenance procedures can save you, the one-man band, when you need to get some work done on a site that you haven't worked on in a while.
- ◆ **Don't waste precious hosting space on production files.** Production files tend to be very large, so they take up a lot of space. Also, Web staff can access anything on your Web server. Files can become inadvertently available to people you don't want having your files, and that can result in unwanted changes (or worse). The best rule is to not post files unless they need to be there.
- ◆ **Archive obsolete files.** If you're no longer using a graphic on the *live site* (the published version of the site), remove it from the Web server. It's all right to create an archive on your local or shared network drive, but remove it from the live server. This is especially important for large, complex sites and sites that have multiple people working on them. Make sure that each member of the team knows that she is responsible for archiving the files that she makes obsolete. This is also true for whole sections of the site. If you intend to remove a file (or more) from the site, remove it fully from the server. Make a backup of it so you can recover it if you need to.
- ◆ **Create a flexible file structure that can handle your site now and into the future.** Most likely, you'll want to add to your site at some point. Make sure you structure your site in a way that makes that easy. Consider how much of a particular type of content you intend to have. If you think you'll have a lot of multimedia elements, put them in a folder. Start off with subcategories within that folder. It might look and sound strange when you initially launch your site to have a folder with other folders in it for only a few multimedia files, but when you start adding files, you'll see a great benefit in having planned ahead.

Implementing Information Architecture

Information architecture involves building the structure of your site from a content standpoint — that is, deciding what information to put in each section. Part of organizing and designing your site entails deciding how to present the content. Upfront, you must determine how to treat images and graphics (whether or not to use captions, for instance); how to handle text and other content (presented in tables or bulleted lists); and, of course, how to handle overall site organization (placing information where it's easy to find but not confusing or crowded).

Information architecture sounds complicated, and it can be for larger sites. Generally, though, it's just deciding what works best as a group. Organize content in groups, which become the sections of your site. For example, in the site map in Figure 3-2, you can see that this site has sections named About Us, Our Services, Our Products, Company Tour, Directions, and Contact Us.

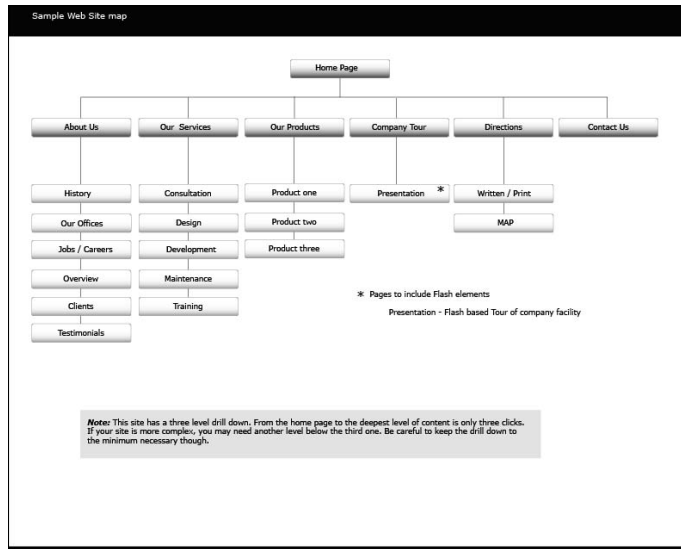


Figure 3-2:
The site map showing major sections of a site.

Those groups are then broken into smaller groups to become the page-level content. In Figure 3-2, for example, you can see that the About Us section includes a History page, an Our Offices page, and a Jobs/Careers page, among others. Organize this information in a logical way, somewhat like if you were creating an outline.

If your information architecture plan for your site is successful, site visitors (as well as your Web staff) can find what they're looking for easily. If you've done a poor job of planning the structure of your site, visitors can get lost

and frustrated. For instance, it's not a good idea to hide your contact information in a section filled with games, where your users can't find it. And remember that someone has to maintain the site after it goes live. Structuring the site in a way that makes sense also makes it easier to maintain. If things are in odd places, you'll have a harder time finding what you need.

On the back end, make your folder structure consistent with your site map. Keep the content for each section in its own folder, with subfolders (if needed) that contain the page-level content. That way, everything is easy to find when you need to make updates. The folder structure should match the site navigation. This is not to say, though, that you must create a navigation button for each folder in your structure. Some folders contain elements such as documents, multimedia, style sheets, and graphics.

Developing section-level information architecture

Like with your page-level information architecture, within your sections are main pages and subordinate pages. While you're considering how to structure your information, keep in mind that users don't like having to click through too many links. About three levels of information are sufficient. This means that from the home page of a site, a user should have to click through only two pages *maximum* to reach the deepest level of your content.

From the home page, you drive your traffic to your section content areas. Make sure your section page and secondary pages reward your users with useful information. Make sure it's high-level, summary-type information so that you can quickly fill the needs of visitors. The summary approach also benefits visitors by letting them know that they're headed down the right path. For instance, on the home page, you can feature a teaser with some marketing-type language about caring for house plants. Include a picture of a beautiful plant with an enticing blurb about what the visitor can find in the House Plants section of your Web site. End the blurb with a `Click here to get more information about caring for house plants (or similar)` link.

The included link, naturally, takes the visitor to the House Plants section and the promised information about caring for house plants. Also, from these secondary pages, you can include links to deeper information about the topic. With the house plant example, you'd link to main pages about specific types of plants or groups of plants (such as cacti or ivy).

So, still using the example of the House Plants section with a page about cacti, you can include general information about cacti and other succulents and add links to a specialized page with specific information — say, a customized aloe vera plant page. If you have a large amount of deeper information, you can offer a bit more content on this page than on the upper-level pages. You might also choose to deliver the more detailed information as downloadable documents so that users can download, print, and read at their leisure.

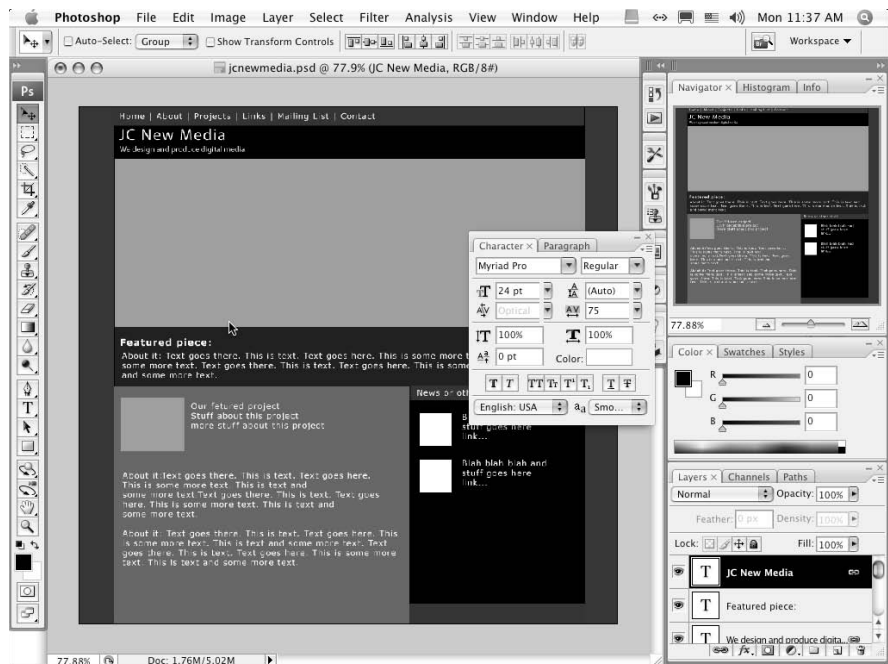
This method of structuring content produces a *drill-down*. Users drill down from the summary on the main page through successively deeper information until they get to the level they need. This makes your site more useful to a wider variety of viewers.

Arranging page-level information

The page layout phase of site design is easiest if you first organize your content. That way, you know how much information you need to accommodate on each page. When deciding how much information to include on a page, remember that visitors get frustrated with pages that scroll excessively. If a page becomes very long, take a second look at your content to see whether you can reorganize it or break long lists into multiple lists. Alphabetical order works well for most topics.

When planning how you'll arrange content on a page, make sure that you design your pages so that the main point of the page is obvious. Lack of focus can create a confusing situation for your users. Lead them to what you want them to see by placing the main idea toward the top of the page so it appears within the browser window. In Figure 3-3, for instance, the main content goes in the large space below the names JC New Media. Tertiary information is best placed on the page below everything else. That way, if users don't scroll to the bottom of the page, they're not missing the important stuff.

Figure 3-3:
Page layout becomes much easier when you group your content first.



Avoiding Content Problems

Other important things to remember about site structure and information architecture follow:

- ◆ **Make each page deliver real content.** Web site visitors should find that even pages that are at the higher levels — and therefore, made up of more shallow, summary information — provide useful content. *Shallow* and *summary* shouldn't mean *useless*. Sometimes, useful information is concise — such as, [Click here to learn about cacti!](#) Simplicity is appropriate if you know your users will understand what they'll get when they click a link.

If you think your higher-level information needs a little explanation, include a short paragraph to get users interested. There are no truly hard rules with this other than you should think about what your users will need in order to decide what they're going to do after they find your site.

- ◆ **Avoid compiling lists of links.** Sometimes, you can group a few links effectively, but we can't think of very many good reasons (other than search results or possibly task lists for groups or students) that a visitor to your site should click a link only to be confronted by another list of links or — even worse — a list of links that don't include any explanations.

The worst site design mistake is a happy invitation to [Click here for information](#) that sends users to a page that invites them to [Click here for more information](#), and so on, until users get tired of clicking and not finding what they want. Users will find a different site to visit if they find that your site doesn't have any actual information, just a series of links to nothing.



In other words, make sure you allow your users to drill down to the level of information they want. Keep the vague [Click here](#) links to a minimum, and ideally, don't use them. Try to use link text that tells the user what they will get when they click it, include the link directly in your content by making the relevant phrase into a link. This will invite your users to follow the trail of information deeper into your topic.

- ◆ **Don't include unfinished work on your site.** Most people agree that clicking a link and finding an [Under Construction](#) or [Coming Soon](#) page is a waste of time. It's all right to allude to features that you intend to add to your site, if you like. This gets people excited about making repeat visits to your site. However, we advise against including links on your page to [Under Construction](#) pages or sections. If the content isn't ready to roll out, don't include it in any navigational element.
- ◆ **Announce an upcoming new feature shortly before its launch.** Frequent users notice when an announcement is up for a prolonged period. They also notice when you announce a new feature that doesn't

materialize. Sometimes, planned features just don't work out — or they aren't ready as soon as you intend them to be. Don't make promises that you might not be able to fulfill.

- ◆ **Give your visitors the ability to customize their experience as much as possible.** Simply making sure they can flow through your information in different ways makes your site more useful and interesting. Be creative about how you help visitors flow through your site. Give them options that allow them to choose their own path while also rewarding them with useful content along the way.

Look for opportunities within your content to make it easier for visitors to get what they need. If you had a site about dogs that included a page about Afghan Hounds, it would be useful to include links to a grooming tips page, to a page with information about where to buy an Afghan Hound, and to a page offering products to help care for an Afghan Hound. If all the pages relate, link them and let your visitors browse their way to what they need. Your visitors will feel helped, not “sold” on your products. It makes their online shopping experience better.

- ◆ **Avoid hiding your content under a pile of lists.** Presenting users with large lists of links with little or no explanation about what they get when they click will not produce a good user experience. You'll want to guide people through your information, gently, by including links to broader content areas within your site's navigational areas. Use links within your text to guide people into deeper information.
- ◆ **Use images and multimedia elements thoughtfully.** Don't include an element just because you like it. Make sure it fits your content. If your site is about cacti, find pictures of cacti. Include a chart of care statistics or create a short, multimedia presentation about how to care for cacti. Leave the pictures of kittens off the page about cacti. They might be cute, but an image that is there for no other reason than that it's cute makes your page look amateurish.

Especially, use Flash and other multimedia with care. Plan and develop each piece of multimedia, whether it's a Flash presentation or a video, making sure it conveys information in addition to being nice to look at. If you expect your users to wait for media to download, make it worth their time. (You can find more about developing and using multimedia elements in Book V.)



Call to action

You'll likely hear the term *call to action* when you start working with Web sites. This term refers to the practice of soliciting an overt direction from your visitors: `Click here` or `Order now`.

A call to action is generally used along with some explanation to visitors about why they would want to follow your directions. You can drive traffic very effectively by using a call to action. `Click here for sale prices`

is sure to catch viewers' attention and get them to follow the link.

When you don't use a call to action, you're leaving your visitors to decide what they'd like to do next on their own. If your content is confusing enough or if your navigational elements don't look like navigation (links that are hidden), your users have no idea what they're expected to do. They might leave the site in favor of a site that helps them find what they need.

Chapter 4: Ensuring Visual Appeal

In This Chapter

- ✓ Understanding colors on Web sites
- ✓ Getting to know JPEGs and GIFs
- ✓ Choosing font families
- ✓ Buying stock images
- ✓ Refreshing an existing site

As you start to work on the visual aspects of your site — also known as the *look and feel* — you have to make a few decisions. You have to choose colors and imagery that support your message and convey a sense of what your organization is about. In some cases, you might have to buy images to use. You might also have to work with existing printed materials. You have many details to tend to, but don't let that overwhelm you. Within a short period of time — and with the help of this chapter — getting the artwork under control will become second nature to you.

Colors on the Web

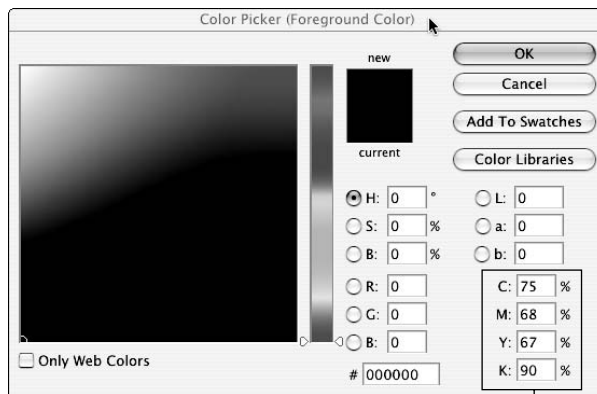
One of the first things you might encounter is a situation where an organization already has printed materials and is looking to enhance its presence with a Web site. The printed materials might have a look and feel already established and most certainly will have a logo and color scheme in place.

Here's something to always bear in mind: *Color on the Web is different than color on paper*. This might sound like an obvious statement, but it actually isn't. Color in printed materials is represented by a system called CMYK (cyan, magenta, yellow, and black). In short, it refers to the colors of ink that are used when printing. Web pages and Web graphics don't use CMYK; they use a system called RGB (red, green, blue), which is represented in HTML, CSS, and other Web scripts with hexadecimal code. More on that later in this chapter. In order to understand *why* it is important that the color systems are different, you need a bit of information about *how* the color systems are different.

CMYK color

As we mention, CMYK is the color system used to produce printed materials. It refers to the colors of ink that are used to print things. CMYK is a *subtractive color system*, in which everything starts off white. White is the color you get when all the colors are being reflected back to your eye from a surface. Subtractive colors absorb, or subtract, some of the color in the light, which creates the effect of colors. CMYK colors are represented by percentages of how much of each ink color are to be used. The Photoshop Color Picker window shown in Figure 4-1 features the CMYK settings.

Figure 4-1:
The Color
Picker,
showing
CMYK
settings.



CMYK settings

RGB color

Red, green, and blue are the primary colors of light. Because computer monitors use light to display colors on-screen, the RGB system is what computers use. RGB is an *additive color system*, which means that you add different amounts of red, green, and blue to create colors. Black is the absence of light, and so is the absence of color in the RGB system. When you add different values of the three primaries, you get different colors. RGB colors are represented as numbers between 0 and 255 — one number for each R, G, and B. Zero means that none of that color exists, and 255 is the maximum amount of the color that can exist. So, for example, 0, 0, 245 represents a very blue color (0 for the value of red, 0 for the value of green, and 245 for the value of blue).

Hexadecimal colors

Hexadecimal colors are how browsers like to interpret color. When you're using HTML or CSS code to produce colors, you will, most likely, use hexadecimal colors to do it. You can use other methods of displaying colors

(such as using color names), but hexadecimal is the preferred standard. Hexadecimal (*hex*, for short) uses six digits or letters to display colors. For example, #000000 is the Hexadecimal code for black. These digits are read as three pairs of numbers; the first pair represents the amount of red in the color, the second pair represents the amount of green in the color, and the third pair (you guessed it right) represents the amount of blue. You use hexadecimal colors for background colors, font (typeface) colors, and link colors.



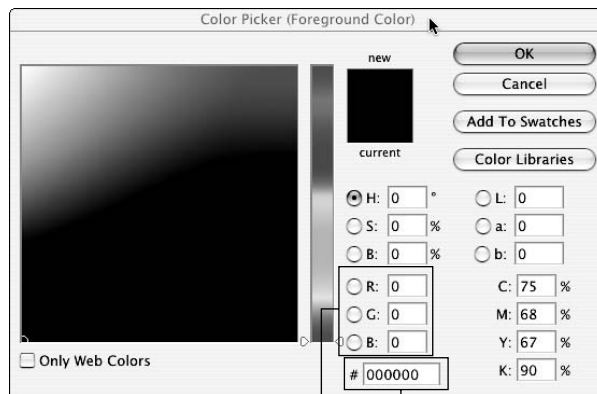
Computer monitors use light to display images, so RGB is what they read. Hexadecimal is just a way to communicate with the browser about what color you want a Web element to be.

To find out more about using hexadecimal in code, refer to Book III, Chapters 2 and 3, where we cover HTML and CSS.

Making nice with established color schemes

If you're going to design graphics for an organization that has a color scheme already in place, you must have them supply the correct RGB or hexadecimal values that correspond to the CMYK colors they're already using; otherwise, you might not get the colors right. We don't recommend "just using the eyedropper," as some print designers might tell you to do. The Eyedropper tool in Photoshop is called that because that's what it looks like. The purpose of it is to get a sample of a color within your document. It is used by clicking the area you'd like a sample of within a Photoshop document. You'll see the Eyedropper in action in the next section. The colors often shift or aren't accurate when you use that method. A best practice is to get the colors straight from the client so that you're sure they're what the client wants. In Figure 4-2, you can see the different color systems available for you to work with in Photoshop.

Figure 4-2:
The Color
Picker
offers color
system
choices.

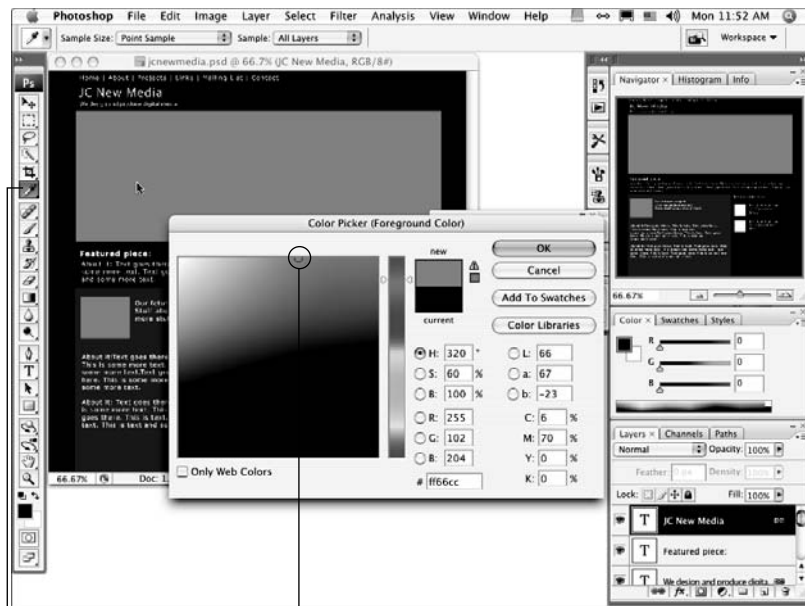


Red, green, blue
(RGB)

Hexadecimal

Despite your best efforts, though, a subtle shift in colors is likely to occur. You can't perfectly match the colors of a brochure to the colors on the computer screen. The reasons for this are that the appearance of the colors on the brochure change depending on the lighting conditions in the room, and every computer monitor is a little different. If you hold the brochure up next to the monitor and work at it, you might get them to match visually. As soon as you go to a different monitor or change the lighting in the room, though, they won't match any more. In other words, if you're trying to match perfectly, don't. You can get them close enough so they'll be consistent, and that should be the goal. The way to do that is to make sure you get the right RGB or hex colors from the client. Figure 4-3 shows using the Eyedropper tool in Photoshop.

Figure 4-3:
The Eyedropper tool's great, but get the right RGB or hex colors.



Eyedropper tool Use the Eyedropper tool to match colors.

Web-safe colors

In addition to RGB and hexadecimal colors, you hear about *Web safe* or the *Web 216* palette. *Web safe* and *Web 216* refer to the set of 216 colors that display reliably across the different platforms. The use of Web-safe colors was more important years ago when people didn't have computers and monitors that could handle millions of colors. A system was developed to make sure that colors could be displayed correctly on both Mac and Windows

computers. With today's standards for computers and monitors, it's becoming increasingly mainstream to disregard strict adherence to the Web-safe, 216-color palette. Figures 4-4 and 4-5 illustrate the difference between a Web-safe, 216-color palette (Figure 4-4) and a non-Web-safe palette (Figure 4-5).

Figure 4-4:
The Web-safe palette.

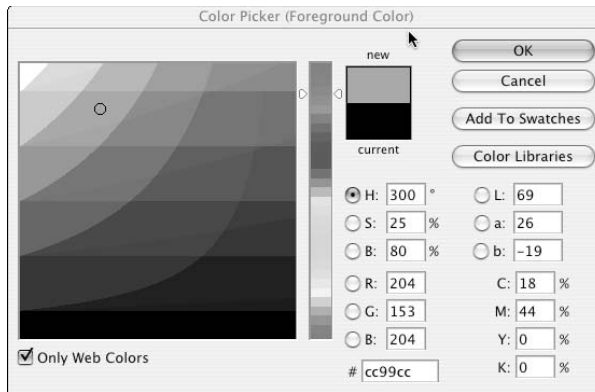
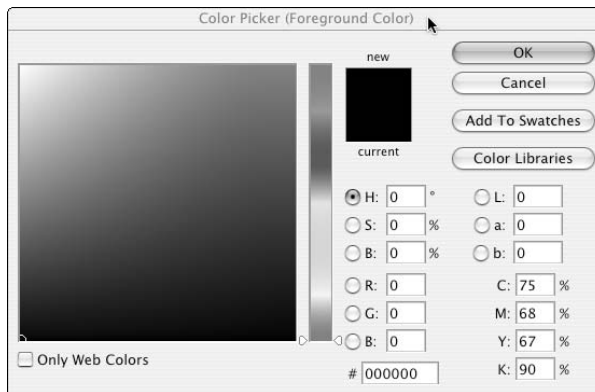


Figure 4-5:
The Color Picker window displaying full color.



Although in many cases, you can pick colors without worrying too much about strict adherence to the Web 216 (Web-safe, 216-color) palette, sometimes you need to. One of those times is when you're making a graphic that needs to sit next to an expanse of code-generated color — for instance, in a banner. If you want to match the background color of the banner and the color in a graphic sitting over the background color, you should use a Web-safe color. That way, you can be sure there isn't a slight color shift between the graphic and the background. *Color shift* is what happens when two colors that should be the same are very slightly off. A color shift between a graphic and the background can be slight to severe but is always distracting. See the example in Figure 4-6 to see how this can produce undesirable results.

Bandwidth

As you start to become familiar with graphics for the Web and Web site design, you need to learn about bandwidth. *Bandwidth* is the amount of information that has to be sent to the user's machine in order for her to see your Web pages.

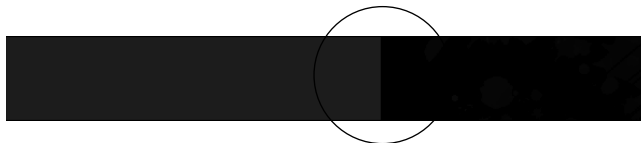
Each Web page is made up of several files working together, including an HTML file, graphics files, CSS files, and possibly some script files or other data that is being transferred from a database. Each element that has to be sent to the user's machine takes up bandwidth. Keep that in mind as you're designing pages. Slow Web pages are a very serious problem for a Web site.

As you work on your site, you might find that you need to increase the download speed of your pages. If you find yourself in that situation, you should look at the following:

- ✔ **Graphics:** Have they been compressed well (the difference between a TIFF and a JPEG file), or could they be worked with?
- ✔ **HTML:** Is the code clean and well written, or does it have a lot of issues like nested tables and unnecessary globs of old HTML?
- ✔ **Back-end systems:** Have your databases and other support systems all been optimized the best way, or has there been sloppy design and coding in the building and maintenance of your back end?
- ✔ **Multimedia:** Have the multimedia elements been designed and developed so that they download gracefully? Have the elements been properly optimized?

Information about how to work with these elements to deliver the best quality while not ruining the download speed appears in Books IV through VI.

Figure 4-6:
A color shift is distracting.



Color shift between graphic and background

Another fix for the color shift (or *dithering*) is to create transparent images that float over the background color. Book IV has instructions and examples of how to create transparent images and other graphics tricks.

Getting Familiar with Graphics File Formats

Two main graphics file formats are used on the Web. You can indeed find many other formats in use on the Web, but they're not as popular as the big two: JPG (or JPEG) and GIF. A third (and much-less popular) format that you might encounter is PNG.

JPEG and GIF files are popular because they work well for most graphics that a Web site needs, and they're viewable by any browser that has graphics-display capabilities. Users don't need to make any special modifications or change any settings.

The JPEG and GIF formats *compress* images, meaning they make image file sizes smaller than they originally were. *File size* refers to the amount of space an image takes up on the computer, not how large it is on your screen. Changing the display size doesn't necessarily change the file size unless you're actually resizing the image in a graphics program.

File compression is important for Web graphics because the larger the file sizes are, the longer it takes for them to download. Compressing files helps them to download more quickly. On the Web, speed is everything.

Although JPEG and GIF both compress graphics, they work in different ways. Each has its purpose:

- ◆ **JPEG (Joint Photographic Experts Group):** This format is named for the organization that created it. JPEG is good at compressing pictures — namely, images with a lot of tonal changes in them, such as photographs, or graphics with a lot of gradients in them. JPEGs aren't good at compressing graphics that have large areas of solid color. Graphics with large areas of solid colors in JPEG format tend to get strange smudges (called *artifacts*) on the image and can also make text hard to read. Figure 4-7 illustrates JPEG artifacts. The only way to eliminate JPEG artifacts is to start over and recompress the original image with higher-quality settings. The trade-off is that the file size will be larger. The whole trick to graphics compression is to balance quality with size; you want the perfect balance of the two. It's different for each graphic, but you will quickly learn how to compress graphics well.

JPEG artifacts make text look messy.

Figure 4-7:
JPEG artifacts make words hard to read.



You can adjust the JPEG's quality and download properties to get the best quality with the smallest file size. You can set JPEGs to *progressively download*, which means that the image begins to display while it's downloading. If you've seen an image that appears to be materializing in waves, it's probably a progressively downloading JPEG. Using this

method, you can make larger images start to appear quickly, which helps your visitor know that something is coming. Otherwise, they might be looking at a blank space for a few seconds.

- ◆ **GIF (Graphics Interchange Format):** GIFs are great for buttons and other graphics with large areas of solid color and sharp lines — such as graphics with text. Where JPEGs get artifacts and leave text looking pretty bad at times, GIFs can deliver clean graphics at much smaller file sizes. GIFs don't get artifacts, so the text stays clean, even at small file sizes. If you're wondering why everyone doesn't use GIFs for all their graphics, GIFs also have limitations. They don't compress photos or other tonal images well. A photograph that's compressed as a GIF often gets *dithering*. This is when the image has too many colors for the GIF format to deal with, so it starts shifting some of the colors to colors that are close but not the same. Figure 4-8 shows an example of an image that doesn't work well as a GIF. The figure shows a gradient as it should appear and the same gradient showing dithering. Dithering can be mild, or it can be severe to the point that gradients end up looking like bands of spray paint.

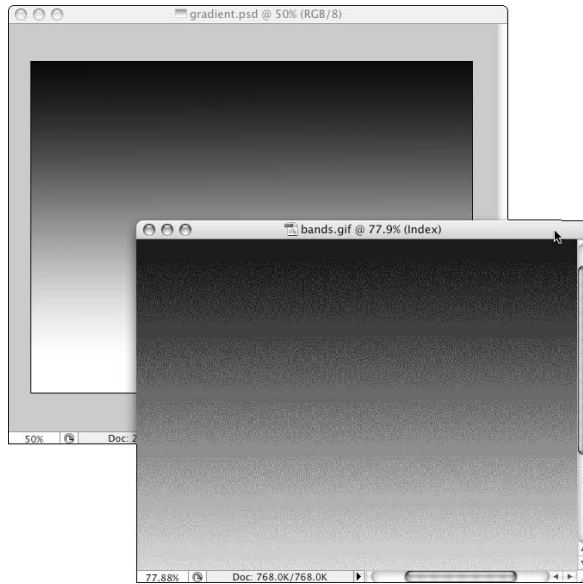


Figure 4-8: Gradients and photos can sometimes degrade when saved as GIF files.

Figure 4-9 is an example of an image that works well as a GIF.

Figure 4-9:
GIF images
do great
with solid
colors and
sharp lines.

Text in my banner

Look to Book IV, Chapter 1 for more information about these file formats.

Using Fonts Effectively on Web Sites

Web sites are different from other media in that you have to rely on the users' fonts to display the site content. Web sites draw from the fonts available on the Web site visitor's machine. When you choose fonts for your Web pages, you need to make sure you choose common ones that most users have on their computers. Otherwise, the computer has to make a substitution, and the results of that can be very unexpected and not at all pleasant. But you must plan for this scenario.

The *font-family method* of choosing fonts for your pages is a tried-and-true practice. With this method, you instruct the browser to use one of a few different fonts. For example, if you specify that the font should be *Verdana*, *Arial*, *Helvetica*, *sans-serif*, the visitor's machine tries to use *Verdana* first. If that's not available, it looks for *Arial*, and so on. If none of the three named fonts are available, the visitor's computer uses its own, default, *sans serif* font. *Serif fonts* have little feet or tails attached to the character; *sans serif fonts* don't. See Figure 4-10. (Book IV, Chapter 1 has more information.) The result of using this font-family method is that you don't have exact control over the font for every single user, but you can establish a look and feel with fonts that are similar. That way, your users can have roughly the same experience. It's a safe bet that no one will be able to tell that your site has *Verdana* on one machine and *Arial* on another — unless the person looking is a typography fan.

Figure 4-10:
Serif fonts
versus sans
serif fonts.

A Serif Font

A Sans Serif Font

Some elements, such as banners and logos, require that you absolutely use a special font. In those cases, you must make a graphic that has the text in it and place that on your page. Display text within graphics sparingly for these reasons:

- ◆ **Text in graphics adds a lot to the information to download.** Make sure you deliver text as graphics only when it's absolutely necessary — like when it is part of your logo or corporate look and feel. Do not deliver body copy as a graphic for the purpose of maintaining a particular layout or font choice. Revise your layout so it works on the Web instead. It will make the site work better for your client and the site's visitors.
- ◆ **Text within graphics requires a very large `alt` attribute for the image (also called *alt tags* or *alt text*) for screen readers and other devices.** If people are trying to access your site via screen readers, they won't be able to read your content unless you include the large `<alt>` tag. (See Book III, Chapter 9 for more information about alt attributes.) Problems can arise if the graphic is inline with other content. The screen reader will read things in the order that they appear in the code. The content can end up being read in a different sequential order than you would expect, which is confusing for the visitor.

In general, though, the aesthetics of using a special font are overshadowed by the poor user experience that is created by placing all your text in a graphic.



Like with many other aspects of Web design, always design for the medium you're using. Web sites aren't the same as printed materials. Printed materials allow for a lot more control over body copy than Web pages do. Make sure you don't make the mistake of choosing the pretty font over your visitor's ability to use the site.

To get the best results from the fonts you choose, just remember a few rules:

- ◆ **Web is not print.** Web pages are *not* printed brochures minus the trip to the printer. Choose fonts that are available on the visitor's machine. Don't try to use a special font that you bought. Users won't have it, and their systems will substitute the font they don't have with one they do.
- ◆ **Be a designer but not a control freak.** You have a lot of control over how your pages look, but ultimately, the user can make many adjustments and settings that can change the way his computer displays your design. Fonts can be substituted, sizes can be changed, and CSS can be overridden to help the visitor. It's a fact of the Web. Embrace it; don't try to force it to act like print.
- ◆ **Make your site user friendly.** Don't embed your text in graphics unless it's a logo, a masthead, or a special promotional graphic that really

needs a special font. When you do embed text, make sure you include an alt text that has the same text in it. Otherwise, some of your visitors won't know what the text in your graphic says.

- ◆ **Use modern standards to build your site.** Use CSS to make modifying and updating your site easier and the user experience better for your visitors. Make sure your code is compliant, well written, and clean. Not everyone using the Web is using a traditional browser.

Implementing Stock Images and Other Elements

Sometimes, you need to use photos, audio files, or other materials that were produced by someone else. When you do so, make sure that you respect the copyrights of others. Stealing images or other materials from a Web site is never okay and can cause large problems if the rightful owner finds out you have been stealing. Everything is subject to copyright laws: multimedia, audio and music, graphics, photos, text, fonts — everything. It's never okay to use things you have found on the Web without permission of the rightful owner.

Many companies sell stock photography, video footage, and music or audio files. Make sure you understand what you're purchasing before you buy.



Different types of licenses are applied to materials available on the Web. Keep these points in mind as you shop around:

- ◆ **Buy only from the person who actually has the right to sell the material.** When you're using materials produced by someone else, you must buy them from the person who actually has the right to sell them. We recommend buying from reputable companies, which might cost you more, but the quality of the materials and the knowledge that you're getting legal materials is well worth it.
- ◆ **Make sure that the license for the materials you buy specifically states that you can use them for the purpose you're buying them for.** Sounds strange, but sometimes, for example, a license clears you to use an image in printed materials but not on the Web — or the other way around. Read the license carefully, and if you have questions, consult your lawyer. Make sure that if a time frame is involved (you can use a song on your site for one year, for instance), you adhere to the restriction.
- ◆ **If you intend to modify the materials, make sure the license allows modification.** Some licenses state that you can't create derivative works, so making a different item from the original is not okay.

- ◆ **When buying on behalf of a client, make sure that it's okay for you to do so.** If it isn't, your client will have to purchase the images. Some clients prefer to do it that way, anyway.



While shopping for stock imagery, you're likely to run into stock photo jargon. The following list highlights just some of the terms you might see. Make sure you always read the licensing terms before using any stock materials.

- ◆ **Royalty-free:** These materials will have specific rules that you will have to abide by while using them. When you buy access to royalty-free materials, you're buying the right to use the materials for the specific purpose stated in the license. You don't own the copyright to the materials. The copyright remains with the *copyright holder*: the originator or, in some cases, a company that has purchased the rights.
- ◆ **Terms of use:** These are the rules that come along with royalty-free materials. You must read the terms of use carefully and make sure that the materials you're buying access to can be used for your intended purpose. **Remember:** You don't actually own the materials, just the right to use them as specified in the terms of use.
- ◆ **Educational use:** Some materials might be marked as "for educational use" or something similar. Again, read the documentation that accompanies the materials because the copyright holders have the right to create their own terms. In general, the term *educational use* — as applied to any materials, including software — means just that: The materials are to be used by educational institutions, students, and teachers for projects relating to the institution or to educational pursuits. Copyright owners restrict what types of projects are allowed. In general, educational use licenses are less expensive than commercial licenses. Because of this, commercial use is probably prohibited.
- ◆ **Rights managed or managed rights:** These types of materials are licensed on a use-by-use basis. The fees for these images vary based on what the image will be used for, how long the image will be used (is it for a short promotion or a longer one?), where it will be used (just in one state or all over the world?), what medium it will be used in (print, Web, video — all three?), and other factors.
- ◆ **Comp or preview images:** These are versions of the materials that you can use temporarily in a project while you are in the design/approval phase of the project. They are usually provided at a lower quality and with a *watermark* (overlaid text or logo that makes them unusable). They serve the important function of making sure everyone approves of the materials before any money is spent obtaining licensing to use them. It's not all right to use preview or comp images in final projects. Figure 4-11 shows a sample of a watermark. If a client provides you with images that have similar marks, ask them where the licensed copies are. Licensed copies don't have a watermark.

Consult your lawyer

Before you start creating Web sites professionally, we recommend consulting with a lawyer concerning copyright and intellectual property laws. A lawyer can help you to protect your rights and help you understand how not to infringe on someone else's rights.

One common misconception with Web graphics is that they are all free for the taking. Although it might be very easy to copy graphics from a Web site, you can't legally use them in your work without permission from the person who holds the copyright. This and many other

confusing and seemingly contradictory rules can get you in a lot of trouble. If you choose a lawyer that specializes in copyright and intellectual property, ask her to help you understand how to stay on the right side of the law.

Another thing a lawyer can help you with is to protect you in the event that a client supplies you with graphics that are questionable in their origins. Make sure you have a clause that allows you the right to refuse graphics that can get you in trouble so that you will be protected if you need to be.



Figure 4-11:
A watermark
can indicate
licensing.

Facelifting an Existing Site

Knowing a little bit about fonts, colors, and images — all of which we discuss in the previous sections of this chapter — can help you effectively apply these elements to a project. A graphical makeover on a small- to medium-sized site can be a quick and easy project if your site is constructed well. If you haven't already done some planning for your project, see Book I, Chapter 1, which contains information that can help you get organized before you roll up your sleeves and get into the work of making changes to a site.

Sometimes a site has good content but just needs a quick freshening up. If you design your sites using HTML and CSS, you can easily modify the look and feel. A *facelift* differs from a *redesign* in that a facelift requires only that you change colors and graphics. Layouts, site structure, and content all stay the same. When you need to redesign a site, you must examine the content, code, layout, structure, and all components to determine what to change or re-integrate into the new site. Comparatively, a *redesign* is a major project. A facelift — which we discuss throughout the rest of this chapter — is just a little sprucing up.

Choosing a new color scheme

Make sure you choose colors that work with any established printed materials that your client is using and that convey the right message. For example, if your site is for a toy store, choose bright, happy colors. If your site is about a more serious subject, choose a conservative palette. As you are choosing your colors, remember you will need a main color, a second complementary color, and a third accent color. You can even choose two complementary colors, but don't try to use too many or your design will become confusing to users. Choose your accent color carefully and use it sparingly, as shown in Figure 4-12. You should use the accent color in small amounts to bring attention to an area of the page. For example, you could use a bright accent color in a thin line at the bottom edge of a banner to bring attention to the main navigation area. Make sure you don't overdo it, though. A little is great; a lot will overpower the design.

Figure 4-12: Sample of a banner graphic/main navigation bar with an accent color applied to bring attention to the area.



Figuring out which graphics to replace

Make a list of all the graphics your site is currently using, taking note of the filenames, functions, and pixel dimensions. Look for banners, promotional graphics, and navigational buttons. Make sure you haven't left anything out. Also, make a list of photos or other elements you will need to revamp. The other elements will range from customized bullet points to multimedia elements.

Decide how you will handle multimedia elements. If their look and feel is very different from the Web site's new look, you might want to revamp them to match. During this process, take the time to clean up old or unused graphics. Figure 4-13 shows a Web site that is considered for redesign; notice the areas that need new or updated graphics. If you find files that aren't being used, don't add them to your list. That way, when it comes time to make the new graphics, you don't waste time making new versions of these obsolete graphics.

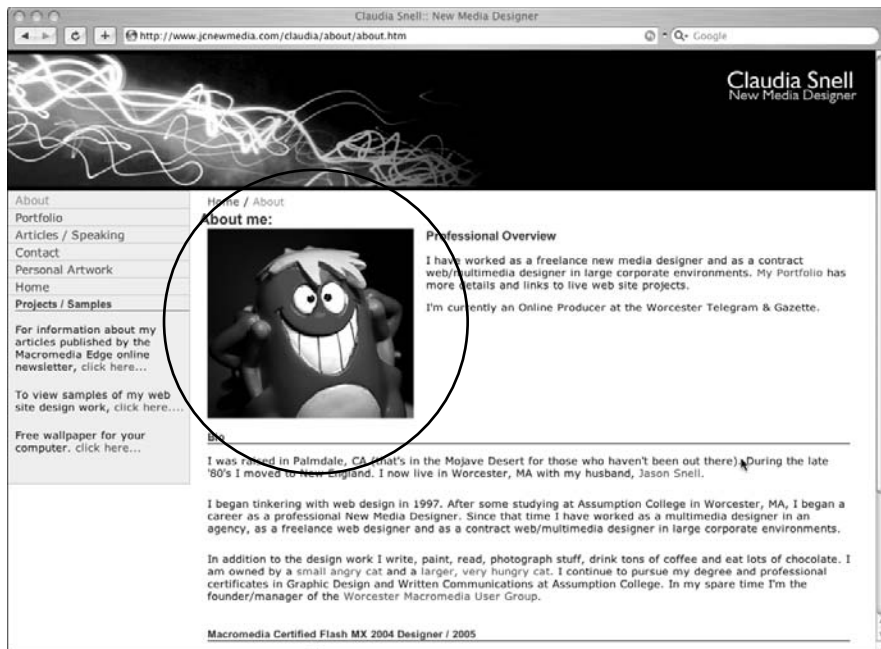


Figure 4-13: Note the graphics on the page and decide whether to replace them. Make sure you look at all the pages on the site.

Creating a new prototype

Using Photoshop CS4 (or another graphics program of your choice), create a design prototype. Remember to keep the same basic layout as your current design. This is just a facelift, not a redesign. Everything will stay in the same place. Only the graphics and colors will change. Experiment with your new colors. The following steps will get you started.

To create a template to work from

1. **Open your current Web site in a browser.**
2. **Take a screen shot of it.**

On a Windows-based PC:

- a. *Press the Print Screen key.*
- b. *Open Photoshop and choose File⇨New⇨OK.*

Photoshop defaults to the same size as your screen capture. The New document dialog box will indicate Clipboard as the Preset size.

- c. *Place your cursor in the new document window and press Ctrl+V to paste the screen capture into the new document.*

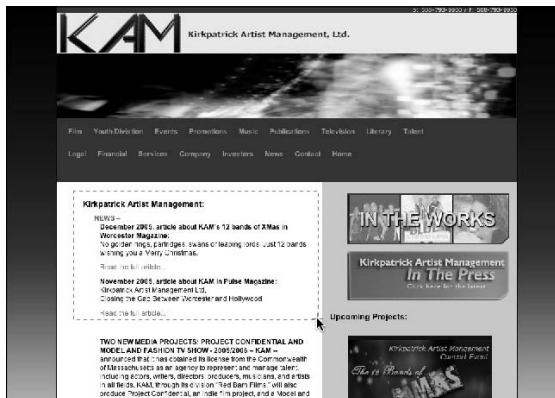
On a Mac, use Grab:

- a. *Choose Capture⇨Window.*
- b. *From the dialog box, select Choose Window.*
- c. *Click the browser window your current Web site is in.*
- d. *Save the file.*
- e. *Quit Grab.*
- f. *In Photoshop, open the screen shot file.*

3. **Copy content areas of your page to use as placeholders with the rectangular Marquee tool. (See Figure 4-14.) Draw a box around a block of text and then press Ctrl+J (Windows) or ⌘+J (Mac) to make a copy of the block on a new layer but in the same place as the original.**

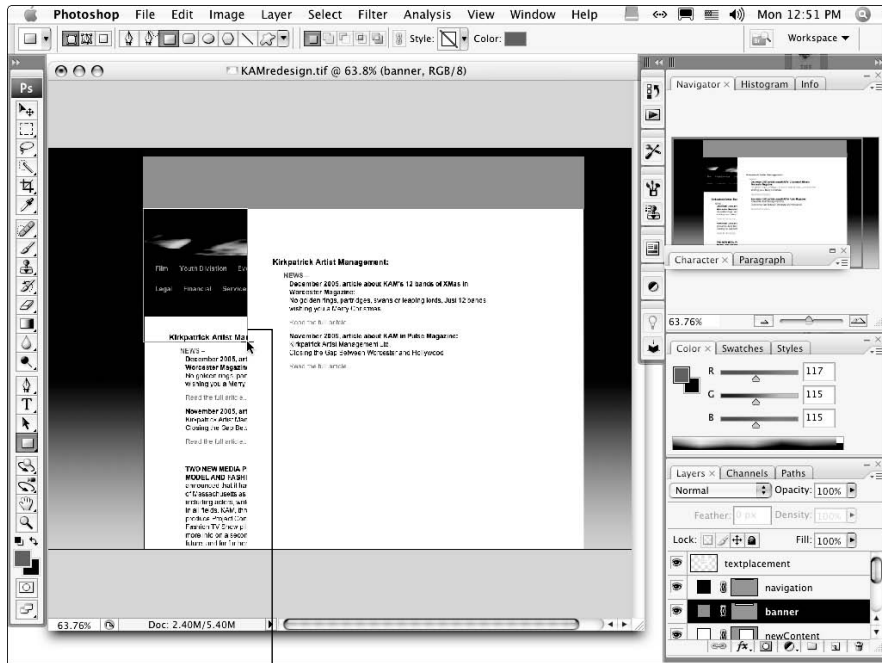
Repeat for all blocks of text.

Figure 4-14:
Use the rectangular Marquee tool to copy content areas of your page.



4. Create page layout elements with the Rectangle tool. (See Figure 4-15.) Draw Rectangles over the banner, navigation areas, and each content block area, making sure you match the size of the elements.

Figure 4-15:
Use the Rectangle tool to draw the elements of your page.



Creating layout elements with the Rectangle tool.

5. Apply the new color scheme to your page elements:

- a. With the Move tool, select the layer that contains the rectangle you're changing.
- b. With the layer selected, double-click the layer — but not on the name or icons.

The Layer Style panel opens; see Figure 4-16.

- c. Select the Color Overlay check box at the lower left of the panel to apply the Color Overlay feature. Click the word “Color Overlay” to open the properties for that feature.
- d. Click the colored rectangle next to the Blend Mode menu.

The Color Picker appears.

- e. Click in the window to select a color; drag the arrows next to the rainbow-colored window to select different colors; see Figure 4-17.

Figure 4-16: Select the layer and launch the layer style.

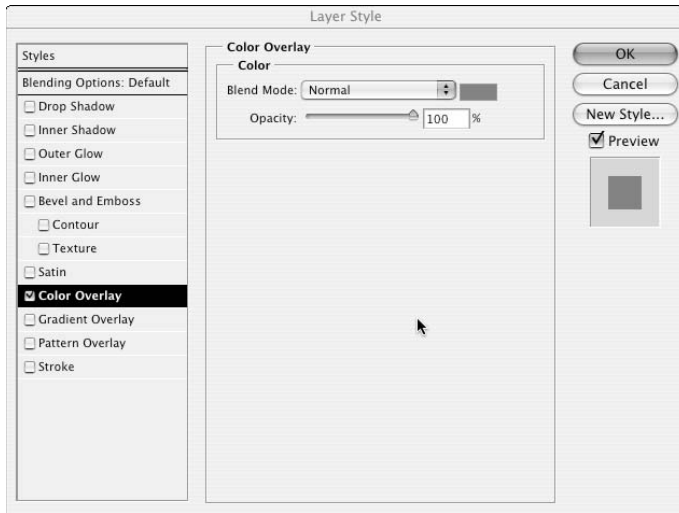
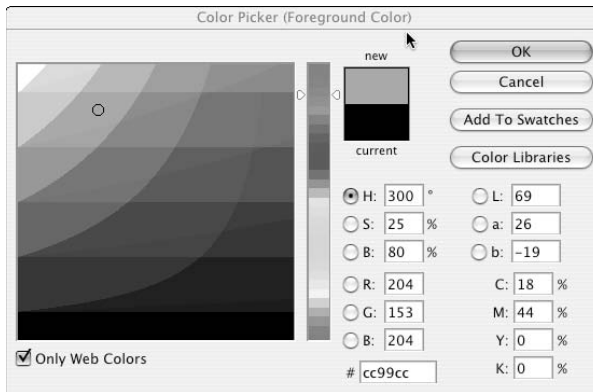


Figure 4-17: Choose a new color here.



Select the Only Web Colors check box if you want only Web-safe colors to be displayed.

f. Click OK.

g. Adjust the Opacity or Blend mode, if desired.

Opacity affects how transparent the color is. 100% is completely opaque; 0% is completely transparent. Blend modes affect how the element reacts to other layers and elements. Experiment with Blend modes to achieve different looks.

h. Click OK when you have finished applying the Color Overlay.

6. Create a new banner, buttons, and other functional graphics.

- a. *In your old Web site, rename your Images folder by appending old to the name. (For example, change the name of the Images folder to ImagesOld, or something similar.)*

By doing this, you have copies of the old images to refer to or revert to if you need them.

- b. *Create a new Images folder with the same name as the original.*

This is where your new graphics go. Book IV, Chapter 3 covers making the graphics you need in a project like this.

While you work, make sure you create graphics that are the same size as your existing ones. If you change sizes, you need to adjust your existing layout accordingly.

Name the new graphics the same names as the old graphics and place them in the new Images folder. If you don't, the graphics won't appear on your pages because the HTML code will be trying to place graphics with the old name and location information. It is important to replace those old name and location graphics with new ones that are named exactly the same thing and are in the same folder where the original graphics were. If you don't, you'll have to manually change the filenames and locations in your HTML files.

Applying the new colors to your CSS file

In previous steps, you decide what color to use for the banner area, and what color to use for the elements on your pages, such as headlines, banner area, and footer — any areas that can be colored through the use of code in the style sheet. You need to sort out what color goes where and apply those changes in the style sheet. To do so, follow these steps:

1. Look in your HTML document and make note of class and ID selector names.

Look for code that looks like this:

```
<div id="banner"><h1>My banner text</h1></div>
```

or

```
<p class="footer">my footer content here</p>
```

2. Open the CSS file and apply the hexadecimal colors to the corresponding classes and IDs.

The corresponding CSS will look like this:

```
#banner h1{color:#000000; font-weight: bold;}
```

or

```
.footer{font-size: small; color: #000000;}
```

#000000 is the hexadecimal color; this code would make the text black. Yours might be a different hexadecimal color.

- 3. Change the hexadecimal color value in the #banner h1 set, and all instances of an h1 within a <div> tag with the ID selector of banner are updated.**

You don't have to open, change, and save each HTML document.

- 4. Follow the same procedure of changing values for all attributes you need to affect in order to create your new look and feel.**

You can change colors, font sizes, weights — anything you like. If you change padding, margins, or positioning, your layouts will change. Make sure you test carefully when you make changes that affect the layout. Also, be aware that if you make drastic changes in font weights or sizes, your layout might be affected.

For more information about CSS and HTML, see Book III, Chapters 2 and 3.

After you make all your changes and test all your pages in a browser to make sure everything looks the way you want it to, it's time to publish your new site.

- 5. Publish (*post*) the new graphics files and the CSS file to your Web server.**

If you didn't add or change the HTML, you don't need to update those files. Please refer to Book III, Chapter 10 for more about File Transfer Protocol (FTP), the method by which Web pages and sites are published to a Web server.

Book III

Site Construction

The 5th Wave

By Rich Tennant



"Okay-looks like the 'Dissolve' transition in my presentation needs adjusting."

Contents at a Glance

Chapter 1: Pulling Together Tools and Materials	125
Matching the Tools to Your Needs	125
Hauling Out the Hardware	126
Setting Up Your Web-Design Studio	132
Desperately Seeking Software	133
Chapter 2: Making a Web Page with HTML	145
Getting Acquainted with the Basic Parts of an HTML Document	145
Starting Your HTML Document Right	147
Helpful Development Practices	159
Chapter 3: Creating a Web Page Layout with HTML and CSS	161
Understanding How CSS Can Help Your Site (And You)	161
Ways of Working with CSS	165
Creating a CSS Document	168
Special Effects with CSS	175
Chapter 4: Introducing Web and Graphics Software	179
Choosing Graphics Software	179
Considering Web-Design Software	183
Deciding on Multimedia Software	185
Touring Photoshop CS4	186
Chapter 5: Getting Started with Dreamweaver	201
Exploring the Dreamweaver Interface	201
Creating a Web Site with Dreamweaver	221
Testing Your Pages and Validating the Code	225
Publishing Your Web Site with Dreamweaver	227
Chapter 6: Using Dreamweaver: Advanced Techniques	229
Creating a Site with Prebuilt Page Designs	230
Creating Templates to Ease Site Maintenance	234
Exploring the Assets Panel and Library Items	238
Adding Functionality with Dreamweaver Behaviors	243
Chapter 7: Creating a Web Page from a Photoshop File	253
Taking a Wire Frame to a Finished Design	253
Finishing the Artwork	259
Slicing a Photoshop Document	264
Optimizing Graphics for the Web	267
Making a Graphic Transparent	270
Chapter 8: Meeting HTML's Powerful Friends	273
Web Technologies Defined	273
Looking on the Server Side	278
Checking Out the Client Side	284
Chapter 9: Web Site Usability and Accessibility	287
Ensuring Your Site Is Easy to Use	287
Helping Users Access Your Site	289
Creating Accessible Web Pages	293
Chapter 10: Publishing the Web Site	299
Look before You Leap: What to Do Before Launch	299
Going Live	304
What Next? The Launch Is Not the End of the Project	310

Chapter 1: Pulling Together Tools and Materials

In This Chapter

- ✓ **Understanding why you need all this stuff**
- ✓ **Choosing hardware**
- ✓ **Carving out a workspace**
- ✓ **Selecting graphics, multimedia, Web-design, and browsing software**

As part of the decision-making process, you need to consider the future along with your current needs. Your hardware and software will grow along with the needs of the sites you're building and with your growing skill set. Buying hardware or software based on current needs and skills alone can result in your gear becoming obsolete more quickly. You can also accidentally limit your personal growth by opting for less-robust software or hardware. For some designers, the issue of outgrowing software and hardware can be a problem; for others, it might not be. If you're looking to build a simple, functional Web site and don't need to build it beyond the basics, you might not need to worry about becoming more ambitious with your Web site's features. As you work on more and more sites, though, your skills will grow along with your creativity. You will want to explore and learn more as you go. Nonprofessional-grade software and hardware — perhaps the perfect match for you when you're starting out — can become very limiting when you get a little experience under your belt. This chapter shows you the hardware, software, and accessories that will grow with you as your skills and demands change.

Matching the Tools to Your Needs

Before you get down to the business of building Web sites, make sure you have the hardware and software and other tools that you need. Start out by assessing your needs so you can make informed decisions about whether you need to buy additional equipment or software.

You have many choices when you're setting up your Web production environment. At the bare minimum, you need the following.

- ◆ **Computer, monitor, and other hardware:** If you're planning on having a hands-on part in designing and building the site, make sure you have a computer to use. This can be your current computer, as long as the computer's specifications meet the needs of the software you decide to use.
- ◆ **Workspace:** At first, you might get by with a workstation set up in the guest bedroom (or your own). Eventually, though, as you take on more projects and get more clients and have more work to do (good for you), you'll need more space (and plenty of it) dedicated to Web design.
- ◆ **Web-design and other software:** You might need to buy additional software with which to build Web sites. Like with the hardware, deciding what to buy is based on your particular needs. If you're in an environment that requires you to use a specific piece of software, make sure you have that software. If you don't have any required programs, decide what software you'll buy, based on what you're going to do yourself and what sorts of elements the sites you're building will have.
- ◆ **Equipment for the whole team:** Commonly, multiple people work on a site. If you have a team working on a site, make sure everyone has the hardware and software needed to do the work. Adding special features to your site, like multimedia or video, also means that you'll have to get equipment and software to support those features.

The rest of this chapter explores these tools in more detail. More importantly, keep reading to help you decide which tools and equipment are right for your projects and situation.



No matter what sort of equipment you decide you need, don't forget that you'll have a period of learning. Even the most basic software, and sometimes hardware, requires a little time to get used to. The more complicated the software and hardware, the more time you need to master it. Professional software ultimately offers more flexibility, albeit with a steeper learning curve. The good news is that it's all "learnable." Just give yourself time, be patient, and keep working. You'll be surprised how quickly you can learn how to build Web sites if you stick to it.

Hauling Out the Hardware

Obviously, you need a computer to build Web sites, and other hardware can make things much easier for you while you work. This section suggests a few things for you to consider but is by no means a complete list.

A computer

You *do* need a computer. As soon as you start talking about building a Web site, you start to get advice from everyone about what type of a computer to buy. You'll hear arguments about PCs versus Macintosh computers.

Some people believe that if you're going to do design work, you must have a Macintosh. Other people say that if you're going to do Web design, you have to use a PC because "everyone" uses PCs. These opinions are just that — opinions. The fact is that you can outfit either a PC or a Mac computer with professional-grade graphic and Web-design software. Both types of computers are just fine to do Web-design work. The choice for you comes down to what you want to use. If you have a computer that can handle your graphics software already, use that. In general, graphics and Web-design software requires between 256MB and 320MB of RAM.



Many professional Web designers actually work on both Macs and PCs. Working *cross-platform* helps the designer to ensure that the site and graphics look good on both types of machines because there *are* differences between the two in how each platform's browsers render Web pages. And PCs tend to display colors a little darker than Macs. As a Web designer, you don't have any control over your visitors' machines and browsers or their personal settings. That's why it's so important that you test your site on a variety of machines before going live: You want to make sure everything works and looks right. This may seem like a lot of work, but it will pay off with a site that does what you expect it to. For a brief discussion of browser simulation programs that are available for testing your site, see the nearby sidebar.

When you look into buying a machine or deciding to use a current machine for production, assess the software you intend to use. Check the specifications that the software needs (RAM, operating system, available hard drive space, processor type, graphics card requirements, and so on). All legitimate software comes with a list of specifications for you to look at. You need to make sure your machine at least meets those requirements. When making the decision, make a list of all the software you intend to use, figure out what each package needs, and then consider your working style. If you like to have many things going on all at once — such as having e-mail, graphics, and Web-design software all open concurrently — consider that and buy a machine that can handle the workload. If you're more focused on completing one thing at a time, you might not have to consider buying a more powerful machine. In general, a slightly better-than-average machine is probably all right.

Computer monitor

Like with all equipment, a better-quality monitor can help you when you work. First rule: *You must have a monitor that displays colors properly.* Although users with a huge variety of computer and monitor configurations can and will visit your site, you should make sure you're working with a monitor that's accurate. If you have an older or less-reliable monitor, consider getting a newer or better one. It is also a good idea to make sure you calibrate (adjust) your monitor regularly.

Your monitor and/or system will likely include tools to help you, or you can buy calibration tools to get a more accurate adjustment. One tool you might find useful is one of the Spyder series calibrators from ColorVision,

(<http://spyder.datacolor.com>). These calibrators analyze your monitor's colors and adjust its accuracy. Many professional designers and photographers use Spyder calibration devices to ensure accurate color.

Another thing you might want to consider is setting up two monitors on your computer. (Check first to make sure your computer supports a dual monitor setup or that you can upgrade it to allow for a dual monitor in the future.) A two-monitor setup gives you twice the screen space. Graphic and Web-design software tend to have a lot of palettes and work areas. Dual monitors allow you to spread things out a bit.

The two monitors work as one large one unless you set them up to *mirror* each other — meaning that the displays are identical. Figure 1-1 shows an example of a dual-monitor workspace setup. This example shows Flash CS4 being used with two monitors. Monitor two (on the right) accommodates the stage, Timeline, and palettes, and monitor one displays the ActionScript window, Flash help, and other code-helpers. This setup greatly increases productivity. Note how palettes can be moved out of the way while you work. This setup isn't mandatory, but it does improve efficiency.

Figure 1-1:
A dual-monitor setup can allow you to work more comfortably.



Backups and storage

When you start to work with graphics, you quickly find that you need a way to store all the large files that you generate. This is particularly important if you're going to work on a large site or on multiple sites. The hard drive of your computer starts to clutter up with pictures, Photoshop documents, and old versions of pages. The answer to the clutter is to have plans for handling the files in the short term and for storing the files on a more long-term basis.

If you need extra space for handling files, consider using an external hard drive. Shop around; you can get hundreds of spare gigabytes for a good price. If you're working with video, your storage needs are much more robust.

Browser simulation programs

As you gain experience with designing Web sites, you might hear about browser simulation sites and programs, which offer an alternative way of viewing and testing your site in various browsers — without you actually doing so on those various *real* browsers. We recommend you opt for the real deal instead: Load the pages and use them in all the target browser/platform configurations you intend to support. Make sure you test buttons, try the page with the graphics turned off and with plug-ins turned off, and use the browser's zoom features to see how the layout holds up. (These features are in different locations on different browsers, so you'll have to experiment at first.)

Try to simulate what real-world users will do with their own computers. It's the only way to be sure that all elements maintain their layout *and* function properly. Scripts can perform differently in different browsers and on different machines, so pay special attention to anything that relies on scripts or plug-ins. Browser simulators can give a rough estimate of what a page may look like, but you lose the ability to really kick the tires (your visitors will be kicking them though). **Remember:** The best way to make sure your site will launch successfully is to *test early and test often*.

In the long run, a good way to store Web graphics files is on either a CD (capable of storing 650MB) or a DVD (capable of storing as much as 10GB). Make sure your computer can create CDs or DVDs and that you have any necessary software. Find a good solution before you accumulate a lot of files. And back up files so that you can have enough room on your machine to work on newer projects. Plus, backing up older files enables you to recover them if you need to.



Make a schedule or routine for yourself for making backups. When it's time, make your discs and then label them well. Be clear about what's on the disc and when you made it. That way, when you need to find something, you won't have to dig through lots of discs trying to find a digital needle in a haystack. A good technique is to burn the digital files to discs and put the discs in a folder along with any hard copies of documents pertaining to the project (signed contracts, brochures, and other materials). That way, you can locate all the materials related to one project very quickly if you need to revisit older projects.

Printers and scanners

Having a printer available when you're doing Web-design work is very handy. Part of the process of designing sites involves making presentations and outlining the plans, and can also include creating and signing contracts. If you're creating a Web-design studio, make sure you can copy and print documents.

You need a scanner, too. Often, photos or other documents are available only in physical form. If you expect to scan a relatively small number of images for use on Web pages only, you can use a common flatbed scanner. If you intend to scan large numbers of images and want to use them for print projects, you'll need a more professional-end scanner. Make sure you consider what type of work you will do and match the equipment to the task.

Note: If you do choose to scan client materials, make sure you build the time it takes to scan those images into your proposal with an option for your client to opt out and scan and deliver the images to you in a suitable format. (Specify the format.)

Cameras and camcorders

Whether you're shopping for a still camera or a video camera, make sure you get quality gear that can produce professional results and that allows for future needs. The benefit is that you get equipment that's versatile and expandable while remaining affordable.



When you start looking for equipment, talk to professionals for advice on what equipment they're using as well as what they like or dislike about what they use. Also, check out message boards and consumer feedback sites.

When you're reading message boards, look for comments that include information about what the poster is using the equipment for as well as his experience level. Look for people who are experienced and doing similar work to what you're looking to do. Also, make sure you read from several sources; that way, you can get a broad overview of what people are saying. The only thing to keep in mind is that you ultimately don't know any of the people posting. Reading many comments from different sources can help you get a more accurate picture.



Marketing materials generally promise professional results, but not all equipment actually produces truly professional quality. Many times, the automatic features actually produce undesirable results. The more the camera controls itself, the less you can use your judgment. In situations the camera isn't programmed to handle, automatic modes can become confused and produce poor-quality video. Make sure that whatever you buy for a camera or a camcorder, it allows you to take complete, manual control. This is important because you need to compress pictures and videos before using them on a Web site. Start with the highest-quality source material you can get. That way, you'll have high-quality Web productions. If your fully automatic camera makes some bad decisions for you, you might be left with video or images that aren't good enough to use.

Digital still cameras

If you need to take photos for your clients, use a quality camera that allows you to take professional-quality images. The better your images are to start, the more flexible they are while you work. It's also common for clients to

want to multipurpose photos, so your camera needs to be able to take high-enough quality pictures that you can use them in other projects, such as printed materials. Research well before buying and try to stick with prosumer or professional gear. *Prosumer* (PROfessional + conSUMER = prosumer), in this context, refers to equipment that is considered to be of a higher quality than consumer-grade, lower-end equipment — these products straddle the fence between hobbyist-grade and professional-level. In general, you want to look for a digital SLR (single lens reflex) camera that can take pictures that are at least 5 or 6 megapixels.

Video cameras

Although consumer-grade video cameras have improved, they really aren't made for professional production. The better the quality of your equipment, the better the quality of your video — and it really shows in the final product. A higher-quality camera has features that will help you produce better-quality video under a wider variety of situations. Make sure you get information from actual users and not from marketing materials.

If you're looking at doing video, consider how you're going to get the video into your computer for editing and preparing it for the Web. You can connect the camera itself to the computer, but that's not necessarily the best way. Each time you use the camera for taping or for getting the video into the computer, you're producing wear and tear on the camera. If you plan to do a lot of video, you might want to consider buying a VTR (also called a *deck*) to do the job of getting the footage onto your computer. A *VTR* is a device that can play back the tapes from your camera — a *VCR* is a VHS deck. Many are available; look for one that can handle your needs. Check the format(s) your camera uses to make sure the deck works with it.

Accessories

Consider the cost of accessories when you're deciding your budget for purchasing camera equipment. The extras can add up quickly. In addition to actual cameras and camcorders, you might want to consider these add-ons:

- ◆ **Camera bag:** Use this for storing and transporting the equipment.
- ◆ **Cleaning equipment and supplies:** Use these to keep everything in good, working order.
- ◆ **Tapes:** Use tapes for video cameras unless you purchase a camera that doesn't require tapes. Research your camera before you buy it to make sure you get the right equipment for your needs.
- ◆ **Audio equipment:** Use this for capturing better sound. Your video camera will come with some type of microphone that's probably adequate but not of great quality. You might need to buy a better microphone(s) to improve the quality of your productions.

- ◆ **Lights:** Use these for improving shooting conditions when available light isn't good. A wide range of lights are available for still and video cameras, ranging from on-camera lights and flashes to whole lighting kits. What you need depends on what type of work you are doing.
- ◆ **Tripod:** Make sure you get a tripod that fits your camera (or camcorder). Make sure you have one that fits your intended use and supports the weight of your equipment. Don't forget to calculate battery weight and the weight of any other accessories you plan to use with your camera.
- ◆ **Storage cards and a card reader:** *Card readers* are devices that attach directly to your computer and allow you to download your pictures easily by removing the card from your camera and inserting it into the reader. From there, you can download your pictures as you would if you had your camera connected to the computer. This can protect your camera from being damaged while waiting to download pictures. Generally speaking, you want at least two or three 512MB storage cards so you don't run out of space. Larger cards are available, but keep in mind that if you have a 1GB card and you fill it, you have to unload all those pictures. It can be more efficient to have two smaller cards so you can unload pictures from one while shooting on the other.

Graphics tablet

Some designers find a graphics tablet useful. A *graphics tablet* is a flat device ranging in size from that of an average mouse pad to about 12 x 19 inches. Designers use a penlike device — a *stylus* — to draw, paint, or write on the tablet. The tablet/stylus is pressure sensitive and generally includes a couple of buttons to substitute for the mouse buttons. The pressure sensitivity works with graphics software to produce more real-looking graphic effects. Whereas a mouse has a more-uniform response to your hand, a stylus or tablet senses how hard you press and produces stronger or thicker lines the harder you press. You can also set the tablet to affect opacity. The settings are up to the user, and you can change them to suit your needs.

Actually, a tablet and stylus can completely replace your mouse if you prefer it, but not all designers use tablets. Whether to use one depends on your personal tastes and what you're trying to do.

Setting Up Your Web-Design Studio

If you're working on a corporate Web site for your employer, you already have a workspace. If you plan to create Web sites for clients, you need to create your workspace. Some resources to have in your Web-design workspace include these:

- ◆ **A comfortable desk and chair:** Building Web sites takes a lot of time. Make sure your chair is comfortable for you to sit in for a long time. Sure, you can work with a hard folding chair and a little table, but you'll be more productive if you make a space that is comfortable and has enough room for you to sit and work for prolonged periods.
- ◆ **Bookshelves:** Web and multimedia production involve a lot of details. As you progress, you'll accumulate reference books, inspirational art books, trade magazines, and other books and materials. If you have a bookcase near your desk, it's easier for you to get to those materials while you work.
- ◆ **Enough desk area to accommodate all of your equipment:** Allow a little extra room for future growth. At least have some idea of how you can rearrange your workspace to accommodate more equipment if need be. If you choose to set your computer up with two monitors, you'll need enough space to have the monitors side by side so you can use them both comfortably.
- ◆ **Space to do paperwork:** Building Web sites also requires some paperwork. You'll find it's a good idea to have enough space to work on proposals, design prototypes, and other paper documents that help you stay on track.
- ◆ **Peace and quiet:** Your workspace needs to allow you to focus. You need a place that lets you keep your attention on all the details involved. If you set up in a major traffic area in your house, you might find it hard to be productive, especially when you have to work through a problem. Professional Web design does require an inquisitive nature and a place that supports reading and problem solving. Be sure to create an environment that will help you be successful.

Desperately Seeking Software

Throughout this book, we cover many different types of software, but this section gives you a quick-and-dirty list of what you should consider. Software is referred to as *industry standard* when it's the preferred package for a particular function amongst professionals. If you're concerned about building a resume, use industry-standard software whenever possible. It can give you a skill set that is valuable to the widest number of employers. If you don't intend to pursue a career or to work with other professionals, industry-standard software is less of an issue for you.



When you're looking for Web- and graphic-design software, check out bundled packages. Adobe Creative Suite 4 (CS4) for Web designers includes everything you need — graphics, Web, and multimedia software. (**Note:** Adobe provides Creative Suite bundles for a variety of design professionals. These bundles are not all the same. Make sure you get one created for Web

designers.) Buying a bundle can save you a bundle of money because the price of the bundle ends up being less than the price of buying each piece of software individually. On the other hand, if you choose to use software from different companies, bundled software might not work for you.

Note: The software discussed in this section is available for both PC and Macintosh computers, unless otherwise specified.

Researching software before you buy

Much of the professional-grade software we overview in this section doesn't come cheap. Before you plunk down your hard-earned cash for a piece of software, do a little research on it. You have several avenues to find out about software, including

- ◆ **Trial versions:** Trial versions of software allow you to test-drive a package before you buy it. Often, some features are disabled because the purpose of the software is to let you try it, not to give you free access to it. Trial software generally has a time limit (two weeks or a month), after which the software ceases to function. If you're unsure about whether a piece of software will do what you want it to do, look for a trial version.
- ◆ **People you know:** Ask around to see whether anyone you know is using the software. That way, you can get some firsthand advice about what the software can do for you.
- ◆ **User groups:** Research on the Internet to see whether any Web-design or software-specific user groups are in your area. A couple of hours at a meeting can get you a lot more information than you could find on your own in several hours of surfing the Web.
- ◆ **Developers' sites:** When you read about software, make sure you read Developer or Community sections of the Web sites in addition to the marketing sections of the sites. It seems that all software boasts of being full featured, professional grade, and easy to learn/use, so make sure that the software manufacturers' ideas of what those things mean match with the reality of your needs before you make the leap. Communities and developer sites can give you a much clearer picture of what it's like to work with the software on a daily basis.



Something that's easy to use for a seasoned professional can be a bit more daunting for individuals just starting out. Don't let that stop you; just be aware that the easier the software, the fewer the features, the quicker you outgrow it.

Graphics software

This category includes software for manipulating photos, drawing illustrations, and creating layouts. You can't build Web sites without them.

Adobe Photoshop

Photoshop is the industry standard for creating graphics and manipulating photos. It's an extremely powerful program with a wide range of uses. Designers who work with Web, video, multimedia, and print all use Photoshop, as do photographers. Figure 1-2 shows the Photoshop interface in its default layout. Chapters 4 and 7 of this minibook cover using Photoshop in Web site design.



Figure 1-2:
Adobe
Photoshop
interface.

A “lite” version of Photoshop is available, called Elements 7 (for Windows) or Elements 6 (for the Macintosh). Elements consists of the most popular and most used features of Photoshop but leaves out many of the professional tools. If your needs are just for basic photograph manipulation and simple graphics creation, Elements might work for you. If you plan to work as a professional designer, you’ll want to learn and use the full version of Photoshop.

You can purchase Photoshop alone or as part of one of the Adobe software bundles. The choice of how to purchase depends on your needs.

Adobe Fireworks

Fireworks (shown in Figure 1-3) is another program that is widely used in professional environments to create graphics and manipulate photos. It also has some useful features, such as batch processing of images and good Web-compression tools.

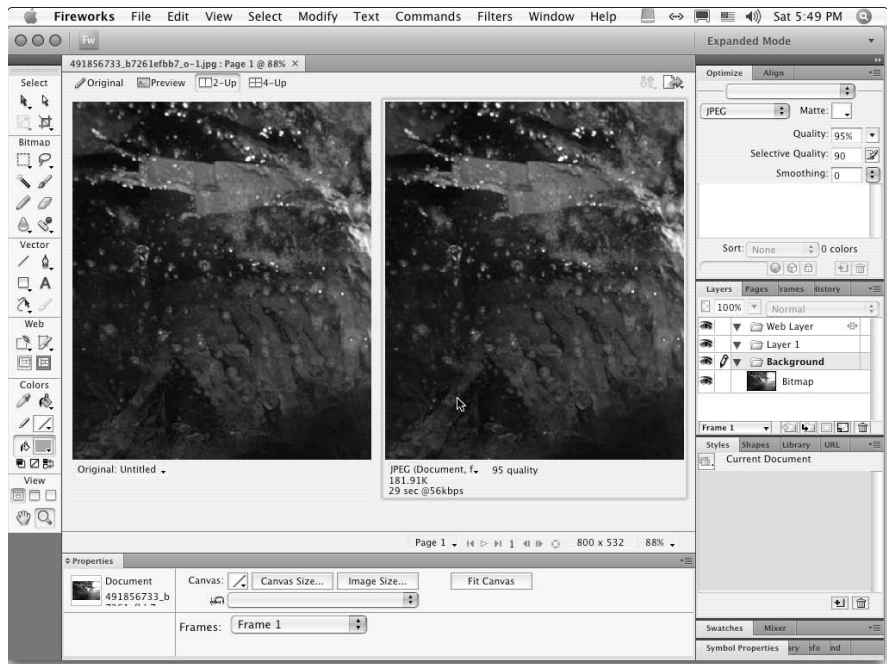


Figure 1-3:
Adobe
Fireworks.

Like with Photoshop, you can create layouts and slice them for use as Web graphics. See Chapter 7 of this minibook for details about how to use Photoshop to make a Web page layout and prepare it to be a Web page. Fireworks also has good tools for making *rollovers* (graphics that change when users place their cursors over them — such as with buttons that appear to light up) and other effects.

Fireworks was created specifically for use as a Web graphics software package, so its tools are specialized for Web designers' use. The focus is on creating great-looking, well-optimized graphics. Fireworks comes alone or bundled with some Adobe CS4 packages and integrates well with Dreamweaver and Flash.

Adobe Illustrator

Illustrator is the industry standard for creating vector images. Vector images tend to look more like line drawings. Illustrator is particularly useful for creating logos. To design a proper logo, you need to create artwork that can scale to any size. Vector artwork does scale without losing quality, so you'll want to create all logos as vector artwork. Figure 1-4 shows an example of the type of art you can create in Illustrator.

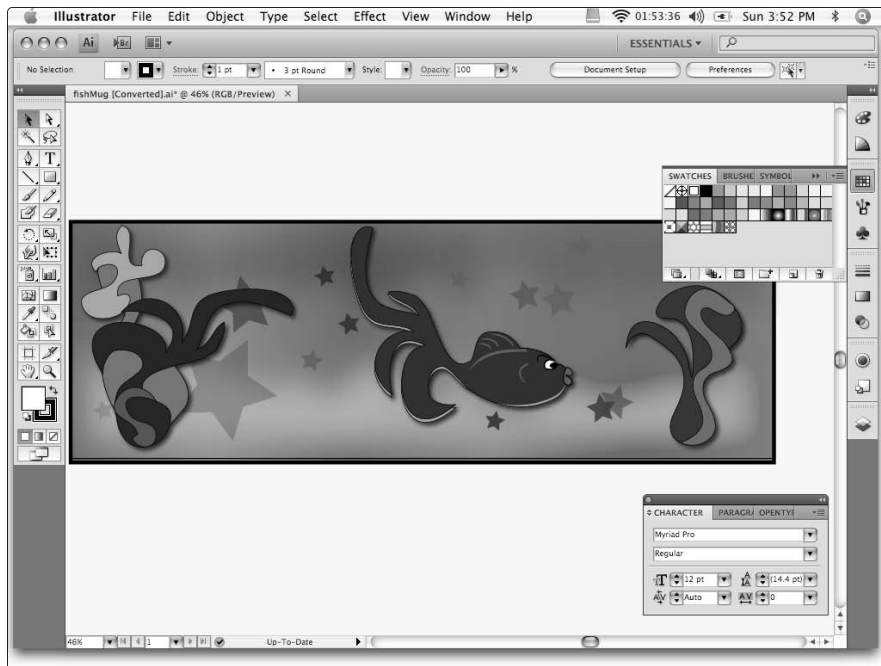


Figure 1-4:
A vector
image
created
in Adobe
Illustrator.

Multimedia software

You use multimedia software to create animations, presentations, and other multimedia elements on your site. If you plan to work with video, you need video editing software. What follows isn't a complete list of all the software available, but it can give you some ideas of where to start.

Combining the best tools from multiple graphics programs

It's common to use more than one piece of software to create graphics and multimedia elements. It's not necessary, but you might find that each piece of software is particularly good at some things but not everything. For instance, Adobe Flash is excellent for creating multimedia. It has a fine toolset for creating graphics, too, but FreeHand and Illustrator are much more powerful in that respect. You

might find that it works better for you to create the graphics in Illustrator and then import them into Flash. If you're using bitmapped images, you need to use something like Photoshop or Fireworks to do the job, and then import the resulting graphics into Flash. See Chapter 4 of this minibook for an explanation of vector and bitmapped art.

Adobe Flash

Flash (shown in Figure 1-5) is the industry standard for creating multimedia interfaces, animations, games, and other multimedia elements for Web pages and CDs. Flash features a design environment where you can create graphics, a Timeline for making animations, and a scripting language (called *ActionScript*) that allows developers to create applications or interfaces for applications. For information about incorporating Flash elements into Web pages, see Book V.



When you use Flash elements on your site, visitors need Flash Player installed on their machine in order to see Flash content. Luckily, most users do have Flash Player. When you create Flash content, include a link to the Adobe site where visitors can get the player (www.adobe.com/downloads). Another option for handling visitors that don't have the player is to include alternate content in the HTML document that instructs people what to do. Flash also has settings within the publishing settings that can put code into your project to check the visitor's machine and give instructions to the visitor if the player isn't the right version. After you get more comfortable with code, you can also create your own detection script.

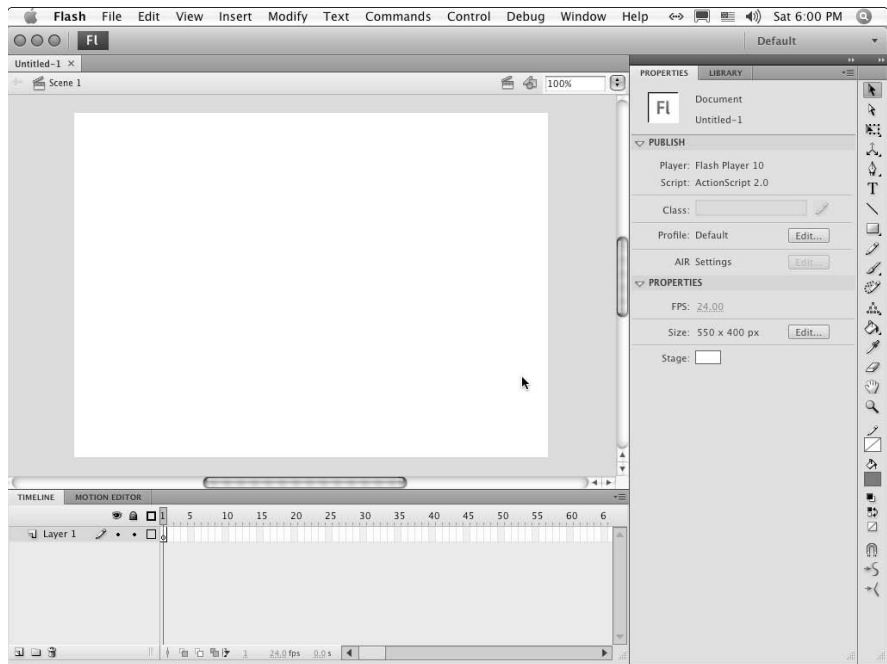


Figure 1-5:
Getting
started
in Adobe
Flash.

For information about or trial versions of Adobe software, visit the Adobe Web site at www.adobe.com.

Toon Boom Studio and Toon Boom Studio Express

Toon Boom Studio is the professional version, and Toon Boom Studio Express 4 is the “lite” version of the software. You can use both to create animations for use in Flash. If you plan on doing a lot of animations — especially if you expect to have a lot of character animation in your work — you might want to check out one of these packages. They specialize in animation and have tools that are intended for that purpose. Trial versions are available on the Web site, so you can try it out before you buy it.

For information about Toon Boom Studio or Toon Boom Studio Express software, visit www.toonboom.com.

Electric Rain Swift 3D

Swift 3D specializes in creating 3D art and animations for Flash. Electric Rain offers two different versions of Swift 3D: a full-blown program and a plug-in for Flash. Which one works best depends on the type of projects you intend to create.

You can find additional information about Electric Rain products at www.era.in.com.

Apple Final Cut Pro and Final Cut Express HD

For Macintosh computers only. Final Cut Pro (shown in Figure 1-6), sold alone or as part of the Final Cut Pro Studio, is a professional, nonlinear, video editing software package. Final Cut Pro has more features than Final Cut Express HD, but depending on your needs, you might find that Final Cut Express HD is enough. Most Web video projects don’t require the more robust features that are available in the Pro version. Before you decide which works best for you, you’ll have to determine what your plans are.

If you do decide to start with Final Cut Express HD and then later move up to Final Cut Pro, you’ll find that it’s an easy transition because the interfaces of the two are so similar. You’ll just have to learn the additional features. Whichever one you choose, you’ll also have to figure out if you want to put motion graphics (*animations*) in your videos. If you do, you need to make sure you have additional software to create the animations — such as Adobe After Effects, Apple Motion, or LiveType.

For more information about the Final Cut products and other software by Apple that can help you with your video projects, go to www.apple.com.

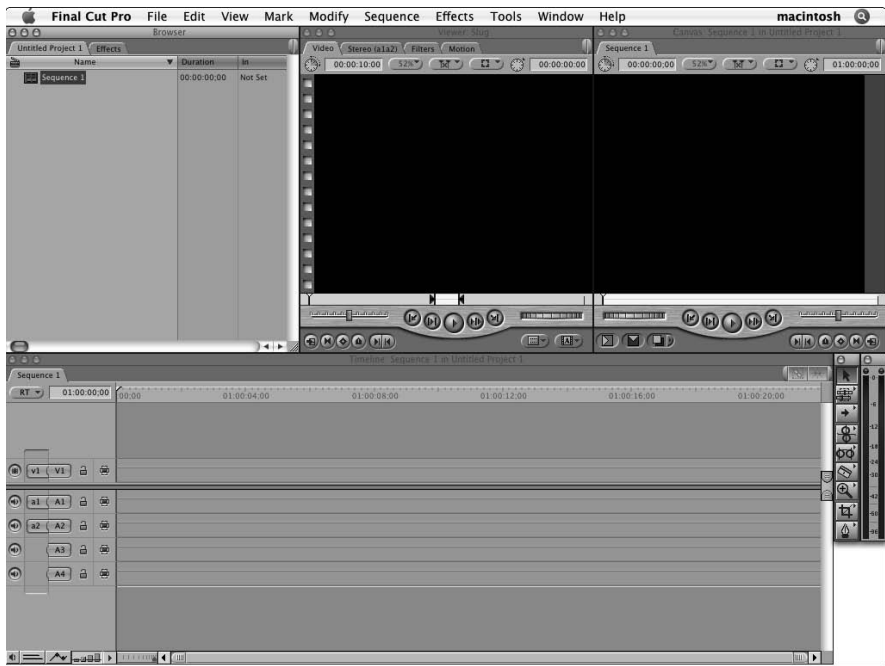


Figure 1-6:
Introducing
Final Cut
Pro.

Adobe Premiere Pro

Available for Windows only. Adobe software's professional, nonlinear, video editing software provides the tools that you need to edit and prepare video. Like the Final Cut products, you need additional software if you want to do fancy, motion graphics in your videos. Premiere is available bundled with other software. Visit www.adobe.com/products for more information about Premiere Pro.

Many other video editing software options are available, but many of them are more robust than you need to create Web video — with price tags to match. If you find that Final Cut Pro, Final Cut Express HD, or Premiere Pro doesn't have the features you need, you might want to look at Avid (www.avid.com) software.

Web-design software

Your Web-design studio needs some type of Web-design software. You can create pages in a plain-text editor or you can buy a WYSIWYG (what you see is what you get) Web page editor. WYSIWYG page editors include both the coding environment and a designer mode that shows you what your page will look like. Your other option is to use software that isn't quite full WYSIWYG but is more robust than a text editor (such as BBEdit). Here are some tips on what type of Web-design software might work for you:

◆ WYSIWYG software

- *Adobe Dreamweaver*: Dreamweaver, considered to be the industry standard for Web design, works with Contribute and other Adobe products — Flash, Fireworks, and Photoshop (via the Edit With command). It comes alone or as part of a package. The Dreamweaver CS4 user interface is shown in Figure 1-7. We cover Dreamweaver in Chapters 5 and 6 of this minibook. Read more about Contribute in Book VIII, Chapter 2.

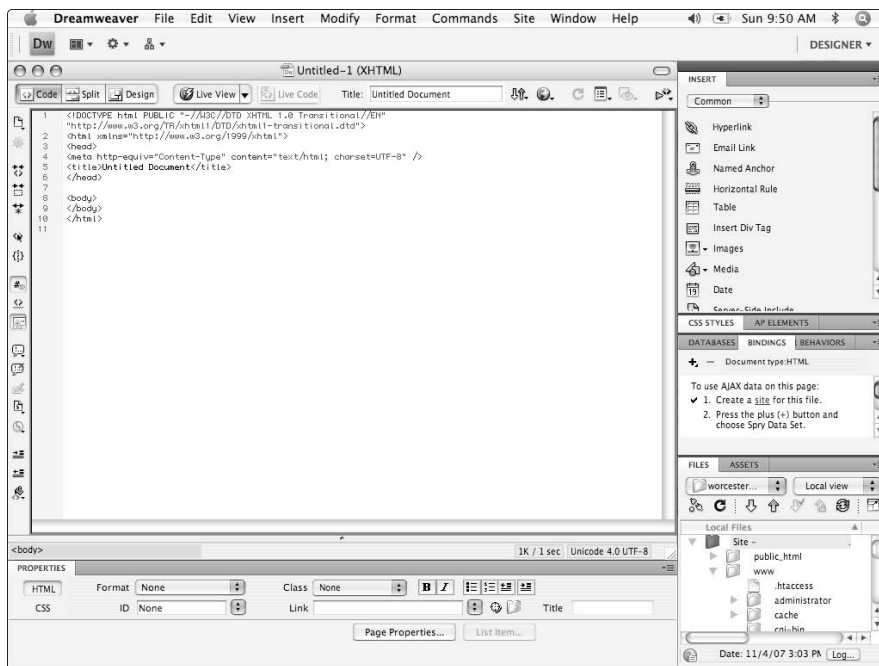


Figure 1-7:
Take a look
at Dream-
weaver CS4.

- *Adobe GoLive*: This Web editing software has many of the same features as Dreamweaver, and many professionals use it. Adobe GoLive is available alone or as part of the Creative Suite Premium bundle. You can find more information about Dreamweaver and GoLive at www.adobe.com.
- *Microsoft FrontPage*: Microsoft's professional Web-design and maintenance software features a full set of tools for creating and maintaining sites for individual users and for larger Web teams.



All three WYSIWYG software packages have features that help keep teams on track. Dreamweaver is the most widely used and integrates with Contribute (which is Adobe's software for nontechnical Web editors). GoLive's interface looks familiar to those Web team members

who already use other Adobe products. Also be on the lookout for older products being used in some settings. For instance, you may encounter some sites being maintained with Microsoft FrontPage, a product that is no longer being made but is still being used by some design/development teams.

- ◆ **Microsoft Visual Web Developer 2008 Express Edition** (www.microsoft.com/express/vwd/): This is a free product that offers powerful Web page authoring capabilities. The tools are not as robust as Dreamweaver but are more than adequate for building and maintaining sites. In addition to HTML, you can also create CSS and JavaScript files using Visual Web Developer Express among other common Web site files (for instance, ASP and PHP files). If you are on a limited budget and don't mind getting into a little code, this product might just do the trick.
- ◆ **Bare Bones Software BBEEdit**: BBEEdit is a widely used HTML and text editor (which means that it's good for all sorts of tasks — CSS and JavaScript included). It's available for Macintosh computers only. BBEEdit is good for individuals who are more comfortable with a code-only environment. Some of its features and functions are similar to those of the more robust WYSIWYG editors, but as the name implies, it really is a pretty bare-bones package. Some developers prefer the stripped-back interface and the reduced toolset for its less-cluttered environment.

Ideally, you'll learn HTML and CSS well enough to be able to create pages in any environment.
- ◆ **HomeSite**: This is Adobe software's HTML/text editor for PC computers only. Like BBEEdit, the tool is geared toward people who are more comfortable with code than with a visual approach to Web site design. HomeSite is a very good package for creating Web page code — HTML, CSS, JavaScript, and others.

The choice of what software you use is really up to you. If you're going to make a career out of creating Web pages, learn HTML and CSS along the way. That will ensure that you're able to handle real-world Web page creation and maintenance. Remember, the more the software does for you, the more you have to trust a machine to figure things out for you. Software is good, but it's still no match for human judgment.

Browser software

You need to have copies of the most popular browsers available on your machine. Ideally, you should have copies available on both platforms. At least make sure that you have access to a way to view your pages on both a Mac and a PC. Some of the most popular Web browsers that you need to have are Firefox, Safari, Internet Explorer, Netscape (depending on your audience), and Opera. You might also need to look at your site in AOL. Researching your sites stats and intended audience data will help you determine which browsers to test in.



Look at your pages in all the most popular browsers because they don't interpret code in exactly the same way. Additionally, pages can look different in PC and Mac versions of a browser. Each browser and platform has quirks that you have to work with. The process of building pages does require a cycle of build, test, adjust, test, and so on until you get acceptable results in the targeted browsers.

The list of browsers and versions that are in common use changes often. Some browsers are no longer in common use or are no longer being supported — such as Internet Explorer for the Mac. You probably don't need to worry about checking your pages in old or outdated browsers. The exception to this is cases where the client or audience will be using older technologies. Make sure you talk with your client about the technological needs of the site before you start building. If you don't, it isn't possible for you to know what to build.

Chapter 2: Making a Web Page with HTML

In This Chapter

- ✓ Getting started on coding your Web page
- ✓ Adding body content
- ✓ Creating bullet lists, numbered lists, and tables
- ✓ Introducing style
- ✓ Developing good coding practices

This chapter helps you understand the basics of HyperText Markup Language (HTML). *HTML* is the underlying code that makes all Web pages work in a browser. In this chapter, we deconstruct an HTML document and show some basics of how to create a Web page.

This chapter focuses on the most basic Web page layout, although many technologies can be added to a site — and the code — to create much more complicated pages. HTML can't use information from a database or to create any multimedia effects on its own. For that, you need other scripting and programming languages (covered in Book VI) and plug-ins, such as Flash or QuickTime (covered in Book V).

Getting Acquainted with the Basic Parts of an HTML Document

The essential parts of an HTML document are pretty simple. Each HTML page consists of a head and a body that are contained in *tags*. In fact, all elements on an HTML page must be contained in tags. The absolute minimum tags needed are the HTML `<head>` and `<body>` tags, as shown here:

```
<html>  
<head>Your header information goes here.</head>  
<body>The content of your page goes here.</body>  
</html>
```

Note that most tags travel in pairs. Each element has an *opening tag* (<html>, <head>, <body>) and a *closing tag* (</body>, </head>, </body>). The opening and closing tags let the browser know where each part of your document begins and ends, respectively. All tags in HTML should be closed; in the current version of HTML 4.01, or XHTML, they *must* be closed. Some tags are *container tags*: That is, one opening tag is followed by some content, which is then followed by the closing tag, as in the preceding example. Other tags, such as a break tag (
), are *empty tags*. Notice how the single tag also has a / to close itself.



If you'd like to have a look at the HTML of any page, view the source code in your browser. It's a good way to see how others are creating their pages. Be aware though, that browsers are very forgiving of bad coding practices. Many older sites and sites created by less-skilled developers might not have standard code. Make sure you verify that a technique is a good one before you put it to use on your own site. To view the source code of a page in Internet Explorer, for example, you can either right-click in your browser window and select View Source, or choose View→Source on your browser's menu bar. (You can also view source code with other browsers. The location of the View Source option might be different, but will be found under a similar view-type menu.) Figure 2-1 shows an example of the HTML code and the page it produces.

Figure 2-1:
A sample
of a Web
page and its
underlying
code.



Picking up pointers from reliable sources

Elsewhere in the chapter, we recommend that you look at what others are doing when you're beginning to write code. We also recommend that you supplement what you're learning with information from the World Wide Web Consortium (W3C). The W3C is the organization that creates the recommendations and standards for properly coded Web sites. Consulting the W3C site (www.w3.org) can

help you ensure that you are picking up the best habits while avoiding bad ones.

As you develop as a Web designer, make regular visits to the W3C site to make sure you stay up to date with the current standards. A very important part of being a competent Web designer is keeping your skills current.

Starting Your HTML Document Right

Throughout this chapter, we work through an HTML document from top to bottom and explain the parts. This will help you develop good, solid code when you start to work on your site. By *good, solid code*, we mean code that is consistent with Web standards, free of typos and other errors, and also well commented. This section explains the document type definition (DTD) and the head and body tags. Each site you create will always have at least these three elements.



As you start to build your HTML document, you might notice when you preview it that it's pretty ugly. The default look of headings and paragraphs can be very unattractive. Don't try to make it look good at this point. The CSS (Cascading Style Sheets) file you create will take care of all the visual aspects of your page. *Cascading Style Sheets* are a type of coding that controls layout. CSS are the instructions for the browser about how to display different elements of your HTML document. CSS can control how much space is between the elements on the page, what colors things are, and how large the text appears. Some very old methods of making HTML look good (such as using the `` tags) shouldn't be used anymore because they create code that's difficult to edit and maintain. The old methods are also not as compliant with modern browsers and devices. A well-coded HTML document is a pretty bland and ugly thing — but that's okay.

The document type definition

The first thing your page needs is the opening DTD and HTML tags. These tags tell the browser what type of HTML document you're sending it. There have been several versions of HTML, and telling the browser which version you're using helps the browser display your page correctly. This example is from a page that is using XHTML 1.0 Transitional, a modified version of HTML 4.01:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//
EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

This code also tells the browser that this page is in English (note the EN) and offers additional information about how to display your page.

The head tag and what goes in it

The head of the HTML document is where *metadata* (information about the document) is contained. Tags associated with metadata are *meta tags*. The purpose of meta tags and the information they contain is to

- ◆ **Inform the browser about what type of document it is receiving**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">
```

- ◆ **Provide information, such as descriptions and keywords, used by search engines**

```
<meta name="Keywords" content="Claudia Snell, Web Designer,
massachusetts" />
<meta name="Description" content="Claudia Snell is a new media designer
based in Worcester, MA." />
```

- ◆ **Provide a title for the page**

This page title is displayed in the title area of the browser and also in most search engine results lists.

```
<title>Claudia Snell:: New Media Designer</title>
```

- ◆ **Link associated files, such as CSS and JavaScript files**

```
<link rel="stylesheet" href="css/
sitestyle.css" type="text/css" />
<script language="JavaScript" type="text/
javascript" src="functions.js"></script>
```

- ◆ **Provide information about when the page was published and by whom**

```
<meta name="copyright" content="2000 - 2005, claudia snell" />
<meta name="publisher" content="claudia snell" />
<meta name="author" content="claudia snell" />
```

HTML versus XHTML

While you're learning about HTML, you're probably going to hear about XHTML (eXtensible HyperText Markup Language), too. XHTML is, in the simplest terms, the next generation of HTML. It's very similar to HTML with some very subtle, but important, differences. These differences are currently in place and will be enhanced in future versions of XHTML to help documents work with XML (eXtensible Markup Language) better. The technical aspects of the differences are beyond the scope of this book, although the World Wide Web Consortium (www.w3.org) has information for the curious.

Older versions of HTML allowed tags to be left unclosed. The current standard is to close all tags and to use XHTML. One of the differences is that XHTML needs all tags to be closed. This is because HTML "sees" the tags as markup; XHTML "sees" them as containers holding distinct types of content within your document. XHTML needs to know where each piece starts and ends. This all becomes important when you want to develop content that works well within many different systems and on different devices. XHTML is designed to be more compatible with XML (a widely used standard for storing and delivering content across different systems and devices). Older versions of HTML weren't designed to work the same way and are thus more forgiving of less-consistent coding practices.

In HTML, you would code a paragraph this way:

```
<p>This is a very short paragraph  
<p>Here is another very short paragraph
```

Closing the tags would be optional.

In XHTML, it would be like this:

```
<p>This is a very short paragraph</p>  
<p>Here is another very short paragraph</p>
```

In XHTML, all tags need to be closed. In HTML, it's a nice thing to do but is not required.

For more information about HTML 4.01 and XHTML, visit the World Wide Web Consortium Web site at www.w3.org. The current HTML specification can be found here:

www.w3.org/TR/1999/REC-html401-19991224/

Note that all the samples of code in the preceding list are closed either with a closing tag or with a slash (/) at the end of the code. A good way to tell whether a tag is a container tag is to determine whether it "contains" anything. The `<head>` tags encapsulate all the header information, so they are container tags. The meta tags merely convey a piece of information — and so are not containers.

Several more meta tags are available, and header information can also contain other types of information, such as CSS styles and JavaScript. To see examples of these in CSS, check out Book III, Chapter 3; Book VI, Chapter 1 covers JavaScript.

So, to have a look at how the code examined so far works together to create the beginning of your HTML document, examine this code:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>
<title>Claudia Snell:: New Media Designer</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
<meta name="Keywords" content="Claudia Snell, web design, worcester,
    massachusetts" />
<meta name="Description" content="Claudia Snell is a new media designer based in
    Worcester, MA." />
<meta name="copyright" content="2000 - 2005, claudia snell" />
<meta name="publisher" content="claudia snell" />
<meta name="author" content="claudia snell" />
<link rel="stylesheet" href="css/sitestyle.css" type="text/css" />
<script language="JavaScript" type="text/javascript" src="functions.js"></script>
</head>
```

This code communicates important information, such as the name of the site, the author and publisher of the content, a description of the site content, and what version of HTML/XHTML the page is using. This information is used by the browser, search engines, and the site's visitors. The code also makes sure that all the correct CSS and JavaScript needed to make the page work is loaded. Figures 2-2 and 2-3 show how the search engine and browser use the head tag information to help your visitors find and use your Web site.

Figure 2-2:
Head tag
information
is used
in search
engine
results.

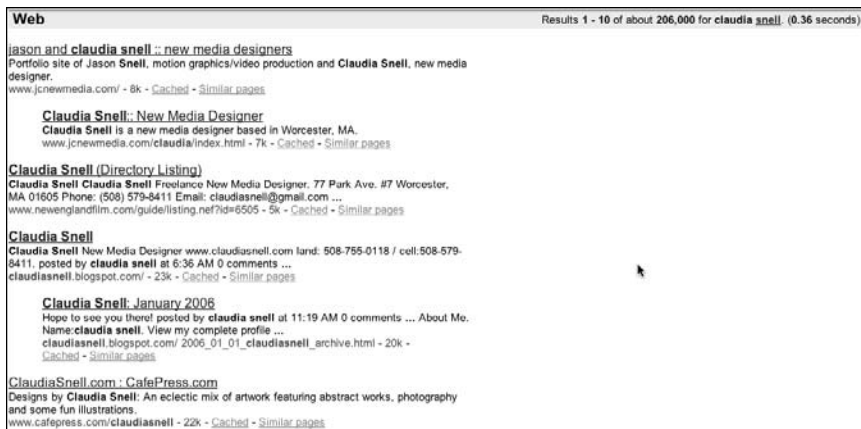




Figure 2-3: Head tag information is also used in the browser.

Now, for some body!

The main part of a page is the *body*. This is where all the visible content of a page goes. All the graphics, images, banners, headings, and paragraphs of text must be contained between the `<body>` and `</body>` tags.

Before you start coding, organize the content as an outline with a main topic heading and then supporting subheadings under each. Although several layers of headings are available, we recommend that you stick with three levels or less. If a subheading has enough supporting information to require `<h4>` and above, it might be a good candidate for being its own page. Remember that people like short, easy-to-access information on a Web page. The cleaner your outline, the cleaner your page — and the better the experience for site visitors.

Adding headings

Headings on your page belong between heading tags. These are container tags that should be used in order of your content structure — `<h1>`, `<h2>`, `<h3>`, and so on. The `<h1>` tag is for the main heading of your page, `<h2>` is for subheadings, and `<h3>` is for subheadings under the `<h2>` subheadings. You can see the progression in the following:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<title>Claudia Snell:: New Media Designer</title>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
```

```
<meta name="Keywords" content="Claudia Snell, web design, worcester,
    massachusetts" />
<meta name="Description" content="Claudia Snell is a new media designer based in
    Worcester, MA." />
<meta name="copyright" content="2000 - 2005, claudia snell" />
<meta name="publisher" content="claudia snell" />
<meta name="author" content="claudia snell" />
<link rel="stylesheet" href="css/sitestyle.css" type="text/css" />
<script language="JavaScript" type="text/javascript" src="functions.js"></script>
</head>
<body>
<h1>Main Heading Of the Page</h1>
<p>An opening paragraph is sometimes nice</p>
<h2>A subheading</h2>
<h3>A subheading of the above h2</h3>
<p>This is a good place for a paragraph</p>
</body>
</html>
```



To see what this code looks like as an actual Web page, copy it directly into a plain-text document. Save the document with an `.html` file extension instead of `.txt`. Open the document in a browser. It won't be very nice looking, but you will be able to see what parts of this code are visible on the page, which parts are in the title bar, and which parts are invisible. You will also be able to see what the different heading types look like by default (before styling). You'll most likely want to apply CSS to make them look better.

Also, notice the closing tags on these container tags. If you don't close the heading or paragraph tags, the browser doesn't know that section has ended. It will treat the rest of the document as if it were part of that section. Imagine an entire page displayed as a main heading!

Coding paragraphs

As you might have noticed in the preceding code example, the container tags for paragraphs are `<p>` and `</p>`, and you use them to separate text into paragraphs. Each paragraph must have its own set of paragraph tags. As you look at the HTML of other sites, you might notice that they don't use `<p>` tags. They use `
` tags instead — sometimes several of them — to create the visual effect of having paragraphs.



This is an incorrect use of the break tag and should never be done. The break tag should be used only when you need a hard break, like in a very long bullet. The reason using multiple `
` tags is incorrect is that when you apply style sheets, you can get inconsistencies in your design because the style sheet will apply attributes to things like paragraphs. The `
` tags will not get the same attributes unless you clutter up your CSS with code to make them the same. Also, coding your site properly will make your content more compatible with other technologies, such as screen readers and hand-held devices.

The anatomy of a whole page

Headings and paragraphs aren't always enough. Sometimes you need lists and tabular data on your pages, too. When deciding how to present your data, consider your other options. If the text fits into short bullets, present it that way. Bulleted lists work very well on the Web where your audience will be looking for quick, easy-to-access information. Tables should be used whenever your information needs that type of structure, like contact lists or price lists. These are scenarios when columns and rows really are best.

Using bulleted and numbered lists

Bulleted lists and numbered lists help your visitors get the point quickly. The code for lists — bulleted or numbered — is fairly simple. You can put the heading for a list in either a paragraph tag or in a heading tag, whichever suits the situation best. Take a look at the following HTML, which creates a simple bullet list:

```
<p>My list of fruit</p>
<ul>
  <li>apples</li>
  <li>bananas</li>
  <li>oranges</li>
</ul>
```

Note that the paragraph tag is closed before the list tags begin. The `` tag in the set means *unordered list*, and the `` tag is for *list item*. To turn this list into a numbered list, use `` (for *ordered list*) in place of the `` tag — and don't forget to close it with the `` tag.

Sometimes a list has nested sub-items in it. The code to make that happen looks like this:

```
<p>My list of fruit</p>
<ul>
  <li>apples
    <ul><li>red</li>
      <li>green</li>
      <li>yellow</li>
    </ul>
  </li>
  <li>bananas</li>
  <li>oranges</li>
</ul>
```

In this example, the sub-items in the `apples` item are part of a second unordered list. Notice how the sub-items list begins and is closed between the `` and `` of the `apples` item. The browser displays them as indented items under `apples` in the main list.

Building tables for your site

Another situation you will likely encounter is a need for tabular data. Tables usually turn up in the form of contact lists on sites but are very common with many types of data, such as price lists or comparison charts. In the old days of Web design, tables were used to control layout of the page, and many sites are still built this way. Sometimes, it's necessary to use some form of layout tables.



If you're using layout tables, avoid nesting tables excessively. *Nesting* refers to the practice of creating a table within a table, within a table, and so on. Building a page with nested tables creates these problems:

- ◆ **The pages load slowly.** If the tables have significant nesting, the browser has a lot more information to interpret and display, and the page loads can be slowed considerably.
- ◆ **The information is less accessible.** Nested tables can make a Web page difficult to view by people who are using devices without traditional browsers — such as PDAs or Web-enabled phones — to visit your site. And they can make your Web page nearly impossible to use by people using screen readers.
- ◆ **Pages are difficult to maintain and update.** Excessively nested tables create a confusing pile of code that is easy to break, hard to repair, and even more difficult to edit, expand, and modify. Edits that should take seconds can end up taking hours.
- ◆ **Pages are difficult to reuse when it's time to redesign.** With rigid table structures in place, it becomes impossible to implement new layouts. Your “new” design will amount to just changing out a few graphics — as long as the new graphics are the same size and shape as the originals.

When you're creating HTML documents, think about how the structure of the code will affect the site's visitors and maintenance of the site. Spending a little time considering the code saves a lot of time when you or your team needs to update or add to your site. Figure 2-4 shows an example of code that has been built with nested tables. Note how the sea of `<td>` and `<tr>` tags (table rows and table data) makes the code difficult to decipher.

Figure 2-5 shows the same information but built with CSS. Using CSS allows a designer to create the layout with much less code. The content of the page is easier to identify, and edits are easier because you need to apply only a few HTML tags as opposed to trying to figure out a complex table and inserting or deleting content from it.

Tables can also be difficult for smaller devices to handle. They don't have as much screen space, so it can be difficult to display content that is in nested tables — especially those that have specific widths included in the code. Try to picture an 800-pixel-wide table trying to be displayed on a handheld device. When you examine the code, it becomes easy to see which page would be easier to maintain or expand.

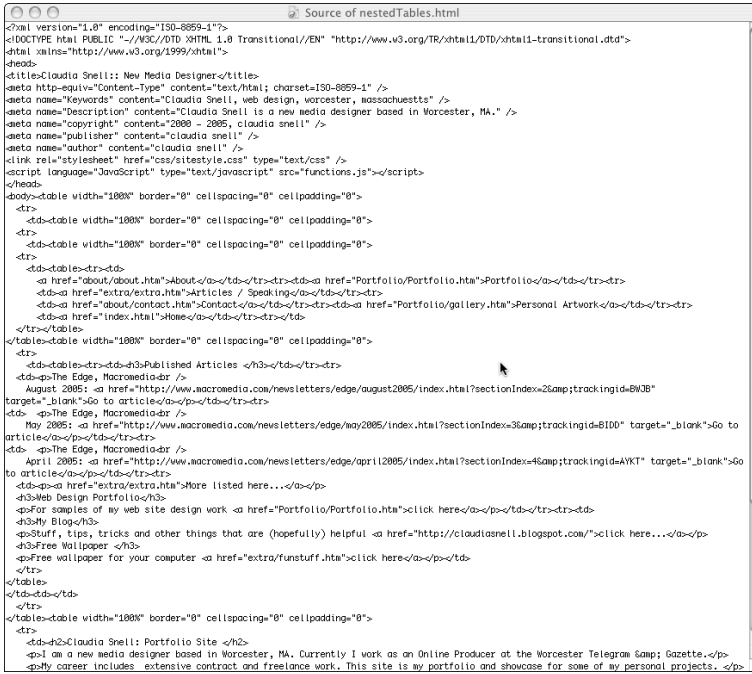


Figure 2-4: Example of nested table code within a page.

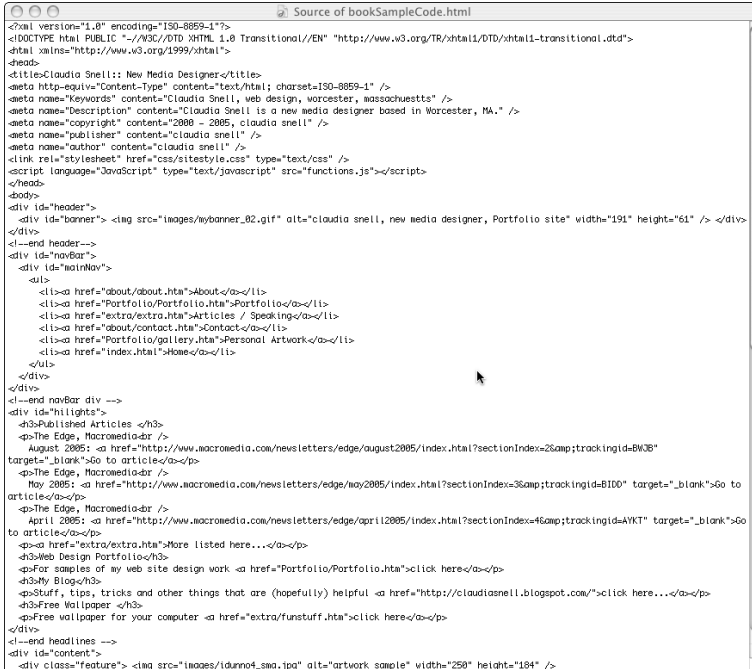


Figure 2-5: Example of the same page but coded with HTML and CSS. Cleaner, easier, friendlier to all.

Book III Chapter 2

Making a Web Page with HTML



A table (properly coded, of course) is sometimes appropriate and necessary. Make sure you keep your table structure as simple as possible. Don't put tables inside the cells of another table unless you must.

The code to create a basic table with three rows and two columns is

```
<table>
  <tr>
    <th colspan="2" scope="col">Table Heading</th>
  </tr>
  <tr>
    <td>Content goes here</td><td>And here</td>
  </tr>
  <tr>
    <td>Content goes here</td><td>And here</td>
  </tr>
</table>
```

The first row of cells has a `<th>` tag. This represents *table heading*. The other tags in this table are `<tr>` for *table row* and `<td>` for *table data*. You might also notice that the `<th>` row has `colspan` and `scope` attributes. Those two pieces of information, respectively, tell the browser that the first row is two columns wide (as in “column span”) and that the table heading is for the columns instead of the rows. If the headings were in the first column, the `scope` attribute would be “row”.

Coding a whole Web page

The following block of code (Listing 2-1) is the code for a whole HTML document that is ready to work with the CSS document in Book III, Chapter 3. This listing shows just the basic structure of an HTML document. The content has been minimized to make it easier to see the code we want to focus on. If you type this code directly into your HTML editor, you will see a very plain document. By adding CSS, the page will become a well-designed and well-coded page. Add this code to the code in the “The head tag and what goes in it” section for the head elements, and you have a whole page.

Note: The `begin header` area in this code (in the second line) refers to the top/banner portion of the visible page — not the head information discussed earlier in this chapter.

Listing 2-1: Basic Structure of an HTML Document

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<title>Claudia Snell:: New Media Designer</title>
<meta http-equiv="Content-Type" content="text/html;
  charset=ISO-8859-1" />
```

```

<meta name="Keywords" content="Claudia Snell, web design,
    worcester, massachusetts" />
<meta name="Description" content="Claudia Snell is a new
    media designer based in Worcester, MA." />
<meta name="copyright" content="2000 - 2005, claudia snell"
    />
<meta name="publisher" content="claudia snell" />
<meta name="author" content="claudia snell" />
<link rel="stylesheet" href="css/sitestyle.css" type="text/
    css" />
<script language="JavaScript" type="text/javascript"
    src="functions.js"></script>
</head>
<body>
<!-- begin header area -->
<div id="header">
    <div id="banner">
        <h1>Welcome to the site</h1>
    </div>
</div>
<!--end header / begin left nav bar-->
<div id="navBar">
    <div id="mainNav">
        <ul>
            <li> <a href="#">About</a> </li>
            <li> <a href="#">Products</a> </li>
            <li> <a href="#">Contact</a> </li>
            <li> <a href="#">Home</a> </li>
        </ul>
    </div>
</div>
<!--add hilights here if you have them -->
<div id="hilights">
    <h3>Come to our open house</h3>
    <p>Come visit us on April 1 for our first <a href="#">open
        house</a></p>
    <h3>Come to our other event</h3>
    <p>Come visit us on April 2 for our <a href="#">other
        event</a></p>
    <h3>Check out our latest project</h3>
    <p>Click here to see our <a href="#">latest project</a></p>
</div>
<!--end left nav bar / begin main content area -->
<div id="content">
    <div class="main">
        <h2>Claudia Snell: Portfolio Site </h2>
        <p>This is my web site. This is a good place for some
            opening text</p>
    </div>
    <div class="subtopic">
        <h2>Another topic</h2>
        <p>Here is a sub topic on my page</p>
    </div>

```

(continued)

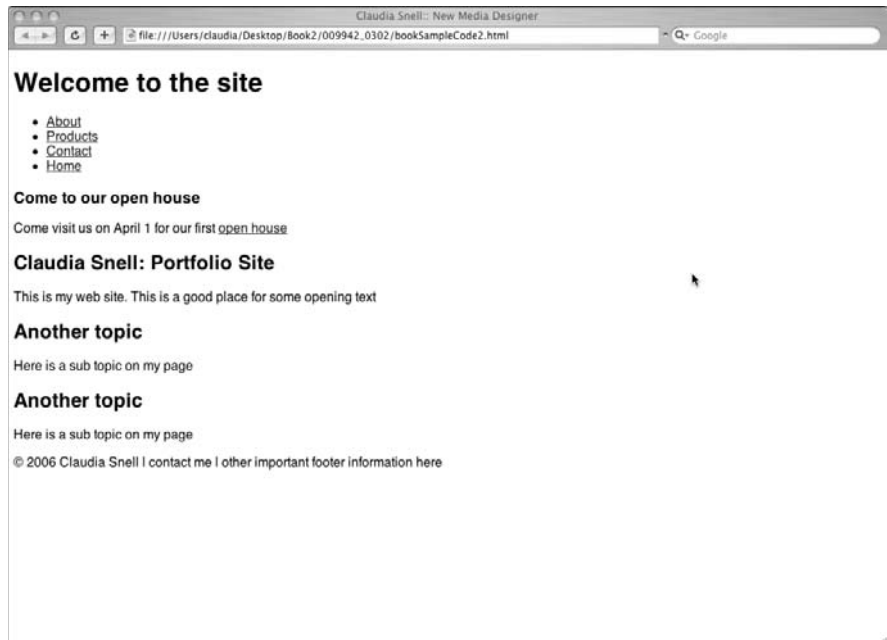
Listing 2-1 (continued)

```
</div>
<div class="subtopic">
  <h2>Another topic</h2>
  <p>Here is a sub topic on my page</p>
</div>
<!--end main content area / begin footer -->
<div id="footer">&copy; 2006 Claudia Snell | contact me |
  other important footer information here </div>
</body>
</html>
```

The page that is generated by the preceding code is not very pretty to look at. (See Figure 2-6.) Still, it's all ready to work with a CSS file to be a well-designed Web page. Note the line of code in the head tag portion that links this HTML document to the CSS file:

```
<link rel="stylesheet" href="css/sitestyle.css" type="text/css" />
```

Figure 2-6:
The HTML page generated by the code sample above. CSS will be used to make this page look good.



Controlling layouts with CSS

The old style of building Web sites used a series of tables and nested tables to create rigid grids that held all the pieces of a page together. Remember that the original purpose of the Web wasn't to support great design but rather to convey information quickly.

The method of using tables to control layout was a misuse of the code to produce a visual effect. It used to be the best solution, but that was before CSS was developed. Another issue that slowed progress in the implementation of better techniques was browser support. In the early days of CSS-driven designs, the browsers didn't support them properly. Pages would sometimes have serious issues. Those times are gone now. CSS has been developed beyond a few font-formatting tricks, and the major browsers now have good support for CSS 1, CSS 2, and even CSS 3. When using more recent versions of CSS, be diligent about checking the site in many browsers and even on multiple devices.



The W3C Web site (www.w3.org) is a good place to go for information about CSS. CSS guru, Eric Meyer, has a Web site (<http://meyerweb.com/eric/css/>) that's also a good reference. If you're curious to see what others are doing with CSS, check out the CSS Zen Garden at www.csszengarden.com. All the designs are created using identical HTML with only the CSS and graphics files changing. The variety of layouts and design possibilities are amazing.

Book III, Chapter 3 shows you how to create a CSS document and make it work with your HTML document.

Helpful Development Practices

Here are some techniques that can help you while you work on your Web site projects:

- ◆ **Use comments in your code.** *Comments* are notes to yourself and other people who might have to work with your files. Whether you're working alone or in a group, commenting your code is an important habit to develop. Common things to include in comments are when the code was added to the document, what the piece of code does, who added it, and so on. Make sure your comments are clear and concise to avoid confusion and frustration. Remember, though, that some comments will be visible via View Source, so don't type things you don't want the general public to read.

You can comment your code whether you're writing HTML, CSS, ActionScript, JavaScript, or any other code. Scripting, markup, and programming languages have their own format for how to create comments. HTML comments are contained between `<!--` and `-->` tags. Everything between the comment tags will be visible only to people who are editing the HTML document and to those who view the source on your pages. CSS comments look like this: `/* comment goes here */`.

- ◆ **Use version control.** Make sure you have something in place to prevent members of the team from overwriting each other's work. Even if you're working solo, accidentally overwriting a file is easy. Make sure you have backups so you can recover the work if something does happen. Explore

your software for version-control options. For instance, Dreamweaver has some tools to help. Power users like tools such as Microsoft Visual SourceSafe 2005, which you can find here:

<http://msdn.microsoft.com/en-us/vs2005/aa718670.aspx>

In general, version-control tools warn or prevent other members of your team from opening and working on files that are already open and being edited. They all work a little differently, and some are more effective than others. That is, they prevent more than one person from opening and working on a file instead of just issuing a warning that someone else is working on the file but letting a second person work on it anyway.

Other version- and source-control features include warning a user when she is trying to post an older version of a file to the server when the file that is currently on the server is newer and presumably has been posted by someone else very recently. Another way to make sure files don't get overwritten is to limit the number of people who have the power to publish files to the site. The designated person acts as a gatekeeper, making sure that members of the team don't overwrite each other's work.

Also keep track of original versions of the photos associated with your site in case you need them later. Keep communication open. There cannot be enough communication between members of a Web team. The nature of the work makes it easy for members to run to their respective corners and hibernate until they're done with their piece. Typically, details of what each person is working on change because planned techniques don't always work when they're put into practice. Communication is especially important when this happens.

For example, if your team decides to use HTML to deliver a company tour and then later discovers that Flash would be a better method, tell all team members about the change of plans. If someone else is developing the content for an HTML page and doesn't realize that you changed direction, the time spent on content development has been wasted. In fact, the team needs extra time to develop the new content.

- ◆ **Organize your supporting files.** While you work on a site, you'll accumulate several files that aren't actual parts of the site. You almost immediately start generating Word documents, Photoshop or Fireworks files, Illustrator files, Flash files, and a host of other files. Make a folder named something like Production Files to keep track of all these files. You should keep this folder with the site folder, but don't post it to your live site. You need to use these files when and if you need to revise your graphics or multimedia elements or if you need to refer back to original content documents. When you launch your site, burn a backup CD or DVD with the site and production files on it. Periodically make new backups.
- ◆ **Make all source documents available to everyone who will need to edit site elements.** Ensure that everyone on the team can edit or create new graphics as needed without having to try to completely re-create the original source files.



Chapter 3: Creating a Web Page Layout with HTML and CSS

In This Chapter

- ✓ **Considering the benefits of using CSS**
- ✓ **Learning some simple CSS**
- ✓ **Using cool style tricks**

In Book III, Chapter 2, you explore a basic HTML document and find some basics about creating an HTML page. This chapter shows you how to create the accompanying Cascading Style Sheet (CSS) file. CSS is a simple language that is a flexible and powerful way to control the layouts of your Web pages. As your site grows, you'll see huge benefits in ease of maintenance and the ability to repurpose or redesign sections of your site very quickly.

Before we jump into the mechanics of creating a Web page layout with HTML and CSS, we first want to show you some reasons why using CSS is a good thing. Next, we tell you how you can use the View Source feature of a Web browser to examine other developers' code. View Source provides a valuable way to see how things are done right (or wrong, as the case may be). We also quickly cover how easy it is to use CSS with other tools, such as Dreamweaver.

Understanding How CSS Can Help Your Site (And You)

We include many code examples in this chapter (as well as in the preceding chapter) that show the old-fashioned HTML way of doing things: that is, compared with the new-fashioned CSS way. In some cases (like with borders and padding), the benefits of using CSS are very clear. For example, you can apply borders to many different types of elements, such as paragraphs and bullets. You can also control the color, width, and style of borders with CSS. You can even specify that the borders appear only on some sides (say, bottom and left, but not on others). Stodgy old HTML allows for borders only on tables, and those borders are either all the way around or none at

all. In other cases, it might not be as clear to you until after you start to work with your site. But know that when you design your site with CSS, you're making things much easier for yourself. When using CSS, you can

- ◆ **Easily make changes to the presentation of your content.** Imagine having to find all the font tags and change them all in an extensive Web site. Using a simple Find and Replace effort often doesn't work because of the inconsistent way tags are coded. It's possible to have hundreds of variations of the font tags — that all look the same on the Web page but are coded differently — which makes using Find and Replace utilities useless.
- ◆ **Easily discern the original meaning of your content.** When working with older, “anything that makes it look good” type techniques, you often can't tell what the different parts of the page content are just by looking at it. Old-style HTML allows for using tags improperly, like applying properties to a font tag that makes a paragraph tag display as though it were a heading. Not easy to fix. If you manage to strip out the old font tags, you also strip out the visual cues that can help you to recode the page properly.

For example, if you set up your page as a block of content with `
` tags to separate it visually into paragraphs and `` tags to make headings look different, it's difficult to figure out what was what if the tags were disrupted or deleted. Another consideration is that users can create their own style sheets. Creating these style sheets is particularly important for some people, especially folks who are visually impaired and develop their custom style sheets with the assumption that your paragraphs will be marked up as paragraphs and that headings will be headings. If your site isn't coded properly, it can make your site much more difficult for these individuals to use because their style sheets won't work as they expect them to.

- ◆ **Make your site friendly to all who visit — no matter where they are or what their situation.** Another reason to use CSS and HTML properly is that of the increasing number of people using the Web, many aren't using a traditional computer. Instead, they're using handhelds, phones, and other mobile devices. In addition, some of your visitors don't have perfect vision and hearing, and many don't have a fancy computer with great speakers. The point is that the Web is more accessible than ever, and you have to be mindful of the wide variety of situations and visitors — and thus code your pages so that your visitors can easily use your site.

Again, some users will have customized style sheets so they can use the Internet. Perhaps a person needs to enlarge font sizes or to specify how a screen reader aurally signals different parts of a document to compensate for visual impairments. Information about aural style sheets can be found on the World Wide Web Consortium (W3C) site at www.w3.org/TR/REC-CSS2/aural.html.



We spend time in this chapter showing how true the preceding bullets are, but first, we want to give you some quick notes about the View Source technique and integrating CSS with other tools.

Checking the source

To get a look at the CSS of a page, use the View Source technique. To view the source code of a page in Internet Explorer (IE), for example, either right-click in your browser window and choose View Source, or choose View⇨Source from your browser menu bar. (You can also view source with other browsers. The location of the View Source option might be different, but it's found under a similar view-type menu.) Look for a line of code in the <head> section of the page that refers to the CSS file *or* look for the actual CSS in the head of the file. If you find a reference to the CSS file, you can use your browser to view the file (more about that in a minute). CSS can also be applied within the document itself, but that method defeats one of the main strengths of CSS: namely, the reusability of code that's kept in only one place.



Save the HTML code so you can play with it on your own. When you view the source, look for signs of CSS. Inline and embedded styles are easy to see. You can also get a look at external style sheets by looking for the path to the file in the head tag. Assuming the URL of the Web site is `http://www.example.com` and the folder you want to keep your CSS in is located at `http://www.example.com/css/`, the URL of the CSS would be `http://www.example.com/css/sitestyle.css`. By typing the URL of the CSS into your browser window, you can view the CSS:

```
<link rel="stylesheet" href="css/sitestyle.css" type="text/css" />
```

Save a copy of the CSS and HTML files so you can play with them and see how they interact. To see a lot of examples of the power of using CSS to control your visual display of a page, visit the CSS Zen Garden at `www.csszengarden.com`. This project, created by Dave Shea, encourages and supports the use of CSS design. There are a series of page designs that are all very different in appearance, but they all share the same HTML code; only the CSS changes from design to design.

For example, at Claudia's site, `http://claudiasnell.com/jcnewmedia/`, choose View⇨Source from the IE menu. A file with the HTML for her site appears. About 16 lines from the top, you see the path to her CSS file:

```
<link href="css/sitemain2.css" rel="stylesheet" type="text/css" />
```

To access the CSS file, type this into your browser's address bar:

```
http://claudiasnell.com/jcnewmedia/css/sitemain2.css
```

The CSS file opens, and you can view it or save a copy for reference.

Hexadecimal colors

HTML and CSS “see” colors in special code called *hexadecimal code*, which is a series of six numbers and/or letters that represent the amounts of red, green, and blue in a color. Red, green, and blue (or RGB) are the primary colors for light; because colors are displayed by using light, RGB are the primary colors of computers, too.

When you look at a Web page’s code, code that looks like #000000 or #993366 refers to hexadecimal values for colors. This six-digit number is broken into three pairs. For example, when a browser reads the code #000000, it reads 00 for the value of red, 00 for the value of green, and 00 for the value of blue (and generates the color black). Or when the browser reads the code #993366, it reads 99 for the value of red, 33 for the value of green, and 66 for the value of blue (and generates a reddish-purple color).

So how do you keep track of which numbers stand for which colors? Don’t worry; many charts are available, such as the one from W3Schools at www.w3schools.com/html/html_colors.asp. And virtually all Web-design or graphics software has a Color Picker tool that gives you hexadecimal values. (See the “Using CSS with other tools” section in this chapter for an example.)

CSS works a little differently. In CSS, you can use shorthand to represent Web-safe hexadecimal colors. Take the first number of each pair and use a three-digit shorthand equivalent of the whole six-digit hexadecimal color. For example, the shorthand for #993366 is #936 (no commas or spaces in the code itself, just #936). A browser assumes that the other three (“missing”) numbers are 9, 3, and 6.

Note: This shorthand works only on hexadecimal colors with pairs for the values of red, blue, and green. Non-Web-safe colors often do not. For instance, the code #efefef represents a non-Web-safe gray. That particular code doesn’t have a valid shorthand because by typing #eee, you’re telling the browser that the color is #eeeeee, which is not the same at all.

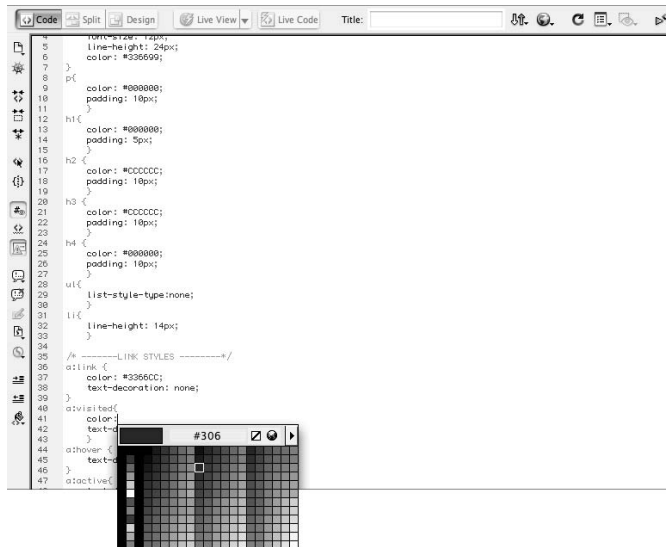
Hexadecimal values work in both HTML and CSS. They also work on all elements that can take a color value.

If you want to know more, check out *HTML 4 For Dummies*, 5th edition, by Ed Tittel and Mary Burmeister (Wiley); it includes a super-size Cheat Sheet with a handy hexadecimal color chart.

Using CSS with other tools

As you begin to work with HTML and CSS, you’ll notice the many tools available to you. Dreamweaver has excellent HTML and CSS support and many tools to help you along the way. Figure 3-1 shows an example of a handy Dreamweaver tool — a Color Picker that pops up automatically when you need it. If you type **color:** into a CSS document in Dreamweaver, the Color Picker pops up to help you. That way, you don’t have to memorize all the possible hexadecimal color values: Simply click the color that you like.

Figure 3-1:
Dreamweaver helps you find the correct hexadecimal value for a color.



Ways of Working with CSS

The three ways that CSS can be implemented on your pages are

- ◆ **Inline:** *Inline CSS* refers to styles that are directly in the content. This method is the least desirable type because it affects only the content that the styles are directly associated with.
- ◆ **Embedded in the head of the HTML document:** These styles offer more flexibility because they can be applied to many elements, such as all the paragraphs, in the document. These styles will not, however, affect parts of other documents.
- ◆ **External:** To have styles that affect all the pages of a site, you need to create an external CSS file and create a link to that file in the head section of your HTML documents. This third way enables you to take advantage of the quick site-wide layout control discussed earlier in this chapter.

Note: These methods can be used alone or in combination with each other. If you do use more than one method, you must know about the Cascade. *The Cascade* refers to the fact that the style that is closest to an element is the one that takes precedence. For example, say you have styles set up for paragraphs in your external CSS file, and you decide that you need a special treatment for paragraphs on one page of your site. You can set up those styles in the header. The CSS in the header will take precedence over the CSS

in the external CSS file because it's closer to the paragraphs of that page. If you then decide that one paragraph in that same document needs a third treatment, you apply inline CSS to just that paragraph. That paragraph — and only that one — will be styled by the inline style.

Inline styles

Inline styles are coded directly into the body of your document. For example, take a look at the following HTML for a paragraph:

```
<p style="color:#000000;">This would be black text</p>
```

Although inline styles are pretty quick to add on the fly, we don't recommend this technique. In essence, you're trading an old style of coding that created hard-to-maintain sites for a new flavor of the same thing. For example, in the code that follows, you can see that this style is applied directly within the `<p>` tag and will affect only this one instance of a paragraph. No other paragraphs will take on the attributes unless you apply the same style attributes to them.

```
<p><font color="#000000">This would be black text</font></p>
```

The problem with coding this way is that if you decide you want to change the color of your text on your site — or anything else coded with inline styles — you have to find *all* the places that you used these inline styling techniques and change them. It also makes for a lot of clutter that isn't necessary.

Styles embedded in the head of the document

When you *embed styles*, you create your CSS styles in the head portion of your HTML document and refer to them in the HTML, like this:

```
<head>
<title>Claudia Snell:: New Media Designer</title>
<style type="text/css">
<!--
p {color: #990099}
-->
</style>
</head><body><p>This text would be purple</p></body>
</html>
```



The advantage to using this technique is that you can have some specialized styles embedded in just one HTML document. This can come in handy if you want a special page for an event or some other reason.

Use this technique to implement styles for that HTML document only. The other pages of your site aren't affected by styles that are embedded this way or created inline. If you're creating styles that you intend to use throughout

your site (which is most often the case), don't use this technique. You'll end up with a site that's a pain in the neck to update because you'll have to open each document and edit each embedded style individually — unless you feel brave and want to use the Find and Replace feature of your software.

External style sheets

Using an external style sheet is generally the preferred method, especially if you want to implement your styles across the whole site. All the CSS style information is created in an external file, or files if you have a more complex site. The files are linked to the HTML document in the head portion of the HTML, like this:

```
<head>
<link rel="stylesheet" href="css/sitestyle.css" type="text/css" />
</head>
```

This code links the CSS file to the HTML document. Notice that the external style sheet's file extension is `.css`. When a visitor goes to your site, the CSS is loaded along with the HTML, and the page looks great. The major advantage to this technique is that if you decide you want to change anything about your design or layout, you can make a site-wide change simply by changing the styles in the CSS file. You don't need to open the HTML files to edit them. Of course, you'll want to preview your pages before publishing them, but that's the rule for all Web pages.

You can also create multiple style sheets and link them to the same HTML document, like this:

```
<head>
<link rel="stylesheet" href="css/sitestyle.css" type="text/css" />
<link rel="stylesheet" href="css/photogallery.css" type="text/css" />
</head>
```

The benefits of this technique are that your site can have a unified look and feel, and you can also implement some specialized styles throughout just one section. For instance, you can implement a color-coding scheme.



The Cascade (again, how the priority of styles work together) is a strange but useful friend. The most basic explanation of how it works is that whichever style declaration is closest to an element is the one that will take effect. However, that's sort of how it works. In some cases, styles interact in unexpected ways. Be on the lookout for multiple styles applied to the same type of element or styles that will affect positioning of elements in relation to each other, such as margins and paddings. If you place a margin on a paragraph and a padding on a table, the two will interact when you put a paragraph in a table. You might need to adjust your styles when things don't look how you expect. Don't get discouraged, though; the best way to master CSS is to get in there and work with it.

Commenting your code

Each type of coding has its own language style or *syntax* (like grammar for computers). You must use proper syntax when creating any code. If you don't, the code won't work, or it might do unexpected things. Even comments have proper markup and/or syntax. If you don't create code comments correctly, the browser might see them as content or code and treat them as such instead of keeping them hidden, the way it should. (See Book III, Chapter 2 for information about using comments in general.) In CSS, comments look like this:

```
/*Banner and header styles - creates background color and  
places images - this is a CSS Comment*/  
#banner{width: 100%; background-color: #000;}  
#banner p{color: #fff;}  
#header{border-bottom: 1px #333 solid;}  
/*End banner and header styles - this is also a CSS Comment*/
```

Note how the comments are between a `/*` and `*/`, which signals to the browser that the information contained is a comment and is to be ignored.

Keep your style sheet clean and easy to understand by:

- ◆ **Grouping styles that work together:** In the previous example, the `#banner` style establishes the width and background color of the banner area, and the `#banner p` style specifies that paragraphs within the banner area will be white. Grouping styles that work together this way in your CSS file is a good thing; that way, seeing all the pieces is easier.
- ◆ **Placing a comment at the beginning of a group of styles stating what the styles are affecting and what they're doing:** Keep it short, but make sure it's informative enough so that others (or yourself in six months) will understand it.
- ◆ **Including a comment at the end of the group.** By doing this, you keep the styles organized and reduce confusion about where one set begins and one ends.

Creating a CSS Document

It's time to start creating your first CSS document. A *CSS document* is the file where the styles you create are kept. The term *CSS* can refer to this file or to the actual styles. That can be confusing, but as you get more comfortable with the way everything works, you'll also get used to the lingo of Web design. For the sake of simplicity, we work with the HTML that we create in Book III, Chapter 2 to create a simple design that helps you get your footing

as you set off into the Web design wilderness. Although the actual styles work as inline, embedded, or external style sheets (which we explain in the preceding section), the examples given here refer to styles that are external.

CSS styles are made up of three parts:

- ◆ **Selector:** Specifies what the style will affect
- ◆ **Property:** Indicates what exactly will be affected (font, color, background, and so on)
- ◆ **Value of the property:** Indicates how the property will be affected (fonts will be black and bold, for instance)

The other thing to notice is the different types of selectors, or ways of attaching styles to parts of your HTML content. Some affect HTML tags (<p>, <table>, <body>) directly. Other styles create classes or IDs. Classes and IDs are a bit more complicated, but we explain them later in the chapter.

Setting default selectors

As you get started, you want to create styles for some common elements to ensure a unified site style. Default paragraph, link, and heading styles are examples. These are all styles that will affect HTML tags directly; they are your most general default values for things like colors and fonts. Those types of styles are important, and we explore those in the next section.

Start your CSS document by listing the default selectors you want to create. This helps you get started and ensures that you won't forget something very basic as you get into developing your CSS file:

body { }	Body
p { }	Paragraph
h1 { }	Heading 1
h2 { }	Heading 2
h3 { }	Heading 3
ul { }	Unordered List
li { }	List Item

These are some, although not all, of the basic elements you want to make sure you create styles for.

At this point, it's easy to begin to write your *base styles*, which are the styles for paragraphs, links, tables, headings, and other common elements that will appear on your site. These serve as the basic default values for each element

although they can be changed by creating custom styles. We cover this in more depth later in this chapter. For now, we focus on the basic parts of a CSS style. The *syntax* (or grammar) for writing CSS is

```
selector {property: value;
  property: value;
  property: value;}
```

A style can have several `property: value` pairs, or just one. A style can also have several selectors. For example, the style that follows has four selectors, all of which happen to be headers:

```
h1, h2, h3, h4 {font-weight: bold; color: #990000;}
```

So, here's an example of a style with one selector and five `property: value` pairs:

```
body{
  font-family: Verdana,Arial,sans-serif;
  color: #000000;
  margin: 0px;
  padding: 0px;
  background-color:#ffffff; }
```

The preceding example does the following:

- ◆ Uses the Verdana as a default font, with Arial the next choice if Verdana is not available, or then a sans serif font if Arial is not available.
- ◆ Specifies that the default color of text is black (#000000).
- ◆ Specifies that the HTML document should entirely fill the browser window with no margin (0px) or *padding* (0px) — spacing — between the edge of the browser window and the content of the page. When creating styles, px refers to pixels. So in this example, the margin and padding are set to 0 pixels.
- ◆ Creates a default background color of white for the whole site (#ffffff).

Setting class and ID selectors

The other types of styles are class and ID selectors. Class selectors can be applied to many elements in your HTML document. Each ID selector can be applied to only one element on a page. There are also differences in how these types of styles are applied.

Class selectors

Class selectors are enhancements to your default styles. You create them if you want to have more than one style of paragraph. For instance, suppose you set your default paragraph to use black, Verdana type with a margin of 5 pixels, but you also want to have a special paragraph type available to use to emphasize a point. You would create the following styles:

```
p {color:#000000; font-family: Verdana, Arial, sans-serif;}
.special {color: #ffffff; background-color: #333333; margin:
    5px;}
```

The first line, as you might already realize, establishes your defaults for paragraphs on your Web page. In the second line, `.special` is a class selector. This line tells the Web browser that when you assign the `special` class to a paragraph, the font of that paragraph should be white (`#ffffff`), and the background for that paragraph should be dark gray (`#333333`).



Notice that the selector starts with a period this time. This is important. This is the proper grammar because *this is a class selector*. You'll see that ID selectors have their own syntax (or grammar). You must use this syntax correctly and apply the styles according to that syntax, or they won't work properly.

In your HTML, you would use the following code to create a paragraph with the default styles:

```
<p>A regular paragraph would look like this</p>
```

And you would use this code to create a paragraph with the class styles you set up in your CSS:

```
<p class="special">A special paragraph would look like
    this</p>
```

In this example, the `.special` class can be applied to any element, such as a bullet in a list, a table cell, a paragraph, or any other element. You can also create the `.special` class for use just with paragraphs like this:

```
p.special{color: #ffffff; background-color: #333333;}
```



You might notice that the preceding example has no `font-family` property. You don't need to specify a new property unless you want to use a value that's different from the default. A good practice is to declare your font values in either the `body` selector or the `p` selector of your CSS. Then the fonts will be the same throughout your document, unless you indicate otherwise. If you decide you'd like to use a different font on a particular element,

create a class and declare a new value for the `font-family` property. When you apply the class to an element in your HTML document, it ignores the default setting you made in the body selector in favor of the class values.



The general rule is that the `property: value` pair closest to the element wins — sort of. The more-specific styles take precedence over less-specific ones. So, a general paragraph selector will lose out to a more specific class selector, which in turn will lose out to an ID selector.

In cases where nothing has been declared, the next value is used. So, using our example, the `.special` paragraph would change its font color to white and its background color to dark gray, but would keep the `font-family` as specified in the `<p>` tag because no new `font-family` was declared. Other things can affect how styles interact, but discussing those is beyond the scope of this chapter. You can find more information at the W3C Web site at www.w3.org/Style/CSS.

Notice that, in these example styles, the class selector looks a little different than the tag selector. In this case, we're using the HTML tag itself to select content for styling. So, if you use a tag selector of `p`, all paragraphs will pick up the style. You don't need to apply any extra code (other than the proper paragraph tags) in the HTML document, unlike the class and ID selectors, which require that you add some extra code so the browser knows where to apply the style.



So, to refresh where we are, with tag selectors, you just start with the tag. Class selectors start with a `.` (period) — this is an important thing to remember. The syntax rules of CSS must be followed carefully, or the CSS won't take effect for the particular selector or property that is incorrect.

ID selectors

An *ID selector* is the third way that styles can be applied to content on your site. In general, you use these as a group of styles that affect several pieces of your content that work together, such as navigation or sidebar content. There can only be one instance of each ID tag per HTML document. ID tags are useful for creating content groups with accompanying CSS that all work as a unit. An example of this in action is a CSS menu. By using HTML bulleted lists and a group of ID selectors, a designer can create a CSS menu that is easy to update, works well for all users, and is easy on the download times.

When you look at the code of an HTML document that works with CSS, you might see code that looks like this:

```
<div id="navBar">
  <div id="mainNav">
    <ul>
      <li> <a href="#">About</a> </li>
      <li> <a href="#">Products</a> </li>
      <li> <a href="#">Contact</a> </li>
```

```

        <li> <a href="#">Home</a> </li>
    </ul>
</div>
</div>

```

The "mainNav" and "navBar" sections of this HTML example use an ID selector for its style information: It's marked in HTML with the <div> tag, which encapsulates the content to be included in the style in the code. In this sample, two sets of ID selectors are used together to create the effect of *rollover buttons* (buttons on a Web site that change appearance when you place your cursor over them), but without using JavaScript or graphics (the usual way of creating this effect). The ID selector information for this particular example looks like Listing 3-1.

Listing 3-1: ID Selector Information to Create a Bullet List Menu Bar

```

#navBar ul a:link, #navBar ul a:visited {display: block;}
#navBar ul {list-style: none; margin: 0; padding: 0;}
#navBar{
    float: left;
    width: 20%;
    height:450px;
    margin: 0px;
    padding: 0px;
    background-color: #eeeeee;
    border-right: 1px solid #cccccc;
    border-bottom: 1px solid #cccccc; }

#mainNav{
    position: relative;
    margin: 0px;
    padding: 0px;
    border-bottom: 1px solid #cccccc;
    font-size: 90%;
    color:#000000; }

#mainNav h3{
    padding: 10px 0px 2px 10px; }

#mainNav a {
    display: block;
    border-top: 1px solid #cccccc;
    padding: 2px 0px 2px 10px; }

#mainNav a:hover{
    background-color: #dddddd; }

```

The preceding code, when applied to a bulleted list, will create a menu bar that interacts with users when they roll their cursor over the "button" areas. The visual cue of this effect helps users understand that they have their cursor over a link. These styles

- ◆ Set the display of the list items to “block.” This means that the element will be seen as a block element within the document with line breaks before and after it.
- ◆ Remove the bullets and set the margins and paddings of the list items to 0.
- ◆ Specify the position, color, borders, and size of the navigation bar.
- ◆ Set the borders and display attributes of the links.

The last style sets the color of the background of the links when the user’s cursor is over it.

Note the new syntax introduced here. ID selectors are indicated by starting with a #. Also notice that ID selectors are implemented in the HTML with `<div id="selectorname"> content </div>`. There is no need to apply styles to each element between the `<div>` tags. Everything contained between the `<div>` tags is seen as part of the whole. The browser understands that an `<h3>` heading occurring within the `<div id="selectorname"> </div>` tags is to be styled according to the `#mainNav h3` properties and values set in the style sheet. Again, if no style is set for an element, the default values are used.

In Figures 3-2 and 3-3, you can see that the difference between a basic HTML bulleted list and one with CSS can be very dramatic.

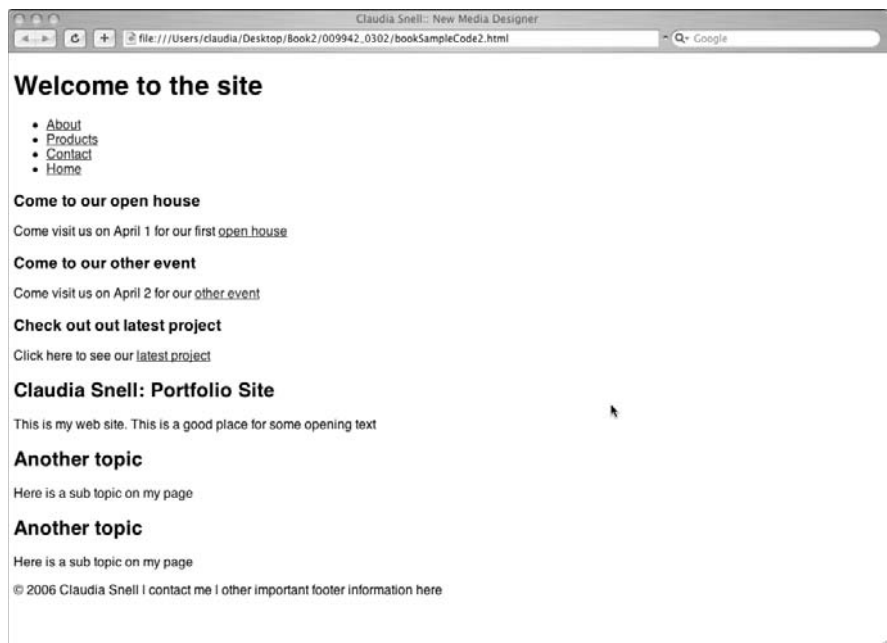
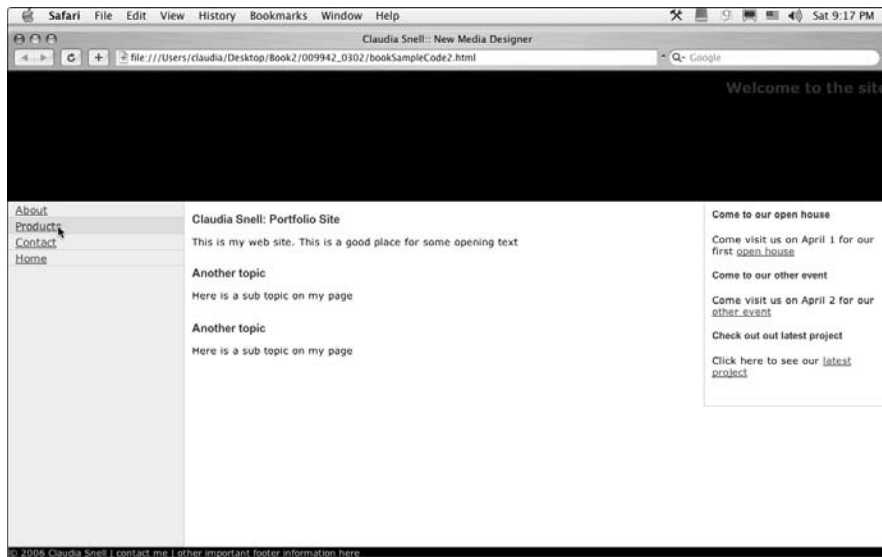


Figure 3-2:
An HTML
bulleted list
without CSS.

Figure 3-3: CSS styles convert the HTML bulleted list to a menu with rollovers.



Special Effects with CSS

There are a few tricks that you can do with CSS that you can't do with HTML. You can create custom rollover effects like the ones we show you in the ID selector example (see the preceding section), and you can set custom looks for links on your page. Everyone has seen the default, brightly colored, underlined links of HTML. CSS allows you to style those links in a more aesthetically pleasing way.

Styling your links

You can set a specific color for everything that has an `<a>` tag (links are created in HTML by using ``), like so:

```
a{
    color: #CC0000;
    text-decoration: none; }
```

This style automatically works on all link tags in your HTML document because it uses a tag selector.

Or, you can spice up the way your links look by creating styles for the different states: link, visited, hover, and active. We go into more detail about these next:

- ◆ **Link:** The default state of a link. The CSS selector for this is `a:link`.

- ◆ **Visited:** A link is considered visited when the user has clicked it. The CSS selector for this is `a:visited`.
- ◆ **Hover:** A link is in the hover state when the user's cursor moves over it. The CSS selector for this is `a:hover`.
- ◆ **Active:** A link is active at the moment it is clicked. The CSS selector for this is `a:active`.

Here is what the CSS code for these settings looks like:

```
a:link{
    color: #c00;
    text-decoration: none; }

a:visited{
    color:#FF6600;
    text-decoration: none; }

a:hover{
    color: #990000;
    text-decoration: underline; }

a:active{
    color: #990000;
    text-decoration: underline; }
```

Like with other selectors, you can also nest these:

```
a:hover, a:active{color:#990000; text-decoration: underline;}
```

Note also that you can have more than one set of these selectors. You can create a default set and then create others for use in different areas of your page, such as in the footer:

```
#footer{
    clear: both;
    border: 1px solid #cccccc;
    font-size: 75%;
    color: #ffcc66;
    background-color:#000000; }

#footer a:link{color:#ffcc66; }
#footer a:visited{color:#ff0000; }
#footer a:hover{color:#ffffcc; }
#footer a:active{color:#FF6666; }
```

Cool headlines

Another interesting thing you can do with CSS is add some style to your heading tags.

```
h2{font-weight: bold;
color:#333333;
border-bottom: #333 1px solid;
font-size: small;}
```

Figure 3-4 shows how this code affects the `<h2>` tags in the HTML document. Note how the underline extends beyond the end of the text, making the heading a nice, page-separating device. Traditional HTML underlining is constrained to the length of the text, but CSS styles stretch across, making them more attractive and useful as page dividers.

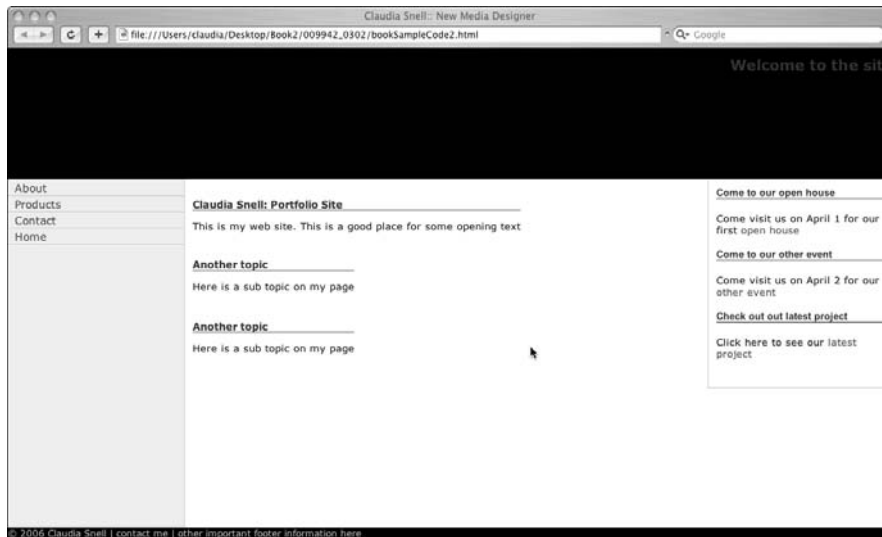


Figure 3-4:
These h2
headlines
are styled
with an
underline.

Custom padding and margins

In CSS, you can set custom padding and margins. With plain HTML, you can set a padding value, but it's the same on all four sides. The same goes for margins. With CSS, though, you can set top, right, bottom, and left independently — you can have four completely different values! The flexibility this allows is a tremendous help to designers and makes *spacer graphics* a thing of the past. The old, spacer graphic technique was to use transparent GIF files or nested tables with tiny spacer cells in them to try to get nice layouts. These techniques create very messy, hard-to-maintain code. And it's a nightmare for anyone trying to visit your Web site with devices other than browsers. (Screen readers for visually impaired users become confused in the sea of weird table structures and unnecessary GIF files.)

Figure 3-5 shows the CSS box model, which is how browsers and other devices actually see your content. *Padding settings* affect the space between the content and the “box” that contains it. *Margin settings* affect the actual

space around the content. For more on padding and margins and how to use them effectively, visit the W3Schools CSS tutorials and reference at www.w3schools.com/css.

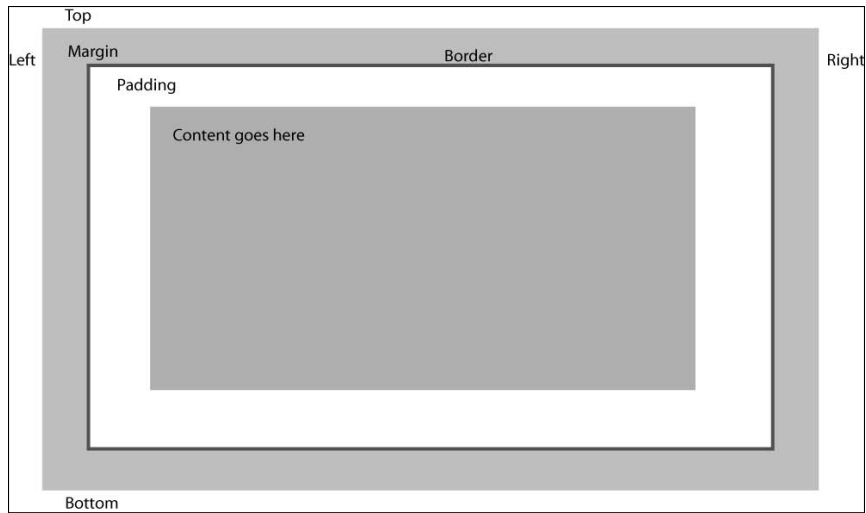


Figure 3-5:
The box
model,
showing
padding and
margins.

Custom borders

Just like padding and margins, in HTML, you either have a border all the way around something or you don't. Also, you're limited to what you can apply a border to.

CSS allows you to put a border in just the place you want it. If you want just a top border, set `border-top` values, and you'll have only a top border. You can also apply borders to many more types of elements, such as heading tags and paragraphs. CSS borders also have more values to choose from. Designers can choose colors, positioning, and thickness; and some browsers support type (solid lines, dotted lines, double lines, and so on).



Figure 3-5 shows where the borders will be placed in relation to the margins and padding. Keep this in mind because it can affect your layout by placing borders in ways that you didn't intend — too close to the content for instance.

Chapter 4: Introducing Web and Graphics Software

In This Chapter

- ✓ Getting your graphics software
- ✓ Looking at Web-design software
- ✓ Taking a tour of Photoshop
- ✓ Getting to know the tools of the trade

Many tools are available to help you with the hands-on tasks of Web site design and maintenance. Eventually, choosing the right tools can be a matter of preference, but we do know a few must-have tools for every Web designer. This chapter gives you a brief overview of these must-have tools plus a quick-and-dirty tour of Photoshop CS4.

Choosing Graphics Software

So many graphics software packages are available, and so many manufacturers tout that their product can do everything and anything, that you might be confused when choosing graphics programs. First, figure out what kind of work you want to do; then, buy industry-standard software for that work. If you plan on creating lots of vector graphics, for instance, look for software that specializes in that; if you plan to work with photos, look for software that has tools for that. By sticking to industry-standard software, you can ensure better compatibility with other designers; better resources; better employability as a designer; better toolsets; and with practice, better overall quality of work.



Another benefit of sticking with industry standards — and especially products from Adobe — is that their interfaces share similar features. They also work together to help users get the job done more efficiently. More about integrated workflows is discussed throughout this book.

Adobe Photoshop

The industry-standard software for working with bitmapped (sometimes called raster) graphics is Adobe Photoshop. (Read about bitmapped images in the sidebar, “Bitmap versus vector images, Part 1.”) Photoshop offers a powerful set of tools that allow you to create and work with any graphics you need for your site, from concept to completion.

People in many different aspects of working with photos and graphics all use Photoshop. Photographers, designers, editors, and video professionals all use it because of its superior tools and vast number of uses. In Photoshop, you can create all original graphics, draw artwork, adjust or manipulate photos, apply effects, resize, and optimize — virtually anything you might want to do with images is possible with Photoshop. Figure 4-1 gives you a peek at the Photoshop interface. The “Touring Photoshop CS4” section, later in this chapter, introduces you to the Photoshop interface and some of the program’s features.



Figure 4-1:
The Adobe
Photoshop
interface.

Adobe Illustrator

The industry standard for creating vector graphics is Adobe Illustrator. (Read about vector images in the sidebar, “Bitmap versus vector images, Part 2.”) Like Photoshop, Illustrator has a very powerful set of tools; the difference between the two is that Illustrator specializes in creating vector images.

If you plan on creating logos or line art, Illustrator is an important piece of software to have. It's also useful to own if you want to do Flash design. Flash works well with vectors because all the drawing tools in Flash are vector based. When you add Illustrator to your set of tools, you can create Flash-friendly illustrations and other artwork. Figure 4-2 gives you a look at the Adobe Illustrator interface. Notice how very similar it is to the Photoshop interface.

Other professional-grade graphics software

Other graphics software that you might encounter includes Adobe Fireworks, Adobe FreeHand, and Adobe Flash. As we mention earlier, all Adobe products have similar features, enabling designers to move easily from one product to another. A few professionals use Corel products, such as Painter, Paint Shop Pro, and CorelDRAW.

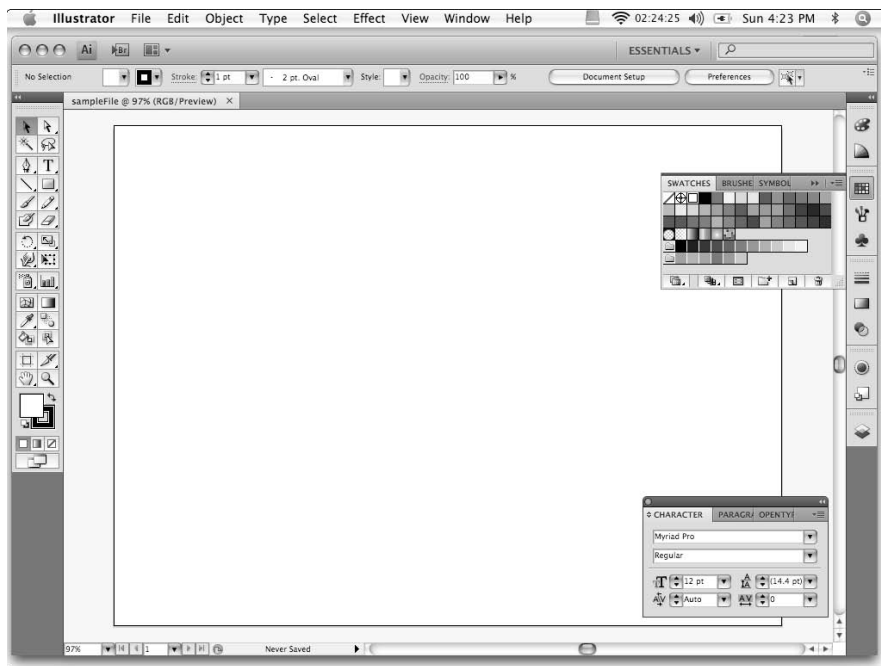


Figure 4-2:
The Adobe
Illustrator
interface.

Bitmap versus vector images, Part 1

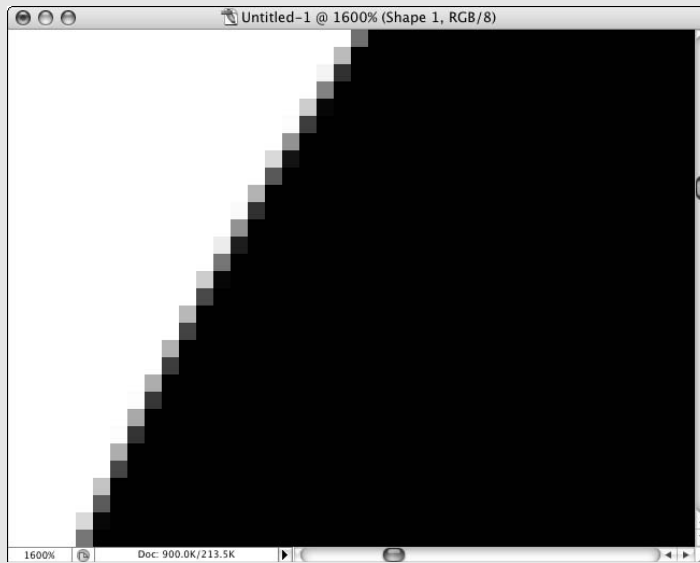
The trick to successful graphics creation is understanding the strengths and weaknesses of the different formats and types of graphics. *Bitmapped images* are made up of *pixels* (px; tiny dots of color) laid out on a grid (or mapped). Think of a bitmap as a screen equivalent of a printed image — lots of dots that, together, make a picture.

Computer monitors display images at 72 or 96 dots per inch (dpi). Sometimes, this is referred to as *ppi* (pixels per inch). When you prepare graphics for a Web page, make sure you save them with 72 dpi. By doing so, you make the file sizes smaller without making the image on-screen look worse.

Printers work with 150–300 dpi. This is why 72-dpi images don't look great when you print them. Although you might be tempted to use higher resolutions to make your images

look better, it doesn't work that way. Always remember that larger files download slower, which is bad for your users.

Bitmapped images resize smaller fairly well. For example, a 400-x-600-pixel image downsizes to 200 x 300 pixels fairly well. Scaling up images is another story, though, because the dimensions are made larger, but the amount of information stays the same. The computer has to make its best guess as to what color to make the additional dots needed to enlarge the picture. This is *interpolation*, which creates very jagged-looking images. If you've tried to scale up an image and noticed little stairs appearing along the edges of elements in the image, you've seen the effect of trying to scale up. This figure shows a bitmap image that has been scaled up.



Bitmap versus vector images, Part 2

Vector images are made up of mathematical statements that define individual objects that are made up of points, lines, and fills. Think geometry class. Because they're created by math, they scale up and down very well. They're also relatively easy to edit because the elements are separate objects as opposed to a series of self-contained dots. Designers can select an element, move points, and change properties of the lines and fills very easily.

Vector art works well for logos and other line art. Because of how they're created, they have very clean lines. The downside is that they don't do well with pictures and other types of images that require lots of tonal changes and soft transitions between those tones.

You will need to convert (*rasterize*) vector graphics into bitmaps before using them on a Web page — with the exception of vector art created in Flash. The Flash Player supports vector graphics where browsers don't. Some plug-ins that are available can support vectors, but not many people download and install them. The main strength of vector art is that you can

have very clean logos and line art that can be scaled to any size you need.

If someone sends you a graphic file, you can tell whether it's a vector or bitmap by examining the pieces in your graphics software. Depending on the software you use and the software the graphic was created in, you might get a warning that shapes will be rasterized. This message is telling you that the software is having trouble dealing with vector-based shapes in the graphic, so the software will convert the shape into a bitmap. Another way you can tell the difference between vector and bitmap is by selecting the objects within the graphic. If a series of lines, curves, and dots appears all around the edges of the object, you have a vector. If you get a box that contains the object or no lines at all, you have a bitmap.

Note: Simply opening a graphic in a vector graphics program, such as Adobe Illustrator, and then saving it as an Illustrator file does not make the graphic a vector. Remember that a vector is a math-based graphic — think geometry class and lines, curves, and points.

Considering Web-Design Software

Although you can code Web pages entirely in Notepad (and some folks do), we advise you to look at Web-design software packages. Some Web developers will tell you that these packages are bloated. The bloat they're referring to is actually a set of helpers and features that, if used properly, can help you be a better Web designer. You might also hear that these packages write bad code. There might be some truth to this, but it's mostly just coder snobbery. Just choose the tools that work best for you.

You shouldn't, however, substitute software helpers for learning HTML (HyperText Markup Language), CSS (Cascading Style Sheets), and the other technologies you intend to use. If you don't learn how it all really works under the hood, you can't fix problems beyond the software's capabilities.

You also won't be able to clean up the mistakes that WYSIWYG (what you see is what you get, or *WHIZ-ee-wig*) editors often make, such as leaving tags behind after you deleted the content.

In actuality, most people (even code snobs) can't tell the difference between pages hand-coded in Notepad and those hand-coded with Dreamweaver. In fact, Dreamweaver, GoLive, and Visual Web Developer have many tools that can help you write better code than someone with a stripped-back text editor. For instance, Dreamweaver and other development tools can help keep your code formatted for easy reading. Text editors will not help you in this regard, so you'll have to be very vigilant while coding or you can end up with very messy looking code (usually a tell-tale sign that someone used a text editor). Also, you can set up Dreamweaver to highlight coding errors in bright yellow. Someone using a text editor isn't alerted to even a basic typo in their code, which is a very small problem that can result in a really messed-up-looking page. Dreamweaver shows you exactly where your mistake is and also tells you what the mistake is. Comparatively, with a plain-text editor, you're left to find and figure it out for yourself. See Chapters 5 and 6 of this minibook for more about using Dreamweaver for your Web-design projects.

Although Adobe Dreamweaver is the industry standard and the software most commonly used by professionals, other common, professional-grade, Web-design software packages are available, including Adobe GoLive and Microsoft Visual Web Developer 2008 - Express Edition.

The choice of what tools to use is often personal for Web designers and developers. If Dreamweaver doesn't suit your personal tastes, one of the other two major packages might. Of course, you might not have this luxury of choice if you're working as part of a team or in a corporate environment. You need to be flexible and use those tools to ensure compatibility. You might even need to use a blend of tools in some corporate environments, especially when multiple sites, legacy sites, and content management systems are involved. These situations are beyond the scope of this book, but aspiring professionals should be prepared to be flexible and willing to learn.

Some Web-design software packages fall between the range of Notepad and Dreamweaver. You might encounter BBEdit for Macs or HomeSite for Windows. These resemble stripped-back versions of the more full-featured packages. And some programmers prefer hardcore development environments, such as Visual Studio. Some designers prefer these; some don't.



Again, learn HTML, CSS, and other coding/scripting technologies you intend to use on a regular basis, and pick your tool based on what you need and want it to do. You can develop quality Web pages and sites in any of the packages we mention here.

Deciding on Multimedia Software

If you plan to include video, audio, presentations, and slide shows on your site, consider what multimedia software you want to use. What you choose depends on what you want to do and your target audience. For instance, if you want to do a lot of animations, Toon Boom Studio (www.toonboom.com) specializes in animation and is probably a good choice for your work. To decide what works best for your situation, take a look at this list that briefly describes the most common multimedia programs available (visit the listed URLs for more information), and then pick the solution that best fits your needs:

- ◆ **Apple QuickTime and QuickTimePro:** QuickTime is available for both Macs and PCs. The player is free, but the Pro version of the software that allows you to create multimedia costs \$29.99 (at the time of this writing). QuickTime Pro allows you to import and edit video, *capture* (record) audio, create slide shows, and do other similar types of projects. This tool is relatively easy to use but also doesn't have the more robust features of the more expensive and more sophisticated multimedia development software, like Flash. To find out more, visit

www.apple.com/quicktime/download

- ◆ **Microsoft Windows Media:** Windows Media Player (available for Windows) is the version used for creating projects. Windows Media Encoder can be used to capture audio or video. Windows Media Encoder is available for PCs. For more information, go to

www.microsoft.com/windows/windowsmedia/default.aspx

You might notice what appears to be downloads for Mac computers as well. Be sure to read reviews; the Mac version is quite old and doesn't work well on newer machines (in some cases, not at all). Don't despair; there is a product for your Mac as well. It's a utility called Windows Media Components for QuickTime and is made by Flip4Mac (www.flip4mac.com/). (The product page can be found here: www.flip4mac.com/wmv.htm.)

- ◆ **Adobe Flash:** Flash has a "player" version and a "developer software" version. Adobe Flash is used to create animations and Web interfaces (Web sites). Flash can compress and play back video, but you can't capture or edit audio or video with Flash. Flash is available for both Macs and PCs. Check out Book V for more about Flash. You can find information about Flash at <http://www.adobe.com/products/flash/>.

A lot of software programs have the capability to create multimedia files. Some software manufacturers claim that their products can create Flash; the reality is that these products can create SWF files (the files that play in the Flash plug-in in the browser), but there's only one, true Flash program. If you want all the functionality of Flash, you need to get the Adobe Flash software and create your SWF files with that.





Just because a piece of software *can* do something doesn't mean that it does it well. For example, you can certainly use PowerPoint and Word to create Web pages and Web presentations, but they don't do it overly well. Using PowerPoint and Word to create a presentation is a good idea; when you want to put your presentation online, however, convert it to a more Web-friendly format via specialized software.

Touring Photoshop CS4

Before you dive right into Photoshop, take a minute to get familiar with the basic features. This quick tour can help you get acquainted quickly.

The default workspace

When you click the Close button on the Welcome Screen (or first thing, if you disable the Show This Dialog at Startup check box), the default workspace appears. (Refer to Figure 4-1.) The most commonly used panels are laid out on the screen, including the Tools panel, Navigator, color sliders, swatches, styles, history, actions, and layers.

You can customize the workspace to your needs. If you find that you like a particular layout of panels or you use some more than others, set up your own screen layout and save it. This feature can be very handy because it saves you time rearranging the screen when you have different preferences for different tasks.

To create a new workspace, follow these steps:

- 1. Arrange the screen how you like and then choose Window⇨Workspace⇨Save Workspace.**
- 2. In the Save Workspace dialog box that appears, give your workspace a name and then choose any of the options that you want to save (panel locations, keyboard shortcuts, menus, and so on).**



You can have custom settings for each of these or only some of them if you like.

- 3. Click Save.**

To use a workspace you saved, choose Window⇨Workspace⇨*the name you gave it*.

You can also delete a saved workspace. Just choose Window⇨Workspace⇨Delete Workspace, choose the name of the workspace you want to delete from the menu, and then click Delete. Confirm the deletion by clicking Yes in the pop-up window.

Some specialized workspaces are available as presets, and all are available from the Window⇨Workspace menu. Explore them to see whether they work for you and the tasks you're working on. Use the reset options to put everything back where it started.

The work area

The work area, not to be confused with the workspace, which is the whole Photoshop interface layout, is the window where your graphics will be. Think of it as the canvas for your work. Figure 4-3 shows the New dialog box where you set up a new work area.

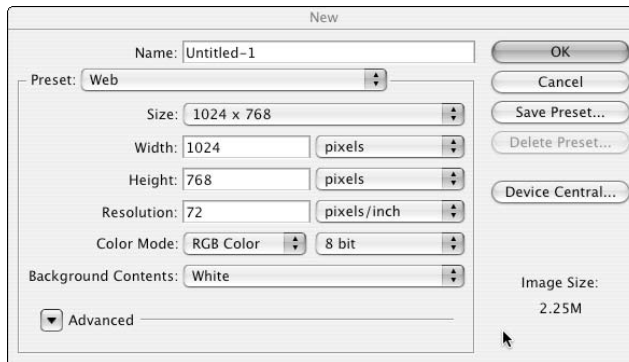


Figure 4-3:
The New
dialog box.

To create a new work area

1. Choose File⇨New.
2. At the top of the New dialog box that appears, give your new graphics file a name in the Name text box.
3. Set the dimensions in the Width and Height fields or select a preset width and height from the drop-down menus.



Choose pixels as the unit of measure because you're designing for the Web. The Web doesn't use inches.

4. Set the resolution to 72 pixels/inch.

The higher numbers are for print design. (See the explanations of graphics types in the sidebars, earlier in this chapter, that compare bitmaps and vector graphics.)

5. Select the RGB Color Mode.

Book II, Chapter 4 explains color space.

6. Set the Background Contents of your file to be transparent or to have a color.

7. Click OK to create the file.

Photoshop creates the new graphics file and places it on-screen.

8. Save your document by choosing File⇨Save As.

9. In the Save As dialog box that appears, choose a location, name the file, and then click Save.

Make sure you save the file to your production files folder so you can find it easily later.



The Advanced button

Check out the Advanced button near the bottom of the New dialog box. When you click this button, you find settings for Color Profile and Pixel Aspect Ratio. These are important if you're working in print or video design. *Color Profile* helps you set up your document so that it can communicate colors more accurately. (Go ahead — click it to check out the options.) *Pixel Aspect Ratio* is for setting up graphics for use in video. If you click that drop-down menu, you see many options as well. Video likes pixels that are rectangular. Different projects have different needs, so you have several options. For Web design, you want square pixels.

Another way to start a new workspace is to open a graphics file by choosing File⇨Open, browsing to the image that you want, and then selecting it. Click Open. When the file opens, the image is in a layer called Background. We discuss layers next.

Layers and the Layers panel

One of the most important panels is the Layers panel. The elements that make up your graphics are held in *layers*, which you can think of as sheets of paper stacked one atop another. The big difference between layers and actual sheets of paper is that the layers can interact with each other. Some layers can be transparent so that layers underneath show through, or you can use *blend modes* (layer attributes that specify how a layer interacts with other layers below it) and *opacity* (the degree to which an object is not transparent) to blend the elements of your graphics. Figure 4-4 shows the areas of the Layers panel, and the following list gives you a crash course in working with layers:

- ◆ **Create a new layer** by clicking the Create New Layer icon.
- ◆ **Duplicate layers** by using any of these methods:
 - Select the layer you want to duplicate and then press Ctrl+J (Windows) or ⌘+J (Mac).

- Right-click the layer you want to duplicate and then choose Duplicate.
- Click the layer you want to duplicate and then drag down to the New Layer icon.
- ◆ **Reorder layers** by dragging them higher and lower in the window: Click a layer that you would like to move in the stacking order and then drag it up or down the stack.
- ◆ **Create new layers** by using the Text or Shape tool, as follows:
 - a. Click the Text tool (the large T in the tool panel).
 - b. Begin typing in the workspace.

The text appears on its own layer.
- ◆ **Delete a layer** by selecting the layer to be deleted and then clicking the Delete Layer icon (looks like a trash can).
- ◆ **Make a layer invisible** by clicking the corresponding eye icon for the layer you would like to make invisible. To make it visible again, click the empty box where the eye icon used to be. (The eye disappears when the layer is invisible.)

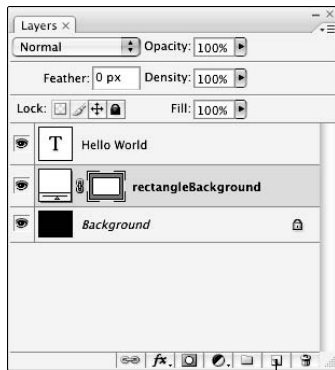


Figure 4-4:
The Layers panel and its many features.

Click to create a new layer.

Tools and the tool panel

In this section, we look at the most commonly used tools. Like with other features we introduce to you, we recommend that you experiment with these tools to get an idea of how they work.

Photoshop files and keeping editable copies

When you create a new document in Photoshop, the file saved by Photoshop is a PSD document by default, which contains all your layers and editable text (as long as you don't convert or rasterize the text). See the "Bitmaps versus vector images, Part 2" sidebar in this chapter for information about rasterizing. Make sure you keep a copy of your original PSD files so you can quickly make changes or build new graphics based on the originals. After you save a graphic as a Web graphic file, you can't edit certain elements, such as text, in the file. Without the editable file, you might not be able

to make changes to your graphics at all. At the very least, a simple task will be much more difficult and time consuming. On the other hand, on some occasions, you must rasterize text. If, for instance, you create a graphic with a special font and you intend to pass the Photoshop file to someone who doesn't have that font on their machine, you need to rasterize the text (make it into a graphic layer). Otherwise, the other machine substitutes a different font for the one you used. Make sure you keep a copy of the original PSD file for editing, though.

Selection tools: Marquee, Lasso, and Magic Wand

The Marquee, Lasso, and Magic Wand tools select areas of a layer to be affected by filters or effects. They also help you work on a specific area of a layer without affecting areas you don't want to touch.

- ◆ The **Marquee tool** gives you a rectangular or elliptical marquee. With the marquee tools, you select an area of your graphic. Depending on which version of the tool you select, the selection will be rectangular or rounded.
- ◆ The **Magic Wand tool** selects everything in your graphic that matches the color you click with it.
- ◆ The **Lasso tools** allow you to make customized selections. You can click around an area and create a selection of any shape with the Lasso tools.



Click the small black triangle in the lower-right corner of any tool icon to see the variations of the tool, like Rectangular Marquee or Elliptical Marquee. Many tools have different versions. You can tell which ones do by looking for the little black triangle. When you click it, the other versions pop out as a mini menu with icons. You can then select which version of the tool works best for the task at hand.

To use a selection tool, select the layer you want to work on and then select the tool that will work best for what you're doing. (See the previous list.) Then take the appropriate action based on the tool you chose.

- ◆ **Marquee tools:** Click and drag over the area you want to select.
- ◆ **Magic Wand tool:** Click an area that's the color you want to select, and your selection appears as a dotted line around the selection. The dots,

often described as *marching ants*, move along the line (see Figure 4-5 for an example) — the marching ants appear around all selections regardless of what tool made them.

- ◆ **Lasso tools:** Click along the edge of the object you want to select. Make sure you go all the way around, and then end by clicking the start point: In effect, you “draw” a complete container around the object you’d like to select.

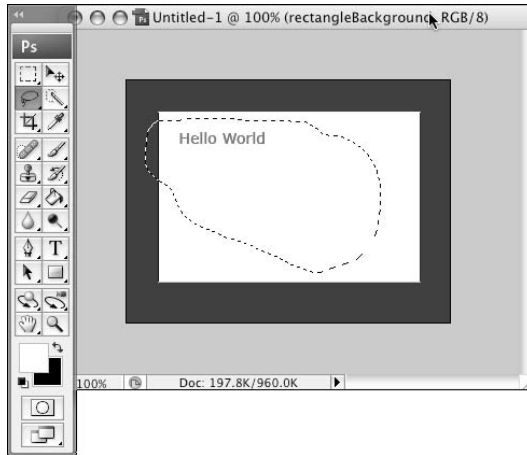


Figure 4-5: Making a selection with the Lasso tool.

Playing with layers

If you haven't worked with layers much, follow these steps to get started playing with blend modes and opacity to see what the different settings do:

1. **Open a photo image.**
2. **Duplicate the layer. (See instructions in the “Layers and the Layers panel” section of this chapter.)**
3. **Select the top layer and then select a blend mode from the drop-down menu at the top of the Layers panel.**

By default it will have Normal selected; when you click the menu, you see the other options, such as Multiply, Lighten, Darken, and Soft Light (to name a few).

4. **Experiment with the different blend modes to see what they do.**

You can also make multiple duplicate copies, use different blend modes on each, and then change the layer order or opacity on the different layers.

Undo the changes via the History panel, which shows a list of the actions you did in the file. Open this by choosing Window⇨History. To step backward, select the step in your process you'd like to go back to.

Remember: As long as you keep at least one layer that holds the original image, you can always delete layers that don't look so good.

The Move tool

Use the Move tool to (what else?) move elements of your artwork around the work area. You just select the layer that holds the artwork you'd like to move, and then click and drag inside the work area window until the element is where you want it.

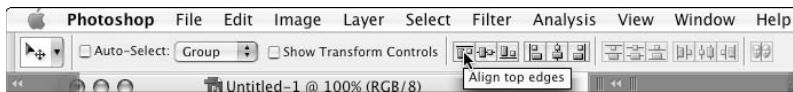


Interested in more precise movement? To move objects 1 pixel at a time, select the layer that contains the object you'd like to move, select the Move tool, click anywhere in the work area, and then use the arrow keys on your keyboard. To move an object 10 pixels at a time, hold down the Shift key while pressing an arrow key.

When you use the Move tool in conjunction with the Align icons, you can align multiple objects at once; just follow these steps:

1. **Select the layers you want to align by pressing and holding the Ctrl key (Windows) or ⌘ key (Mac) and clicking each layer.**
2. **Click the Move tool.**
3. **Click the appropriate Align icon (located at the top of the screen; see Figure 4-6).**

Figure 4-6:
The Align tool helps you line up elements in your design.



The Crop tool

Use the Crop tool to cut down your file and remove any extra space floating around the main subject area. You can use the tool freehand-style by selecting it and then clicking and dragging a box around the area you want to keep. You can adjust the box (if you need to) and then press Enter/Return. The entire document is cut down to the size of the selected area.



Any elements that remain outside the box are deleted — completely. Make sure you don't cut off anything unless you mean to. Duplicating the image you're cropping and then working on the copy is a good idea; that way, you'll always have the original to go back to.

A more precise way to use the Crop tool is to set dimensions that you want your cropped document to be.

1. Open the photo you're cropping and then click the Crop tool.

Notice how the options available in the panel at the top of the screen change as you choose different tools — this is a *context-sensitive* panel. It allows the interface to make a lot more tools available to you without cluttering up the screen.

2. At the top of the screen, type a number in the Width text box and another in the Height text box; also, enter a number in the Resolution text box.

If, for example, you want to make a 75-x-75-pixel icon from a 600-x-400-pixel photo, type **75 px** in the Width text box and **75 px** in the Height text box. A resolution of 72 dpi works because computer monitors don't require a higher resolution; see the sidebar information earlier in this chapter about bitmaps and resolutions.

3. Click and drag the Crop tool over the area you want your image cropped to.

Everything outside that area will be chopped off.

When you click and drag your crop area, the result is a perfect 100-x-100-pixel square at 72 dpi. Make sure you place your box over the area you want to keep; everything else will be lost when the crop is finalized. You finalize the crop by pressing Enter/Return, or by clicking the check mark at the top right of the panel at the top of the screen.

Move around the crop area until you have the perfect image framed in the crop area, and then either press Enter/Return or click the check mark in the contextual menu at the top of the screen. To remove the settings, just click the Crop tool and then choose Clear from the menu. See Figure 4-7 for an example of this technique.

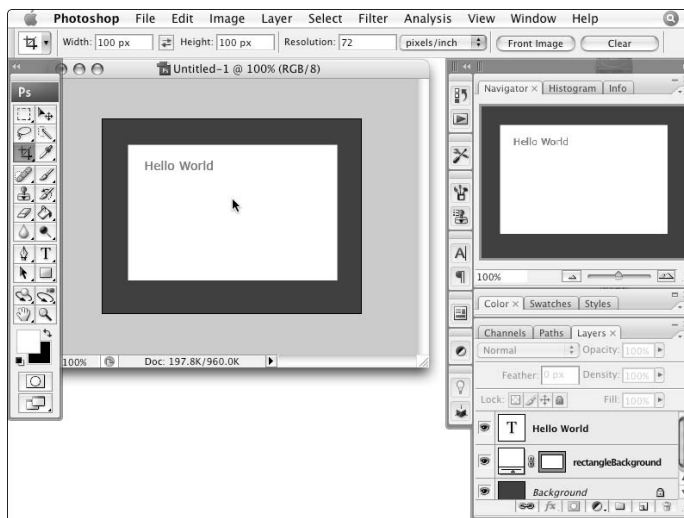


Figure 4-7:
Cropping
to exact
dimension
and
resolution.

The Slice tools

With the Slice tools (the tool in the Tools panel that looks like a small knife), you can take a large graphic and cut it into smaller graphics to be exported from Photoshop as graphics and an HTML document. This is useful when you design an entire interface and you need to break it up into the separate graphics that make up the working HTML interface. Figure 4-8 shows a Photoshop file that's been sliced. See Chapter 7 of this minibook for more information about using this tool.

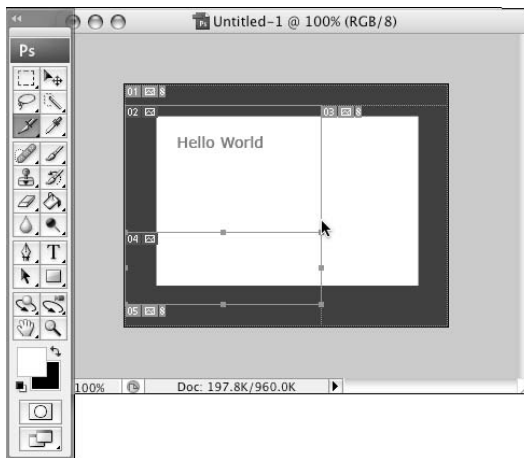


Figure 4-8:
A sliced
Photoshop
file ready for
export.

The Brush and Pencil tools

Use the Brush and Pencil tools to paint or draw in your document. Select one of the tools — for example, the Brush tool — and a menu appears at the top of the screen. (See Figure 4-9.) You can choose what color to paint with, the style and size of brush, and many other options.

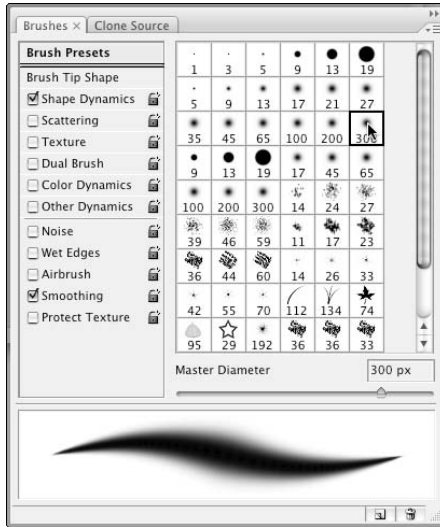
The Eraser tool

The Eraser tool does exactly that — it erases parts of a layer. To use it, select the layer you want to erase things from, click the Eraser tool, customize its size and other properties in the menu at the top of the screen, and start erasing.



Things that are erased are gone forever (see Figure 4-10). The Eraser tool is a *destructive tool* — what it erases is really gone. We highly recommend that you duplicate the layer and make the extra layer invisible before you start erasing so you have a backup. Refer to the “Layers and the Layers panel” section for instructions on making a layer invisible.

Figure 4-9: The Brush tool has many options to help you work with your graphics.



This area has been erased.

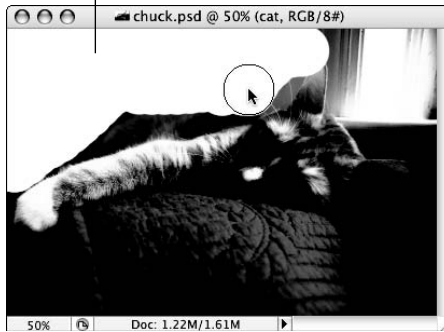


Figure 4-10: If you erase part of a layer, it is gone.



An alternative to using the Eraser tool is to use layer masks. When you use a *layer mask*, part of the layer becomes invisible. It's still there, but it's hidden from view. When you use a layer mask, what you “erase” doesn't go away forever. This option offers a nondestructive way to alter your images.

The Paint Bucket and Gradient tools

The Paint Bucket and Gradient tools color an entire layer or an entire selected area with the color of your choice. The Paint Bucket will just dump the color into the layer or selected area, filling it entirely. The paint brushes allow you to, well, actually *paint* the color in where you like. The Gradient tool fills the area with a *gradient fill*; you can select one from the context-sensitive menu or customize the gradient in the dialog box that opens when you double-click a preset.

Putting the Paint Bucket tool through its paces

To use the Paint Bucket tool, follow these steps:

1. **Select or create a layer or select an area of a layer.**
2. **Double-click the foreground color tile, select a color from the Color Picker (shown in Figure 4-11), and then click OK.**

The foreground/background color tiles are at the bottom of the tool panel. They are two overlapping colored squares: The top one is foreground, and the other is background.

When you select a color from the Color Picker, your cursor looks like an eyedropper.

The new color is shown in the top half of a rectangle at the top/center of the dialog box. The original color is shown in the bottom half for comparison.

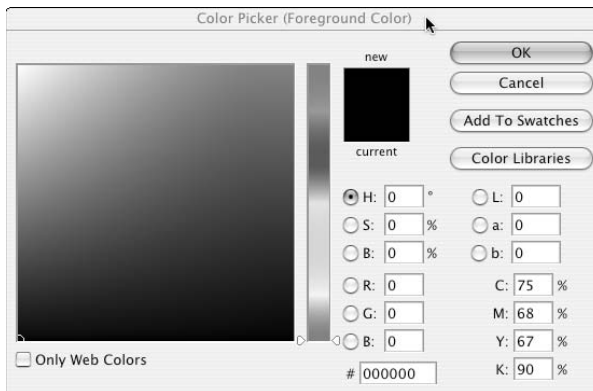


Figure 4-11:
Choosing a color to fill a layer with.

3. **Click the Paint Bucket tool and then click inside the work area or selected area.**

The chosen area is filled with the color you pick in Step 2.

Experimenting with the Gradient tool

Gradients have different types — different ways of displaying. Although you have several to choose from, two common ones are

- ◆ **Linear:** A *linear gradient* displays from left to right, top to bottom, or diagonally — in a linear way, from one color to the other.
- ◆ **Radial:** A *radial gradient* displays in a circular fashion from one color to the other. One color is a round circle, and the gradient radiates outward around it.

Use the icons in the gradient tool contextual menu at the top of the screen to experiment with the different types to see what they look like.

The contextual menu also contains a drop-down menu at the upper left that features a gradient sample, instead of words, as the menu choice. This is the menu that you use to select the colors of your gradient. Click the menu button to see all the available preset gradients. The more exciting way to work with gradients is to click the color swatch in the menu itself, which launches the Gradient Editor (shown in Figure 4-12). In the Gradient Editor, you can change the colors of the gradient, add colors to the gradient, change the position of the colors within the gradient, and save gradients that you create (as a preset).

To change colors, select one of the presets that are available to start, and then either double-click one of the color chips that are displayed under the sample of that gradient or click the Color drop-down menu. Either path launches the Color Picker tool. You can move the position of the colors within the gradient by clicking and dragging the color chips or by clicking a color chip and then changing the number in the Location box. The “right” way depends on your personal tastes. Experiment with it and have fun.

To use the Gradient tool, follow these steps:

1. **Select a layer or create a layer or select an area of a layer.**
2. **Click the Gradient tool.**
3. **Click the Click to Edit menu at the top of the screen to open the Gradient Editor. (See Figure 4-12.)**

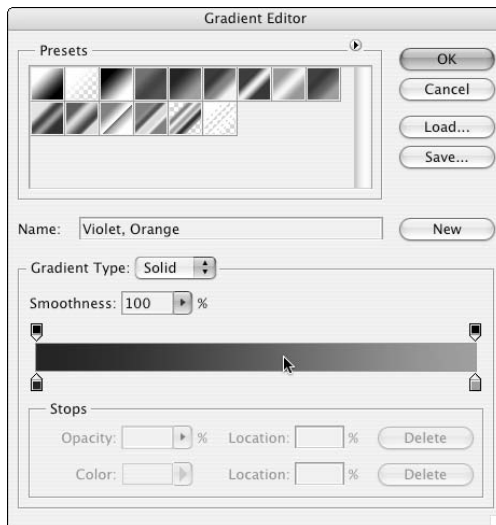


Figure 4-12:
Calling up
the Gradient
Editor tool.

4. Select a Preset or adjust the settings to create your own gradient; then click OK.
5. Click and drag across the layer you want to fill with the gradient.



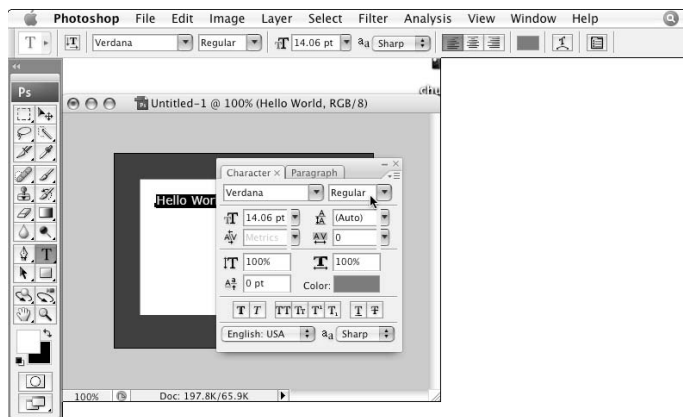
The layer is filled with the gradient you create in Step 4.

If you don't like the result, try again. Adjust the gradient, or keep the one you already have and just drag again. Repeat until you get what you want.

The Text tool and panel

You can add text to your graphics with the Text tool. To use this tool, click it, click in the work area where you want your text to appear, and type. A new layer is automatically created with your new text. The Text tool has its own menu bar and a panel that contains more options for formatting, including options for handling paragraphs. Figure 4-13 shows the Text menu bar and panel.

Figure 4-13:
Use the Text menu bar and panel to set the properties of your text.



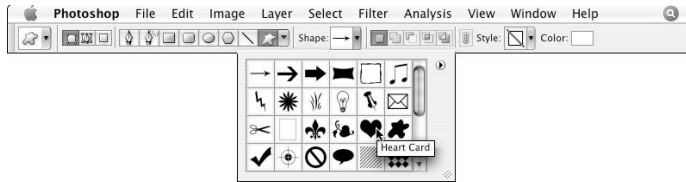
Vector tools: Shape, Pen, and Path Selection tools

Photoshop has a few vector graphics tools, and these can be tricky to use. We recommend experimenting with them until you feel comfortable using them. Here are the highlights:

- ◆ **Shape tool:** Draw shapes, such as rectangles and ellipses. To use it, just select the tool, and then click and drag in the work area. The Shape tool has an option that allows you to draw a variety of preset shapes, see Figure 4-14.

- ◆ **Pen tool:** Draw lines, curves, and shapes. With the Pen tool, click in the work area to create points; after you create all your points, Photoshop connects them to create a shape.
- ◆ **Path Selection tool:** Edit *paths* — the lines in a vector graphic — that you have created.

Figure 4-14:
Using the
Shape tool.

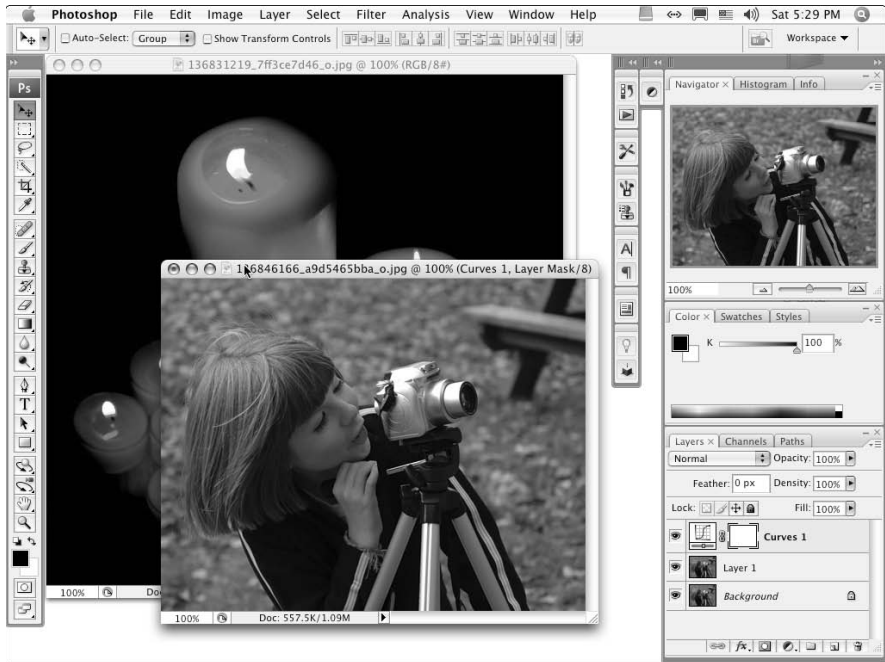


The Magnify and Hand tools and the Navigator panel

Use the Magnify and Hand tools, along with the Navigator panel, to help you move around your document and zoom in or out for a better look.

- ◆ **Magnify tool:** To zoom in on a portion of the work area, choose the Magnify tool and then click that area. Double-click the Magnify tool icon to go directly to 100 percent. Use the Navigator panel (by sliding the small red triangle under the preview window) or enter a percent value into the box at the lower left of the document window to zoom back out.
- ◆ **Hand tool:** This tool moves your viewing area around the document; it doesn't move any of the artwork. Select the tool, click within the document work area, and then drag the work area around. The whole Photoshop file appears to move around within the document window area. You're not actually moving the graphics — only the visible area. This is particularly helpful if you are zoomed in far enough that your whole document is no longer visible in your document window. You can drag the area that you're looking at without having to move the actual graphics around.
- ◆ **Navigator panel:** This panel performs the same functions as the preceding two. The slider zooms in and out, and the red box indicates your viewing area. You can also use the Navigator to orient yourself in the document. Click the red box and drag it around the document thumbnail so that you can change the viewing area as you work. See Figure 4-15 for an example of how the Navigator works. If you can't see the Navigator on your screen, open it by choosing Window⇨Navigator.

Figure 4-15:
Use the Navigator to zoom in and out and move around your work area without moving the artwork.



Chapter 5: Getting Started with Dreamweaver

In This Chapter

- ✓ **Touring the Dreamweaver interface**
- ✓ **Creating a Web site**
- ✓ **Creating pages and supporting files in Dreamweaver**
- ✓ **Testing your pages and validating the code**
- ✓ **Publishing your Web site**

Dreamweaver CS4, the industry standard software for Web design, has some great features and some very useful tools to help designers and developers to be more efficient. Some of the features include code formatting, comment features to help keep code understandable, and code collapsing to help you focus on an area of your code. The handy placement and functionality these features add are great assets when you start to work with your first Web pages and HTML code. This chapter gives you a quick-and-dirty tour of Dreamweaver before delving into using this very handy tool.

Before you start working in Dreamweaver, make sure you have a Web host and a URL. Sure, you can work on your site before you do these things, but that's not a great idea because you must be sure you can find and afford hosting that will accommodate your site. A word to the wise: Not all hosts are the same. The URL is important, especially if the name of the site will be the URL. Designing a site only to discover later that someone else already owns the name is a real drag. This scenario has happened more often than you'd think, so don't let it happen to you. Book I has information about choosing hosts and URLs.

Exploring the Dreamweaver Interface

Before you dive right in to Dreamweaver and start making pages, you'll probably want to get acquainted with the interface. The following section gives you the quick rundown of the key features and a little information about how they work. After you get familiar with Dreamweaver's default set of features, you can look for additional features to download at the Adobe Exchange at www.adobe.com/exchange.

The first time you launch Dreamweaver, you see the Welcome Screen (shown in Figure 5-1, a bit later in the chapter). You can select the check box that asks not to show the Welcome Screen when you launch Dreamweaver again. If you do not select that box, you will see the Welcome Screen each time you launch Dreamweaver. The next thing that happens the very first time you launch Dreamweaver is that you're asked to set up a Web site. A dialog box opens, giving you Manage Sites and Cancel options. Dreamweaver is trying to *index* and keep track of the files that will make up your Web site. This is important. Dreamweaver helps you with site maintenance (uploading, synchronizing versions, checking links, and so on) — *if* you set it up properly, that is. More on that later in this chapter.



You can work in Dreamweaver without creating a site; just be aware that Dreamweaver can't help you keep track of the files and their relation to each other. Either way, you need to deal with Dreamweaver's request to set up a site.



◆ **To set up a Web site, click Dreamweaver Site.**

The benefit of setting up a site is that Dreamweaver keeps track of all the files associated with your Web site on your local machine, on your Web server, and on your (optional) test server.

◆ **To skip this step and begin working with Dreamweaver, just open a file.**

Sometimes, you just want to work on one file that's not part of a whole Web site. Simply open that file and proceed. If you choose to work this way, Dreamweaver warns you that you aren't working with files that are part of a single site whenever you try to insert secondary files. This comes in handy when you want to borrow a bit of code from another site you worked on. You may want to open some files from that site to access that code but not include them in the site you're working on at the moment.

The Welcome Screen

When you launch Dreamweaver, you're presented with the Welcome Screen. (See Figure 5-1.) This screen gives you the option to open a file you recently worked on, create a new file, create a file from a sample, take a tour, or follow a tutorial. It also provides information about updates or new products. Adobe sends these updates to your Welcome Screen when you're connected to the Internet.

The Dreamweaver Welcome Screen gives you tons of options for getting started, organized into three lists: Open a Recent Item, Create New, and Top Features. Open a Recent Item and Create New are lists of files you can open or create. Top Features is a list of links to information about the top

features in Dreamweaver CS4. In the Getting Started area, at the bottom of the Welcome Screen, you can find additional resources and information to help you build sites in Dreamweaver. And in the same window, you see some of the menus and panels that we discuss later in this chapter. If you haven't used Dreamweaver before, the Open a Recent Item list has only one item in it: the Open folder button. If you've worked in Dreamweaver and saved some files, you see a list of the ones you used most recently. To open a recent item, just click the filename. Or, you can click Open to see a list of the files in the last site worked on.

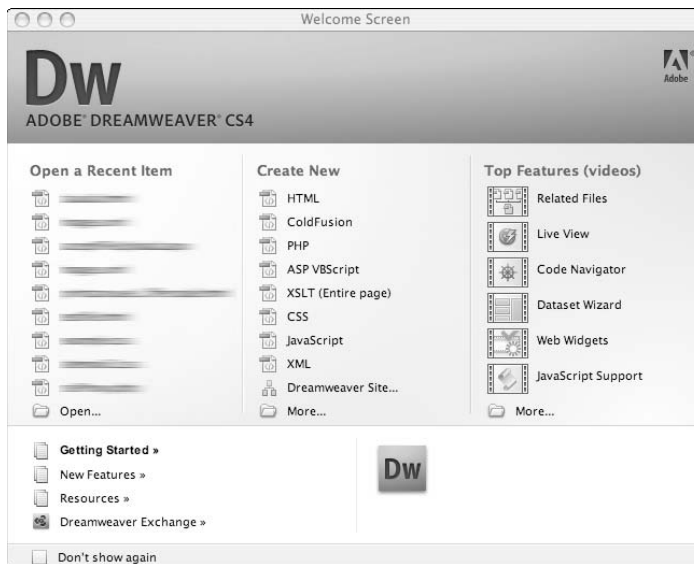


Figure 5-1:
The Dreamweaver Welcome Screen.

In the middle of the Welcome Screen, the Create New list gives you the option to create an HTML, ColdFusion, PHP, or CSS document, among others. To immediately create a file, just click the type of file you need. To see more options, click More at the bottom of the list.

Dreamweaver comes with a large number of sample or starter files that can get you going quickly. To access them, click More in the Create New list and then select Page from Sample to launch the New Document dialog box. (See Figure 5-2.) From that dialog box, you can select the type of file you would like to create, make some optional selections, or choose to start your document from a premade template. Read the next sections for more information about that.

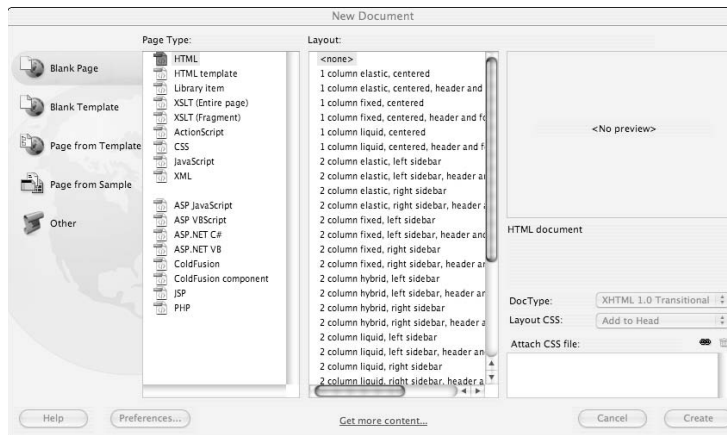


Figure 5-2:
The New Document dialog box.



What's in a name? Local site, remote site, testing servers . . .

The following terms keep popping up throughout the site creation and management process:

- ✓ **Server:** A *server* is just a computer hooked up to the Web but that also has server software running on it. *Server software* receives requests for information from other computers and “serves” the requested information back to the requesting machine.
- ✓ **Local site:** This folder on your own computer is where you keep all your working files for a Web site. It's your unpublished Web site.
- ✓ **Remote site (or live site):** This is the site that's on the Web server. It's the collection of files that have been copied to the Web server. The remote site has your URL (Web address) pointed to it. This site is visible to anyone who has your URL or finds your site via a search engine or a link.

- ✓ **Testing server (optional):** A very good thing to have, the testing server is where you can publish the site so people within your team can see it, but it isn't visible to the whole world. It can be the same machine as your live server, but the files are put into a different folder than the live files — or you can put the files on a totally different machine. The testing server needs to be equipped with the same server software (or server software that supports the same features) as the live environment.

Note: If your testing server and live server are too different, you can end up with content that works on one server but not on the other. That's not a big deal if your Flash video doesn't work correctly on your test server but looks great on the live server. However, this is a very bad thing if it's the other way around. You could end up with content that took a lot of time and effort to produce, but that you can't use. That's no fun!

The New Document dialog box

You can also access the New Document dialog box by choosing File⇨New from the menu. Click the appropriate item from the navigation choices on the left side of the screen: Blank Page, Blank Template, Page from Template, Page from Sample, or Other. Your choice depends on whether you're starting from scratch to build a page or template; using a template you built; needing a little help from prebuilt samples and templates; or creating a CSS, JavaScript, or other type of file. A *new document* creates one document; a *new template* creates a file that you can use to make and maintain multiple files. Dreamweaver has many degrees of assistance to help you get started.

The Page from Template option holds any custom templates you build, listed by the site they're associated with. For more on custom templates, see Book III, Chapter 6. In the following sections, we describe the Blank Page options as well as a few of the Page from Sample options. If you are just starting out, you may want to open a few of the Starter Page Theme pages under the Page from a Sample menu. By starting with one of these prebuilt designs, you can get a look at how the whole page, CSS and graphics, fits together. It's a great learning tool and can get you off to a quick start on building your pages.

The New Document dialog box is divided into a left navigation area and three fields (refer to Figure 5-2), starting with the Page Type field to the right of the navigation area. The Page Type field lists groups of files that you typically want to build to start your HTML, CSS, or other site files. Next to the Page Type field is the Layout Field, which shows more options for each page or file type. On the right side of the dialog box is the Preview pane. After you select an item in each of the two fields to the left, the Preview pane gives you a rough sketch of what your selections translate to on the actual page (**Note:** Selections show only if a preview is available. Some file types don't have a preview.)

Blank Page starter files

Use the Blank Page starter files to create HTML files, HTML templates, Library items (see Book III, Chapter 6 for more), ActionScript files (for use with Flash), ASP, JSP, ColdFusion, PHP, CSS files, and JavaScript files. These starter files include just the basics (tags for DTD, HTML, head, title, and body). For more about these initial tags, see Book III, Chapter 2.

In the case of HTML, Dreamweaver creates a document with an HTML file extension and that includes the document type, HTML tags, head tags, and body tags. It's all ready for you to begin creating your page.

Other options for the Blank Page files start off with just a comment at the top to identify what sort of page it is as well as the correct file extension to get you started. Although these files don't have any code in them, they do help

you remember to save the file with the right extension. It's easy to make a mistake with all those files flying around. Whether to use them is really your preference. As a quick way to get going, you can use a Blank Template, a Page from Sample, or a Page from Template. The Page from Template option refers to templates you build yourself.

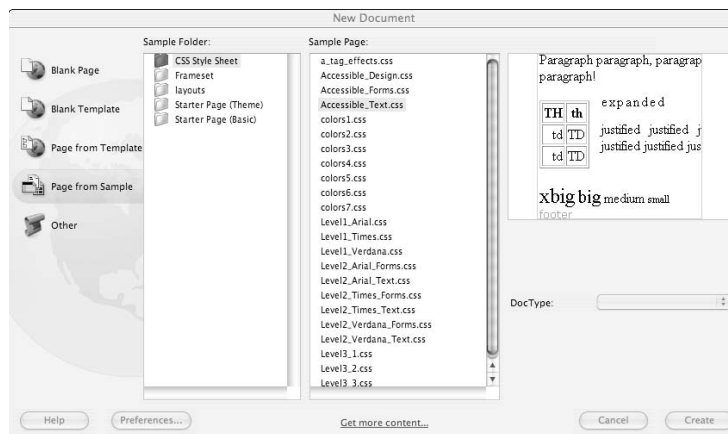
Page from Sample starter files

There are several types of starter files. Each type will give you some help getting started. Some offer just a little help, others are fully developed. Spend some time looking at the Sample Files. Open them and learn from them or use them in your own project. They can be a very useful tool.

CSS Style Sheet starter files

The CSS Style Sheet starter files, found in the Page from Sample area (see Figure 5-3), aren't complete CSS files, but they do include font treatments and basic color schemes to get your design started. Use these files to set the foundation styles, and then you can add your own custom styles to build your design.

Figure 5-3:
Choose the Basic: Arial option and Dreamweaver formats text on the page as Arial. So easy!



Frameset starter files

When you select the Frameset category, Dreamweaver sets up the initial files you need to create a frame-based Web site. (See Figure 5-4, under the Page from Sample section of the New Document dialog box.) Framesets are a little different from regular Web pages, and here's how. A regular HTML page has one Web page document. A *frameset* is one file that actually uses multiple HTML documents to create the whole page. The benefit of building a site this way is that you can have parts of your page stay the same while other pieces

of the page change. For instance, you can put your navigation in the HTML document that loads into the left frame but launch different HTML documents in the right frame.

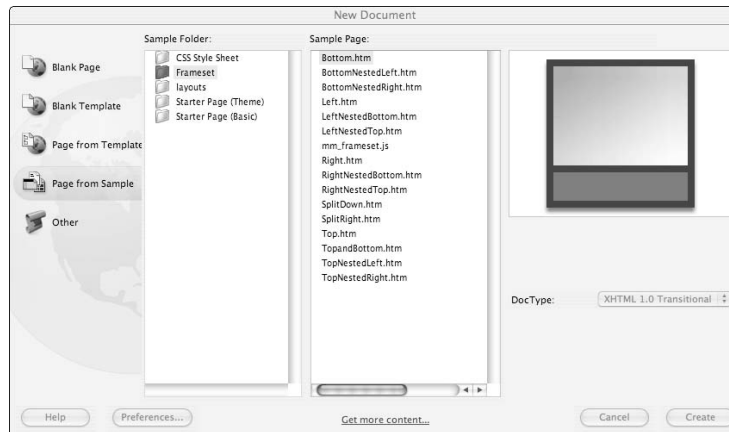


Figure 5-4:
Frameset
layout
choices.



Using frames has a downside, though. Search engines sometimes find only part of the page and launch it without the rest of the page, so visitors can get the main content frame but no navigation or banner area. Another problem with using frames is that they can reduce usability for some users because screen readers (an adaptive device or software that visually impaired visitors use to interact with computers) sometimes have trouble navigating frames. Framed layouts can also be a little more difficult to manage than straight HTML. For instance, linking to information within your site can present problems unless you take special care to direct the browser to load the new page into the right frames. Weigh the pros and cons before deciding to use frames in your design. Many designers don't use frames in their designs because of the associated cons.

Starter Page (Theme)

The Starter Page (Themes) category includes sets of files that work together. HTML and CSS are created when you use these options. With starter pages, you actually get an HTML file and a CSS file already set up to work together, with layouts already started. Some are fairly complete, but others are pretty simple (found under Starter Page [Basic]). The other starter files (discussed in the preceding sections) don't include the layouts. Find out how to build pages and sites with these prebuilt page designs in Book III, Chapter 6.

The Files panel

After you make your selections in the New Document dialog box and click the Create button, the dialog box closes and you see the following:

- ◆ **The Document window:** This is the work area where you design and code Web pages.
- ◆ **The panels on the right:** Choose from different workspace layouts, which you access by choosing Window⇨Workspace Layout. The panels to the right vary, as does the location of the panels, based on what Workspace Layout you choose. You can also customize and save your own Workspace Layout by simply positioning objects where you want them by choosing Window⇨Workspace Layout⇨New Layout and giving it a name.

For the examples in this book, we use the Designer Workspace Layout. By default, this workspace includes the Insert, CSS Styles, AP Elements, Files, and Assets panels to the right of the Document window. This workspace can also include other panels that you choose to open. Look under the Window menu for more panels and tools.

- ◆ **The Property inspector (at the bottom of the page):** This context-sensitive menu panel helps you work with images, text, and other parts of your page or code.

You use the Files panel for site management and organization. The Files panel is the list of all the files in your Dreamweaver site. We specify a *Dreamweaver* site because files will only show up in this window if you have set it up as described earlier in this chapter. You can open site files from the Files panel. You can work with the Files panel in either the Expanded view for site management or file uploading or in the Collapsed view while you work on individual pages. Upcoming Figure 5-6 shows the Collapsed view of the Files panel. Note how the site structure is also displayed.

Expanded view

The Expanded view of the Files panel (shown in Figure 5-5) is divided into two main sections. On the left, you see (initially) an empty window. This will be populated with the list of files on your remote or testing server after you connect (more on that later). On the right is the list of your *local files* (the Web site files saved on your computer but not necessarily published to the Web).

Across the top of the Files panel (Expanded view), you see a menu and a series of buttons. From left to right, they are

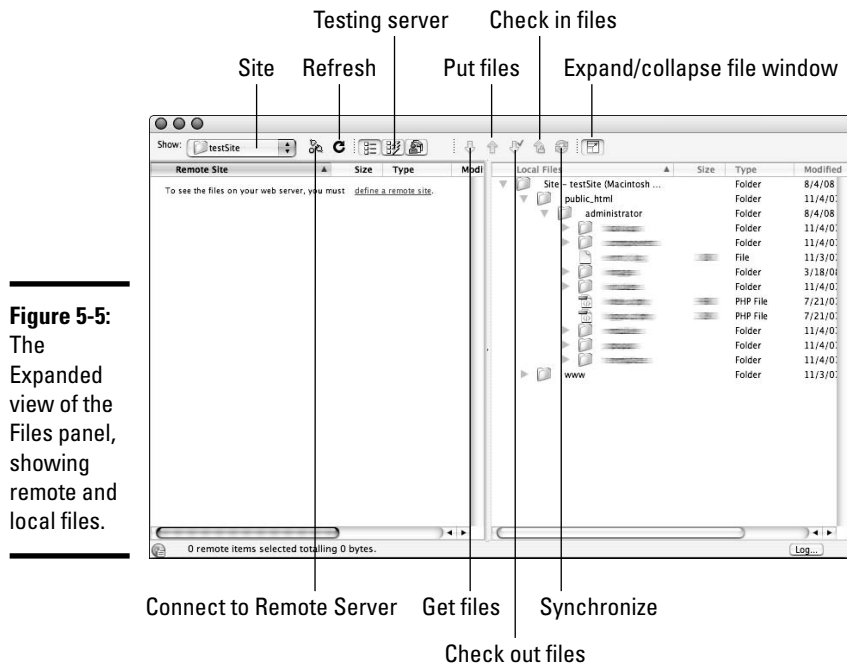


Figure 5-5:
The Expanded view of the Files panel, showing remote and local files.

- ◆ **Site:** This drop-down menu is a list of the different sites you have set up. And, yes, you can set up multiple Web sites in Dreamweaver. Dreamweaver keeps track of each one separately. If you want to work on a different site, use the Site menu to select the site you want to work on. The files for that site will replace the current listing.
- ◆ **Connect to Remote Server/Disconnect from Remote Server:** Click this button to make a connection with your remote Web server. You have to do this before you can see a list of the files on that server and before you upload (or download) anything to it (or from it).
- ◆ **Refresh:** Clicking this button refreshes the file lists, both remote and local. This can take a few minutes if you have a very large site. Sometimes, you'll have better luck right-clicking just one folder within a site and choosing Refresh from the menu that appears. This method can be much faster.
- ◆ **View Site FTP Log:** Clicking this button shows the details of your last FTP transaction with the server. FTP (File Transfer Protocol) refers to a set of rules (a protocol) by which computers and servers can communicate, as long as the server is set up to accept FTP and the person trying to send things to the server via FTP has a valid FTP address, user name,

and password (all provided to you by your host when you sign up for services). FTP transaction simply refers to sending or getting information via FTP. **Note:** HTTP (HyperText Transfer Protocol) is how servers and browsers communicate.

- ◆ **Remote Files, Testing Files, and Repository Files:** Use these three buttons to switch among the remote server, the testing server (if you're using one), and the file repository (used with Contribute/Version Control features).
 - The *remote files* are those that are on your live Web server. These are the files that people see when they visit your site.
 - The *testing server* is an optional secondary server location where you can test new functionality before incorporating it into your live site. Using a testing environment is strongly recommended to avoid having to publish unfinished work to a live site, risking that your visitors will see it.
 - The *file repository* is another optional feature. It stores multiple versions of your Web documents. This enables you to roll back to a previous version if necessary. This can come in handy if you've inadvertently broken a page while editing. Simply roll back to a version that was not broken and retry your edits.
- ◆ **Get and Put:** These two buttons get (down arrow) files from the remote server or put (up arrow) files to the remote server (or testing/repository if you have those selected instead of the remote server). To *get* a file is to use FTP to retrieve a copy of a file from your Web server. To *put* a file is to publish (upload) it to the Web.



Put and Get overwrite existing files on the destination computer, so if you get the home page from the remote (or testing) server, you overwrite the copy on your local machine. When you put (publish) a page from your local machine to the remote (or testing) server, you overwrite the copy that is there. If you're trying to overwrite a newer copy with an older copy, Dreamweaver warns you that you're about to do that and gives you the option to cancel. This is one of the benefits of allowing Dreamweaver to manage your site because it keeps track of and compares the dates and times on your files.

Both the Put and Get commands create the identical file structure, if it doesn't already exist. For instance, you add a new folder to your site on your local machine. When you publish the folder and new content, you don't need to create a place for it to go because Dreamweaver creates the new folder in exactly the right place on your remote server. The same thing goes when you download files from a site. The site structure is duplicated exactly. This ensures that your links work correctly and things are where you expect them to be. Pretty cool, huh?



- ◆ **Check Out and Check In:** Use these two buttons to manage workflow for a team. Members of the team *check out* files when they're going to work on them. That way, anyone else on the team who tries to open and work on the file at the same time is alerted that someone is already working on it. This prevents members of a team from overwriting each other's work. When done working on the file, the team member simply has to *check in* the file so that others can work on it.

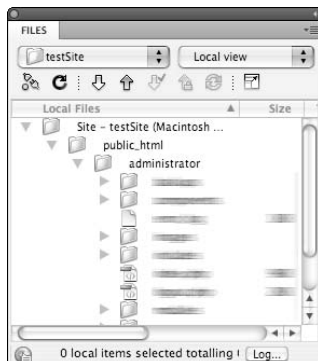
Using this built-in tool — or some other type of *source control* (also known as *version control*) — is very important so that multiple people simultaneously working on a site avoid overwriting each other's work.

- ◆ **Synchronize:** This function helps with site maintenance. Synchronize compares the files on your local machine with those on the remote server and lets you know which files are out of synch. You then have the option of putting (or getting) files to and from the server to synchronize your local and remote sites. This can be very helpful when you update several pages. Use Synchronize to find and upload the newly updated files. This ensures you won't accidentally forget a file when it's time to upload.
- ◆ **Expand/Collapse:** This button toggles between Expand the Panel and Collapse the Panel. It allows you to see both remote and local files when needed and then collapse the panel while you are working. This helps keep the on-screen clutter to a minimum.

Collapsed view

The same functionality that's available in Expanded view is also available in Collapsed view. (See Figure 5-6.) This view can be helpful if you just need to make a quick change to one file and upload it. The drop-down menu on the left is the Site Management menu; use it to select the site you want to work on. The menu on the right is the Site View menu; use it to choose the Local, Remote, Testing, or Site Map view. The same Connect/Disconnect, Refresh, Get, Put, Check Out, Check In, and Expand/Collapse buttons (explained in the preceding bulleted list) are available, too.

Figure 5-6:
The Collapsed view of the Files panel is handy for seeing and opening site files.



The buttons for Site Files, Testing Server, Site Map, and Synchronize are omitted from the Collapsed view to conserve space. These functions and several others are available via a drop-down menu in the upper-right corner of the panel or via the pop-up menu available through the button at the extreme upper-right corner of the panel. Look for an icon that looks like a tiny bulleted list. These show up in many panels, so be on the lookout.

Other options available through the Files panel include file and site management tools. In the far, upper-right corner is a button that looks like a tiny bulleted list. Click it, and a pop-up menu opens with options to work with files and folders in your site, group files for better organization, and change the view you're using in the Files panel. One of the most important options is the Site menu, which enables you to do the following:

- ◆ **Create or edit site settings.** Find out more about this in the “Creating a Web Site with Dreamweaver” section, later in this chapter.
- ◆ **Set up a site to work with Adobe Contribute.** *Contribute* is content management/editing software. See Book VIII, Chapter 2 for information about using Contribute to maintain a Web site.
- ◆ **Turn on or off cloaking.** *Cloaking* allows you to set some local files to not upload, which is a handy feature for working with files such as raw Flash files and Photoshop documents.
- ◆ **Run site reports.** Running reports enables you to check links and do other tests to make sure your site is working well.
- ◆ **Synchronize files.** You can ask Dreamweaver to compare the files on your local machine with those on the remote server (or the other way around) and give you a list of files that have been updated more recently. You can then choose to upload all files, or you can individually select files to be skipped and upload just the ones you'd like to publish. This is helpful when you make several edits to a site, and you want to make sure you publish all the new files to the server. It can be easy to forget which files you need to upload; Dreamweaver can help you with that.

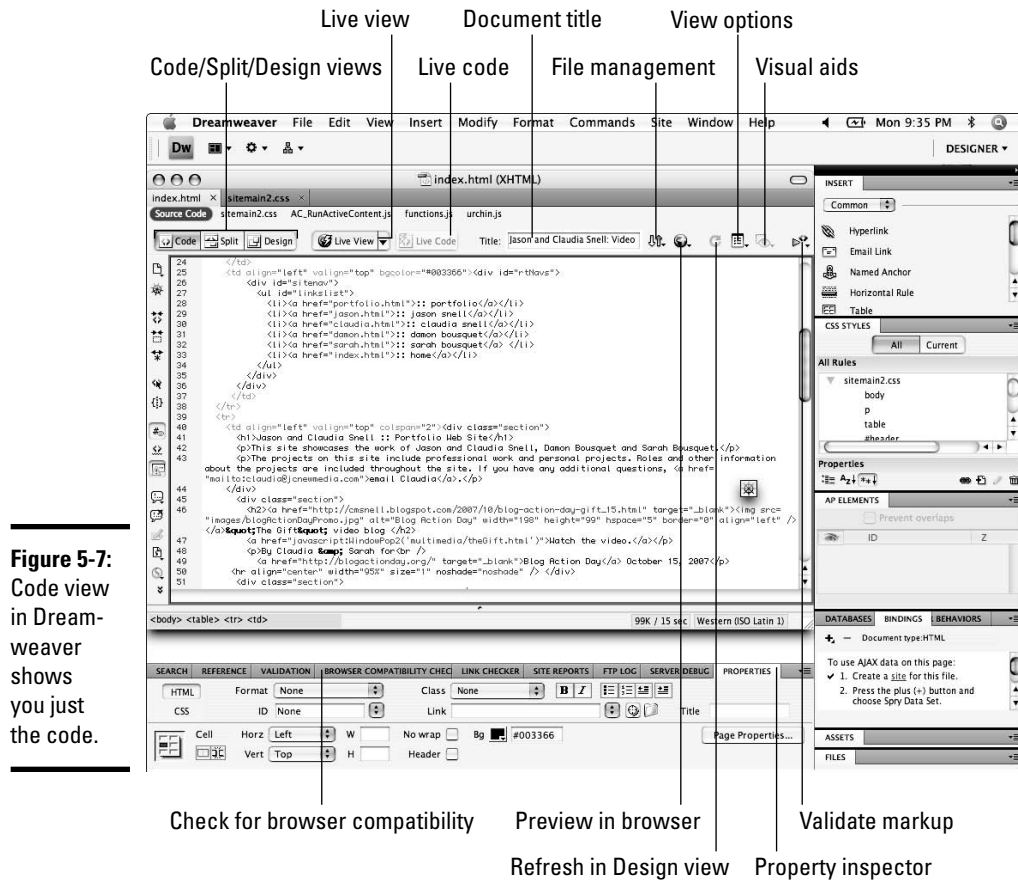
The Document window

The Document window is where the file you're working on is displayed. There are different ways you can work with this window. The “best” way depends on what you're doing and what you're most comfortable with. In this section, we show you the three views you can choose (Code, Design, or Split) and describe some of the Document window's features.

Code view

Code view (shown in Figure 5-7) is just a window with the HTML in it. No graphics or colors are displayed: just the underlying code of your page. This view is good if you're comfortable with hand-coding your pages or if you

need to use as much space on your screen as possible to look at a larger block of code. The downside is that you can't see the end results of what you're coding without previewing in a browser or switching to a view that includes Design view.



Design view

Design view is the opposite of the Code view. In Design view, you see only the visual aspects of your page (as shown in Figure 5-8). This can be helpful if you're not as comfortable with code or if you need to see more of your design for a moment. The downside to this mode is that you can't see what's happening with your code. Dreamweaver is a great tool but is no substitute for a person. Sometimes, edits that you make in the Design view can produce issues in the code, such as *empty tags* (tags that are left behind after the content in them has been removed). Check over your code with a validator

(covered later in this chapter) before you go live with the page. A *code validator* is a test that you can run on a page's code to make sure it's written properly. The test results alert you to any issues so that you can fix them.



Figure 5-8:
Design
view shows
the visual
appearance
of the page.

Split view

Split view mode is really the best of both worlds. You get Code view at the top of the screen and Design view at the bottom. In Split view, you can hand-code *or* make edits in the design window. It allows you to see the effects of your edits right away, whether you make them in the Code portion of the view or in the Design portion of the view. When it's necessary to update the Design portion of the view, a Refresh button appears in the Property inspector. (We cover the Property inspector later in this chapter.)

Additional code-viewing helpers

At the top of the Document window, near the Code View, Split View, and Design View buttons, are some additional buttons: Live View and Live Code. You'll also find Source Code and the names of all the linked script and CSS files that you used in the document.

Live View gives you a preview of what your page will look like in a browser, directly in Dreamweaver. You can use this quick and handy tool when you'd rather just get a look at the page without using the Browser Preview button.

Live View also lets you turn off JavaScript or plug-ins to see what your page looks like when visitors without those functions load your page.

Live Code highlights all the Live View source code. The highlighting is visible in Split and Code view modes.

Source Code shows a list of the linked CSS and script files. If you'd like to review the code in any of those files, click the name of the file in that list. It will be displayed in your workspace. This saves you from needing to have several files open while working. To return to the Web page code, click Source Code.

Document title

At the top of the Document window is the Title field. It's important to give each document a proper title. Remember that the page title shows in browsers and in search engines, so make sure you give your page a title that makes sense. You can add a title by typing `<title>My Title Here</title>` in the head of the document. See Book III, Chapter 2 for more information about HTML. Dreamweaver automatically adds the title tags in the correct place if you create your HTML documents via the New Document dialog box. Simply replace the default `Untitled Document` with the title of your choice. (And now, you know why so many pages seem to be titled `Untitled Document` out there.)

Check and validate code

The Document window toolbar offers code-validating and -checking functionalities. Code *validators* and *checkers* are quick tests that you can run to make sure your page is going to work well for your visitors. Clicking the Check Page button enables you to deal with checking for browser errors and support; clicking the Validate Markup button allows you to check the code for errors and see where the errors are.



Using checkers and validators is a good habit to develop. You can set up these tests to look for different things, such as problems with different types of browsers or different versions of HTML and CSS. Decide what versions of browsers or code you will support, based on your site's purpose and the target audience. Try to find out as much as possible about the types of technologies that work best for your audience, and then develop your site to work with those technologies.

File Management: Put and Get

The Put/Get buttons on the Document toolbar look the same as the ones in the Files panel. In the Document window, clicking the Put button uploads the current document, and clicking the Get button retrieves the live version of it for you. In all cases, Put/Get asks you whether you'd like to upload (or download, respectively) the dependent files. This means, "Would you like all the

graphics, CSS, JavaScript, and multimedia files that are linked to that, too?” It doesn’t mean that Dreamweaver will download other HTML documents, though — just the files that are needed to create the finished page you’re posting or getting.

Preview in Browser

Preview in Browser is really the only way to see what your page will look like. To see the preview, click the Preview in Browser button. (Refer to Figure 5-7.) Although Design view shows a very close rendition, different browsers display things a little differently. You need to preview your pages in the actual browsers you’re designing for. We recommend viewing your pages on both Mac and Windows computers and in all the major browsers. Also, make sure you test all interactive and dynamic elements to make sure they work. Be especially careful with JavaScript and CSS because browsers can vary greatly in how they use them.

Refresh in Design view

After you make changes in Code view of the Document window, click the Refresh in Design View button (refer to Figure 5-7) to see the changes take effect.

View Options

Some additional helpers are available to help you as you work. These include things like the following:

- ◆ **Word wrap:** Without this, the code and content can stretch out for miles. Word wrap keeps it all in the visible window without affecting the code.
- ◆ **Code line numbers:** Displays numbers in the left margin of the Code view window.
- ◆ **Highlight invalid code:** This one highlights errors in the code with bright yellow, which is very handy!

Other code-formatting options and many other tools are available to make life easier. Explore them to find which ones work best for you.

Visual Aids

Visual aids allow you to see various parts of your page more easily. Some of the options available are CSS layout outlines, table borders and widths, image maps (graphics that have areas that are clickable links), and invisible elements like scripts. These elements are all invisible when the page is viewed in a browser, but it can be very useful to see where they are while you are working on a page. You can turn on or off the different options to help get a better idea of what is going on “behind the scenes” of your page.

The Preferences dialog box

You can easily customize how Dreamweaver works for you. The Preferences dialog box (shown in Figure 5-9) is available by choosing Edit⇨Preferences (Windows) or Dreamweaver⇨Preferences (Mac). You can enable or disable many settings in this dialog box; the best way to become familiar is to explore them. We highlight a few of the features in the text that follows.

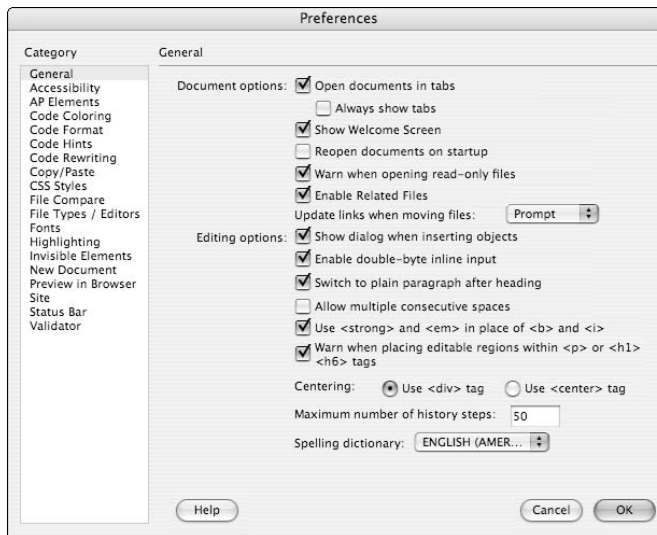


Figure 5-9:
Customize
Dream-
weaver
with the
Preferences
dialog box.

When you select the General category, you see (among other options) the check box that you use to turn the Welcome Screen on and off. You also see options to adjust settings for how Dreamweaver handles spell checking and updating links. In this dialog box, you also have the option of setting Dreamweaver to warn you when you're opening *read-only files* (files that are locked and can't be edited, just viewed).

Other categories in the Preferences dialog box enable you to customize how you handle accessibility features, what colors are used when displaying code (setting up different colors for different types of code helps you read your code), and how your code is formatted, for easier reading.

Also, when you select the Code Hints category, you get the option of setting up Dreamweaver to give you hints when you begin typing tags. With this feature enabled, Dreamweaver attempts to finish tags while you type them to remind you of proper syntax or parameters available for that tag. Also, this feature shows a list of styles available if you have CSS set up for the page.

The Property inspector (panel)

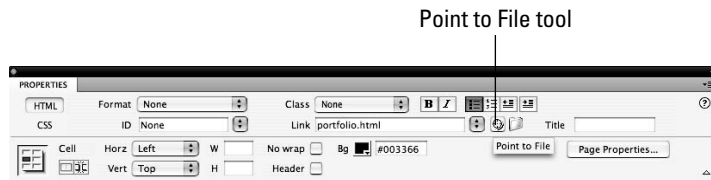
The Property inspector shows you the options available for the page element you're working on. Plus, the Property inspector is a *context-sensitive panel*, which means that its functionality changes depending on what you select in the Document window. If you select a table, the options for editing a table are displayed. When text is selected, the text formatting options are shown. You can edit elements using the Property inspector; or you can set alignments, change colors, join table cells, or apply CSS styles. Many other options are also available, depending on what type of element you select.

Making links with the Property inspector



One of the interesting tools in the Property inspector is the Point to File tool (shown in Figure 5-10), which looks like a clock or a small crosshair, located to the right of the Link text field. To use the tool, click and drag the clock to the file in the Files panel to make a link. A line drags from the box to the file you point to. This is a quick and easy way to make a link, saving you time when linking to files within your site.

Figure 5-10:
Making links
is a snap.



To link to files that aren't part of your site, type the full URL in the Link text field. You can also choose to target a window when you make a new link. The default is for pages to load into the same window as the page you're already viewing, replacing the current page with the new page to be viewed. For some links, you don't want the new page to replace the one the visitor is viewing. For example, when you link to another site, you probably want that to open in a new window so that the visitor can easily return to your site. In that case, you want to choose `_blank` from the Target drop-down list to the right of the link box. This opens the new page in another window. When the user is done viewing the linked page, she closes that window, and the original page is still there in its window. Use this technique sparingly because it can be annoying to visitors if your site launches a lot of windows.

Setting page properties

Click the Page Properties button on the Property inspector panel to adjust settings regarding default font and size, page color, and margin size. You can also customize the color of a link — including its *rollover color* (the color the link changes to when you hover your cursor over the link). Making settings on this panel automatically generates CSS in the head portion of your document.

If you're building multiple pages that will share the same look and feel, you should cut the CSS from the page head and paste it into an external CSS file. See Book III, Chapters 2 and 3 for more information about how to use CSS and HTML together.

The CSS Styles panel

The CSS Styles panel is where you can write and edit CSS styles. You can also write CSS directly into the HTML document or into the CSS document. Book III, Chapters 2 and 3 show you how to use CSS and HTML together. The CSS Styles panel (see Figure 5-11) gives a list of styles associated with the page.

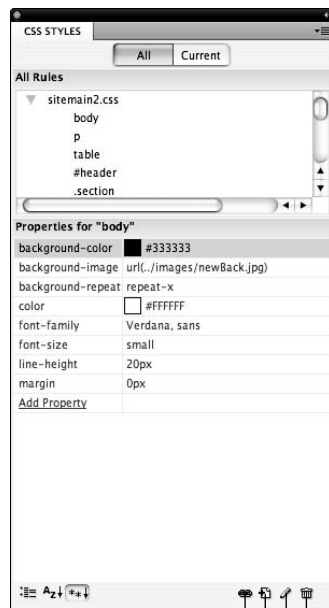


Figure 5-11:
The CSS
Styles panel
makes
editing CSS
easier.

Attach style sheet
New CSS rule
Delete CSS rule
Edit style sheet

Here are ways you can manipulate styles with this panel:

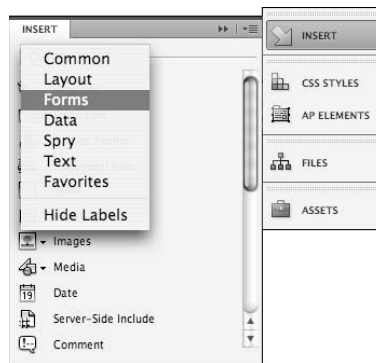
- ◆ Click the All tab to see all the styles, or click the Current tab to see the style associated with an element on your page.
- ◆ Double-click a style to edit it.
- ◆ Add new styles by clicking the New CSS Rule button.
- ◆ Attach a style by clicking the Attach Style Sheet button. A browse window opens so you can locate the CSS file you'd like to attach.

The CSS Properties, which appear in the middle section of the CSS Styles panel, allow you to edit the properties of a particular style. When you select a style in the CSS Styles panel, the panel is populated with the properties available. (See the middle portion of Figure 5-11.) Click in the area to the right of a property to see a menu of options for that property. You can also type a value into the field.

The Insert panel

The Insert panel (shown in Figure 5-12) is another helpful tool located in the series of panels in the default workspace. This panel has several categories of tools for elements that you might want to quickly and easily add to a page. Some items in the default set include the following:

Figure 5-12:
Use the
Insert panel
to insert
elements
onto your
pages.



- ◆ **Common:** These elements include the basics: link, e-mail link, *anchor points* (those invisible links that make it so people can jump from place to place on your page), table, image, date (which has several options for formatting), and comment (for adding a code comment).
- ◆ **Forms:** Inserts the elements you need to build Web forms. (See Book VI for details about how to make a form.) The elements found under this toolbar are radio buttons, check boxes, and text boxes. Each of these has their own special attributes that you can set via the Property inspector after you've added them to the page.
- ◆ **Text:** Another place with options for formatting text. Special types of formatting are available here, such as headings and definition lists.

All the elements available on the Insert bar are available via the Insert menu. You can also add more categories by going to the Dreamweaver Developer Center at www.adobe.com/devnet/dreamweaver and finding extensions.

Creating a Web Site with Dreamweaver

Now it's time to set up your first Web site. Although you can work with Dreamweaver without setting up a site, it's really a better idea to set one up. Dreamweaver creates a log of information about your site and the files included in it, called a *site cache*. This enables Dreamweaver to do many things. It can make sure that paths to linked files are correct; synchronize your content on your live server; and perform other, helpful, site management tasks. With that being said, it's time to set up your site.



Before you start working in Dreamweaver, we recommend having a Web host and a URL. Book I has information about choosing hosts and URLs.

- 1. Create a folder on your computer and name it after your project.**

This will be your local site; all site files will go in this folder. (Dreamweaver uses the term *local root folder* to refer to this.)

- 2. Create a folder inside your project folder and call it Production.**

This folder is for all the working files you create while working on your site. Don't post this to the server when you publish. Use the cloaking feature to keep it local. (Refer to the earlier "Collapsed view" section for instructions on how to set this up in the Files panel.)

- 3. Create a folder inside the Production folder and call it Images.**

This is where all your site images go. (Dreamweaver refers to this as your *Default Images Folder*.)

- 4. Create folders for the major areas of your Web site.**

For the About Us area of your Web site, create an About folder; for the Products area of your Web site, create a Products folder; and so on. Make sure you follow Web-legal naming conventions.

- 5. Create a folder for your included files.**

These files include CSS, JavaScript, and any other special files that you want to keep organized.

- 6. Launch Dreamweaver.**

- 7. In the Create New menu of the Welcome Screen, select Dreamweaver Site.**

The Site Definition dialog box appears.

- 8. In the Site Definition dialog box, click Advanced at the top of the screen.**

The Advanced view shows options for Local Info, as shown in Figure 5-13.

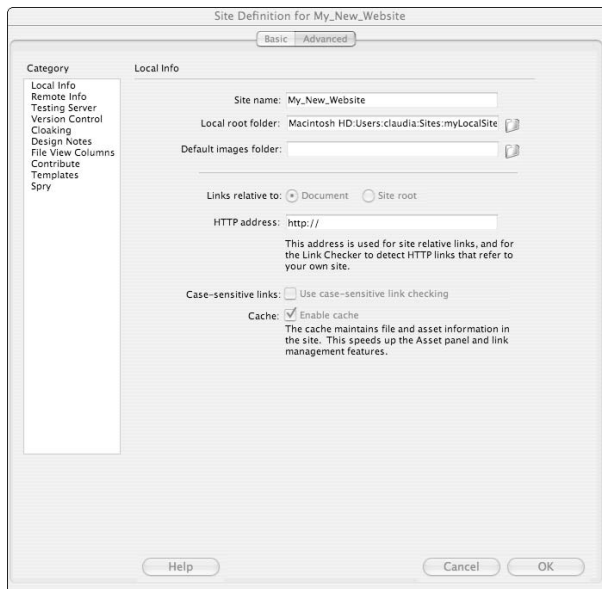


Figure 5-13:
The Site
Definition
Dialog box.

9. Enter a name for your site in the Site Name field.

This name appears in the site list (not on your Web site), so make it something that you'll easily recognize.

10. Select the local root folder.

Use the folder icon to browse to the location on your computer where you'd like to keep your local site folder. You can even create a new folder from the browsing dialog box if you need to.

11. (Optional) Back in the Site Definition dialog box, select a default images folder.

This is a folder within the local site folder where you will place the site images.

12. Select either the Document or the Site Root radio button to make your links relative to one or the other.

Links Relative To refers to how Dreamweaver will build links when you use the automatic linking tools. Links that are relative to the site root use the main site folder as the starting point for building the link code. Comparatively, links relative to the document use the document you're linking from as the starting point for building the link.

13. (Optional) Make your link-checker case-sensitive by selecting the Case-Sensitive Links check box.

Be aware that if you choose to select the “case-sensitive” option that the link-checker could miss broken links due to differences in capitalization. You may find your results to be more accurate if you do not select that option.

14. Set Dreamweaver to maintain a cache of your site files.

This is important to select for the site maintenance and synchronization tools.

15. Click Remote Info (under Category on the left) to move to the next step — connecting to your remote host.

Connecting to the host is important. If you do not, you will not be able to publish your site to your live server.

16. Tell Dreamweaver how to connect to your server by choosing an option from the Access drop-down menu (see Figure 5-14).

Set access to Local/Network or FTP (or one of the others if your server requires it). For the purposes of this chapter, we focus on the FTP settings because FTP is the most common access.

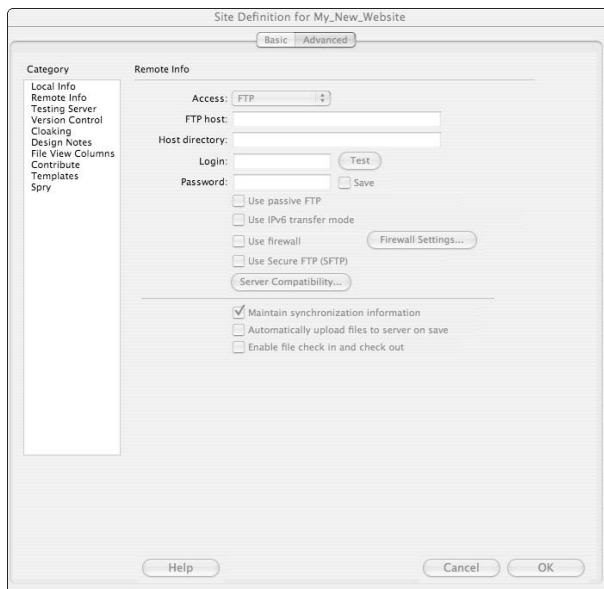


Figure 5-14:
The Remote Info dialog box.

The information you need to input into the fields at this point has to come to you from your Web host, usually via e-mail. When you sign up for hosting services, you will be sent information about how to access your account. Some hosts will give you a user name and password to log onto their site and get the information. Other hosts will send you

all the information in an e-mail. In either case, you will be looking for “FTP information.” That information is not the same as your customer login information. Be sure to look for the specific “FTP” information.

Also, be aware that buying a URL is not the same as buying hosting services. You can buy these services from different vendors or from the same vendor. It is often good to find a hosting company that also provides URLs. That way you can get everything you need from one spot. Make sure you buy BOTH a URL (your Web address) and hosting (the Internet server where your site will be located) or your site will not be available on the Internet.

When you have what you need, you can finish filling out the remaining fields in the Remote Info screen.

- **Remote Folder** (for Local/Network connections): If you are using a local/network server, you will need to enter the path to that folder in the dialog box that becomes visible when you select “Local/Network” from the drop-down menu.
- **FTP Host** (visible only if you select FTP as your access): This is the location of their server. It will look like a regular URL except that it starts with `ftp` instead of `http`. (As we mention earlier, FTP stands for File Transfer Protocol, which is how files are published to Web servers. Using the FTP tells the server what sort of communication is coming.)

Note: Your host might be a local/network server in cases of intranets or when your Internet server is available through your local network.

Make sure you choose a host that has FTP access. Otherwise, uploading and maintaining site files can be a time-consuming nightmare.

- **Host Directory** (optional): Many hosts don’t provide a directory name for you to input.
- **Login and Password:** This is the user name and password as given to you by the Web host. You get all the necessary information to log in and FTP files when you sign up for Web hosting.
- The next three check boxes are optional. Your Web host tells you whether you need to use them. If you’re having trouble connecting and you’re sure all the information you input is correct, ask your host whether you should be using any of these options, too.

17. Click Test to make sure you can connect.

That’s it. You’re ready to build a site in the local folder and then publish it to the Internet.

Other options are also available through the Manage Sites dialog box. You can set up a test server and make settings on the types of files you wish to cloak, among other things.



Testing Your Pages and Validating the Code

When you validate your code and check it for browser compatibility or accessibility, you're making sure that it's coded so that it complies with the version of HTML or XHTML you're using, that it works in the browsers you want to target, and that your site is accessible to as many users as possible. (See Book III, Chapter 9 for more about testing usability and accessibility.)

Because of the different versions of HTML, XHTML, and browsers, make sure you decide what works best for your audience before you start coding. You can check the World Wide Web Consortium (W3C) Web site (www.w3.org) for more information about the versions of HTML, XHTML, and other Web programming and scripting languages.

Use any of these methods to use the built-in Dreamweaver browser checkers and code validators:

- ◆ **The Site and Commands menus:** Click these menus and select the test you want to run.
- ◆ **The Validation panel:** This panel (shown in Figure 5-15 with test results) automatically launches when you run tests using other methods, too. The arrow activates the menus of options within each tab. This is where the results of the tests and validators are displayed, along with additional information where appropriate. Double-click a line item, and Dreamweaver takes you to the error in the code; this helps you quickly locate and fix issues.

Also, the Validation panel can give the code line numbers for each error that it finds. Just turn on the code line numbers in the Code View Options by choosing **View** ⇨ **Code View Options** ⇨ **Line Numbers**. The line numbers are displayed in the margin of Code view.

Figure 5-15:
The
Validation
panel.



The Validation panel also has options to find and replace things within your code or text, which is handy if you find that you misspelled someone's name. You can also check the links on your page or within your site. Other site reports are also available; experiment with them to see which ones work best for you.

- ◆ **The Validate Markup button on the Document toolbar:** Refer to Figure 5-7.

To check your code's validity, follow these steps:

1. **Click the Validate Markup button on the Document toolbar.**
2. **From the pop-up menu, choose Current Document, Current Local Site (to do the whole site), or Selected Files in Site (to check a few files at once).**

The Validation panel displays the results of the test along with information about what the error is, how severe it is, and where you can find it in your code. Double-click an item, and Dreamweaver takes you directly to the error.

To check your page for browser compatibility, follow these steps:

1. **Click the Check Page button on the Document toolbar.**
This causes Dreamweaver to display your results in the Browser Compatibility Check panel.
2. **If you haven't set up Dreamweaver to check for the browsers and versions you want to test for, select Settings; if you already have, skip to Step 6.**
3. **Choose the browsers and versions you'd like to test for in the dialog box and then click OK.**



Check browsers for both Windows and Macintosh computers. Important browsers at the time of this writing are Internet Explorer, Firefox, Safari, and Netscape. Look for versions that are current and past versions that are two years old or less. We recommend that you test on many browsers and test often. That way, you'll catch problems as they occur instead of finding that your site has issues after you developed it to an advanced point.

4. **Click the Check Browser Support button.**
5. **Choose Check Browser Support from the drop-down menu.**
6. **Review and handle any errors according to the information supplied by Dreamweaver.**

Additional tools to help you write your code are also available under the Commands menu:

- ◆ **Code Formatting:** This tool straightens out your code and makes it easier to read. It doesn't affect your site's performance or the validity of your code.
- ◆ **Clean up Word HTML:** These tools fix errors in HTML code generated by Microsoft Word. These code clean-up tools look for common messy things that happen to code, such as empty tags. They clean up some of these things and alert you to possible issues they find but can't clean up. They're handy tools but shouldn't be used instead of validators and checkers.

Publishing Your Web Site with Dreamweaver

You're ready for this section when you have a completely designed site; you've bought your hosting and URL and made sure your URL is pointed to your space; and you've developed your content, designed your pages, ran usability tests, and validated your code. All systems are go.

In other parts of the book, we take you through all the steps leading up to this point. When you're ready to upload your files to the server, here's how you do it:

1. Open the Files panel.

Make sure that you turned on cloaking for any files or folders you don't want to publish. (Working files, albeit important, just clutter your host space and take up room, and you pay for that space.) Take a look at the earlier section, "Collapsed view," to find out how to turn on cloaking.

2. Select the folder you would like to cloak.

3. Click the Options button in the upper-right corner of the Files panel.

It looks like a tiny bulleted list.

4. Choose Site⇨Cloaking⇨Cloak to cloak the folder you selected. Or you can select from the same menu Settings to target specific file types.

5. Click the Connect button on the Files panel's toolbar to connect to your remote server.

6. Click the Put button, and then click Yes when Dreamweaver asks whether you'd like to Put the entire site.

If you already posted your site, you can use the Synchronize feature to post only new or updated content.

7. Select the folder you want to synchronize (or select the main folder to synchronize the whole site).

8. Select the folder(s) or files that you'd like Dreamweaver to consider while checking files to synchronize.

9. Click the Option menu button in the upper-right corner of the Files panel and then choose Site⇨Synchronize.

10. Select a Direction from the Direction drop-down menu.

That is, put new files to the remote server or get new files from the server. The drop-down menu has options for putting (from local to live) or getting (from live to local).

Synchronize works both ways. Make sure you choose the correct direction, or you could overwrite your newer files with older ones.



Your options are

- *Put Newer to Remote:* This uploads files from your computer to your remote Web server.
- *Get Newer from Remote:* This gets files that are newer and copies them to your local site.
- *Put and Get:* This choice compares all the files and either puts or gets the newer files, depending on which copy is newest.

You can also elect to delete any files that exist on the remote site but not on your local site. If you're very careful to make sure you won't be accidentally deleting files that your site needs, it can be a great way to clean up site files on the remote server. If you are not careful, you can end up with broken links, images that don't appear correctly, or scripts that don't work anymore. Proceed with caution when using this feature.

11. Click the Preview button.

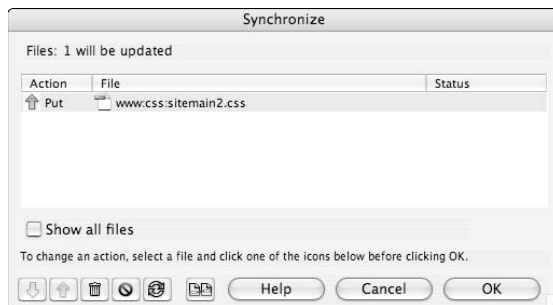
Dreamweaver dubs this Preview to let you know that it's not actually posting or getting anything yet — just compiling a list for you to see.

Dreamweaver compares the local site with the remote site and presents you with a list of files that need to be updated. You can accept the list as is, or you can select files in the list and remove them from the upload cue by clicking the Ignore File button (red circle with the line through it). Figure 5-16 shows the Synchronize dialog box with a list of files to be posted to the Web upon approval.

12. When you're ready to post, click OK.

That's it!

Figure 5-16:
The
Synchronize
dialog box.



Go to your URL and check to make sure all the files you wanted to post made it to your live site. If you don't see your changes, try refreshing your browser and clearing your browser's cache. If that doesn't work, you'll have to explore the pages for another issue, such as links going to older versions of pages.

Chapter 6: Using Dreamweaver: Advanced Techniques

In This Chapter

- ✔ Using prebuilt page designs in Dreamweaver
- ✔ Creating your own templates in Dreamweaver
- ✔ Managing your assets and library items
- ✔ Implementing behaviors on your pages

Dreamweaver CS4 has several built-in tools and features to ease your job as a site builder. One especially useful feature is the Dreamweaver collection of *prebuilt pages* — pre-existing designs into which you place your content — that can help you get up and running very fast. You can also build special pages called *templates* — which are essentially the same as prebuilt pages, but you create these according to your specific needs — to make site building and maintenance easier for you and/or your team. The great thing about using prebuilt pages and templates is that you can reuse them time and time again instead of having to reinvent the wheel every time you start a new site project.

And if you have content that will appear in multiple places on your site — such as footers, headers, and navigation aids — Dreamweaver also provides features that make handling this situation and updating a site easier. You can build little modules of code to include in your pages.

In this chapter, we look at two other handy Dreamweaver tools:

- ◆ **Library items:** This Dreamweaver-specific feature comprises pieces of code that you can quickly add to a page and reuse countless times.
- ◆ **The Assets panel:** Here, you can see all the elements of a site.

We also cover *behaviors*, which are small bits of JavaScript code that you use to add functionality to your pages, such as opening a new browser window from a link or creating an image rollover effect. (An *image rollover* is when an image changes to a different image when a user moves his cursor over it.)

Creating a Site with Prebuilt Page Designs

The Dreamweaver prebuilt page designs can be good tools to use to get a project up and running very fast. Also, because you don't have to start entirely from scratch, using the prebuilt designs can be a great way to figure out how to create and code pages; you can change a few things in the HTML and CSS code to see the effects.

Some prebuilt designs are complete designs; others are just starters, like the CSS files that supply you with a basic HTML page and a CSS document that covers the basics of a color scheme and font choices but leaves the layout code to you. Each category has benefits; the decision of which to use depends on your needs and how comfortable you are with creating pages on your own. The important thing to know is that Dreamweaver can help you if needed, or it can stay out of your way. The choice is completely yours.

To build a site from a Dreamweaver prebuilt page design, just follow these steps:

1. Create a folder on your hard drive and name it after your project. Put it somewhere you can easily find it.

This is a preference issue, so feel free to create a system that works for you.

2. Inside the first folder, create three more folders.

Name the folders Images, Includes, and ProductionFiles, or create your own naming conventions. The point is to make it easy to find these files even if you come back to the project weeks or months later. Plan now, stay tidy. (That's right: Don't use spaces in folder names, and always use Web-legal naming conventions; see Book II, Chapter 2 for more information about Web-legal naming.)



In addition to the folders you create to hold your supporting files, you'll need to consider the actual content of your site and build folders to hold those pages. If your site has subsections that will have multiple pages in each, create a folder for each section. This helps you keep your site easy to maintain. Only very small sites (ten or fewer pages) are manageable with all the files in one folder together. Start with a structure that will support expansion. That is, if you expect to have additional pages within your subsections later, start with folders for each subsection up front. Figure 6-1 shows the file structure for a basic site.

3. Launch Dreamweaver.

4. Create a new site:

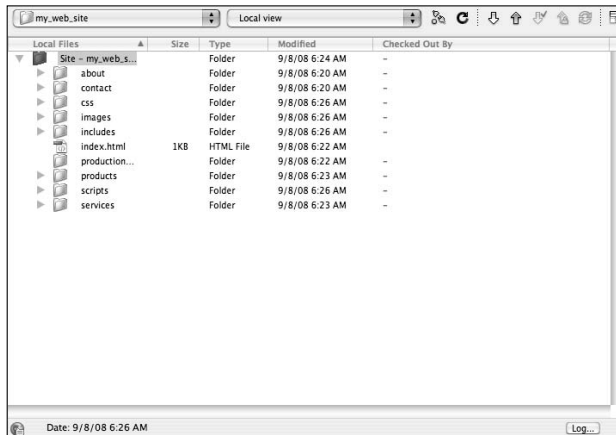


Figure 6-1:
The file structure of a basic Web page.

If Dreamweaver prompts you with a Site Management dialog box with a Manage Sites choice, click the New button.

From the Manage Sites dialog box, select New.

If Dreamweaver doesn't prompt you with a Site Management dialog box, open the Manage Sites dialog box by choosing Site⇒New Site.

- a. In the New Site dialog box, set the local site folder to the folder you create in Step 1.
- b. Set the default images folder to the folder inside your local site folder called Images, which you create in Step 2.
- c. If you have your Web site hosting set up, put the remote site information in the appropriate fields. (See Book III, Chapter 5 for information about setting up a site in Dreamweaver.)

5. **Create your first page by choosing File⇒New, clicking the Starter Pages category, and then choosing a design in the field to the right of the Category field.**

For purposes of exploring the interface, choose Starter Pages⇒Entertainment-Home Page.

The Preview pane displays a rough idea of what your design will look like.

6. **In the lower-right section of the dialog box, select a document type definition (DTD) for your pages. For now, choose XHTML 1.0 Transitional.**

See Book III, Chapter 2 for more information about DTDs.

7. **Click Create to create the page.**

- 8. When Dreamweaver prompts you to Save As, browse to the local site folder and save the file as `index.html`.**

This is your site's home page.

Dreamweaver displays a Copy Dependent Files dialog box stating that additional files need to be copied to your site. *Dependent files* are the graphics, CSS, and any other files that will be needed for your page to display and work properly.

- 9. Browse to the local site folder and click the Copy button in the Copy Dependent Files dialog box.**

At this point, you have an HTML document open in the Document window, a CSS file that you can open and edit as you need to, and a few supporting sample graphics supplied by Dreamweaver (banner, placeholders, and others, which vary depending on the design you choose).

- 10. Customize the HTML document with your content:**

- *Replace the placeholder text with your own.*
- *Check image sizes so you can create your own graphics.*

- 11. Open the CSS document and determine what (if anything) you'd like to change to create your own design.**

Figure 6-2 shows how you can use a predesigned page as the base for your own design.

- 12. Proceed with creating other pages on your site.**

You can select other layouts from the same set you have chosen for your home page or just use the Save As feature to rename the file you just created to make the other pages on your site.

What is a DTD?

Although the code that makes up Web pages is generically referred to as HTML, there are actually several different versions of HTML as well as *XHTML*. While you're building pages and learning how to code, try to use the most current versions. Information about the versions of HTML can be found on the World Wide Web Consortium Web site at <http://www.w3.org>.

A DTD (document type definition) in an HTML or XHTML document should be the first line

of code in your page. It tells the browser what version of HTML or XHTML your page is coded with. This helps browsers know what your page is made of and assists with proper display of your pages. For more information about DTDs, you can visit the section of the World Wide Web Consortium Web site that deals specifically with Web document DTDs at www.w3.org/QA/2002/04/valid-dtd-list.html.

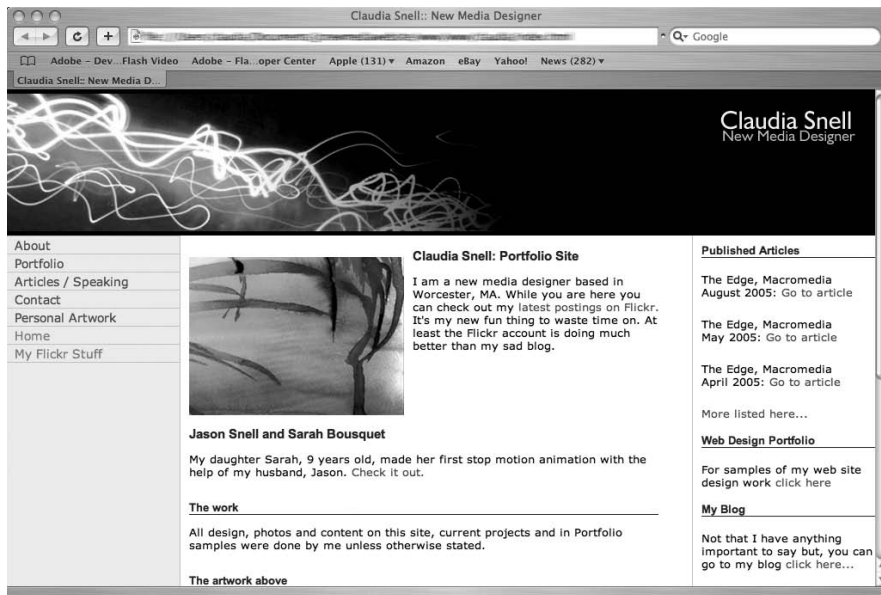


Figure 6-2:
An example
of a page
that started
as a
predesigned
page.

Why call your home page “index” instead of “home”?

You can actually call your home page file `index` or `default` but nothing else. Web servers look in the folder for a file called `index.html` or `default.html` (or `index.asp`, `index.php`, `index.cfm`, or `index.jsp`, if you’re building a dynamic Web site). These are the universal home page names. If you don’t give your home page one of these names, people will have to type the whole path to your home page to see it. For example, if you call your home page `homepage.html`, a visitor has to type <http://www.yoursite.com/homepage.html> to see your site. If he doesn’t, he gets an error message saying the site can’t be found or that a page can’t be displayed. If you name your home page properly, a visitor needs to type only <http://www.yoursite.com> to get to your

site. The same rules apply to subfolders. If you have a section of your site inside a folder called `Plants`, name the main page `index.html` within that folder so that visitors only need to type <http://www.yoursite.com/plants> to see the page.

Another thing to know is that `.htm` and `.html` are not the same thing to all servers. Make sure you check with your host to see what its system is set up to recognize. If your host’s server considers only `default.htm` to be the default page of a site (or home page) and you use `index.html`, the server won’t recognize the home page. If you experience difficulty getting a site to show up after you upload it, this is one thing to check. Also, make sure you’re using the right name and file extension for your host’s setup.

Creating Templates to Ease Site Maintenance

In Dreamweaver, you can create your own templates from HTML documents you create from using the Dreamweaver prebuilt pages found under Starter Pages. These templates can have sections of the pages locked (uneditable) to protect content that's consistent on each page. You can set up templates that have several different layouts to accommodate different types of content.



Locked regions of these templates are editable only within the original template document itself. You create individual pages with the template to start and then set the boundaries for the locked regions. These templates also work in conjunction with *Adobe Contribute*, which is a Web page–editing program that allows nontechnical content providers (with permission) to update and maintain pages.

Making a template

If you don't already have a site set up, create the folders as we describe in the previous section. Also, adjust your site settings as described in Book III, Chapter 5. Then, follow these steps to create a template:

1. Create a new page:

- a. Choose *File*→*New* from the Dreamweaver menu bar.
- b. On the *General* tab of the *New Document* dialog box that appears, click *Basic Page* in the *Category* field.
- c. In the field to the right of *Category*, click *HTML*.

Take a look at the *Preview* pane to get a rough idea of what the design will look like.

- d. Click the arrows on the *Document Type (DTD)* drop-down menu and choose the appropriate *DTD*.

For information on what a *DTD* is, see the nearby sidebar.

- e. Click the *Create* button.

2. Save the document to your site folder.

Design your page as you would for a normal page. See Chapters 1–5 of this minibook for information about creating pages. Or you can start your design with a prebuilt design, which we discuss in the preceding section.

3. To make the page into a template, choose *Insert*→*Template Objects*→*Make Template from the Dreamweaver* menu bar. An alternative way to do this is to choose *File*→*Save as Template*.

Like with so many things in Dreamweaver, the method to use is a matter of personal preference. The good thing about having multiple ways to get a job done is that you can develop a working style that is comfortable for you. The way that is easiest for you to remember is the way you should do it.



Make sure your template stays in the same folder with your site. If you move the template, your template won't work properly. Notice that Dreamweaver gives templates a special file extension (.dwt) and also creates a special folder (Templates) within your site where it stores the template files for the site.

Including editable regions

Editable regions allow you to make changes to designated areas of a particular page. Changes to noneditable areas have to be made in the template itself and are reflected in *all* the documents that have been or will be created using the template. Follow these steps to include editable regions in your template:

1. **Highlight the area that you'd like to make editable — a table cell or a layer, for example — and then choose Insert>Template Objects>Editable Region.**
2. **Name the editable region in the dialog box.**

These areas are the only areas that are directly editable within the pages created using the template. Figure 6-3 shows a page with editable areas included. These editable areas are where each page's unique content will go. Content that stays the same on each page should not be made editable. This helps you maintain consistency in areas where consistency is vital — such as navigation and site-branding areas.

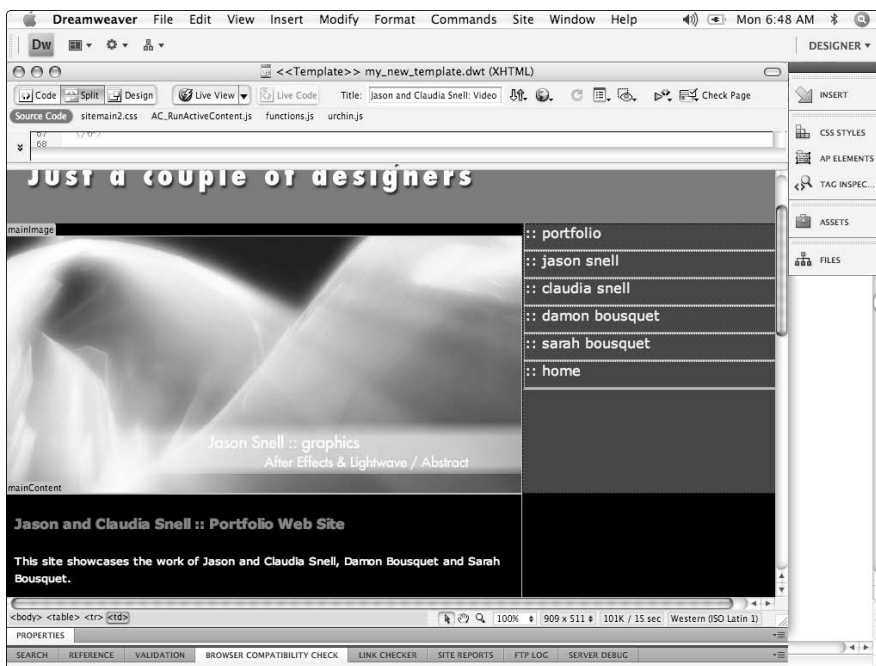


Figure 6-3: A Dreamweaver template page with editable regions.

Creating a page from a template

After you create your own template for site building, creating a page with it is easy; just follow these steps:

1. **Choose File↪New.**
2. **On the Page from Template tab of the New Document dialog box that opens, select the name of your site from the Templates For list.**

Select the template you'd like to use.

3. **Enable the Update When Template Changes check box.**

This allows you to make changes to just the template, and Dreamweaver changes all the pages that were created using the template. The benefit of this functionality is that you can build pages that allow other users to change the main content of a page but protect the banner and navigation areas from unauthorized edits. If edits need to be made, make them in the template, and the rest of the pages are updated, too.

4. **Click Create.**

Dreamweaver templates also work with Contribute, an easy to use Web page editing tool that helps nontechnical content editors work on Web pages without accidentally editing parts of the page they shouldn't. So, you can build a site in Dreamweaver but supply Contribute to individuals who need to make regular changes to text or pictures. Combining these programs helps you protect your site from unauthorized changes or unintentional mistakes that break the pages while giving you the ability to hand off routine maintenance to nontechnical members of a team. Figure 6-4 shows what areas of a page would need to be editable on the average page. Notice how much of the page can have a consistent look protected by noneditable regions.

Updating a template

Updating a template is nearly as easy as creating and using one; just follow these steps:

1. **Open the template file you'd like to change. Opening a template is just like opening any other file in your site. Use File↪Open or double-click it in the Files panel.**
2. **Make the admin-level edits to the noneditable areas you need to update.**
3. **Save the document.**

As shown in Figure 6-5, the Update Template Files dialog box opens, showing all the files you have created based on the template.

Figure 6-4: Most of this page is protected from edits, which helps maintain consistency while sharing the workload of general updates.

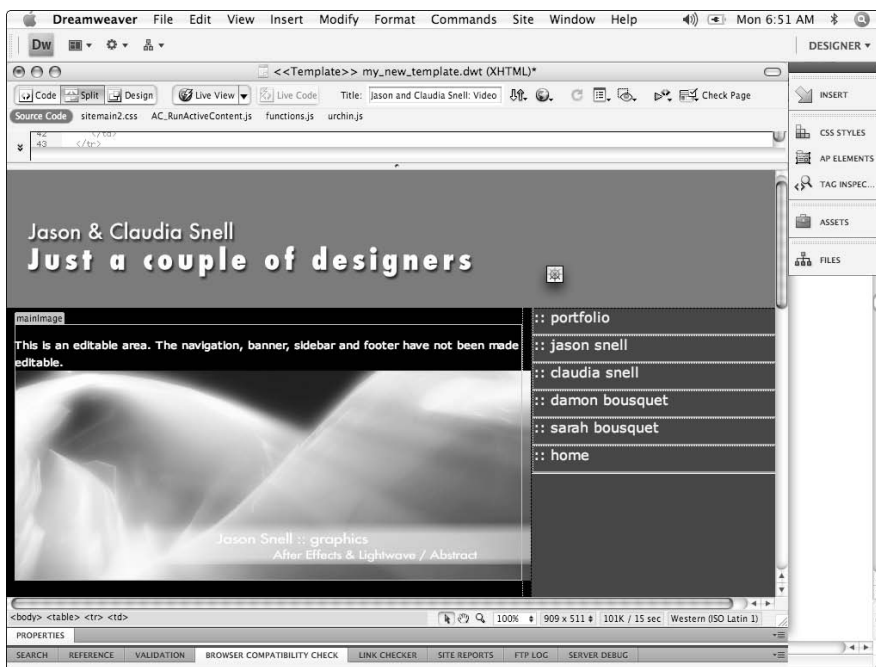
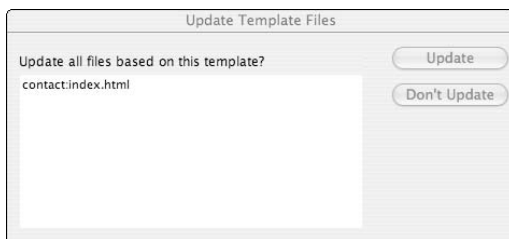


Figure 6-5: Choosing files to update.



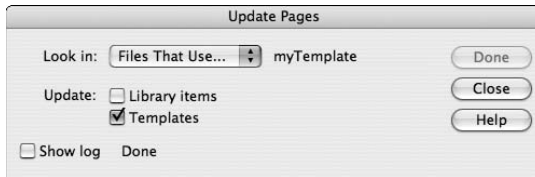
4. Click Update to update the pages with the new changes or click Don't Update Pages to leave the pages as they are.

If you choose to update, Dreamweaver automatically makes the changes as soon as you click the Update button. You will then see a dialog box that enables you to scan for and update additional files that were generated by the template. If you choose to search another area of your site for files to update, the Done button (shown in Figure 6-6) changes to a Start button. When you are ready to update the files, click Start.

If you choose not to update additional files, click Close. The dialog box will close and the site files will not be updated.

5. Click Close when the updates are complete.

Figure 6-6:
The Update
Pages
dialog box.



If you need to check your site for template-based files and update them, choose **Modify** → **Templates** → **Update Pages**. The Update Pages dialog box will open allowing you to choose the site folder(s) and template you would like to scan and (if needed) update.

Exploring the Assets Panel and Library Items

Two other handy tools in Dreamweaver are the Assets panel and library items. Even though they're grouped together and they both help you keep things organized, they aren't exactly the same thing.

- ◆ **Assets panel:** This panel provides a quick and easy way to see all the things that are available within your site. It provides easy access to elements, such as images, templates, multimedia elements, and color schemes.
- ◆ **Library items:** These are pieces of code that you make into modules that you can quickly add to a page.

Unlike *code snippets*, which are also bits of code that you can use on your pages over and over, you build library items, and they're associated with a particular site. They work a bit like minitemplates. When you add them to a page, they aren't editable unless you detach them from the original library item. However, if you make changes to the library item itself, a similar Update Files dialog box appears.

Touring the Assets panel

If the Assets panel isn't open when you launch Dreamweaver, open it by choosing **Window** → **Assets**. The Assets panel groups on-screen with the Files panel when the Files panel is in Collapsed view.

The Assets panel has several categories, represented by buttons in the left margin. (See Figure 6-7.) The categories (from top to bottom) are

- ◆ **Images:** Lists all the JPEGs, GIFs, and PNGs within the site.
- ◆ **Colors:** Shows color chips and the hexadecimal code for the colors used on the site.

- ◆ **URLs:** Lists the pages and URLs your site is linked to. You can quickly make links in your documents by highlighting the text or item you'd like to be the link, and then dragging the desired link onto the highlighted element in your Document window.
- ◆ **Flash:** Lists all files that are different from the Shockwave files. If you want to use images or Flash pieces, click the Flash button on the vertical toolbar and drag the item from the list to the place in the Design view window you'd like to place it.
- ◆ **Shockwave:** Lists multimedia created using Adobe Director. Flash files are sometimes referred to as *Shockwave*, but they aren't really Shockwave; they're Flash.

Shockwave is still in heavy use in building games (popcap games, for instance) and other intense applications. Flash and Shockwave are not the same thing. It is a common mistake to think they are. It can lead to some trouble if you are expecting Flash to be Director and Director to be Flash. At this level it is important just to know that Shockwave and Flash are not the same thing despite the fact that it's called "shockwave flash."

- ◆ **Movies:** Lists all movie files that don't fit on the Flash or Shockwave tabs, such as QuickTime or Windows Media.
- ◆ **Scripts:** Lists all the external scripts, such as JavaScript files, that are on your site.
- ◆ **Templates:** Lists the Dreamweaver templates you built.
- ◆ **Library:** Lists the library items you created for the site.

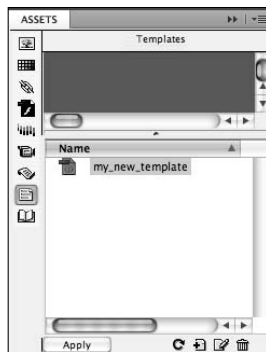


Figure 6-7:
The Assets panel showing the Templates category.

You can access the assets of each type by clicking the appropriate button. This method of locating assets is helpful when you know you're looking for a particular item, and you'd like a list of just that type of object. For instance, if you want to place a Flash element on a page, click the SWF button. You're presented with a list of the Flash elements available on that site.

At the bottom of the Assets panel, you'll notice a few more buttons (refer to Figure 6-7): Refresh Site List, Add to Favorites, Edit, and Delete (looks like a trash can). At the top are a few more: Sort by Whole Site and Sort by Favorites. If you marked elements as Favorites, you can use the second option to narrow your search for assets while you work.

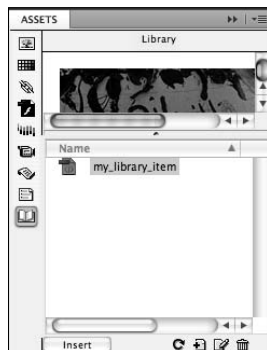
Discovering library items

Webmasters have several methods at their disposal for implementing similar types of functionality on a Web site; using the library items feature is the Dreamweaver-specific method. The Assets panel is usually open within the Files panel and can be accessed by clicking the Assets tab. If the Assets panel isn't open, you access library items through the Assets panel. (See Figure 6-8.) Choose Window⇨Assets from the Dreamweaver menu bar to open the Assets panel. After you create an HTML document, you can make parts of it into library items. With library items, you can reuse that content on other pages by dragging it from the Library into the spot on the document you'd like for it to be. By default, there aren't any library items: You have to make them yourself.

- ◆ **Create part of a page once and then quickly apply the code from it to other pages on your site.** You might find library items useful to implement and maintain consistency in footers, banners, and navigational elements.
- ◆ **Update all instances of a library item at once.** Library items maintain an association to the library item file, so if you update the original, the other instances of it are updated, too.

Library items are sort of like minitemplate pieces with a few differences. Like with templates, you can create as many library items as you like. Just keep in mind that the more variations you have, the more difficult maintenance can become.

Figure 6-8:
Viewing
library
items in
the Assets
panel.





With Dreamweaver library items, the code of the item does appear in the individual HTML document but is linked in Dreamweaver to the originating file. Other HTML editors won't recognize the association, so if you implement this method, you need to maintain and edit the files within your Dreamweaver site.

Creating a library item

To create a Dreamweaver library item, follow these steps:

- 1. Create a Web page or open an existing one.**
- 2. Create the element you'd like to make into a library item.**
Create a navigation bar or site banner, for instance.
- 3. Open the Assets panel by choosing Windows⇨Assets from the Dreamweaver menu bar. Click the Library button on the Assets panel.**
- 4. Highlight all the code that makes up the element.**
- 5. Click the New Library Item button located in the lower-right corner of the Assets panel.**

A dialog box warns you that style sheet information that applies to the code is not copied along with the code.

- 6. Click OK.**
- 7. Name the library item.**

Use a name that makes sense, such as `Navigation` for a navigation bar. Make sure you call it something Web legal. Making up strange naming schemes is not helpful and can actually confuse things later when you or other members of the team need to work with the files.

Dreamweaver automatically creates the library item folder called `Library` in your site. The new library item is in the folder with the name you gave it and the file extension of `.lbi`.

At this point, you might notice some new code in your document that looks like this:

```
<!-- #BeginLibraryItem "/Library/Navigation.lbi" -->
<div class="section">
<h2><a href="http://cmsnell.blogspot.com/2007/10/blog-
  action-day-gift_15.html" target="_blank"></a>&quot;The Gift&quot; video blog </h2><p><a
  href="multimedia/theGift.html">Watch the video.</a>
</p><p>By Claudia & Sarah for <a href="http://
  blogactionday.org/" target="_blank">Blog Action Day</a>
  October 15, 2007</p>
</div>
<!-- #EndLibraryItem -->
```

The first line in this block of code alerts Dreamweaver that this piece of code is linked to the library item called `Navigation`. The last line tells Dreamweaver that it has reached the end of the block. Do not remove this code, or Dreamweaver won't be able to update this properly.

The block of code is also highlighted in Code view. This is just a visual cue to help you see that you have a library item on your page. It won't affect the visual display of your page at all.

Using a library item

Inserting library items into a site is very easy and takes only a couple of steps:

- 1. Create or open an HTML document that you'd like to use the library item in.**
- 2. Drag the library item from the Library panel into the Document window into the location you'd like to place the block of code.**

Your page has a block of code that resembles the code generated when you created the library item.

Updating library items

To update the library item, follow these steps:

- 1. Select the library item in the Assets panel list.**
- 2. Click the Edit button.**
It's in the lower-right corner of the Assets panel.
- 3. Dreamweaver opens the document in the Document window.**
- 4. Make the desired edits.**
- 5. Save the document.**

The Update Library Items dialog box opens, displaying a list of documents that will be affected.

- 6. Click Update.**

The Update Pages dialog box appears, giving you additional options and a status on the update process.

- 7. Make any additional optional choices and then click Start in the Update Pages dialog box.**
- 8. Click Done to close the dialog box.**

Detaching library items

In some cases, you might want to detach an instance of the library item from the main file. For instance, you could decide to expand an area of a site that has a library item–generated navigation bar. You can detach that navigation bar from the library file; make changes; and then make a new, secondary navigation library item for use within that section.

To detach an instance of the library item from the parent file, follow these steps:



1. **Open the page that has the block of code.**
2. **Select the library item code.**

You'll be able to find it easily; library item code is highlighted by Dreamweaver.

3. **In the Property inspector, click Detach from Original.**

An alert box opens, warning you that you won't be able to affect this section of code by editing the library item after detaching.

4. **Click OK to detach the item, or click Cancel if you change your mind.**

Note: You can also choose to edit a library item by selecting it and clicking Open in the Property inspector. To select the library item, just click anywhere in the library item code or click it in the Design view window.

An additional function you can perform by selecting an instance of a library item is Recreate. If you click the Recreate button in the Property inspector, Dreamweaver creates a new copy of the library item. If the LBI file (a Dreamweaver library item) already exists, a dialog box opens, warning you that you're about to overwrite the original.

Adding Functionality with Dreamweaver Behaviors

Dreamweaver *behaviors* are pieces of prebuilt, client-side JavaScript. *Client-side* refers to scripts that are executed by the browser on the user's machine; comparatively, *server-side* scripts are executed by the server.

Behaviors allow Web designers to add functionality quickly and easily. Dreamweaver comes with several behaviors included. You can further customize Dreamweaver by downloading and adding more behaviors.

Two popular behaviors are

- ◆ **Open Browser Window:** A link opened via JavaScript allows for some customization and can also receive other commands via JavaScript. We give you instructions on how to use this behavior later in this section. When this code is added to a link, the linked file opens in a new browser window. (See Figure 6-9.)

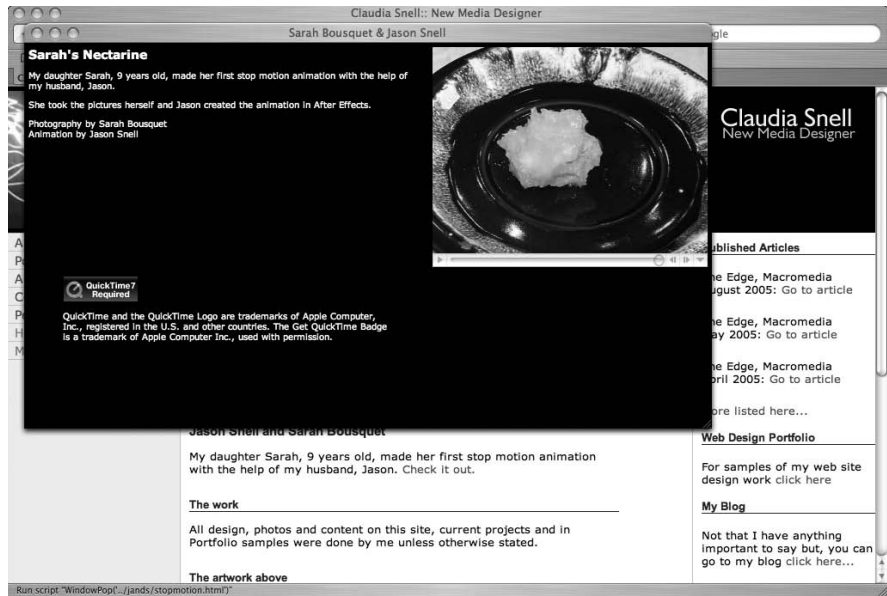


Figure 6-9: Opening a browser window via a JavaScript behavior.

This is a little different from simply opening a link in a new window via HTML, which is shown in Figure 6-10:

```
<a href="http://www.sample.com target="_blank">LINKED TEXT</a>
```

- ◆ **Pop-up message:** A *pop-up message* is a dialog box that pops up and gives your users instructions or other information. (See Figure 6-11.) Pop-up messages (or *alerts*, as some people call them) can be very helpful and useful, but be careful to use them only when they're the best solution. Users tend to get annoyed by pop-ups and alerts, so you should avoid them unless they're conveying important information. Information about how to implement and work with pop-up messages appears later in this section.

Other functionality available through the Behaviors panel includes functions for DHTML (dynamic HTML) and rollover effects.

Figure 6-10: Using target="_blank" to open a second browser window.



Figure 6-11: A pop-up window, also called an alert.



One thing to keep in mind when you use behaviors is that browsers support JavaScript in different ways. Some functionality that works fine in one browser doesn't work in other versions — or can work differently in other browsers. Before you start to use behaviors, decide what browsers and versions you want to support. You can then set up the Behaviors panel to show only the behaviors that work for your targeted browser(s). In the same menu that appears when you click the Add Behavior button (the plus sign at the top of the panel) is the Show Events For option. If you choose that menu item, a list of available browsers is displayed. Scroll to the one you'd like to target and click it. Behaviors become available or unavailable immediately, depending on your selection.

For more behaviors, click Get More Behaviors (which is also available through the Add Behavior menu). When you choose that menu item, Dreamweaver launches a browser and browses to the Dreamweaver

Exchange Web site (as long as you're connected to the Internet, that is). If not, it asks you to connect. You can then select, download, and load new behaviors into Dreamweaver.

Including the Open Browser Window behavior

To include the Open Browser Window behavior in your code, follow these steps:

1. Highlight the item you'd like to use as a trigger.

The trigger should be an image or bit of text; you need to have a link on the page so the user can make the window open.

2. If the Behaviors panel isn't open, you can access it by choosing Window ⇨ Behaviors.

3. In the Behaviors panel, click the Add Behavior button (the plus sign at the top).

4. From the pop-up menu, choose Open Browser Window.

5. Make changes in the Open Browser Window dialog box that appears. (See Figure 6-12.)

- a. *Type the URL you would like to open in the URL to Display text box, or click the Browse button and browse to the file within your site that you'd like to open.*
- b. *Set the window width and height to customize the size. Just enter the numbers; the unit of measure is pixels. So, if you'd like your window to be 500 pixels wide and 400 pixels high, enter **500** in the width box and **400** in the height box.*

You can set the size to any size you'd like.

- c. *Enable the check boxes of the attributes you'd like to select.*

Attributes further affect the features and appearance of the new browser window, such as whether scroll bars appear in the new window, whether the user can resize the window, and so on.

- d. *Type a name in the Window Name text box.*

Give the new window a name that can be referred to by other JavaScripts you might want to implement.

A benefit of using a window name is that you can prevent multiple new windows from opening by using the same name in other instances of the open browser window script within your site. If a visitor clicks an instance, a new window opens. When he clicks another instance that targets the same window, the new content replaces the old content.

There's another benefit of using this method of opening another browser window: You can control that window via JavaScript that calls that new window by the name you give it. A downside is that a few users don't allow JavaScript to run in their browsers. It is possible that they could miss out on your content.

Figure 6-12:
The Open
Browser
Window
dialog box.



6. Click OK.

Dreamweaver inserts code into the head section of your Web page that makes the behavior work. Dreamweaver also inserts some code at the location that you highlighted in Step 1. This code is there to trigger the behavior that is placed in the head.



If you want to reuse the code on multiple pages or if you'd like to keep your pages clean, you can cut and paste the JavaScript from the head of the document into an external JavaScript file by following these steps:

1. **Choose File⇨New from the Dreamweaver menu bar.**
2. **In the New Document dialog box, choose Category⇨Blank Page⇨JavaScript.**
3. **Click Create.**
4. **Save the new file to your local Web site folder.**

If you don't have a folder called Scripts in the site folder, create one:

- a. *Choose File⇨Save As from the Dreamweaver menu bar.*
- b. *Browse to the folder that holds your local Web site.*
- c. *Click New Folder.*
- d. *Name the folder Scripts.*
- e. *Save the new JavaScript file with a Web-legal name. (See Book II, Chapter 2 for information about Web-legal names.)*

5. Look for code that looks like the following in the head section of your HTML document:

```
<script language="JavaScript" type="text/JavaScript">
<!--
function MM_openBrWindow(theURL,winName,features) { //
    v2.0
    window.open(theURL,winName,features);
}
//-->
</script>
```

6. Cut all the code that starts with `function` and ends with the second curly brace, like this:

```
function MM_openBrWindow(theURL,winName,features) { //
    v2.0
    window.open(theURL,winName,features);
}
```



The curly brace is very important. If you don't have that curly brace, the code will be broken!

7. Paste this code into the new JavaScript file.

The remaining code looks like this:

```
<script language="JavaScript" type="text/JavaScript">
<!--
//-->
</script>
```

8. Remove the comments and add a link to the JavaScript file:

```
<script src="scripts/myscript.js" language="JavaScript"
type="text/JavaScript"></script>
```



You can add this code to any of the pages of your site, and it will make the open browser window behavior available on that page, too.

The second piece of code that makes this work is in the body section of your HTML document:

```
<a href="onclick="MM_openBrWindow('http://www.YOURLINK.com',
'windowName','menubar=yes,width=440,height=444')">Text
that is linked</a>
```

This piece of code creates the link text that triggers the open browser window. It also sends instructions to the code called *parameters*, which tell the code how large the window should be, what URL should be displayed, and whether there should be scroll bars — all the things you set in the Open Browser Window dialog box. You can make changes to this code, if you like,

by typing directly into this code. The JavaScript gets the new instructions and follows them. No need to open the JavaScript file at all. It's been built using variables that await the instructions that are passed by the preceding code.

About extensions and the Extension Manager

Extensions are bits of code and other plug-ins created by Adobe or third parties and made available for download through the Adobe Exchange Web sites. You can download and install these extensions to add functionality to Dreamweaver, Flash, Fireworks, and/or other Adobe software. Some extensions are available for a fee, and others are free. Make sure you read the license agreements and user reviews before you download files. They can help you choose the most helpful tool for the task at hand.

To install behaviors or other items downloaded from the Adobe Exchange Web sites:

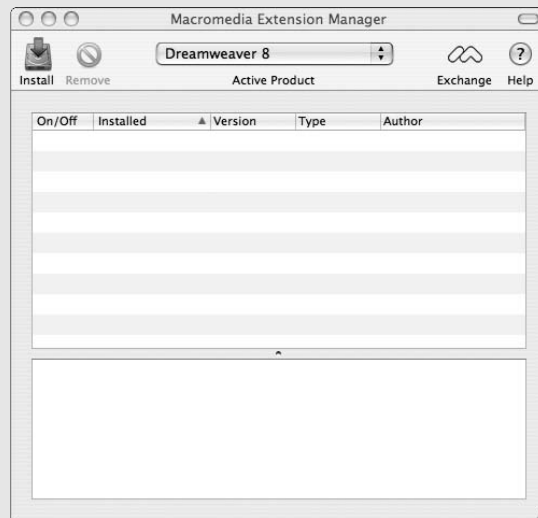
1. Double-click the MXP file.

.mxp is the file type of Adobe extension files.

Extension Manager launches and handles the installation. Extension Manager also has additional information about the file you installed, sometimes including copyright information or tips for use. You can also *manage* your extensions — enable and disable them and keep track of which extensions you have installed — for your Adobe products. The image that follows shows the Extension Manager, which is actually a separate program that manages any extensions you collect for all of your Adobe products.

2. Restart Dreamweaver for the new extension to be available.

You must restart. The same goes for all the Adobe products.



Editing a behavior via the Behaviors panel

If you're not comfortable messing with the code (as we show you in the preceding section), you can use the Behaviors panel to alter behaviors. Just follow these steps:

- 1. Select the trigger code in the body of the HTML document.**
- 2. In the Behaviors panel, double-click the Edit Behavior button.**
It looks like a little yellow gear, next to the name of the behavior.
- 3. In the Attributes dialog box that opens, make your edits.**
- 4. Click OK to accept the changes.**

Another thing you can edit is the type of interaction that triggers your behavior.

You will notice that the behavior has some code that looks like `onClick` or `onLoad`. These are actions that trigger the code. In other words, `onClick` means that when someone clicks the spot you designated, the behavior executes. And `onLoad` means that the script will execute as soon as the page or other trigger element loads into the visitor's browser. You can change the trigger action by selecting a new one from the drop-down menu next to the edit gear in the Behaviors panel.

Creating a pop-up message

Another behavior that might come in handy is the pop-up message (also known as an *alert box*). If you've browsed the Web, you've seen alert boxes before. They're the little windows that pop up to give you some extra information. These windows require some acknowledgement from users that they have seen them: In other words, users have to click OK before the alert lets them proceed. Pop-ups are most often used to tell users that they need to include some extra information in a form and other things of that sort.



Although many users don't like pop-ups, they can be a valuable tool when you need to convey information. Just make sure you don't overuse them because they can lose their effect. Sometimes, users won't see the vital information; all they see is just another roadblock while they try to use your site. Also, make sure your message is easy to understand and concise, and doesn't include a lot of technical detail. Remember, it's just a quick alert, not a minisite. Be considerate of your users, and they will pay attention to your messages.

To include a pop-up message on your page, follow these steps:

1. Open the document to which you're adding the message and then place the cursor in the body section of the document.
- Alternatively, you can highlight a section of the code.
2. In the Behaviors panel, click the Add Behavior button (the plus sign) and select Popup Message.
 3. In the Popup Message dialog box that appears (see Figure 6-13), type the message and then click OK.

Dreamweaver adds this code to the head of your document:

```
<script language="JavaScript" type="text/JavaScript">
<!--
function MM_popupMsg(msg) { //v1.0
    alert(msg);
}
//-->
</script>
```

Figure 6-13: Type what you want users to see in a pop-up message.



It also adds this code to the body of your document:

```
<body onLoad="MM_popupMsg('This is a popup message.')">
```

This particular example tells the browser to create an alert message box when the page loads and display the text: This is a popup message.

You can edit the message by changing the text in the code. You can also change the trigger and content of this behavior in the way we describe in the “Including the Open Browser Window behavior” section earlier in this chapter.

Chapter 7: Creating a Web Page from a Photoshop File

In This Chapter

- ✓ **Finishing a wire frame design**
- ✓ **Dealing with graphics: Slicing and optimizing**
- ✓ **Creating a transparent image**

Planning and wire framing can save you a lot of headaches later because it's easier to fix mistakes before the whole site is built. In Book II, Chapter 2, we show you how to create a Photoshop wire frame that you can use for a page layout. In this chapter, we show you how to take that wire frame to the finished product. Creating a wire frame helps you decide what belongs on the screen, figure out the underlying structure of your site, and make adjustments as necessary before committing to a whole design. We also show you how to develop a layout, slice a document, and prepare graphics for use on your Web page.

Taking a Wire Frame to a Finished Design

With a complete wire frame, follow these steps to create your page layout:

- 1. Launch Photoshop.**
- 2. Open your wire frame file.**
- 3. Choose your color scheme.**

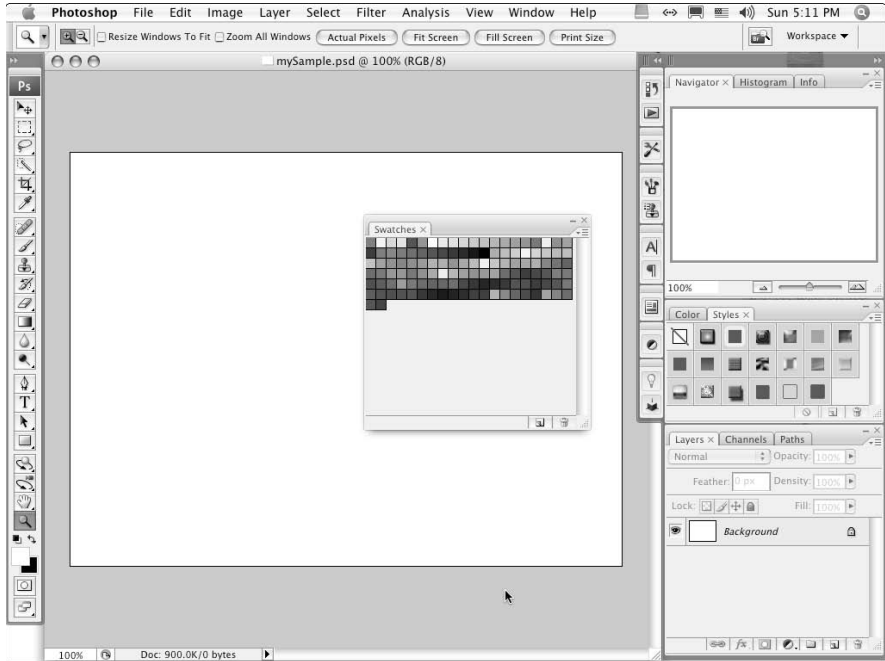
When choosing colors for the Web site, check whether the client has already chosen colors for its communications. If so, ask for the RGB or hexadecimal codes for the colors to ensure that you use the correct ones. Book IV, Chapter 1 has more details about the different types of colors and why you need RGB or hexadecimal colors instead of CMYK.

Some designers advise you to use the Photoshop Eyedropper tool to get the colors from a CMYK document, but this isn't a good way of getting the colors. It can be unreliable, and the resulting colors can be incorrect.



If no set of graphics standards exists or if the site you're designing needs a unique look, you have to choose the colors yourself. If you need information about choosing a good color scheme, read Book II, Chapter 4. You can also make a custom panel (see Figure 7-1), a process that is explained further in the “Managing colors with customer panels” sidebar.

Figure 7-1:
Create a
custom
panel to
make the
production
process
easier.



4. Create placeholder elements to fill in your layout.

To create the text elements, use the Text tool.

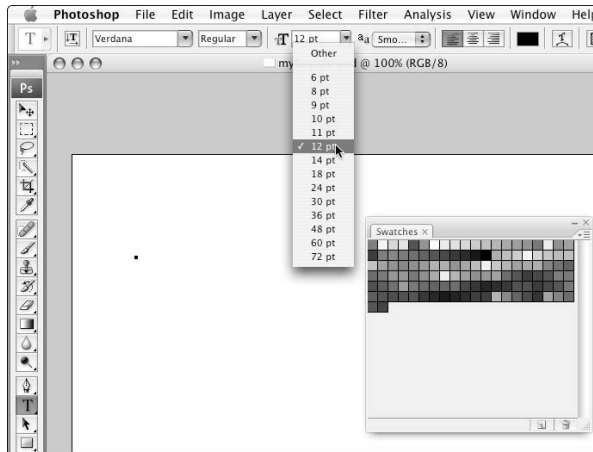
- a. With the Text tool (the one that looks like a capital T in the Tools panel) selected, click anywhere in the work area window.

Photoshop automatically creates a text layer for your new text.

- b. Select a font and color by using the context-sensitive menu that appears at the top of the screen, under the default Photoshop menu bar.

The Text tool menu appears when you select the Text tool or when you double-click an existing text layer to edit. See Figure 7-2 for an example of the menu.

Figure 7-2:
Select your font from the Text tool menu.



- c. Select a color from your custom panel by clicking the color chip in the Text menu and then clicking a swatch in your custom panel window.
- d. Click the spot in the work area where you want your text. Type the text.

You can edit, move, and change the text later, so don't worry if it's not perfect out of the gate. If you know what some of your headings are, use that text. If not, use some placeholder text — say, **Example Header** — so everyone understands that the text is just for example, not the finished content.

Figure 7-3 shows the color swatches (from Step c), as well as typed text.

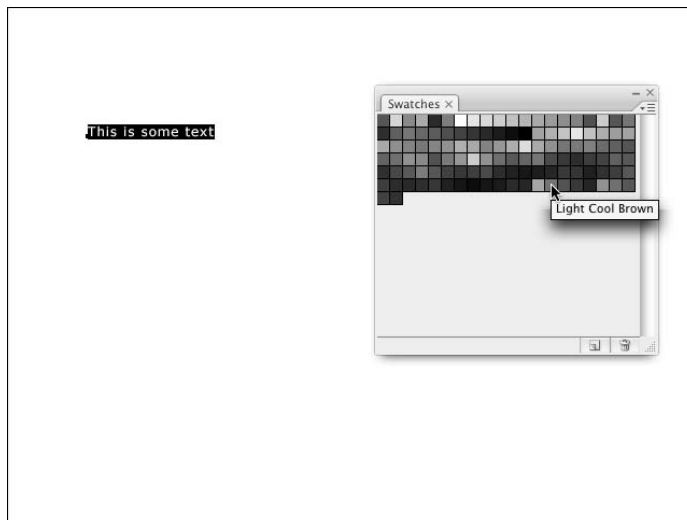


Figure 7-3:
Use a custom swatch to change the color of the text.

- e. *Create headings of at least three sizes for your primary, secondary, and tertiary headings. You want to apply colors, underlines, or any other special treatments you would like for your headings.*

Make sure that the priority of the headings is clear. Heading 1 should look more prominent than Heading 2, which is more prominent than Heading 3, and so on.

- f. *Create a placeholder paragraph with a font that's clear and easy to read.*

Verdana is a good font to use for Web content. At this stage, just type some nonsense so that you can see what the paragraphs will look like within your final design. Place a few of them into your design. Make sure you place headings into the layout, too (as discussed in Step e). Doing so gives you an idea of how everything will look together. The point is to make this version of your design look more like the finished product.

- g. *Create placeholder buttons by using the Rectangle tool to draw the shapes. Note: When you select the Text tool or the Rectangle tool and then click in your working document, a new layer is automatically created for you.*

Click the small black triangle in the Rectangle tool area to see other shape tools. Use the Line tool to create borders for your buttons.

After you make the button shapes, double-click the layer that Photoshop made when you created the rectangle; this launches the Layer Effects dialog box. From here, you can apply drop shadows, gradients, patterns, and many other effects. Click the name of an effect to open the options for that effect.



If you're creating a navigation bar, use just one rectangle and then break it up into buttons visually with the Line tool. This way, you have only one rectangle to contend with when it comes to lining up things, changing effects, or adjusting colors.

- h. *Create footer text.*

Make footer text small and discreet. This text includes important information, such as copyright information and contact information, but it shouldn't overpower the rest of the page.

- i. *Create banner text.*

Banner text should make a big impression. This text is going to be the name of your site, and it's part of your brand. This text must be more eye-catching than other text on the page.

Managing colors with custom panels

A custom panel isn't necessary, but it can certainly make things easier. It allows you to keep colors consistent throughout the project. You can also e-mail the file to other members of the team so they can match the colors.

To make a custom panel, follow these steps:

1. **Create a layer.**
2. **Make boxes with the colors in your color scheme.**
3. **Choose File⇨Save for Web & Devices.**
4. **In the color table panel, open the Color Panel menu (by clicking the *twirl-out menu* — the small black triangle icon at the upper-right of the panel).**

Figure 7-1 shows you where to find the panel and menu selection.

5. **Choose Save Color Table from the pop-up menu.**

Photoshop saves it to the Adobe Presets folder called Optimized Colors, by default. You can save the file to your Production Files folder so that it stays with the project, even after you back up the files for storage.

6. **Name your file something that is easy to identify.**

Using the project's name will help keep things tidy.

Load the custom panel by following these steps:

1. **In Photoshop, open the Swatches panel (Window⇨Swatches).**
2. **From the swatches pop-up menu (found when you click the small black triangle in the upper right of the swatches panel), choose Replace Swatches.**
3. **Select your panel and click Open.**

Note: Replace Swatches removes the current panel and replaces it with your custom one. If you choose Append instead of Replace Swatches, your custom panel is appended to the end of the list of swatches already available in the swatch panel.

You can bring the default panel back by selecting the twirl-out menu and choosing Reset Swatches — or you can replace it with another panel the same way you loaded your custom swatches.

Figure 7-4 gives you an idea of what your design might look like at this point. Making these placeholders helps you make sure your content is readable and that emphasis is on the right elements without cluttering the design with finished artwork. Design is an iterative process: You will layer changes on changes until you have the completed product.

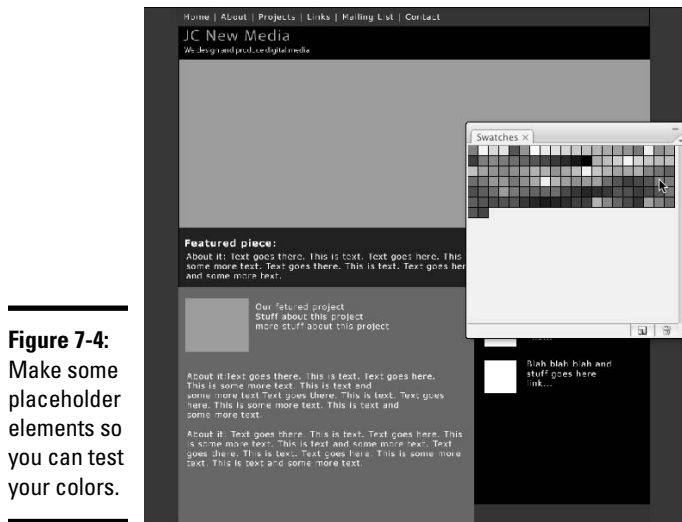


Figure 7-4:
Make some
placeholder
elements so
you can test
your colors.

At this point, you have a wire frame layout that's starting to have a little look and feel to it. It's by no means ready to go, but it's much more developed than a bunch of boxes. These next steps take you through the process of adding a little more personality to the layout by editing fonts and colors, adjusting text, and so forth. Just follow these steps:

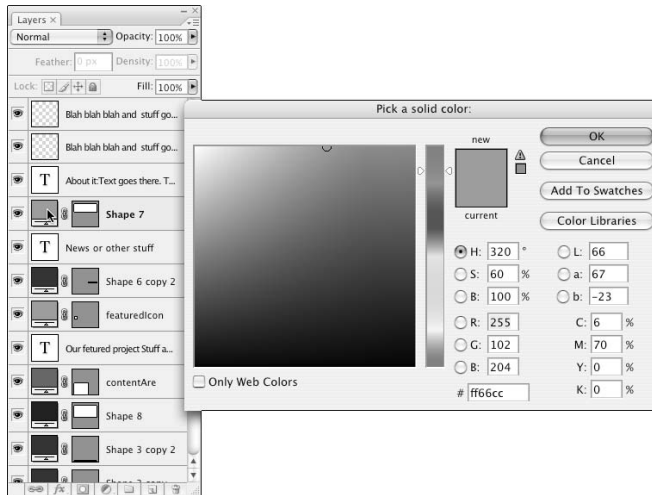
- 1. Edit fonts, content, colors, or any other properties of text after you create it by double-clicking the text layer or by single-clicking the layer, selecting the Text tool, and then directly selecting the text in the work area.**
- 2. Apply the color scheme to the layout.**
- 3. Double-click the color thumbnail to open the Color Picker and change the color.**

For the rectangles you created to represent your layout, notice that the layers have two thumbnails. (See Figure 7-5.) One is the layer thumbnail; the other is the vector mask thumbnail. Photoshop actually sees shapes created with the Rectangle/Shape tool as a layer that is completely filled with color and a vector mask that shows only a portion of that color fill. This allows you to easily edit the area filled with color without losing any quality through resizing or changing it.

- 4. Mouse over the custom swatches; when the cursor changes to an eye-dropper, you can click one of your custom colors to apply it.**

If you're not using Web-safe colors, you must clear the Web Only Colors check box in the Color Picker. See Book III, Chapter 4 for more details about how to use Photoshop Color Picker.

Figure 7-5: Click the color thumbnail to bring up the Color Picker.



5. Adjust the colors of the text until it has enough contrast that you can easily read it.
6. Adjust the headings, banner text, and other supporting text until they're readable and have the amount of emphasis you want them to have.
This is a judgment you have to make, and it's different for each project's content and objectives.
7. Apply color to the boxes you have created to represent your layout, experimenting until you produce the effect you want.



While you're adjusting colors and making decisions, don't sacrifice readability for a cool design. The point of a site is to convey information. Low contrast (for example, light gray text over a medium gray background) might look very cool but is hard for many users to read. Any design — even a nice-looking one — is a failure if visitors can't use the content. Visitors can and will look for a site that is easier to use. See Book II, Chapter 4 for more information and pointers about choosing colors and fonts for your site.

Finishing the Artwork

In Book II, Chapter 1, we show you how to create a wire frame. The first section of this chapter shows you how to apply a little look and feel to that wire frame by adding and adjusting color, text, headings, and so on. Here, we continue to modify the wire frame by focusing on completing the banner. Note that the techniques can be applied to any of your site graphics.

Choose images or create artwork for your banner and your site pages. Like with every other step in the process, choosing or creating artwork should be driven by reinforcing your message. Of course, you'll use your own taste when selecting images, but ultimately, the direction must be driven by communication and not artistic statements.



Make sure you have licenses to use any images and artwork that you didn't create. A license to use an image for print is not necessarily the same thing as a license to use it online. Read the agreement carefully before using images coming from another source.

We focus on the banner graphics for now, but keep in mind that your other images and graphics need to match the look and feel.

Placing a photo in your banner

The *site banner* is that identifying strip that lies across the top of a page. Often, the banner contains a company logo or motto or other identifier that viewers can recognize and relate to. You can jazz up this element of your site by adding photos to it. To do so, just follow these steps:

- 1. Open your layout in Photoshop and then open the images you're adding to the design.**
- 2. Drag the image to the layout file.**

Photoshop automatically creates a new layer for the image when it places it in the layout file.

- 3. Select the image file and then click the Background layer.**

The Background layer is the initial layer created by Photoshop by default. In a new document, this layer is empty or contains only a solid color, depending on the settings you make when creating the file. With a photo or other image, the Background layer contains the image.

- 4. Drag the layer into the layout document window.**

Creating a clipping mask

You create a *clipping mask* when you use a layer to constrain how much of another layer is visible. Here, we show you how to use a plain box (like the one you create in Step 4g in the “Taking a Wire Frame to a Finished Design” section) to constrain a photo to just the banner area — without having to distort or resize the photo. All you need is a rectangle that will represent the area you want to have as your banner. See Figure 7-6 for an example of a clipping mask used to constrain a graphic to the banner.

Figure 7-6: You can use this method any time you want to constrain the effects of a layer to only the layer used as the mask (second layer).



To constrain a graphic to the banner with a clipping mask, follow these steps:

1. Move the new image layer you created in Step 2 of “Placing a photo in your banner” so that it’s directly above the banner area box.
2. If your image layer isn’t selected, click to select it.
3. Choose **Layer > Create Clipping Mask**.
4. Move, resize, and edit the image layer as you normally would. However, the only portion of the layer that will be visible is the area that is precisely above the box layer.



Keep the background directly behind text or other important elements relatively clear of patterns. In other words, keep it simple behind the text because it’s harder to read text that’s on a busy background.

Implementing filters, adjustment layers, and blend modes

While you're working with your banner, don't be afraid to try some filters to change the look of the graphic or apply a blend mode or adjustment layer to change the look of the image until it looks how you want it to. Read Book III, Chapter 4 for information about using blend modes. Adjustment layers are covered later in this chapter.

To use a filter, follow these steps:

1. **Select the image layer.**
2. **Choose one of the selection tools. (Review Book III, Chapter 4 for a tour of the Photoshop tools.)**
3. **Select the area you would like to affect — if you want to affect the whole image, select the whole layer — and then choose **Filter** ⇨ **Filter Gallery**.**

This opens the entire gallery of filters (shown in Figure 7-7) and gives you thumbnail examples of what each does.



Figure 7-7:
The filter gallery can help you choose which effect to apply to an image.

4. Select the filter you want directly from the Filter menu by clicking it.

The filter is applied, and you see the effects. Some filters present you with a dialog box with customization options. Most allow you to preview the results before applying, so feel free to play with the options until it looks how you want.

To use an adjustment layer, follow these steps:

1. In the Layers panel, click the Adjustment Layer button.

The Adjustment Layer button looks like a half-black, half-white circle at the bottom of the Layers panel.

2. Select the type of adjustment you'd like to apply.

Several options control setting, such as brightness, color, and contrast. The best way to learn about them is to experiment. One that you will most likely find helpful is Hue/Saturation. You can change how the colors in your document look, or you can click Colorize and tint everything to the color you select.

You can constrain an adjustment layer to only one layer, the same way you constrain an image to the banner area. For example, choose the Hue/Saturation adjustment layer. Make sure the Preview check box is selected. Click and drag the saturation slider. If you slide to the left, the colors in your image *desaturate* (become less intense). Sliding to the right makes them more intense. Experiment with the other sliders to see how they affect the image. For a different effect, select the Colorize check box.

To use a blend mode, follow these steps:

1. Select the layer you would like to affect.**2. From the drop-down menu at the top of the Layers panel, choose a blend mode.**

This is another feature that's easiest to understand after you experiment a little. Basically, it affects how the chosen layer interacts with the layers below it. You can further augment the effect by changing the transparency of the layer (*opacity*).

3. For an interesting experiment, apply different transfer modes to different layers, modify the transparencies, and then reorder the layers.

The effects are different, depending on the images and transfer modes you select and any other adjustments you make. Have fun. If anything looks bad, you can step backwards in the History panel or just change the layer order for a different effect. Figure 7-8 shows the use of transfer modes to create effects.

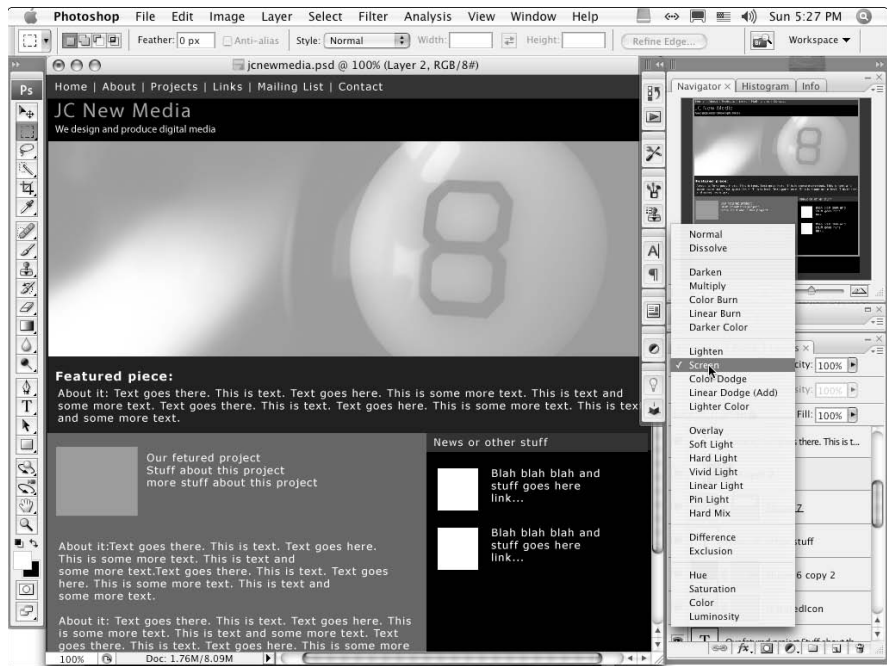


Figure 7-8: Use transfer modes to change the effects of layers on each other.

Slicing a Photoshop Document

After you create a Web page design in Photoshop, you're ready to turn it into an actual Web page. That involves creating graphics for the page and the HTML/CSS so that visitors can view your pages online. The process of creating graphics is *slicing*, *optimizing*, and *exporting*. When you slice a Photoshop document, you're telling Photoshop where to create areas that will become a series of graphics and HTML files.

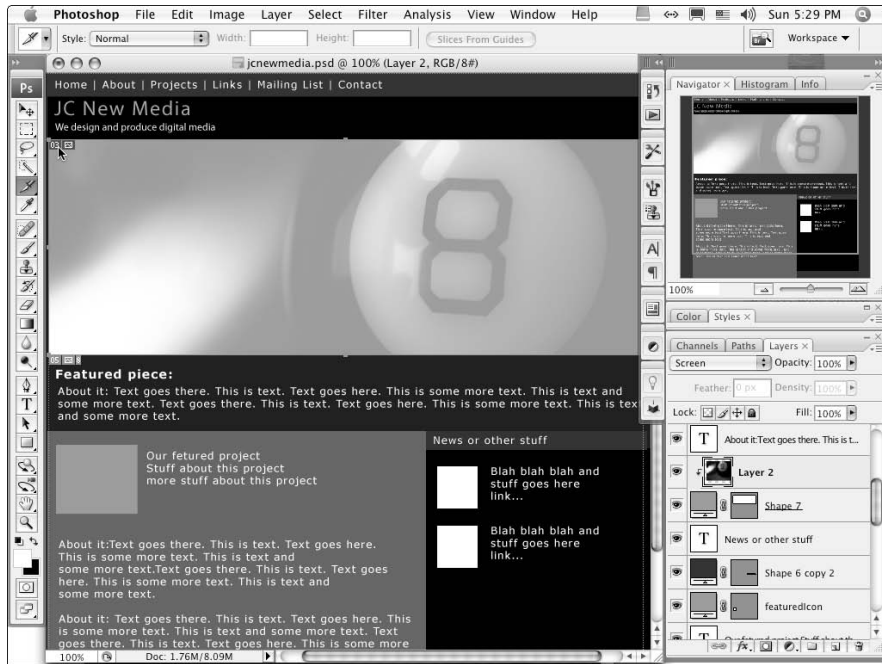


It is necessary to create slices only in areas that need to be graphics on the finished page. Content areas that will be filled with text shouldn't be graphics, so you don't need to create slices for them. Large areas of continuous color also don't need to be sliced because you can tell the browser to display colors via code. Use code as much as possible to generate your design because it's faster to download and also easier for more people to use.

Figure 7-9 shows you which areas of a document should be sliced and which will be generated with code.

- 1. Open the Photoshop document you created while designing your page.**

Figure 7-9: Slice only those areas that need to be graphics. Text and large areas of solid color are taken care of in the HTML.



2. Click the Slice tool — it's the one in the Tools panel that looks like a little knife.

Notice the second option under the Slice tool, called the Slice Select tool. That will be important after you create a few slices. For now, select the Slice tool. (See Figure 7-10.)

The Slice tool



Figure 7-10: The Slice tool looks like a little knife.

3. **Create your slices by clicking and dragging with the Slice tool over an area that needs to be a graphic. Repeat the click and drag over each area.**

Be careful to get the slices as accurate as possible. If you're off, it can create undesirable results, like weird lines of color in your finished page that shouldn't be there. You can adjust your slices as necessary. If you are unsure about the accuracy, zoom in to get a better look.

4. **If necessary, click the Slice Select tool, click the slice, and then move or edit the slice.**

As you create your slices, you see numbered boxes appear over your document; these represent the borders of the graphics you're creating. Make sure the lines of those boxes are aligned with the edges of your graphics areas. Figure 7-11 shows you what a sliced document looks like.



Figure 7-11:
A sliced
Photoshop
document.

Extra, slice tool tricks

The Slice and Slice Select tools have additional tools to help you work with your graphics. As we recommend with all the tools in Photoshop, experiment with the options to see what works best for you. The basics in the following list are simple, but you can explore many options after you get more comfortable:

- ✓ With the Slice Select tool active, double-click the small box next to the slice number (the one that looks like a rectangle with a jagged line in it). A dialog box opens that allows you to make some settings on the slice, such as a URL for a link, a target window, ALT text, and a message.
- ✓ In the context-sensitive menu bar, you'll also find a button that allows you to subdivide a slice. Click the button to see the dialog box with options for number of slices, orientation, and sizes.
- ✓ The Slice tool has options in its context-sensitive menu bar that allow you to specify a size or aspect ratio for your images.

Optimizing Graphics for the Web

After you have a Web page designed in Photoshop and you slice the document, the next step is to prepare your graphics for export, which is also called *optimizing and exporting* your graphics. This is the process that actually creates the graphics for your Web page. The Photoshop document will stay intact and will be ready for you to use if you need to make changes to the graphics or design. Keep the Photoshop document with the rest of your production files in the Production Files folder of your Web site. See Chapter 3 of Book II about Web site structure and how to organize your files.

To optimize and export your graphics, follow these steps:

1. Choose File → Save for Web & Devices.

Notice that your slices still have the lines and numbers but now look like semi-opaque white boxes over your document. Don't worry; the finished graphics don't have that appearance. Selected slices become transparent, so you can see what the graphics will really look like.

The Save for Web & Devices dialog box launches. Tabs along the upper left allow you to choose among different ways of working in this window.

- *Original*: Shows you the original document.
 - *Optimized*: Shows you the optimized version.
 - *2-Up*: Shows you the original document in the left window and the optimized version in the right.
 - *4-Up*: Shows you the original document and three versions with different optimization settings, which is useful if you need to compare several possibilities.
2. **For these steps, select 2-Up (shown in Figure 7-12) because it lets you see what you're doing to your image while you compress it, but doesn't overwhelm you with too many things to look at.**

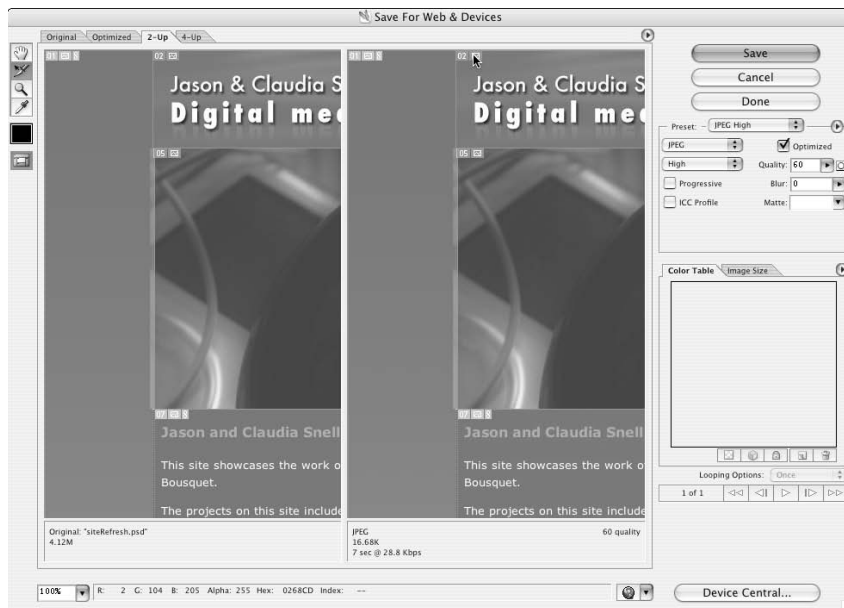


Figure 7-12:
The Save for Web & Devices dialog box in 2-Up mode.

You can select each slice individually or select groups of them for optimization.

3. **Click the Slice Select tool.**

It's located in the left margin of the Save for Web & Devices dialog box.

4. **Click a slice to select it; shift-click to select multiple slices.**

Slices become transparent when they're selected.

5. **After you select a slice or slices, make your settings for file format and compression amounts.**

See the upper-right area of the Save for Web & Devices dialog box, as shown in Figure 7-13. JPEGs have presets for low, medium, and high quality. A slider gives you more control. In general, the higher the number, the less the image is compressed, the better it will look, and the larger the file size will be. The file size and approximate download time are displayed at the lower left of the Optimized preview window. You want to adjust the optimization settings until you get the lowest file size while still keeping the best quality in your image. Although there are general guidelines to get you started, optimizing properly will take some experimentation because each graphic will have its own unique needs.

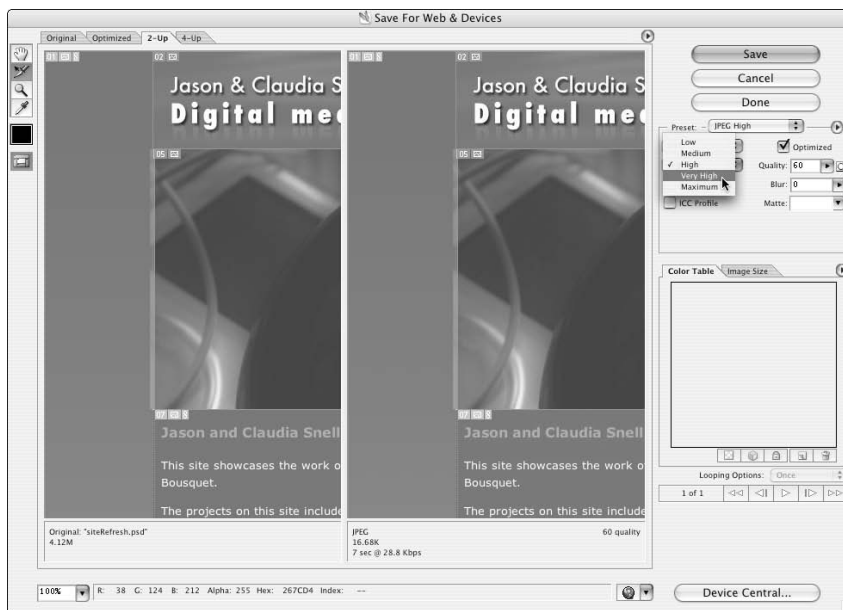


Figure 7-13: Optimizing graphics in the Save for Web & Devices dialog box.

GIFs have different settings; using more colors will generally give better image quality and higher file sizes.



For all images, the trick to compression is to find the right balance between file size and quality. It can take some playing around, so don't get discouraged. See Book IV, Chapter 1 for more information about which file formats work best with different types of graphics.

6. **If necessary, resize a graphic by using the Image Size tab (in the lower-right of Figure 7-13); enter the new size and then click Apply for the changes to take effect.**

Notice that you don't have to make the same compression settings for each graphic in your document. This is a very useful feature. Your design is very likely to have some areas that will be best as JPEGs and

others that need to be GIF. The ability to make different settings also allows you to make different settings for different graphics even if they are going to be the same type. This is handy because some graphics compress better than others. Each one will need special attention. The problem areas to look for are

- *Around text:* Make sure you don't compress graphics with text (like banners and logos) so much that the text becomes messy looking. If it does get messy, you went too far, and you need to increase the quality — and, consequently, the file size — just a bit.
 - *Gradients or other areas with color changes:* An example of this is a person's face, which can get *banding* or *dithering* when it is compressed too much. (To see what this effect looks like, take a photograph that has gradient type color changes and purposely overcompress it. You'll see strange bands of color with blotchy edges start to appear. It won't look good; you'll need to back off the settings a bit.)
7. **After making settings, click Save and navigate to the folder you'd like to save the images in.**
 8. **Click Save to save the images and files.**

Photoshop automatically creates an Images folder and saves your images to it. **Note:** Photoshop doesn't create an Images folder if you're saving out a single image that you didn't slice. The process of using the Save for Web & Devices dialog box is the same except that you make settings for the whole image all at once, and the file saved out is the whole document in its entirety.

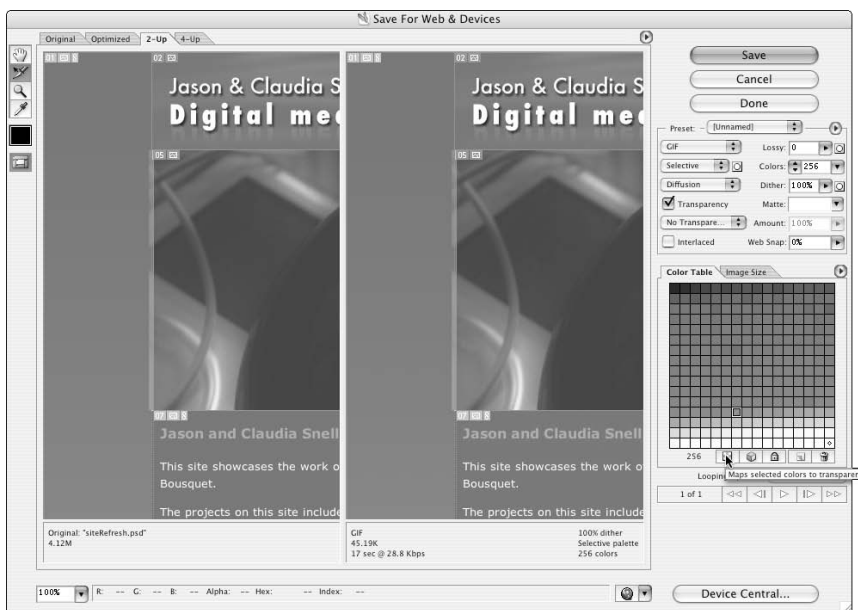
Making a Graphic Transparent

The GIF format supports transparency. This is very useful when creating banners or heading text that you'd like to "float" over a background graphic.

1. **Open the Save for Web & Devices dialog box by choosing File⇨Save for Web.**
2. **Select GIF as the format in which to save your graphic.**
3. **With the Eyedropper tool, click the color you'd like to make transparent.**
4. **Click the Maps Selected Colors to Transparent button (shown in Figure 7-14) in the Color Table panel.**

The color becomes transparent in your exported graphic file only; the original is not affected.

Figure 7-14: Click the Maps Selected Colors to Transparent button to make a color disappear.



Make sure the background color is the same as or similar to the color you'd like to float the graphic over. If you have a dark/black background, create your transparent image over that color, and then make it transparent by following the preceding steps. Same goes for light colors. The reason is that when you select a color and make it transparent, it makes just that color go away, but the image actually has a bunch of intermediary colors that are just a little different all around the elements in your graphic. Those won't become transparent. This is a good thing because it keeps the edges of your graphic nice and clean, as shown in the bottom image in Figure 7-15. However, if you have dark intermediary colors but your background is light, you'll get a weird aura or halo all around your graphic, shown in the top image in Figure 7-15.

Figure 7-15: The halos disappear when the graphic is placed over the same color it was designed with.



Chapter 8: Meeting HTML's Powerful Friends

In This Chapter

- ✓ **Discovering Web technologies**
- ✓ **Exploring the server side**
- ✓ **Inspecting the client side**

When you want to take a Web site to the next level, you add bells and whistles to your designs that make things happen. Of the many ways to achieve interactivity on a Web site, Flash is one solution. But we're talking HTML here, and HTML interactivity comes in two flavors: *server side* (occurs by code that's interpreted by the server) and *client side* (occurs by code that's interpreted in the client's [the person viewing the Web page] browser). Yes, Virginia, that's right; you're actually writing a program to get the interactivity you want. This chapter introduces you to Web technology written for both sides of the street.

Web Technologies Defined

All interactive technologies contain code that must be interpreted before the interactivity occurs. What separates the technologies is how the code is deciphered. The software that deciphers the code and causes the interactivity is either a plug-in for the user's browser (client side) or the Web server (server side). When you're deciding what type of code to use, a couple of factors come into play:

- ◆ **If you use software that must be interpreted by the user**, you need to know whether your client's intended audience has the plug-in. If not, will they have the inclination to download the applicable plug-in?
- ◆ **If the technology relies on software housed on the server**, make sure that your client's server has the applicable software available and also that it's the proper version of the software for the code you plan to write — or is used in the application (such as a blog) that you're adding to the client's site.

The upcoming sections discuss the objects that are used by many interactive technologies.

Vary your content with variables

Most interactive technologies use variables. A *variable* is, quite simply, a placeholder for information that can vary. Talk about your logical names. An example of a variable is a form field. When a user enters information, the variable is no longer *null* (empty). ASP, CGI/Perl, JavaScript, and PHP are a few technologies that use variables. An example of a variable in action is a PHP page transmitting the information from a form into a database. The variable populates the applicable field in the database. Listing 8-1 shows some PHP code with variables.

Listing 8-1: PHP Code with Variables

```
if ((isset($_POST["MM_insert"])) && ($_POST["MM_insert"] == "RegisterUser")) {
    $insertSQL = sprintf("INSERT INTO users (FirstName, LastName, Address, City,
        `State`, Zip, email, InfoStore, announcements, productsUsed, feedback)
        VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)",
        GetSQLValueString($_POST['FirstName'], "text"),
        GetSQLValueString($_POST['LastName'], "text"),
        GetSQLValueString($_POST['Address'], "text"),
        GetSQLValueString($_POST['City'], "text"),
        GetSQLValueString($_POST['State'], "text"),
        GetSQLValueString($_POST['zip'], "int"),
        GetSQLValueString($_POST['Email'], "text"),
        GetSQLValueString(isset($_POST['InfoStore']) ? "true" :
        "", "defined", 'Y', 'N'),
        GetSQLValueString(isset($_POST['announcements']) ? "true"
        : "", "defined", 'Y', 'N'),
        GetSQLValueString($_POST['productsUsed'], "text"),
        GetSQLValueString($_POST['feedback'], "text"));
```

The variables are in single quotations: for example, 'FirstName'. Notice that the variable names don't have spaces. When naming variables, spaces are verboten, as are other symbols — such as punctuation — because these are used in the actual code.

Variables can hold text, numbers, or Boolean values (True or False). When you name a variable in your code, you *declare* it. To access the data stored within a variable, you *call* the variable's name. The data within a variable can be interpreted by a function. When you name a variable, you must

- ◆ **Give the variable a unique name.** If you give a variable the same name as a variable that's already been declared and is supposed to hold different information, or if you give it the same name as a *function* (a sequence of code that performs a task), chaos ensues.
- ◆ **Give the variable a descriptive name.** For example, when you name a variable `FirstName`, you know exactly what information the variable is destined to hold. For that matter, so does anyone else working with your code.



You declare a variable before it's used in your code. Many types of variables are defined in the head of a document. JavaScript is an example of Web technology that declares variables in the head of an HTML document. (See Listing 8-2.)

Listing 8-2: Defining Variables

```
function MM_preloadImages() { //v3.0
    var d=document; if(d.images){ if(!d.MM_p) d.MM_p=new Array();
        var i,j=d.MM_p.length,a=MM_preloadImages.arguments; for(i=0; i<a.length; i++)
            if (a[i].indexOf("#")!=0){ d.MM_p[j]=new Image; d.MM_p[j++].src=a[i];}}
    }
```

The line of code that begins with `var d=document;` is declaring that variable `d` is equal to the HTML document. This is part of the JavaScript code that preloads images for a drop-down menu when a user hovers the cursor over a menu link.

A variable also has scope, not to be confused with a popular hygienic product that gives you minty fresh breath or some such nonsense. The *scope* of a variable determines where it can be used. A variable can be local or global. You define a new variable in JavaScript by typing **var** followed by an equal sign and then the variable's name. If you're creating code for an ASP page, a variable is designated by `dim` followed by an equal sign and the variable's name. If the variable is declared within a function, as in Listing 8-2, the variable can be used only within that function (*local*). However, if the variable is defined outside a function, it's *global* in nature and can be used anywhere in the code. When used in a function, a local variable takes precedence over a global variable with the same name. However, to avoid confusion, give each variable a unique name.

Conditional statements

Another staple frequently used in interactive technologies is the conditional statement. In a nutshell, a *conditional statement* executes code depending on whether one or more conditions are true. If the condition(s) are true, certain lines of code are executed; otherwise, different code is executed. You might think of a conditional statement as the proverbial fork in the road. Listing 8-3 shows the basic structure of a conditional statement.

Listing 8-3: A Conditional Statement in Its Most Basic Form

```
if (condition)
    {(action);
}else;
    {(else-action);
}
```

A conditional statement almost always begins with the word `if`. If the condition evaluates as true, the next lines of code are executed. If the condition evaluates as false, the script advances to the line of code that begins with `else`, which is followed by the lines of code that execute when the condition evaluates as false. Listing 8-4 is an example of a conditional statement from some PHP code.

Listing 8-4: A Conditional Statement

```
<?php if ( comments_open() ) : ?>
<h4 id="postcomment"><?php _e('Leave a comment'); ?></h4>

<?php if ( get_option('comment_registration') && !$user_ID ) : ?>
<p>You must be <a href="<?php echo get_option('siteurl'); ?>/wp-login.
php?redirect_to=<?php the_permalink(); ?>">logged in</a> to post a comment.
</p>
<?php else : ?>
<form action="<?php echo get_option('siteurl'); ?>/wp-comments-post.php"
method="post" id="commentform">
```

The code in Listing 8-4 is part of a blog page. If the option for users to leave a comment is open, the next lines of code execute, which enables the user to leave a comment. The second conditional statement queries the blog options to see whether unregistered users can leave comments. If the blog requires registration, the user is told that he must be logged in to leave a comment. The fork in the road — namely, the code that reads `<?php else: ?>` — is interpreted if registration is not required. In this case, a comment form appears on the page, and the user can add a comment about the post.

Here we go loop-de-loop

Another item commonly used in programming is the loop, which executes lines of code a given number of times. The most commonly used loops are `while` and `for`. A `while` loop executes lines of code while a certain condition is true. A `for` loop specifies how many times the lines of code repeat. Listing 8-5 shows the basic structure of a `for` loop.

Listing 8-5: For Loop

```
for (initial value; test; increment)
{
    execute this code;
}
```

The loop is set up as follows. The line of code that begins with `for` shows the initial value, what must occur for the loop to continue, and the amount by which the initial value will increment. As soon as the test evaluates as false, the loop stops, and the next bits of code are executed. Listing 8-6 shows a real-world example of a `while` loop as it would appear in a Web page.

Listing 8-6: Using a Loop

```
$i=0;
while ($i < $num) {
$name=mysql_result($result,$i,"name");
$contact=mysql_result($result,$i,"contact");
$city=mysql_result($result,$i,"city");
echo "<b>Name:</b> $name </br><b>Contact Person:</b>
    $contact</br><b>City:</b> $city</br><hr>";
$i++;
}
```

Here's how to break down the code in Listing 8-6:

The initial value of the variable `i` is 0.

The loop continues while `i` is less than the variable `num`, which is equal to the number of rows in a database that contains contact information in a mailing list.

The next lines of code retrieve information from a MySQL database and enter the values in a table row.

The line of code that reads `i++` increments the value of `i` by a value of 1, loops the code again, retrieving the next row of information from the database, creating a new row and populating it with the information from the database.

The loop stops when the value of `i` exceeds the number of rows in the database.

Creating functional functions

You can find lots of ready-made code by surfing the 'Net, but the fun comes when you create your own code and put it to use in your Web designs. Functions are used in JavaScript, Perl, and C++. When you create a *function*, you create code that performs a specific task. The function accepts information from the user in the form of an argument. The argument is examined by the function's code, and a result is displayed. Listing 8-7 shows a function in its simplest form.

Listing 8-7: A Really Simple Function

```
function testPassword(argument)
{
the code;
}
```

In the example in Listing 8-7, the function's name is `testPassword`. It has one argument. The code determines the end result while evaluating the password. Listing 8-8 shows a real-world example of a function used in JavaScript.

Listing 8-8: A Real-World Function

```
<script language="javascript">
function getPassword () { Passwords = new Array('x001', 'xoxo1', 'de10r');
    var passwordMatches = false;
    var userPassword = prompt ("Enter your password:");
    for (i=0; i < Passwords.length; i=i+1)
    {
        if (userPassword == Passwords[i]) passwordMatches = true;
    }
    return passwordMatches;
}
var theResult = getPassword();
if(theResult == true) { alert("Correct")
}
else
{
alert("The password you entered is incorrect");
}
}</script>
```

The name of the function is `getPassword`. The argument defines the password as an array and sets the value of the variable `passwordMatches` to `false`. The next line of code prompts the user to enter his password. A loop evaluates the user-entered password against those in the array. If the password matches, the function alerts the user that the password is correct. Otherwise, the user is notified that the password is incorrect. Code like this can be used to password-protect a Web site, but is not really secure. This is just a simple example.

Looking on the Server Side

The preceding sections of this chapter (you did read them, didn't you?) are a brief primer in writing code. You write code when using programming languages that are executed on the server side as well as on the client side. The following sections introduce server side programming languages.

ASP/ASP.NET

ASP (Active Server Pages) is the powerful Microsoft server side language. When a user surfs to a Web site with an ASP page, the Web server determines the content delivered to the user's Web browser. ASP pages are used when content of the page changes frequently. An example of a use for an ASP page is a list that is generated from information in a database. The ASP code on the server side plucks the information from the database and delivers it to the user's browser.



The ASP code is parsed by a DLL (Dynamic Link Library) file called ASP.dll. Figure 8-1 shows ASP code that is used to retrieve data from a Microsoft Access database.

```

97 </td>
98 </table>
99 <table width="100%" bgcolor="#6699FF" border="0" cellspacing="0" cellpadding="0" >
100 <tr>
101 <td >
102 <div align="left">
103 <table width="600" border="0" cellspacing="0" cellpadding="4" align="center" bgcolor="#FFFFFF">
104 <tr>
105 <th align="left" colspan="6" class="unnamed1">Employee Phone Numbers</th>
106 </tr>
107 <tr>
108 <th align="left">Title</th>
109 <th align="left">Last Name</th>
110 <th align="left">First Name</th>
111 <th align="left">Department</th>
112 <th align="left">Location</th>
113 <th align="left">Phone Number</th>
114 </tr>
115 <!-- Database connection begins here -->
116 <
117 Dim Connect, ObjEmployee
118 Set Connect = Server.CreateObject("ADODB.Connection")
119 <
120 <
121 <
122 <
123 <
124 <
125 <
126 <
127 <
128 <
129 <
130 <
131 <
132 <
133 <
134 <
135 <
136 </table>
137 </div>
138 </td>
139 </tr>
140 </table>
141 </body>
142 </html>

```

Figure 8-1:
Creating
ASP code
for fun
and profit.

To create ASP pages, you must have a Microsoft Web server (the current version is IIS 6.0) on your local system. As of this writing, the URL to find out more information about IIS 6.0 is

www.microsoft.com/WindowsServer2003/iis/default.msp

Another derivative of an ASP page is ASP.NET. ASP.NET is an ASP page created within the Microsoft .NET Framework. You can think of ASP.NET as ASP on steroids. ASP and ASP.NET feature *session variables*, which are variables that can be accessed by different ASP pages on the site. In essence, a session variable is a variable that is remembered during a user's *session*, which is a visit to the Web site with the ASP.NET pages.

.NET Framework

The .NET Framework is Microsoft's brainchild. Microsoft .NET Framework is an environment for building, deploying, and running Web services and other applications. The .NET Framework consists of three main parts: Common Language Runtime (CLR), the Framework classes, and ASP.NET.

The languages available to developers upon installation of the SDK (software development kit) are Managed C++, C#, VB (Visual Basic), and JavaScript. The .NET Framework features multiple language support, offers a vast set of libraries, is open-standard friendly, and is supported by HTTP and XML. There is a learning curve, however. If you decide to dip your toe into the vast .NET ocean, you'll have to become familiar with programming in this platform. Currently, .NET is available for Windows only.

JavaScript

JavaScript has been around for some time. Back in the Jurassic period of the Internet, Web pages were *static*. In other words, what you saw was what you got. Then, someone got the bright idea that it would be “way cool” if visitors could interact with Web pages. Of course, that required some type of language that instructed the pages what to do when visitors interacted with them. Enter JavaScript.

With JavaScript, all sorts of cool things are possible. You can write JavaScript scripts to make a menu drop down when a user hovers his cursor over a menu link, swap images, have form elements interact with each other, and more. JavaScript works with all major browsers.

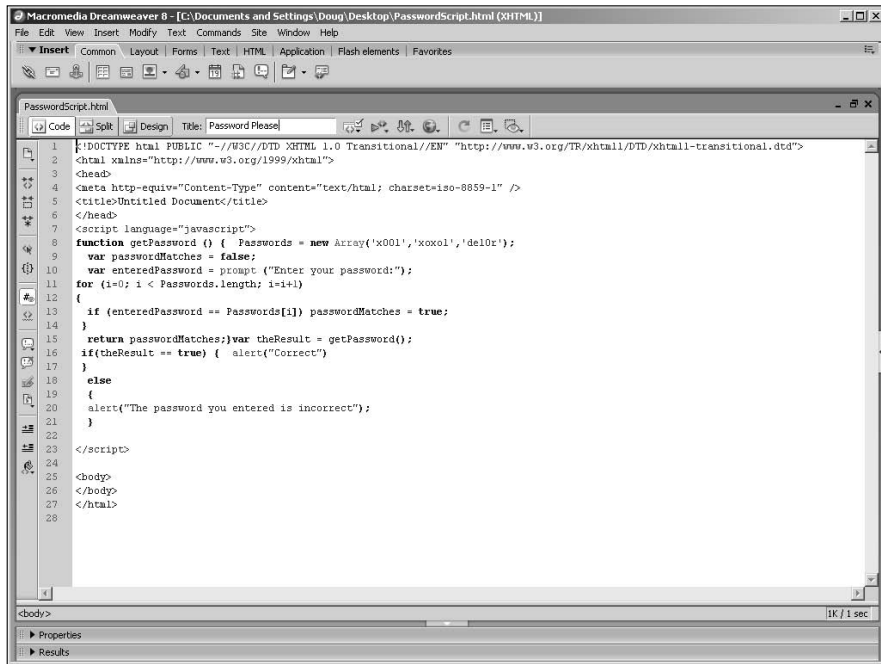
One of the great things about JavaScript is that you don't need to know a lot about programming. And you don't need a Web server to test your JavaScript code. You can create JavaScript scripts in any HTML editor and test them in your default Web browser.



You can find scripts you can paste into your HTML documents by typing **JavaScript scripts** in your favorite search engine. Using other people's scripts is an excellent way to learn JavaScript. Simply view the source code, and you'll have an idea of how the code achieves the desired result. You can also change the author's code to get a feeling of how JavaScript works. JavaScript uses variables, conditional statements, and functions.

JavaScript is object-oriented programming (OOP). If you've ever dabbled with Flash ActionScript, you've worked with object-oriented programming. JavaScript objects are defined in classes. An image is an object, which is defined by the title of `img`. A class consists of objects that are similar. Objects have properties. For example, the `image` object has the following properties: `name`, `height`, `width`, and `alt`. Objects have methods. For example, the `Form` object has a `submit` method (used to submit a form) and a `reset` method (used to reset the form). To find out more information on JavaScript, see Book VI, Chapter 1. Figure 8-2 shows a script that's used to validate a user's password. A script like this isn't very secure, but can be used for simple security, such as for a client's photo gallery.

Figure 8-2:
I get by with
a little help
from my
JavaScript.



PHP

PHP is an open source scripting language that you can use in HTML pages. Pages created with PHP code are delivered after being decoded by the server. Pages with PHP code will work only if your Web server has the applicable version of PHP. The version of PHP needed depends on the code you write. Many applications use PHP code. For example, blogs use PHP code to add blog posts to a database and to display them in the blog visitor's browser. If you're working with PHP code, the only way to test the code is to upload the page to a server or to install a Web server with PHP. Pages with PHP code have a file extension of `.php`.

The beauty of PHP code is that a visitor doesn't need any special plug-in to evaluate the PHP code; it's all done with PHP software on the server. When you create PHP code (shown in Figure 8-3), you will be seeing red. Literally. When you write PHP code in an HTML editor such as Dreamweaver, the code is preceded by `<?php`, and the text is red. (Okay, you can't see the red in this black-and-white book.)

Figure 8-3:
Talkin' 'bout
my pu..pu..
pu..pu.. PHP.

```

161 <td><!-- InstanceBeginEditable name="Page Text" -->
162 <?php
163 $connection=mysql_connect ("
164 database.");mysql_select_db (
165 $name,$_POST['name']);
166 $query="SELECT * FROM members WHERE name = '$name'";
167 $result=mysql_query($query);
168 $num=mysql_numrows($result);
169 echo $num;
170 mysql_close();
171 print "<b><center>Database Output</center></b><br>";
172 if ($num=0) {
173 print "No matches for your query.";
174 } else {
175 $i=0;
176 while ($i < $num) {
177 $name=mysql_result($result,$i,"name");
178 $contact=mysql_result($result,$i,"contact");
179 $city=mysql_result($result,$i,"city");
180 echo "<b>Name:</b> $name <br><b>Contact Person:</b> $contact<br><b>City:</b> $city<br><br>";
181 $i++;
182 }
183

```

CGI/Perl

CGI/Perl is another programming language that's executed on the server side. Before you create scripts for a client's Web site using this programming language, make sure the Web server supports CGI scripts. Unix servers generally support CGI scripts.

When your client's needs exceed the capabilities of your bread-and-butter HTML pages — such as creating a chat room or displaying a page counter — you can employ a CGI/Perl script to achieve the desired result. The information is sent from an application on the Web server to the user's browser.

CGI (common gateway interface) is the protocol that defines the manner in which a Web server interacts with a program. For example, when a Web site visitor fills out a form and presses Enter, the HTML code determines how the form is delivered to the Web server and how it's processed by the Perl script. CGI conventions determine how the script processes the information.

You'll find many Web sites that offer free CGI/Perl scripts. Most of the sites also offer more advanced scripts for a reasonable price.

To show you the power of CGI/Perl scripts, Doug downloaded a script to display the current date on his Web page. The steps were simple and as follows:

1. Download the nms-textclock script from

www.scriptarchive.com/nms-download.cgi?s=nms-textclock&c=zip.

The zip file contains a ReadMe page with instructions on how to configure the script as well as the `text-clock.pl` file, which is the Perl script.

2. Follow the instructions to reconfigure the script and upload it to the applicable folder on your server.

Doug uploaded the file to the cgi-bin on his server. If you're unsure what folder the file should be in, contact your Web hosting service. You might also need to change the extension of the file to `.cgi`. Your Web server can provide you with this information as well.

3. **Change the permissions on your server so that the script can be executed. You can use an FTP client to change the Permissions property of the file to 755.**

This means the owner can read, write, and execute the file, whereas visitors can only read and execute the file.

4. **Add the following to your HTML document where you want the date displayed:**

```
Today is <p align="left">Today is <!--#exec cgi="/
cgi-bin/textclock.pl"--> </p>
```

5. **Save the file with an extension of `.shtml`.**

This tells the Web browser that the page includes code that will be executed with a server side script.

6. **Upload the page to your server.**

Figure 8-4 shows Doug's home page with the date displayed.

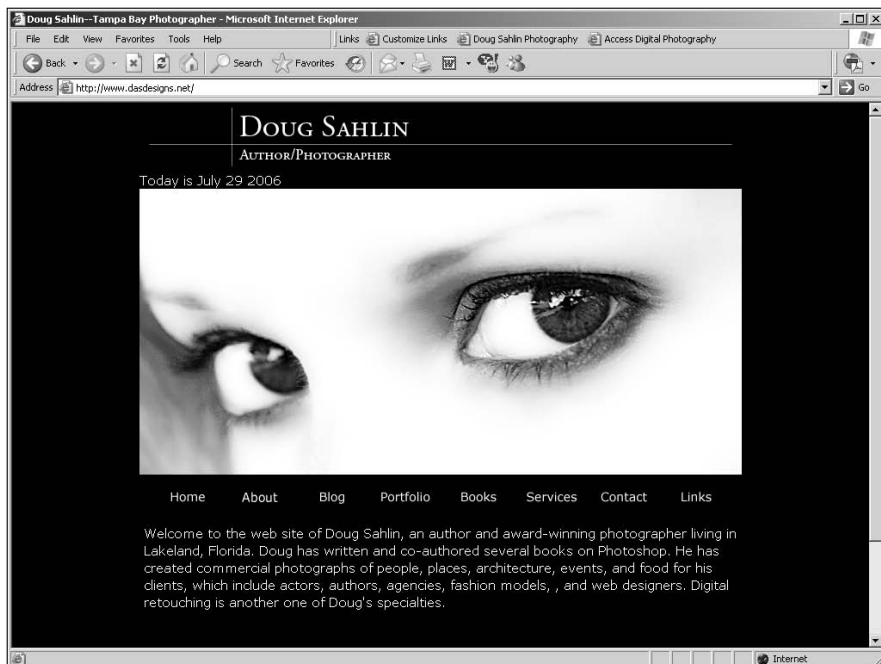


Figure 8-4:
Use a CGI/
Perl script
to display
the date.

Programming for the faint of code

If you've read this chapter from the start, you're probably firmly convinced that there are more programming languages than any mere mortal needs to learn. Yet, there are some developers (pencil-pocket-protected-duct-taped-four-eyed-card-carrying geeks) who know a lot of programming languages. You might be thinking, "I'm a Web designer, and I don't need to know all that stuff." And you're right; you don't. You'll find many scripts and applications where someone else has already taken care of the

pain of writing and debugging the scripts. And the really cool thing is that many of them are free. You can download them, complete with instructions, and plug them into your pages. Simply type the name of the programming language followed by **scripts** (for example, **CGI/Perl scripts** or **JavaScript scripts**) in your favorite search engine. You'll have pages of results from which to choose. When you find a really useful site, bookmark the page and pilage and plunder to your heart's content.

Checking Out the Client Side

Server side applications are wonderful, but there's also a good case for applications that execute *client side* — that is, within the user's browser. The only caveat is that Web site visitors must have the technology (plug-in) to decipher the code installed on their computers. Client side programs come in three flavors: Java, VBScript, and DHTML (dynamic HTML). The upcoming sections give you the skinny on the client side interactivity you can implement using Java or VBScript.

Java

Java — not to be confused with your morning picker-upper — is an object-oriented programming language developed by James Gosling and his colleagues at Sun Microsystems. It's not related to JavaScript. You can implement Java on the client side by writing code that is included in your HTML document. Creating Java code is unfortunately beyond the scope of this book. In addition to checking out *Java All-in-One Desk Reference For Dummies*, 2nd Edition, by Doug Lowe and Barry Burd (Wiley), you can find a number of resources online for writing client side Java code.

Java is chameleon in nature in that it also works server side. The easiest way to implement Java interactivity into your Web designs is by using Java applets. An *applet* is a program written in Java that can be downloaded and included in an HTML page. When you download a Java applet, the author supplies the software necessary to parse the code. The Java applet is uploaded to your server. You insert the author's code in your HTML document. You can find Java applets by typing **Java applets** in your favorite search engines. The results page will be filled with links to Java applets; many of them are free.

When you download a Java applet, you'll find source code that you add to your HTML document and the actual software for parsing the code. For example, a downloaded Java applet called `Mdataclock` contained three files in a zip file:

- ◆ `config.html`: Instructions on how to configure the applet
- ◆ `index.htm`: Contained the code
- ◆ `Mdataclock.class`: Code that is parsed by Java at runtime on the user's machine



The last file must reside in the same folder as the HTML document in which you're inserting the code for the applet. When you upload the site to the server, the `.class` file that parses the Java applet must be uploaded as well.

Figure 8-5 shows the code as inserted into an HTML document.

```

155 </td>
156 </td>
157 <td></
158 </td>
159 </table>
160 </table>
161 <table width="700" border="0" align="center">
162 <tr>
163 <td><applet code="Mdataclock.class" width="320" height="20">
164 <param name="background" value="000000">
165 <param name="font_color" value="FFFFFF">
166 <param name="font" value="verdana">
167 <param name="font_size" value="14">
168 <param name="font_type" value="0">
169 <param name="clock_type" value="4">
170 <param name="autore" value="Massimo Giari">
171 </applet><br>
172 </td>
173 </tr>
174 </table>
175 <br>
176 <br>
177 <table width="700" border="0" align="center" cellpadding="5" cellspacing="0">
178 <tr>
179 <td><p>Welcome to the web site of Doug Sahlin, an author and award-winning photographer living in
180 Lakeland, Florida. Doug has written and co-authored several books on Photoshop. He has created
181 commercial photographs of people, places, architecture, events, and food for his clients, which
182 include actors, authors, agencies, fashion models, , and web designers. Digital retouching is another
181 </td>
182 </table>
    
```

Figure 8-5:
A tasty little
applet.

VBScript

Guess what folks? VBScript is yet another programming language. VBScript is Microsoft's brainchild. Therefore, it's used only with Windows and Internet Explorer. VBScript can add functionality to your Web pages, and it's relatively easy to learn. In the end, though, you'll have to make the decision of whether VBScript is something you want to take on.

When you want to create a page with VBScript, you have to include instructions for the browser by using the `<script>` tag. You can add scripts in the head or the body of a document. Listing 8-9 shows a VBScript that will show the date and time on a Web page.

Listing 8-9: Telling Time with VBScript

```
<script type="text/vbscript">
document.write("Today's date is " & date())
document.write("<br />")
document.write("The time is " & time())
</script>
```

In a nutshell, the code uses objects (`date` and `time`) from the VBScript language. The code tells the browser to retrieve the current date and time from the user's computer and display the results (`document.write`) in the user's browser. Figure 8-6 shows the date and time as displayed on a Web page.



Figure 8-6:
Does anybody really know what time it is?

VBScript uses variables, arrays, and conditional statements. You can also use VBScript to jump through hoops by creating functions. A full-blown VBScript tutorial is beyond the scope of this book.

Chapter 9: Web Site Usability and Accessibility

In This Chapter

- ✓ **Avoiding common mistakes that make a site less friendly**
- ✓ **Understanding usability**
- ✓ **Addressing accessibility concerns**
- ✓ **Using Dreamweaver to make your pages more accessible**

Designers, developers, and professionals involved in Web site design and implementation work to improve user experience on their sites. Web standards have been developed to help Web professionals create sites that work well for users and producers themselves. If you're in the Web development industry or you're in the midst of developing your own Web site, chances are you have heard the terms *usability*, *accessibility*, and *standards* — and wondered what the buzz is all about.

Ensuring Your Site Is Easy to Use

If you browse the Web regularly, chances are good that you've visited a site that's confusing and difficult to navigate or use. That experience is frustrating and often prevents full use of the site. If you've experienced this, you've suffered poor Web site usability; so have many other customers. You might wonder about the logic behind these sites. Didn't the designers and developers think of their users? They probably thought they did — but in reality, didn't. Web teams often make a couple big mistakes, including

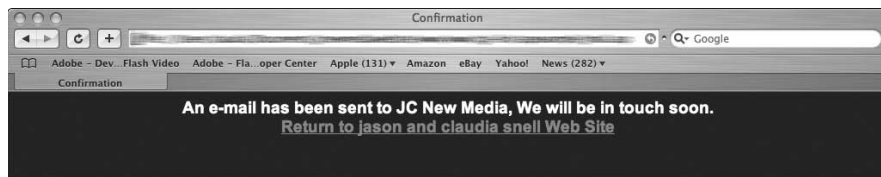
- ◆ **Assuming users will understand what they're building without asking them whether they do:** Always make testing a part of your plan up front; that way, you won't be rushed when it's time for testing.
- ◆ **Viewing negative feedback from users as nitpicking:** Why ask for feedback if you don't plan to improve the site?
- ◆ **Not asking themselves, "Who are we building this for?":** If the answer is that they're building a site for themselves, ignoring people who end up visiting the site is okay.

Realistically, all sites are ultimately built for the visitors. Logic follows that you conduct usability testing to build a site that works well for visitors. Usability testing can be informal or formal. The approach you take depends on the scope of your project, the size of your budget, and the nature of your audience. For informal usability testing, sit down with a person who represents your target audience. While the tester tries to use your site, note any areas that cause problems, ask questions about how easy or hard it is to use the site, and give your tester a list of tasks to accomplish (again, noting how easy or difficult any task is to do). More formal testing might involve e-mailing a link to the site to several individuals, along with a questionnaire that each person fills out and returns to you within a specified time frame.

After you get some feedback, you'll have a better idea of what your users need your site to do for them. You can start making design choices based on their needs instead of your own (or your client's). The resulting Web site benefits your users, which ultimately helps your site be successful. Book III, Chapter 10 has more detailed information about gathering and implementing feedback, but here are some simple rules of the road that every site designer must consider:

- ◆ **Everything has to work.** Think about the user's point of view: When she clicks a submit button after placing an order, for example, she should see an acknowledgment of receipt of her order, not a presentation about the company. That would be very frustrating, and that user would think twice before trying to use your site again. Figure 9-1 shows an expected *Message Received* screen the user sees after submitting a form.

Figure 9-1:
Create an acknowledgment page for form submissions.



- ◆ **Pages and graphics should be efficient and easy to navigate.** Users expect instant responses from Web sites, so pages and images must load quickly. Also, the site must be laid out so that users can easily find what they need. If a user has to hunt around your site and read a lot of heavy text, he will leave your site and go to one that helps him get what he wants with less hassle. Your site must have clear navigation. Refrain from using too many buttons and make everything reachable within a couple of clicks. If your site is large, consider a site map to help people get in and get what they want.

- ◆ **A site needs to be forgiving.** People make mistakes, so plan how to handle user errors cleanly and effectively. Polite error messages with easy-to-understand instructions are important and helpful. Whenever possible, users should have a way to step back and correct errors. Make sure you consider how the experience will work for users and minimize the number of times they have to reenter information if they do make mistakes.



When writing error messages, keep in mind that users don't need to know the technical information about the error. They only need to know how to fix the error — in simple, common language. Remember that your users aren't as familiar with how your system works as you are. Write error messages and instructions accordingly. People appreciate a concise set of bulleted instructions.

Also, give users correct contact information so they can get hands-on help if they need it. They should be able to call or e-mail your Web site support person. Make sure you put these customers in touch with the person who has the knowledge, training, and experience to help them, not a general-purpose, customer service contact.

- ◆ **Web sites must look good.** A site should look professional to the audience and the content should be appropriate. After all, visitors are developing an impression about the organization as well as the products and services that it represents. You wouldn't send poorly produced printed materials to your audience. Hold your Web site to the same, professional standard. After all, it's still your company's face — just on the Internet.



Some people believe that content is king, and they're partially right. With so many sites out there boasting good content, you must also make a good impression. You wouldn't send your top salesperson to a client meeting wearing pajamas and looking sleep deprived. You shouldn't send your Web site out there looking like that, either. Even if the rest of your organization is very professional, users who first encounter your company through a poorly designed site have no way of knowing that. They just see a messy, amateurish site — and move along.

Helping Users Access Your Site

Accessibility is closely related to usability but has important differences. Making sites *accessible* involves designing sites that allow people with disabilities to access the content of that site. Many individuals with disabilities (ranging from motor and visual impairments to cognitive and seizure disorders) have special needs when it comes to using Web sites. Some of these individuals depend on assistive technologies, which have special requirements in order to work properly. Other individuals have difficulties with color schemes, blinking content, confusing navigation, or audio-only content. Properly planned and executed Web sites vastly improve the user experience for these individuals.

A site that doesn't plan for accessibility issues can be difficult or impossible for some visitors to use. And beyond the common-sense aspect of making sites more accessible, some types of organizations are required to comply with laws pertaining to Section 508a of the Americans with Disabilities Act (ADA). Details about these laws are beyond the scope of this book, but you can find more information at the United States Access Board Web site (www.access-board.gov). The World Wide Web Consortium (W3C), which creates standards for Web development, offers a Web Accessibility Initiative (WAI/WCAG; or Web Accessibility Initiative/Web Content Accessibility Guidelines) to help site developers make their sites accessible. For more information, go to www.w3.org/wai.

Throughout the rest of this section, we highlight — give you pointers on — important things to remember when working to improve your site's accessibility.

Images need alternate content

Alternate content (also known as *alt text*) is text that is included within the image tag's `alt` attribute. This text needs to convey the meaning of the graphic it represents. Web page authors need to include `alt` attributes for each image on a page, even invisible formatting images (which should actually be avoided in favor of cleaner, easier to maintain CSS for formatting). The alternate text must include the content and/or context of the graphic image. Screen-reading technologies used by visually impaired visitors read the text of the `alt` attributes, so make sure you write them in a way that makes sense to visitors who can't see the images.

The alt text is what is displayed when a visitor hovers over the image, and it is also displayed in place of the graphic when a user has graphics turned off. Because of the many ways the alt text is used, typing just **button graphic** is not helpful. A more helpful tag is `Click here to see our specials`. (Please note, `alt` attributes are only one of several coding issues that must be addressed by Web site designers who are trying to make their sites more accessible.)

This is the correct way to code `alt` attributes:

```



```

Notice that the alternate content conveys what the graphic is functionally, describes the content of the image, or mirrors the text of the graphic. It's also acceptable to include `alt` attributes with no content at all, like this:

```
<img src='bullet.gif' alt="">
```

This conveys to a user's screen reader that this is a nonessential image that the user should ignore completely. Without it, some screen readers will read the whole tag to the user. That can create a very frustrating user experience for someone who has to hear each image tag read, especially if there are multiple instances of images with no alternate text.



Using `alt` attributes as a clever place to hide marketing keywords is not acceptable. This practice was started by marketing and search engine optimization “experts” who didn't understand how Web pages actually work. They discovered that text inside the `alt` attributes isn't displayed by the browser as part of the layout but does show up in some browsers when visitors place their cursor over them (and thus can be “seen” by search engines).

Oddly enough, using `alt` attributes to pack in a bunch of extra keywords can hurt your rankings in a search engine because most search engines are programmed to recognize pages that have an unusually high number of what appear to be keywords or unusual implementations of them. For more information on search engine optimization, look at Book VIII, Chapter 1.

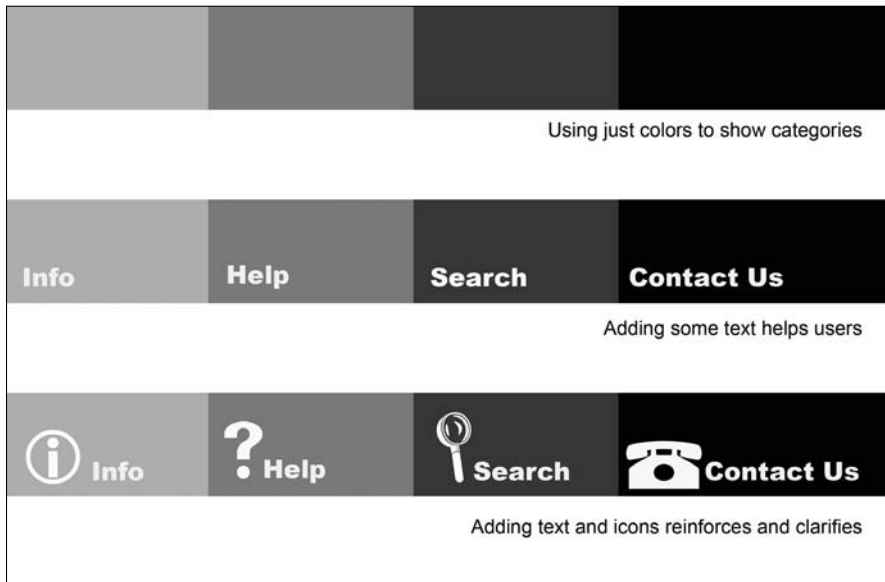
Another reason not to replace useful `alt` attributes with random words is because this affects not only visually impaired users but also users who have turned off their graphics or are using devices other than browsers. These individuals also need `alt` attributes to use your Web site.

Use more than one method to convey your information

Color-impaired visitors to a site can become confused or miss important information if it's conveyed only through color. Although *color-coding* (using color to indicate categories or to attract more attention to an item on the page) helps some individuals, you must also convey the information without the color-coding.

Using a set of colors plus icons (as shown in Figure 9-2) can be a way to ensure a good experience for a wider audience. Many people benefit from visual cues, so don't avoid visual cues in favor of designing for those who don't benefit; just make sure you use more than one method to convey your information. Using layout and groupings to signal that information belongs together is also useful. Even a well-planned, color-coding and icon scheme won't save a page that has a poor layout.

Figure 9-2: Icons can enhance a site for people, whether they have issues with colors.



If you plan to use color-coding or icons to convey categories, make sure you keep their numbers small. Three or four color-coded categories should work fine; more than that starts to look like a rainbow. Users can start to feel dis-oriented rather than assisted.

Pictures are important

Although images without useful `alt` attributes can be confusing for users with visual impairments, pictures assist those who have cognitive disorders. Choose images that support your content. If your pages are about taking care of dogs, include images that illustrate your points or convey the overall message about dog ownership. Don't include images of your last vacation or some other random image "just because it looks pretty." Make sure that the images match the tone or emotion the site is conveying. A site about a serious topic can be confusing if it includes images of laughing children.



Animations are also very popular and can be helpful. If you decide to include animations, be aware that you should avoid creating animations that flicker and flash excessively. Blinking and flickering can cause problems for people who suffer from seizure disorders. Most people don't enjoy these sorts of animations anyway, so avoiding them makes your site more pleasant for everyone.

Orientation tools are helpful

Orientation tools help users navigate the site as well as tell them where they are, no matter what page they're on. A site map and breadcrumbs are two such tools:

- ◆ **A site map (or site index) is a page that lists the links to the pages of your site.** It should mirror the navigation of the site and be presented in an outline style. Users find these to be helpful at some point, especially if the site is large. A site map is also particularly helpful to visitors who have cognitive or memory disorders. Users can pinpoint the pages they need very easily with the bird's-eye view of the site that the map provides.
- ◆ **Breadcrumbs are a trail of links that shows the user the path from the home page to the current page.** Breadcrumbs are typically displayed at the top of the screen and look like this:

Home >> About >> Our Products >> Widget

Each word in the breadcrumb trail should be a link to that page so that a user can backtrack up the trail, if necessary. Just like Hansel and Gretel.

Creating Accessible Web Pages

The good news is that several page-checking utilities are available to help you make accessible pages. Professional Web-design software (such as Adobe Dreamweaver) has many tools, validators, checkers, and resources available to keep you on track.

Using Dreamweaver to insert accessible elements

To use Dreamweaver to help you with properly coding your pages for accessibility, you must have the Accessibility feature turned on. To turn it on, follow these steps:

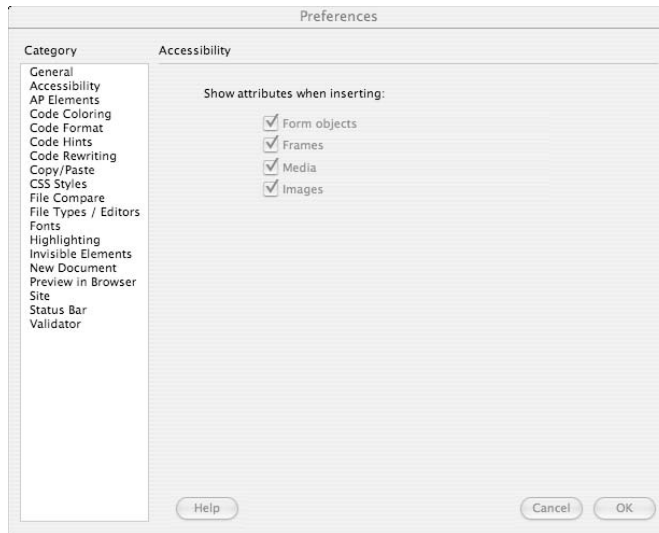
1. **Open the Preferences dialog box by choosing Edit⇨Preferences (Windows) or Dreamweaver⇨Preferences (Mac).**
2. **Select Accessibility under Category (as shown in Figure 9-3).**
3. **Enable the check boxes for the elements you want Accessibility help with.**

We recommend selecting them all. That way, Dreamweaver will remind you that you need to take special care with these possibly troublesome areas.

The choices are

- Form objects
- Frames
- Media
- Images

Figure 9-3:
The Dreamweaver Preferences dialog box in Dreamweaver.



4. Click OK to exit the Preferences dialog box.

To insert an image and `alt` attribute using the Insert menu, follow these steps:

1. Choose Insert→Image.

The image browsing window pops up. Select the image you want to add.

2. Click OK (Windows) or Choose (Mac).

The Image Tag Accessibility Attributes dialog box appears, as shown in Figure 9-4.

3. Enter the `alt` attribute text and then click OK.

The image and descriptive tag are inserted into your code.

To insert an image via the Insert bar, click the Insert Image icon on the Common tab. When the Image Tag Accessibility Attributes dialog box opens, enter the `<alt>` tag text and then click OK.

Figure 9-4: Insert an image and include the alt attribute.



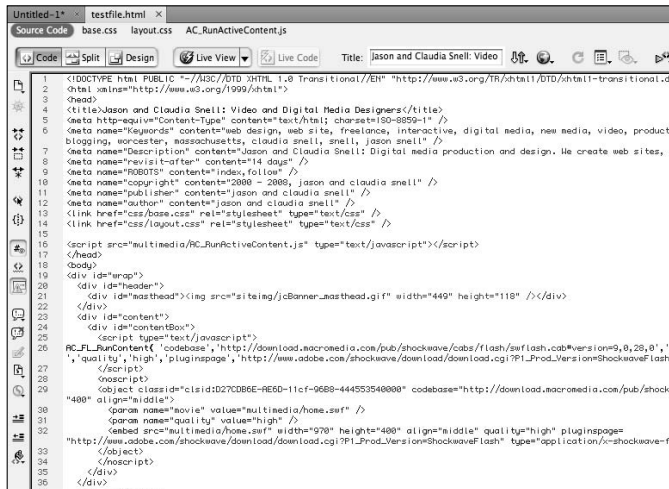
Using Dreamweaver to check your site's accessibility

Dreamweaver has tools to check your code, including code validators and site-testing reports that can check the accessibility of your page or site. The accessibility reports show the areas of your code that might cause problems for individuals with disabilities.



Before you start running reports in Dreamweaver, you might want to turn on line numbers in your Code view. The line numbers in the Code window help you find your place in the code. To turn on line numbers, choose View⇨Code View Options⇨Line Numbers. The line numbers appear in the left margin of the Code window, as shown in Figure 9-5.

Figure 9-5: Enable line numbers in the Code window.



Using Dreamweaver in Split mode (as shown in Figure 9-6) enables you to see the code and the design at the same time. Being able to see both code and design helps you make sure you don't code something that looks bad.

It also lets you keep an eye on your code to make sure that it doesn't get messy when you use the WYSIWYG tools. To turn on Split screen mode in Dreamweaver, click the Split button.



Figure 9-6:
Use Split
mode to
see the
code and
the design
at the
same time.

Running reports

Dreamweaver provides different methods for checking a site's accessibility:

- ◆ **One page at a time:** To check the accessibility of the page you're working on, save your document and choose File⇨Check Page⇨Accessibility. The Report window opens with the results.
- ◆ **More than one page, a folder, or the site:** Another way to use the Accessibility report utility is to save your document and choose Site⇨Reports. The Reports dialog box appears (see Figure 9-7).

From this dialog box, you can also choose to run the report on just the page you're working on, several selected pages, a whole folder, or the whole site via a drop-down menu at the top. Then, do the following:

- a. *Select the reports you want to run.*

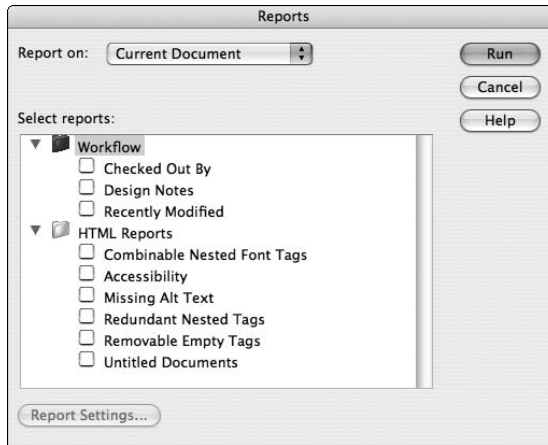
For accessibility testing, enable the Accessibility and the Missing Alt Text check boxes.

- b. *Click Run.*

The Site Reports window opens with the results.



Figure 9-7:
Check
several
pages by
running
reports.



Fixing mistakes

With the results of your report in hand, you can correct mistakes quickly and easily. The Results window shows code line numbers, the severity of the issue, and some brief information about the issue and where you can research it further.

- ◆ **An X:** Issues marked with red Xs are things that have completely failed and need to be corrected.
- ◆ **A ?:** Issues with question marks are things that might or might not be an issue and require human judgment to determine whether corrections need to be made.

An example of this type of error is `Color is not essential`. The report has no way of knowing whether the page is conveying information via color only, so it reminds you to make sure your page makes sense even without the colors.

To view the piece of code that's causing an issue, double-click the issue description's text in the Results window. The code is highlighted for you.

Chapter 10: Publishing the Web Site

In This Chapter

- ✓ **Preparing to launch the site**
- ✓ **Uploading the site**
- ✓ **What to do after the launch**

Designing Web sites is a creative process that can be lots of fun. Fun doesn't put bread on the table or a Web site on your hard drive, though, and it isn't going to pay the bills, either. After you build it, you have to upload it before visitors swarm your client's site. Before that blessed event can happen, make sure you have your ducks in a row. In other words, make sure it ain't broke — and if it is broke, fix it. Nothing is worse than uploading a site with bugs and ending up with egg all over your face. After all: You're the designer, so the yolk's on you. In this chapter, we show you how to exterminate any bugs that might be lurking in your code, how to upload the site to the hosting server, and what to do after the launch.

Look before You Leap: What to Do Before Launch

Your client is all over you like white-on-rice to make the site live, but if you do so before making sure everything is up to snuff, you do yourself and your client a disservice. First and foremost, you damage your reputation, both with the client and potential clients, if you put a site out there that contains code errors, broken links, and so on. In the upcoming sections, we show you a few things you need to consider before uploading the site to the client's server.

Develop a checklist

When you create a checklist that you can refer to before you launch the site, you can be sure you won't forget to double-check anything. You can even use the list for all the sites you launch. Write it once; use it countless times. Like reusable code, reusable lists are a wonderful thing. Your list should include the following items:

- ◆ **Check each and every link.** Nothing is more embarrassing than creating a Web site with links that work on some pages but not on others. You can safeguard against this happening if you create a template that

includes the navigation links, and then use this for each page you create. While you're checking links, make sure you have the target right. If the site has links to pages outside your client's domain, good practice mandates opening them in another window.



If you're using Dreamweaver to create your pages, choose Site⇒Check Links Sitewide. Press Ctrl+F8 (Windows) or ⌘+F8 (Mac), and Dreamweaver lists any broken links in the Results panel.

- ◆ **Does your fancy code work?** If any of the pages in your site are PHP, ASP, or DHTML, make sure each script works as expected.
- ◆ **Does the site load quickly?** Savvy Web surfers are an impatient lot and won't wait for a site to download. If you did your homework and optimized all your images and other interactive content, the site should download in fewer than 12 seconds. And that's 12 seconds for the worst-case scenario in your client's intended audience. If your site loads in 12 seconds for users with a cable hookup but some of the intended audience use DSL, you have to go back to the drawing board.



If you feel that your Web pages are not running on all twelve cylinders, consider using Yahoo's YSlow. This is a tool that is integrated with Firefox's Firebug Web development tool. For more information on YSlow, visit <http://developer.yahoo.com/yslow/>.

- ◆ **Does the site include a call to action?** If your client is selling merchandise or services, Marketing 101 dictates that the site should ask the visitor to do something: a *call to action*. At the very least, the site should include some type of special offer that tempts the visitor into clicking the checkout button. Another call to action might be a form that requests contact information for a mailing list or newsletter.
- ◆ **Is the site easy to navigate?** Make sure that site visitors don't need a PhD to figure out the navigation menu. Try to avoid being cute and designing an *avant garde* menu that uses only icons. Some people will get it, but the ones that don't will be visiting the site of your client's competitor. If your site design uses graphics in the navigation menu, make sure you have redundant text-only links at the bottom of each page, below the fold. Search engine robots use text-only links when indexing the site.
- ◆ **Is the content relevant and easy to understand?** Chances are your client created most of the text for the site and provided images as well. It's your job to put it in a palatable format that visitors can easily digest. Scan each page and make sure the headlines and links provide a message to visitors. This is especially important if the site has a lot of text.



You can break up a lot of content using headlines, bullet points, bold text, white space, hyperlinks, or images. Savvy Internet surfers use these visual clues to quickly find the information they want.

- ◆ **Read each page.** The information provided by your client should pique visitors' curiosity and inform them. The home page should pique visitors'

curiosity and make them want to click through to other pages on the site. The text should also be relevant and peppered with keywords potential visitors will use when searching for sites similar to your client's.

- ◆ **Make sure that each page has a balance of text and images.** Unless you're doing a portfolio page for a photographer, the images on each page should complement and balance the text. Too much of one or the other presents a confusing message.
- ◆ **Is the text easy to read?** Make sure the target audience can easily read the text. Even if your client is a techie, but the visitors are technophobic, make sure the text doesn't include jargon. Write the text for the least-common denominator: in other words, for the person who knows nothing about your client's product or service. If the content doesn't meet this standard, tell your client she needs to dumb it down before the site goes live. Also, make sure that the text font is easy to read and that the font color contrasts well with the background for easy readability.
- ◆ **Are the paragraphs short?** Long blocks of text might make visitors shy away from the site — that trudging through it is too much work. Send long blocks of text back to your client and ask him to cut out anything that isn't relevant. When all else fails, refer your client to a good copy editor.
- ◆ **Are the pages consistent?** Each page should have a common look and feel. The navigation menu needs to be consistent on all pages. If it's not, the visitor might think he's clicked away from the site.
- ◆ **Is the site complete?** In other words, do all or most of the pages have content? It's bad practice to leave a bunch of Coming Soon or Under Construction messages throughout the site. Missing content frustrates both visitors and your client. If you're under a deadline to launch the site by a certain date and the client hasn't given you all the information, tell her it's in her best interest to not launch the site until all pages are complete.
- ◆ **Make sure every image loads.** If you end up with a placeholder with no image, this indicates you might have changed the image's filename or inadvertently moved the image to another folder.
- ◆ **What's above the fold?** The most important (must-see) information on every page should appear above the *fold*. This is the top portion of a Web page that's visible when the page first loads, without scrolling. This is the most important part of the entire page, so use it wisely. At the same time, make sure that no images are cut off by the fold, and also that no part of a paragraph is cut off by the fold. When performing this test, make sure to resize your desktop to that of the intended audience.
- ◆ **How much of the information is below the fold?** If each page of the site has a lot of information that appears below the fold, visitors have to scroll down to access all the information. If this is the case, consider splitting a lengthy page into two or more pages. Alternatively, have your client edit the content.

- ◆ **Test all forms and other interactive content.** When you submit a form, make sure the data goes to the intended destination or is added to the applicable database.
- ◆ **Check the spelling.** Most HTML editors come with a spell checker. Don't be unprofessional and upload a site with typos or bad grammar. If you're in doubt of the correct spelling of a technical term, ask your client or find the correct spelling at a reputable online dictionary.
- ◆ **If the site has options to order merchandise, make sure that transactions can be completed.**

Get opinions

A dozen jurors are used in a court of law. You should have at least that many people rendering judgment on your client's Web site. Big caveat: The people who give their opinions on your design should be totally impartial. In other words, don't use your client's employees or your own when you want feedback on your design. For that matter, don't use your client's family, or even your own, when getting feedback.

Do a beta test

If possible, set up a beta test before going live. When you conduct a beta test, you upload your design into a folder on the server and send the URL to your testers. They provide feedback about site usability, aesthetics, and so on. Beta testing is especially useful when you have a really big site with lots of pages and almost as many links as there are Smiths in the New York City phone book. You'll be a candidate for a good hair-coloring product if you tackle testing a huge site by yourself.

In addition to testing the site for usability, beta testers can find any bugs that are lurking in your code. Identifying usability issues and other problems greatly enhances the chances of the site succeeding when it goes public. The test can also determine how real-world users react to the site.

In a typical beta test, you contact a potpourri of users to try out the site for a few weeks. During that time, you get feedback on the site design, usability, any potential problems, such as broken links or missing images, and any features that are not clear to the testers. During this time, you monitor the server side to make sure that the data is going to the right places. You also monitor any CGI scripts (such as mail forwarding) to make sure that data is being forwarded to the proper parties. During a beta test, you typically forward all data and e-mails to one address.

Your best candidates for beta testers are actually users of the product or service. Your client might be able to supply some beta testers from his customer base. With a good cross-section of the client's customer base or intended audience, the test is more effective. The information you receive

can give you a good idea of site usability and the value of the site from members of your client's intended audience.

Get feedback

Setting up a beta site and enlisting testers is a lot of work. Your beta testers might have a good time filling in the forms, placing mock orders, and checking out the features of the site, but if you don't get feedback, it's an exercise in futility. When you enlist beta testers, tell them what you're looking for. Any group of individuals has varying degrees of initiative, and beta testers are no exception. Expect some of your testers to give you lots of feedback: others, little or none. Keep tabs on who says what, and send e-mails to the people you're not hearing from.



Throughout the course of the beta test, address issues when they come up. If you come across usability issues, tweak your design. Exterminate any bugs as soon as they are reported. After correcting the issue, contact the person(s) who reported the bug and ask them whether your tweak resolved the problem.

E-mail is an excellent way to get feedback, but forms enable the designer to create a set of structured questions. With this type of survey, you can target questions that show how often the tester visits similar sites and how often he uses the product or service. Figure 10-1 shows a mini-survey form for beta testers of a hypothetical photography Web site.

The screenshot shows a Microsoft Internet Explorer window with the following content:

Thank you for your feedback!
To make your feedback even more valuable to us, please tell us yourself.

How often do you purchase photography equipment?
[select one]

How many times have you visited this site, including today?
[select one]

If you have visited in the past, what have you done with our site?

- Viewed portfolios
- Purchased photos
- Read articles
- Downloaded tutorials
- Other

Please select the option which best describes the nature of your feedback:

- Bug report
- Usability problem
- Feature suggestion
- General help question
- Other

What is your question or comment?
[text input field]

Figure 10-1: Send me your answer; fill in a form, mine forever more. . . .

Follow up with beta testers

After the test is over, you have a lot of data that shows you the best features of the site as well as issues that need attention. However, you can gain some closure, so to speak, by doing a follow-up survey. Send an e-mail to all test participants and ask them about their overall experience with the test site — and, most importantly, whether they liked it and found the content relevant and useful.

In addition to the basic questions, find out how often each participant used the site. You can count on the answers being somewhat inflated (numbers inflated 20 to 30 percent would be the norm), unless you have a meticulous tester who logged everything concerned with the test. When you're compiling your results, you know that the answers from the power users carry more weight than those of casual users.

Another set of pertinent questions is how the testers like specific features or areas of the site. Find out how often they used specific sections of the site plus how they liked the navigation system and any interactive features on the site. If the site includes forms and features, such as a search engine, make sure you get feedback on those as well.

Another set of questions should address the issues most reported by testers. The answers to these questions can enable you to find out whether your beta test tweaks took care of the issues.

Finally, ask your testers to rate the site compared with their experience on similar sites. You can ask users to rate the site on a scale from 1 to 10, or ask them to sum up their experience in a short paragraph. Again, you'll weigh the pertinence of each answer based on how often the tester visited the site.

Tweak your design

The responses from your beta testers enable you to perform last-minute tweaks on the site before it goes live. The issues raised by the beta testers might be related to design, content, or both. If it's design, you have to put on your thinking cap to figure out the best way to overcome the issues. If it's content, your client has to belly up to the table and do some editing. The worst-case scenario would be there are too many issues to resolve given the current design. If this is the case, postpone the launch and address the issues, perhaps with a major redesign. If so, you'd be wise to do another beta test before launching.

Going Live

After tweaking the site based on your response from beta testers or from your checklist, you're ready to go live. If you did your test on the server, it's a simple matter of transferring the files from the beta test folder to the

root folder of the Web site. If you did all your testing on your local machine, upload the pages and related files to the server. You can use an application (such as Adobe Dreamweaver) to upload your content, or you can use an FTP client. The FTP client is the application you use to upload files to the Web server. We use a combination of Dreamweaver and the FileZilla FTP client to handle all of our file-transfer needs. We cover both applications in the upcoming sections.

Using Dreamweaver to upload content

Dreamweaver features a sophisticated set of commands that enable you to connect to a Web server and upload files to the server (*Put* in Dreamweaver-ese, as in, *put the files on the server*) or download files from the server (*Get* in Dreamweaver-ese, as in, *get the files from the server*). To upload pages using Dreamweaver, follow these steps:

1. If the Files panel is not visible, choose **Window** ⇨ **Files** to open the Files panel. (See Figure 10-2.) Alternatively, press **F8**.

The Files panel shows the local files associated with the site on which you're working.

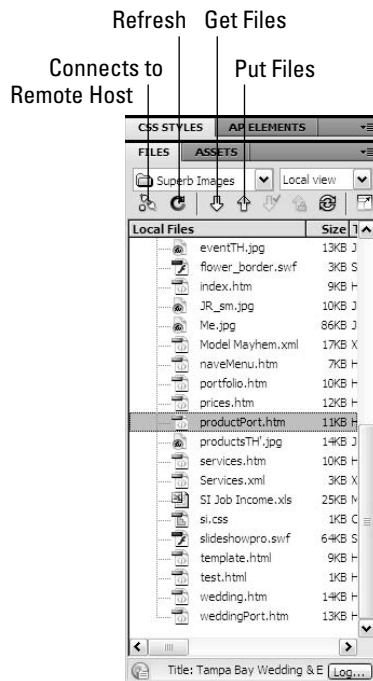


Figure 10-2:
The default layout of the Files panel.

2. Click the Connects to Remote Host button.

Dreamweaver connects to the remote server using the credentials you supplied when you set up the site.

3. Click the Expand/Collapse button.

Dreamweaver increases the size of the panel and splits the panel into two panes. The pane on the left shows the files on the remote server, and the pane on the right shows the files on the local machine. (See Figure 10-3.)

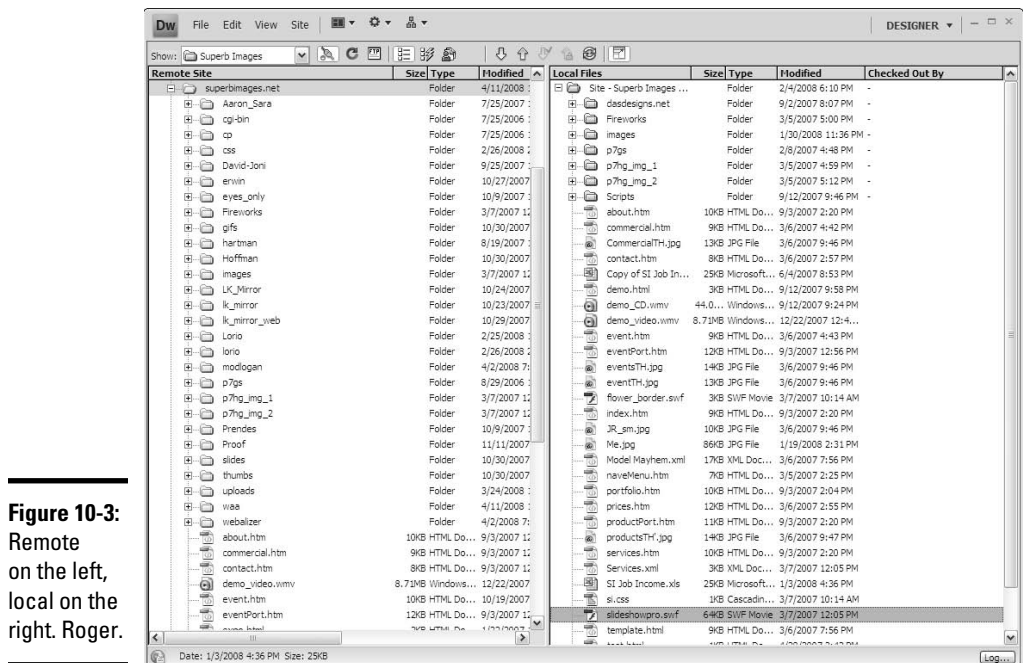


Figure 10-3:
Remote
on the left,
local on the
right. Roger.

4. Select the files you want to upload to the server from the Local Files window.

If you're uploading edited pages, you don't need to select files, such as images, that are associated with the files you're uploading unless they've changed.

5. Click the Put Files button.

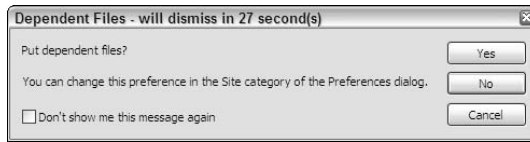
Dreamweaver opens the Dependent Files dialog box. (See Figure 10-4.) This dialog box enables you to upload files associated with the page, such as images and so on.

6. Click Yes.

Dreamweaver uploads the selected files, along with all files associated with the selected file.

Alternatively, click No if you know that the files associated with the file you're uploading have not been changed and no new files have been linked to the file.

Figure 10-4:
Include associated files with the upload, or not.



You can also upload a page you're editing by clicking the File Management icon (looks like two arrows, one pointing up and one pointing down) and then clicking Put. Alternatively, you can press Ctrl+Shift+U (Windows) or ⌘+Shift+U (Mac).

You can also download files from the remote server by following these steps:

1. Choose Window⇧Files to open the Files panel. Alternatively, press F8.

The Files panel shows the local files associated with the site on which you are working.

2. Click the Connects to Remote Host button.

Dreamweaver connects to the remote server using the credentials you supplied when you set up the site.

3. Click the Expand/Collapse button.

Dreamweaver increases the size of the panel and splits the panel into two panes. The pane on the left shows the files on the remote server, and the pane on the right shows the files on the local machine.

4. Select the files you want to download to the server from the Remote Site window.

You don't need to select files associated with the files you're downloading, unless they're not already present on your computer.

5. Click the Get Files button.

Dreamweaver displays the Dependent Files dialog box. Refer to Figure 10-4.

6. Click Yes.

Dreamweaver downloads the selected files into the applicable site's folders on your local machine.

You can also upload a page you're editing by clicking the File Management icon (looks like two arrows, one pointing up and one pointing down) and then clicking Put. Alternatively, you can press Ctrl+Shift+U (Windows) or ⌘+Shift+U (Mac).

Uploading pages with an FTP client

Dreamweaver file management is a wonderful thing. However, if you're working on documents for which you haven't set up a site in Dreamweaver, an FTP client is the ideal solution. An FTP client also comes in handy when you want to download files from a server — for example, when you're doing a site makeover. Quite a few FTP applications are available. We use FileZilla. It's free as of this writing, and is flexible enough to upload an entire Web site. You can download a copy of the FileZilla FTP client from <http://filezilla-project.org>. To upload files using FileZilla, follow these steps:

1. Launch the application.

The FileZilla interface is shown in Figure 10-5.

2. Navigate to the folder where you stored the files you want to upload.

The window on the left side of the interface shows the files on your local machine. Click the down arrow to the right of the current folder window in the Local Site window to access your system's directory structure. The window below Local Site displays individual files.

3. Enter the host URL, username, and password in the appropriate fields.

You don't have to change the default port. Figure 10-6 shows the proper nomenclature for the site URL. Some Web host companies are different. If you enter the URL and are not able to connect, refer to your Web hosting company's technical support.

4. Click the QuickConnect button.

FileZilla connects to the Web site. The status of each action performed by the application appears in the window below the URL and other text fields. After the application connects to the Web server, it displays the remote files on the right side of the interface. Folders have a question mark (?) icon on them.

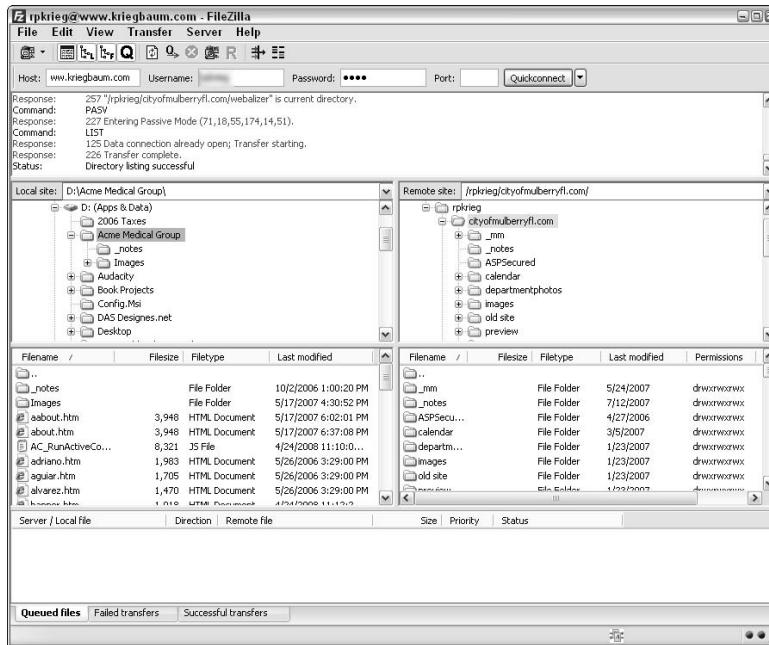


Figure 10-5:
An innocuous interface for an application named FileZilla.



Figure 10-6:
Beam them up, Scotty.

5. Click the folder.

FileZilla reads the contents of the folder. The window below Remote Site displays individual files in the selected folder. (See Figure 10-7.)

6. In the left window, select the files you want to upload.

7. Drag the files into the applicable folder in the right window.

FileZilla uploads the files to the server. If you're uploading edited files with the same name as those on the remote server, a warning dialog box appears telling you that the target file already exists, listing actions you can take. Many times, you want to overwrite a file on the remote server with an updated file with the same name. In some cases, you might want to save the old file in a different folder in case you ever

need to revert to it. When we need to save old files, we use FileZilla to set up a folder called Archive at the Remote Host. We copy the previous version of a file to this folder before uploading the new one. When you upload a file, the progress is displayed in the window at the bottom of the application.

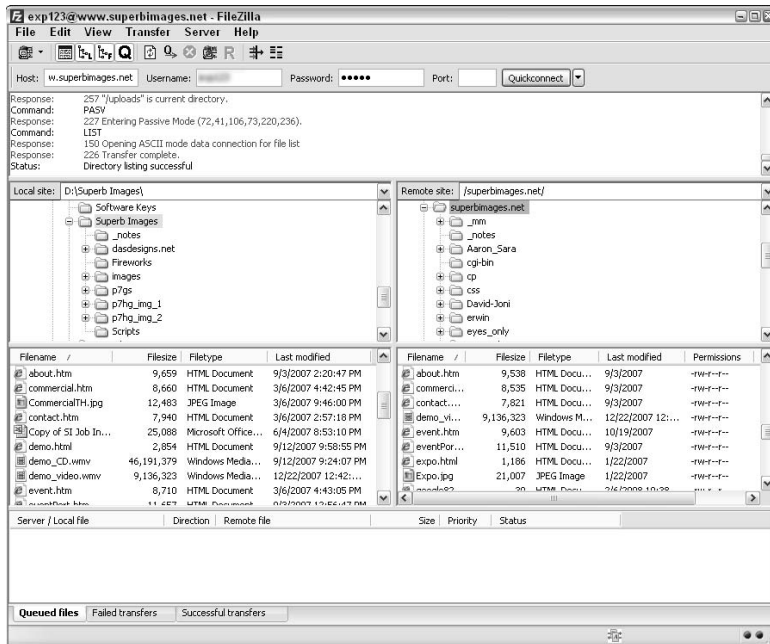


Figure 10-7:
Ready for
lift off! Er,
we mean,
upload.

What Next? The Launch Is Not the End of the Project

If you think you bask in the glory of a job well done and break out the bubbly after the site goes public, think again. Just because you built it, doesn't mean they will come. Given the number of Web sites on the 'Net — everyone and his little brother has one, it seems — your client's site will bask in anonymity unless the site is made visible to the search engines and some marketing is implemented. In the following sections, we discuss briefly what needs to be done to make a site succeed. You can find detailed information on making your client's site succeed in Book VIII, Chapter 1.

Submitting the site to search engines

Search engines are technological marvels. Think about it: All you have to do is type in some words regarding what you're looking for, and a few seconds later, you have hundreds of Web pages to peruse. Search engines aren't mind readers, though. They don't know when you uploaded your client's

squeaky-clean Web site that loads at the speed of sound and is turbocharged with enough interactive bells and whistles to keep visitors entertained for hours. To be on a search engine's radar screen, you have to submit the site.

Submitting the site is like throwing a needle into a haystack, and that needle is going to be hard to find unless you do something to make it stand out. To make your client's site rank high in search engine results pages, you have to optimize the pages for the keywords and key phrases that your client's intended audience is likely to type into the search engine text box to find your client's product or service. You can either optimize the site yourself or hire a company to optimize the site. Site optimization is a black art that warrants a book of its own.

After you optimize the pages for search engines, you're ready to submit the site to the search engines that your client's intended audience is most likely to use. Many search engines are out there. However, you should concern yourself with only the most popular search engines, such as Google and MSN. If (perchance) your client's site caters to a niche market, maybe you can find a search engine dedicated to that market. Your client should be able to tell you whether that's the case.

After you identify which search engines you want to submit to, you can consider several ways of doing that. The topic of search engine optimization and submitting your site to search engines is ever changing, but we cover the current methods of search engine optimization and submission in Book VIII, Chapter 1. (You can read more about this topic in *Search Engine Optimization For Dummies*, 3rd Edition, by Peter Kent [Wiley].)

Implementing marketing

After you leap the hurdle of search engine submission, you still have one hurdle to clear: marketing. Very rarely can a bricks-and-mortar business survive on word of mouth. The same is true of Web sites. Granted, the site can get repeat visitors because a friend e-mailed a friend and so on. To achieve your client's lofty goals, though, marketing has to be implemented.

There are lots of ways to market a Web site. First and foremost, your client should use marketing vehicles he already has in place. For example, the client can add a brief blurb about visiting the Web site in all his print ads. Of course, the URL should be prominently displayed in all ads — and for that matter, on any printed material that leaves the client's place of business.

Traditional marketing techniques work well for a local business in a local market. However, for the site to be truly effective, your client might have to resort to online marketing as well. *Pay per click* is a form of marketing where your client budgets how much money he pays each time a link on a search engine results page is clicked. The client submits a budget, which is the maximum he's willing to pay per month. The frequency with which the

link appears is determined by how your client's budget stacks up against those of competitors who are vying for the same keyword or phrase. These are *sponsored links*. Web site marketing is a heady topic, which we cover in detail in Book VIII, Chapter 1. (Read more about pay per click in *Pay Per Click Search Engine Marketing For Dummies*, by Peter Kent [Wiley].)

Requesting link exchanges

Like sausages, search engines love links. The more sites that link to you client's site, the higher the site will rank in the search engines. The best way to get sites to link to your client's is to submit a request to the Webmaster of a popular site. Tell the Webmaster that your client's site has content that would be of interest to visitors of his site. Ask the Webmaster to please add your client's URL and a description of the Web site (which you or your client have previously prepared) to the site's link pages. The Webmaster might ask you to supply a link in return.

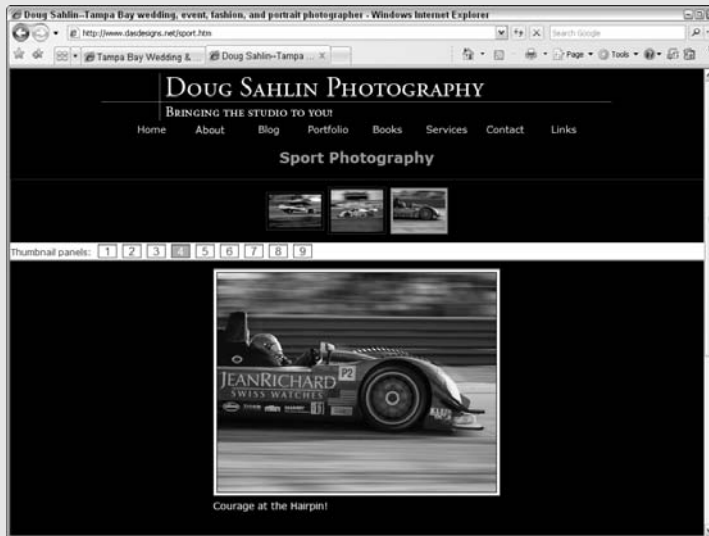
Another method of marketing is link exchanges. You can find link exchange services by typing **link exchange** in your favorite search engine. You'll find hundreds of results from which to choose. Typically, a *link exchange* lets subscribers peruse through their directory and choose the sites they'd like to link to, and vice versa. The request is submitted to the site Webmaster, who chooses whether to accept the request and ask for a cross-link. Some link exchange services are free. However, just as there is no free lunch, a free link exchange service generally comes with strings attached. Before signing up for a free link exchange service, read the fine print and make sure you can live with the service's agreement. For example, you may be required to add a link to the link exchange service's Web site. You should also make sure that the links propagated by the link exchange are from relevant sites — in other words, sites that relate to your client's business.



If the service randomly generates links to nonrelevant sites, you run the risk of being blackballed from major search engines. Do your homework before agreeing to use the service. You might ask for the URLs of other sites using the service and then contact the Webmasters to see if they're happy with the service.

Book IV

Web Graphics



Displaying photos online.

Contents at a Glance

Chapter 1: Finding Inspiration	315
Finding Fresh Ideas	315
Respecting the Copyrights of Others.....	316
Protecting Your Own Copyrights	318
Working with Colors (Web Safe versus Not-Web Safe).....	319
Creativity Tools — Color Charts and More.....	322
Font/Type Issues on Web Sites	323
Print to Web: Making Your Web Site Work with Existing Materials.....	324
Understanding Image Formats for Web Design	325
Chapter 2: Bringing Your Vision to Life	329
Introducing Fireworks CS4	329
Creating Art with Other Tools	340
Chapter 3: Workhorse Graphics	345
Creating Buttons in Fireworks	345
Creating a Navigation Menu	351
Creating a Pop-Up Menu	354
Recycling: Reuse Everything.....	359
A Doodle to a Working Page: Concept to Completion	361
Optimizing Artwork in Fireworks	362
Exporting Artwork from Fireworks	366
Chapter 4: Creating Compelling Graphics	367
Creating Banner Graphics	367
Create Promotional Graphics	371
Creating Fast-Loading, Beautiful Photos	374
Creating a Photo Gallery.....	385

Chapter 1: Finding Inspiration

In This Chapter

- ✓ Finding inspiration
- ✓ Understanding copyright issues
- ✓ Getting advice and tools for using colors
- ✓ Considering font issues
- ✓ Dealing with image formats: GIF, JPEG, and PNG

When a client hires you to design a Web site, he expects something original and unique. This can be a difficult task. After you pump out a few hundred designs, you have a tendency to rely on old habits and create sites that look similar. And the task is doubly hard because there are really only 27 original ideas. (Everything else is just a subset of the original 27.) It's your interpretation of the original idea that makes the site unique. To make matters worse, your client wants his original Web site yesterday. So the trick is to be original without spending a whole lot of time doing it. In this chapter, we show you a few ways to feed your creative muse.

Finding Fresh Ideas

Before you search for ideas, get a handle on your client's likes and dislikes. When you initially interview a client, we recommend that you always ask for the URLs of your client's competitors' sites as well as those sites she likes. This can give you an idea of your client's tastes. The client's business and printed material can also give you some ideas. Then, there's the matter of the client's intended audience. If you know the demographics of your client's intended audience, you know their likes and dislikes. If you're dealing with a large company, it probably already has this information. Or perhaps, you and your client can create a persona to define the likes and dislikes of your client's intended audience. (For more information on personas, see Book I, Chapter 4.)

Another factor you have to consider is the client's industry. Chances are that the sites associated with your client's industry have a similar look and feel. Stretching the envelope is okay, but it's probably not a good idea to venture outside the envelope. Therefore, your first method of looking for ideas should be to use your favorite search engine — enter keywords that relate to your client's business. The top-ranking hits you get (sites) might

not be the most inspirational, so randomly pick a few sites from the first few pages of results. Bookmark the ones you like. Let your creative muse stir the porridge for a day or two, and then go back to your bookmarks and make notes of what you like.

After you peruse your bookmarked sites, you might still need additional inspiration. If so, find some portals that link to exemplary Web sites. One way to find URLs to inspiring Web sites is to type something like **cool Web sites**, **100 best Web site designs**, or **best type of business Web sites** in your favorite search engine. You'll come up with a few places to look for inspiration. Here are a few you can try:

◆ ***Time Magazine's 50 Best Web Sites 2007***

www.time.com/time/specials/2007/article/0,28804,1633488_1639316,00.html

◆ ***PC Magazine's Top Web Sites 2007***

www.pcmag.com/category2/0,1874,7488,00.asp

◆ ***Web Marketing Association WebAward, Best Real Estate Web Sites***

www.webaward.org/winners_detail.asp?yr=all&award_level=best&category=Real%20Estate

◆ ***Entertainment Weekly Magazine's 25 Best Music Web Sites***

www.ew.com/ew/article/commentary/0,6115,1195793_4_0_,00.html

◆ ***Planetizen Top 10 Web Sites of 2007***

www.planetizen.com/websites/2007

If your search lands you on a site that looks like nothing but advertisements, it probably is. Pick and choose the portals that look like the real deal. Another great source for inspiration is the Adobe Customer Showcase (www.adobe.com/cfusion/showcase/index.cfm). Here, you'll find inspiring designs that were created using Adobe (and the company Adobe acquired: namely, Macromedia) software.



We're not suggesting you blatantly rip off someone else's design, but rather that you look at as many Web sites as you can. Incorporate those ideas that you like with your own sense of design. By doing this, you'll take your design skills to the next level and give your client what he wants: a unique Web site.

Respecting the Copyrights of Others

It's one thing to mix and match ideas from other designers with your own. However, blatantly copying someone else's design isn't creative. In fact, the copyright of objects, such as photographs and text, is owned by the creator.

Web sites also fall into this category. Some people might be tempted to steal an image from a Web site, thinking they'll never get caught. However, always remember that as soon as an image or block of text is saved to a hard drive for the first time, that creator retains copyright to the image. Some images might fall under the Fair Use Clause of copyright law.

However, taking an image (without permission) from one Web site for use on a commercial Web site can hardly be considered fair use. If a Web site owner does give you permission to use his image, make sure you get that permission in writing. Covering thine own posterior is better than dealing with legal fees and a "he said, she said," knock-down, drag-out situation. If your client needs photographs of products for the Web site, have her contact the manufacturer and ask for images you can incorporate in the design. Most major manufacturers can provide you with a URL to a page from which you can download high-resolution images.

If you or your client needs an image that's perfect for your design, consider licensing one from a local photographer or purchasing licensing rights to a photo from a stock agency, such as Comstock Images (www.comstock.com) or Jupiterimages (www.jupiterimages.com). You can also set up an account with a stock photography firm. After setting up an account, you can peruse the agency's collection and find the ideal image for a Web site. iStockphoto (www.istockphoto.com; see Figure 1-1) is one site where you can find reasonably priced images.

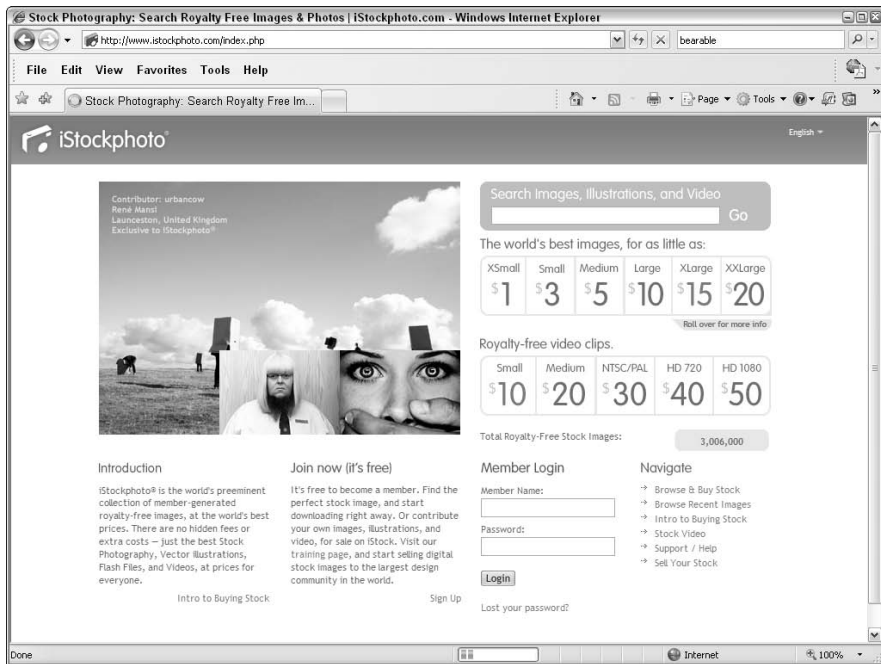


Figure 1-1:
Searching
for the
perfect
image.

Stock photo sites operate differently, and some even have different methods for purchasing images. You may be able to purchase a yearly membership, buy single images as needed, or purchase points that are then used to purchase individual photos. When you purchase a photo, you purchase the rights to use a photo for a certain destination. The price of the photo varies depending on the size and resolution of the photo, and whether you're using it on the Web, in printed media, and so on. Many stock houses charge a higher price if more people see the images. The details pertaining to photo usage are generally easy to find on each site or are listed alongside the photo.

And don't forget about logos. Logos also fall under the copyright laws. You might think it's perfectly okay to take a logo from a manufacturer's Web site and use it in your design because your client is selling that manufacturer's product. However, most manufacturers restrict the use of their logos. To legally use a logo, your client has to accept the manufacturer's terms of use and agree to some kind of licensing. Some manufacturers won't let you alter a logo in any way. For example, you can't get creative in Photoshop with the MSNBC logo and apply it to an image to make it appear as though it was part of a scene or embroidered on an article of clothing.



If you're designing a lot of sites that do need photos or other elements that you and your client can't supply, consider buying a clip art package of royalty-free images. When you purchase a package like this, you're free to use the image wherever you want. You can find clip art packages at your local office supply store or on the Web.

Protecting Your Own Copyrights

You can't copy or use content created by another Web designer without permission. Period. However, the same protection applies to the content that *you* create for your client. You own the copyright to all the graphic elements you create for the design, such as the interface and buttons, as well as the layout of the page. However, the content you create is often mingled with other content, such as images and text created by the client. Therefore, who owns the copyright to what is kind of a gray area. As the Web designer, you can claim copyright to the content you create for your design by including the following statement on the bottom of each Web page:

Web Design Copyright © 2009 *your name or company name* All Rights Reserved

You should copyright your client's original material with the following statement:

Content Copyright © 2009 *your client's name* All Rights Reserved



In Adobe Dreamweaver, you can create a Library item for the Web-design and content copyright information. When the year changes, you change the copyright year in the Library item, and all site pages are updated. Upload the revised files to the server, and your copyright is covered for the new year.



This book is written by Web designers, not legal experts. The information presented in this section should not be construed as advice from legal counsel. To fully protect your material and your client's material, you should contact a lawyer whose area of expertise is intellectual property.

Working with Colors (Web Safe versus Not-Web Safe)

When you create a Web site, you strive to create an aesthetically pleasing blend of images, text, and graphics, such as banners and navigation menus. Your design must be harmonious: a careful blend of what the client supplies you and what you create. If your client supplies you with images for the site, you can often choose a pleasing color palette by sampling colors from one of the images. If your client has a logo that readily identifies the business, you might be able to create a pleasing color palette by sampling colors from the logo.

Limit your design palette to four colors. Figure 1-2 shows a Web site with colors and design elements chosen from the client's logo. It's hard to tell in the black-and-white reproduction of the page, but if you go to www.antonioswinterhaven.com, you can see that the site uses a harmonious selection of burgundy and black.



Figure 1-2: Choosing a color palette from a client's logo.

When you design a Web site, the colors must be specified in hexadecimal format, which contains six characters. For example, white is hexadecimal #FFFFFF, and black is hexadecimal #000000. The first two characters comprise the red color component of the RGB (red, green, and blue) color model, the third and fourth characters are the green component, and the fifth and sixth are the blue color component. To designate 256 hues of a color in hexadecimal format, you specify a value from #00 to #FF. Therefore, pure red is #FF0000, pure green is #00FF00, and pure blue is #0000FF.

Common sense tells you that text color is very important. If you create a site with text that's hard to read, you're not doing yourself — or your client — any good. If you take the Santa Claus approach and slap red text on a green background, you have a recipe for disaster. Traditionally, Web sites have used black text on a white background. Some designers use a shade of gray for text (for example, #333333 or #666666). If you go any lighter than the latter, though, text is hard to read.

The Web-safe palette, which has been around since the early days of the Internet, consists of 216 colors that any video card can display, on any monitor, by any platform. The Web-safe color palette is the default color palette of applications such as Adobe Dreamweaver, Adobe Fireworks, and Adobe Flash. The Web-safe palette is shown in Figure 1-3. We know what you're thinking — that figure is in black and white (well, actually grayscale). If you want to see the Web-safe palette in live, living, and glorious color, create a new Web page in Dreamweaver. Before adding anything to the page, open the Property inspector, click Page Properties, and then click the Background color swatch. Walla, the Web-safe, 216-color palette appears.



Most modern computers have video cards and monitors capable of displaying millions of colors. Therefore, sticking with the Web-safe palette isn't imperative. If you like to err on the side of caution, most Web applications enable you to create a reasonable facsimile of the desired color by picking its closest counterpart from the Web-safe palette.

Many designers are using pure-white text (#FFFFFF) on a black or dark gray background. If you use this color palette, increase the line height to a percentage of 130 to 150 percent of the text. This increases the space between lines and makes the text easier to read. Figure 1-4 shows a Web page with white text with a line height of 150 percent on a gray background.

000000	000033	000066	000099	0000CC	0000FF
003300	003333	003366	003399	0033CC	0033FF
006600	006633	006666	006699	0066CC	0066FF
009900	009933	009966	009999	0099CC	0099FF
00CC00	00CC33	00CC66	00CC99	00CCCC	00CCFF
00FF00	00FF33	00FF66	00FF99	00FFCC	00FFFF
330000	330033	330066	330099	3300CC	3300FF
333300	333333	333366	333399	3333CC	3333FF
336600	336633	336666	336699	3366CC	3366FF
339900	339933	339966	339999	3399CC	3399FF
33CC00	33CC33	33CC66	33CC99	33CCCC	33CCFF
33FF00	33FF33	33FF66	33FF99	33FFCC	33FFFF
660000	660033	660066	660099	6600CC	6600FF
663300	663333	663366	663399	6633CC	6633FF
666600	666633	666666	666699	6666CC	6666FF
669900	669933	669966	669999	6699CC	6699FF
66CC00	66CC33	66CC66	66CC99	66CCCC	66CCFF
66FF00	66FF33	66FF66	66FF99	66FFCC	66FFFF
990000	990033	990066	990099	9900CC	9900FF
993300	993333	993366	993399	9933CC	9933FF
996600	996633	996666	996699	9966CC	9966FF
999900	999933	999966	999999	9999CC	9999FF
99CC00	99CC33	99CC66	99CC99	99CCCC	99CCFF
99FF00	99FF33	99FF66	99FF99	99FFCC	99FFFF
CC0000	CC0033	CC0066	CC0099	CC00CC	CC00FF
CC3300	CC3333	CC3366	CC3399	CC33CC	CC33FF
CC6600	CC6633	CC6666	CC6699	CC66CC	CC66FF
CC9900	CC9933	CC9966	CC9999	CC99CC	CC99FF
CCCC00	CCCC33	CCCC66	CCCC99	CCCCCC	CCCCFF
CCFF00	CCFF33	CCFF66	CCFF99	CCFFCC	CCFFFF
FF0000	FF0033	FF0066	FF0099	FF00CC	FF00FF
FF3300	FF3333	FF3366	FF3399	FF33CC	FF33FF
FF6600	FF6633	FF6666	FF6699	FF66CC	FF66FF
FF9900	FF9933	FF9966	FF9999	FF99CC	FF99FF
FFCC00	FFCC33	FFCC66	FFCC99	FFCCCC	FFCCFF
FFFF00	FFFF33	FFFF66	FFFF99	FFFFCC	FFFFFF

Figure 1-3: The Web-safe, 216-color palette.

SUPERB IMAGES

CREATING SUPERB IMAGES OF MEMORABLE MOMENTS



Home About Portfolio Services Contact

Commercial Photography

A picture is a thousand words, especially when the picture pertains to your business. Our experienced photographers can capture the true essence of your products, your place of business and your employees. The images we create can be optimized for print and for the web. We can deliver your photos in a variety of formats, including high-resolution print-ready files and web-optimized versions for your website.

Figure 1-4: Increase the line-height percentage when you use white text on a black or gray background.

Creativity Tools — Color Charts and More

When you design a site from scratch, visualizing what colors will look like can be difficult. Fortunately, you can easily experiment with different colors in Fireworks or Dreamweaver when you lay out the elements of your design, such as the banner, navigation menu, text, and so on. But when you're under the gun to create a site quickly, it helps to have a bit of inspiration or a tool to help you in your quest. The following list shows some books you can use to inspire your creativity, aid you in choosing colors, and show you some inspiring design ideas:

- ◆ ***Creativity for Graphic Designers: A Real-World Guide to Idea Generation*, by Mark Oldach (North Light Books):** This book isn't filled with eye candy. Instead, it guides you through processes you can use to come up with ideas. The author also explains the process of creating a good design.
- ◆ ***Artist's Way: A Spiritual Path to Higher Creativity*, by Julia Cameron (Tarcher):** The book was written in 1992, yet it still contains a wealth of information on being creative. The author leads you through a comprehensive, 12-week course on recovering your creativity from things such as limiting beliefs, self-sabotage, jealousy, and other things that deter your creativity. The 10th anniversary edition of this popular book is available at most booksellers.
- ◆ ***How To Think Like Leonardo da Vinci: Seven Steps to Genius Every Day*, by Michael Gelb (Dell):** This insightful book lists seven principles found in Leonardo da Vinci's thinking and creative process and shows how to tap into these principles and use them in your own work.
- ◆ ***Color Index: Over 1100 Color Combinations, CMYK and RGB Formulas, For Print and Web Media*, by Jim Krause (How Design Books):** Keep this handy reference on a bookshelf near your computer. The book is divided into several sections that show color combinations for print and Web. Each color combination shows a graphic using the colors and swatches that list the components needed to create the color using the CMYK and RGB color models. The swatches in the Web design section are designated in hexadecimal format and the RGB color model.
- ◆ ***Color: A Course in Mastering the Art of Mixing Colors*, by Betty Edwards (Tarcher):** This book delves into color theory. Although much of the information is geared towards artists, you can use the information to create harmonious colors for your Web designs. The author gives you an understanding of primary, secondary, and tertiary colors; and shows you how to use the color wheel to understand color values and intensity.



If you're really color challenged, consider purchasing an application such as ColorSchemer (www.colorschemer.com). ColorSchemer features a color wheel and a Photo Schemer section, which enables you to create a color palette using colors from a photo. As of this writing, Color Schemer Studio retails for \$49.99.



You can also find some interesting color schemes at Adobe's Kuler Web site (<http://kuler.adobe.com>). Here you'll find color combinations created by Web designers and graphic artists. If you have an Adobe ID, you can log in and create your own color combinations by modifying an existing one, or uploading a photo as reference.

Font/Type Issues on Web Sites

If you've been designing Web sites for a while, you know that not everyone has the same fonts on their computers that you have on yours. When you design a site, you must take this into consideration. If you design a site with a nonstandard font that's not installed on the client's machine, the page will look fine to you, but it won't display like you designed it. Several fonts are commonly used for Web sites because they can be used on Web sites that are displayed cross-platform. Your universal font friends are Arial, Times New Roman, Courier New, Courier Mono, Helvetica, and Verdana.

In addition, you can specify serif or sans serif as the font. *Serif* refers to the decoration at the end of strokes that make up letters or characters, such as the diagonal at the end of the angular ascenders of a capital A. When a font is sans serif, there is no decoration at the end of strokes or letters. Figure 1-5 shows a sans serif and a serif font.

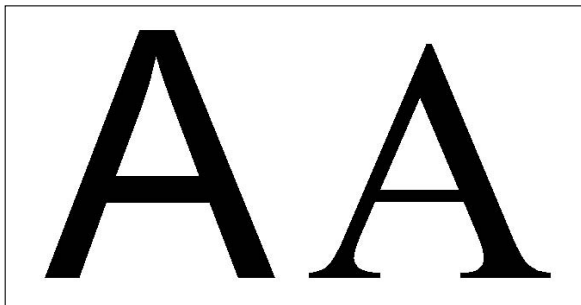


Figure 1-5:
To serif, or
not to serif?
That is the
question.

When you create Web pages and specify a font, you can specify the default and alternate fonts for a block of text. If the user doesn't have the default font installed on her machine, the HTML page displays one of the alternate

fonts. Fortunately, you don't have to hand code the alternate fonts. When you use an application like Dreamweaver, you can choose a set of fonts from a pop-up menu. (See Figure 1-6.)

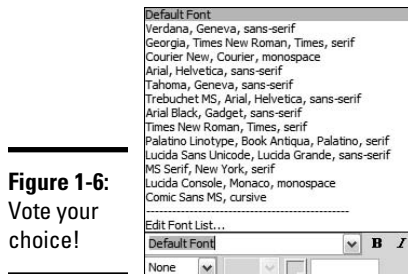


Figure 1-6:
Vote your
choice!

Just when you think it's safe to make an educated choice, there's another issue to contend with: Web browsers. When you use *pixels* (the default unit of measure for text), your text looks fine in the Internet Explorer Web browser. However, if you view the same page in Mozilla Firefox, the text appears smaller. The solution is to use points as the default unit of measure for text. Then your text will display correctly in all browsers.



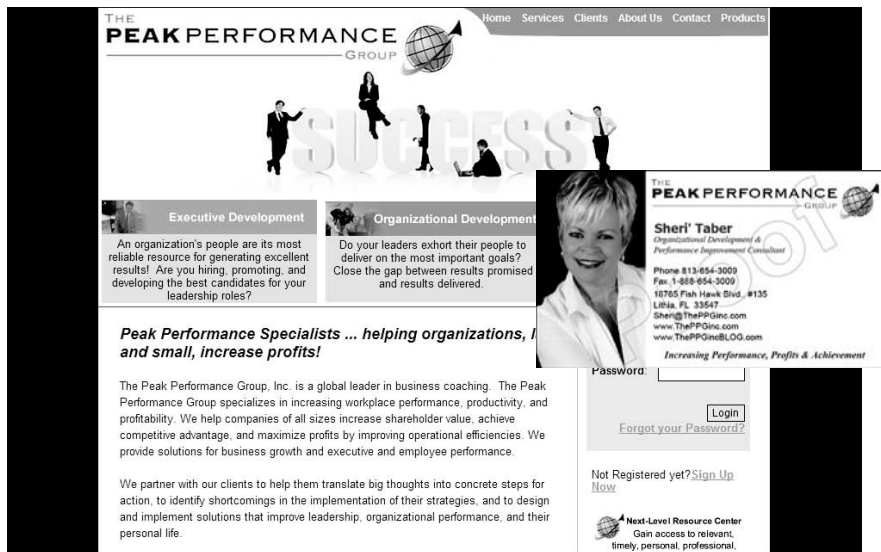
Use a CSS (Cascading Style Sheet) to specify the parameters for text. This saves you from individually formatting each block of text you add to a Web page. For more information on using CSS to specify text, see Book III, Chapter 3.

Print to Web: Making Your Web Site Work with Existing Materials

What came first, the Web site or the bricks-and-mortar business? In most instances, the bricks-and-mortar business. When this is the case, you need to incorporate as many elements as you can from the client's bricks-and-mortar business on his Web site. The obvious way to do this is to use the client's printed materials as a prototype for your Web design.

When you're thinking about design elements for your client's site, start with the client's logo. That needs to be displayed prominently on every page of the site. If the client has a letterhead with his logo, you can use this as the basis for the site banner. If the client doesn't have a letterhead, use elements from her business card. Figure 1-7 shows a business card for a business coach and the Web site that was designed using elements from the business card.

Figure 1-7:
A Web site design using elements from a client's business card.



Understanding Image Formats for Web Design

When you use images on a Web site, you have two things to consider: the image format and optimization. When you get all these factors spot on, you have a crisp-looking image that loads quickly and looks the same in any Web browser. If you get it wrong, you either have an image that loads at the speed of light but looks horrible, or you have an image that looks absolutely gorgeous but takes three forevers to load. In the upcoming sections, we discuss the image formats used in Web designs plus the uses and limitations for each.

Using the GIF image file format

An image saved as a GIF (Graphics Interchange Format) file comprises 256 (8-bit) colors. Images saved with the GIF format can be viewed on all current Web browsers. The GIF (pronounced *giff* or *jiff*) image format is commonly used for images with large areas of solid color, such as logos, or graphic symbols, such as buttons. This file format also works well when the banner is predominantly text. The banner image for the Web page in Figure 1-8 is a GIF image.

The resulting file size of the image depends on how you optimize the image. When you *optimize* a GIF image, you specify the number of colors with which the image is saved and the color reduction algorithm. The goal is to optimize the image so that it still looks good and loads quickly. Find out more about image optimization in Chapter 4 of this minibook.



Figure 1-8:
A GIF image
at work as a
banner.

GIF images also support transparency. This option is useful when you're displaying text over a background image. When you create the image, use a background color similar to, or the same as, that of the background image for the Web page.



Don't use the GIF format to save photorealistic images with millions of colors. The restrictive color palette degrades the image quality. If you use *dithering* (where colors from the 256 color palette are mixed to create a reasonable facsimile of a color not in the palette), the file size is too large for practical use in a Web page.

Using the JPEG image file format

The JPEG (Joint Photographic Experts Group) file format is ideally suited for photorealistic images with millions of colors. The original JPEG (pronounced *JAY-peg*) format also has a derivative known as *JPEG 2000*, which features better image compression that results in smaller file sizes. Unfortunately, JPEG 2000 is not supported by all browsers. The JPEG format does its magic by compressing images. When you compress an image, data is lost; therefore, the JPEG format is known as a *lossy* format. Figure 1-9 shows a JPEG image from a photographer's Web portfolio.



Figure 1-9:
The JPEG
format
is ideally
suited to
photographs.

The image quality is determined by the amount of compression you apply when saving the image. The ideal compression depends on the amount of detail in the image and the size of the image in the Web page in which you'll insert it. If the image is small without much detail, you can apply higher compression to the image. If the image takes up a large portion of the Web page and contains a lot of detail, you'll have to apply less compression to the image. Otherwise, the detail will be muddy, and the image won't be crisp. Fortunately, most image-editing applications, such as Fireworks and Photoshop, have options that enable you to compare the original image side-by-side to a version of the image with compression applied. The details about image optimization are in Chapter 4 of this minibook.



If you have an image with a lot of text, saving the image in the JPEG format will result in text with jagged edges. This is because of the text anti-aliasing blending the edges of the text with the surrounding image. The best way to overcome this is to use an image-editing application (such as Fireworks) that enables you to slice an image into sections and then save them in different formats. Save the sliced sections with text in the GIF format and the rest of the image as JPEG files. When you optimize the image, make sure you specify enough colors; otherwise, you'll see jagged edges where the anti-aliasing blends the font to the background.

Using the PNG image file format

The PNG (Portable Network Graphics) format was originally designed to replace the GIF format. For graphics such as buttons and banners, you can save a file using PNG-8 format, which results in an 8-bit file that supports transparency. When you're working with photorealistic images with millions of colors, you can save the files using the PNG-24 format, which saves the file with millions of colors.

The best feature of the PNG (pronounced *ping*) format is that it uses *lossless compression*, similar to the LZW (Lempel-Ziv-Welch) compression used when compressing files saved as TIFFs (Tagged Image File Format). This results in a good-looking graphic with a relatively small file size. The drawback to the PNG format has been differing levels of support by popular Web browsers. Current browsers offer better support for the PNG format, but there are still issues when PNG files saved with transparency are displayed.

Chapter 2: Bringing Your Vision to Life

In This Chapter

- ✓ **Exporting graphics with Fireworks CS4**
- ✓ **Understanding graphics files**
- ✓ **Getting to know the Fireworks interface**
- ✓ **Discovering some alternatives: Photoshop CS4 and Corel Painter**

You have a client who needs a Web site. The client has a vision, which might or might not be possible. After you become the voice of reason, you have a workable vision that you can translate into an effective Web site. Now, all you need to do is start fleshing out the design. But before you can assemble the HTML pages, you have to take your grand vision and translate it into useable graphics that you can implement in your Web design. In this chapter, we show you the tools you can use to go from concept to completion.

Introducing Fireworks CS4

You can use lots of applications to create the graphic elements of your Web design. However, if the elements are exported only as image files, you have to assemble them yourself in your handy-dandy HTML editing application. And that means working with tables and images and spacers, oh my! In other words, the task won't be a whole lot of fun. However, some applications can export your graphic design as images and HTML, and you don't have to write the first bit of code. Bingo! That's just the ticket! This section introduces you to Adobe Fireworks CS4, which is an application you can use to export a graphic design as images and HTML. As of this writing, the Adobe Creative Suite has not been released. If Adobe follows suit with CS3, Fireworks CS4 will be part of Adobe Creative Suite CS4 Master Collection, Adobe Creative Suite CS4 Web Design Standard, and Adobe Creative Suite CS4 Web Design Premium.

Vector and raster images

Fireworks has been around for a long time, albeit not quite as long as the Internet. *Fireworks* is an application that enables you to work

with vector and raster graphics to create the graphic elements for your Web design. Vector graphics give you the power to edit and resize to your heart's content.

Vector graphics are generated by using math to determine where the points and line segments that comprise paths appear within a graphic. When you resize a vector graphic, mathematical formulas are used to regenerate the graphic. It all adds up. Figure 2-1 shows a vector graphic in Fireworks. Notice the points. The points have what look like handles: These are *curve points*. The handles are *tangents*, which are used to reshape the curve segment between adjoining points.

Figure 2-1:
Vector
graphics; it
all adds up.



Raster graphics are also known as *bitmaps*. These rascals are composed of dots of color — *pixels*. You also have something known as *resolution*, which shows how many pixels per inch (ppi) are used to create an image. Monitor resolution is 72 or 96 ppi. When you create a graphic for print, the ideal resolution is 200 ppi or greater. Now, if you're good at math, you can see the problem when you enlarge a raster graphic. Say, for example, you want to increase the size of an image from 4 inches by 5 inches to 8 inches by 10 inches. You're still working with the same number of pixels per inch; therefore, you're asking the application to double the size of each pixel. The math adds up, but the end result is not very useful, and usually not very pretty. The top image in Figure 2-2 shows a small JPEG image; the bottom image shows the same image magnified several times. Notice that you can actually see blocks of pixels.

You use a combination of raster and vector graphics to create your Web designs in Fireworks. In fact, in Fireworks, you can flesh out the entire Web page, leaving placeholders for your HTML text. We use Fireworks to create site banners and navigation menus. The application has powerful tools to optimize graphic images.



Figure 2-2:
I've got
pixels. Who
could ask
for anything
more?

The PNG file format

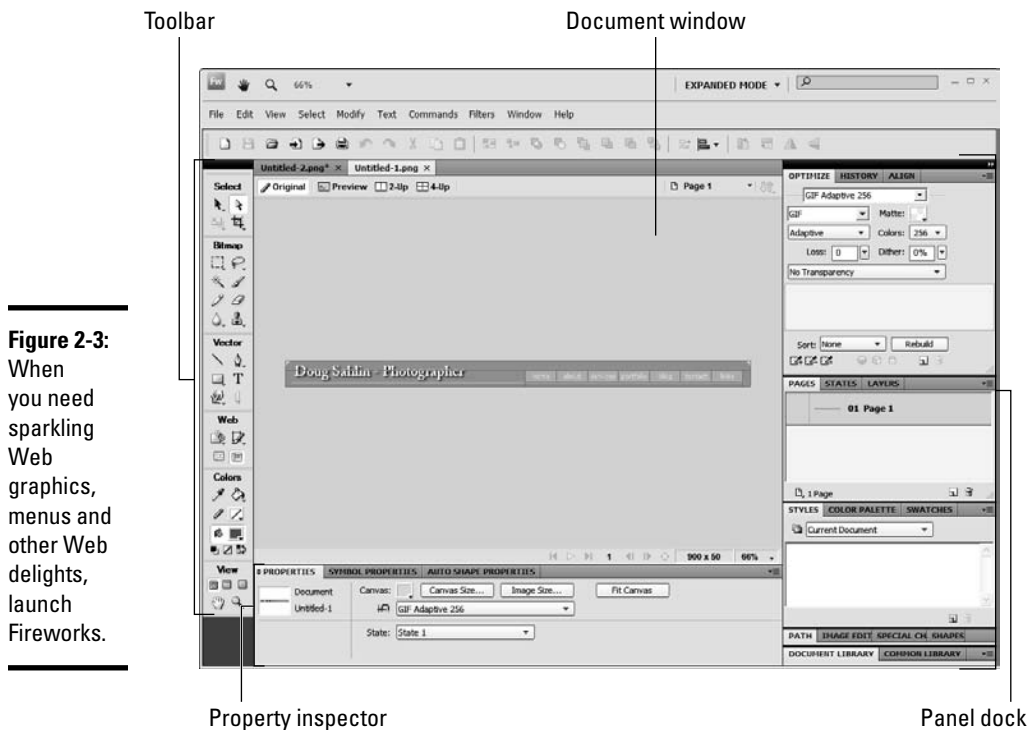
Fireworks uses the PNG (Portable Networks Graphics) file format as its native file format. However, it's not the same format as when you export an image from Photoshop using the PNG format. The Fireworks PNG format saves all the layers and elements used to create the document. The Fireworks file also has a Web layer, which contains information such as the slices and hotspots you created. Slices and hotspots are used to add interactivity to a document. When you export the document, Fireworks takes all the pieces you sliced and diced and houses them in a table. The resulting export is an HTML document and images.

Fireworks interface mini-tour

Fireworks CS4 is an incredible application. You can use it to create all the graphics for your Web designs. You can use the application to create complex interfaces, Web banners, navigation menus, and so on. If your client has more content than will fit comfortably on a navigation menu, you can create drop-down menus that — well — actually drop down. But you get the gist; you can use Fireworks to create some very cool stuff for your Web designs. In the upcoming sections, we show you the Fireworks workspace.

The Fireworks work area

Fireworks might look like other image-editing applications you have used. However, Fireworks has a few other elements that are specifically related to Web design. For example, in the Layers panel, you find a layer labeled Web Layer. You also find an object called the Button Editor, which (you guessed it) enables you to create buttons. Find out more about buttons in Chapter 3 of this minibook. Figure 2-3 shows a navigation menu being created in Fireworks.



The Fireworks interface consists of a main window where you assemble the elements of your design, the ever-present menu bar, a toolbar, and the window in which panels are docked. In fact, the panel dock has almost as many panels docked as there are Smiths in the local phone book. The Property inspector has different options depending on the object you've selected. When you're creating text, the Property inspector is configured as shown in Figure 2-4.

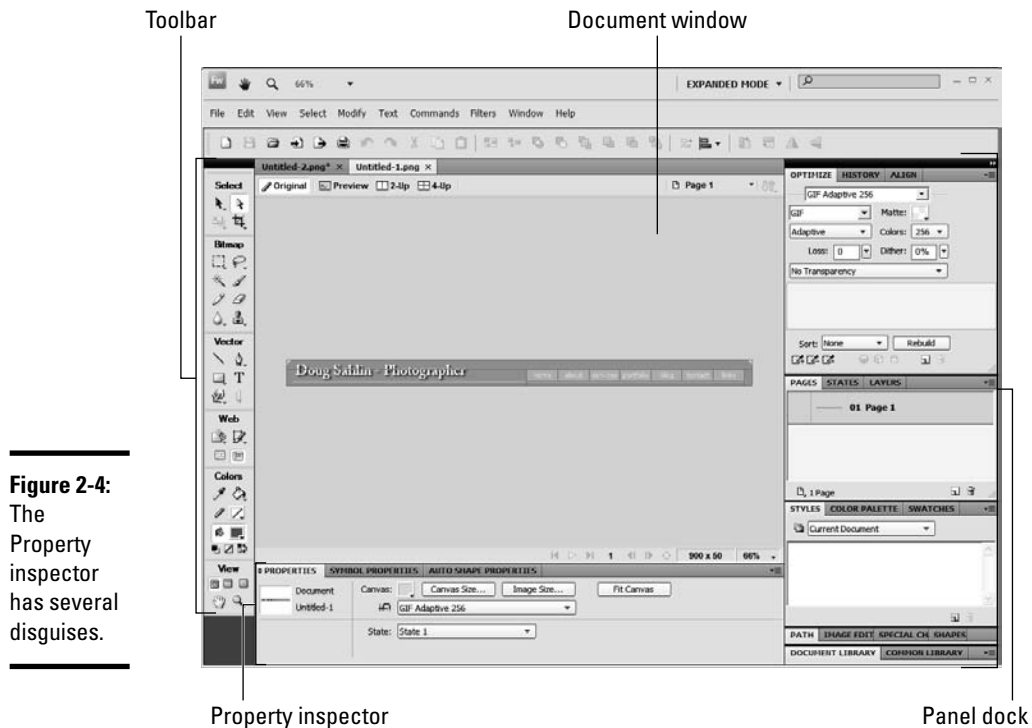


Figure 2-4:
The Property inspector has several disguises.

The Fireworks toolbar

Fireworks has a wide array of tools that you use to create and manipulate objects for your Web designs. The toolbar shown in Figure 2-5 houses the following tools.



When multiple tools reside on a toolbar, the last-used tool is at the top of the heap. When you click the down-arrow, the other tools appear to fly out of the last used tool, hence the term *fly-out menu*.

- ◆ **Pointer:** Use to select and move objects. This tool has a cohort in crime — the Select Behind tool — that hangs out on the fly-out menu. The tool is used to select objects that are hidden partially behind other objects.

- ◆ **Subselection:** Use to select and move objects within a group or on points along a path.
- ◆ **Scale:** Use to resize objects and rotate objects. You can resize objects proportionately by dragging a corner point, change the width by dragging a point in the center of either side, or change the height by dragging a top or bottom center point. When you move your cursor toward a center or corner point and it changes to a curved arrow, you can rotate the object. This is the default tool for this little piece of real estate on the toolbar. You can access three other tools by clicking the down-arrow at the lower right of the Scale tool icon. They are
 - *Skew:* Use to slant objects; you know, get them off kilter. You can also use the Skew tool to rotate an object. If you really get miffed at an object, you can tell it, “Skew you and rotate!” And then you can do it.
 - *Distort:* Use to change the shape of objects. With this tool, you click and drag individual points to morph an object into something totally different than you started with.
 - *9-Slice Scaling Tool:* This tool displays nine slice guides within a selected object.

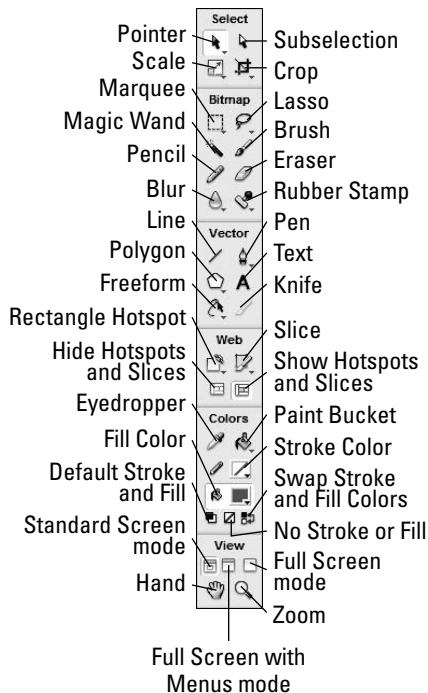


Figure 2-5:
Got a whole lotta tools.

- ◆ **Crop:** Use to trim the document to a smaller size. Click and drag inside the document to create a cropping rectangle. While you drag the tool, a bounding box appears, showing you the current size of the cropping rectangle. Release the mouse button when the cropping rectangle is the desired size. After creating the cropping rectangle, you see eight small squares that you can use to change the width, height, or width and height of the cropping rectangle. When the rectangle is the desired size, double-click inside the rectangle to crop the document.
 - *Export Area Tool:* Use to export a select area of a document.
- ◆ **Marquee:** Use to create rectangular selections. Press Shift while dragging the tool to create square selections. Or click the down arrow in the lower right of the tool icon to select a tool that creates oval selections:
 - *Oval Marquee:* Use to create oval selections (or circular selections, if you press the Shift key while using the tool).
- ◆ **Lasso:** Use to create freeform selections. Click and drag around the area you want to select. Or click the down arrow in the lower right of the tool icon to access this tool:
 - *Polygonal Lasso:* Use to create selections by clicking to define the first point of the selection and then clicking to define the other points of the selection. Fireworks creates a line segment between points. Double-click to close the selection.



You can add to an existing selection by holding down the Shift key and using one of the selection tools to define the area you want to add. Hold down the Alt key (Windows) or Option key (Mac) while using one of the selection tools to remove an area from a selection.

- ◆ **Magic Wand:** Use to select areas within a document. Click inside an area to select areas of like color. The tool has a Tolerance value from 0 to 255 that you specify in the Property inspector. A low Tolerance value selects hues that are close to that of the area in which you click. A high Tolerance value selects a wider range of colors.
- ◆ **Brush:** Use to paint areas of color within the document. You specify the brush parameters such as size, stroke, and edge in the Property inspector. The Brush tool strokes are divided into categories, such as Watercolor and Oil, which enable you to create strokes like those from an artist's brushes. Each stroke has options that enable you to tailor the brush to your liking.
- ◆ **Pencil:** Use to create freeform lines within a document. You specify the thickness of the stroke within the Property inspector. You can constrain the tool to a straight line or 45-degree diagonal by holding down the Shift key while creating strokes with the tool.
- ◆ **Eraser:** Use to erase areas of color within the document. You specify the shape, diameter, edge, and opacity within the Property inspector.
- ◆ **Blur:** Use to blur areas. You specify the diameter of the tool, softness of the edge, shape, and intensity within the Property inspector. This is

another spot on the toolbar that is home to many tools. Click the down arrow in the lower right of the tool icon to access these tools.

- *Sharpen*: Use to increase the contrast between edges, which makes an object look sharper. You specify the diameter of the tool, softness of the edge, shape, and intensity within the Property inspector.
 - *Dodge*: Use to lighten areas within the document. You specify the diameter of the tool, softness of the edge, shape, range, and exposure within the Property inspector.
 - *Burn*: Use to darken areas within the document. You specify the diameter of the tool, softness of the edge, shape, range, and exposure within the Property inspector.
 - *Smudge*: Use to create areas of smudged color within the document. If you've ever dipped your finger into a bucket of paint and rubbed it against a solid object, you have an idea of the effect created with this tool. You specify the diameter of the tool, edge, shape, pressure, smudge color, and intensity within the Property inspector.
- ◆ **Rubber Stamp**: Use to clone areas from one part of the document to another. After selecting the tool, press Alt (Windows) or Option (Mac) and click the area from which you want to clone pixels, and then paint in the area to which you want the pixels cloned. This tool comes in handy when you want to retouch images you'll use in a Web design. The Rubber Stamp tool is just the tip of the iceberg, though. Click the down arrow at the lower right of the tool icon to access these tools:
- *Replace Color*: Use to replace areas of solid color in the document with another color. In the Property inspector, you can choose the color to replace by using the default From option of Swatch and then clicking inside the document to sample a color, or by choosing Image from the From drop-down menu. You then choose the replacement color in the Property inspector. Click and drag inside the document to replace color. If you choose the Swatch option, the tool replaces the color that matches the swatch. If you choose the Image option, the tool replaces the area of color you first click.
 - *Red Eye Removal*: Use to remove red-eye in images. This tool has two parameters, Tolerance and Strength, which you set in the Property inspector. Click inside the red area of the eye and drag to get the red out.
- ◆ **Line**: Use to create vector-based straight lines in the document. After you begin to draw the line, hold down the Shift key to constrain the line horizontally, vertically, or diagonally on a 45-degree angle. The line is constrained in the direction in which you start dragging.
- ◆ **Pen**: Use to create paths in the document. Click to create a point and then click to create another point. The resulting path comprises points that are connected by line segments. This tool can be used to create complex shapes. If you click and drag, you create a curve point. Curve

points have tangent handles that you can drag to modify the line segment to which the curve point is attached. You can edit paths created with the Pen tool with the Subselection tool. Other path tools can be accessed by clicking the down arrow in the lower right of the tool icon. They are

- *Vector Path*: Use to create freeform paths that you can edit with the Subselection tool. The tool works similar to the Brush tool except that when you release the mouse button, Fireworks creates editable points along the path. You set the tool parameters in the Property inspector. The Precision parameter determines how often Fireworks creates points.
 - *Redraw Path*: Use to modify a path created with the Pen or the Vector Path tool. You must first select the path with the Pointer tool, then select the Redraw Path tool, and then click and drag along the path to change its shape.
- ◆ **Rectangle**: Use to create rectangles. Hold down the Shift key while using this tool to create a square. You set parameters, such as stroke shape, stroke diameter, fill, stroke color, and much more in the Property inspector. But wait, there's more — three more, in fact, that you access when you click the down arrow at the lower right of the tool icon.
- *Ellipse*: Use to create ovals — you know, those rotund critters that form the basis for your generic smiley face. Hold down the Shift key while using this tool to create a circle. You set parameters such as stroke shape, stroke diameter, fill, stroke color, and so on in the Property inspector.
 - *Polygon*: Use to create polygons, those multisided wonders that inspired the Pentagon. You set parameters, including the number of sides, in the Property inspector. Click and drag to create your polygon. The tool also has an option for creating stars, which is a handy option if you're creating a Web site for an actor or rock star.
 - *Auto Shape Tools*: These useful critters (Arrow, Arrow Line, Beveled Rectangle, Chamfer Rectangle, Connector Line, Doughnut, L-Shape, Measure, Pie, Rounded Rectangle, Smart Polygon, Spiral, and Star) are just below the Polygon tool. They work like regular shapes except you modify them using the Property inspector and the Auto Shape Properties panel.
- ◆ **Text**: Use to add words of wisdom to your client's Web site. You use the Text tool to add text to buttons and banners as well as add other cool and groovy things to your design. And you guessed it — you set the parameters for the Text tool in Ye Olde Property inspector.
- ◆ **Freeform**: Use to do yet more cool things with paths. This tool modifies vector paths in a freeform manner. For example, select the tool and drag it across a path you made with the Line tool to transform the path from the straight and narrow to whatever you care to conjure up. And yes, Virginia, Fireworks adds the points to define the reshaped path. If you

take a good look at Figure 2-6, you might notice that one of those cute little down arrows also resides on this tool. Click the arrow to reveal the following.

- *Reshape Area*: Use to reshape an area previously folded, spindled, and mutilated by the Freeform tool. In the Property inspector, set the Size and Strength and then drag it across the path to reshape it.
 - *Path Scrubber (Additive)*: Use to add points to a path created by using a pressure-sensitive tablet. Select the path and then click and drag to redraw the path.
 - *Path Scrubber (Subtractive)*: Use to remove points from a path created by using a pressure-sensitive tablet. Select the path and then click and drag to remove unwanted portions of the path.
- ◆ **Knife**: A clever cleaver of a tool you use to sever (ouch!) a vector object into pieces. Click and drag across the vector object at the point where you want it to split into two pieces. After wielding the knife (“Careful with that axe, Eugene!”), you can use the Pointer tool to move either piece to a different part of the document, or use the Subselection tool to grab one or more points of either piece by the scruff of the neck and modify the shape.
- ◆ **Rectangle Hotspot**: Use to create *rectangular hotspots*, which are interactive areas of the document. You can use rectangular hotspots for image rollovers and other delights. We show you more about this tool in Chapter 3 of this minibook. This tool also has close relatives that you can access by clicking the down arrow in the lower right of the tool icon. They are
- *Circular Hotspot*: Use to create circular hotspots in the document.
 - *Polygonal Hotspot*: Use to create hotspots for irregularly shaped areas. Click to define the first point of the hotspot and then click to define the other points of the hotspot. Fireworks connects the dots.
- ◆ **Slice**: Use to divide (hence the name *slice*) a document into pieces. For example, you can create a slice and use it as the basis for a pop-up menu, image rollover, and so on. Each slice becomes an individual image when the document is exported. Click the down arrow in the lower right of the tool icon to reveal
- *Polygonal Slice*: Use to create irregular slices by clicking to define each point of the slice. And if you guessed that Fireworks connects the dots, you would be correct.
- ◆ **Hide Hotspots and Slices**: Use to hide hotspots and slices (designated as light blue and lime green areas) in the document.
- ◆ **Show Hotspots and Slices**: Use to reveal hidden hotspots and slices.
- ◆ **Eyedropper**: Use to sample colors from within the document.

- ◆ **Paint Bucket:** Use to fill an object with the current fill color. You can also set parameters for the tool such as fill color, tolerance, and opacity in the Property inspector. There's another tool lurking with this tool. Click the down arrow in the lower right of the tool icon to reveal
 - *Gradient:* Use to fill an object with a *gradient*, which is a blend of two or more colors. You can choose a preset gradient or mix one of your own in the Property inspector.
- ◆ **Stroke Color:** Use to specify the outline color of objects you create with tools, such as the Rectangle tool. Click the swatch to open the Color Picker and choose a color. Alternatively, you can click the icon to the left of the color swatch and then use the Eyedropper tool to sample a color from within the document.
- ◆ **Fill Color:** Use to specify the color inside of objects you create with tools, such as the Oval tool. Click the swatch to open the Color Picker and choose a color. Alternatively, you can click the icon to the left of the color swatch and then use the Eyedropper tool to sample a color from within the document.
- ◆ **Set Default Stroke/Fill Colors:** Use to set the color swatches to their default colors: black for the Stroke color and white for the Fill color.
- ◆ **No Stroke or Fill:** Click the Stroke swatch, and then click this tool to create an object with a fill, but no stroke. Click the Fill swatch, and then click this tool to create an object with a stroke but no fill.
- ◆ **Swap Stroke/Fill Colors:** Swap the current Stroke and Fill colors.
- ◆ **Standard Screen mode:** Use to revert to standard viewing mode, the document, tools, and menu.
- ◆ **Full Screen with Menus mode:** Maximize Fireworks to fill the monitor while displaying all elements of the Fireworks workspace.
- ◆ **Full Screen mode:** Display the current document surrounded by a black screen.

Press F to toggle through each viewing mode.

- ◆ **Hand:** Use to pan within the document.
- ◆ **Zoom:** Use to zoom in on the document. Click inside the document to zoom to the next highest degree of magnification. Click and drag diagonally to zoom to a specific area of the document. Press Alt (Windows) or Option (Mac) to zoom out.

Press the spacebar to momentarily switch to the Hand tool. After panning to the desired area, release the spacebar to revert to the last used tool.

Press Tab to hide the tools and all panels. Press Tab again to display the hidden tools and panels.



Creating Art with Other Tools

Fireworks is considered the workhorse for creating graphics for Web pages. However, you can create some very artistic elements for your Web pages using Photoshop CS4. Photoshop has a Save for Web & Devices command, which is used to optimize images for the Web. We show you how to use this command to optimize graphics for the Web in Chapter 4 of this minibook.

Adding Photoshop CS4 to your graphics toolbox

Photoshop CS4 is a powerhouse image-editing application used by professional photographers and artists. The application features an extensive feature set of filters, tools, and commands to perform functions such as color-correcting images, removing red-eye, creating images that look like paintings, and so on. Many Web designers use Photoshop to create content such as splash images and banners. The native Photoshop file format, PSD, supports multiple layers, 16-bit color depth, a unique set of blend modes, and much more. Figure 2-6 shows a splash page for a Web site being designed in Photoshop CS4. Notice the Layers panel, which shows the layers that comprise the final image.

The toolset of Photoshop deserves a book of its own. Some of the features appear in Chapters 3 and 4 of this minibook (which covers creating a banner and Web gallery) and in Book III, Chapter 4 (which covers Web and graphics software in general).

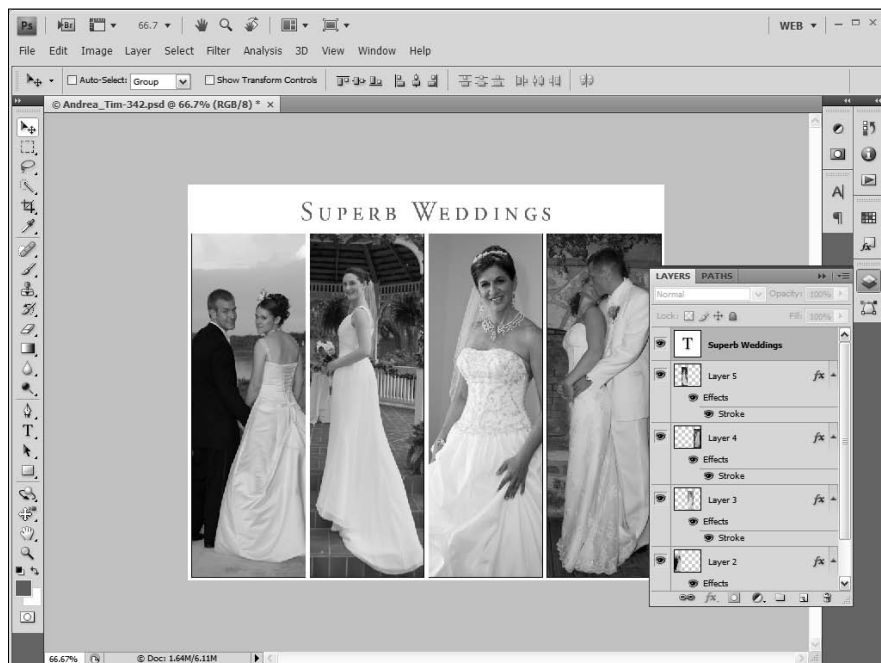


Figure 2-6:
Creating Web
artwork in
Photoshop
CS4.

Getting Images ready for the Web with Photoshop

ImageReady was put to pasture when Photoshop CS3 was introduced, and with good reason. The Fireworks application Adobe acquired when it bought Macromedia is a much better application for creating Web-ready graphics with hotspots, interactive links, and so on. However, there are still times when you create a document in Photoshop when you need to export different parts of the document with different file formats. For example, if you create a splash page similar to the one in Figure 2-7, you get your best results and the smallest file size when you export the text by using the GIF format, and export the sections with images by using the JPEG format. To get an image ready for the Web in Photoshop, follow these steps:

1. Create your image.

This can involve layers, filters, adjustment layers, as well as tips and tricks. Mastering Photoshop deserves a book of its own. A good bet is to check out the *For Dummies* series for the one that best fits your needs.

2. Select the Slice tool.

This tool is introverted by nature and shares space with the Crop and Select Slice tool, which is the fifth slot on the toolbar. The last-selected tool appears at the top of the heap. Click the arrow in the lower right of the last-selected tool, and then select the Slice tool from the fly-out menu (see Figure 2-7).

3. Slice and dice the image as needed.

When you first select the Slice tool, the entire image appears as a slice. The tool snaps to layers in the image, which makes it easy for you to segregate images and text (see Figure 2-8).

Figure 2-7:
The Slice tool cuts images like a surgeon's scalpel.



4. Continue creating slices as needed.

If you're working on a complex image, you might end up with lots of slices. You can select individual slices with the Slice Select tool and then, if needed, resize them by clicking and dragging one of the eight handles that appear around a selected slice. After you slice and

dice your image to perfection — which could be a picture of piece of pizza with pepperoni, say if you're creating a Web site for a pizza restaurant — you're ready to save the image, as described in Chapter 4 of this minibook. This command enables you to use each document slice as an image and also create an HTML file that will assemble them neatly using a table.

Creating painterly images in Corel Painter

Many artists use Corel Painter to transform photographs into images that resemble paintings. If you have a client that wants something a little “different,” consider using Painter. The current version of Painter is version X. Painter features a plethora of brushes that make marks resembling strokes from different media, such as a paint brush or pastel crayons. There are way too many groups of brushes to mention here; Oil, Pastel, and Watercolor are some of its groups. You can modify each brush to create a custom brush. Painter also features layers and blend modes that determine how layers interact with each other. You can also use third-party plug-ins with Painter. Figure 2-9 shows an image prior to being cloned and modified in Painter X (left); the resulting image is on the right.



Figure 2-8: Slicing and dicing for the Web with the Slice tool.

Figure 2-9:
The painterly result of an image edited in Painter X.



Chapter 3: Workhorse Graphics

In This Chapter

- ✓ **Creating simple buttons**
- ✓ **Creating rollover buttons**
- ✓ **From concept to completed page**
- ✓ **Optimizing and exporting your Fireworks design**

When you design a Web site, you create graphics that you use over and over and over and . . . you get the picture. These graphics become the workhorse graphics for your Web pages: buttons, for example. Buttons can range from the mundane to passing-for-sane. In other words, you can have a site with buttons ranging from very utilitarian (you know, functional?) to very cool that are both functional and artistically gorgeous. Buttons are an integral part of your design. This chapter shows you how to create buttons and other graphics for your Web design. So roll up your sleeves and get ready for a magical Fireworks *tour de force*.

Creating Buttons in Fireworks

You have to have navigation for a Web site. Without navigation, your visitors, looking at the home page, might yawn and say something like, “Is that all there is?” Hopefully not, if you expect to be a Web designer for any length of time. Clients also have a tendency to frown on one-page Web sites. Therefore, you need to provide visitors a way to navigate from page to page. In the old days (before cool applications like Fireworks appeared), navigation menus consisted of underlined text that, when clicked, transported the visitor to a related page. Although functional, this method was far from being artistic or compelling. Fortunately, Fireworks makes child’s play of creating cool buttons. In Fireworks, you can use preset buttons.

Introducing the Button Editor

When you create a button in Fireworks, you use the Button Editor. The Button Editor can create anything from a simple two-state button to a multistate button used for a navigation menu. Figure 3-1 shows a Fireworks preset button in the Button Editor.

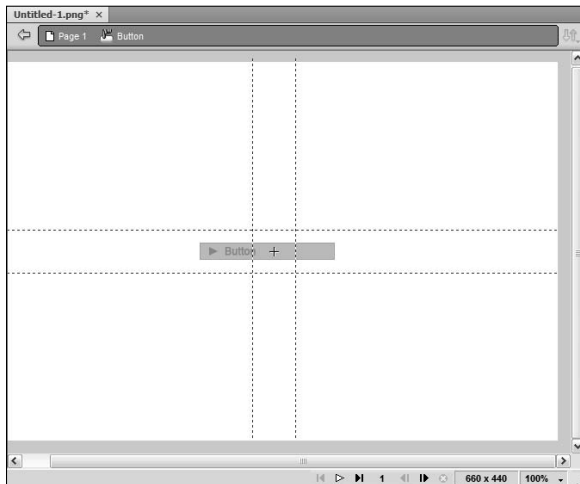


Figure 3-1:
Edit me a
button, my
lovely!

Creating a simple two-state button

The concept of a two-state button is scary, especially if the states are California and Florida. The left coast meets the right coast. Yikes! But we digress. As we mention earlier, a two-state button is the default Fireworks button. You create the graphics and text for the *State 1*, which is what the viewer sees when the page initially loads. Then you modify the same graphics for the *State 2*, which is what the user sees when hovering his cursor over the button. The different graphics are a dead giveaway that something will happen when the button is clicked. Of course, the site visitor also sees the ubiquitous hand with the pointing finger like he would with a text link, but using different graphics for each button state makes your design much cooler.

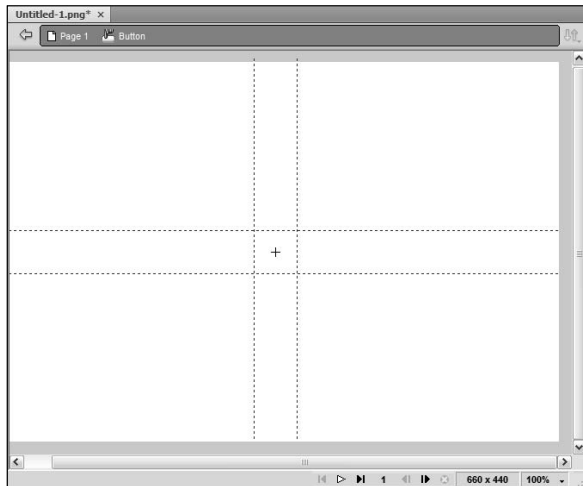
Before you can create any button, figure out what size to make it. When you decide to create a Web site, you do some sort of planning. We describe one method of going from concept to completion in the upcoming “Doodle to a Working Page: Concept to Completion” section of this chapter. Whichever method you use, you know how much area you delegated for navigation. If you have only a few links, you can use a horizontal navigation menu. If you have lots of links, though, you need a vertical menu. After you meet with and create a mockup for your client, you should know the title for each button. Your button should be slightly larger than the longest title. With all that figured out, you’re ready to create a two-state button as follows:

- 1. Choose Edit→Insert→New Button.**

The Button Editor appears. (See Figure 3-2.)

- 2. Choose your drawing tool of choice.**

Figure 3-2:
So this is
the Button
Editor!

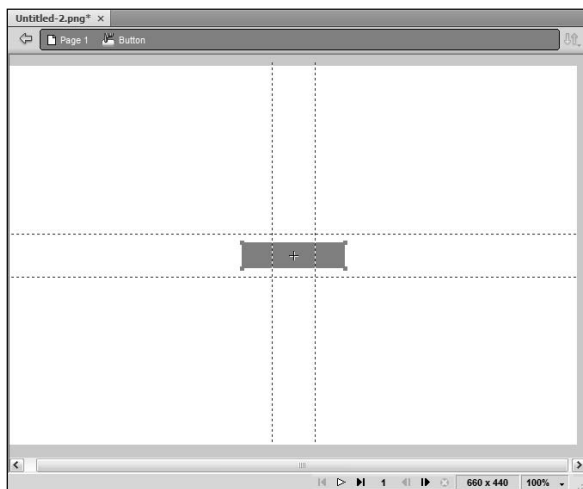


The logical choice is the Rectangle tool. If you're creating a pill-shaped button, you can choose the Rectangle tool and then use the Rectangle Roundness parameter to round the corners of the rectangle.

3. Create a shape with your tool of choice in the Button Editor.
4. In the Property inspector, specify the width and height of your button and any other parameters, such as stroke color and fill color.

Figure 3-3 shows a button under construction.

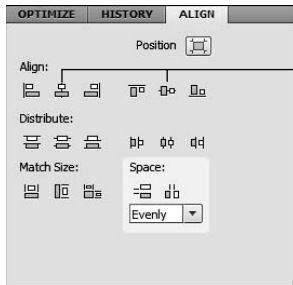
Figure 3-3:
Building
the perfect
button.



5. Choose **Window**⇧**Align**.

The Align panel opens. (See Figure 3-4.)

Figure 3-4:
Web site
designers
align
objects, not
wheels.



Click these to align
the button to the center
of the Button Editor.

6. Click the **Position** icon, and then click the middle icons in the **Align** section.

This aligns the shape to the center of the Button Editor.

7. Select the **Text** tool. In the **Property inspector**, specify the parameters for the button text.

Verdana and Arial are both good choices for the font face. Choose a color that's harmonious with your design. The font size depends on the size of your button. If you make a small button and try to pack a lot of text on it, the button is hard to read.

8. Type the desired text on the button.

9. With the text selected, choose **Window**⇧**Align**, and then align the button to the center of the canvas.

Figure 3-5 shows a button with text on it. The neat thing about Fireworks is you don't have to create a button for each menu item. An upcoming section of this chapter shows you how to flesh out a navigation menu from one button.

10. Choose **Select**⇧**Select All**.

If you're like us and like keyboard shortcuts, just press **Ctrl+A** (Windows) or **⌘+A** (Mac) instead. Selecting everything makes it easy to copy the graphics and text to another state.

11. Choose **Window**⇧**States**.

The States panel opens in the Panel docker (see Figure 3-6).

12. Right-click (Windows) or Control-click (Mac) **State 1** to display the contextual menu and then choose **Copy to States**.

Back button

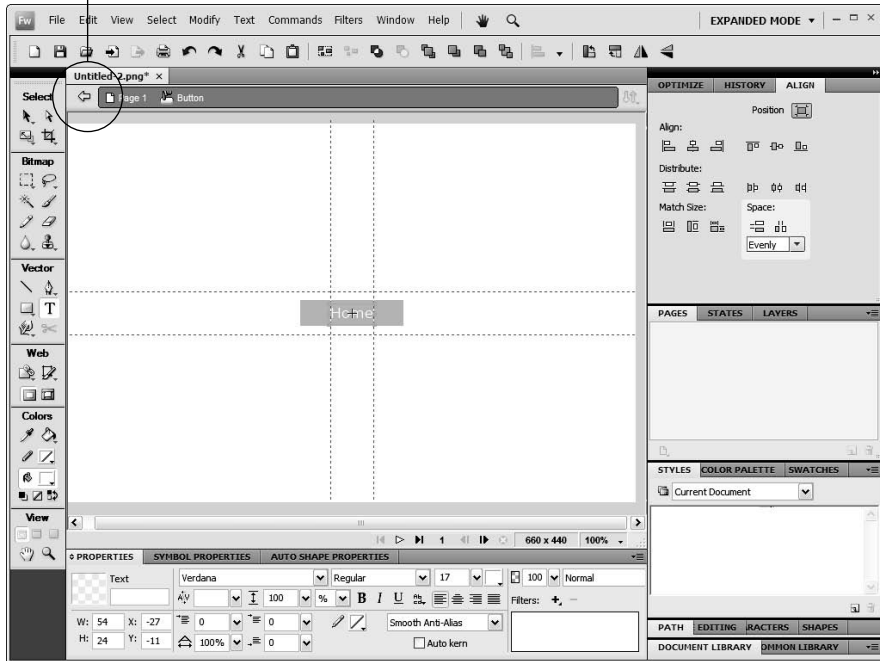


Figure 3-5: A button's gotta have text. It's a law.

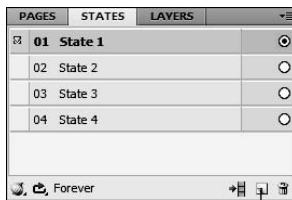


Figure 3-6: There is not a state of confusion here.

New/Duplicate State icon

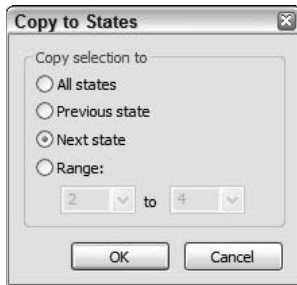
- 13. In the Copy to States dialog box that opens (see Figure 3-7), select the Next State radio button and then click OK.**

You have a carbon copy of the shape and text in State 2. This is the graphic users will see when they pause their mouse over the button.

- 14. Select State 2 and then modify or change the graphic you copied from State 1.**

Of course, you could go with a different shape, but that's not practical for a conventional navigation menu button. You modify the button by changing its color and/or the text. For a conventional site, we vote for changing the color of the text.

Figure 3-7:
If this dialog box could copy us to another state, we could save lots of gas.



15. Click the Back button to return to the main window.

The button is added to your design.

Creating buttons with pizzazz

Two-state buttons work great for most Web sites. However, if you want to kick the button up a notch, you can include two more states:

- ◆ **State 3:** The graphic that's displayed on the down stroke of a mouse click.
- ◆ **State 4:** The graphic that's displayed when a user moves his cursor over a button that is part of a navigation (nav) menu and the page linked to the button is displayed.

To create a multistate button, repeat Steps 1–15 in the preceding steps for creating a simple two-state button, and then continue with these steps:

1. In the Copy to States dialog box, select the All States radio button and then click OK.

You have a carbon copy of the shape and text in States 2, 3, and 4.

2. Select State 2 in the States panel and then modify the graphic.

As a rule, we change the button and/or text button depending on the site design and the client.

3. Modify the graphic in State 3 and 4.

What you do to the shape and text is a matter of personal taste — yours or your client's.

4. Click the Back button to return to the main window.

The button is added to your design (see Figure 3-8).

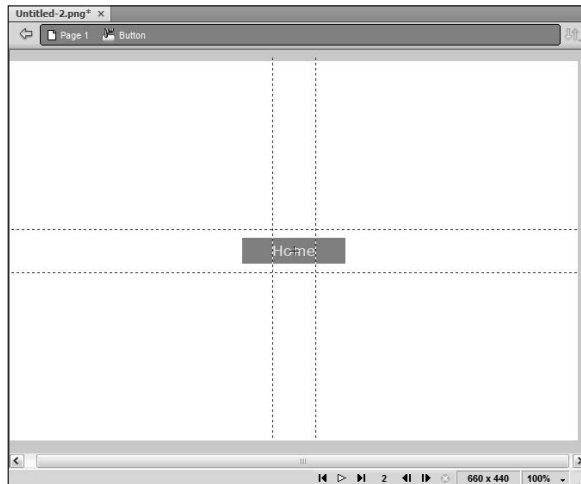


Figure 3-8:
A button
in need of
a home.

Creating a Navigation Menu

A *navigation menu* is nothing but a collection of buttons all lined up neatly horizontally or vertically. You can do all the grunt work in Fireworks and then add the menu to an HTML document you're creating in Dreamweaver. Talk about your applications that play well with others!

To create a vertical navigation menu, follow these steps:

- 1. Create a new document in Fireworks.**

The document dimensions are the width and height of the area in which the menu appears in your HTML design.

- 2. Create a single button.**

If you fast-forwarded to this section, please rewind to the "Creating Buttons in Fireworks" section of this chapter.

- 3. Select the button with the Pointer tool.**

- 4. Hold down the Alt key (Windows) or Option key (Mac) and drag down.**

This creates an instance of the button. After you start dragging, hold down the Shift key to constrain the motion vertically.

- 5. Release the mouse button when the instance is aligned to the bottom of the previous button.**

Objects in Fireworks have a magnetic attraction to each other. The button actually snaps to the bottom of the button above it.

6. Repeat Steps 4 and 5 to flesh out your menu.

At this stage, you have a column of buttons with the same title. (See Figure 3-9.) Not to worry; you'll give each button its own identity in the following steps.

7. Select the first button in your menu. If the Property inspector isn't already open, choose Window⇨Properties.

8. In the Property inspector, enter the following parameters:

Link: The URL to the page that opens when the button is clicked.

alt: The alternative text that's displayed in screen readers. alt text is also used by search engines when indexing a site. Certain browsers, such as Internet Explorer, display alt text as a tooltip when a user pauses her cursor over the button.

Target: The target window in which the linked page appears. In most instances, `_top` is the proper choice. Your options are

- `_blank`: Displays the document in a new and unnamed browser window.
- `_parent`: Displays the document in the parent window currently displaying the frame.
- `_self`: Displays the document in the same window or frame as the link.
- `_top`: Displays the document in the body of the current window. This option ensures the document called by the form action displays in the full browser window, even if the document was originally displayed in a frame.

Show Down State on Load: Shows the button's Down state when the page to which the button is linked loads.

9. Select the second button. If the Property inspector isn't already open, choose Window⇨Properties.

You modify parameters for each button, but you perform one extra step for each button other than those on the home page.

10. Type the title of the button in the Text field. (See Figure 3-10.)

This changes the text that displays on the face of the button.

11. Use the Property inspector to modify the other parameters of the button.

12. Repeat Steps 9 through 11 for the remaining buttons.

The finished menu is shown in Figure 3-11.

Figure 3-9:
Creating instances of the original button.

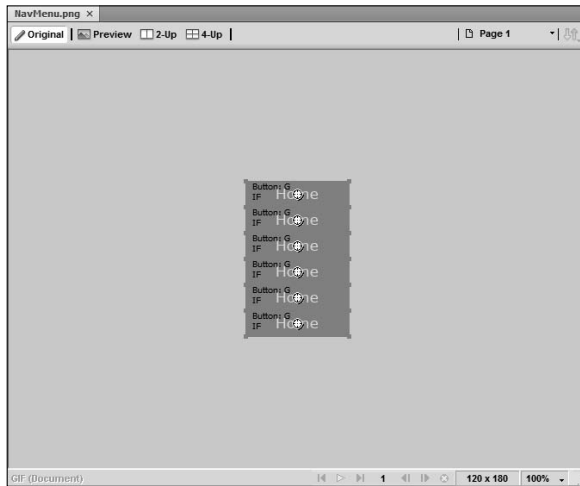


Figure 3-10:
Change the button face text.

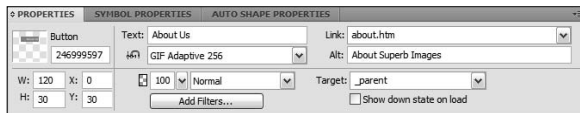
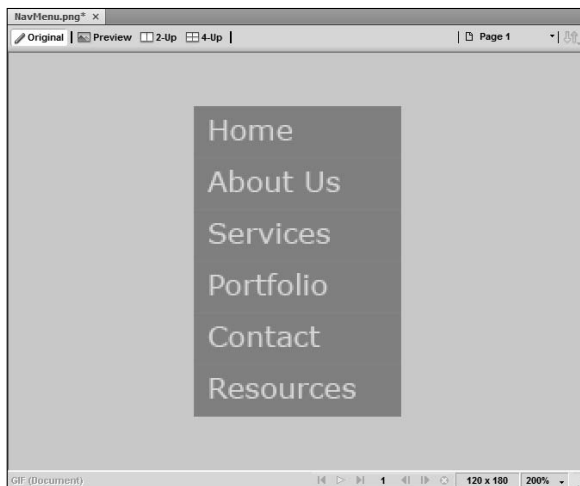


Figure 3-11:
The finished menu.



You can design horizontal navigation menus in the same manner. Create a document that's the height of your button. The width of the document is the button length multiplied by the number of buttons. After you create the first button, use Alt (Windows) or Option (Mac) to drag the button right while you hold down the Shift key to create an instance of the button.

Creating a Pop-Up Menu

If you have a lot of pages to link to but only a limited amount of space, a pop-up menu is the obvious choice. A *pop-up menu*, um, pops up (hence the name) when a site visitor pauses his cursor over the button that triggers the pop-up menu. (That's right, Roy — *triggers*.) You might be thinking doomsday thoughts like JavaScript code, but you'd be only half right. A pop-up menu created in Fireworks uses JavaScript to create its magic. However, Fireworks writes all the code. To create a pop-up menu, follow these steps:

- 1. Use the Pointer tool to select the button to which the menu will be attached.**

A white button appears in the middle of the button. This signifies that you can add interactivity to the button. When you create a button, Fireworks creates a slice that conforms to the size of the button. The slice is saved as an image when you export your menu.

- 2. Click the white button.**

A menu appears with a list of interactive behaviors you can add to the button. (See Figure 3-12.)

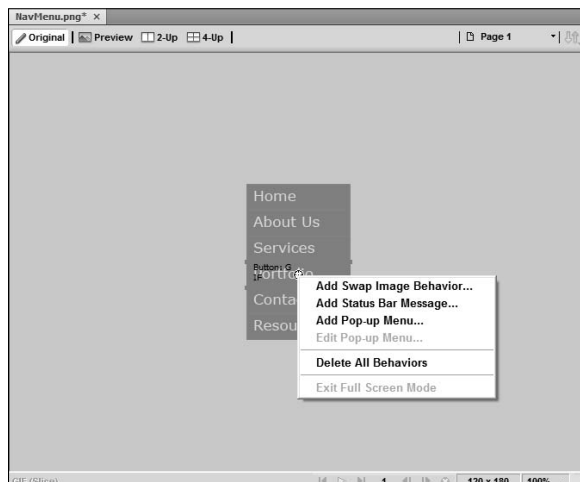
- 3. Choose Add Pop-Up Menu.**

The Pop-Up Menu Editor appears. (See Figure 3-13.)

- 4. Enter the Text, Link, and Target for the item.**

The Text is the button label, the Link is the page to which the button links, and the Target is the window in which the linked page opens. If you're not familiar with targets, check out the earlier section, "Creating a Navigation Menu."

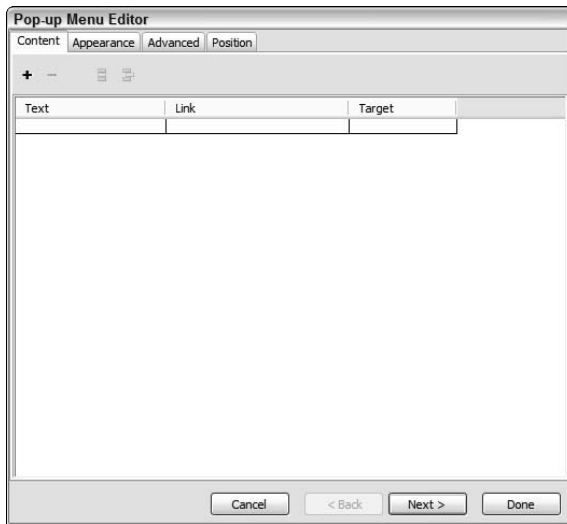
Figure 3-12:
Gonna get
down, get
interactive
with the
button.



5. Click the plus sign (+) to add another item to the menu.
Fireworks adds blank fields for the button's Text, Link, and Target. Alternatively, you can press Tab to add another item to the menu.
6. Repeat Step 4 for this button.
7. Continue adding the other items to your pop-up menu.

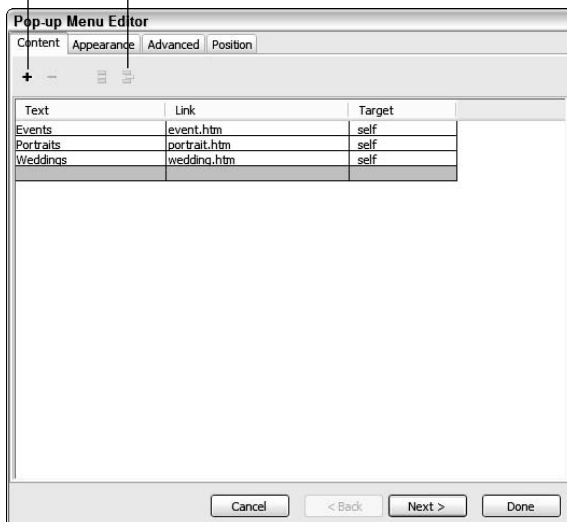
Figure 3-14 shows a pop-up menu with several items.

Figure 3-13:
Pop-ups
R us.



Add Indent Menu

Figure 3-14:
Adding
items to
the pop-up
menu.



To change the order in which items appear on the menu, click an item and then drag it to the desired position.

To indent menu items, click the Indent Menu button. When you indent menu items, they appear as their own pop-up menu (or submenu if you prefer) when a visitor pauses her cursor over the parent menu item.

8. Click Next.

The Appearance tab of the Pop-Up Menu Editor displays. (See Figure 3-15.)

9. Specify the parameters for the pop-up menu items' appearance:

- *Cells:* Select HTML or Image. If you choose HTML, Fireworks creates the code for creating the menu items. If you choose Image, Fireworks creates the images for each menu item and saves them when you export the menu.
- *Menu alignment drop-down menu:* Choose Horizontal Menu or Vertical Menu.
- *Font:* Choose a font set. These are identical to the font options you have in Dreamweaver. The first font is the default. The Web browser defaults to the next font face if the default font isn't installed on the user's computer.
- *Choose a font size from the drop-down menu.*
- *Font style and alignment options:* These are the same options you find in your friendly word-processing application: Bold; Italic; and left, center, or right align.

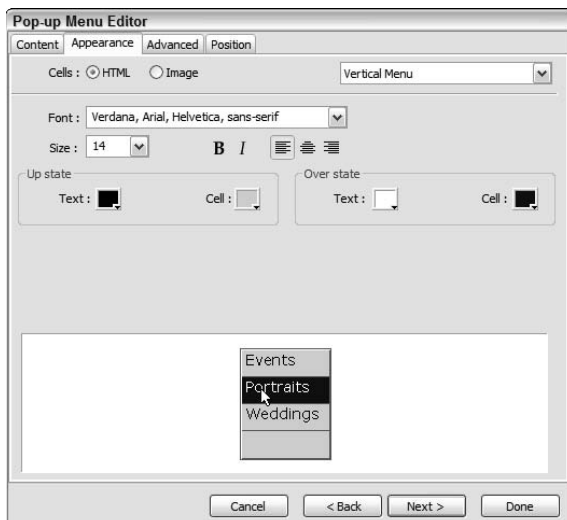


Figure 3-15:
The
Appearance
tab
makes an
appearance.

- *Up State:* Click the color swatches and choose the color for the cell and text. As a rule, you choose the same colors as the menu to which the pop-up menu is attached.
- *Over State:* Click the color swatches and choose the color for the cell and text that appear when a visitor pauses the cursor over a menu item.

10. Click Next.

The Advanced tab of the Pop-Up Menu Editor appears. (See Figure 3-16.)

11. Accept or modify the following parameters:

- *Cell Width:* Fireworks determines the width based on the menu item with the longest text. To modify the width, choose Pixels from the drop-down menu. This opens the Cell Width text box into which you can enter a value.
- *Cell Height:* Fireworks chooses a height, which can be modified by entering a different value in the Cell Height field. Alternatively, you can choose Automatic from the drop-down menu.
- *Cell Padding:* Accept the default value, or enter a different value. This value determines the area in pixels around the cell text.
- *Text Indent:* Accept the default value (zero) or enter an amount by which to indent the text.
- *Cell Spacing:* Accept the default value of 0, or enter a value. If you enter a value, a space appears between each item on your pop-up menu. We like our menus tight, so we accept the default value of 0.

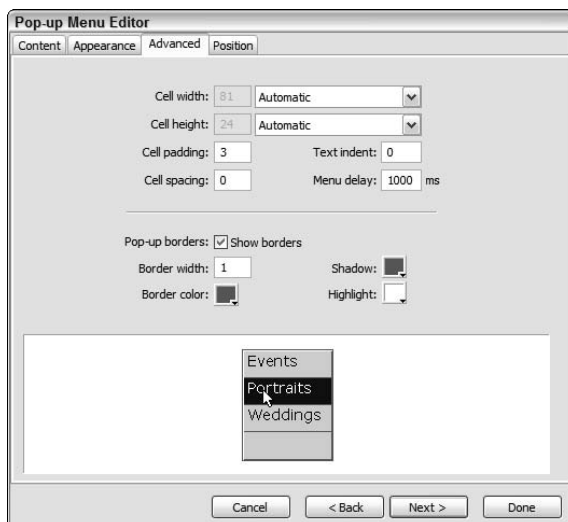


Figure 3-16:
And now for the Advanced properties of your pop-up menu.

- *Menu Delay:* This is the amount of time for which the menu appears after a visitor pauses his cursor over the button. If no action is taken within that amount of time, the menu disappears. The default delay is 1000 ms, or 1 second. Enter a different value to increase or decrease menu delay.
- *Pop-up Borders:* The Show Borders check box is selected by default, with a default value of 1. Enter a value to display a bigger border, and then click the applicable color swatches for the border: Shadow, Border Color, and highlight. This option is not available if you choose Image on the Appearance tab.

12. Click Next.

The Position tab of the Pop-Up Menu Editor appears. (See Figure 3-17.)

13. Click an icon to align the pop-up in one of the following configurations: to the bottom right of the slice, to the bottom of the slice, to the top of the slice, or to the top-right of the slice.

After you choose an option, Fireworks inserts values in the X and Y fields. If you want, you can enter different values to further define the positioning of the pop-up menu. If you have indented menu items, the submenu options appear, which enable you to specify the position of submenus.

14. Click Done.

The pop-up menu is added to the button.

15. Press F12 to preview the menu in your default Web browser. (See Figure 3-18.)

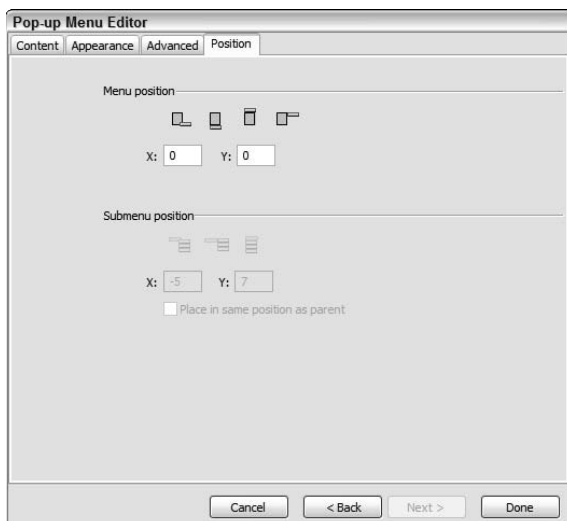
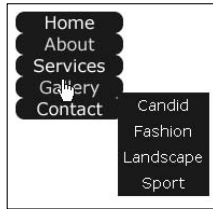


Figure 3-17:
Position is everything.

Figure 3-18:
A pop-up menu. Now everyone will want one.



Recycling: Reuse Everything

Creating graphics for your Web designs is time consuming. When you create something for your own or for a client’s site, keep it. Everything you create in Fireworks and save as a PNG file (Fireworks’ native file format) is fully editable. For example, if you create a vertical navigation menu in Fireworks, you can change the document size, the size of each button, and the other parameters such as the button text, URL, target window, and so on. If the colors clash with the site where you’re going to put them, change the colors. It’s a lot easier than creating an item from scratch.

Making a reusable graphic template

Why work harder when you can work smarter? When you create a new Web design, you don’t have to reinvent the wheel: Just modify it to suit the client. Most designers develop a signature style. If you fall into that category, you can use Fireworks to create templates for the things you use often, such as banners and navigation menus. As we mention, Fireworks documents are saved with the PNG extension. The resulting document is different from the document you get when you save a file using the PNG format in an application like Photoshop. The Fireworks PNG document is fully editable. Figure 3-19 shows the document for Doug’s wedding and event Web site navigation menu.

He can easily use this menu on a client’s site by doing the following:

- ◆ **Changing the Canvas size and color in the Property inspector.**
- ◆ **Modifying the button color and size by selecting it in the Library, and then double-clicking it to open it in the Button Editor.** If the menu is created using the techniques outlined in the earlier section, “Creating a Navigation Menu,” editing the button also changes the color and size of each instance of the button.
- ◆ **Modifying the text and URL of each button in the Property inspector.**



Keep a folder of assets that are editable Fireworks and Photoshop files. When you’re creating a new site for a client, just pick and choose the assets that you can easily modify to suit the site you’re designing.

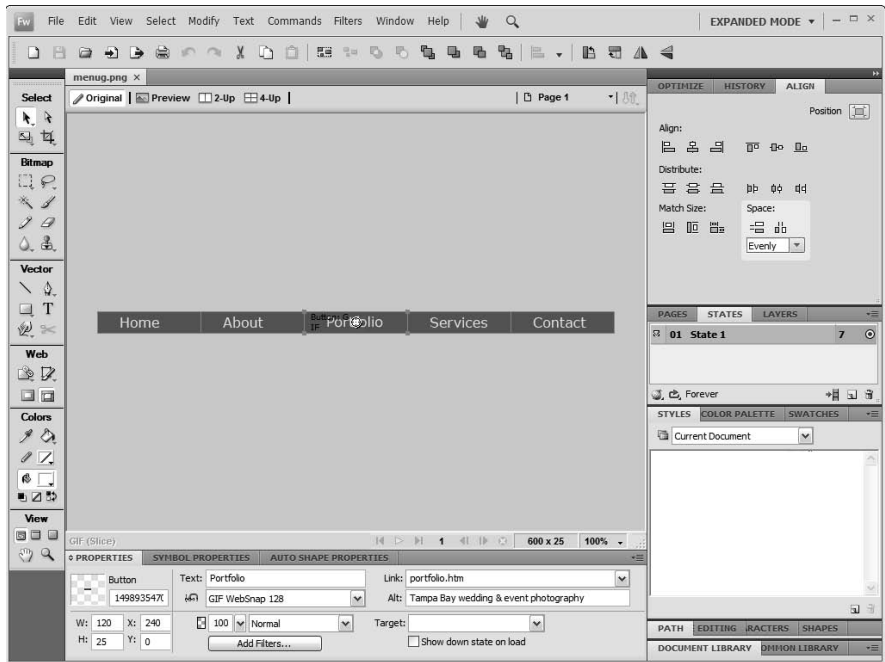


Figure 3-19:
A fully
editable
navigation
menu.

Organizing a site

When you create a single Web site, you might end up with hundreds of files. Some of them are items supplied by your client that you need to optimize for the Web site. Others are files you create in applications, such as Fireworks and Photoshop. When you deal with that many files, it's definitely in your best interest to be organized. Consider adopting some version of the system Doug uses to organize site assets:

- ◆ **Client:** This is the main folder for all files pertaining to the Web site. The folder bears the client's name.
- ◆ **Client Supplied Assets:** You can lump all assets supplied by the client into this folder. If the assets are of several different types, subdivide the folder keeping all text files in one subfolder, images in another subfolder, multimedia assets such as video in yet another folder, and so on.
- ◆ **Assets:** Store all the assets you create in this folder. This folder might also be subdivided. For example, you could keep all Fireworks and Photoshop files in a subfolder named Images, and keep Flash and video content in a folder named Multimedia.
- ◆ **Site:** This folder contains all the files that you'll upload to the site server. In the main folder, keep all the files that will be in the root folder on the server. Keep a separate folder for images. If the site needs to be subdivided into different folders on the server, create those folders on your local computer as well.

A Doodle to a Working Page: Concept to Completion

When your client first contacts you, he gives you an idea of the concept behind his Web site. If the first meeting is by phone, jot down as much information as you can and then arrange to meet the client. The meeting can be at his office or your office. Sometimes neither is convenient, and you end up meeting in a coffee shop. Bring your notes from the original phone consultation, a legal pad, and a few writing implements. Bringing a few pens with different color inks to create sketches helps ensure that you and the client are on the same page. (The strangest client consultation Doug ever held occurred when he was introduced to a potential client in a restaurant. Instead of arranging a meeting for the next day, the client cut right to the chase, and Doug ended up using paper napkins to jot down ideas that he later transferred to a legal pad.)

Mind mapping

You can do mind mapping by using an application like Fireworks or by using a very large sheet of paper. In a nutshell, the exercise amounts to creating shapes that define the major parts of the Web design. For example, you create a rectangular shape at the top of the document for the banner, another rectangular shape for the menu, and other rectangular shapes for the text and image components of the page. Feel free to experiment with different sizes and locations for the elements that will eventually be incorporated into your design. The next step is to provide the client with a mockup of the Web site — or, as some Web designers call it, a *comp*.

Creating a client mockup

After your mind-mapping session, you're ready to create something for your client to approve. You can easily create several versions of a client mockup in Fireworks. If you used Fireworks to do your mind mapping, use that document as the basis for your client mockup. Flesh out the shapes with actual content. If you have assets already created, you can incorporate them into the design. Add link titles to the shapes you created for the navigation menu. Then you're ready to create alternative versions of the mockup. To create alternative versions of your mockup in Fireworks, just follow these steps:

- 1. Open the States panel.**
- 2. Select the first frame and then click the New/Duplicate State icon (refer to Figure 3-6).**

Fireworks creates a carbon copy of the first state.

- 3. Create a variation of the first state.**

This is the second design you present to the client. You can change the manner how the menu is displayed, the placement of objects, choice of colors, and so on.

- 4. Create additional states to create different variations of the design.**
- 5. After creating different variations of your design, choose File→Export.**
Fireworks displays the Export dialog box.
- 6. Choose the desired file format in which to save the images. Then from the Export drop-down menu, choose States to Files.**
- 7. Name the document and then click Save.**

After you export the states, you have multiple files with the same file-name, appended by the number of the state. You can now send the individual versions to your client for consideration.

Optimizing Artwork in Fireworks

When you *optimize* artwork in Fireworks, you choose the file format for export and then specify other parameters. The file format in which you export the document depends on the type of artwork you're creating. When you export the document, you specify other parameters, such as the amount of compression applied to a JPEG file or the number of colors and the palette for a GIF image file.

Optimizing GIF artwork

If your design has large areas of solid color and other elements, such as a client logo, the GIF file format is the ideal format in which to optimize your design. Banners and navigation menus are other candidates for the GIF format. To find out more about the GIF format, see Chapter 1 of this mini-book. To optimize an image for export in the GIF format, follow these steps:

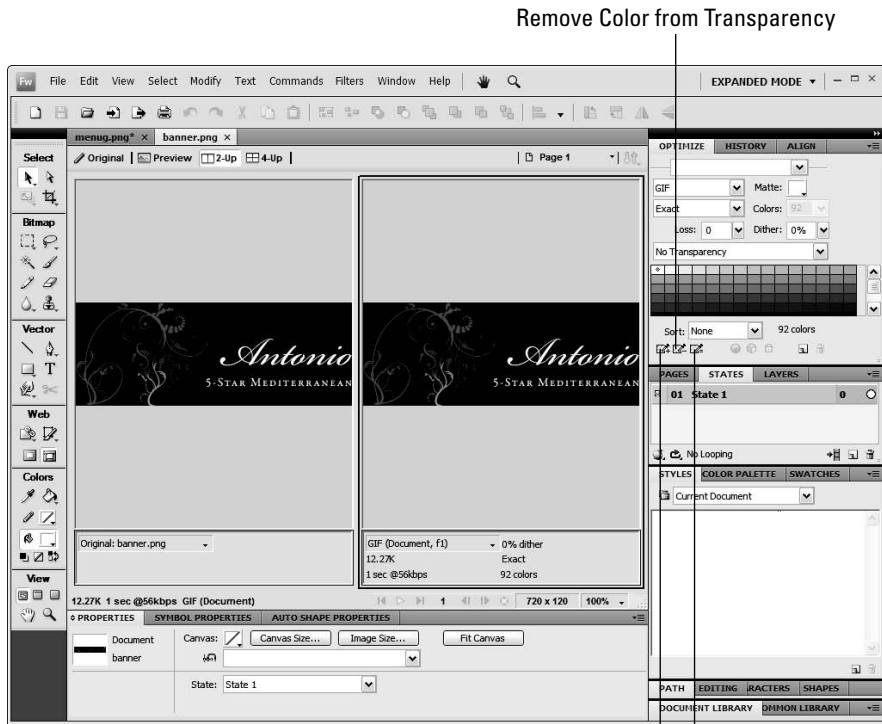
- 1. Click the 2-Up icon at the top of the document window.**

Fireworks displays two versions of the image: the original and a copy with the current optimization settings applied. Comparing the original with the optimized version lets you decide the best setting for optimizing the document.

- 2. Choose Window→Optimize.**

The Optimize panel is displayed. Figure 3-20 shows the document window in 2-Up display and the Optimize panel. The right side of the window displays the image with the current optimization settings applied. Below the image is the file size, number of colors in the palette, and the estimated time to download the file with an Internet connection speed of 56 Kbps.

Figure 3-20:
Please optimize me — set me free.



Remove Color from Transparency

Add Color to Transparency

Select Transparent Color

3. Choose a preset from the Saved Settings drop-down menu (in the upper right of Figure 3-20).

Preview each setting and pay attention to the right pane of the document window. Choose the preset that's the best compromise between image quality and file size. Your choices are

- *GIF Web 216*: Converts all colors to Web-safe colors. The color palette contains up to 216 colors.
- *GIF WebSnap 256*: Converts non-Web-safe colors to their closest Web-safe colors, creating a color palette that contains up to a maximum of 256 colors.
- *GIF WebSnap 128*: Converts non-Web-safe colors to their closest Web-safe colors creating a color palette that contains up to 128 colors.
- *GIF Adaptive 256*: This color palette contains only the actual colors used in the document up to a maximum of 256 colors.

4. Modify the preset to suit your document.

At this stage, you can experiment with reducing the number of colors in the Indexed palette. You can either enter a value in the Colors text field, or choose a preset from the Colors drop-down menu. If you notice image degradation, you went too far. When this occurs, choose the next highest value from the Colors menu.

To export an image with transparency, follow these steps:

1. Choose Index Transparency from the Transparency drop-down menu.

The default is No Transparency, as shown in Figure 3-20. When you choose a transparent color, the background of the HTML document shows through. This is a useful option when you have an image as a background on the HTML page on which the graphic will appear.

2. Click the Select Transparent Color eyedropper and click inside the document to sample the transparent color.

Fireworks displays the transparent area as a checkerboard in the right pane of the document pane when displayed in 2-Up mode. (See Figure 3-21.)

3. You can modify the transparent colors as follows:

- Click the Add Color to Transparency eyedropper (refer to Figure 3-20). Then, inside the right pane of the 2-Up window, click the color you want to be transparent.
- Select the Remove Color from Transparency eyedropper. Then, inside the right pane of the 2-Up window, click a transparent area. The associated color will no longer be transparent.



Figure 3-21:
Now you
see me; now
you don't.



You can also find the option for Alpha Transparency, which is only supported by images optimized in the PNG format — and is not widely used by Web designers.

Optimizing JPEG artwork

The JPEG format is best suited for artwork that is photorealistic in nature. The JPEG format compresses the file by losing color data; therefore, JPEG is known as a *lossy* format. Your goal is to export the image at the smallest possible file size without noticeable degradation. To optimize a document using the JPEG format, follow these steps:

1. Click the 2-Up icon at the top of the document window.

Fireworks displays two versions of the image: the original and a copy with the current optimization settings applied. Comparing the original with the optimized version lets you determine the best setting for optimizing the document.

2. Choose Window → Optimize.

The Optimize panel is displayed. When you optimize a document, you can click the 2-Up button to display two versions of the image. The right side of the window displays the image with the current optimization settings applied, and the left side shows the original image. Below the image is the file size, number of colors in the palette, and the estimated time to download the file with an Internet connection speed of 56 Kbps. Make sure you have image magnification set at 100 percent so you see the pixels at their actual size.

3. Choose an option from the Saved Presets drop-down menu.

We recommend starting with the JPEG-Better Quality option. The theory is to start with a high quality and apply compression until noticeable degradation occurs.

4. Gradually apply more compression to the image by specifying a lower image quality.

You can enter a value in the Quality text box or drag the Quality slider. When you see noticeable degradation in the image, gradually bump up the image quality until the end result looks good to you.



If you have to apply heavy levels of compression to get the desired file size, choose an option from the Smoothing drop-down menu in the Optimize panel to smooth out the jagged edges caused by heavy compression.

Exporting Artwork from Fireworks

After you optimize the image, it's time to export it from Fireworks. Exporting images is fairly straightforward; choose File⇨Export and then specify the folder in which to save the image. When you export a document like a navigation menu with multiple slices, you export the document as images and HTML as follows:

1. Choose File⇨Export.

The Export dialog box appears.

2. Choose HTML and Images from the Export drop-down menu.

This option exports the document as an HTML file with all the code necessary to reassemble the exported images, create links for navigation menus, and so on.



When exporting a JPEG image, choose Images Only. Normally this is selected by default when you're exporting an image without slices or hotspots.

3. Choose Export HTML File from the HTML drop-down menu.

Your alternative is to copy the HTML code to the Clipboard.

4. Select the Include Areas without Slices check box.

This exports the areas of your document such as the background that has not been designated as a slice.

5. Click the Put Images in Subfolder option.

This helps keep your files neat and tidy. By default, Fireworks creates a folder named Images. You can specify another folder by clicking the Browse button and then navigating to the desired folder.

6. Click Save.

Fireworks exports the document (as an HTML file) and images to the specified folders.

Chapter 4: Creating Compelling Graphics

In This Chapter

- ✓ **Creating site banners**
- ✓ **Creating compelling graphics**
- ✓ **Using Web-friendly photos**
- ✓ **Incorporating Web photo galleries**

Almost every Web site you see has graphics. Face it — a Web site without graphics is text. And how boring is that? To make the sites you design stand out among the plethora of sites on the Web, you need to incorporate fast-loading, compelling graphics. Graphics come in many flavors. Some of them we covered in the preceding chapter of this minibook. In this chapter, we discuss the graphics that elevate the sites you design above the competition. These include such graphic elements as the banner, splash images, and photo galleries. When you create these elements, you incorporate your own design preferences with the graphic elements from your client's brick-and-mortar business if he has one.

Creating Banner Graphics

Banners are proudly displayed at the top of just about every Web site you visit. The banner often incorporates the client's marketing tools, such as the company logo. When you create a banner for a site, you use the company logo and a color scheme that's harmonious with your client's logo. If your client doesn't have a logo, you use your own sense of design to create the color palette for the site.

Using Photoshop to create a banner

You can create a banner in just about any image-editing application that enables you to create a document from scratch. Photoshop, however, gives you the power to augment your creativity with its rich feature-set. Photoshop ships with a set of Adobe Pro fonts that give you the capability to create a very unique banner.

When you design a banner, your first consideration is size. Your client's logo — if she has one — is nestled in the design. Therefore, the banner must be tall enough to prominently display the logo. Fortunately, most professionally designed logos look good in sizes from the sublimely small

(the customer's letterhead) to the ridiculously large (a billboard ad). If the client's logo already has text, the logo and a harmonious background color are all you need. If the client logo displays only an artistic arrangement of the client's initials, you have to add text. As a rule, centering banners and site content is preferable.

When you decide on the size for your banner, you must take into account the amount of available monitor real estate for the desktop size of your client's intended audience. A good practice is to optimize Web graphics for the lowest common denominator, which is currently 800 x 600 pixels. In order to keep all your content above the fold, you have to deal with an area of 760 x 420 pixels, which is all you have left after browser toolbars, menus, scroll bars, and so on. Therefore, the maximum size of any banner you create should be 700 x 120 pixels. This leaves you room for a navigation menu, images, and text.

Figure 4-1 shows a simple banner being designed in Photoshop. The banner is text-only because the business being promoted on the Web site is a restaurant that didn't have a logo when the initial site was being designed. The banner size is 720 x 120, large enough to elegantly display the company name in Edwardian Script font and the tag line in Adobe Garamond Pro font. The text for the tag line was further formatted using the Photoshop Character panel, which enables the designer to incorporate font styling such as small caps, faux bold, and faux italic. Faux bold and italic are especially useful if the desired font face does not come with bold or italic styling options. Figure 4-2 shows the banner as displayed on the client's Web site.

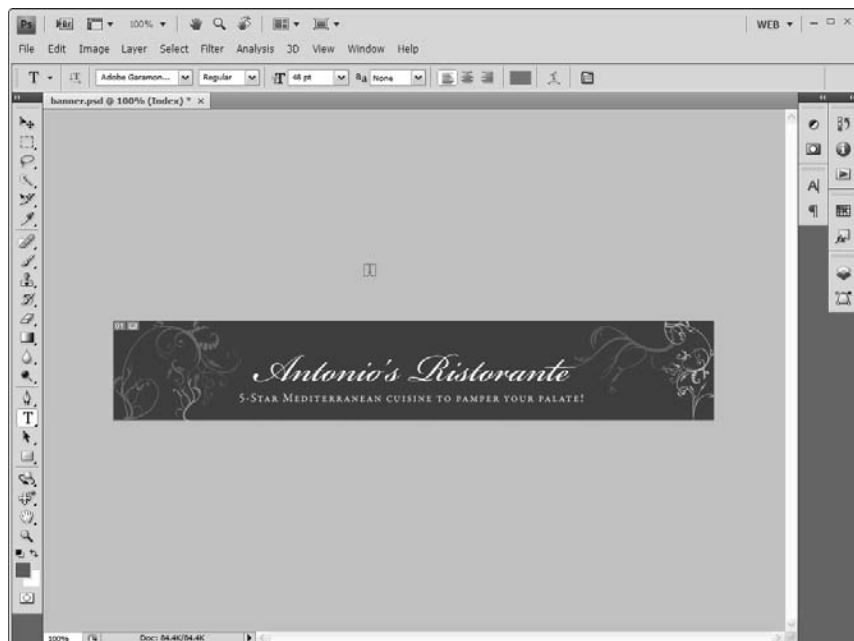


Figure 4-1:
Designing
a simple
banner in
Photoshop.



Figure 4-2: The banner incorporated with the Web design.

Including all the important information

The size of your banner is determined by the amount of information your client wants to display. Sometimes your client goes over the top and asks you to include everything but the kitchen sink: for example, the name of the company, address, phone number, Web site URL, and so on. That's simply way too much information, and why most Web sites include a Contact Us page. In reality, all a banner needs are the following elements:

- ◆ **Client logo:** If your client has a logo, it should be displayed prominently on the banner. Logos aligned to the left side of the banner look good.
- ◆ **Company name:** If the client's logo doesn't display the company name, add the company name to the banner. If you're designing a banner for a client who doesn't have a logo, place the company name or Web site name on the banner, centered. When designing a banner for a client with a logo but no company name, display the logo on the left with the company name immediately following. Leave enough space so that the banner does not look cluttered.
- ◆ **Tag line:** If the company has a tag line, display it below the company name. A tag line is a sentence usually less than ten words that describes what the company does. For example the tag line for Doug's wedding and event photography company Superb Images is: "Creating superb images of memorable moments." If the client has a logo with a company name, the tag line is displayed to the right of the logo,

centered vertically. If the company has a logo without a company name, the tag line is centered below the company name. The logo is sized to fill the height of the banner.

Working with an existing logo

When you design a banner — or, for that matter, a Web site — your client’s logo plays an important part in the overall design. If the logo is well designed, you can incorporate elements from the logo into your design. The banner and the Web site should include colors from the client’s logo. Figure 4-3 shows a logo being incorporated into a banner design in Photoshop. Figure 4-4 shows the banner incorporated with the Web design. If this book were in color, you’d be able to see that the site colors were pulled directly from the client’s logo.

Figure 4-3:
Incorporate a client’s logo with a banner design.

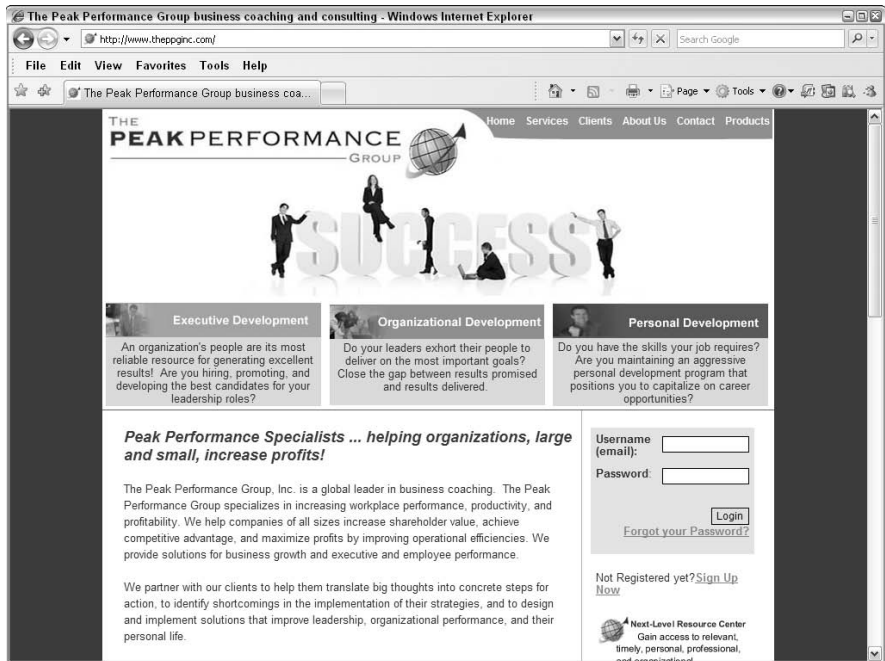


Figure 4-4:
Incorporate the banner with the site design.

Create Promotional Graphics

In today's business climate, a Web site often drives a small business. Therefore, the site often comes before any printed material is created to promote the business. When this is the case, it's logical to use elements from the site design to promote the business. For example, when Doug designed a Web site to promote his photography business, he originally started out with an artistic photo of a very vexing set of eyes. He changed the splash image after doing a ten-minute presentation for a local networking group. He showed a print of an image he stylized in Photoshop. The image was originally in color, but he converted it to black and white and painted some color from the original image into the subject's lips and eyes. The photo received a lot of oohs and aahs, so Doug now uses it on his home page. (See Figure 4-5.) This photo also appears in Doug's promotional materials. When recipients of the printed material visit the Web site, they immediately see a familiar face.

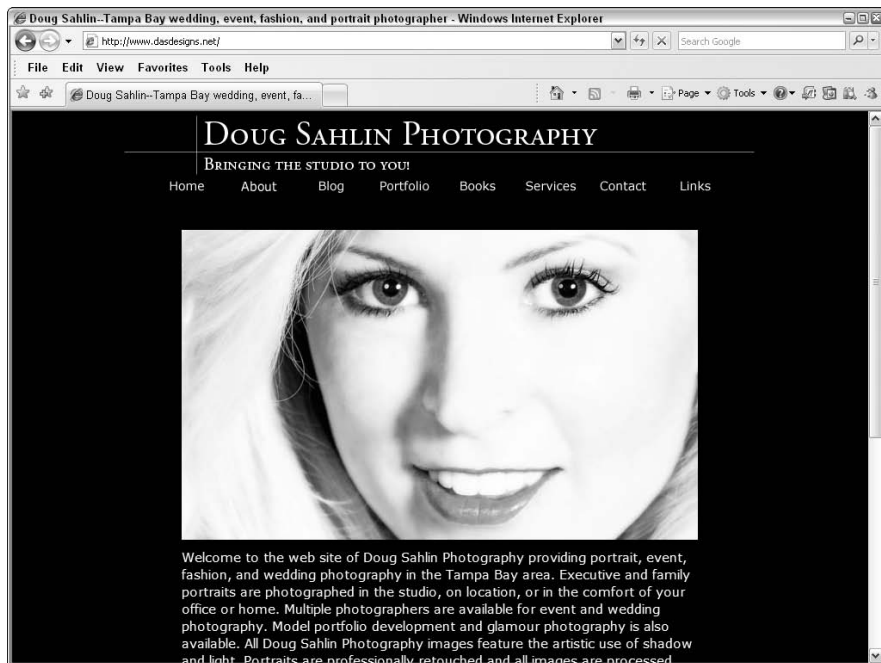


Figure 4-5:
Use
Web site
elements for
promotional
items.

Emphasize your point

When Doug decided to create promotional materials to increase his photography business, he knew that the Web site would play an integral part in his

marketing plans. When he embarked upon a local marketing campaign, he started with postcards. The front of the postcard features the image from the home page. (See Figure 4-6.)

Even though Doug did all the design work for his printed material, he consulted a marketing specialist when deciding what to emphasize on the postcard. She suggested emphasizing a particular service instead of listing all his services. Doug specializes in portrait photography on location. Therefore, he decided to focus on that aspect of the business in the first printed material he created. He laid out the back of the postcard in Adobe Illustrator, using bullet points to bring attention to his portrait photography service. He also added a *call to action* on the back of the card. A marketing guru had told him to give recipients a reason to call. The card has contact information including, of course, the URL to the section of Doug's site where recipients can find examples of his portrait photography. And, of course, his tag line appears below the company name (see Figure 4-7), just as on the Web site.




Figure 4-6:
Incorporate
design
elements
in printed
material.

Figure 4-7:
Use material from the Web site to drive home a point.

PORTRAIT PHOTOGRAPHY

Save time by having Doug Sahlin Photography shoot professional portraits in your office, on location, or in the comfort of your home.

- ✦ Individual or group portraits
- ✦ Professional studio lighting
- ✦ Color and/or Black & White images
- ✦ Images professionally retouched
- ✦ Images for the web and print
- ✦ High quality prints available
- ✦ Competitive prices
- ✦ Quality Assured



DOUG SAHLIN PHOTOGRAPHY
Bringing the studio to you!
Doug Sahlin Photography
(863) 648-5574
<http://www.portraits.dasdesigns.net>

PLACE
STAMP
HERE

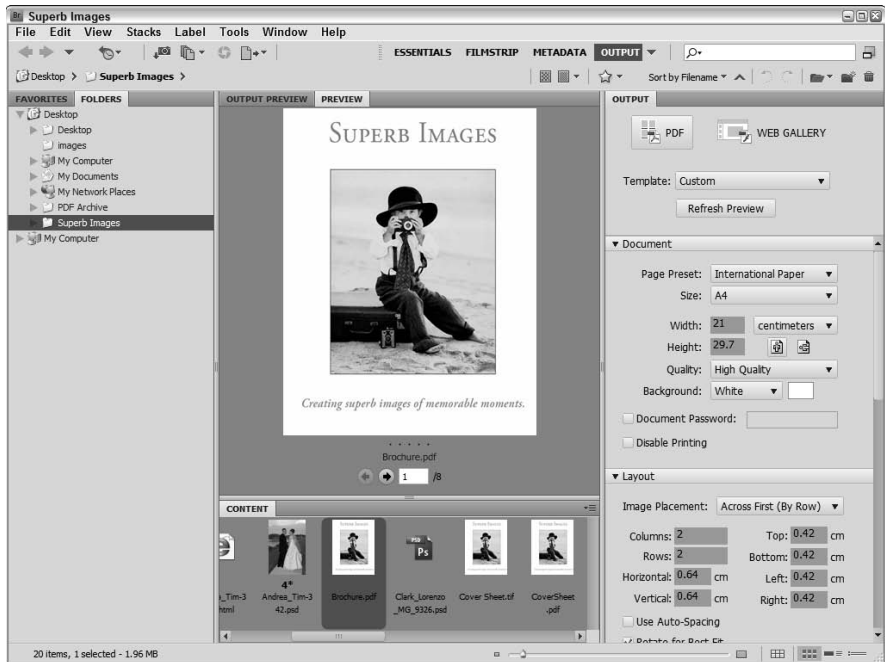
Complement your style

When you design promotional material to promote a business that's driven by a Web site, the style of the promotional material should be similar to the style of the Web site. The printed material should look as though it were pulled right from the Web design. Fortunately, this is a fairly easy task if you design the graphic elements of your Web site using the Adobe Creative Suite, which as of this writing is Version 4, or CS4. This powerhouse application enables the same elements you designed for the Web site in Photoshop to be used in other applications — such as Adobe InDesign (page layout application) or Adobe Illustrator.

All applications hinge around Adobe Bridge, which is available from InDesign, Illustrator, and Photoshop. Adobe Bridge (see Figure 4-8) enables you to choose saved items and open them in their host application. You can then modify the item for the application in which you'll use it. For example, you can resize a photo for use in a poster you're creating in Adobe Illustrator or for use in a brochure you're creating in InDesign. This capability to quickly access assets designed for the Web site in other applications lets you create printed material with the same style as your Web site.

When the time was right to create a brochure, Doug took advantage of the interoperability of Adobe Creative Suite applications. After consulting with his marketing guru, he created the graphic design for the brochure in Adobe InDesign, using high-resolution versions of images from the Web site. The brochure emphasized the portrait photography side of the business. The images for the brochure were picked from the portrait section of his Web site. Adobe Bridge made it easy to see the images ahead of time.

Figure 4-8:
Use Adobe
Bridge to
access
Web site
assets from
Adobe CS
applications.



The InDesign Place command (File→Place) places the image in the design. The Place command is very powerful because it allows you to shrink the original image to fit into a specific portion of your design. If you later decide to increase the size of the image, you can do so with no degradation. When you lay out a brochure using InDesign, you can take advantage of the powerful Text Wrap panel to wrap text around images in the design. (For all the ins and outs of using InDesign, check out *InDesign CS4 For Dummies* by Galen Gruman, Wiley.)

Creating Fast-Loading, Beautiful Photos

Photos are an integral part of any Web site. The trick is to display photos that look great and load quickly. You can always tell when you're at a site that wasn't created by a professional designer because the images are either huge and load slowly, or they load quickly but look bad. The next few sections show you how to resize and then optimize your images for the Web.

Resizing your photos for the Web

If you're working with images supplied by your client or images from a clip art disk, chances are that you're dealing with high-resolution images suitable for print. High-resolution images have large file sizes. The combination is totally unsuitable for display on a Web site. Granted, you can control the

size at which the image is displayed by entering the desired width and height in the `` tag. The file size is still the same, though, and the image will load slowly.

Images for the Web need a resolution of 72 pixels per inch (ppi). Some designers insist on 96 ppi, but this is more than you need. Images with a resolution of 72 ppi are sufficient for all modern computer monitors. Images for print have a resolution of 150 ppi or greater.

In addition to resolution, your next concern is image size. The size of the image is determined by where the image will reside. If you're posting the image inline with text, you size the image smaller than you would size an image for a photographer's gallery. Also consider the desktop size of your audience. Keep in mind that many people are still surfing the 'Net with an 800 x 600 pixel desktop size, which (with a maximized browser) has an available display area of 760 x 420 pixels. If you resize an image with *portrait orientation* (an image that's taller than it is wide) to 600 pixels in height, people with an 800 x 600 desktop size will have to scroll to see the entire image. When you resize images for inline display, a good practice is to never exceed 200 pixels in width or height.

If the image is displayed as a *splash image* (the main image on a page) on a home page or in an image gallery, we recommend never exceeding an area of 600 x 360 pixels. Granted, this gives the advantage to *landscape* (wider than tall) images, but the available display area in a Web browser is also wider than it is tall. You can resize images by using Fireworks or Photoshop CS4, as shown in the following steps.

To resize images in Fireworks, follow these steps:

1. Choose Modify → Canvas → Image Size.

The Image Size dialog box (see Figure 4-9) appears.

2. Enter a value of 72 in the Resolution field.

This doesn't change the print size, but it does reduce the pixel dimensions of the image.

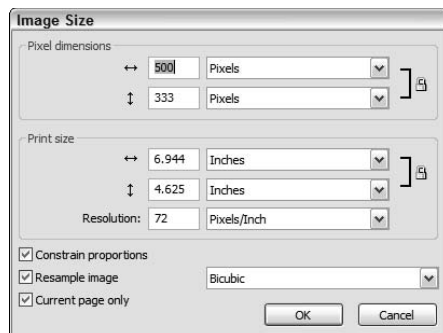


Figure 4-9:
Resizing an
image in
Fireworks.

3. Enter the desired width or height in the Pixel Dimensions section.

You can enter either value because the option to constrain proportions is selected by default. We advise that you never deselect the Constrain Proportions check box because the image will be distorted.

Downsizing (reducing in size) an image is okay, but when you try to *upsized* (enlarge) a small image, you're asking the image-editing application to increase the size of each pixel in the document. Image degradation is the unhappy result of trying to upsize an image.

4. Click OK.

Fireworks resizes the image to the desired dimensions.

To resize an image in Photoshop, follow these steps:

1. Choose Image⇨Image Size.

The Image Size dialog box (see Figure 4-10) appears.

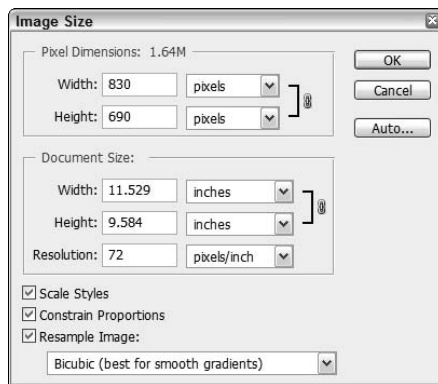


Figure 4-10:
Resizing an
image in
Photoshop.

2. Enter a value of 72 in the Resolution field.

This doesn't change the document size, but it does reduce the pixel dimensions of the image.

3. Enter the desired width or height in the Pixel Dimensions section.

You can enter either value because the option to constrain proportions is selected by default. We advise that you never deselect the Constrain Proportions check box because the image will be distorted.

4. Click OK.

Photoshop resizes the image.

Enlarging images with Alien Skin Blow Up

If you have clients who present you with small, low-resolution images that need to be enlarged, consider investing in the Blow Up plug-in made by Alien Skin Software (www.alienskin.com/blowup/index.html). Blow Up 2 works with Photoshop Elements 4.0.1 through Photoshop Elements 7 (Windows), and Photoshop CS2 9.0.2 or later. Adobe Photoshop Elements 6 is the latest release for Macintosh users. Doug has used the plug-in extensively to enlarge (blow up) client photos and his own images. The results are truly amazing. The

manufacturer claims you can enlarge a good-quality image four times (1,600 percent) without jagged edges, halos, or artifacts. The plug-in comes complete with an instruction booklet and can be used with a Macintosh PowerPC G5 processor running Mac OS X 10.4.11 or later, or with an Intel Pentium 4 processor (or compatible) running the Windows 2000, Windows XP, or Windows Vista operating system. You can download a fully functional demo with a 30-day expiration at www.alienskin.com/downloads/getmail1.asp.

Using professional optimization techniques

As long as you begin with a high-quality image, you can resize and optimize a photo for the Web in Adobe Photoshop or Fireworks. Optimizing an image in Fireworks is covered in Chapter 3 of this minibook. Photoshop CS4 also has a stout optimization algorithm. You can optimize images in the GIF format or the JPEG format.

Using the Photoshop Save for Web & Devices command

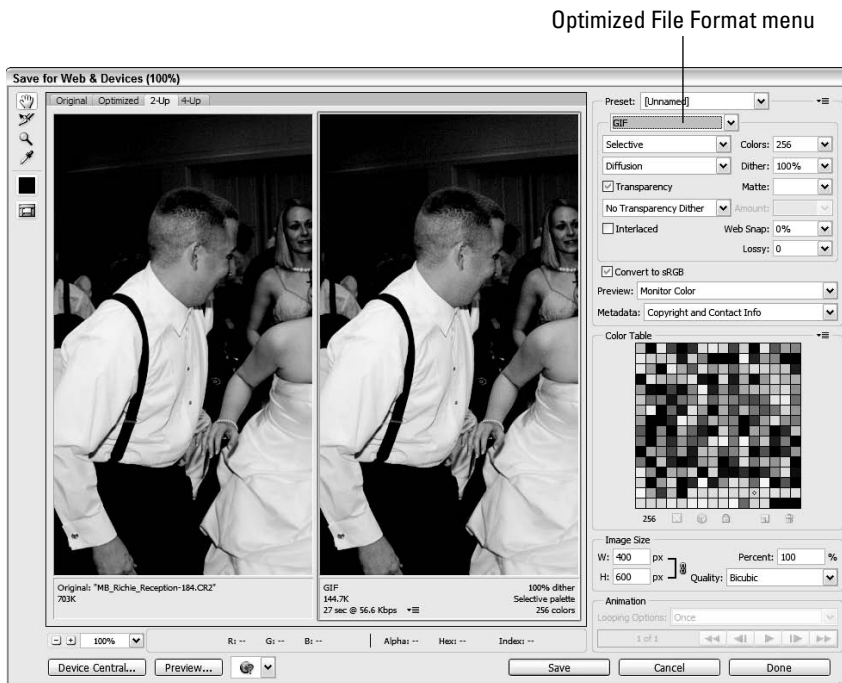
Photoshop CS4 gives you all the tools you need to create pixel-perfect images for print as well as for the Web. The Web has different considerations, though. First and foremost, you must choose the right file format. The next step is to optimize the image — or, if you've used the Slice tool, optimize each slice.

To optimize JPEG images you created in Photoshop CS4 for the Web, follow these steps:

- 1. Choose File⇨Save for Web & Devices.**
- 2. In the Save for Web & Devices dialog box that appears, click the 2-Up tab.**

This option enables you to compare the original image with the optimized version (see Figure 4-11).

Figure 4-11: Optimizing is so easy when you use the Save for Web & Devices command.



3. Select the Slice Select tool.

The tool allows you to select individual slices in the image and apply different optimization settings to each slice. Steps 3 and 4 are necessary only if you created slices in the image and intend to use different optimization settings for each slice.

4. In the optimized section (the bottom or right section when you're viewing the image in 2-Up mode), click the slice you want to optimize.

5. Choose the desired optimization setting from the Optimized File Format drop-down menu (see Figure 4-12).

For the first slice of this image (which is text), the ideal format is GIF, which is ideal for areas of an image that have large areas of solid color. We show you everything you need to know about optimizing GIF artwork in Photoshop in the next section. For more information on file formats, see Chapter 1 of this minibook.

6. To optimize an image using the JPEG format, choose JPEG from the Optimized File Format drop-down menu.

When you choose JPEG, the options shown in Figure 4-13 appear. The default JPEG setting is Maximum when you create an image from scratch in Photoshop; and the Quality is 100 percent, which results in a squeaky-clean-looking image but is way too big for a Web site. You can modify the Quality setting and other applicable settings as outlined in the next step.

Figure 4-12: Choose the desired optimization method.

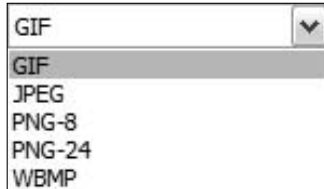
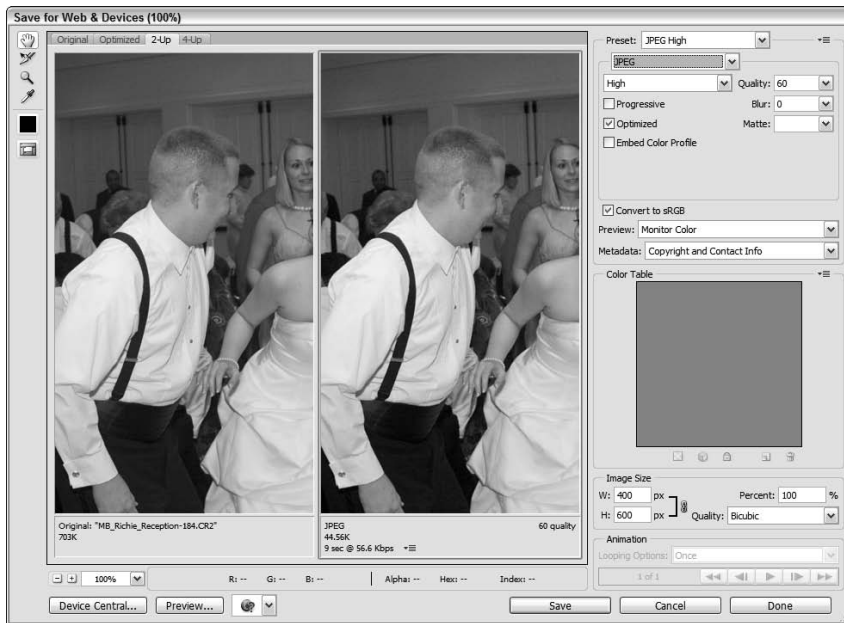


Figure 4-13: Optimizing an image, using the JPEG format.



7. Modify the following parameters if desired:

Quality: Enter a value or click the arrow and drag the scrubby slider. Higher values apply less compression, resulting in a better-looking image albeit at the expense of a larger file size. Lower values apply higher compression, resulting in a smaller file size with poorer image quality. You can tell the optimal setting for an image by comparing the original in the left pane if your image is portrait (as shown in Figure 4-13), or in the bottom pane if your image is landscape, with the compressed version in the right pane. When image degradation is noticeable, specify a higher value.

Progressive: Downloads the image into the user’s browser in multiple passes. This option enables the visitor to see a low-resolution version of the image as soon as the page loads. The optimized version appears when the image finishes loading.

Blur: If you have to apply heavy compression to achieve the desired file size, enter a value or drag the Blur scrubby slider to blur the image and minimize the appearance of jagged edges.

Matte: Specifies a color to fill pixels that were transparent in the original image. Choose an option from the Matte drop-down menu. Your choices are

- *None:* Choose this option when you don't want to specify a matte color.
- *Eyedropper:* Choose this option, and then click the Matte Color Swatch below the Eyedropper tool on the left side of the dialog box to choose the fill color from the Color Picker.
- *Foreground Color:* Choose this option and the current foreground color becomes the matte color.
- *Background Color:* Choose this option and the current background color becomes the matte color.
- *White:* Fill transparent pixels with white.
- *Black:* Fill transparent pixels with black.
- *Other:* With this, you can select a matte color from the Color Picker.

Embed Color Profile: Embeds the current color profile with the saved file. We recommend against this option; leave it unchecked and use the Convert to sRGB option described in the next step. If the current color profile is anything other than sRGB and you embed it, the image won't display properly on a computer monitor.

8. Make sure you leave the Convert to sRGB option enabled.

If your image was captured from a digital camera using the Adobe RGB color profile, or if you converted the image to the Adobe RGB color profile, you definitely need to have the resulting image converted to sRGB because the Adobe RGB color profile cannot accurately be displayed outside Photoshop CS4.

9. Choose an option from the Preview drop-down menu.

Your choices are Monitor Color, Macintosh (No Color Management), Windows (No Color Management), or Use Document Profile. In most cases, the default Monitor Color options works best.

10. Choose an option from the Metadata drop-down menu.

Your options are None, Copyright, Copyright and Contact Info, All Except Camera Info, or All. Digital cameras add *metadata* to an image when a picture is taken. The information includes the camera make and model, exposure information, and so on. The photographer might have also added copyright and contact info in Photoshop, Photoshop Lightroom, or a similar image-editing application. If the image was supplied by a photographer, leave his contact and copyright image with the optimized image. Adding the camera data isn't necessary.

11. Choose the desired options in the Image Size section.

You can resize the image proportionately by entering a value in the W or H text field. We don't advise clicking the link icon, which makes it possible to resize the width and not the height, or vice versa. This distorts the image. You can also resize the image by change the value in the Percent text field, or by dragging the Percent scrubby slider. If you resize the image, you can choose an option from the Quality menu. In most cases, the Bicubic option works well.

12. Click Save.

The Save Optimized As dialog box appears.

13. Choose the desired option from the Save As Type drop-down menu.

If you're saving an image for a banner, accept the default Images Only option. If you sliced the image, choose HTML and Images to save an HTML file and image files in the specified file formats.

To optimize GIF images in Photoshop CS4, follow these steps:

1. Choose File → Save for Web & Devices.

The Save for Web & Devices dialog box appears.

2. Select an option from the Preset drop-down menu.

Your choices are GIF Dithered; GIF No Dither with a palette of 128, 64, or 32 colors; or GIF Restrictive. If you created your design using colors from the Web-safe palette, choose one of the No Dither options. Choose one of the Dithered options to export the image with Web-safe colors. Any non-Web-safe colors in your design are mixed (dithered) using colors from the Web-safe palette. GIF Restrictive pulls colors from the Web-safe, 216-color palette. Figure 4-14 shows the Save for Web & Devices dialog box after choosing the GIF 128 Dithered preset.

3. Select an option from the Color Reduction Algorithm drop-down menu.

The Color Reduction Algorithm determines the method by which the colors for the image palette are created. Your choices are

- *Perceptual*: Creates a color table by giving priority to colors for which the human eye has greater sensitivity.
- *Selective*: The default option creates a color table similar to that created by using the Perceptual algorithm, but it preserves large areas of color in the original image and also Web-safe colors. This algorithm creates a color palette that closely resembles the original image.

Figure 4-14: Optimizing an image using the GIF 128 Dithered option.



- *Adaptive:* Creates a color table using the predominant color spectrum in the image. If you're optimizing an image that's predominantly red and orange, the color table is made up primarily of reds and oranges. Choose this algorithm if the image you're optimizing is made up of similar color hues.
- *Restrictive:* Creates a color palette using colors from the Web-safe, 216-color palette.
- *Custom:* Creates a color palette that you create or modify. If you open a previously saved GIF or PNG-8 file, it will have a custom color palette.
- *Black and White:* Creates a color palette comprising black and white. Use this algorithm to save images that will be displayed on PDAs (personal digital assistants) that can't display color.
- *Grayscale:* Creates a color palette comprising 256 shades of gray from solid black to white. This is similar to how black-and-white film photos display.

- *MAC OS*: Creates a color palette based on the colors from the Mac operating system (OS) color palette.
- *Windows*: Creates a color palette based on the colors from the Windows OS color palette.

4. Modify the number of colors in the palette.

You can modify the number of colors by choosing an option from the drop-down menu. When you change the number of colors, you see the results in the Color Table. When you reduce the number of colors in the palette, pay careful attention to the optimized image. If you specify a color palette that doesn't have enough colors, image degradation occurs. If this is the case, choose a higher value that doesn't result in image degradation.

5. Choose an option from the Dither Algorithm drop-down menu.

Dithering creates a facsimile of colors in the image but not in the palette by mixing colors in the palette. Your choices are as follows:

- *Diffusion*: Dithers by creating a random pattern that, in most cases, is less noticeable than Pattern dither. Dithering is diffused across adjacent pixels.
- *Pattern*: Dithers by creating a pattern that looks like halftone squares to simulate image colors not in the color table.
- *Noise*: Dithers by creating a random pattern similar to the Diffusion method, but the pattern is not diffused across adjacent pixels. Seams don't show if you use this dithering method.

6. Specify a Dither value.

This option determines the percentage of colors that are dithered. A higher value gives you greater detail at the expense of a higher file size. Drag the slider to the left until you notice image degradation in the bottom pane of the dialog box, and then drag the slider to the right until the image looks better. This will be the optimal dithering value for the image you're optimizing.

7. Transparency is selected by default. Deselect this option if you don't want to optimize the image with areas of transparency.

Transparency is used when you want areas of the image to be transparent. This option is most often used when you want areas of a tiled background image to show through the image you're optimizing. However, even though transparency is enabled by default, you have to choose the transparent colors by specifying Matte and Transparency colors.

8. If desired, choose an option from the Transparency menu.

You have the following options:

- *No Transparency Dither*: Doesn't apply dithering to partially transparent pixels in the image.

- *Diffusion Transparency Dither*: Dithers by applying a random pattern that is, in most cases, less noticeable than Pattern dither. Dithering is diffused across adjacent pixels. When selecting this option, specify a Dither percentage to control the amount of dithering applied to the image.
- *Pattern Transparency Dither*: Dithers by applying a halftone-like square pattern to partially transparent pixels.
- *Noise Transparency Dither*: Dithers by applying a random pattern similar to the Diffusion color-reduction algorithm, but the pattern isn't diffused across adjacent pixels. Seams don't show if you use this dithering method.

9. The Amount option becomes available if you select Diffusion Transparency Dither.

Drag the slider to specify the percentage of dithering applied to the image. Alternatively, you can enter a value in the text field.

10. Choose the Interlaced option to download the image in stages.

This option displays a low-resolution version of the image when the page loads. The fully optimized version is revealed when the image loads completely.

11. Specify the Web Snap value.

This option determines the amount of non-Web-safe colors that are shifted to Web-safe colors. Choosing a higher value snaps more colors from the original image to the nearest Web-safe counterpart.

12. Drag the Lossy slider to reduce file size.

This option discards color data to reduce file size. If you use this option, pay careful attention to the image in the right pane of the Save for Web & Devices dialog box. When the image begins to degrade, drag the slider to the left until you no longer see the degradation.

You can't use the Lossy option if you use the Interlaced option, or with the Noise or Pattern Dither option.

13. Make sure you leave the Convert to sRGB option enabled.

This is the optimal color profile for monitor viewing.

14. Choose an option from the Preview drop-down menu.

Your choices are Monitor Color, Macintosh (No Color Management), Windows (No Color Management), or Use Document Profile. In most cases, the default Monitor Color option works best.

15. Choose an option from the Metadata drop-down menu.

Your options are None, Copyright, Copyright and Contact Info, All Except Camera Info, or All. Digital cameras add metadata to an image when a picture is taken. The information includes the camera make



and model, exposure information, and so on. The photographer might have also added copyright and contact info in Photoshop, Photoshop Lightroom, or a similar image-editing application. If the image was supplied by a photographer, leave his contact and copyright image with the optimized image. Adding the camera data isn't necessary.

16. Choose the desired options in the Image Size section.

You can resize the image proportionately by entering a value in the W or H text field. We don't advise clicking the link icon, which makes it possible to resize the width and not the height, or vice versa. This distorts the image. You can also resize the image by changing the value in the Percent text field, or by dragging the Percent scrubby slider. If you resize the image, you can choose an option from the Quality menu. In most cases, the Bicubic option works well.

17. Click Save.

The Save Optimized As dialog box appears.

18. Choose the desired option from the Save as Type drop-down menu.

If you're saving an image for a banner, accept the default Images Only option. If you sliced the image, choose HTML and Images to save an HTML file and image files in the specified file formats.

19. Click Save.

Photoshop saves the image in the GIF format using the options specified.



The Color Table at the bottom of the dialog box shows swatches of the colors that comprise the image color palette. You can modify the palette by adding colors to the palette, snapping colors to the nearest Web-safe color, and defining transparent pixels.

Creating a Photo Gallery

Many Web sites feature photo galleries. You can create a photo gallery the old-fashioned way by creating thumbnails and optimized full-sized images in an image-editing application, such as Photoshop or Fireworks. If you do this, you have to house all the thumbnails in an HTML table and create the links to the full-sized versions. If you do it right, the full-sized photos should be in their own HTML document. If that sounds like a lot of work, you're right. If you have Photoshop or Dreamweaver, you can create a photo gallery with a menu command.

Choosing your photos

If you're creating a Web site for a client, your assets should all be neatly assembled in a folder. If you're creating a Web gallery for yourself, your photos should also be in a folder. The easiest way to choose which photos you want to include in a Web gallery is to use an application like Adobe

Bridge, or any application that lets you view a folder of photos as thumbnail images. The following steps show you how to choose your photos using Adobe Bridge:

1. Launch Adobe Bridge.

Navigate to the folder that contains the candidates for your Web gallery.

2. Choose Window⇨Workspace⇨Light Table.

Adobe Bridge displays all images as thumbnails. (See Figure 4-15.)

3. Examine all the images in the folder.

4. When you see an image you want to include in the gallery, select the image and press Ctrl+5 (Windows) or ⌘+5 (Mac).

This assigns a five-star rating to the photos. Continue assigning five-star ratings to images you want to include in the Web gallery.

5. Press Ctrl+Alt+5 (Windows) or ⌘+Option+5 (Mac).

Adobe Bridge displays only the five-star images. (See Figure 4-16.)

6. Choose File⇨New Folder.

Adobe Bridge creates a new folder.



Figure 4-15:
Separate
the wheat
from the
chaff.



Figure 4-16: Displaying the best images.

7. Name the folder Web Gallery.
8. Select the images and drag them into the Web Gallery folder.

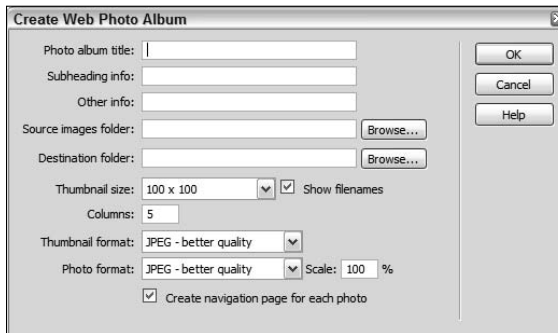
Using Dreamweaver to create a photo gallery

If you prefer, you can create a Web gallery in Dreamweaver. When you choose this option, Fireworks creates the thumbnails and full-size images. When you create a Web gallery in Dreamweaver, your only option to resize the images is by using a percentage. We strongly recommend that you use Fireworks or Photoshop batch processing to resize the images to the desired size. To create a photo gallery in Dreamweaver, follow these steps:

1. In Dreamweaver, create a new document.
2. Choose **Commands** > **Create Web Photo Album**.
The Create Web Photo Album dialog box (see Figure 4-17) appears.
3. Enter the following information: **Photo Album Title, Subheading Info, and Other Info**.

This is the information that will display for the title of your album. For the Subheading and Other Info, you might enter your client's name and contact information.

Figure 4-17:
Yet another
way to
create a
Web gallery.



4. Choose the Source Images and Destination folders.

Click the Browse button to navigate to the folder in which the source images are stored, and then click the Browse button to navigate to the folder where the finished photo gallery will be stored.

5. Select an option from the Thumbnail Size drop-down menu.

This determines the size of the thumbnail that Fireworks creates.

6. Accept the default option to show filenames.

Doug always deselects this option unless the photo gallery is for a photographer whose client will be choosing which images she wants to purchase. The only time you should select this option is when the filename is important, such as when the filename needs to be present in order to identify a photo.

7. Accept the default value of 5 for Columns, or enter another value.

This determines how many columns are created in the HTML page that houses the thumbnails.

8. Choose options from the Thumbnail Format and Photo Format drop-down menus.

These options enable you to choose the image quality and file format. When you're creating a photo gallery, the obvious choice is JPEG. If your client's intended audience accesses the Internet with high-speed connections, choose JPEG-Better Quality.

9. Accept the default Scale value of 100 percent, or enter a different value.

The default displays the full-size images at their original sizes. Enter a smaller value to resize the images. Entering a larger percentage is possible, but not advised because image degradation occurs.

10. Accept the default Create Navigation Page for Each Photo.

This option creates an HTML page for each image with navigation buttons to go forward, back, or to the home page of the image gallery.

11. Click OK.

Fireworks launches and creates the optimized full-size images and thumbnails, and Dreamweaver handles the task of creating the HTML documents for the image gallery. Figure 4-18 shows the finished gallery as displayed in the Firefox Web browser.

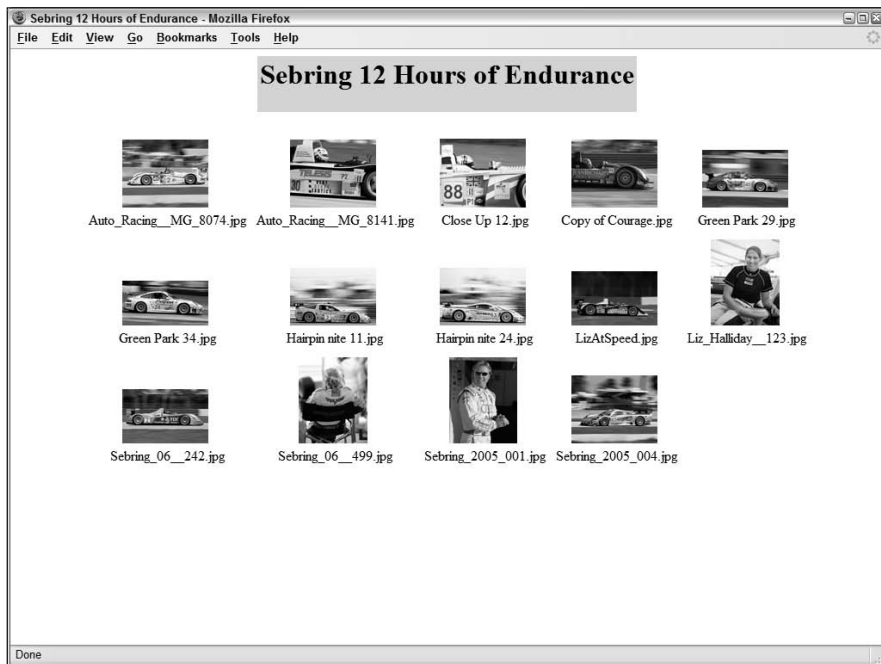


Figure 4-18:
A Dreamweaver Web gallery in action.

Book V

Multimedia

The 5th Wave

By Rich Tennant



"You ever notice how much more streaming media there is than there used to be?"

Contents at a Glance

Chapter 1: Incorporating Flash Creations	393
Introducing Flash CS4	393
Building Flashy Navigation	409
Flash as an Animation Tool	420
Create a Flash Presentation	421
Exploring the Many Tricks of Flash.....	422
Chapter 2: Using Sound	423
Exploring Audio Formats for the Web	423
Adding Flash Audio to a Page	424
Embedding a QuickTime Player in a Page.....	429
Adding the RealMedia Player to a Page.....	431
Adding the Windows Media Player to a Web Page.....	433
Delivering Your Message.....	436
Chapter 3: Using Web Video	437
Exploring Web Video Formats	437
Working with Digital Video.....	438
Creating and Encoding Flash Video	442
Adding Flash Video to a Web Page.....	447
Adding QuickTime Player to a Page.....	449
Adding Windows Media Player to a Page	451
Avoiding Digital Video Pitfalls.....	453
Chapter 4: Active Slideshow Pro	455
About Active Slideshow Pro.....	455
Installing the Extension	456
Creating a Slide Show.....	457

Chapter 1: Incorporating Flash Creations

In This Chapter

- ✓ Adding Flash interactivity
- ✓ Adding Flash navigation to a Web site
- ✓ Creating Flash animations
- ✓ Creating a Flash presentation

If you need to kick up a Web design with some multimedia, Flash is an excellent tool to use. With a bit of imagination, you can use this powerful tool to create Web sites that make those of your competition look positively lame. This chapter introduces you to the all-singing, all-dancing Flash CS4 workspace. Here, you find out how to use it to create flashy navigation in a flash (almost the same duration as a New York minute), add animation to your designs, and produce stellar presentations. Let the games begin.

Introducing Flash CS4

Historically, Flash has been used to create animated introductions to Web sites and full-fledged Web sites. However, Flash intros are no longer in vogue, and Flash Web sites are not search engine friendly. Despite all this, Flash is as strong as ever; Web designers just use it in different ways.

Designers can control the user's experience in a Web site by how they place items on the Flash Timeline. They can also control the flow of the Flash movie via ActionScript. Flash navigation menus consist of buttons that can transport the visitor to other parts of the Web site.

Whether you're creating an animation or a menu in Flash, the end result is an SWF file, which is also known as a movie. When constructed properly, Flash movies are relatively small files. Yet a designer can go over the top by adding so many graphics and other bells and whistles that the finished product loads at a snail's pace.

The tempo or pace of a Flash movie is defined by the frame rate. The default frame rate of a Flash movie is 24 frames per second (fps), which is fine for most Flash movies. You can also incorporate Flash video (FLV file format) in a Flash project or in an HTML Web page. For more information on Flash video and other Web video formats, see Chapter 3 of this minibook.

Touring the interface

The Flash interface might seem a little daunting to the uninitiated. We certainly had our doubts when we first launched Flash 4. However, the workspace is really quite civil after you work with Flash for a while. To know Flash in all its glory, you need a good book on the topic, such as *Building Flash Web Sites For Dummies* (by Doug), *Flash All-in-One Desk Reference For Dummies* (Doug again, with Bill Sanders), or *Flash CS4 For Dummies* (Ellen Finkelstein and Gurdy Leete; all from Wiley).

To start the tour, we bring you up to speed on the various parts of the Flash interface (see Figure 1-1) in the upcoming sections.

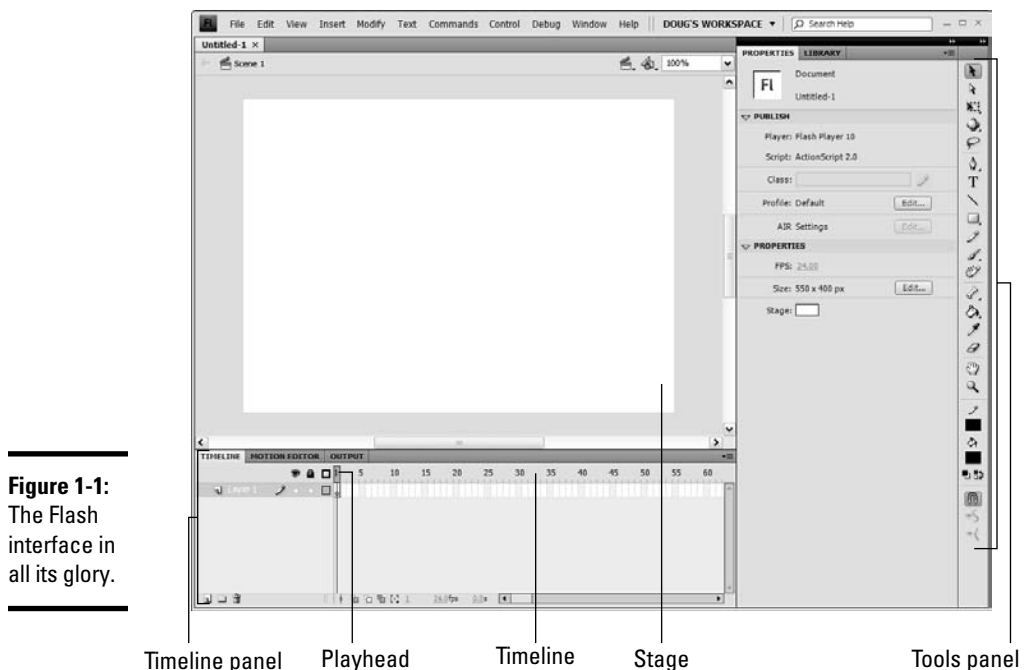


Figure 1-1:
The Flash
interface in
all its glory.

Exploring the Tools panel

The right side of the interface is home to the Tools panel, which is shown in Figure 1-2. This part of the interface is home to the tools you use to create objects and text. For the purpose of this illustration, the Tools panel is floating in a two-column format. You can also use the tools to move and modify objects, or you can choose a different workspace from the drop-down menu to the right of the menu groups. Some of the less frequently used tools share space in the Tools panel. When you see a right-pointing down arrow, you've found a tool that's subletting space to another. Click the arrow to reveal a fly-out menu that shows more tools.

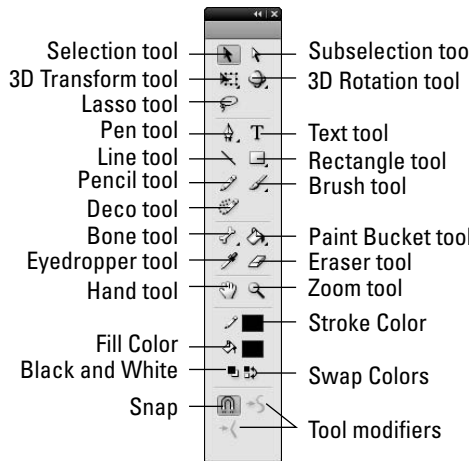


Figure 1-2:
Tools you
use for fun
and profit.

The following tools reside in the Tools panel:

- ◆ **Selection tool:** Select and moves objects.
- ◆ **Subselection tool:** Select points along a path and objects within a group.
- ◆ **Free Transform tool:** Scale objects; change the width or height of an object; skew an object; change an object's center of rotation; rotate an object.
 - **Gradient Transform tool:** Transform a gradient that fills an object. This tool shares space on the Tools panel with the Free Transform tool.
- ◆ **3D Rotation tool:** Take objects for a spin. Flash isn't true 3-D yet, but you can simulate 3-D in an animation with this tool.
 - **3D Transform tool:** This tool has three axes — x, y, and z — that you use to move an object left and right, up and down, forward and backward. This tool makes it possible for you to simulate 3-D in a motion tween animation. A motion tween animation is one where an object moves from Point A to Point B and Flash creates the objects in the in-between frames. This tool shares space with the 3-D Rotation tool in the Tools panel.
- ◆ **Lasso tool:** Create a freeform selection.
- ◆ **Pen tool:** Create a path. You can choose other tools from the fly-out menu that enable you to add points to or delete points from a path. There is also a tool to convert a straight point to a curve point and vice versa.

- ◆ **Text tool:** Add text to a document.
- ◆ **Line tool:** Create lines.
- ◆ **Oval tool:** Create ovals. Hold down the Shift key while using this tool to create a circle. The Oval tool shares a one-story walkup with the Rectangle, Rectangle Primitive, and Oval Primitive tools.
- ◆ **Rectangle tool:** Create rectangles. Hold down the Shift key while using this tool to create a square. You can add a radius to rectangle corners to create rectangles with round corners by clicking the Set Corner Radius button in the Options area of the Tools panel.
 - **Rectangle Primitive tool:** This critter is also used to create rectangles and rounded rectangles. The beauty of a primitive shape is that you can edit the shape's attributes at any time using the Property inspector.
 - **Oval Primitive tool:** Create ovals that can be edited in the Property inspector any time after they are created.
- ◆ **Pencil tool:** Create freeform lines in the document.
- ◆ **Brush tool:** Paint objects within the document.
 - **Spray Brush tool:** Use to add graffiti-like splashes of color to a document. You can change the color that sprays from the nozzle, or spray a symbol out the tool's nozzle from the document Library. This tool shares the rent with the Brush tool.
- ◆ **Deco tool:** Use to draw a shape that looks like a flowering vine. You specify the colors in the Property inspector, and you can also choose a symbol from the document Library for the leaf and the vine.
- ◆ **Bone tool:** Use to create an Inverse Kinematics (IK) animation. You use the tool to add bones to the object you're animating. The bones are the Inverse Kinematics chain. You yank a bone with the Selection tool, and the bones that are higher in the chain move.
- ◆ **Bind tool:** Use this to fine-tune an IK animation that's not performing up to snuff — “bound up,” if you will. You use the tool to bind points from an object with a bone.
- ◆ **Polystar tool:** Create multisided polygons and stars. The Polystar tool appears on a fly-out when the arrow in the corner of the Rectangle tool is clicked. When the Polystar tool is selected, it appears on the Tools panel; the Rectangle tool is now on the fly-out.
 - **Ink Bottle:** Change the color of an object's stroke (outline). The Ink Bottle and Paint Bucket share space in the Tools panel.



- ◆ **Paint Bucket:** Change an object's fill, which can be a solid color or gradient.
- ◆ **Eyedropper:** Sample a color from an object within the document.
- ◆ **Eraser:** Erase an area or object within the document.
- ◆ **Hand tool:** Pan within the document.

While using any tool, press and hold down the spacebar to momentarily activate the Hand tool. Release the spacebar to revert to the previously used tool.

- ◆ **Zoom:** Zoom in within the document. Click to zoom to the next highest level of magnification. Click and drag to zoom to a specific area of the document. Press the Alt key (Windows) or Option key (Mac) and then click inside the document to zoom out to the next-lowest level of magnification.
- ◆ **Stroke Color:** Click the swatch to open the Color Picker and then choose a stroke color. This color is applied to all the tools that create objects with a stroke.
- ◆ **Fill Color:** Click the swatch to open the Color Picker and then choose a fill color. This color is used by all the tools that create objects with a fill.
- ◆ **Black and White:** Click this icon to revert to the default stroke color (black) and fill color (white).
 - **No Color:** Click the Fill Color swatch to open the Swatches panel, and then click this icon to create an object with no fill. Or, click the Stroke Color swatch and then click this icon to create an object with no stroke.
- ◆ **Swap Colors:** Swap the current stroke and fill colors.
- ◆ **Options:** This part of the Tools panel changes depending on the tool you select. For example, when you choose the Brush tool, you are presented options that specify how the tool disperses color, the brush size, tip, and so on.

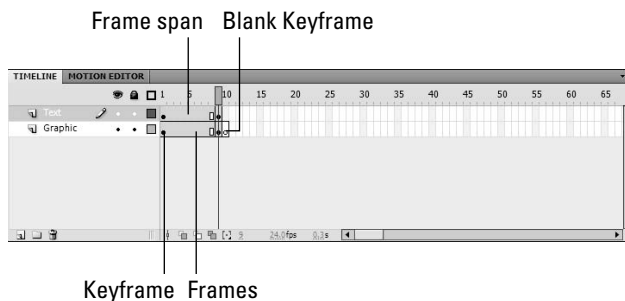
Someone told me it's all happening on the Timeline

You use the Timeline to control the tempo of your Flash movie. The Timeline consists of frames. You add keyframes where you want a change to occur. When you create an animation, you add a keyframe and then move the object to a different position.

You create blank keyframes to signify the end of a set of frames. For example, if you're creating a slide show, you add an image to a keyframe where you want the animation to start. The frames downstream from the keyframe duplicate the contents of the keyframe — in this case, the image. The number of frames signifies the length of time for which the image is displayed. We don't show you how to create an animation just yet — that comes later, in the “Creating your first animation” section — but we get you started:

- ◆ **To create a keyframe, click a regular frame and press F6.** Alternatively, choose Insert⇨Timeline⇨Keyframe. A keyframe with objects is signified by a solid dot. (See Figure 1-3.)
- ◆ **To create a blank keyframe, click a regular frame and press F7.** Alternatively, choose Insert⇨Timeline⇨Blank Keyframe. A blank keyframe is signified by a hollow dot. (See Figure 1-3.)
- ◆ **To create a regular frame, press F5.** Alternatively, choose Insert⇨Timeline⇨Frame. Frames duplicate the content of the previous keyframe. The area between keyframes is a *frame range*. The last frame in a frame range is signified by a rectangle. (See Figure 1-3.)

Figure 1-3:
Frames,
keyframes,
and blank
keyframes.



Panels not cast from wood

You use panels to do a lot of your work in Flash. The three panels on the right side of the interface are the Property inspector, the document Library, and the Tools panel. You open other panels using commands on the Window menu. When you use a menu command to open a panel (for example, Window⇨Color), it appears in the workspace as a floating panel. (See Figure 1-4.) The Flash interface is customizable. If you don't like the default layout, choose another preset. You can also modify any preset workspace to suit your working preferences.



Many of the panel names are self explanatory, whereas others require a little clarification. You can use Flash Help (F1) or pick up a copy of the tremendously helpful and infinitely amusing *Flash CS4 For Dummies* by Ellen Finkelstein and Gurdy Leete (Wiley).

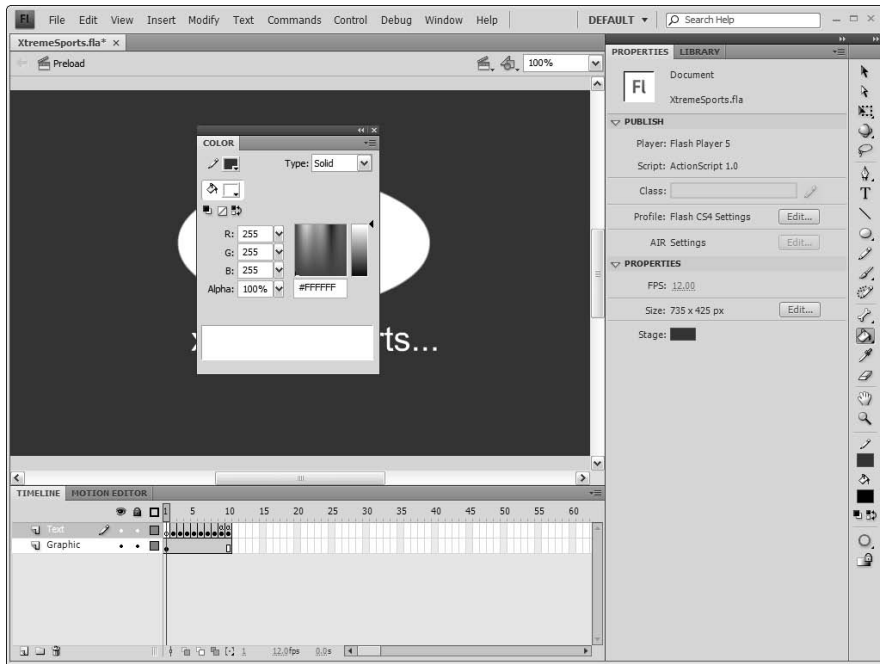


Figure 1-4: Panels can be docked or can float in the workspace.

Inspecting the Property inspector

Objects have properties. It's a Flash law. When you create an object, such as the lowly and infinitely rotund oval with the Oval tool, you don't have to take what you get. You can change the properties of the oval by using the Property inspector. Figure 1-5 shows the Property inspector as it's configured when an oval is selected. Notice you can change the width, height, x coordinate, y coordinate, stroke, fill, and more. Talk about your useful inspectors.

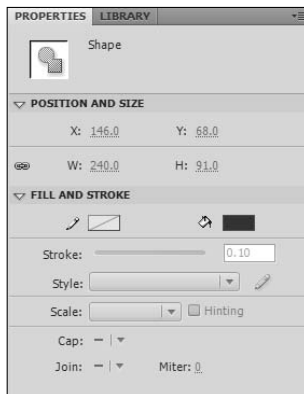
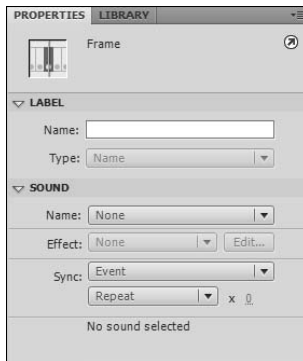


Figure 1-5: The Property inspector at work, sans monocle and magnifying glass.

You also use the Property inspector with other objects. Keyframes also have properties. As we mention previously, keyframes are used in animation. You can set several parameters for animation by selecting a keyframe and then opening the Property inspector. (See Figure 1-6.) Notice the options that add sound to a keyframe and give the keyframe a label. A keyframe with a label — isn't that like giving it a name? Yikes!

Figure 1-6:
The
Property
inspector is
used when
creating an
animation.



Getting comfortable with Flash

Flash is big, huge, vast. Even if you learn just half the bells and whistles in the application, Flash will never be half-vast. But we digress. One chapter just isn't long enough to learn all that Flash has to offer. To do so, you have to get your fingers a little greasy — get under the hood, so to speak. However, the next sections show you the fundamentals for creating a new document, an object, and then an animation. Buckle up and enjoy the ride.

Creating a Flash document

When you launch Flash CS4 for the first time, the first thing you see is a welcome screen that gives you options to create new documents or view tutorials. We generally look at welcome screens once, and click anything that piques our curiosity. The next time we launch Flash and the pesky critter appears, we mark the Don't Show Again check box to make the welcome screen exit stage left. Forever.

When you launch Flash, the first thing you want to do is create a document. The interface is pretty, but it won't do you a bit of good without a document. You can create several types of documents. In this section, we show you how to create a basic document with the default document size, frame rate, background color (refrigerator white, which is boring), and so on. It's a good place to start, but you generally need to modify the document to suit the Web page in which the Flash movie will be embedded. We're also assuming that you disabled the welcome screen and are using Flash like the frequent Flash fliers.

To create a Flash document, follow these steps:

1. Launch Flash and choose File⇨New.

The New Document dialog box appears (see Figure 1-7).

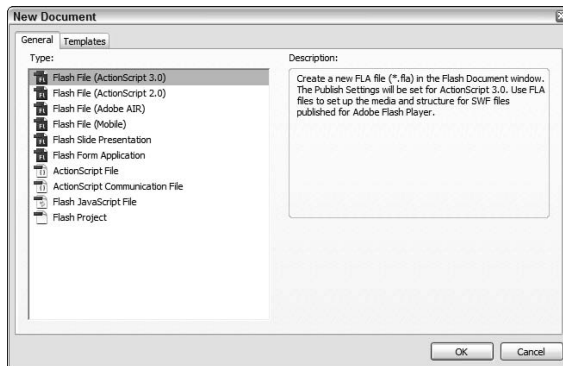


Figure 1-7:
It's time
to create
a new
document.

2. Click Flash File.

For a Flash document you're creating for the Web, choose one of the first two options. ActionScript 3.0 was introduced with Flash CS3. It's a robust language, but is not appropriate for casual Flash users. We also don't have the space to give you a proper introduction to ActionScript 3.0. For the purpose of this exercise, create a Flash File (ActionScript 2.0).

Flash creates a new document with the default document size of 550 x 400 pixels, the default frame rate of 24 fps, an appliance-white background color, and the default publishing parameters. You can easily change any or all of these parameters via the Property inspector.

3. Open the Property inspector.

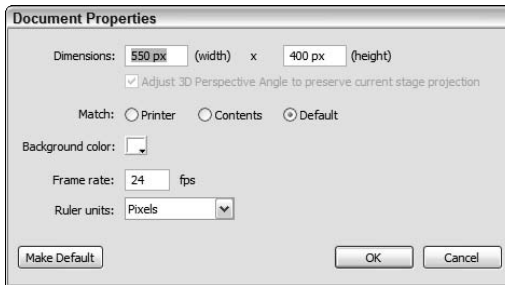
The default location of the Property inspector is on the right side of the interface. If the Property inspector isn't visible in the workspace, open it by doing any of the following:

- Click Properties on the title bar of the Property inspector.
- Choose Window⇨Properties.
- Press Ctrl+F3 (Windows) or ⌘+F3 (Mac).

4. In the Size section, click the Edit button.

The Document Properties dialog box appears (see Figure 1-8).

Figure 1-8:
Set
document
properties
here.



5. Type the desired values in the width and height fields.

The default document size is 550 pixels by 400 pixels. If you're creating a small animation for a banner, you may end up with a document size of 150 x 90, or something similar. If you're creating a Flash Web site optimized for a 1025 x 768 desktop, you'll create a document that's 955 x 600 pixels.

6. Select an option in the Match section:

- Choose *Printer* to set the Stage size to the maximum printable area, which is determined by the paper size minus margins as specified in the Page Setup dialog box (Windows) or the Print Margins dialog box (Mac).
- Choose *Contents* to set the Stage by putting equal space around the contents on all sides. Before choosing this option, align all elements at the upper-right corner of the Stage.
- Choose *Default* to set the Stage size to that specified in Step 5. Use this option when you're creating Flash for the Web.

7. Click the Background Color swatch.

Flash opens the Swatches panel shown in Figure 1-9.

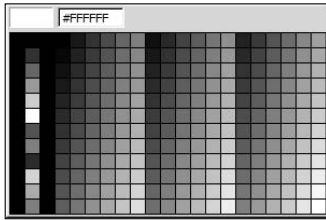
8. Select the desired background color.

You can choose a color by clicking a swatch. The colors in the default palette are from the Web-safe, 216-color palette. Alternatively, you can enter the hexadecimal value for the color in the text field. This determines the background color of the Flash movie as well as the background color of the HTML document in which the Flash movie is embedded.

9. Back in Document Properties, type the desired frame rate in the Frame Rate field.

The default frame rate of 24 fps works well for most Flash movies. If you're streaming video to an audience that has a high-speed cable connection, you can specify 30 fps and deliver beautiful full-motion video.

Figure 1-9:
Please, pick
anything but
white.



10. Choose the desired unit of measure from the Ruler Units drop-down menu.

The default unit of measure is Pixels, which works well in most instances. Your alternatives are Inches, Inches (decimal), Points, Centimeters, and Millimeters.

11. Click the Make Default button if you want these parameters to be the defaults for future Flash documents you create.

This option applies all parameters.

12. Click OK.

Flash creates a new document.

Now that you have a new document, you're ready to create something very cool (or not so cool depending on the needs of your client). The next section points you in the right direction.

Creating an object

After you create a document, you can add objects to it. Objects can be plain old ho-hum static objects, or they can be symbols. When you create a *symbol*, you create something that can be used over and over and over again. Just drag the object from the document Library and drop it in the desired place on the Timeline. When you use a symbol instead of creating a new object, you don't increase the file size of the published movie because Flash re-creates an instance of the symbol using the information stored in the document Library. Symbols come in three flavors: Button, Graphic, and Movie Clip. These steps show you how to create a Graphic symbol:

1. In the Tools panel, set the Stroke and Fill colors.

To set Stroke and Fill colors, click the applicable swatch to open the Swatches palette. The colors you select are applied to all objects until you change the colors.

2. Select the Oval tool.

3. Click and drag an oval on the Stage.



When you release the mouse button, the oval appears on the Stage.

Hold down the Shift key to create a circle.

4. Use the Selection tool to select the object.

If you created an object with a stroke, double-click the object to select both the fill and the stroke.

5. If desired, open the Property inspector and change the parameters for the circle.

In the Property inspector, you can change the Width, Height, and x and y coordinates.

6. Press F8.

The Convert to Symbol dialog box appears.

7. Type a name for the symbol and choose the Graphic option.

Flash gives the symbol a default name. However, it's not very descriptive. A better idea is to give each symbol a unique name, which makes it easier for you to select the proper object from the Library. This is a distinct advantage when you're creating complex Flash movies with hundreds of symbols.

8. Click the desired registration point.

This determines the center of rotation for the symbol. The upper-left point works well in most instances.

9. Click OK.

The symbol is added to the Library.

Creating your first animation

After you add a symbol to the Library, you have the bare bones needed for a Flash animation. In this case, we show you how to make an oval go from Point A to Point B. It's not the jiffiest of animations, but it's a start. To create an animation, follow these steps:

1. Create a Graphic symbol, using the Oval tool.

Creating a symbol is easy if you know Flash or if you read the last section. If neither is the case, please rewind and read the previous section.

2. Use the Selection tool to move the symbol to the desired starting point.

To move an object with the Selection tool, click and drag the object to the desired location.

3. Right-click (Windows) or Control-click (Mac) and choose Create Motion Tween from the context menu.

Flash creates 25 frames and positions your cursor in the 25th frame.



If you don't create a symbol as outlined in Step 1, Flash displays a dialog box telling you the object cannot be used in a motion tween animation, but Flash will convert the option if you click OK.

4. Move the oval to the point where you want to end the animation.

Flash creates a motion path and converts the 25th frame to a keyframe (see Figure 1-10).

5. Press Ctrl+Enter (Windows) or ⌘+Enter (Mac).

Flash displays the animation in another window. You're probably thinking that the animation is pretty ho-hum. You're right. So kick it up a notch. After you get tired of looking at the animation, close the window.

6. Click a frame between the beginning and ending keyframes. Using the Selection tool, move the object to another position.

Flash creates another keyframe when you move the object to another position.

7. With the Selection tool, move your cursor toward the first half of the motion path.

When you see a curve appear under the arrow cursor, you can bend the motion path.

8. Click and drag the motion path to curve it.

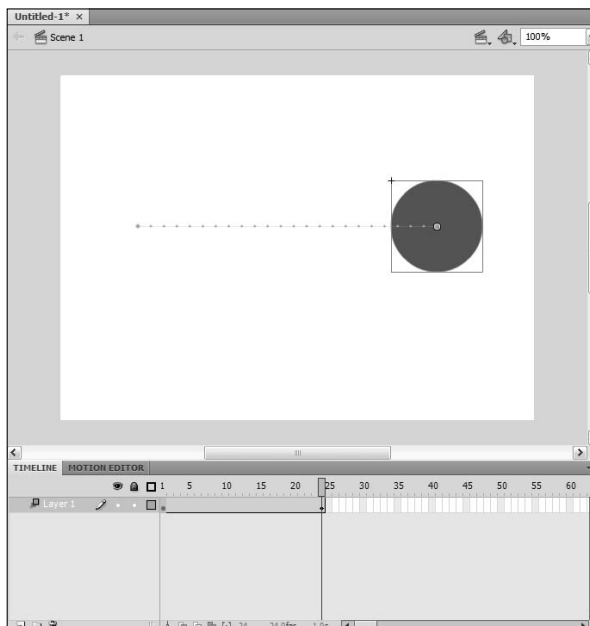


Figure 1-10:
Creating
a motion
tween
animation.

9. Repeat Step 8 for the other half of the motion path.

Your motion path should resemble Figure 1-11.

10. Press Ctrl+Enter (Windows) or ⌘+Enter (Mac).

Now you're cooking. To give you an idea of what you should be seeing, Figure 1-12 was created using the Onion Skin option, which displays multiple frames at a lower opacity. Close the window when you're done patting yourself on the back.

Figure 1-11:
A motion path that's been folded, spindled, and mutilated.

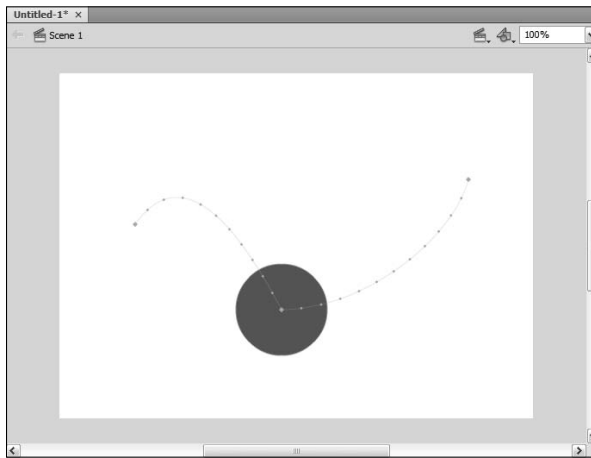
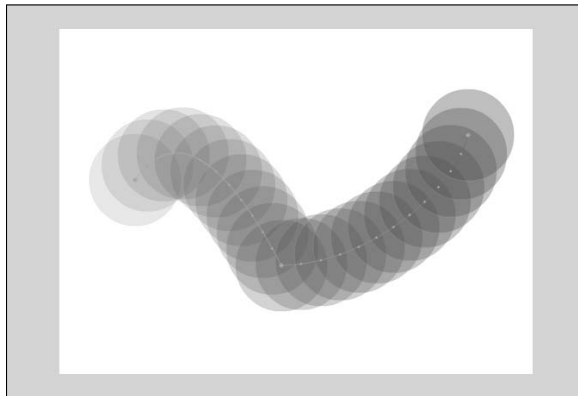


Figure 1-12:
A motion tween animation with multiple keyframes.



Setting up your workspace

If you're like most designers, you have a set way of doing things. The designers of Flash set up the workspace to suit the needs of most Flash users. However, you can customize the Flash workspace to suit your own working preferences. In this section, we show you a few things you can do to set up the workspace just the way you like it.

Setting Flash preferences

Everybody's got preferences. Doug prefers blondes, but that's a whole different kettle of fish. You can modify the default Flash Preferences for the way you work. We don't go through each and every Flash Preference, but we show you a few that we think are important:

1. Choose Edit→Preferences.

Alternatively, you can press Ctrl+U (Windows) or ⌘+U (Mac) to open the Preferences dialog box. By default, the dialog box opens to the General section. (See Figure 1-13.)

2. Modify the default preferences to suit your working style.

Most of the preferences are self explanatory. However, we do recommend that you choose New Document from the On Launch drop-down menu. This option creates a blank document with the default document size, frame rate, background color, and ActionScript 3.0 when you launch Flash, enabling you to cut to the chase instead of dealing with the pesky welcome screen. Another option you might consider changing is the number of Undo levels, which by default is a whopping 100 steps. If you have an older computer or limited memory, lowering the number of Undo levels cuts down on the overhead that Flash places on your system.

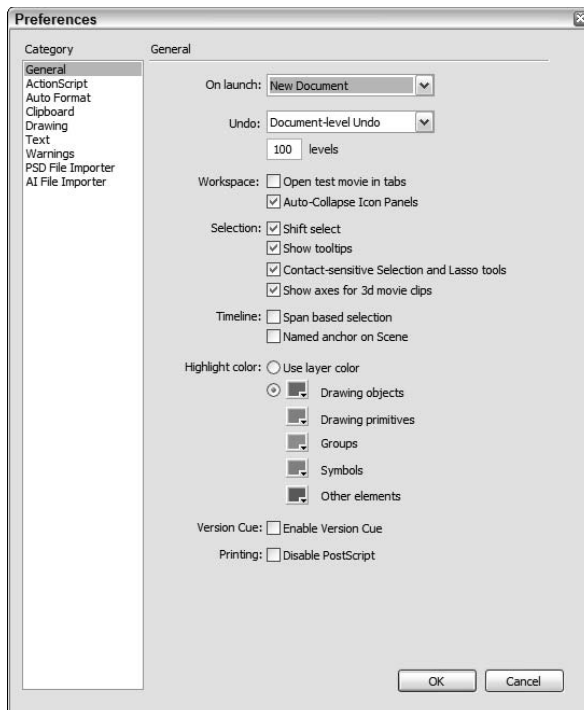


Figure 1-13:
Preferences,
and
General
ones at that.

3. Change the remaining preference categories to suit your working style.

The only other change we recommend you make depends on whether you use ActionScript and whether you have a large desktop size. If both conditions apply, change the font size from 10 to 11 or 12, but keep the default Courier New font, which has code geek written all over it.

4. Click OK to apply the new preferences.

Modifying the workspace

The layout of the Flash workspace suits most designers. You can also choose a different workspace. Essentials is the default workspace, but alternatively, you can choose Classic, Debug, Designer, or Developer. There are also options to reset the current workspace. You can, however, modify the workspace by opening panels and docking them or by placing them in a convenient position. If you really like your changes, you can save them as a custom workspace. This is also a handy option if you work with another designer on the same computer. If the other designer is a gal and you're a guy — you know, the Mars and Venus thing? — you can change the workspace to suit your style of working, save it as a workspace, and then let your femme fatale Flash cohort in crime do the same. Here are a few things you can do to customize the workspace:

- ◆ **Undock a panel.** Click its title and drag it into the workspace.
- ◆ **Dock a floating panel.** Click its title bar and drag it to one of the interface sides. When you see an opaque vertical bar, release the mouse button, and the panel will develop a magnetic attraction to a side of the workspace. When you see an opaque horizontal black bar, release the mouse to dock the panel with the top or bottom of the workspace.
- ◆ **Dock one floating panel with another.** Drag its title bar toward the other panel. When an opaque blue overlay appears over the other panel, release the mouse button; the panels are joined at the title bar. If you see an opaque blue line appear over the panel to which you are docking, the panel you are moving is docked on top of the other panel.
- ◆ **Resize a panel.** Move your cursor toward a corner. When it becomes a line with a dual-headed arrow, click and drag to resize the panel.
- ◆ **Go retro.** (Imagine that you're a Flash veteran who feels bereft because of the new default workspace.) Choose Window⇨Workspace⇨Classic.

When the workspace is just the way you want it, choose Window⇨Workspace⇨New Workspace, which opens the New Workspace dialog box. Enter a name for the workspace (Doug calls his Flash Space Nine) and then click OK. Your new workspace is ready to use and also appears on the Workspace menu.



TIP

If someone plays a nasty trick on you while you're at lunch and rearranges your custom workspace, choose Workspace⇨Reset *the name of your custom workspace*, and everything will be where it was when you saved the workspace.



REMEMBER

If you feel the need to rename or delete a workspace, choose Window⇨Workspaces⇨Manage Workspaces, which opens the Manage Workspaces dialog box. Select the workspace you want to manage, and then click Rename to open the Rename Workspace dialog box, or click Delete to delete the workspace.

Building Flashy Navigation

You can do so many things with Flash, and creating a *très* cool navigation is one of them. You can indeed create cool navigation with HTML, but Flash lets you take it a step further. If you want animated buttons (for instance, a button that makes noise), Flash is your answer. In this section, we show you how to create cool (*Flashy*, if you will) navigation.

Creating the navigation menu document

Before you can create a Flash navigation menu, you create a document to be the base for your menu. Before you create this document, you must know the size of the area in which you plan to place the navigation menu. Typically, this is the width of your site banner and the height you want for your menu. Choose a background color that's harmonious with the Web page in which the menu will be placed. For this example, we create a menu 600 pixels wide and 25 pixels high with a black background, using the default frame rate of 24 fps. (Variety is the spice of life!) Figure 1-14 shows the Classic workspace. Notice that the document is set up for Flash ActionScript 2.0. There are two reasons for this. If you use Flash only for creating animations and navigation, there's no need to learn a new programming language. The second reason is that showing you how to use ActionScript 3.0 would require many more pages than we've been allocated. (If you only knew how huffy editors get when you go over the specified page count. But that's a story for another day.) Now, on with the subject at hand: creating flashy navigation.

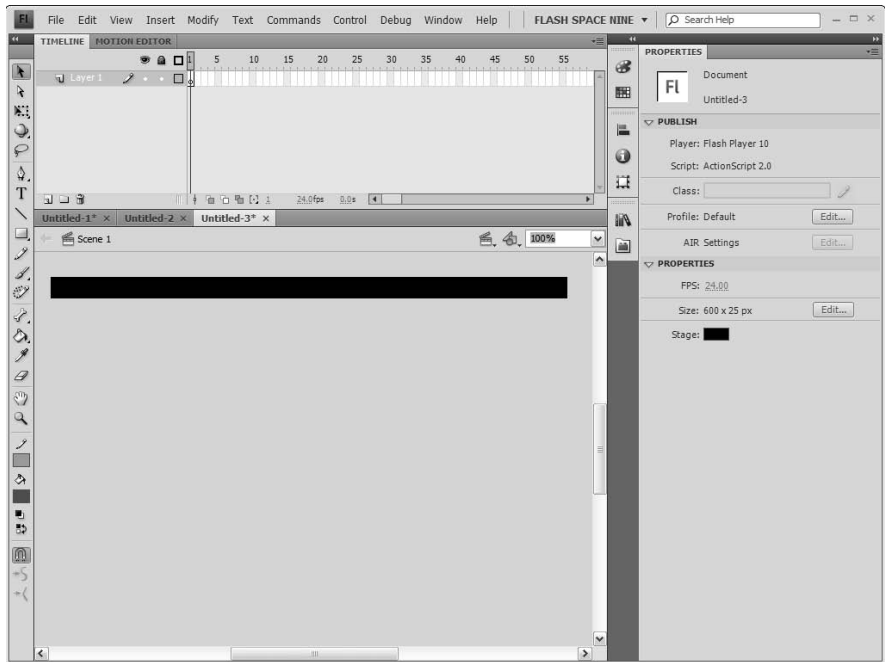


Figure 1-14:
Creating a document for the Flashy navigation menu.

Creating buttons in Flash

Navigation menus need buttons. Flash gives you all the tools to create some really cool buttons. If you haven't created Flash buttons before, consider this your baptism by fire. To create a button for your menu, follow these steps:

- 1. Choose Insert → New Symbol.**

The Create New Symbol dialog box appears.

- 2. Type a name for your symbol and choose the Button behavior.**

You're creating a menu, and the obvious place to start a menu is with the Home button. Therefore, name the button **Home**. (See Figure 1-15.)

- 3. Click OK.**

- 4. Flash enters symbol-editing mode. (See Figure 1-16.)**

Notice that you have one layer as well as four states: Up, Over, Down, and Hit. These states determine what displays when the user's cursor is not over the button (Up), when the user passes his cursor over the button (Over), and when the user clicks the button (Down). The final state, Hit, determines the target area of the button. You add the Hit state only if you have a particularly small button that might be hard to select.

Figure 1-15:
A take-me-home kind
button,
Jenson.

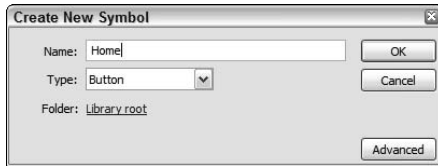
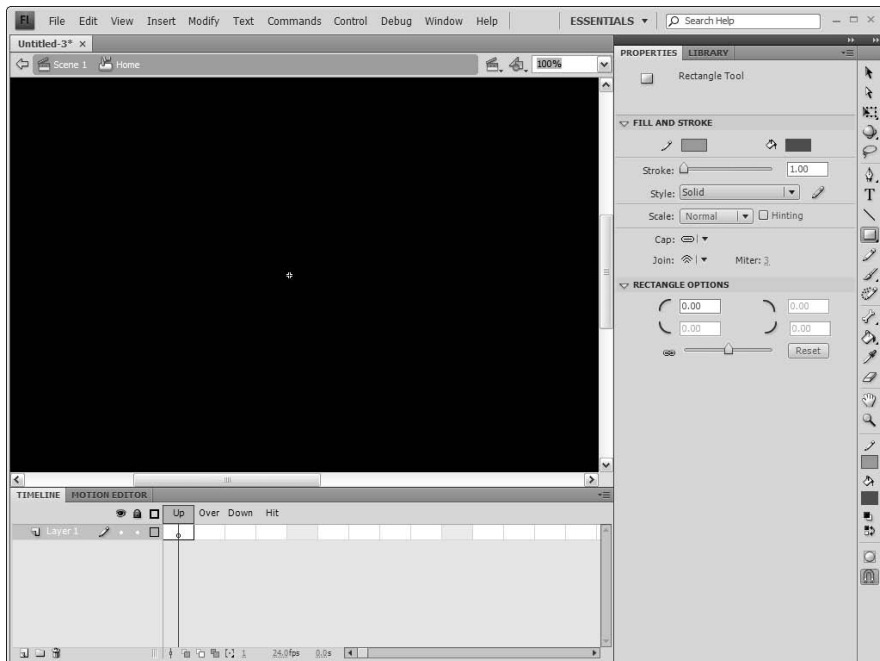


Figure 1-16:
Welcome
to symbol-
editing
mode.



5. If it's not already open, choose Window⇨Properties to open the Property inspector.

6. Create the shape for your button.

When creating buttons, use the Rectangle tool with a corner radius of 50 pixels to get a generic pill shape. Before creating the shape, change the Rectangle Options value in the Property inspector to 50. Choosing View⇨Rulers enables you to create an object that's the approximate size of the button.

7. Type the desired dimensions for your button in the width and height fields.



To figure out what size to make each button, divide the width of your navigation menu by the number of buttons. In this example, we're creating six buttons for a navigation (nav) menu that's 600 pixels wide and 25 pixels high. Therefore, each button is 100 pixels wide and 25 pixels high. (See Figure 1-17.)

8. Select and then align the button to the center of the Stage.

Press Ctrl+K (Windows) or ⌘+K (Mac) to open the Align panel. Click the To Stage icon, and then click the icons that center the button vertically and horizontally.

9. Click the New Layer button.

Flash creates a new layer. Layers help you organize your work. For the button, we create a separate layer for button text.

10. Double-click the layer title.

This opens a text box into which you can type a new name for the second layer. It's not imperative, but crafting an intuitive name now helps you decipher later which layer is used for what when you're creating a complex movie or symbol.

11. Enter a name for the layer.

Text is as good a name as any. While you're at it, rename the layer on which the button resides. *Button shape* is a good name.

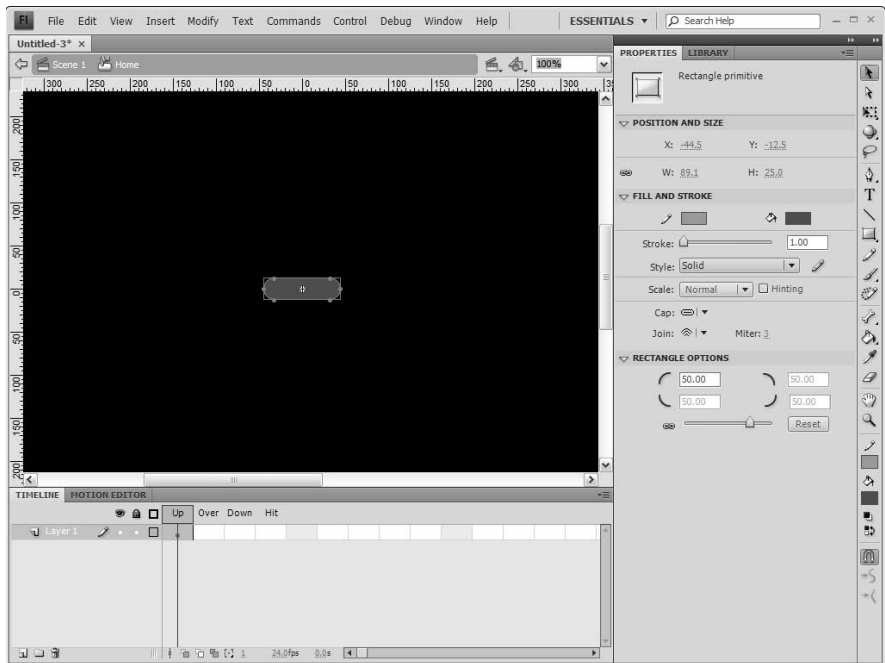


Figure 1-17:
The button
takes shape.

12. Select the Text tool and type the desired name.

In this case, the button is called Home. When a visitor clicks this button, he goes home. A full-blown tutorial on the Text tool is beyond the scope of this book. However, if you look at the Property inspector in Figure 1-18, you see how easy it is to set parameters for the Text tool.

13. Press F5 in the Down frame on the Text layer.

This copies the contents of the Up frame to the Over and Down states.

14. Press F6 to create keyframes for the Over and Down states on the Button Shape layer.

The fact that you created keyframes enables you to change or modify the contents of these frames, or make some other change that will notify users they have indeed found a clickable button. The graphic on the Down state appears when users successfully click the button.

15. Make changes in the Over and Down states for the Button Shape layer.

For the purpose of this demonstration, we changed the color of the button to red in the Over frame and to orange in the Down frame.

16. With the Text layer selected, click the Insert Layer icon.

This houses the sound that plays when the user clicks the button.

Flash creates a new layer.

17. Name the layer Sound.**18. Click the Down frame in the Sound layer and press F6.**

Flash creates a keyframe.

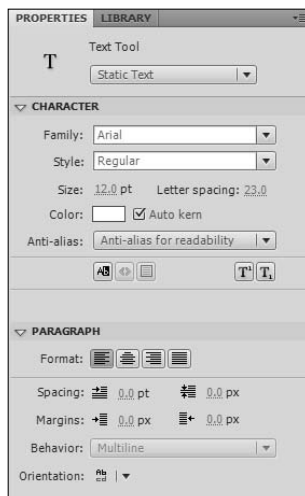


Figure 1-18:
Add text to
the button.

19. Choose File→Import→Import To Library.

Select a short sound file. For a button, choose a sound less than two seconds in duration. If you decide to add sound to a button, choose one that fits the site you're designing. For example, if you're creating a navigation menu for a photographer's Web site, a shutter click is an ideal sound.

20. With the Down frame of the Sound layer selected, drag the sound from the Library to the Stage.

It doesn't matter where you place it. Remember that sounds are heard and not seen. Figure 1-19 shows the completed button. The waveform in the Down frame is Flash's way of notifying you that the sound is in the frame. This is a sound with low amplitude. The waveform is the solid line in the down frame. After all, a shutter click doesn't fluctuate like a Beethoven symphony.

21. Click the Back button (looks like an arrow, and is left of the Scene title above the uppermost layer).

The button is added to the Library and is ready for use.

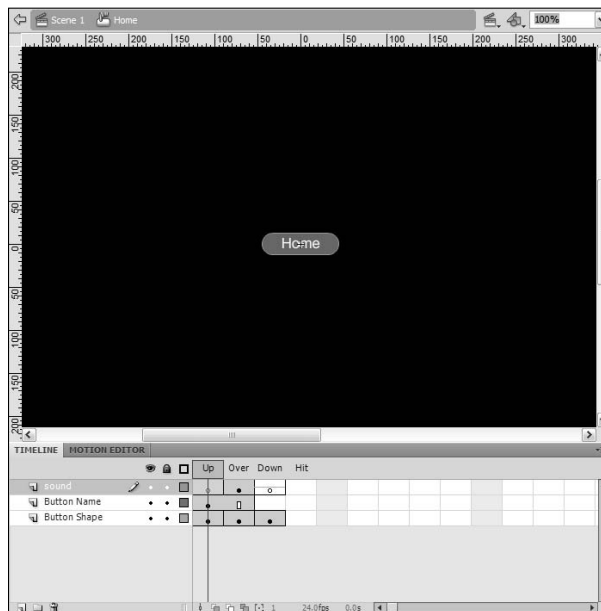


Figure 1-19:
A button
wired for
sound.

Assembling your menu

After you create a button, you can duplicate the button and then change the text to flesh out your menu. If you did the math right, your button is perfectly sized. When you align the buttons end to end, your menu is assembled. To assemble the menu, follow these steps:

- 1. Right-click the Home button in the document Library and then choose Duplicate from the contextual menu.**

The Duplicate Symbol dialog box appears. (See Figure 1-20.)

Figure 1-20:
Duplicating
the button.



- 2. Name the symbol.**

The logical choice would be the page to which the button will link.

- 3. Click OK.**

Flash duplicates the button.

- 4. Create enough duplicates to flesh out your menu.**

For the menu we're creating here, you need four more buttons.

- 5. Double-click one of the duplicated buttons.**

Flash enters symbol-editing mode.

- 6. Change the button text.**

- 7. Repeat for the other buttons in your menu.**

After creating the duplicate, arrange the buttons on the Stage. To do so, follow these steps:

- 1. Select the first button in your menu and drag it on the Stage.**

You don't have to position it exactly. You can use the Align panel to align and distribute the buttons.

- 2. Drag the remaining buttons onto the Stage.**

Position the buttons end to end.

- 3. Select all the buttons.**

You can do this easily with the Selection tool. Simply click and drag around the perimeter of the buttons.

- 4. Press Ctrl+K (Windows) or ⌘+K (Mac).**

The Align panel opens.

- 5. Click the To Stage icon, and then click the Vertical Centers icon.**

Flash aligns the buttons vertically to the Stage.

6. Click the To Stage icon, and then click the Distribute Horizontal Center icon.

The buttons are distributed horizontally to the Stage.

7. Click the Align Vertical icon.

Flash aligns the buttons vertically to the center of the Stage. (See Figure 1-21.)

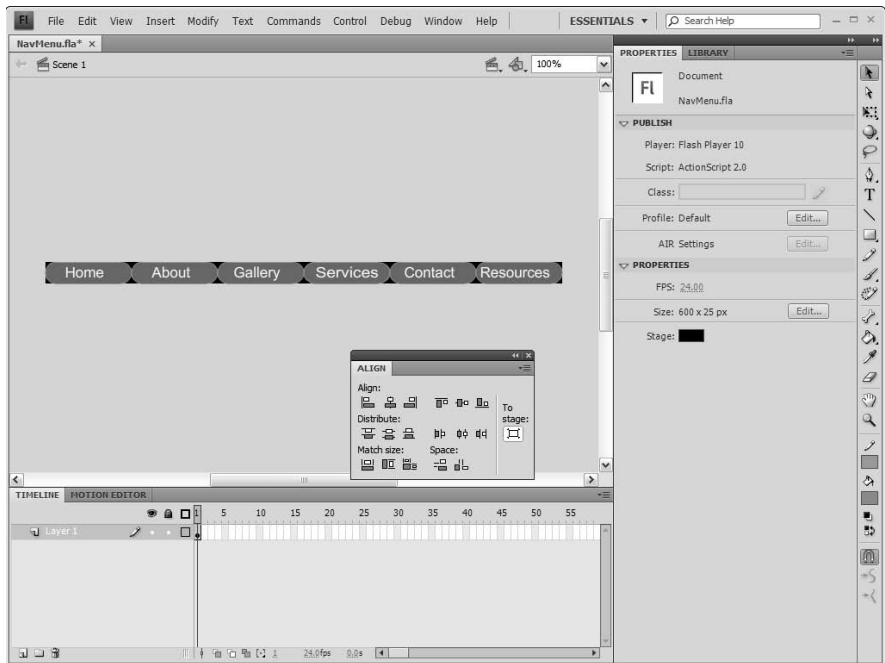


Figure 1-21:
Six little buttons all lined up neatly in a row.

Adding functionality to the buttons

If you test your menu now, the familiar pointer-hand icon appears when you pause your cursor over a button, and the button changes color and makes a noise when clicked. However, the button will do zilch, nada. To make the button truly functional, you need to specify the page to which the button links. To do that, you must venture into the wild and woolly world of ActionScript. As we mention earlier in this chapter, this segment covers ActionScript 2.0. For a full treatise on Flash CS4 and ActionScript 3.0, check out *Flash All-In-One Desk Reference For Dummies* by Doug Sahlin and Bill Sanders (Wiley). To make the buttons truly functional, follow these steps:

1. Select the first button in your menu.

2. Press F9 (or choose Window⇧Actions) and then click the divider to display all the action books.

This opens the Actions panel shown in Figure 1-22. If this is your first introduction to ActionScript, you might be reminded of Pandora's box. The Actions panel has so many actions that they're divided into what the Flash designers refer to as *books*. When you click a book, you have access to all the actions within that book. But sometimes books open other books.

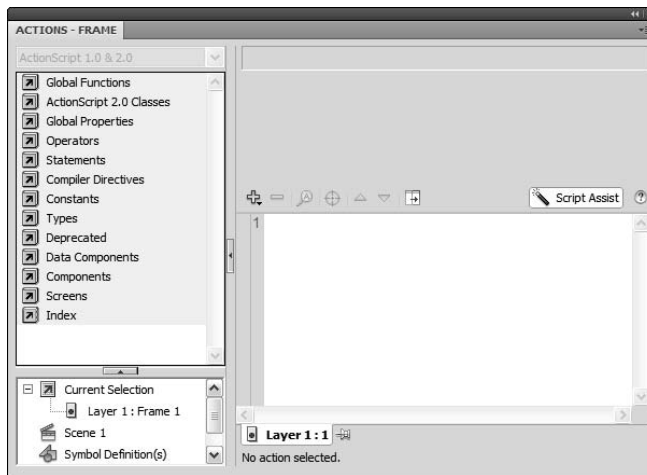


Figure 1-22:
Yikes! Look
at all those
Action
books.

3. Click the Script Assist button.

This button helps novice ActionScript users select the right parameters for each action. If you don't click this button, you've got to write some of the code yourself. Not fun.

4. Click the Global Functions book, click the Browser/Network book, and then double-click the `getURL` action.

Flash adds the action to your script. Just fill in the blanks. Notice that Flash adds a line that begins with `on (release)`. This is an *event*; in other words, this event must occur for the action to execute. The default button event, `on (release)`, means the action executes when the user releases the mouse button, just what the doctor ordered for a navigation menu.

5. In the URL text field (shown in upcoming Figure 1-23), type the URL for the Web page that opens when the button is clicked.

The page you want to open when the button is clicked should be in the same folder as your Flash navigation menu. Therefore, you just type the name of the file. For example, if you're creating a button that links to the home page, the link is `index.htm`.

6. Choose one of the following options from the Window drop-down menu:

- `_self` opens the page in the current frame in the current window.
- `_blank` opens the page in a new window.
- `_parent` opens the page in the parent of the current frame.
- `_top` opens the page in the top-level frame in the current window.

If the button links to a page within the site, choose `_self`. If you're opening a page from another site, choose `_blank`. At this point, your Actions panel should look like the one shown in Figure 1-23.

7. Repeat Steps 4–6 for the remaining buttons in your menu.

You have to select each button in turn, and then add the ActionScript.

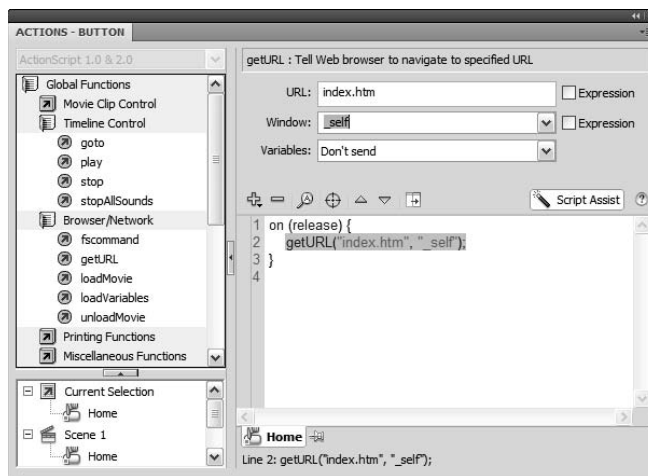
8. Choose File↔Save.

The Save dialog box appears.

9. Enter a name for the file, specify the location where you want to save the file, and then click Save.

Saving Flash documents in the same folder as the other assets for the Web site is typically a good idea. Flash files are saved with the `.fla` extension. You can modify the FLA file at any time by reopening it. All the files you import and objects you create are saved with the file.

Figure 1-23:
The
`getURL`
action in
action.



Publishing your file and adding it to your page

When you finish your menu in Flash, you publish it as a Flash movie. The default publishing options publish the Flash movie embedded in an HTML document. However, you plan to use the menu in an HTML document with other items. Therefore, publish the file as follows:

1. **Choose File⇨Publish Settings.**

The Publish Settings dialog box appears. (See Figure 1-24.) The default settings publish the file as a Flash SWF movie embedded in an HTML document. However, for a navigation menu that you want to add to an HTML document, all you need is the Flash movie.

2. **Deselect the HTML option and then click OK.**

3. **Choose File⇨Publish.**

Flash publishes the file as a SWF movie.

After you publish the navigation menu as an SWF file, you're ready to add it to your Web pages. In Dreamweaver CS4, follow these steps:

1. **Create the document in which you want to add the Flash navigation menu.**

2. **Position your cursor where you want to add the menu.**

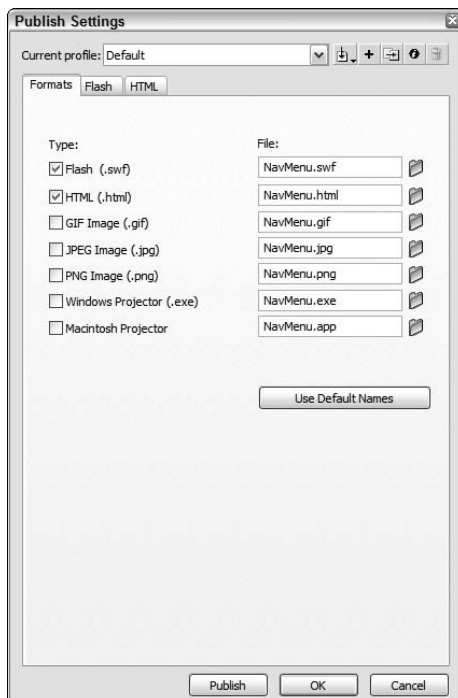


Figure 1-24:
The Publish
Settings
dialog box.

3. Choose **Insert** ⇨ **Media** ⇨ **SWF**.

The Select File dialog box appears.

4. Select the SWF file and then click **OK**.

5. In the Object Tag Accessibility Attributes dialog box that Dreamweaver displays, fill in the desired attributes and then click **OK**.

Dreamweaver inserts the Flash movie in your Web page.

6. Press **F12** to test the file in your default browser.

Dreamweaver displays the Web page in your default browser.

Flash as an Animation Tool

In addition to creating nav menus with Flash, you can use the application to create impressive animations. Recently Doug needed to create a presentation for his photography business. He wanted something cool and interactive, so he used Flash. The first part of the presentation was an animated intro in which images danced across the screen and then stopped to create a collage. Figure 1-25 shows the end of the animation when all the images align as a collage. Notice the multiple layers. One layer has a sound track, whereas the other layers have motion tween animations. The start of each animation is staggered so that the images fly in at different times. Because this was not a complicated animation, the frame rate was changed to 12 fps.

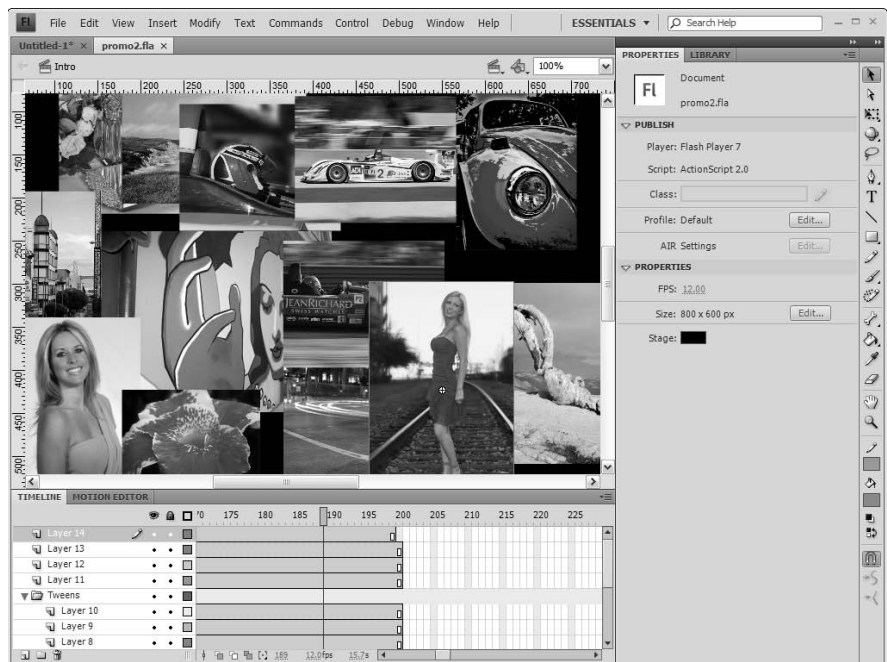


Figure 1-25:
You can
create
impressive
animations
with Flash.

As you can see, quite a bit was involved in creating the animation. The animation dovetails into the actual presentation. Doug used quite a few high-resolution images to create the animation and presentation. The file size of the published movie is 2MB, clearly too large for the Web, but not too large for a computer or CD presentation. If Doug deletes a few images and optimizes the others for Web viewing, the animation would be a reasonable size for use on the Web.

Create a Flash Presentation

When Doug decided to create a presentation for his photography business, he used Flash. He began the presentation with an animation of images (as discussed in the previous section). After the animation plays, a menu and a splash screen load. The menu contains links that load other content into the main Flash movie. A bit of ActionScript wizardry was involved. One of the portfolio slide shows from the presentation is shown in Figure 1-26.



Figure 1-26:
The menu
for an
interactive
Flash
presenta-
tion.

Each button on the menu loads another Flash movie into a target movie clip in the main movie. Working in this manner enables a designer to manage a lot of content using ActionScript. If all the content loads at once, the file size of the movie becomes too large, even for a CD presentation.

Doug wanted to distribute the presentation on business card CDs. He could have published the file as a *Flash Projector file*, which is a self-contained executable file complete with the Flash Player. However, a Projector file displays Macromedia Flash Player 10 in the title bar, and that's where he

wanted to display the business name. The answer was an application known as SWF Studio V3.5. The application converts an SWF file into a customizable, self-executing Projector file. You can add an expiration date to the file after which time it won't play. You can also determine what text displays in the title bar. Auto-run is another feature you can add to your presentation. The end result looks like a custom application created solely for his business. You can download a trial version of SWF Studio V3.5 from www.northcode.com.

Exploring the Many Tricks of Flash

No discussion of Flash would be complete without mentioning the power of ActionScript. *ActionScript* is a scripting language similar to JavaScript. When you sprinkle liberal doses of ActionScript into your Flash movies, you take your designs to the next level. ActionScript comes with a bit of a learning curve. The good thing about ActionScript, though, is you don't have to master each and every action to get the job done. With a basic knowledge of ActionScript, you can rummage through the Actions panel and learn the actions you need to get the job done. Figure 1-27 shows a design that is powered by ActionScript.

The banner across the top of the site moves from side to side. The direction of movement depends on where the user's cursor is. If the cursor is on the right side of the site, the banner moves from left to right. When the user moves his cursor to the other side, the banner reverses direction. The speed at which the banner moves is determined by how far the cursor is from the middle of the design. When the user moves his cursor toward the center of the design, the banner slows to a crawl and then stops.



Figure 1-27:
A Flash
Web design
powered
by Action-
Script.

Chapter 2: Using Sound

In This Chapter

- ✓ **Discovering audio formats for the Web**
- ✓ **Working with Flash, QuickTime, RealMedia, and Windows Media audio**
- ✓ **Audio tips and tricks**

Some people think Web sites are just pretty pictures and text. That's all changed, though. You can add sound to a Web site without breaking a sweat. When you add sound in a Web-friendly format to a Web site, the sound streams into the user's browser. The user must have the proper plug-in to hear the audio content. In this chapter, we show you a thing or two about sound. After reading this chapter, you can create Web sites that can be seen and heard.

Exploring Audio Formats for the Web

Audio for Web sites is saved in formats that enable the sound to stream into the visitor's browser. Streaming sound is similar to an interlaced picture that loads in stages. With streaming sound, the sound begins as soon as enough of it downloads into the user's browser to play without interruption. The most popular audio formats for the Web are

- ◆ **AIFF:** The AIFF (Audio Interchange File Format) sound format was once exclusively for Macintosh users but can now be shared cross-platform. This format uses the `.aif` and `.aiff` extensions and it's played through the Apple QuickTime Player.
- ◆ **Flash:** You can import audio into a Flash document, optimize it, and then publish the document as an SWF file. Add the published file to a Web page, and you've got sound, baby.
- ◆ **QuickTime:** The Apple QuickTime player can play video and audio files. The player recognizes the AIF, AIFF, and MOV sound formats.
- ◆ **MP3:** The MP3 audio format is a derivative of the MPEG video format. The MP3 sound format combines excellent fidelity with relatively small file sizes and is the favorite format for audio podcasts.

Copyright issues

The person who creates the recording (or the recording company) owns the copyright for songs and recordings. If you use a copyrighted song or recording on a Web site you're designing, you are in violation of the copyright laws. You might be able to obtain a license from the copyright owner. Another option is to obtain royalty-free songs or recordings. You can find these by typing **royalty-free recordings** in your

favorite search engine. Some office supply stores sell royalty-free music collections. Or, if you're the really adventurous and artistic type, you can create your own background sounds using a program such as ACID Pro (Windows) or GarageBand (Mac). Both applications enable you to create music by mixing and matching royalty-free music loops.

- ◆ **RealMedia:** RealMedia multimedia content has been around for some time. Each version of the player gets more sophisticated. You can save sound files in the RealMedia .rm format and embed the RealMedia Player in a Web page to play them.
- ◆ **Windows Media Player:** You can embed Windows Media Player in a Web page and play the WMA and MP3 sound formats.

Adding Flash Audio to a Page

Flash audio is the perfect solution when your client wants background music on a Web page. With Flash audio, you can compress music by using the popular MP3 format and then choosing a data rate that assures good-quality sound and a quick-loading file. The next sections cover how to create Flash audio and add it to a Web page.

Creating Flash audio

Creating Flash audio isn't rocket science, but you do need a copy of Flash. You do, that is, unless you're fortunate enough to have a client who sends you the sound file in the SWF format. To create Flash audio, follow these steps:

1. Create a new Flash document.

Accept the default frame rate (24 fps), choose a color that matches the Web page into which you're adding the sound, and specify a size of 1 pixel x 1 pixel.

2. Choose File⇨Import⇨Import to Library.

The Import to Library dialog box appears.

3. Select the sound and then click Open.

Flash imports the sound to the document Library.

4. Select the first keyframe.

You only have one, so it shouldn't be hard to find.

5. Open the Property inspector.

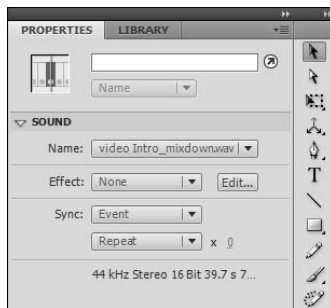
6. Choose the imported sound from the Sound drop-down menu.

The filename of the sound file appears in the Name text field.

7. Accept the default Sync options of Event and Repeat, but change the value of the last option to 0 (zero).

These settings play the sound once. If you have a quiet background sound, you might want to change the second option to Loop and then enter the number of times you want the sound to repeat. After you add a sound to a keyframe, your Property inspector should resemble Figure 2-1.

Figure 2-1:
Add sound to a keyframe — the only keyframe.



A word of caution is in order here. If the song is too loud or doesn't suit all audiences, looping the sound is a surefire way to have part of your intended audience escape by clicking the Back button.

8. Choose Window⇧Library.

The document Library opens.

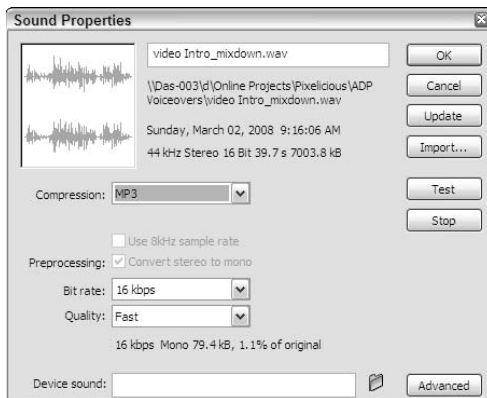
9. Select the sound file, right-click (Windows) or Control-click (Mac), and then choose Properties from the contextual menu.

The Sound Properties dialog box appears.

10. Choose MP3 from the Compression drop-down menu.

The Sound Properties dialog box reconfigures, as shown in Figure 2-2.

Figure 2-2:
Modify
the sound
properties.



11. Choose an option from the Bit Rate drop-down menu.

The default option of 16 Kbps works well for simple background music with one or two instruments. However, if you use more complex music to entertain your viewers (or your client is a diva, opera star, or rock musician), you need to specify a higher bit rate to get a better-quality sound. When you exceed 20 Kbps, preprocessing converts stereo sound to mono. If desired, you can convert the sound to stereo by deselecting the Convert Stereo to Mono check box. A higher bit rate does increase the file size, so you can't go too far unless you have a sound clip less than 30 seconds in duration.



If you import an MP3 sound file, a check box appears in the dialog box with the option to use the imported MP3 quality. The Bit Rate options appear if you deselect this option.

12. Choose an option from the Quality drop-down menu.

The default option Fast renders the file quickly. However, you get better-quality sound if you choose Medium or Best. The latter options take a while longer to render the sound.

13. Click the Test button to preview the sound clip with the current compression settings.

If the sound isn't to your liking, choose a higher bit rate.

14. After modifying the sound to suit your Web page, click OK.

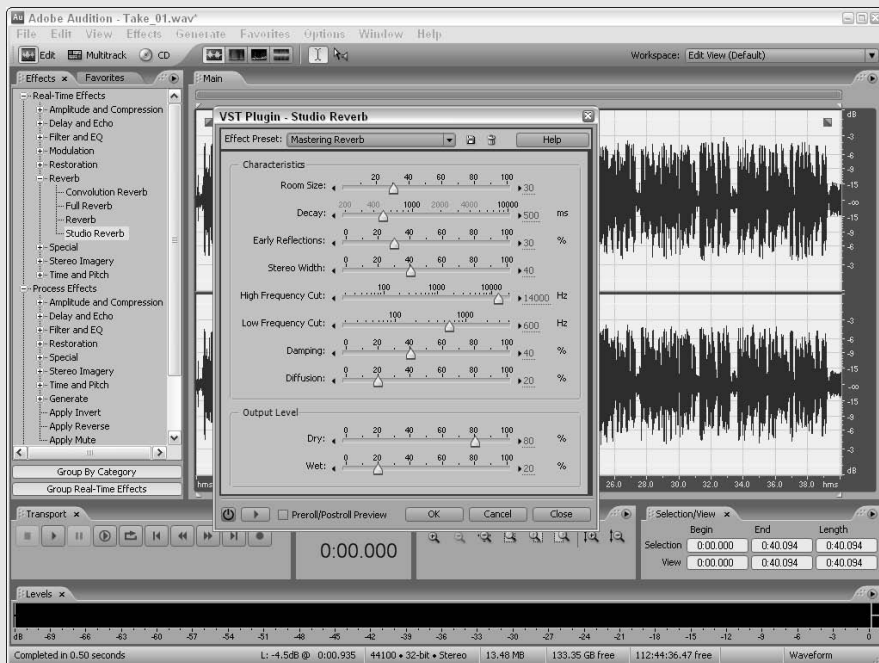
The new settings are applied to the sound.

15. Publish the document as a SWF file.

Editing sound for fun and profit

If you do a lot of work in Flash with soundtracks or add soundtracks to many of your Web pages, consider investing in a sound-editing application, with which you can apply special effects, such as reverb and echo, to a sound to equalize the volume (and much more). You can also use a good sound editing application to create voice-overs for a video track, record

a podcast, and so on. Another definite bonus is the capability to save the sound in just about any format available today. Sony Sound Forge and Adobe Audition are two of the better sound editing applications. The following illustration shows reverb being added to a sound in Adobe Audition 3.0.



Adding sound to a Web page

After you publish your Flash sound, you're ready to add it to your Web page. Remember, this file is infinitesimally small, as in really, really tiny. No one will see the file, so you can tuck it in just about anywhere. Adding the file above the header section of the document is generally a good idea. Honestly, who's going to notice an extra pixel, especially when it's the same color as the background? These steps show you how to add the sound file.

1. In Dreamweaver, open the Web page to which you'll add the sound.
2. Position your cursor where you want to add the sound.

You can put the sound anywhere that's a blank space — say, above the header or below the fold (you know, the stuff you can't see in the browser window and need to scroll to).

3. Choose **Insert**⇨**Media**⇨**SWF**.

The Select File dialog box appears.

4. Select the soundtrack SWF file and then click **OK**.

The Object Tag Accessibility Attributes dialog box appears. This information is used by visually challenged people using screen readers to access the page. For more information on usability and accessibility, see Book II, Chapter 2.

5. Enter the desired information and click **OK**.

Dreamweaver adds the soundtrack to the Web page.

6. Press **F12**.

Dreamweaver prompts you to save the file. After you save the file, it opens in your default browser. (See Figure 2-3.) You should hear your soundtrack.

Figure 2-3:
I'm gonna inject your soul with some sweet rock 'n' roll, and shoot you full of rhythm and blues.



Embedding a QuickTime Player in a Page

Another sound format that's quite popular is AIF, which you can play in the Apple QuickTime Player. You can export AIF audio from a good sound editing application. You can also export a sound file as a MOV file from a sound editing application or from Apple QuickTime Pro. However, these files are rather large and thus not well suited for the Web. When you use the AIF form, you embed the applicable controller in the Web page. This enables visitors to play the associated sound file on demand. To embed a QuickTime audio controller in a Web page using the Dreamweaver HTML editor, follow these steps:

- 1. Position your cursor at the point in the Web page where you want the video to appear.**

- 2. Choose Insert⇨Media⇨ActiveX.**

The Object Tag Accessibility Attributes dialog box appears.

- 3. Enter the desired information and then click OK.**

Dreamweaver inserts an ActiveX object in the page with the default dimensions of 32 x 32. *ActiveX* is a control used to play multimedia objects in software such as the Flash Player.

- 4. Select the ActiveX object in the Document window.**

- 5. Open the Property inspector.**

- 6. Enter the following text into the Class ID text box:**

```
clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B
```

This tells the Web browser that the ActiveX content is QuickTime.

- 7. In the Property inspector, specify the following parameters:**

- Select the Embed check box.*
- Set the width and height in the W and H text boxes. Generally speaking, resizing the object to 320 pixels x 16 pixels is ideal: The default height of the QuickTime controller is 16 pixels, and 320 pixels is wide enough for the visitor to see the progress bar move while the sound plays.*
- Click the Browse folder icon (next to the Src text box). Navigate to and then select the QuickTime AIF sound you want to embed in the page.*
- Enter <http://www.apple.com/qtactivex/qtplugin.cab> in the Base text box.*

8. Click the Parameters button in the Property inspector.

The Parameters dialog box appears.

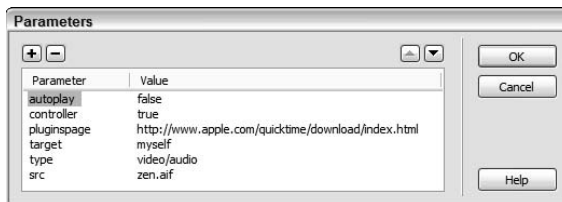
9. Add the parameters for the ActiveX object and then click OK.

Click the plus (+) sign to add a parameter. Each parameter has a value. To run a QuickTime audio file, you need the following parameters and values:

- Parameter 1 = **autoplay**, Value = **false**
- Parameter 2 = **controller**, Value = **true**
- Parameter 3 = **pluginspage**, Value = **http://www.apple.com/quicktime/download/index.html**
- Parameter 4 = **target**, Value = **myself**
- Parameter 5 = **type**, Value = **video/audio**
- Parameter 6 = **src**, Value = **the path and filename of the movie you're embedding in the page**

At this stage, your Parameters dialog box should resemble Figure 2-4. If desired, you can set the autoplay parameter to `true` to have the sound play as soon as it loads.

Figure 2-4:
Set the parameters for a QuickTime sound file.



10. Press F12 to preview the page in your default browser.

Dreamweaver prompts you to save the document. After you save the document, Dreamweaver launches the page in your default browser. The audio begins playing. Figure 2-5 shows a QuickTime controller in a Web page playing the associated AIF file.

Figure 2-5:
The
QuickTime
player
embedded
in a Web
page,
playing an
audio file.



Adding the RealMedia Player to a Page

The RealMedia RM format is a good format for Web sound. The file sizes are relatively small, and you can embed the RealMedia player in a Web page. This gives visitors the capability to play an RM file. To embed the RealMedia Player in a Web page, using the Dreamweaver HTML editor, follow these steps:

1. **Position your cursor at the point in the Web page where you want the video to appear.**

2. **Choose Insert → Media → ActiveX.**

The Object Tag Accessibility Attributes dialog box appears.

3. **Enter the desired information and click OK.**

Dreamweaver inserts an ActiveX object in the page with the default dimensions of 32 x 32.

4. Select the ActiveX object in the Document window.

5. Open the Property inspector.

6. Enter the following text into the Class ID text box:

```
clsid:CFCDAA03-8BE4-11cf-B84B-0020AFBCCFA
```

This tells the Web browser that the ActiveX content is RealMedia.

7. In the Property inspector, specify the following parameters:

- a. Select the *Embed* check box.
- b. Set the width and height in the *W* and *H* text boxes. We recommend resizing the object to 320 pixels x 30 pixels because the default height of the RealMedia audio controller is 30 pixels, and 320 pixels is wide enough for the visitor to see the bar move as the sound plays.
- c. Click the *Browse folder* icon (next to the *Src* text box). Navigate to and then select the RealMedia RM sound you want to embed in the page.
- d. Enter **rvocx** in the *ID* text box.

8. Click the **Parameters** button in the Property inspector.

The Parameters dialog box appears.

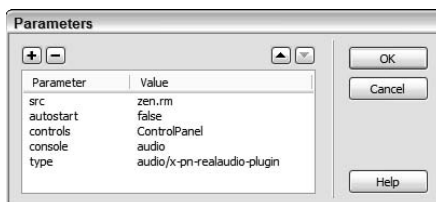
9. Add the parameters for the ActiveX object and then click **OK**.

Click the plus (+) sign to add a parameter. Each parameter has a value. To run a RealMedia audio file, you need the following parameters and values:

- Parameter 1 = **src**, Value = **myaudiofile.rm**
- Parameter 2 = **autostart**, Value = **false**
- Parameter 3 = **controls**, Value = **ControlPanel**
- Parameter 4 = **console**, Value = **audio**
- Parameter 5 = **type**, Value = **audio/x-pn-realaudio-plugin**

At this stage, your Parameters dialog box should resemble Figure 2-6. If you want, you can set the autostart parameter to `true` to have the sound play as soon as it loads.

Figure 2-6:
Set the parameters for a RealMedia sound file.



10. Press F12 to preview the page in your default browser.

Dreamweaver prompts you to save the document. After you save the document, Dreamweaver launches the page in your default browser. The audio begins playing. Figure 2-7 shows a RealMedia controller embedded in a Web page, playing the associated RM file.



Figure 2-7:
A RealMedia audio console embedded in a Web page.

Adding the Windows Media Player to a Web Page

The Windows Media Player is a versatile application. It can play video files as well as audio files. You can embed the Windows Media Player in a Web page and play WMA (Windows Media Audio) and MP3 sound files. To embed the Windows Media Player configured for audio in a Web page using the Dreamweaver HTML editor, follow these steps:

- 1. Position your cursor at the point in the Web page where you want the video to appear.**
- 2. Choose Insert > Media > ActiveX.**

Dreamweaver inserts an ActiveX object in the page with the default dimensions of 32 x 32.

The Object Tag Accessibility Attributes dialog box appears.

3. Enter the desired information and click OK.
4. Select the ActiveX object in the Document window.
5. Open the Property inspector.
6. Enter the following text into the Class ID text box:

```
clsid: 22d6f312-b0f6-11d0-94ab-0080c74c7e95
```

This tells the Web browser that the ActiveX content is Windows Media Video or Audio.

7. In the Property inspector, specify the following parameters:
 - a. Select the *Embed* check box.
 - b. Set the width and height in the *W* and *H* text boxes. Set the height to 42 (the height of the Windows Media Audio controller) and the width to 320.
 - c. Click the *Browse* folder icon (next to the *Src* text box). Navigate to and then select the Windows Media Video WMV movie you want to embed in the page, and then click OK.
 - d. Enter **mediaplayer2** in the ID text box.

The Windows Media Player can also play MP3 audio files. You can link an MP3 file to the Windows Media Player by browsing for the desired MP3 file.



8. Click the **Parameters** button in the Property inspector.

The Parameters dialog box appears.

9. Add the parameters for the ActiveX object and click OK.

Click the plus (+) sign to add a parameter. Each parameter has a value. To run a QuickTime video, you need the following parameters and values:

- Parameter 1 = **src**, Value = **the filename of your movie**
- Parameter 2 = **autostart**, Value = **false**
- Parameter 3 = **showcontrols**, Value = **true**
- Parameter 4 = **showstatusbar**, Value = **false**
- Parameter 5 = **showdisplay**, Value = **false**
- Parameter 6 = **autorewind**, Value = **true**
- Parameter 7 = **type**, Value = **application/x-mpayer-w**
- Parameter 8 = **pluginspage**, Value = **<http://www.microsoft.com/Windows/Downloads/Contents/MediaPlayer>**

At this stage, your Parameters dialog box should resemble Figure 2-8. If you want, you can set the autostart parameter to `true`, which means the audio plays as soon as it loads. The site visitor can use the controller to pause or stop the file.

Figure 2-8: Parameters for an embedded Windows Media Audio controller.



10. Press F12 to preview the page in your default browser.

Dreamweaver prompts you to save the document. After you save the document, Dreamweaver launches the page in your default browser. The audio file begins playing. Figure 2-9 shows the Windows Media Player configured for audio in a Web page, playing the associated sound file.

Figure 2-9: A Windows Media Audio controller embedded in a Web page.



If you attempt to test an embedded Windows Media Player in the Mozilla Firefox browser on your local machine, it won't play. It will, however, play when you upload the HTML page and associated files to your server.

Delivering Your Message

When you add audio to a Web site, you're entertaining site visitors or informing them. When you entertain visitors, it can be in the form of music or perhaps comedy. When you inform people, it's generally in the form of the spoken word with, perhaps, some background music. In either event, your goal is to deliver a professional audio track that's clear and easy to understand. Here are a few things to consider when preparing audio for a Web site:

- ◆ **If your client is recording audio for a Web site, make sure he uses a professional microphone and an application, such as Sony Sound Forge or Adobe Audition, to record the track.** Making a professional recording from a Webcam isn't possible. You should also make sure your client positions the microphone in such a manner that the sounds of his breath are not recorded. The microphone also needs to be far enough away from his mouth to ensure the plosive sounds from the letters such as P and S are not recorded. Your client can also purchase a *pop filter* to minimize plosive sounds. Pop filters take the form of a screen that is between the microphone and the speaker's mouth, or a foam sock that fits snugly over the microphone.
- ◆ **If the sound of your client's voice isn't pleasing or if she speaks with a heavy accent, tactfully suggest she hire a professional to record the narration.**
- ◆ **When compressing soundtracks for the Web, strive for the smallest possible file size, while still maintaining a clean, crisp sound.** The .mp3 audio format is well suited for Web delivery.
- ◆ **Don't compress music soundtracks to 8-bit depth.** The recording will sound scratchy in soft passages.
- ◆ **Make sure the soundtrack isn't overmodulated.** The loudest parts of overmodulated soundtracks are *clipped*, which means that the sound is distorted. If your client is recording the track with a good sound-editing application with *VU meters* (meters that visually display the intensity of the sound), tell him to make sure the levels don't go into the red area.
- ◆ **Don't use recordings from other Web sites unless you can secure a license to use the recording in your Web design.**
- ◆ **Don't use digital versions of copyrighted recordings unless you can secure a license to use the recording in your Web design.**
- ◆ **Don't use an application such as Musicmatch or iTunes to create a digital version of a commercial recording for use on your Web site.** This is a clear violation of the copyright laws.

Chapter 3: Using Web Video

In This Chapter

- ✓ Understanding Web video formats
- ✓ Incorporating streaming video on a Web page
- ✓ Encoding video
- ✓ Working with Flash, QuickTime, and Windows Media video

If sound is cool, then motion and sound combined must be cooler, right? Well, of course! That's why you see so much video on the Web these days. Web video can range from anything simple (say, a talking head introducing visitors to a Web site) to full-motion video of sporting events (such as auto races). Video for the Web streams into the viewer's browser. Before the advent of streaming video, Web pages with video weren't feasible. After all, who in their right mind would wait for a 15 or 20MB file to download? Not too many people. With streaming video, though, large file sizes are possible because the video starts playing as soon as enough data downloads for the movie to begin playing. This chapter gives you the skinny on what you need to know to add a lean, mean, streaming video machine to your Web designs.

Exploring Web Video Formats

There aren't quite as many video files as Carter has little liver pills, but close. During the infancy of video for the Web, video was segregated to platforms. Macintosh users had Apple QuickTime, and Windows users had Windows Media Video. After a while, both platforms could play both formats. Then, along came RealMedia. And finally, the designers of Flash invented their own video *codec* (a file format that compresses the video when it's rendered and then decompresses the video when it's played). Here are the popular video formats for the Web:

- ◆ **Apple QuickTime:** One of the granddaddies of video for the Web, QuickTime has been around since 1992. The Apple QuickTime player is required to view QuickTime video. The most popular QuickTime video format for the Web is MOV, which (you guessed it) is derived from *movie*.

- ◆ **RealMedia:** RealMedia is the brainchild of RealNetworks. RealMedia incorporates RealVideo and RealAudio. RealMedia streaming files for the Web can contain RealAudio and RealVideo streams, and several other formats, including SMIL (Synchronized Multimedia Integration Language). The RealPlayer is required to view RealMedia content. Really! Yes, there are a whole lot of *reals* in the previous sentences, but not a single *reel*, as in *movie*.
- ◆ **Windows Media Video:** Microsoft's answer to the Web video wars is Windows Media Video, the WMV (Windows Media Video) format. This format has improved steadily over the years and can be viewed cross-platform. The Windows Media Player acts as a plug-in when Web pages with WMV video are displayed.
- ◆ **Flash Video:** Those wild and crazy guys (and probably gals) at Macromedia jumped on the video bandwagon with their showpiece application, Flash. Before Macromedia invented its own video codec, Apple QuickTime MOV video files could be incorporated into a Flash Web design — and they still can be. However, Macromedia wanted its own format and must've taken lessons from that Portuguese chef when they kicked it up a notch and invented the Flash Video codec known as FLV (Flash Video). Now that Adobe owns Flash, Flash video is better than ever. The Flash Video On2VP6 codec produces breathtaking, crystal-clear video. Flash video can be played by Flash Player 6, 7, 8, 9, and 10. When you upload a video to YouTube, it's converted to Flash video, which makes it real easy for everyone and his little brother to watch what's on YouTube because everybody and his little brother has the Flash Player installed on their computers.
- ◆ **H264/MPEG-4:** If you've downloaded video podcasts from iTunes or watched them online, you've witnessed this format firsthand. The format was designed to deliver high-quality video at lower data rates, which of course means that the files stream into the viewer's browser more quickly and are a smaller file size.

Working with Digital Video

Video is very much a part of the Web. Therefore, to be successful as a Web designer, you need to know about Web video and how to incorporate it on your page. As we mention earlier in this chapter — which was toiled over late in the evening and with great precision — many video formats are available for the Web. Like images, video files start out as something completely

different from what you end up putting on a Web page. If you're lucky, you have a client who knows all about video and gives you perfectly encoded video all ready to plop onto your Web pages. (Right! And we have a bridge in Brooklyn, New York, that we'll let you have for a song.)

The reality of the matter is that you're likely to get a video cassette from your client's camcorder — or, worse, a CD with encoded video that's been severely compressed and instructions for which bits he wants for the Web site. When that happens, you have to encode the video yourself or hire a professional videographer to do it for you. And if you opt for the latter scenario, the video has already been compressed, which means the end result won't be high quality. You know: Garbage in, garbage out.

Capturing video

If your client hands you a cassette tape, you have to get the contents of it into your computer (*capture* it) before you can encode it. To capture video, you need either a video capture card or an application such as Adobe Premiere or Sony Vegas. Well, for that matter, Windows Movie Maker and Apple iMovie can also capture video. Most applications capture video in a proprietary, digital video (DV) format. We say *proprietary* because even though the file format name is the same, different applications seem to use slightly different algorithms to capture video as DV files. If you've dabbled with digital video, you're familiar with these terms. However, if you need a full-course serving that shows you how to capture and process digital video, trot on down to your local bookstore and pick up a copy of *Digital Video For Dummies*, 4th Edition, by Keith Underdahl (Wiley).

The application itself isn't all you need, though. You also need a connection between the digital video camera or digital video deck and the computer. Most digital cameras and digital video decks use FireWire (IEEE 1394) or USB connections. On the computer side, either a FireWire card or a video capture card is required. Capturing video is a fairly boring process because you just can't speed it up. If your client gives you a cassette with 20 minutes worth of video, it takes 20 minutes to capture it. But that does give you an excuse to catch up on something more important, like creating a shag rug with recycled hairballs. Figure 3-1 shows video being captured by Sony Video Capture 6.0, which is part of the Sony Vegas 7.0 video-editing application. As of this writing, Vegas 8.0 is the most current version of the software.

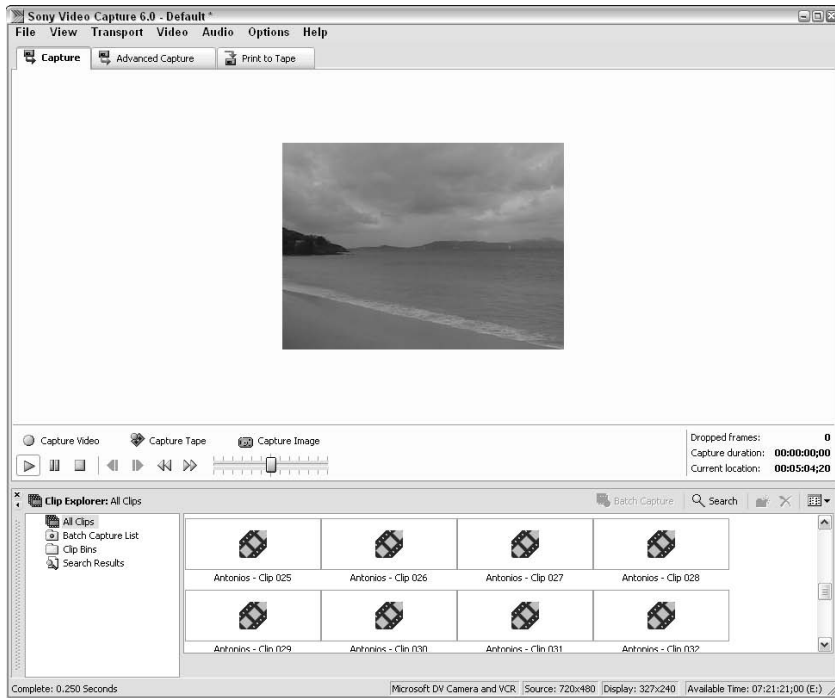


Figure 3-1:
Video
doesn't
stand still,
but it's easy
to capture.

Encoding video

When you *encode video*, you specify a video codec. A *video codec* compresses the video to the desired data rate, which — when combined with the length of the video — also determines the file size. A higher data rate means better-quality video at the expense of a large file size. The other end of the spectrum decreases video quality while yielding a smaller file. When the encoded video is played, the codec decompresses the video. When you encode video, you specify the frame size and data rate. The data rate is specified in Kbps (kilobytes per second). Where Web video is concerned, data rate is directly related to bandwidth. If you try to cram a video with a data rate of 512 Kbps through a modem with a connection speed of 56 Kbps . . . well, you can see it just won't happen.

So how do you know what data rate to use? And why do you need to worry about data rate when you already have your hands full with CSS and other HTML delights? Well, you really don't *need* to know a lot — that is, if you have a good application to encode your video. A good video encoding application, also known as a *compression application*, shows you which options to use for a specific destination. If you're preparing video for a client whose intended audience will be viewing the video with a broadband connection, you can choose the proper preset to suit the need.

Editing digital video

We know what you're thinking: This is a book about Web design. However, it never hurts to master multiple skills, especially where digital video is concerned. Many video editing applications are sheer torture to work with. We won't name names, but some video timelines show only the beginning and end of a clip, which makes it pretty difficult to know what to leave in and what to leave out. Other video editing applications show intermediate frames, and if you zoom in close enough, every frame on the timeline. This makes it easy to slice and dice a video to perfection.

If you've worked with Flash, you're familiar with timelines. Video editing—application timelines are no different. You navigate to a specific spot in the timeline to perform a task such as splitting a video, inserting a video clip, adding a video transition, and so on. If you use a video editing application with an intuitive interface, creating a video production can be extremely rewarding.

The following illustration shows a podcast video being edited in Sony Vegas 7.0. If you're interested in getting your feet wet in video editing, try editing some video in Windows Movie Maker II or Apple iMovie.



Sorenson Squeeze (www.sorensonmedia.com) is an excellent compression suite that enables the user to compress a file in the following Web video formats: Flash Movie (SWF), Flash Video (FLV), QuickTime (MOV), Windows Media Player (WMV), and RealMedia (RM). In addition, you can compress

audio in the popular MP3 format and encode video for DVD. The current version of Sorenson Squeeze is 5.0. Although the application lists for \$499, you'll recoup that in a heartbeat if you work with a lot of video. You can process a video into single or multiple formats. If your client needs three different versions of a video in the MOV format for dialup connections, DSL connections, and cable connections, you simply open the source video, select the presets, and click the Squeeze It button. Within a few minutes (or hours, if you're processing a long video into several different formats), you have perfectly compressed video ready to embed in your Web page. Another added bonus is the ability to batch process a folder of videos. Figure 3-2 shows a video being encoded using the Sorenson Squeeze 4.5 Compression Suite.



Figure 3-2:
Isn't compression like a diet or something?

Creating and Encoding Flash Video

If you read the early parts of this chapter and start reading this section with tears in your eyes, swearing you'll never do video, take a deep breath. Flash has its own video encoder. Yup, it's part of Flash CS4 Professional. The upcoming sections show you how to encode digital video into the FLV format within Flash. Now that you've caught your breath, read on.

If you have a client who wants to add video to his Web site, you can do so easily with Flash. Full motion video in Flash is a beautiful thing. You tailor the video for your client and the type of equipment used by the client's audience. You can also add a controller to the video, which enables visitors to control playback of the video.

When you need to quickly add a video to a Flash movie, you can do so by encoding the video within Flash. When you encode a video in Flash, you can launch the Adobe Media Encoder, which enables you to encode the file. You then come back to Flash to add the encoded video to your project. These handy steps show you how to encode a video in Flash:

1. Create a new document the same size as the video you’re incorporating on your Web page.

Make sure you match the frame rate of the main movie.

2. Choose File→Import→Import Video.

The Import Video dialog box (see Figure 3-3) appears.

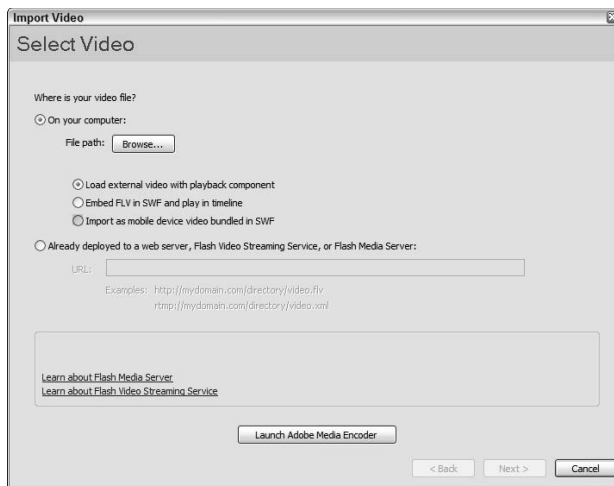


Figure 3-3:
Start by
importing a
video into
Flash.

3. Click the Browse button.

The Open dialog box appears.

4. Select the video you want to import and then click Open.

The path to the video and filename appears in the File Path field. If Adobe Flash Player doesn’t support the video format, a message to this effect appears telling you it’s necessary to use Adobe Media Encoder to convert the file to a supported format.

5. Accept the default Load External Video with Playback Component option unless you have a client who owns several Ferraris and is rich enough to afford Flash Video Streaming Service or Flash Media Server.

6. Click Launch Adobe Media Encoder.

The Adobe Media Encoder dialog box appears (see Figure 3-4).

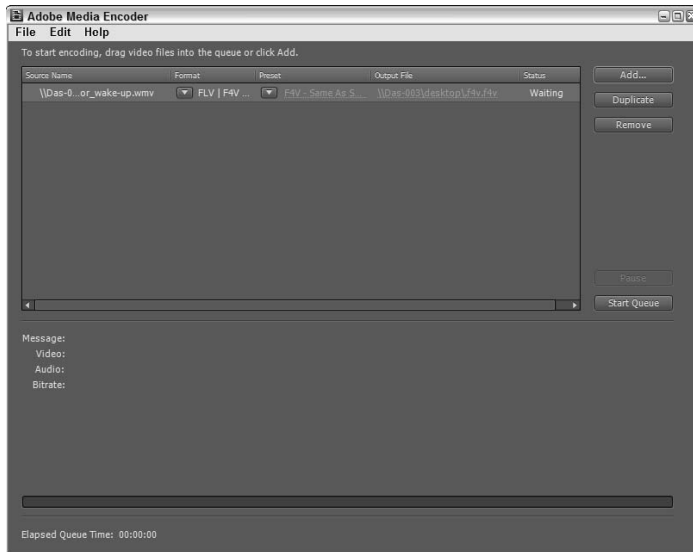


Figure 3-4:
Encode
video into
Flash format
with Adobe
Media
Encoder.

7. Accept the default setting or choose a different setting from the Preset drop-down menu.

This menu has lots of choices. In most instances, the default preset is right on the mark. However, if you're familiar with video encoding, feel free to choose the setting you deem best for your intended audience.

If you're familiar with encoding digital video, choose Custom and tweak the settings to suit the Web page in which the video will appear. When you modify the settings, Custom, rather than one of the defaults, appears in the encoding window.

8. Click the Start Queue button.

Adobe Media Encoder does its thing. If you have a long video, this would be a good time to grab your favorite snack. After the video is encoded, you can close Adobe Media Encoder.

9. In Flash, click Browse.

Navigate to the video file you just encoded and open it.

10. Click Next.

The Skinning page of the Import Video Wizard appears (see Figure 3-5).

11. Select a skin from the Skin drop-down menu.

After you select a *skin* (or interface for the video), a preview appears in the preview pane. Unless you're going to use the video as part of a Flash Web site, choose None. All you need is the encoded Flash video. You can add a skin when you add the video to your Web page in Dreamweaver.





The minimum width of the skin is listed after you choose it. If the minimum width is larger than the width of your video, choose a skin that doesn't have as many controls.

12. **Accept the default skin color, or click the Swatch and choose a different color from the Swatches panel.**
13. **Click Next.**

The Finish Video Import page of the Import Video Wizard appears (see Figure 3-6).

Figure 3-5:
Will your video have a thick or thin skin?

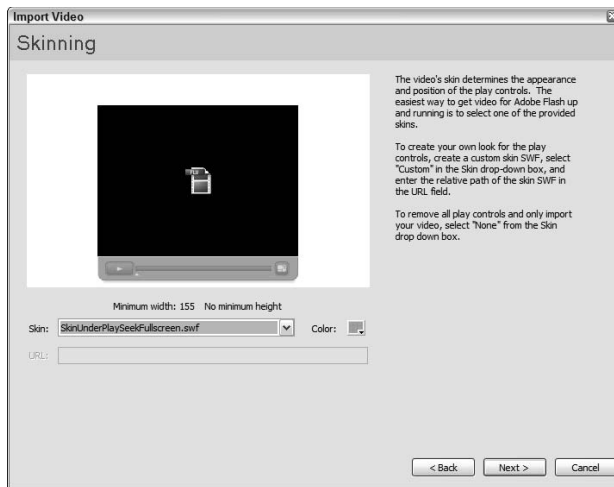
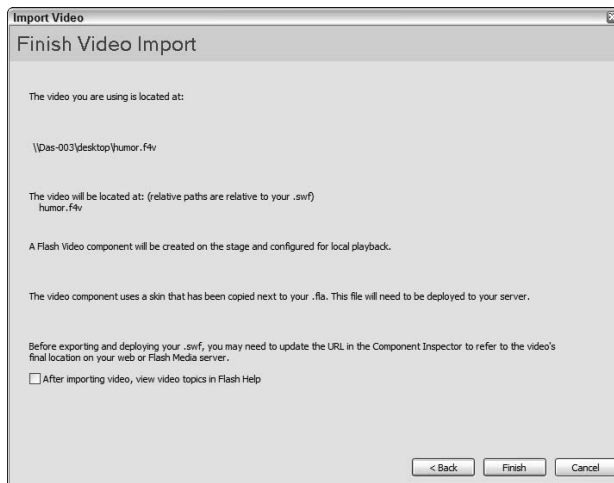


Figure 3-6:
That's a wrap.



14. Review the information and click Finish if you're satisfied.

Alternatively, you can click Back and navigate to the page on which you need to make changes.

15. Choose Control↔Test Movie to preview your video.

16. Save the document.

This step is optional, as all you really need is the FLV file.

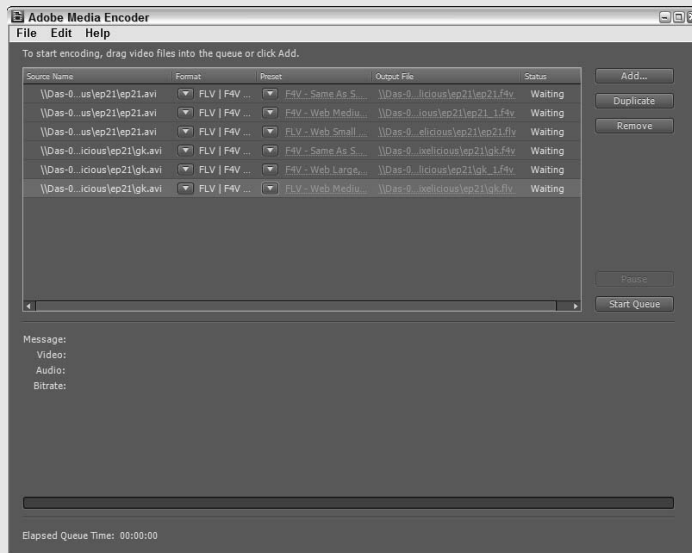
Encoding video with Adobe Media Encoder

If you're encoding multiple videos, use Adobe Media Encoder, which ships with Flash CS4 Professional. This gem gives you the capability of encoding multiple video files. The settings options are identical to those you find when you import video into Flash and use the progressive download method of deployment.

After you apply settings to a file, you can duplicate the file and modify the settings. This option is useful if you need to deploy different versions of the video for visitors who access the Web with dialup, DSL, or cable modems.

You can add as many videos as you want to the queue. The application encodes one video at a time. The following illustration shows Adobe Media Encoder with two video files that will be encoded with three settings. In this example, the application will encode six videos.

Here's a good tip: When encoding multiple videos, set up the application prior to finishing work for the day. Then before you leave, just click the Start Queue button. The encoded files will be ready to use in a Web page when you start work the next day.





When you encode video via Adobe Media Encoder from within Flash, the video encoder creates an FLV (Flash Video) file with the same name as the source file with the `.flv` extension. You need to upload this file to your Web server along with your Web page.

Adding Flash Video to a Web Page

Adding Flash video to a Web page in Dreamweaver is easy. As an added bonus in Dreamweaver, you can add controls to the video to enable Web site visitors to start and stop the video as they please. This is much better than having the video loop endlessly while the visitors are trying to pay attention to the other content on the page. To add Flash video to a Web page, follow these steps:

1. **Position your cursor at the point in the Web page where you want to insert the video.**
2. **Choose Insert → Media → FLV.**

The Insert FLV dialog box appears. (See Figure 3-7.)

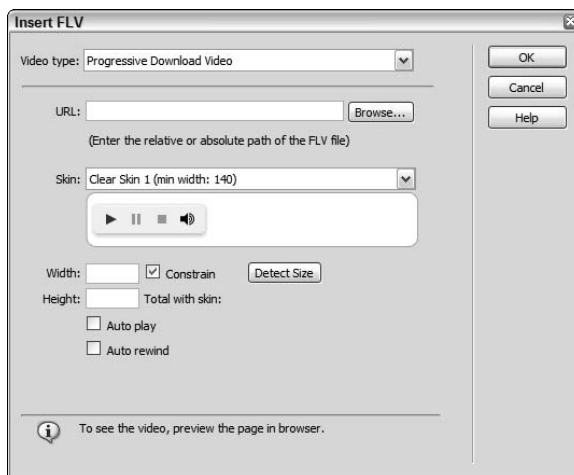


Figure 3-7:
Inserting
Flash
video in a
Web page.

3. **Accept the default Progressive Download Video option for Video Type.**
The other option is Streaming Video, which requires a special server.
4. **Click the Browse button next to the URL field.**

Navigate to the FLV file you want to add to the Web page. If you've been neat and tidy, the file should be in the Web site folder. If you haven't been neat and tidy, Dreamweaver lets you know by displaying a dialog box offering you the option of copying the file to the Web site folder. Take it.

5. Choose a skin from the Skin drop-down menu.

There are several options that give Web site visitors different options, such as playing the video or playing and pausing the video as well as adjusting the sound volume. Each skin lists the minimum width required to display all of the controls. Your video width must be equal to or larger than the width of the skin you choose.

6. Enter values in the Width and Height fields.

These are the dimensions of the video you're adding to the page. Alternatively, you can click the Detect Size button to have Dreamweaver fill in the blanks.

7. Choose additional options as needed.

You can have the video Auto play when fully downloaded and Auto rewind when finished. If you choose both options, the video will loop endlessly until the user clicks the Stop button on the skin.

8. Click OK.

Dreamweaver adds the video to the page.

9. Press F12 to preview the page in your default browser.

Dreamweaver prompts you to save the document. After you save the document, Dreamweaver launches the page in your default browser. The video controller becomes partially transparent after the video begins playing. (See Figure 3-8.)



Figure 3-8:
Flash
video in a
Web page.

Adding QuickTime Player to a Page

QuickTime video is another popular choice for Web video. Unfortunately, Dreamweaver has no menu choices to add QuickTime video to a page. You can, however, add ActiveX content to a page, which is what QuickTime video is. You specify the parameters to get the file to play correctly within the page. To embed a QuickTime player that plays an associated MOV file in a Web page, open a Web page in Dreamweaver and follow these instructions:

- 1. Position your cursor at the point in the Web page where you want the video to appear.**
- 2. Choose Insert ⇨ Media ⇨ ActiveX.**

The Object Tag Accessibility Attributes dialog box appears.

- 3. Enter the desired information and click OK.**

Dreamweaver inserts an ActiveX object in the page with the default dimensions of 32 x 32 pixels.

- 4. Select the ActiveX object in the Document window.**
- 5. Open the Property inspector.**
- 6. Enter the following text into the Class ID text box:**

```
clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B
```

This tells the Web browser that the ActiveX content is QuickTime.

- 7. In the Property inspector, specify the following parameters:**
 - a. Select the Embed check box.*
 - b. Set the width and height in the W and H text boxes. Add 16 to the height if you're including a controller (play and pause buttons).*
 - c. Click the Browse folder icon (next to the Src text box). Then navigate to and select the QuickTime MOV movie you want to embed in the page.*
 - d. Enter <http://www.apple.com/qtactivex/qtplugin.cab> in the Base text box.*

- 8. Click the Parameters button in the Property inspector.**

The Parameters dialog box appears.

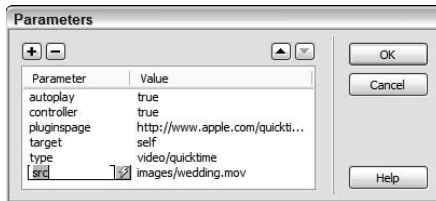
- 9. Add the parameters for the ActiveX object and click OK.**

Click the plus (+) sign to add a parameter. After you click the plus sign, a text box appears in which you type the parameter. Each parameter has a value that you type in the next text field. To run a QuickTime video, you need the following parameters and values.

- Parameter 1 = **autoplay**, Value = **true**
- Parameter 2 = **controller**, Value = **true**
- Parameter 3 = **pluginspage**, Value= **http://www.apple.com/quicktime/download/index.html**
- Parameter 4 = **target**, Value= **self**
- Parameter 5 = **type**, Value = **video/quicktime**
- Parameter 6 = **src**, Value = ***path and filename of the movie you're embedding in the page***

At this stage, your Parameters dialog box should resemble Figure 3-9. If desired, you can set the autoplay parameter to false, in which case the video won't play until the visitor clicks the Play button on the controller. You can also set the value for controller to false, in which case a controller isn't displayed. Logic dictates it's not advisable to set both of these parameters to false because your visitor will see only the first frame of the movie and have no way to play it.

Figure 3-9:
Setting the parameters for a QuickTime video.



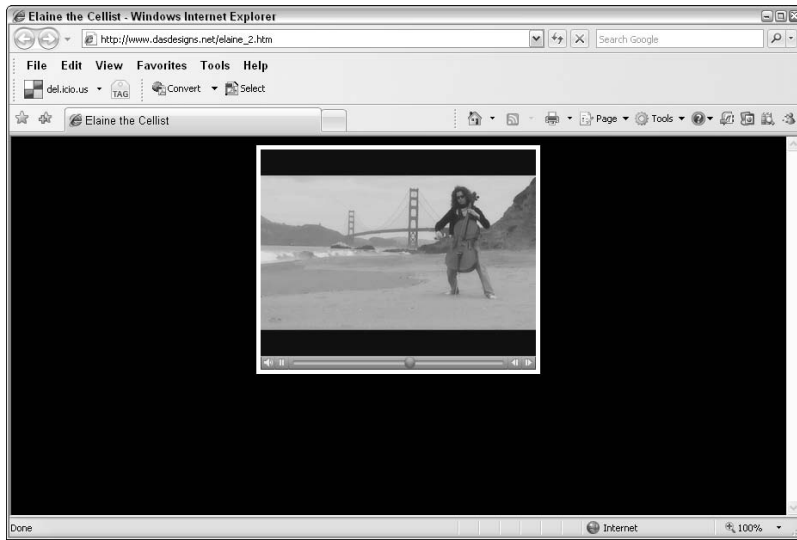
10. Press F12 to preview the page in your default browser.

Dreamweaver prompts you to save the document. After you save the document, Dreamweaver launches the page in your default browser. The video begins playing. Figure 3-10 shows a QuickTime Player in a Web page playing the associated video file.



If you embed a lot of video in Web pages, switch to Code view and select the code that embeds the ActiveX object into the Web page. Open the Snippets palette and then click the New Snippets button at the bottom of the palette. You can now insert the *Snippet* (the code you captured and saved) into other pages. After inserting the Snippet, change the parameters that specify the source video in the Property inspector. For more information on Snippets, check out Book VIII, Chapter 2.

Figure 3-10:
A QuickTime
video
embedded
in a Web
page.



Adding Windows Media Player to a Page

Choices are a good thing. When it comes to Web video, you have quite a few. Yet another option you have to satisfy your client's need to have full-motion video on her Web site is Windows Media Video. You can embed the Windows Media Player in a Web page that gives visitors the controls they need to play the associated video file. To embed the Windows Media Player in a Web page, open the Web page in Dreamweaver and follow these steps:

1. **Position your cursor at the point in the Web page where you want the video to appear.**
2. **Choose Insert → Media → ActiveX.**

The Object Tag Accessibility Attributes dialog box appears.

3. **Enter the desired information and then click OK.**

Dreamweaver inserts an ActiveX object in the page with the default dimensions of 32 x 32 pixels.

4. **Select the ActiveX object in the Document window.**
5. **Open the Property inspector.**
6. **Enter the following text into the Class ID text box:**

```
clsid: 22d6f312-b0f6-11d0-94ab-0080c74c7e95
```

This tells the Web browser that the ActiveX content is in Windows Media Video.

7. In the Property inspector, specify the following parameters:

- a. *Select the Embed check box.*
- b. *Set the width and height in the W and H text boxes. Add 50 to the height if you're including a controller.*
- c. *Click the Browse folder icon (next to the Src text box). Navigate to and then select the Windows Media Video WMV movie you want to embed in the page.*
- d. *Enter **mediaplayer1** in the ID text box.*

8. Click the Parameters button in the Property inspector.

The Parameters dialog box appears.

9. Add the parameters for the ActiveX object and then click OK.

Click the plus (+) sign to add a parameter. Each parameter has a value. To run a QuickTime video, you need the following parameters and values:

- Parameter 1 = **src**, Value = *filename of your movie*
- Parameter 2 = **autostart**, Value = **false**
- Parameter 3 = **showcontrols**, Value = **true**
- Parameter 4 = **showstatusbar**, Value = **false**
- Parameter 5 = **showdisplay**, Value = **false**
- Parameter 6 = **autorewind**, Value = **true**
- Parameter 7 = **type**, Value = **application/x-mpplayer-w**
- Parameter 8 = Flash CS4 Professional, Value = **http://www.microsoft.com/windows/windowsmedia/player/11/default.aspx**

At this stage, the Parameters dialog box should resemble Figure 3-11. If desired, you can set the autostart parameter to true, in which case the video plays as soon as it loads. You can also set the value for showcontrols to false, in which case a controller isn't displayed. Logic dictates it's not advisable to set both of these parameters to false because your visitor will see only the first frame of the movie and have no way to play it.

10. Press F12 to preview the page in your default browser.

Dreamweaver prompts you to save the document. After you save the document, Dreamweaver launches the page in your default browser. The video begins playing. Figure 3-12 shows a Windows Media Video in a Web page complete with controller.

Figure 3-11: Set parameters for an embedded Windows Media Video.

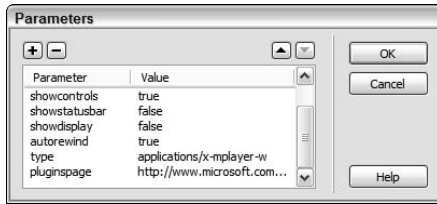


Figure 3-12: A Windows Media Video embedded in a Web page.



If you attempt to test an embedded Windows Media Player and associated video file in the Mozilla Firefox browser on your local machine, it won't play. It will, however, play when you upload the HTML page and associated files to your server.

Avoiding Digital Video Pitfalls

Nobody likes to wind up with egg on his face. For most, the additional cholesterol is unwelcome — not to mention the mess. But we digress. Egg on your face where Web video is concerned is a file that doesn't play, or doesn't play properly. Here are some handy tips and tricks to help you keep your client's Web video squeaky clean:

- ◆ **Don't copy video from other Web sites and use them on yours.** This violates copyright laws.
- ◆ **Don't use a video from one of your client's suppliers before reading the licensing rights to the video.** After you read the rights, make sure your client agrees to them. Being named as a second party in a copyright infringement suit isn't a good thing.

- ◆ **If your client is recording a video for his Web site, which you or someone else will later compress and encode for the Web, make sure he uses a good-quality camcorder and not some el cheapo Web cam.** Garbage in, garbage out.
- ◆ **Make sure your client's intended audience has the necessary plug-ins to view the video.** When in doubt, list the required plug-in (Flash Player, Windows Media Player, and so on) and include a link to the Web site from which visitors can download it.
- ◆ **Apply the right level of compression for your intended audience.** If your client's intended audience accesses the Internet with dialup modems, don't embed a video with a 756 Kbps data rate.
- ◆ **If your client's intended audience accesses the Internet at different connection speeds, create links to videos with different compression rates.** Have one file with a data rate of 40 Kbps for dialup access, one file with a data rate of 400 Kbps for DSL access, and another file with a data rate of 756 Kbps for cable access.
- ◆ **Don't embed more than one video per Web page.** If your client wants multiple videos to be accessible from a page, use a video editing application to export one frame of the video as an image. Do this for each video the client wants to be accessible from the page. On the Web page, each photo serves as a link to the full video.

Chapter 4: Active Slideshow Pro

In This Chapter

- ✓ **Introducing Active Slideshow Pro**
- ✓ **Creating a slide show**
- ✓ **Adding effects and links**
- ✓ **Editing a slide show**

The first chapter of this minibook covers using Flash to add interactivity to your Web pages. Some people love Flash. Others would rather have a tooth pulled than learn how to use a new application. If you fall in the latter group and still want to add cool interactivity to your Web pages, you can easily do so from within Dreamweaver by using the Active Slideshow Pro extension. Active Slideshow Pro enables you to easily add Flash slide shows — slide shows with special effects such as motion and frame overlays — to your designs. In this chapter, we show you some of the cool things you can do with Active Slideshow Pro.

About Active Slideshow Pro

Active Slideshow Pro is a Macromedia Dreamweaver extension created by the folks at DMXzone. The extension makes it possible for you to add compelling multimedia slide shows to your Web pages. Adding a slide show is as simple as invoking a menu command, navigating to a folder of images you want to include in the slide show, and then choosing the options you want in your slide show. The options you can include are lengthy. You can choose an overlay for your slide show, which, incidentally, is a Flash SWF file. You can add motion, cool transitions, dazzling text effects, and more to your slide shows.

If you want a truly custom slide show, you can add your slides one at a time, add a text overlay, control motion for the slide and text independently, add a custom fade between slides, and more. You can also add a music soundtrack to your slide show, complete with a Mute button that site visitors can opt for if they don't like your choice of music. The options allow you to create a one-of-a-kind slide show guaranteed to dazzle site visitors. As of this writing, Active Slideshow Pro sells for \$149. For more information, visit

www.dmxzone.com/ShowDetail.asp?NewsId=11360

The Ken Burns effect

Ken Burns is a documentary filmmaker. He intersperses his documentaries with original prints and photographs. Many of his documentaries are about periods of history when only still photography was available. Burns often gives life to still photographs by slowly zooming in on subjects of interest and panning from one subject in the image to another. For example, in a photograph of a baseball team, he might slowly pan across the faces of the players and come to rest on the player whom the narrator is discussing. Ken Burns didn't originate this technique, but he is so associated with it that it is now known as the *Ken Burns effect*.

The effect is also used when transitioning from one clip to another. For example, to segue from

one person in the documentary to another, he might open a clip with a close-up of one person in a photo and then zoom out to make another person in the photo visible. Burns used this technique extensively when creating documentaries about subjects who lived when video cameras weren't available or in wide use. The zooming and panning across photographs creates the feeling of motion and keeps the viewer visually entertained.

Active Slideshow Pro gives you the capability to pan across a photo and zoom in, thereby replicating the Ken Burns effect. This technique has become a staple of documentaries, slide shows, presentations, and even computer screen savers.

Installing the Extension

You purchase the Active Slideshow Pro extension from DMXzone.com, where it's available as a download. After you download the extension to your hard drive, follow these steps to install the extension:

1. Double-click the extension.

Active Slideshow Pro is an MXP file, which when double-clicked launches the Adobe Extension Manager CS4. After the Adobe Extension Manager CS4 launches, a dialog box appears with a disclaimer from Adobe. This is included because any developer can create extensions, and Adobe doesn't claim responsibility for them. However, you can safely install extensions from a reputable source like DMXzone.

2. Click Accept to agree to the disclaimer.

Adobe Extension Manager CS4 begins installing the extension. After the extension is installed, the Adobe Extension Manager CS4 dialog box displays information about the extension. (See Figure 4-1.)

3. Launch Dreamweaver.

The Install Active Slideshow Pro 1.5 dialog box appears. It looks like many other dialog boxes, so we won't waste valuable resources by including a picture of it. You need an Internet connection to complete the installation process.

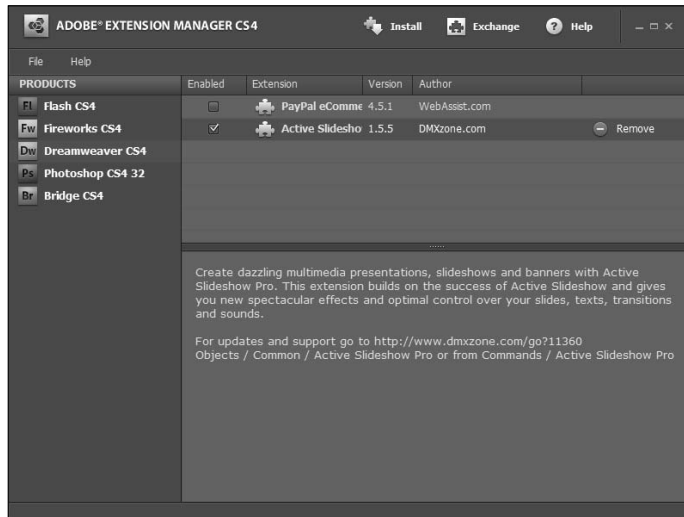


Figure 4-1:
Installing
the
extension.

4. Click Next.

The EULA (End User Licensing Agreement) appears.

5. Click I Agree.

If you don't, you won't be able to install the extension. After agreeing to the EULA, the dialog box refreshes.

6. Enter your serial number and then click Next.

The dialog box refreshes.

7. Enter your DMX username and password, and then click Next.

The dialog box refreshes, with a message thanking you for registering the product.

8. Click Next.

The installation process proceeds, and then a message appears telling you the installation is complete.

9. Click Close and restart Dreamweaver.

When you restart Dreamweaver, the plug-in appears on the Commands menu.

Creating a Slide Show

After installing the extension, you're ready to rock and roll and create your first slide show. You access Active Slideshow Pro from within Dreamweaver. The resulting Flash file is linked to your HTML document. You can use a

slide show anywhere, but a slide show is particularly effective on the home page of a Web site. It's a great way to pique Web site visitor interest and show people what your client does. If your client is a photographer, it's a really classy way to showcase his best pieces to potential customers.

Launching Active Slideshow Pro

To create a slide show, you begin with an HTML document. You can create the slide show anywhere in the document. However, a slide show is a thing of beauty, and a thing of beauty is a joy forever. Therefore, it makes sense to give your slide show a place of prominence on the page. To start your slide show, follow these steps:

1. Create an HTML document.

Start with a template from the site that includes the navigation menu, site banner, and all the other associated accoutrements. (See Figure 4-2.)

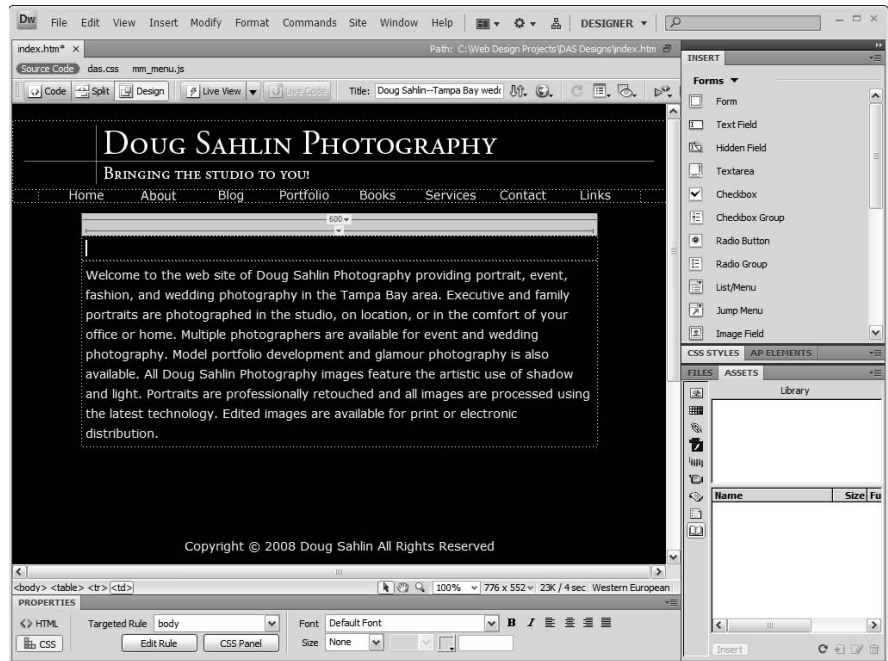


Figure 4-2:
Start with
an HTML
document.



2. Save the document.

The document must be saved prior to adding a slide show. If you omit this step, a warning dialog box to that effect appears.

3. Position your cursor where you want the show to appear and then choose **Commands** → **Active Slideshow Pro**.

The Active Slideshow Pro dialog box appears. (See Figure 4-3.)

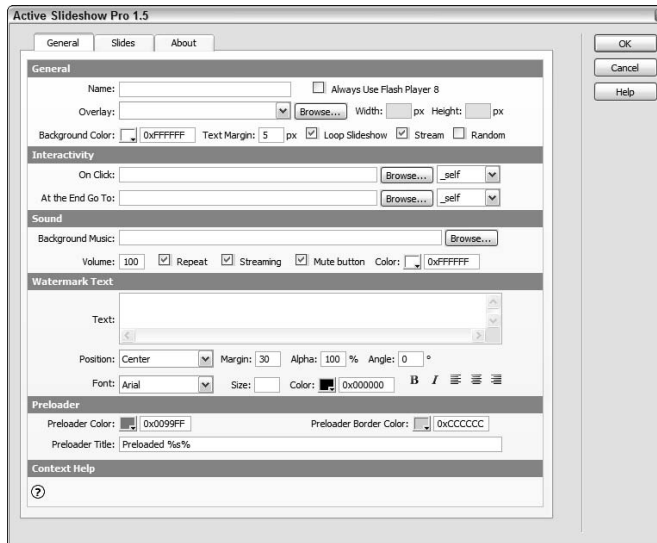


Figure 4-3:
Launching
Active
Slideshow
Pro.

4. Enter a name for the slide show.
5. (Optional) Select the **Always Use Flash Player 8** check box.
This option determines whether the slide show works in only the Flash 8 Player. If you choose this option, make sure you inform site visitors that the Flash 8 Player is required and provide a link they can use to download the player.
6. (Optional) Choose an overlay from the **Overlay** drop-down menu.
Overlays are SWF files that play over the slide show. They run the gamut from fancy frames to moving objects and frames. If you choose an overlay, the application fills in the Width and Height fields.
7. Click the **Background Color** swatch to choose the background color for the slide show.
If you specify a border, choose a color that complements the background color of the HTML page; otherwise, choose the same color as the document. Alternatively, you can enter a hexadecimal value in the text field to specify the background color.
8. Accept the default **Text Margin** value of 5 pixels or enter another value.

This option determines how far text is indented. If you choose an overlay, make sure you specify a large enough value so that the text isn't partially eclipsed by the overlay.

9. (Optional) Select the Loop Slideshow check box.

If you choose this option, the slide show starts again at the beginning after it finishes.

10. (Optional) Select the Stream check box.

This option begins playing the slide show and music while data is still streaming into the user's browser.

11. (Optional) Select the Random check box.

This option plays the slides in a random manner.

12. (Optional) If desired, enter a URL in the On Click field.

If you enter a URL, site visitors can launch the Web page by clicking the slide show. The option is available only if you deselect the Loop Slideshow option. (See Step 9.) You can also specify a file within the Web site, such as an image that is the same size as the slide show.

13. (Optional) If desired, enter a URL in the At the End Go To field.

This option is available if you deselect the Loop Slideshow option. (See Step 9.) It loads the Web page when the slide show is finished. You can also specify a file within the Web site, such as an image that is the same size as the slide show. The file or Web page does, however, load in the same window if you choose `_self`. When you're creating a slide show for a Web page, we don't recommend using this option because it takes the viewer away from the client's Web site.

14. Click the Background Music Browse button.

This enables you to browse to an MP3 file to use as background music for your slide show. After selecting the file, click OK or Open to add the music to the slide show.

15. Accept the default Volume value of 100 or enter another value.

The default option plays the soundtrack at 100 percent of its volume. Enter a lower value to play the soundtrack at a lower volume.

16. Specify the other options for your soundtrack as follows:

- *Repeat*: The default option repeats the soundtrack, as long as the slide show is playing.
- *Streaming*: Begins playing the soundtrack while the data is still streaming into the user's browser.
- *Mute Button*: Determines whether a mute button is displayed with the slide show. If you accept this default option, users can click the button to mute the sound.

- *Color*: Enables you to choose the color for the Mute button from the Color Picker. The default color is white.

17. If you want to display watermark text on the slides, specify the following options in the Watermark Text section:

- *Text*: Type the text you want to display over each slide.
- *Position*: Choose an option from the drop-down menu to determine how the text is positioned over each slide. The options are self-explanatory.
- *Margin*: Determine how much the text is indented. Specify a larger value if you're using an overlay with your slide show.
- *Alpha*: The default option of 100% displays the text at full opacity. Specify a lower value if you want the text to be partially transparent and let some of the underlying slide show through. If your client is a photographer or the photos are important to the Web site, choose an opacity value of less than 50% to display the photos with minimum intrusion from the watermark.
- *Angle*: Enter a value to tilt the watermark text. A value of 0 or 45 works well in most instances.
- *Font*: Choose an option from the drop-down menu.
- *Size*: Enter the desired value in the text field. A value of 30 pixels works well in most cases.
- *Color*: Click the Color swatch and choose the text color from the Color Picker. Choose a color that you can see over the images.
- *Style*: Choose one or more options to style the text. Your choices are Bold and/or Italic.
- *Align*: Choose an option to align the text. Your options are Left, Center, or Right.

18. Specify the following options in the Preloader section:

- *Preloader Color*: Click the Preloader Color swatch and choose the preloader color from the Color Picker.
- *Preloader Border Color*: Click the Preloader Border Color swatch and choose the color of the preloader border from the Color Picker.
- *Preloader Title*: Accept the default preloader title or select the text and type the title you want. Loading – Please Wait is a good option.

Adding slides to your show

A slide show with no slides is like popcorn with no pop, which, incidentally, is corn. But you're not creating a corny slide show, so the next logical step after you get past the General tab of the Active Slideshow Pro dialog box is to add some slides as follows.

1. Specify the general options for your slide show as specified in the previous section.

If you haven't read the previous section, please do. Doug spent a good amount of time writing it in his favorite Wi-Fi café. Besides, this section makes more sense if you do read the previous section.

2. Click the Slides tab.

The Active Slideshow Pro dialog box reconfigures to show the options for adding slides. (See Figure 4-4.)

Add Slide button

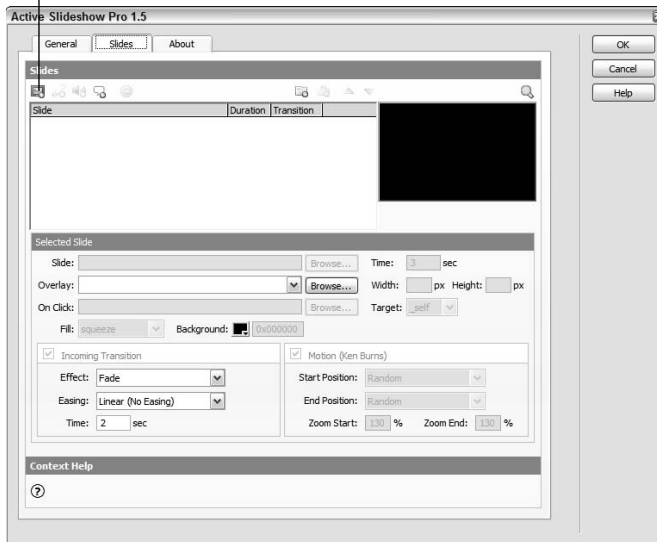


Figure 4-4:
Adding a slide.

3. Click the Add Slide button.

A folder icon appears in the Slides window.

4. In the Selected Slide section, do the following:

- a. Click the Browse button to the right of the Slide field.

This opens the Select File dialog box.

- b. Navigate to and select the slide you want to start your show. Then click OK to return to the Slides tab.

The first slide is the first impression site visitors will get of your client's handiwork. Hit them with your best shot.

- c. Accept the default time of 3 (seconds) or enter a different value.



This determines the number of seconds for which the slide is displayed after the transition from the previous slide ends.

d. *Choose an option from the Overlay drop-down menu.*

If you choose an overlay for an individual slide, it overrides (for this slide only) the overlay you chose from the General section. I advise you to choose one overlay style and stick with it. The slides, not the overlays, are the stars of the show.

e. *Enter a filename or URL in the On Click field.*

This option enables the visitor to click the slide and load a Web page or file. You can manually enter the URL or filename or click the Browse button and navigate to the desired file. This supersedes what you specified for this option in the General tab.

f. *Choose a fill option.*

This determines how the slide is displayed if you don't choose an Incoming Transition option. Your choices are

- **Center:** Centers the image in the slide window without resizing the image.
- **Stretch:** Fills the image across the entire slide window. This option can change the image proportions.
- **Squeeze:** Resizes the image while maintaining the original proportions. This option makes the entire image visible within the slide. The image has borders, however, if the image proportions are different from those of the slide.
- **Trim:** Resizes the image while preserving the original proportions. This makes the entire image visible within the slide. The image has borders if the image proportions are different from those of the slide.

g. *Choose a background color.*

Click the Background color swatch to determine the slide background color. This overrides the background color you might have chosen in the General section.

5. Select the Incoming Transition check box.

This option is selected by default. If you accept the option, choose the effect you want for the slide transition from the Effect drop-down menu:

- *Fade:* This option gradually increases the opacity of the incoming slide until it's fully visible at the end of the transition period.
- *Slide In:* The incoming slide transitions from the direction specified in the option you choose and overlaps the previous slide. For example, if you choose Slide In Right, the incoming image transitions from the right.

- *Push*: The incoming slide pushes the previous slide out of frame, transitioning in from the direction specified in the option you choose. For example, if you choose Push Left, the incoming slide transitions from the left and pushes the previous slide to the right.

6. Choose an option from the Easing drop-down menu.

Easing allows you to add special effects to the transition, such as acceleration or elastic movement. The best way to understand what each choice does is to experiment and remember which effects you like best.

7. Accept the default Time value of 2 (seconds) or enter a different value.

This option determines how long it takes the incoming slide to complete its transition.

8. Select the Motion (Ken Burns) check box.

This option is selected by default and it enables you to create a slide show with the Ken Burns effect. (Read all about this in the sidebar in this chapter.) If you accept this option, specify the following options:

- *Start Position*: Choose an option from the Start Position drop-down menu. You can choose Random, which moves the slide into the frame from a random direction. This emulates the original Ken Burns effect. Alternatively, you can choose one of the Slide From options, which slides the image from a specified direction.
- *End Position*: Choose an option from the End Position drop-down menu. You can choose Random, which places the slide in a random position at the end of its duration. Alternatively, choose one of the Slide To options, which slide the image to a position specified in the option you choose.

9. Accept the default Zoom Start value of 130% or enter a different value.

This determines how large the image is when it appears in the slide window. If you specify 100, the image fills 100 percent of the frame; larger values magnify the image by the value entered.

10. Accept the default Zoom End value of 130% or enter a different value.

This determines how large the image is at the end of its duration. Specify a lower value than the Zoom Start value to zoom in and a higher value than the Zoom Start value to zoom out.

11. Click the Interactive Preview button, which looks like a Play button, at the lower-right corner of the slide preview.

This previews the selected slide with the parameters you applied. The settings are applied to the selected slide. You can have different settings for each slide. You can now add additional slides to your show and modify the settings for each image.



If you planned ahead and collected all the photos for the slide show in one folder, click the Folder icon and then follow the prompts to add the folder of images to the slide show. After adding a folder of slides, click each slide in the Slides window to set Incoming Transition, Motion, and other parameters. Why work harder?

12. After adding slides to your show, you can modify the order in which they appear by selecting a slide and then clicking the Up or Down arrow.



If you decide that a slide doesn't belong in the show, select it and then click the Delete button — it looks like a minus (–) sign.

Adding text and sound to a slide

You can display text on a slide and also have a sound play when a slide appears. The sound is in addition to any background music you select in the General tab. Logic dictates that you shouldn't display watermark text and text on an individual slide. In fact, we rarely use text overlays in slide shows. Let the images do the talking. Adding tasteful background music is, however, a nice touch. To add text and sound to a slide, follow these steps:

1. Select the slide to which you want to add text.
2. Click the Add Text button, which looks like a pencil.

The Slides tab reconfigures to show options for adding text. (See Figure 4-5.)

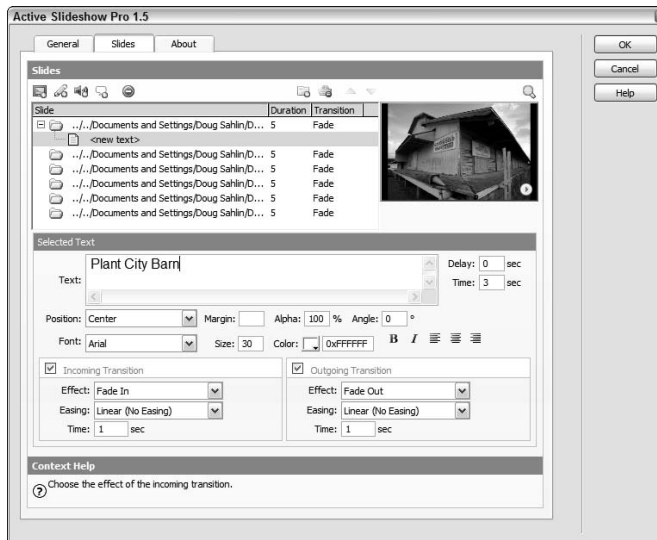


Figure 4-5:
Adding text to a slide.

3. Accept the default Delay value of 0 (seconds).

The default value displays the text as soon as the image fully transitions into the slide window. If you specify a value larger than 0 (zero), the text doesn't appear until the duration expires.

4. Accept the default Time value of 3 (seconds) or enter a different value.

This option determines the length of time for which the text displays over the image.

5. Enter the desired text in the Text field.**6. Specify parameters for the text. Your options are as follows:**

- *Position*: Choose an option from the drop-down menu to determine how the text is positioned over each slide. The options are self-explanatory.
- *Margin*: Determines how much the text is indented. Specify a large value if you're using an overlay with your slide show; otherwise, the overlay might hide some of the text.
- *Alpha*: The default option of 100% displays the text at full opacity. Specify a lower value if you want the text to be partially transparent and let some of the underlying slide show through.
- *Angle*: Enter a value to tilt the text.
- *Font*: Choose an option from the drop-down menu.
- *Size*: Accept the default value of 30 pixels or enter a different value.
- *Color*: Click the Color swatch and choose the text color from the Color Picker. Choose a color that you can see over the images.
- *Style*: Choose one or more options to style the text. Your choices are Bold and/or Italic.
- *Align*: Choose an option to align the text. Your options are Left, Center, or Right.

7. Select the Incoming Transition check box.

This option is selected by default. This option adds a transition effect to the text as the image appears. If you accept this option, choose an option from the Effect drop-down menu. Your choices are

- *Fade In*: The text gradually fades in from transparent to full opacity.
- *Zoom In*: The text zooms into view.
- *Blur In*: The text is blurry when it first appears and is clear when the transition period ends.

8. Choose an option from the Easing drop-down menu.

This determines the type of motion that occurs when the text comes into view. The best way to understand what each choice does is to experiment and remember which effects you like best.

9. Accept the default Time value of 1 second, or enter a different value.

This determines the duration of the incoming transitions.

10. Select the Outgoing Transition check box.

This option is selected by default. If you accept this option, choose an option from the Effect drop-down menu. Your choices are

- *Fade Out*: The text gradually fades out from full opacity to transparent.
- *Zoom Out*: The text zooms out of view.
- *Blur Out*: The text is clear when it is displayed and is blurred during the transition out.

11. Choose an option from the Easing drop-down menu.

This determines the type of motion that occurs when the text goes out of view.

12. Accept the default Time value of 1 second, or enter a different value.

This determines the duration of the outgoing transitions.

13. To have a sound play when the slide is displayed, click the icon that looks like a speaker.

This reconfigures the dialog box for sound options.

14. Click the Browse button, and then navigate to and select the sound.

Select a sound with a duration of a second or two. If you specify a sound with a longer duration, it continues playing while the next slide loads. A perfect example of that is a shutter click for a photographer's slide show.

15. Accept the default Volume value of 100 or enter a different value.

The default value plays the sound at 100 percent volume. Enter a lower value to lower the volume of the sound.

16. Accept the default delay of 0 (seconds) or enter a different value.

This option determines whether the sound is delayed; and, if so, by what amount of time.

17. Accept the default Repeat value of 1 or enter another value.

The default value repeats the sound once. For a slide with a display time of 3 to 5 seconds, the default is the ticket. Besides, too much noise is going to detract from the beauty of the images.

18. To add a comment to a slide, click the Comment icon.

This displays the Comment field. Type the desired comment. Comments aren't displayed with the published slide show. You can use a comment as a note to yourself about a particular slide.

19. Click OK to add the slide show to your Web page.

Active Slideshow Pro does its work quickly and adds a Flash file to your Web page.

20. Press F12.

Dreamweaver prompts you to save the file. After saving the file, it's displayed in your default browser. (See Figure 4-6.)



Figure 4-6:
Previewing
the slide
show in a
browser.

Editing a slide show

After you create a slide show and preview it, you might find that you need to tweak it. You can make minor tweaks in Dreamweaver and, if need be, edit the show in the Active Slideshow Pro extension. To edit a slide show, follow these steps:

1. Select the slide show.

It's the big rectangle with the capital F for Flash.

2. Choose Window→Properties.

The Property inspector opens. From within the Property inspector (see Figure 4-7), you can edit many of the same parameters found in the General tab of the Active Slideshow Pro dialog box.

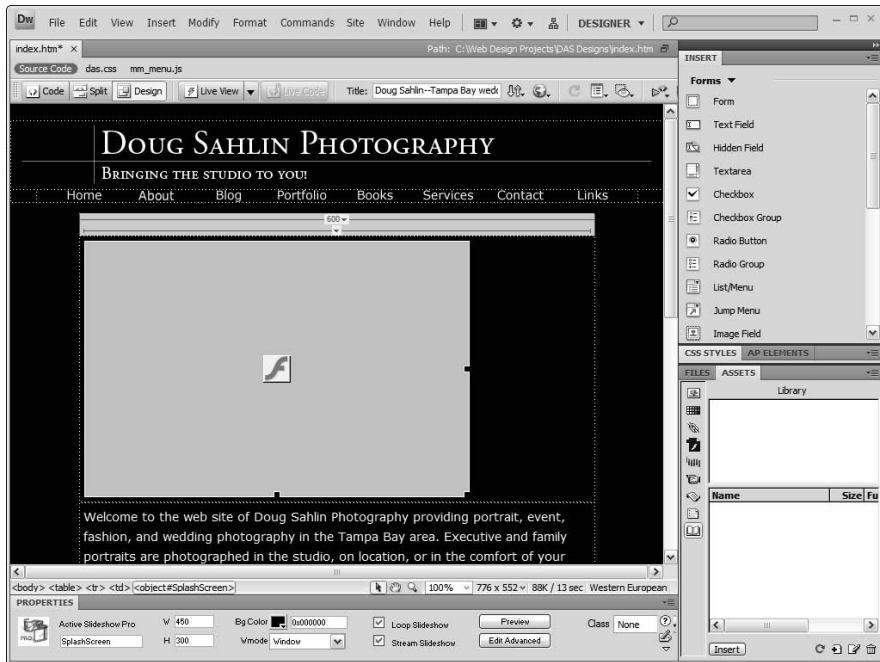


Figure 4-7:
Editing
slide show
properties in
the Property
inspector.

3. Click Preview.

The slide show previews from within Dreamweaver.

4. Click Edit Advanced.

The slide show appears in the Active Slideshow Pro dialog box.

5. Make your edits and then click OK.

Book VI

Audience Interaction



Creating a user form.

Contents at a Glance

Chapter 1: Adding Basic Interactivity with HTML Forms and JavaScript	473
Understanding Interactivity	473
Getting Ready to Code	474
Building an HTML Form	476
Adding Elements to Your Form	483
Event Handlers and JavaScript	494
Making Dynamic Pages with Server Side Technology	496
Chapter 2: An Introduction to ASP	499
Understanding ASP	499
Creating ASP Pages and Testing Them on Your Local System	501
Creating ASP Pages in Dreamweaver	506
Discovering Cool ASP Tricks	513
Chapter 3: Introduction to PHP	519
Defining PHP	519
Fulfilling PHP Server Side Requirements	520
Creating PHP Pages	520
Declaring and Using Variables	521
Working with Conditional Statements	523
Repeating Lines of Code, Using Loops	526
Generating a Random Image	528
Chapter 4: MySQL and PHP	531
Creating a Mailing List Form	532
Creating the Database	534
Creating the PHP Code	537
Retrieving Information from a Database	539
Chapter 5: Additional Site Interactivity	543
Engaging Visitors with an Online Journal (Blog)	543
Initiating Online Conversations: Forums and Message Boards	545
Establishing an Online Community: A Wiki	548
Feeding Them Your Content: RSS	550
Branding through E-Mail	560
Keeping Them Up to Date: Newsletters	561

Chapter 1: Adding Basic Interactivity with HTML Forms and JavaScript

In This Chapter

- ✓ Engaging Web site visitors with interactive content
- ✓ Considering coding
- ✓ Creating HTML forms
- ✓ Adding buttons and boxes to Web forms
- ✓ Working with password fields, hidden fields, and text areas
- ✓ Giving Web site visitors multiple choices
- ✓ Using JavaScript event handlers
- ✓ Thinking about server side technology

Interactivity is a wonderful thing. You can employ it to make a Web site dance and sing. As Web designers, though, we are devout cowards when the need arises to write code. However, you can create many forms of interactivity without needing to write the first word of code. The Web form is a perfect example, comprising items such as text fields, check boxes, radio buttons, submit buttons, and so on. You can create one in an application like Dreamweaver by merely choosing the proper menu commands. In this chapter, we show you how to create forms as well as how to master JavaScript event handlers.

Understanding Interactivity

An *interactive Web site* does more than appear on your monitor: It has bells and whistles that the user can interact with. Any Web site that requires input from a visitor and delivers different content based on user input can be considered interactive. Interactivity can be as simple as greeting a visitor based on the name she inputs in a Web form, or as complex as delivering information from a database. You can add a lot of interactivity to your designs by adding a bit of JavaScript code to a page, by creating a DHTML (dynamic HTML) page, or by using VBScript (Visual Basic Script).

The most complex types of interactivity take place on the server side and involve adding code to a page that's interpreted by server software. ASP, PHP, and CGI/Perl are examples of programming languages that are interpreted on the server side. (Read all about these in Book III, Chapter 8.)

Now, just because we mention a lot of scary acronyms that are related to code, don't think this chapter has a lot of geek-speak. We mention code only where we need to. In most instances, you won't have to write any code — and when you do, we explain it in terms that you're sure to understand. Now that you have an idea of what interactivity is, it's time to get interactive.

Getting Ready to Code

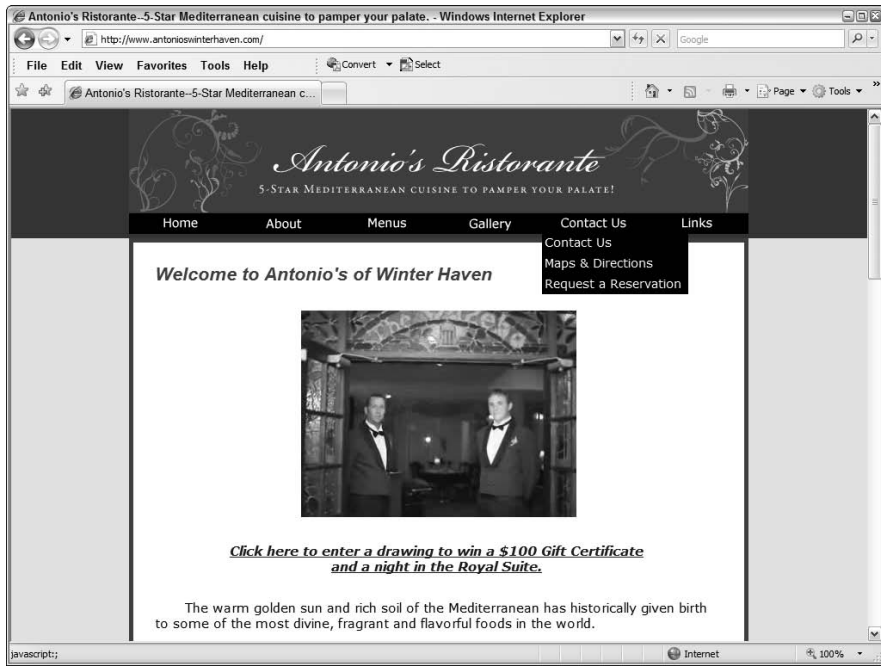
Yes, at times, you do need to manually write code for the pages you design — and that means learning a language, like JavaScript, VBScript, or one of the other Script Sisters. That's also, like, a lot of work. In most cases, though, you don't have to learn an entire programming language, just like you don't have to know the entire English language to communicate.

Working with code is kind of like working for the government on a need-to-know basis. You learn only as much code as you need to know to get the job done. In many instances, a good HTML editor (say, Dreamweaver) can create the difficult code for you. All you need to know is which menu command to choose. Fireworks (Dreamweaver's partner in interactivity) can also create code for you. For example, in Fireworks, you can create a functional navigation menu, with a drop-down menu thrown in for grins and giggles.

And you can exercise your creativity in graphic design when creating the menu. Then, export the menu as images and HTML. Figure 1-1 shows a menu as created in Fireworks. After exporting the file as HTML and images, the images are all neatly stored in a folder, and the code that most designers couldn't even think of creating is stored in an HTML file. To get the menu into one of your designs, you use the Fireworks HTML command in Dreamweaver to plop the menu (all neatly sliced and diced inside a table), as well as all the code needed to make the menu work, in your document.

Figure 1-2 shows just some of the code created by Fireworks. In all, Fireworks generated more than 100 lines of code for the menu. Can you imagine writing all that by hand? No wonder hardcore Web developers have carpal tunnel syndrome and an addiction to Naproxen.

Figure 1-1:
A drop-dead gorgeous menu with a drop-down menu.



Book VI
Chapter 1

Adding Basic
Interactivity with HTML
Forms and JavaScript

Figure 1-2:
A whole lotta scary code.

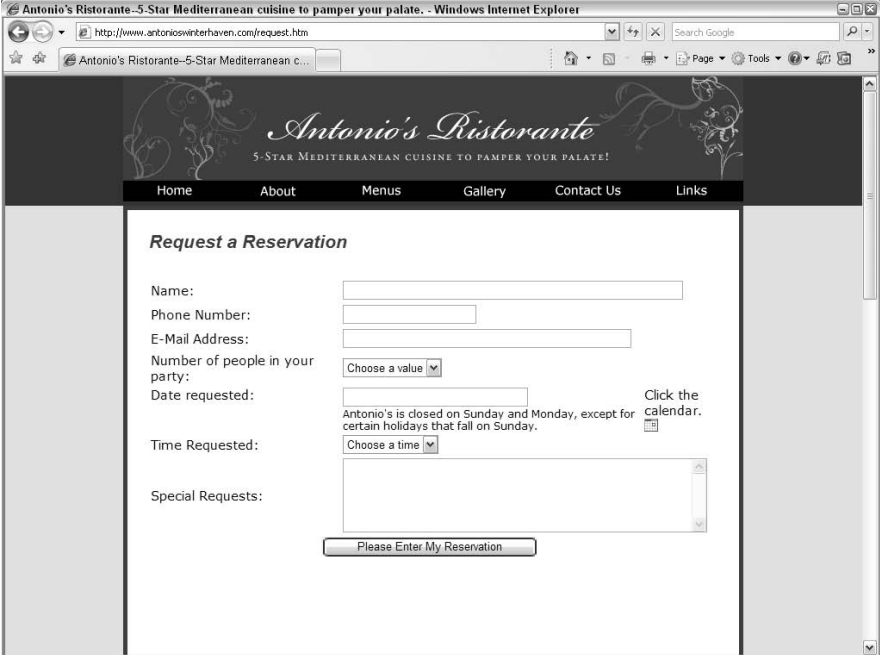


But what happens when the dynamic duo of Dreamweaver and Fireworks can't create the effect you're after? You can find lots of code on the Internet. All you need to do is type the desired programming language and examples into your favorite search engine. You'll have lots of results from which to choose. Granted, you'll have to change some of the code to suit your design, but that's a lot easier than thumbing through a manual on the programming language that's heavy enough to be a doorstop.

Another alternative is to grab someone else's code, provided that it's not on a page generated by server side technology. If you see a way-cool effect on a Web site that you want to mimic on your Web site, you can view the code by choosing the proper View Source command from your Web browser. You must remember, though, that there are only about 27 original ideas. Everything else is a mutation, permutation, or combination of the original ideas.

Building an HTML Form

One of the most basic and useful forms of interactivity is the lowly Web form. You see them on just about every Web site. *Forms* are used to gather information for mailing lists, add entries to blogs, subscribe to RSS (really simple syndication) feeds, and more. Figure 1-3 shows a form used by a restaurant. In the following sections, we show you how to create a Plain Jane form that gets 'er done.



The screenshot shows a web browser window titled "Antonio's Ristorante - 5-Star Mediterranean cuisine to pamper your palate. - Windows Internet Explorer". The address bar shows "http://www.antonioswinterhaven.com/request.htm". The page features a dark header with the restaurant's name "Antonio's Ristorante" and the tagline "5-STAR MEDITERRANEAN CUISINE TO PAMPER YOUR PALATE!". Below the header is a navigation menu with links: Home, About, Menus, Gallery, Contact Us, and Links. The main content area is titled "Request a Reservation" and contains the following form fields:

- Name:
- Phone Number:
- E-Mail Address:
- Number of people in your party:
- Date requested: Antonio's is closed on Sunday and Monday, except for certain holidays that fall on Sunday. [Click the calendar.](#)
- Time Requested:
- Special Requests:

At the bottom of the form is a button labeled "Please Enter My Reservation".

Figure 1-3:
An HTML
form in all
its glory.

Form tags: <form> </form>

Everything has a beginning. In the case of a form, the `<form>` tag designates where the form starts in a document. And a form's got to end somewhere; it's a law. The `</form>` tag neatly bookends the form. If you're a geek, you type the beginning and ending `<form>` tags, and for that matter, all the code in between. If you're a smart Web designer, though, you let your HTML editor do the heavy work for you. If you use Dreamweaver, you're in luck. Dreamweaver has a Forms toolbar and a menu devoted to creating form elements.

Your HTML editor can take care of another task for you: It lets you specify both the action that occurs when the form is submitted and the method used. The action can be another Web page, or it can be a script on the server, such as a mail-forwarding script. If the form is appending information on a Web page, `GET` is the method. If the form is being used to send information to a user, or to post information to a database, the method used is `POST`.

Text fields: <input type="text">

You can add all manner of elements to a form. One of the most useful form elements enables visitors to add text. When you add a text field to a form, you give the field a unique name. When the information is submitted, the name of the field is used in the ensuing code. For example, the information in the text field can be inserted into a variable, which has the same name as a field in a database. The name of the text field is what the programmer uses when creating the code to make the leap from the text field to the variable to the database.

When you create a text field, you specify the width of the text field. You also determine whether the text is a single line, multiple lines, or password text. Another very helpful option is the ability to limit how many characters an end user can type into the field. If you don't limit the number of characters for a text field, you end up having a character (um, the smart-aleck type that likes to wreak havoc) enter 50 pounds of gibberish in your svelte, 25-character field.

Submit buttons: <input type="submit">

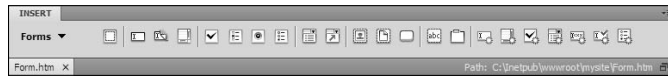
The `</form>` tag might be the end of your form in HTML, but it isn't the end of the form as far as the user is concerned. The last interactivity (you might remember that this is a chapter on interactivity) the user sees is a button that enables him to submit the form — a button that might be accompanied by its faithful companion, Reset. The Submit button executes the action specified by the beginning `<form>` tag. When you add a button to a form, you also have the option to change the default text, change the style of the dull and drab default text with a style from your style sheet, or choose Submit Form, Reset Form, or None as the action. You choose None as the

action if you want the button to function as something else — for instance, a hyperlink that loads another Web page. Reset does just as you would expect, it clears all of the fields in the form in case the user changes her mind and wants to start from scratch.

Creating a simple form in Dreamweaver

The past few sections of this chapter cover the basic elements of a form. If you fast-forwarded to this section to create a form, you should probably at least skim the previous sections. It can make the process of creating a form a little easier. Now, you could *hard-code* — developer-speak for typing the code by hand — the form, but that's a lot of work. And if you don't know all the parameters, your form will be as useless as clothes in a nudist colony. The easy way to create a form is to choose menu commands from your HTML editor. This section shows you how to create a form in Dreamweaver CS4, which comes with a handy-dandy Forms toolbar, which is available when you work in the Classic workspace. (See Figure 1-4.) If you work with the Designer workspace, you access everything from the Form menu, found under the Insert menu.

Figure 1-4:
The Forms
toolbar.



You can create a form within a new Dreamweaver document or add the form to an existing document. Follow these steps to create a simple form with a text box (or *text field*, as Dreamweaver calls it) and a Submit button to submit user information using a Web site's mail redirect script:

- 1. Launch Dreamweaver and open an existing document into which you want to insert a form. Or, open a new document.**
- 2. Switch to Design view and position your cursor where you want the form to begin.**
- 3. Choose Insert⇨Form⇨Form.**

Dreamweaver designates the form area with a dotted, red line.

- 4. In the Property inspector, enter the action that will take place when users submit the form.**

The action can be another Web page with a script or a script at a Web server. In the case of this form, the action executes a mail-forwarding script on a Web server.

Book III, Chapter 5 has more detailed information on using the Property inspector.

5. In the Action field of the Property inspector, specify the method by which the form data will be submitted.

You have three options:

- *Default*: Uses the browser's default method of submitting form data — which is usually `GET`
- *GET*: Appends the information to a URL
- *POST*: Sends the data as a body of data

6. Choose the target from the Target drop-down menu (if applicable).

This option is applicable only if the action calls another Web page that will display in the browser. The *target* is the window in which the page will be displayed. By default, the page will open in the same window. Your choices are

- *_blank*: Displays the document in a new and unnamed browser window.
- *_parent*: Displays the document in the parent window currently displaying the form.
- *_self*: Displays the document in the same window as the one in which the form is submitted.
- *_top*: Displays the document in the body of the current window. This option ensures the document that is called by the form action displays in the full browser window, even if the form were originally displayed in a frame.

7. If applicable, specify the MIME encoding type of the data from the Enctype drop-down menu.

The default encoding type of `Application/x-www-form-urlencoded` is generally used with the `POST` method. If you're creating a file-upload field, choose the `Multipart/form-www` encoding type.

8. Insert the desired form object. For the purpose of this demonstration, choose Insert → Form → Text Field to begin creating a text field.

A *text field* is the same thing as a *text box* — where users type information.

9. In the Input Tag Accessibility Attributes dialog box that appears, enter the desired information.

See Figure 1-5. This information is designed to aid disabled people who visit the Web site.



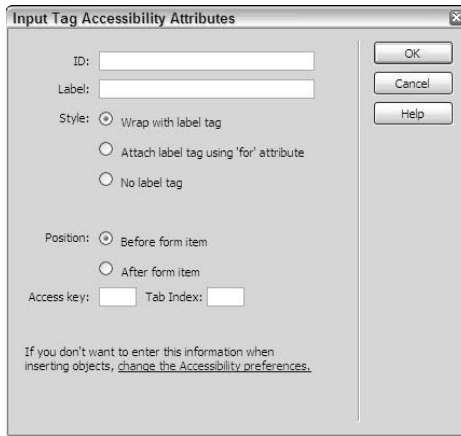


Figure 1-5:
Enter the attributes, please.

Even if your client’s site won’t have disabled visitors, the dialog box still has one useful attribute: namely, Label. For more information on usability and accessibility, see Book II, Chapter 2.

- 10. In the Label text box, type the desired label for the text field you’re creating; then click OK.**

This is the label — the text field name, to the left of the field — that visitors see. If you’re creating a text field that asks for an account number, your label would probably be something like Account Number. Go figure.

- 11. In the workspace, select the text input field you just created and fill in the parameters in the Property inspector. (See Figure 1-6.)**

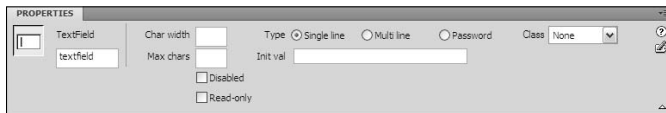


Figure 1-6:
Every form element has properties.

- 12. In the Property inspector, type a name in the name text field.**

The default name for a form object is its actual name, which is appended by a number if there is more than one in a document. In this case, you’re renaming a text field, which has a default name of `textfield`. Each form element must have a unique name. Names cannot contain spaces. Choose a logical name that reflects the type of information users will enter in the field. For example, if the text field requires the user to enter his first name, a logical name for the text field is `firstName` or `first_name`.

- 13. Enter the other parameters for the text field you’re creating.**

- *Type*: Sets the type of text field that will be displayed on the Web page. Choose Single Line to display the text field as a single line in the document, or choose Multi Line to display a text box in which users can type more than one line of text. Or, you can choose Password, which displays each character as an asterisk or a bullet (filled-in circle) to preserve the anonymity of the user's password. (Read how to set this in the upcoming section, "Password fields.")
- *Char Width*: Sets the maximum number of characters that can be displayed in the field. This number can be less than Max Chars, which specifies the maximum number of characters that can be entered in the field. The default value is 20.
- *Max Chars*: Specifies the maximum number of characters that can be entered in the field. You can use this parameter to limit the number of characters for a ZIP code field to five. In some browsers, an alert sounds if a user enters more than the Max Chars value.
- *Num Lines*: Becomes available if you specify Multi Line as the Type. This value determines the height of the text box, in lines.
- *Init Val*: Displays an initial value in the text field when the form loads. This parameter is useful for displaying instructions to the user. The user then selects the text and types the desired information into the field.
- *Class*: Lets you stylize the text entered in the text field by using a style from the document or an attached CSS.
- *Disabled*: The text field cannot be filled in. Use this option when a field is Read Only.
- *Read Only*: The information in the Init Value parameter is displayed. The field cannot be filled in, but the text can be selected. If you use this option and don't want the text to be selectable, enable the Disabled option as well.



14. Add other form fields as needed.

If you have a complex form with multiple fields, create a table and then add each element to the desired cell. This creates a good-looking form that's easy to use.

15. For the purpose of this demonstration, choose Insert ⇨ Form ⇨ Button to begin creating a button.

The Input Tag Accessibility Attributes dialog box, which you use in Step 9 of this list, appears again.

16. Enter the desired information in the Input Tag Accessibility Attributes dialog box and then click OK.

Dreamweaver adds the button to the document.

17. In the Property inspector, click the Submit Form radio button to select an Action.

While you're in the Property inspector, you can modify other attributes of the button. (See Figure 1-7.) For example, you can change the text displayed on the button by entering new text in the Value field. You can also apply a style to the button by choosing an option from the Class drop-down menu. The items on the Class menu are from styles defined within the document or within an attached style sheet.

Figure 1-7:
Setting properties for a Submit button.



18. Accept the default name for the button (Submit). Or, in the Property inspector, enter a different name.

Select the default name in the Value field, and type the desired name.

19. Repeat Step 16, only this time use the Property inspector to define a Reset button. In the Action area, select the Reset Form radio button.

Figure 1-8 shows a form with several text fields and a Submit button, as displayed in the Firefox Web browser.

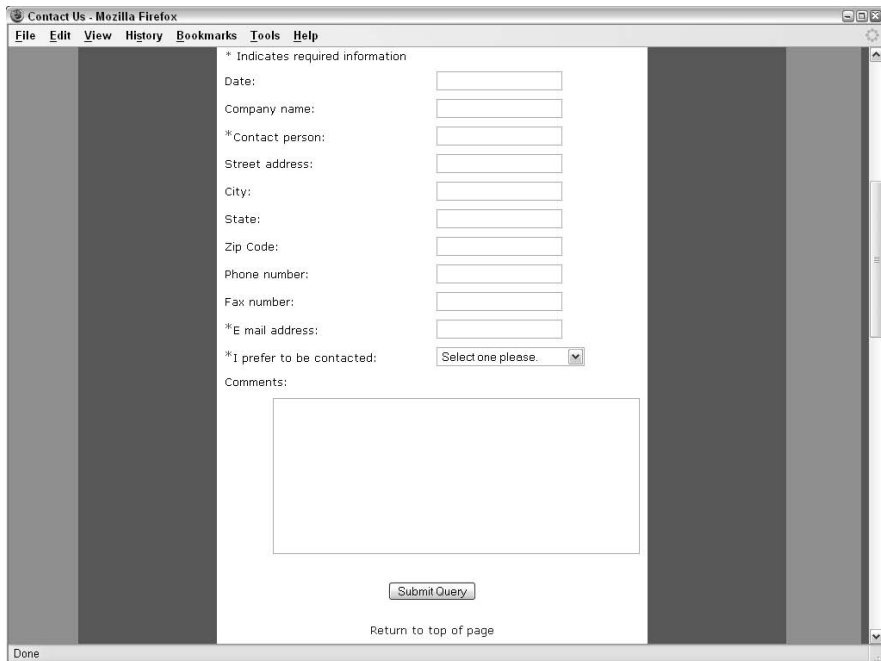


Figure 1-8:
Previewing the completed form.

Testing the form

After creating a form, test it prior to uploading it to your client's Web server. When you test a form, make sure that all the fields and elements in your form function perfectly. Enter data in each text field and use the other elements in your form, such as radio buttons, check boxes, and list boxes. Click the Submit button to make sure that the data transmits properly. Depending on the action, this might require being online or using a local testing server.

Adding Elements to Your Form

It would be difficult to get the job done if the only form elements available were a text field and a button. Fortunately, that's not the case. You can flesh out a form by adding check boxes, radio buttons, and list boxes. And then there are form elements that are so shy, they stay in hiding.

Password fields

If you're creating a form that requires a user to submit confidential information, such as a password to a Web site, you can preserve the anonymity of the information by creating a text field in the form and assigning Password to the Type parameter (in the Property inspector) of the Text Field form object. Listing 1-1 shows the code for a password text field.

Listing 1-1: A Password Text Field

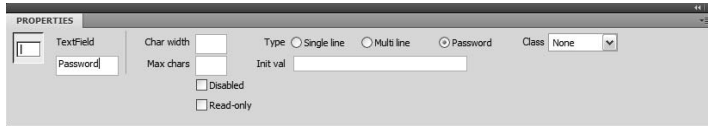
```
<input name="Password" type="password" id="Password"
      size="20" maxlength="20" />
```

Creating a password field is identical to creating a text field except that you choose Password as the field type. If you're creating a password field in Dreamweaver, choose Insert⇨Form⇨TextField and then choose the Password option in the Type section of the Property inspector. (See Figure 1-9.) When information is entered in the password field, it appears as asterisks or filled-in circles on the computer monitor.



If you're creating a form that requires a user to submit confidential information, such as a password to a Web site, you want to preserve the anonymity of the information. If you specify a field in your form as a password field, you can prompt users' Web browsers to replace each character they type with an asterisk or a filled-in circle.

Figure 1-9:
Choosing
the
Password
option for a
text field.



Hidden fields

Hidden fields are very useful form elements. They contain information pertinent to the form, such as the form method, but they don't need to be seen by the person filling in the form — hence their name, hidden. When you add a hidden field to a form, you add pertinent information to the form, such as the e-mail address of a form recipient. A hidden field has a name that indicates the purpose of the hidden field. Listing 1-2 shows the first few lines of code for a form used on a caterer's Web site.

Listing 1-2: A Form with Hidden Fields

```
<form method="post" action="http://www.myhypotheticalcateringsite.com
/cgi-bin/webform.cgi">
<input type="hidden" name="recipient" value=
"paul@ myhypotheticalcateringsite.com">
<input type="hidden" name="redirect" value=
"http:// myhypotheticalcateringsite.com/thankYou.htm">
<input type="hidden" name="subject" value="Catering Questionnaire">
```

The form is used in conjunction with a mail-forwarding script on the site's Web server. Typically, scripts reside in the site's cgi-bin folder. The script calls for the recipient's e-mail address. This is stored in the first hidden field. The script also redirects the visitor to another page after submitting the form. This information is stored in the second hidden field. The third hidden field contains the subject that appears in the e-mail that is sent to the recipient.



If you use an application like Dreamweaver, you don't have to write all the code. In Dreamweaver, choose Insert ⇨ Form ⇨ Hidden Field. Fill in the blanks in the Property inspector, and Dreamweaver writes the code for you.

Textarea form objects

Extra, extra, tell us all about it in a big-ol' text box. If your client wants a space in a form for his visitors to leave copious amounts of information, a text box — also known as a *text area* — is the ideal solution. You can specify the width of the text box and the number of lines in the text box, which in essence, sets the width and height of the text box. You can create a text box that is 80 characters wide and 15 lines tall by adding the code in Listing 1-3 to your form.

Listing 1-3: Code to Add a Text Area to a Form

```
<textarea name="textarea" cols="80" rows="15"
  wrap="physical"></textarea>
```

If you use Dreamweaver to create your Web pages, the solution is much simpler than writing the code yourself. Just choose Insert⇨Form⇨Textarea. By default, a textarea field contains multiple lines, and text wraps to the next line when a user enters text that exceeds the width of the field. After you add the field to the document, specify the following parameters in the Property inspector:

- ◆ **Char Width:** Sets the maximum number of characters that can be displayed in the field. This number can be less than Max Chars, which specifies the maximum number of characters that can be entered in the field. The default value is 45.
- ◆ **Init Val:** Displays an initial value in the field when the form loads. This parameter is useful for displaying instructions to the user. The user then selects the text and types the desired information into the field.
- ◆ **Class:** Allows you to stylize the text entered in the text field, using a style from the document or an attached CSS.
- ◆ **Disabled:** The resulting text field is grayed out when displayed in a Web browser. Text characters entered in the Init Val field are displayed.
- ◆ **Read Only:** The resulting text field is displayed in the document. Text characters entered in the Init Val field are displayed, but visitors can't enter text into the field.

A text box has a scroll bar that enables the user to scroll up or down if the entered text exceeds the text area. Figure 1-10 shows a text field in a form with the scroll bars on full alert.



Figure 1-10: Use the scroll bar if you need to enter more text in a field.

Multiple choice types (check box and radio buttons)

As a rule, people like it when you give them the opportunity to make a choice. Visitors to Web sites like choices, too; that's why some Web genius invented the navigation menu. But we're straying from the topic at hand, which is forms. When you want Web visitors to make a choice, they can easily do so by selecting a check box (or boxes) or a radio button. The value for the form element is transmitted with the form data.

Check boxes: `<input type="checkbox">`

When you add check boxes to a form, you give the visitor an option of making a choice — or choices — by clicking one or more check boxes. This is an efficient way of laying out a form when you require possible multiple responses to a question. You add the question to the form, and then list the answers, with a check box next to each answer. The value for the check box tells the form recipient what the visitor's answer or choice is. Listing 1-4 shows the code needed to add several check boxes to a form with a multiple-choice question.

Listing 1-4: Check Off, Check On!

```

<tr>
    <td colspan="7" valign="top"><div align="left">Will you require any
of the following services? Check all that apply. </div></td>
</tr>
<tr>
    <td colspan="7" valign="top"><div align="left">
        <table width="100%" border="0" cellspacing="0"
cellpadding="5">
            <tr>
                <td width="35%">Bar/Liquor</td>
                <td width="6%"><input name="bar/Liquor"
type="checkbox" id="bar/Liquor" value="Bar_Liquor"></td>
                <td width="49%">Espresso/Cappuccino bar </td>
                <td width="10%"><input name="Serving_Equipment"
type="checkbox" id="Serving_Equipment" value="Espresso_Cappuccino"></td>
            </tr>

```

If you use Dreamweaver to create your HTML forms, you add a check box to a document by choosing Insert⇨Form⇨Checkbox. After you add the element to the form, you set the parameters in the Property inspector. Check boxes have the following parameter choices:

- ◆ **Checkbox Name:** By default, Dreamweaver assigns the name `checkbox`, followed by a number if you add more than one to a form. Unless Doug is working on a drop-dead simple form with less than four check boxes, he renames the check box to something more descriptive. Each check box must have a unique name and cannot contain spaces or special characters.
- ◆ **Checked Value:** This is the value that's transmitted when the form is sent. In other words, this is the answer to your question, or the choice, depending on the type of form you're designing.
- ◆ **Initial State:** This determines whether the check box is checked or unchecked when the form loads in the user's browser.
- ◆ **Class:** This allows you to apply a class from a CSS style sheet to the check box.

Check box groups

If you need to give visitors the option of choosing multiple responses for a form option, you create a check box group. For example, if you're creating a questionnaire and you want to know what publications a visitor reads, you create a check box group, where the name of the group is something like `pubsRead`, and the choices are the names of magazines. The visitor can then select more than one option when responding to the query. In previous versions of Dreamweaver, this was tedious work because you had to give each button the same name in the Property inspector. In Dreamweaver CS4, though, you can create a check box group in a Jumping Jack Flash by doing the following.

1. Open an existing document in Dreamweaver and position your cursor where you want the check box group to appear.
2. Choose **Insert** ⇨ **Form** ⇨ **Checkbox Group**.

The Checkbox Group dialog box appears. (See Figure 1-11.)

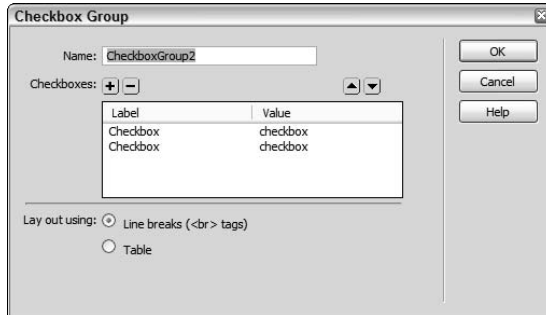


Figure 1-11:
Create a
check box
group here.

3. Change the default Name of the check box group.

Make sure you choose a name that makes sense. Remember to not include any spaces or special characters in the name.

4. Rename each Label and Value.

The Label is what users see, whereas the Value is what is transmitted with the data. If, for example, the group is a list of popular magazines, name one Label and Value `People`, and another Label and Value `Us`. You have two check boxes in the group, by default.

5. Click the plus sign (+) button to add additional check boxes to the group.
6. Rename the Label and Value of the new check boxes.

At this stage, you probably want to rearrange the check boxes to appear in alphabetical order.

7. To change the order in which the check boxes appear, click the check box label and then click ↑ or ↓ to rearrange the list.
8. Determine how the check boxes will be displayed and select the applicable Lay Out Using radio button to lay out the check boxes with line breaks or in a table.
9. Click OK.

Now wasn't that easier than the old way? Instead of mucking about in the Property inspector, you quickly create the group and then move onto more pressing matters — like another cup of coffee.

Radio buttons: <input type="radio">

Radio buttons are another option to use when you want users to make a choice — but only one choice. A radio button looks like a round dimple in the document that gets filled with black (or another color, if you play with CSS styles) when the user clicks the radio button, instead of a square (checkbox) that gets filled with a check. But the button doesn't tune into anything. Where they came up with the name *radio button* is beyond us.

For example, say you want the site visitor to choose which method should be used to reply to his query. You set up a group of radio buttons (with the input name of Response) with values of E-mail, Phone, Fax, and so on. The user is limited to only one choice from the group because they all have the same input name. Listing 1-5 shows the code for inserting a radio button in a form.

Listing 1-5: You Can Tune a Radio, but You Can't Tune a Radio Button

```
<input name="CreditCard" type="radio" value="Master Card" />
```

If you use Dreamweaver to create your HTML forms, you can add a radio button to a document by choosing Insert⇒Form⇒Radio Button. After you add the element to the form, you set the parameters in the Property inspector. Radio buttons have the following parameter choices:

- ◆ **Radio Button:** By default, Dreamweaver assigns the name `radiobutton`, followed by a number if you have more than one in your form. Unless Doug is working on a drop-dead simple form, he renames the radio button something more descriptive. When you want to restrict a visitor to only one choice from a group, all buttons will have the same name, and the Checked Value will be different.
- ◆ **Checked Value:** This is the value that's transmitted when the form is sent: in other words, the visitor's choice.
- ◆ **Initial State:** This determines whether the radio button is selected or not when the form loads in the user's browser.
- ◆ **Class:** This allows you to apply a class from a CSS style sheet to the radio button.



If you're a visual kind of designer, you can drag any form type from the Insert palette and drop it into the desired place in your form.

Radio groups

If you want to limit visitors to one choice, insert a group of radio buttons. The buttons all have the same input name but have different values. Therefore, a user can select only one radio button per group to be submitted

with the form. Listing 1-6 shows the code for a group of radio buttons with charge card type choices, all neatly housed in a table. Notice the name of each button is the same.

Listing 1-6: No Multiple Choice For You!

```
<table width="200">
  <tr>
    <td><label>
<input type="radio" name="CreditCard" value="Visa" />
Visa</label></td>
  </tr>
  <tr>
    <td><label>
      <input type="radio" name="CreditCard"
value="MasterCard" />
      MasterCard</label></td>
  </tr>
  <tr>
    <td><label>
      <input type="radio" name="CreditCard"
value="AmericanExpress" />
      American Express</label></td>
  </tr>
  <tr>
    <td><label>
      <input type="radio" name="CreditCard"
value="Discover" />
      Discover</label></td>
  </tr>
</table>
```

If you use Dreamweaver to create your HTML forms, you can easily insert a radio group as follows:

- 1. Open an existing document in Dreamweaver and position your cursor where you want the radio group to appear.**
- 2. Choose Insert→Form→Radio Group.**

The Radio Group dialog box appears. (See Figure 1-12.)

- 3. Change the default Name of the radio group.**

Give the group a name that makes sense. Just remember that you can't use spaces or special characters in the name.

- 4. Rename each Label and Value.**

The Label is what users see, whereas the Value is what is transmitted with the data. For example, if the group is a list of credit cards, you could name one Label and Value `Visa`, and name another Label and Value `MasterCard`. You have two buttons in the group, by default.

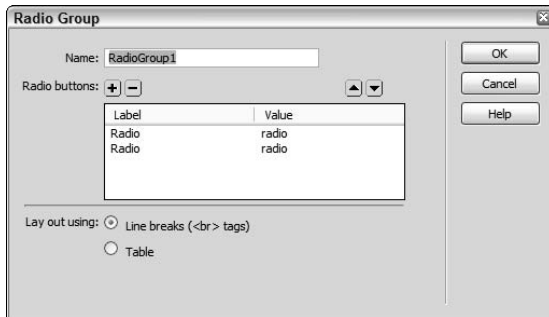


Figure 1-12:
Look! It's a
radio group.

5. Click the plus sign (+) button to add additional buttons to the group.
6. Rename the Label and Value of the new buttons.
At this stage, you probably want the buttons to appear in alphabetical order.
7. To change the order in which the buttons appear, click the button label and then click ↑ or ↓ to rearrange the list.
8. Determine how the buttons will display, and select the applicable Lay Out Using radio button to lay out the buttons with line breaks or in a table.
9. Click OK.

Look, Ma! I created a radio group without writing any code!

Drop-down lists

Drop-down lists are cool. They take up one little slot on your Web page, but click the little down arrow, and a list of choices appears. You can use drop-down lists anywhere in a form where you need to give the visitor lots of choices but don't want to waste a lot of real estate doing it.

When you create a drop-down list, you can enable the user to select multiple items. Listing 1-7 shows the code needed to create a drop-down list. Notice the lines of code that start with `<option value=`. These are the choices on the drop-down list.

Listing 1-7: Code for a Drop-Dead Gorgeous Drop-Down List

```
<select name="select" size="1">
<option value="Select one please.">Select one please.
  </option>
<option value="By phone">Mid-morning by phone</option>
<option value="Mid-afternoon by phone">Mid-afternoon
  by phone</option>
```

(continued)

Listing 1-7 (continued)

```
<option value="Early evening by phone">Early evening
    by phone</option>
<option value="By fax">By fax</option>
<option value="By E-mail">By E-mail</option>
<option value="By mail">By mail</option>
</select>
```

You can write out your own code, but if you've read this chapter from the start, you know there's a better way to build this mousetrap in Dreamweaver. Here's how you do it:

1. **Open an existing document in Dreamweaver and position your cursor where you want the drop-down list to appear.**
2. **Choose Insert⇨Form⇨List/Menu.**
3. **In the Input Tag Accessibility Attributes dialog box that appears (refer to Figure 1-5), fill in the desired values and then click OK.**

A drop-down list is born.

4. **In the Property inspector, click the List Values button.**

The List Values dialog box appears, as shown in Figure 1-13. Your cursor is hovering in the Item Label field, signifying that the dialog box is ready for you to do your thing.

5. **Type the name for the first Item Label.**

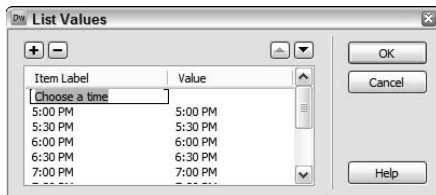
This is what visitors will see in your drop-down list. You have the option of displaying an initial value. This is a great way for you to tell visitors what you want them to do.

6. **Press the Tab key and then type a name for the Value.**

7. **Click the plus sign (+) button to add another option to the list.**

Yep, you guessed it. You have to type another value for the Label and Value. Figure 1-13 shows a drop-down list under construction.

Figure 1-13:
This List Values dialog box is filled with choices.



8. **Click the Item Label of a button and then click ↑ or ↓ to rearrange the order in which the buttons appear in the list.**

9. Click OK.

Dreamweaver inserts the drop-down list in your form.

10. In the Property inspector, choose the rest of your options. (See Figure 1-14.)

- a. Enter a name for the list in the text field in the upper-left corner of the Property inspector.

Dreamweaver gives the list a default name, such as select. It's considered good practice to give the object a name that reflects what it does in the document. The name can't contain spaces or special characters.

- b. Click the option you'd like displayed when the list loads from the Initially Selected window.

If you don't choose an option, the drop-down list window is blank.

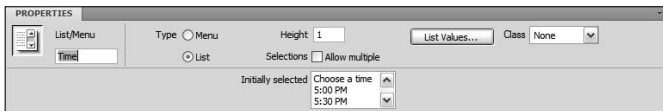
- c. Select the Allow Multiple check box if you want viewers to be able to select multiple items from the list.

To select multiple items, viewers must hold down Ctrl (Windows) or ⌘ (Mac) while clicking.

- d. In the Type area, select the List radio button to have the form object function as a drop-down list, or select Menu to have the form object function as a jump menu.

Of course, if you choose Menu, your values have to be URLs to the pages you want to open when users click the labels. If you select Menu, the option to allow multiple selections is grayed out, which is logical because even the best multitasker can't navigate to more than one page at a time.

Figure 1-14:
Choose options for the drop-down list.



You can add other elements to a form by choosing Form⇨Insert and choosing an option from the submenu, including the following:

- ◆ **Image Field:** Insert an image in a form.
- ◆ **File Field:** Insert a Browse button and field in a form. (You see this type of field on a form that wants you to attach a file saved on your hard drive.)

- ◆ **Fieldset:** Add a border around data.
- ◆ **Label:** Insert label tags in your form. When you add this option, the Split workspace appears. In the Code section of the workspace, enter the name for the label between the tags.
- ◆ **Spry Validation Text Field:** Create a text field that's validated to make sure the correct format is entered. For example, if you choose E-Mail as the validation method, the user is prompted if she forgets to enter an @ character followed by a domain address with a period.
- ◆ **Spry Validation Text Area:** Create a text field that's validated. For example, you can specify the maximum number of characters, and the response will not be valid if the user exceeds the maximum allowable number of characters.
- ◆ **Spry Validation Check Box:** Create a check box or check box group that's validated. For example, if a response to the check box query is required, the response is invalid if the user leaves the check box or check box group blank.
- ◆ **Spry Validation Select:** Create a drop-down menu that's validated. For example, if you create a drop-down menu with several groups of objects that are separated by divider lines, an invalid response would be selecting the divider line.
- ◆ **Spry Validation Password:** Create a password field that's validated. For example, you can create a password field that's validated for the number and type of characters that can be entered. An invalid response would be too many or too few characters, or the incorrect type of character.
- ◆ **Spry Validation Confirm:** Create a text field or password field that's validated. For example, if you have two password fields that require the user to submit the same response, the result is invalid if the user enters two different passwords.
- ◆ **Spry Validation Radio Group:** Create a radio button group that's validated. For example, if you create a radio group that is a required response, an error message appears if the user tries to submit the form without choosing an option from the radio group.

Event Handlers and JavaScript

When you add JavaScript to the body of a document, you need to specify when the code is executed. For example, you've probably visited more than one Web site where one of those annoying pop-ups appeared before all the page content loaded. The designer was responsible for that piece of magic by using the `onload` event handler. An *event handler* tells the Web browser when to execute (*handle*) the code.

But there's more than just knowing which event handler to use when creating a script. You have to know which browser your client's intended audience is using. That's right — the browser wars come into play because different browsers have different event handlers. When you work in Dreamweaver and use Behaviors, you can choose the browser used by the intended audience from the Show Events For drop-down menu.

Listing 1-8 shows code used to open a Web page in another browser window. The code refers to the JavaScript function "MM_openBrWindow" that's in the head of the document. The onLoad event handler instructs the browser to execute the function when the page loads.

Listing 1-8: You Will Open a New Window onLoad

```
<body onLoad="MM_openBrWindow('http://www.dasdesigns.net/  
blog', '', 'scrollbars=yes,width=640,height=480') ">
```

The onLoad event handler has other purposes. For example, the event handler is used to load all images used in the page as soon as the page loads. This makes it possible for images that appear when a user pauses his cursor over an image to appear immediately because they've already been loaded into the host computer's memory. Here's a list, with descriptions, of a few more event handlers:

- ◆ **onBlur** — Code executes when a text field loses focus. A text field *gains focus* when a user clicks inside the text box, and it *loses focus* when the user clicks outside the text box. A common use for this event handler is to display a message if a user clicks inside a text box and then outside the text box without entering data.
- ◆ **onClick** — Code executes when a user clicks the left mouse button on an object. For example, you can write code to redirect a user to a Thank You page when the user clicks a form's Submit button.
- ◆ **onDbClick** — Code executes when a user double-clicks the left mouse button on an object.
- ◆ **onError** — Code executes when a JavaScript error occurs. Developers often use this event handler with code that displays an error dialog box or suppresses all JavaScript errors in the document.
- ◆ **onFocus** — Code executes when a text field *gains focus* (when a user clicks inside the text box). Code with this event handler can be used to display a dialog box with instructions on what information to enter into a form when the first text field in the form gains focus.
- ◆ **onKeyDown** — Code executes when a user presses a key.
- ◆ **onKeyPress** — Code executes when a user presses and holds a key.
- ◆ **onKeyUp** — Code executes when a user presses and then releases a key.

- ◆ `onMouseDown` — Code executes on the down stroke of a left mouse button click.
- ◆ `onMouseMove` — Code executes whenever the user moves his mouse.
- ◆ `onMouseOut` — Code executes when a user moves his mouse over and then out of a specified object on the page. In other words, the object gained and then lost focus.
- ◆ `onMouseOver` — Code executes when a user pauses his cursor over active text or an object.
- ◆ `OnMouseUp` — Code executes when the user releases the left mouse button.
- ◆ `onUnload` — Code executes when a user leaves a page. If you've ever seen a window pop up when you navigate to another Web page, you've been a victim of a script with an `onUnload` event handler. These are almost as annoying as pop-up windows that occur when a page loads.

Making Dynamic Pages with Server Side Technology

Every single section in this chapter — with the exception of this one — concerns code that's interpreted by the client, who is the visitor to the wonderful page you created for your wonderful client, which may sound redundant, but the client in this case is your customer. However, some Web technology requires server side technology to decipher your code. We describe two prime examples of pages that require server side technology to decipher code in the list that follows.

- ◆ **ASP (Active Server Pages):** ASP is the Microsoft server side technology for generating Web page content on the fly. When you create a Web page with ASP code, you can simplify the process by using built-in objects. ASP has six built-in objects:
 - *Application*
 - *ASPError*
 - *Request*
 - *Response*
 - *Server*
 - *Session*

ASP pages are generated on the fly, using software on the server. The page delivers information based on how the user interacts with the page. ASP code is often used to deliver information from a database. For example, you can design a Web page to display contact information from

a database of members. The page visitor can then search for members in a certain ZIP code, and the ASP code causes the page to display members in that ZIP code.

The default script language for ASP is VBScript. ASP is designed to run on a Windows server. However, other servers might have software that will interpret ASP pages. When in doubt, check with your Web hosting service.

To create ASP pages, you should read a book devoted to the subject. Chapter 2 of this minibook, however, introduces you to the kiddie side of the ASP pool.

- ◆ **PHP (PHP HyperText PreProcessor):** PHP is an alternative for ASP pages. Most Web hosting services have the software to interpret PHP code. Blog applications rely heavily on PHP code. PHP runs on different platforms and is compatible with most Web servers. PHP supports many databases. In fact, most blog applications use a combination of PHP with a MySQL database. PHP syntax is similar to the Perl and C programming languages.

Dipping your toe into the shallow end of the PHP pool requires only that you read Chapter 3 of this minibook.

Chapter 2: An Introduction to ASP

In This Chapter

- ✓ Defining ASP
- ✓ Running ASP on your computer
- ✓ Installing IIS
- ✓ Introducing ASP scripting
- ✓ Rotating images and ad banners

Writing code is not a Web designer's favorite task. However, at times, you have to buck up and actually dive into the deep end of the pool to satisfy a client. *ASP* (Active Server Pages) is code that is parsed by the server; therefore, it's known as *server side technology*. ASP pages can change on the fly to suit the need of the site visitor.

One prime example of ASP pages is plucking information from a database and then serving it up to the client (the Web site visitor). The client submits the request to the server, which can be in the form of a query. The server responds to the query, writes the page, and sends it back to the client. With a fast server, the transaction is seamless. Although this chapter is by no means a substitute for the heady subject of creating ASP pages, it shows you enough to know whether you want to venture further or put a clove of garlic on a piece of string and wave it in front of any client who says, "Oh, by the way, do you do ASP?"

Understanding ASP

ASP is the brainchild of Microsoft, the same people who brought you Windows Me. Okay, Windows Me was a bust, but ASP isn't. ASP can do many things that HTML could never even dream of. For example, you can create code in an ASP page that will retrieve the date and the server's local time. If you're going to create ASP pages for a client, you must run them on a Windows server. You can create ASP pages in Dreamweaver or any good HTML editing application.

An ASP page shares many attributes with an HTML page. The page consists of tags, text, and images. Where ASP pages differ is when interactivity is needed. To obtain the interactivity, you add *scripts* to the page. You can write scripts using the VBScript or JavaScript language. ASP pages have the `.asp` extension.

When a visitor accesses an ASP page, the server hands off the page to *IIS* (Internet Information Services), an application that reads each line of the ASP page and returns the results as an HTML page to the visitor's browser. Using ASP pages has several advantages, including the following:

- ◆ You can modify or dynamically create additional content for a Web page.
- ◆ You can answer visitor queries or respond to data submitted from HTML forms.
- ◆ You can get information from server data or database information and return the results to the visitor's browser.
- ◆ You can personalize a Web page to make it more useful for individual visitors.
- ◆ ASP code is easier to learn and it executes faster than CGI and Perl.
- ◆ Your code is more secure because it can't be viewed as source in the visitor's browser.
- ◆ Your ASP pages can be viewed in any browser because they're returned as garden variety HTML.

Creating ASP pages is made easier thanks to seven built-in IIS objects. The objects are grouped based on the functions they perform. The IIS objects used within ASP pages are

- ◆ **ASPError**: ASP objects that return detailed information when an error in your ASP script occurs
- ◆ **ObjectContext**: ASP objects that manage ASP transactions that are controlled by Microsoft Transaction Server (MTS)
- ◆ **Request**: ASP objects that obtain data from the user
- ◆ **Response**: ASP objects that send information from the server to the visitor
- ◆ **Server**: ASP objects that control IIS
- ◆ **Session**: ASP objects that store data about, and change settings based on, information obtained during the user's current Web-server session
- ◆ **Application**: ASP objects that share application-level information and control settings for the lifetime of the application

You might be familiar with objects if you've ever delved into an object-oriented programming (OOP) language such as ActionScript or JavaScript. Each object can perform a set of functions, and each object has methods and properties. For example, the `Response` object has a method known as `Write` that displays information for the user.

Creating ASP Pages and Testing Them on Your Local System

It might seem like a case of the blatantly obvious, but ASP pages have the .asp extension. You can write ASP pages in a word-processing application, such as Windows Notepad. However, most designers prefer the safety net of an HTML editing application, such as Dreamweaver. In Dreamweaver, you don't have to write all the code, and the application can help you in some instances. You can write the pages with Dreamweaver, but you need another application such as IIS (Internet Information Server) installed on your system to test the pages.

Working with IIS

ASP pages need IIS to process them and return them as HTML pages. To test ASP documents on your local machine, you need to install IIS, which functions as a Web server. Installation instructions for Windows XP and Vista follow.

IIS is included with Windows XP Professional, and you can install it as follows:

- 1. Insert the Windows XP Professional CD into your CD-ROM drive.**
- 2. Choose Start⇨Settings⇨Control Panel.**
- 3. In the Control Panel window, click Add/Remove Programs.**
- 4. In the Add/Remove window, click Add/Remove Windows Components.**

The Windows Components Wizard appears. (See Figure 2-1.)



Figure 2-1:
Installing
IIS.

- 5. Select the Internet Information Services (IIS) check box and then click Next.**

Windows installs IIS on your system. This process takes a few minutes, but Windows entertains you with a progress bar while the application is installed. When the installation is complete, Windows displays a message telling you that the component has successfully installed.

- 6. Click Finish.**

IIS is installed on your system.

To install Windows IIS 7.0 with the Windows Vista operating system:

- 1. Choose Start⇨Control Panel.**

The Windows Control Panel dialog box appears.

- 2. Double-click Programs and Features.**

The Programs and Features section of the Control Panel appears.

- 3. Choose Turn Windows Features On and Off from the Tasks menu.**

The User Account Control dialog box appears.

- 4. Click Continue.**

The Windows Features dialog box appears.

- 5. Select the Internet Information Services check box and then click OK.**

IIS 7.0 is activated on your system.

Setting up the Web site

After successfully installing IIS on your computer, you're ready to start creating ASP pages. You can't create the pages in just any folder on your computer, though. You have to set them up in a special folder, which is the equivalent of a Web server. To set up your Web site, follow these steps:

- 1. Navigate to C:\Inetpub\wwwroot.**
- 2. Right-click and choose New⇨Folder from the shortcut menu.**

Windows creates a new folder.

- 3. Name the folder.**

Defining the site in Dreamweaver

After you create the folder, you're almost ready to start creating pages in Dreamweaver. But, as we mention previously, ASP pages don't work without a local server. Before you can create a little ASP magic in Dreamweaver, tell the application where the local server is. You do that when you define the site as follows:

1. Choose Site↔New Site.

The Site Definition dialog box appears.

2. Enter the name of your site.

3. Enter the URL of the site.

So you might be thinking, “The site is on my local machine; how do I enter a URL?” The answer is that when a site is on a local machine, the machine is the host server: hence, the name `localhost`. The URL for a site named `mysite` in the `wwwroot` folder is `http://localhost/mysite/`. (See Figure 2-2.)

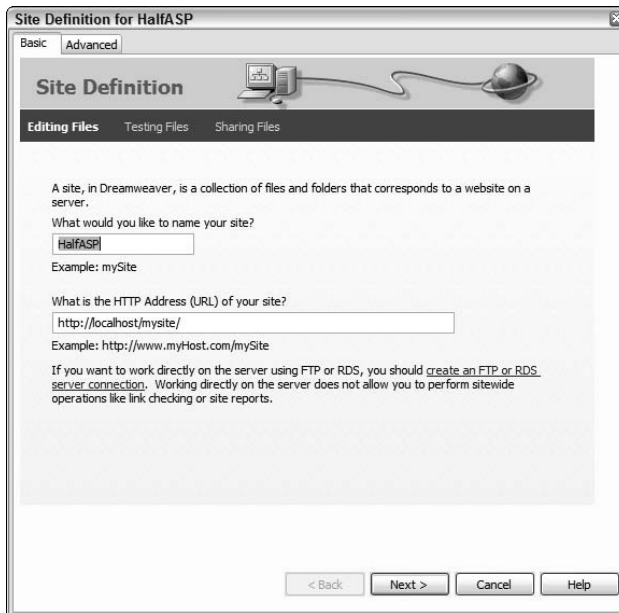


Figure 2-2:
But my sites are always well defined.

4. Click Next.

The Site Definition dialog box refreshes, and you have the option to specify a testing server.

5. Select the Yes, I Want to Use a Server Technology radio button.

The Site Definition dialog box does its chameleon thing and adopts a new guise.

6. Choose an option from the server technology drop-down menu.

Choose the applicable ASP server technology. (See Figure 2-3.) The upcoming examples will be VBScript. If your code will be VBScript, your choice would be ASP VBScript.

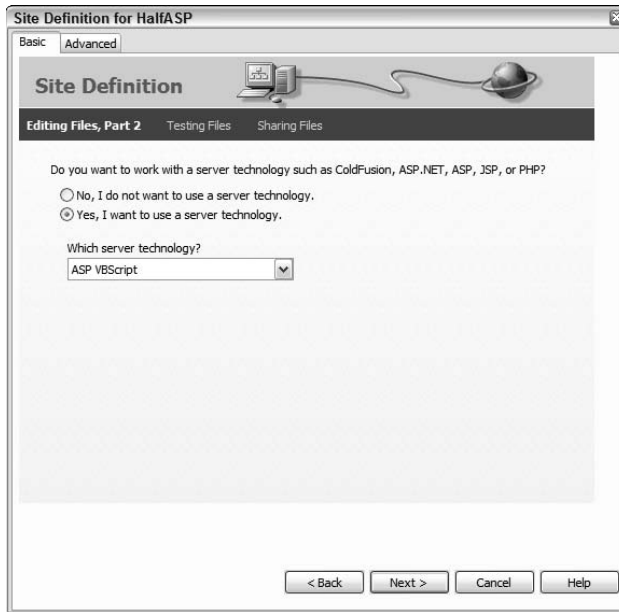


Figure 2-3:
Choose
a testing
server.

7. Click Next.

The Site Definition dialog box dons another disguise; this time, you get the option of choosing how you will test your files. (See Figure 2-4.)

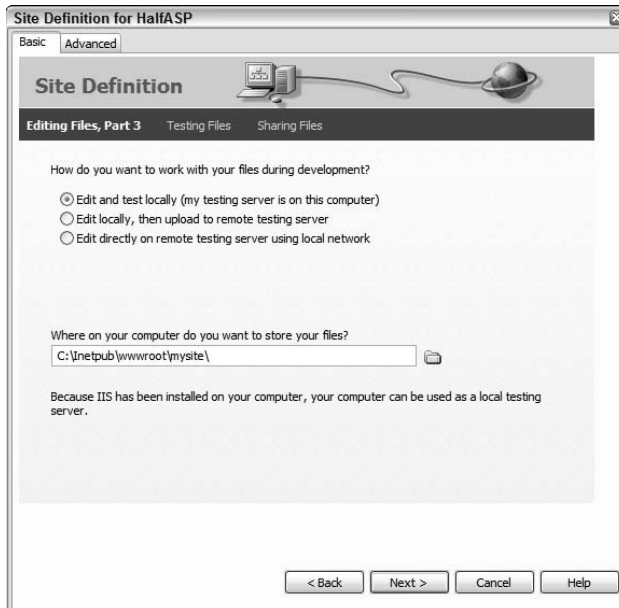


Figure 2-4:
Tell Dream-
weaver
where
you're going
to put your
files to
the test.

8. Accept the default option to test on your local machine. Verify that the folder where you store your files is correct.

9. Click Next.

The dialog box reconfigures, giving you the option to specify the root URL of your site. This should already be filled in. (See Figure 2-5.)

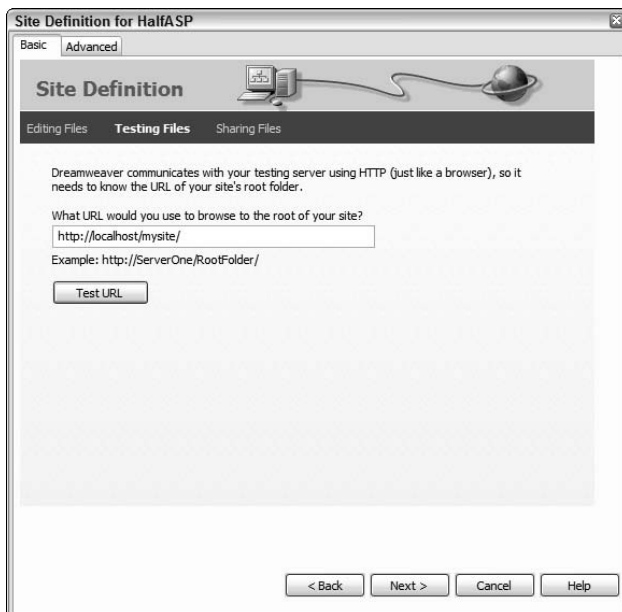


Figure 2-5:
Specify the
root URL of
your site.

10. Click the Test URL button.

Dreamweaver runs a test to ensure that the application can test ASP files on the specified folder. If the test is successful, Dreamweaver displays a dialog box informing you that the test is successful. In the rare instance that the test is unsuccessful, the URL in Step 9 is incorrect.

11. Click Next.

The dialog box reconfigures and gives you the option of copying the files to another machine when you're finished editing.

12. Accept the default option of No and then click Next.

The Site Definition dialog box reconfigures and shows a summary of the setup. If anything is incorrect, click the Back button to navigate to the page that contains the parameters you need to change.

13. Click Done.

The dialog box closes, and you're ready to venture into the wild and wonderful world of ASP.



If you ever test an ASP page and you don't get the results you're expecting, make sure that IIS is running. You can check this by opening the Administrative Tools section of the Windows Control Panel. Open the Services section and then double-click the World Wide Web Publishing icon. If the Status reads Stopped, click the Start button to enable IIS.

Creating ASP Pages in Dreamweaver

After you install IIS, set up a Web folder, and define the site in Dreamweaver, you're ready to begin creating pages. An ASP page looks a lot like HTML; it has images, text, HTML tags, and so on. Where the ASP magic comes into play is when you use ASP objects and create scripts in the page that define how the page looks after IIS reads the code. Sometimes, you have only a few lines of script; other times, you have a lot. If half of your page is script, you could say the page is half-ASP. At any rate, it's time to create an ASP page in Dreamweaver.

1. Choose File→New.

The New Document dialog box appears.

2. Click Blank Page and then choose ASP VBScript from the Page Type column.

These are the options you need when creating ASP pages with the VBScript language.

3. Click Create.

Dreamweaver creates a new document that's ready to receive your VBScript.

4. Click the Code button.

Dreamweaver switches to Code view.

5. Position your cursor after the <body> tag and then press Enter.

Dreamweaver creates a new line.

6. Enter the following code:

```
<%  
Response.Write("Hello World")  
%>
```

While you type your code, Dreamweaver displays code hints. For example, after you type the W, Dreamweaver displays `Write(string)`. Click the code hint, and Dreamweaver adds `Write` to the code. How cool is that?

7. Press F12.

Dreamweaver prompts you to save the document. After you save the document, Dreamweaver displays the page in your default browser. You should see the words `Hello World`. If you don't see this message, make sure you've created an ASP document and that the code is exactly like what's shown in Step 6. If you forget a percent sign or ending bracket, the code won't execute properly.

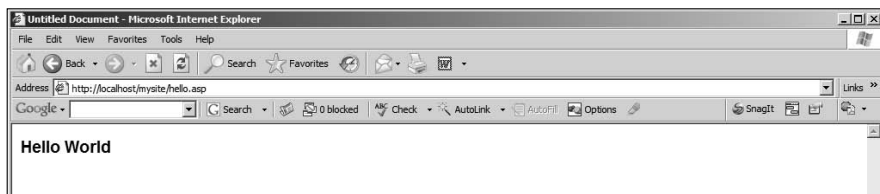
Although it doesn't look very impressive, you changed the Web page based on the instructions you included between the `<%` and `%>` tags. The code uses the `Write` method of the `Response` object to display the information between the opening and closing parentheses. The information is enclosed with quotation marks, which designates that it's a text string. In fact, you might have noticed Dreamweaver displays a tooltip that displays `Write(string)` after you type the opening parentheses.

You can also format text by adding HTML tags. Change the code in Step 6 to the following:

```
<%
Response.Write("<h2>Hello World</h2>")
%>
```

Figure 2-6 shows the text with the `<h2>` (heading 2) tags applied.

Figure 2-6:
Format text with HTML tags.



Using Variables

Variables store information. It's kind of like giving a Web browser a brain. When you incorporate variables on your ASP pages, you can store and dispense information. Also, you can use variables to store information entered in forms and display them on an ASP page. The steps that follow show you how to create a form that collects user information and then displays it on another page. To explore the power of variables, launch Dreamweaver and then do this:

1. Create a new HTML document.

That's right: The form you use to collect the information is an HTML document.

- 2. Switch to Code view, create a new line after the <body> tag, and then enter the following:**

```
<form method="GET" action="welcome.asp">  
First Name: <input type="text" name="fname" />  
<br />  
Last Name: <input type="text" name="lname" />  
<br /><br />  
<input type="submit" value="Submit" />  
</form>
```

The code you enter creates a simple form that collects the visitor's first name and last name. The form action opens another document, "welcome.asp", when the site visitor clicks the Submit button. The form method, GET, makes the information available to other documents. The variables are fname and lname.

- 3. Save the document.**

Save the document with the filename `form.htm`. Leave the document open in Dreamweaver.

- 4. Choose File↵New.**

Dreamweaver displays the New Document dialog box.

- 5. Click Blank Page, and then choose the ASP VBScript option from the Page Type column.**

- 6. Switch to Code view, create a new line after the <body> tag, and then enter the following:**

```
Welcome  
<%  
Response.Write(Request.QueryString("fname"))  
Response.Write(" " & Request.QueryString("lname"))  
%>
```

- 7. Save the document.**

Save the document as `welcome.asp`.

- 8. Click the form.htm tab.**

Each open document in Dreamweaver has its own tab.

- 9. Press F12.**

Dreamweaver opens the document in your default browser. If you entered your code correctly, your browser should show the form shown in Figure 2-7.

- 10. Fill in the form and then click Submit.**

The form action calls the ASP page, which uses the variables from the form to greet the visitor. (See Figure 2-8.)

Figure 2-7:
Fill in your answer; submit the form.

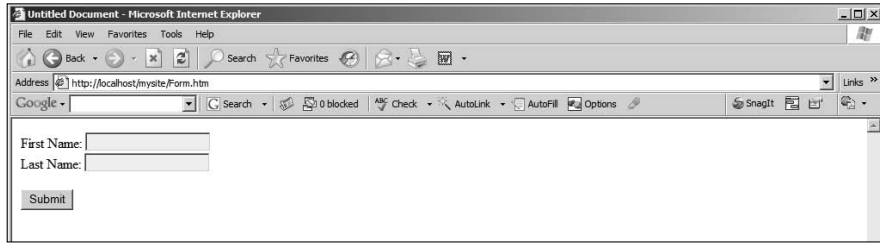
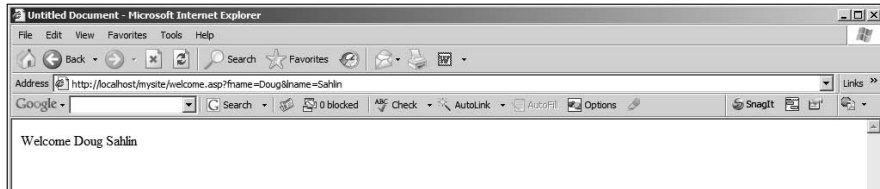


Figure 2-8:
Giving the visitor a variable welcome.



Before moving on, you need to know a bit more about the code used in the ASP page. The code is using the `Write` method of the `Response` object to greet the visitor. The code is using the `QueryString` property of the `Request` object to pluck the information from the variables `fname` and `lname`.

But there's yet another method to interact with a form, and that's the `POST` method. When you use the `POST` method, the information from the form isn't displayed in the address window of the user's browser.

To interact with form elements by using the `POST` method, follow these steps:

1. Create a new HTML document.

That's right; the form you use to collect the information is an HTML document.

2. Switch to Code view, create a new line after the `<body>` tag, and then enter the following:

```
<form method="POST" action="welcomePost.asp">
First Name: <input type="text" name="fname" />
<br />
Last Name: <input type="text" name="lname" />
<br /><br />
<input type="submit" value="Submit" />
</form>
```

The code you enter creates a simple form that collects the visitor's first name and last name. The form action opens another document, "welcomePost.asp". The form method, POST, makes the information available to other documents. The variables are `fname` and `lname`.

3. Save the document.

Save the document with the filename `formPost.htm`. Leave the document open in Dreamweaver.

4. Choose File → New.

Dreamweaver displays the New Document dialog box.

5. Click Blank Page, and then choose the ASP VBScript option from the Page Type column.

6. Switch to Code view, create a new line after the <body> tag, and then enter the following:

```
<%  
dim fname  
fname=Request.Form("fname")  
dim lname  
lname=request. Form("lname")  
%>  
Response.Write("Hello" & " " & fname & " " & lname &  
"!<br >" & "How are you?")  
%>
```

7. Save the document.

Save the document as `welcomePost.asp`.

8. Click the form.htm tab.

Each open document in Dreamweaver has its own tab.

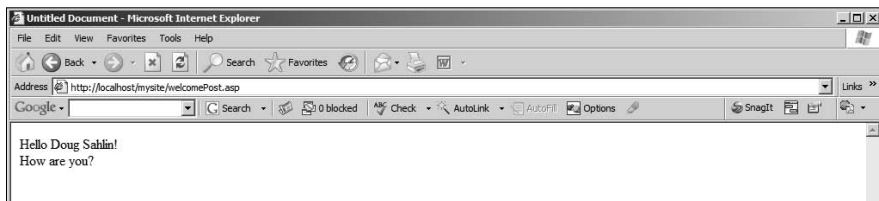
9. Press F12.

Dreamweaver opens the document in your default browser.

10. Fill in the form and then click Submit.

The form action calls the ASP page, which uses the variables from the form to greet the visitor. (See Figure 2-9.)

Figure 2-9:
Giving the
visitor a
hearty hello.



Differences between this scenario and the previous scenario include these:

- ◆ **The method in which the data is transmitted: POST.** With the post method, the information is not displayed in the address window of the user's browser.
- ◆ **The manner in which the data is transmitted to the ASP page:** The code `dim fname` declares a new variable. The next line of code sets the value of the variable equal to the data "fname" that was posted from the form. The `form` collection of the `Request` object is used to get the information.
- ◆ **The manner in which the information is written:** Instead of creating several lines of code using the `Write` method of the response object, it's compiled on one line of code. The variables and *text strings* (the text surrounded by quotation marks) are *concatenated* (string-speak for *combined to form a single phrase*). The HTML tag `
`, which is used to create a line break, is also in quotes. Notice the two spaces surrounded by quotation marks. If these were not in the script, the greeting, first name, and last name would run together.

Introducing VBScript functions

VBScript has some built-in functions that enable you to perform certain tasks. The functions are divided into groups, such as Math Functions, Format Functions, String Functions, and so on.

To demonstrate the power of functions, this section covers using the date/time functions to display the date and time on the server's machine. To add the date and time to an ASP page, complete the following steps:

1. Open an ASP document to which you want to add the date and time.

Position your cursor where you want to display the date and time.

2. Switch to Code view.

3. Enter the following code:

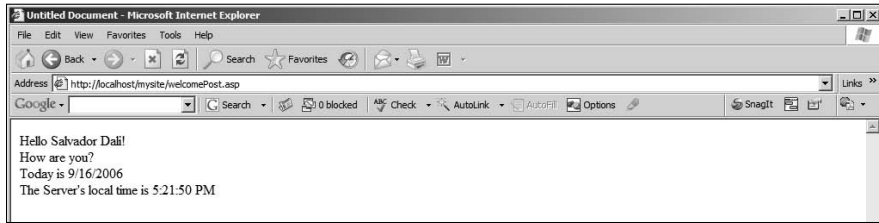
```
<%
Response.Write("Today is" & " " & (Date()) & "<br />" &
"The Server's local time is" & " " & (Time ()))
%>
```

4. Press F12.

Dreamweaver prompts you to save the page. After you save the page, it's displayed in your default browser. Figure 2-10 shows the date and time added to an ASP page.

When you use the VBScript `Date` function on an ASP page, it gets the date from the server's computer, not the user's computer. However, this is still useful information, especially if your client has a Web site in a different time zone than that of the visitors. You can incorporate the date and time in the greeting; for example, The date and time in London, England is. . . .

Figure 2-10:
ASP can
tell time.



You also have formatting options when you add the `Date` function to a script. The default formatting displays the short version of the date showing the month, day, and year in this format: `mm/dd/yyyy`. Table 2-1 shows the formatting you can use for the `Date` function.

Table 2-1 **Formatting the Date and Time**

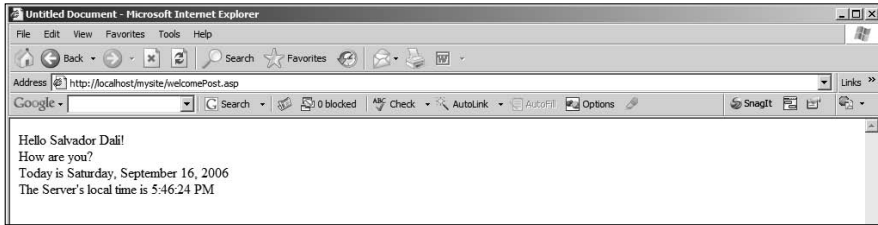
<i>Constant</i>	<i>Value</i>	<i>Description</i>
<code>vbGeneralDate</code>	0	Display a date in format: <code>mm/dd/yy</code> . If the date parameter is <code>Now()</code> , it also returns the time, after the date.
<code>vbLongDate</code>	1	Display a date using the long date format: weekday, month day, year.
<code>vbShortDate</code>	2	Display a date using the short date format, like the default (<code>mm/dd/yy</code>).
<code>vbLongTime</code>	3	Display a time using the time format: <code>hh:mm:ss PM/AM</code> .
<code>vbShortTime</code>	4	Display a time using the 24-hour format: <code>hh:mm</code> .

By adding either a constant that you previously defined and given a value or the value, you can format the date and time. For example, change the code used in the previous example to

```
<%
Response.Write("Today is" & " " & (FormatDateTime (Date(),1))
    & "<br />" & "The Server's local time is" & " " & (Time
    ()))
%>
```

Figure 2-11 shows the result. The day of the week and month are spelled out.

Figure 2-11:
The day and month displayed in the long date format.



Discovering Cool ASP Tricks

If you've read this chapter from the start, you know that you can do a lot of things with ASP. The previous sections show you how to do some useful things with ASP; now it's time to take it to the next level. The upcoming sections show you how to add some cool features to your designs using ASP.

Displaying a random image

If you have a client (say a photographer) who wants to display a different image each time a page loads, read on. VBScript can generate random numbers. ASP pages are generated when the browser loads the page. Combine the two, and you can easily display a different image every time the page loads. To create a page that displays a random image each time the page loads, complete the following steps:

- 1. Rename the images you want to display randomly, adding a sequence of number suffixes, starting with 1.**

We recommend using Adobe Bridge to rename the photos to `IMG_` followed by a number. You end up with a folder of images named `IMG_1.jpg`, `IMG_2.jpg`, and so on.

- 2. Open the ASP document in which you want to display a random image each time the page loads.**
- 3. Position your cursor where you want the image to appear.**

To constrain the position of the image, put the VB code in a table row.

- 4. Enter the following code:**

```
<%
RANDOMIZE
Dim RndNum
RndNum = Int(30 * Rnd) + 1
%>

```

That's not a lot of code, but what it achieves is powerful:

- The first line of code, `<%`, tells IIS that this is a script.

- The first statement, `RANDOMIZE`, generates the `Rnd` function's capability to create a random number.
- The third line of code declares a variable called `RndNum`, which is the shortcut for random number.
- The fourth line of code sets the value of the variable. `Int` is a function that generates an *integer*, which you might remember is a whole number.
- The computation in the parentheses takes the number of images, which in this case is 30, and multiplies that by the `Rnd` function. This generates an integer between 0 and 29; therefore, a value of 1 is added, so the range is 1 to 30.
- The fifth line of code, `%>`, ends the VBScript.
- The `` tag has the VBScript code `<% =RndNum %>`, which appends `IMG_` with the randomly generated number. If your client has a fairly large number of images, say ten or more, chances are his visitors will see a different image each time they visit the site.

5. Press F12.

Dreamweaver prompts you to save the document. After you save the document, the page launches in your default browser. Click the Refresh button a few times, and you'll see the script do its magic. (See Figure 2-12.)



Figure 2-12:
Generate a random image on the home page.

Create a rotating ad banner

If you have a client who wants to generate extra revenue, you can create ASP code to generate an ad banner every time a page is visited. The ad banner, when clicked (of course), redirects the visitor to another site.

To add a rotating ad banner to a Web page, complete these steps:

- 1. Open the ASP document to which you want to add the ad banner.**
- 2. Add a table where you want the ad banner to appear.**

We recommend putting the ad banner at the top of the document. If you're adding the ad banner to an existing ASP page, position your cursor at the top of the document and then choose Insert→Table. The table should be 1 row and 1 column.

- 3. Position your cursor inside the table and then switch to Code view.**
- 4. Enter the following code:**

```
<%
set adrotator=Server.CreateObject("MSWC.AdRotator")
adrotator.Border="1"
Response.Write(adrotator.GetAdvertisement("ads.txt"))
%>
```

The first line of code tells IIS to compile a script. The second line of code creates the `AdRotator` object. The third line of code puts a 1-pixel border around the ad. The fourth line of code uses the `Write` method of the `Response` object to read an advertisement from a text file. The fifth line of code ends the script.

- 5. Create a new document in Notepad (Windows) or TextEdit (Mac).**

This is the text file that contains three things: the path to the images, the URL, and the frequency with which the ad should be displayed.

- 6. Enter the information for your ads. The following is a sample:**

```
REDIRECT banners.asp
*
banners/adp.gif
http://www.accessdigitalphotography.com/
Visit Access Digital Photography
50
banners/si.gif
http://www.superbimages.net
Visit Superb Images
30
banners/pL.gif
http://www.pixelicious.info
Pixelicious - The Digital Photography Podcast
20
```

The first line in the document specifies the document that contains the code to redirect the visitor to the ad site URL. The asterisk tells IIS that the ads follow. The three ads in this document will be rotated by the `AdRotator` component.

Here's a rundown of what each line in the ads designates:

- *Path*: The first line of each ad is the path to the image file. In this case, all of the image files are in a folder called `Banners`.
- *URL*: The second line of each ad is the URL to which the visitor will be redirected.
- *ALT text*: The third line of each ad is the ALT text that will be generated for the image.
- *Frequency*: The fourth line of each ad is the frequency in which it will appear. The first ad will appear 50 percent of the time, the second ad 30 percent, and the third ad 20 percent.

7. Save the document as `ads.txt`.

Save the document in the root folder of your Web site.

8. In Dreamweaver, create a new ASP document.

Choose the ASP VBScript option from the Page Type column of the New Document dialog box.

9. Switch to Code view.

10. Create a new line after the `<body>` tag and enter the following code:

```
<%  
    url=Request.QueryString("url")  
    If url<>"" then Response.Redirect(url)  
%>
```

In a nutshell, the script reads the URL associated with the image from the `ads.txt` document and redirects the visitor to the page.

11. Save the document as `banners.asp`.

You also need to save this file in the root folder of your Web site.

12. In Dreamweaver, click the tab for the document in which you inserted the `AdRotator` component.

13. Press F12.

Dreamweaver prompts you to save the document. After you save the document, Dreamweaver opens it in your default browser and displays a banner ad. (See Figure 2-13.) Refresh the browser a few times to see the ads rotate.

14. Click an ad.

The `banners.asp` page is summoned, the script runs, and the visitor is redirected to the URL associated with the ad. (See Figure 2-14.)



Figure 2-13: Banner ads, love 'em or ignore 'em.

Book VI
Chapter 2

An Introduction
to ASP

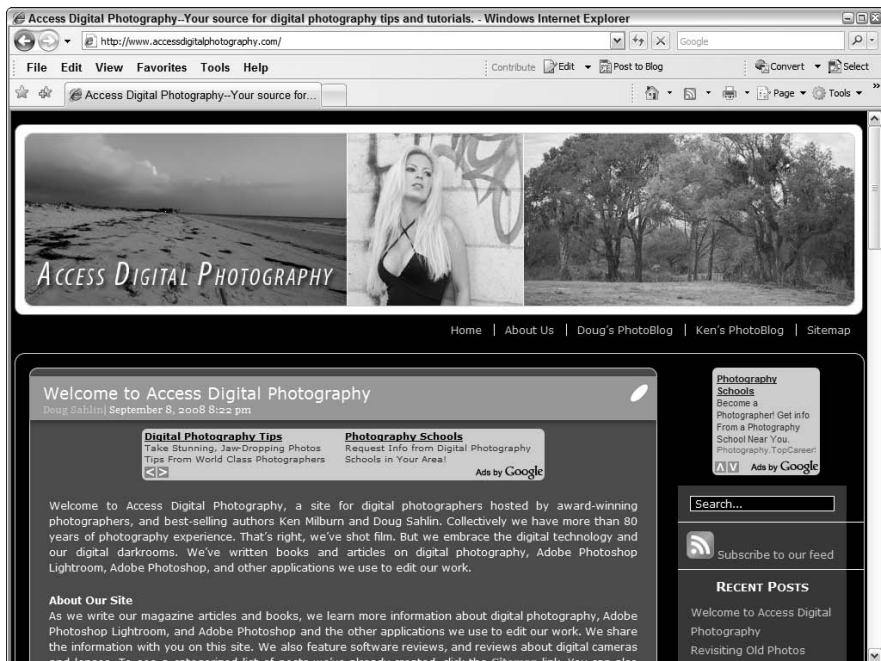


Figure 2-14: The adbanner .asp page redirects the visitor to the URL associated with the ad.

Chapter 3: Introduction to PHP

In This Chapter

- ✓ Understanding PHP
- ✓ Discovering PHP server side requirements and PHP syntax
- ✓ Adding variables to the code
- ✓ Creating conditional statements
- ✓ Getting loopy: `while` and `for`

If your client needs an interactive Web site, PHP is yet another language you can use. Like ASP (the subject of the previous chapter), PHP scripts within PHP pages are interpreted by the server. PHP supports databases. In fact, many bulletin board and blog applications are created with PHP code. These applications use a MySQL database to store bulletin board and blog data. You can also create your own PHP pages by using an HTML editing application like Dreamweaver. If you need interactive pages that update when they're visited, read on, for the pages that follow introduce you to PHP.

Defining PHP

PHP originally stood for Personal Home Page. Now, the acronym means PHP HyperText PreProcessor. PHP pages contain HTML tags and PHP scripts to create a page that can change depending on user interactivity. The PHP page, when visited, is converted into HTML by the server. You can't view the original PHP code by viewing the source code from the browser. All you see if you do this is the HTML code generated by the server after parsing the PHP code.

PHP pages can have all the other goodies that HTML pages have, such as text, hyperlinks, and so on. Then you can kick them up a notch or two with PHP code. Many applications, such as interactive calendars, blogs, and bulletin boards, are powered by PHP. Unlike ASP, PHP is *cross-platform*, which means it can be used on servers running Unix, Windows, Macintosh, and other operating systems.

Fulfilling PHP Server Side Requirements

If you're creating PHP pages for a client's Web site, you have to choose a server that supports PHP. Originally, PHP required using an Apache server. However, many Linux servers can also parse PHP code. In the same regard, you should choose a server that supports MySQL databases. Your customer might not need a database in the beginning. However, as his site grows, he might decide to add a blog or create a mailing list from user e-mail addresses he's collected. When this occurs, having a server with MySQL database capabilities is a definite plus.

You also need to make sure that the server has the version of PHP and MySQL to support the application you intend to use or the pages you intend to create. As of this writing, the current version of PHP is 5.2.6, and the current version of MySQL is 5.0. Check with your Web server's technical staff to see which version they have installed on their system.



You can download an Apache server, PHP, and MySQL for installation on your local machine. However, if you already have an IIS server installed, these items will be in conflict. Doug creates his PHP documents on his local machine and then uploads them to his Web server for testing.

Creating PHP Pages

When you create PHP code, you must create the code with the correct syntax. Otherwise, the code won't execute properly when parsed by the server. If you've created ActionScript code from scratch, you're familiar with the need for proper syntax. All PHP code begins with `<?php` and ends with `?>`. A semicolon (;) is required at the end of each line of code. In the following steps, you can see how to create a PHP page with a simple script:

1. In Dreamweaver, choose File⇨New.

The New Document dialog box appears.

2. Click Blank Page and then choose PHP.

3. Click Create.

Dreamweaver creates a new PHP page.

4. Switch to Code view.

Your cursor is positioned after the `<body>` tag.

5. Press Enter (Windows) or Return (Mac) to create a new line.

6. Enter the following code:

```
<?php
    echo "Hello World";
?>
```

The `echo` command tells PHP to display the text between the quotation marks. Dreamweaver displays the beginning and ending PHP tags (`<?php` and `?>`) as red, boldfaced text. If desired, you can use `print` in lieu of `echo`.

7. Save the document and upload it to your server.

When you save the document, choose PHP from the Files of Type dropdown menu.

8. When you view the page in a Web browser, you see the text Hello world. (See Figure 3-1.)

Figure 3-1:
Using PHP
to say hello
to the world.



Declaring and Using Variables

Variables are a very important part of any code language. *Variables* are code objects that are used to store and dispense information. For example, you can have a variable with no content — a *null variable* — that’s filled with the content a user enters into a form. You use variables when sending information to a database or retrieving information from a database. Variables in PHP are preceded by a dollar sign (`$`). Listing 3-1 shows a simple PHP script that contains variables.

Listing 3-1: Using Variables in PHP

```
<?php
    $fname="Doug";
    $lname="Sahlin";
    print "Hello $fname $lname";
?>
```

If this were added to a PHP page, the resulting page that would be delivered to the user’s Web browser would display as shown in Figure 3-2.

Figure 3-2:
Using
variables in
a PHP page
to display
information.



Variables are much more powerful than that, though. As we mention previously, you can use them to transfer data from a form to a database, or from a form to a page. Listing 3-2 shows a simple HTML form that asks the user for first name and last name.

Listing 3-2: Retrieving Data from a Form

```
<form id="form1" name="form1" method="post" action="welcome.  
  php">  
  <table border="0" cellspacing="0" cellpadding="0">  
    <tr>  
      <td>First name </td>  
      <td><input type="text" name="fname" /></td>  
    </tr>  
    <tr>  
      <td>Last Name </td>  
      <td><input type="text" name="lname" /></td>  
    </tr>  
  </table>  
  <p>  
    <input type="submit" name="Submit" value="Submit" />  
  </p>  
</form>
```

The form has two fields, named `fname` and `lname`. The form method is `post`, which means the data is transferred transparently and isn't visible in the visitor's browser. When the visitor clicks the Submit button, the form action calls a page named `welcome.php`. The PHP code for the `welcome.php` page is shown in Listing 3-3.

Listing 3-3: PHP Code That Retrieves Variables from a Form

```
<?php  
  $fname=$_POST['fname'];  
  $lname=$_POST['lname'];  
  print "Hello $fname $lname";  
?>
```

The first line of code signifies that the script is PHP. The second line of code defines a variable called `fname` and sets it equal to the value of the `fname` form field by using the `POST` method. The third line of code defines a variable called `lname` and sets it equal to the value of the `lname` form field using the `POST` method. The fourth line of code prints the result with the text `Hello`. The fifth line of code ends the script. Figure 3-3 shows the result when the script is parsed by the server.

Figure 3-3:
Populating
variables
with data
from a form.



Working with Conditional Statements

Conditional statements are like a fork in the road. When you create a dynamic Web site with PHP, you want your Web page to be as smart as possible, with code that can make decisions based on the information visitors input, the type of browser they're using, and so on. With a conditional statement, you can achieve this. If the condition is true, one thing happens; if not, another thing happens. For example, a conditional statement can verify a user's e-mail address by seeing whether it has a valid format (*somebody@somewhere.com*). If the user didn't enter a valid e-mail address, the code you created asks the techno-phobic user to enter a valid e-mail address.

Putting conditional statements into action

The easiest conditional statement is to verify whether a condition is true. To do this, you use an `if` statement. The syntax for the statement is shown in Listing 3-4.

Listing 3-4: Proper Syntax for an `if` Statement

```
if (condition is true) {
    do this
}
more code
```

If the condition verifies as true, the code within the curly brackets is executed; otherwise, the next line of code is executed. For example, say you have a form on your Web site that asks for the visitor's first name and last name. You can use an `if` statement to validate the form. Listing 3-5 shows the HTML code for the form.

Listing 3-5: A User Information Form

```
<form id="form1" name="form1" method="post" action="validate.php">
  <table border="0" cellspacing="0" cellpadding="0">
    <tr>
      <td>First name </td>
      <td><input type="text" name="fname" /></td>
    </tr>
    <tr>
      <td>Last Name </td>
      <td><input type="text" name="lname" /></td>
    </tr>
  </table>
  <p>
    <input type="submit" name="Submit" value="Submit" />
  </p>
</form>
```

The form is pretty straightforward, asking for the user's information. As soon as the user clicks the Submit button, the `validate.php` page loads. This page contains the code shown in Listing 3-6, which tests whether the user entered her first name and last name.

Listing 3-6: Validating the Form in PHP

```
<?php
    $fname=$_POST['fname'];
    $lname=$_POST['lname'];
    if ($fname==" " Or $lname==""){
        print '<meta HTTP-EQUIV="refresh" content=0,url="form.htm">';
    }
?>
```

The variable `$fname` is set equal to the information entered in the form field `fname`, and the variable `$lname` is set equal to the information entered in the `lname` form field. The information is retrieved by using the `POST` method. The conditional statement uses the logical operator `"Or"`. The comparison operator (`==`) is used to see whether either variable contains no information.

Quotation marks with nothing between them indicate that the variable is null. If either variable contains no information, the next line of code refreshes the browser to the form. However, if information is in both fields, you need additional code; otherwise, the user is stuck on the `validate.php` page.

Enter the `else` statement. Listing 3-7 shows the PHP code with the addition of an `else` statement.

Listing 3-7: Creating a Fork in the Road

```

<?php
    $fname=$_POST['fname'];
    $lname=$_POST['lname'];
    if ($fname==" " Or $lname==""){
        print '<meta HTTP-EQUIV="refresh" content=0,url="form.htm">';
    } else {
        print '<meta HTTP-EQUIV="refresh" content=0,url="index.php">';
    }
?>

```

The addition of the `else` statement creates another course of action when both variables do contain data. A third statement can be added to evaluate a condition: `elseif`. This statement enables you to add an alternative outcome if the `elseif` statement evaluates to be true. You can add as many `elseif` statements to cover any possible scenario. Listing 3-8 shows an example in PHP of an `elseif` statement added to evaluate another possible outcome.

Listing 3-8: Using an elseif Statement

```

<?php
    $password=$_POST['password'];
    $fname=$_POST['fname'];
    $lname=$_POST['lname'];
    if ($password==" " Or $fname==" " Or $lname==""){
        print '<meta HTTP-EQUIV="refresh" content=0,url="form.htm">';
    } elseif ($password!= "letmein") {
        print '<meta HTTP-EQUIV="refresh" content=0,url="http://www.google.com">';
    } else {
        print "Welcome $fname $lname ";
    }
?>

```

This scenario evaluates a login form for a Web site. The code creates three variables, retrieving the first name, last name, and the password submitted by the user. The `if` statement checks to be sure the user entered data in each form field. The `elseif` statement evaluates whether the password does not equal `letmein`. If the `elseif` statement evaluates as true, the visitor is redirected to Google. The final scenario is what happens when the previous statements evaluate as false and welcome the visitor to the Web site.

Using comparison operators

Comparison operators test for equality, inequality, or whether a value is greater than or less than another value. You use comparison operators in conditional statements to compare elements, such as whether a variable contains certain information. Comparison operators are often confused with assignment operators. For example, when you create a variable, you use the

assignment operator =. When you create a conditional statement and want to verify whether the content of a variable is equal to a specific value, you use the comparison operator ==. Table 3-1 shows the comparison operators you can use when creating conditional statements.

Table 3-1		Comparison Operators	
<i>Operator</i>	<i>Example</i>	<i>Operator Name</i>	<i>Result</i>
==	\$a==\$b	Equal	True if the value of \$a equals the value of \$b
!=	\$a!=B	Not equal	True if the value of \$a does not equal the value of \$b

Using logical operators

You use *logical operators* when you want to evaluate multiple conditions. When you evaluate multiple conditions, you can specify whether a statement is true if all conditions are met or if only one condition is met. Table 3-2 shows the logical operators you can use when creating conditional statements.

Table 3-2		Logical Operators	
<i>Operator</i>	<i>Example</i>		<i>Result</i>
And	\$a=="Jack" And \$b=="Jill"		True if both sides of the statement evaluate as true
Or	\$a=="Jack" Or \$b=="Jill"		True if either side of the statement evaluates as true
Xor	\$a=="Jack" Xor \$b=="Jill"		True if either side of the statement evaluates as true, but false if both sides of the statement evaluate as true

Repeating Lines of Code, Using Loops

Loops are very handy when you need to repeat certain lines of code for a specified number of times. You can loop through a database while searching for certain data. A loop continues as long as a given condition is true. There are different types of loops you can add to your PHP code. We show you how to get loop in the upcoming sections.

Using a while loop

A `while` loop is a loop in its simplest form. The loop executes while a given condition is true. As soon as the condition is false, the next lines of code are executed. Listing 3-9 shows an example of a `while` loop.

Listing 3-9: Using a while Loop

```
<?php
    $MyNum = 1;
    while ($MyNum <= 10)
    {
        print ("MyNum");
        $MyNum++;
    }
?>
```

The code in Listing 3-9 will print the numbers 1 through 10. The loop continues as long as the value of the `$MyNum` variable is less than or equal to 10. The line of code that reads `$MyNum++`; increases the value of `$MyNum` by a value of 1 every time the loop executes.

Using a for loop

A `for` loop executes code for a given set of iterations. The value by which the loop increments can vary. For example, you can use a `for` loop to search every record in a database or every other record in a database. Listing 3-10 shows a `for` loop.

Listing 3-10: Using a for Loop

```
<?php
    for ($MyNum = 1; $MyNum <= 10; $MyNum+=2) {
        print $MyNum;
    }
?>
```

This loop also executes as long as the value of `$MyNum` is less than or equal to 10. The `for` loop has three statements that are separated by semicolons. The first statement in the loop sets the value of `$MyNum` to 1; the second statement tells the PHP compiler to execute the loop as long as the value of `$MyNum` is less than or equal to 10; and the third statement specifies the loop to increment the value of `$MyNum` by a value of 2. The code will print the following: 13579.

Generating a Random Image

The previous sections give you a look at PHP code. If you read through them, you might now be interested in seeing a real-world application using PHP code. You can find plenty of applications on the Web. For that matter, you can even find examples of code on the Web. This chapter is but a brief introduction showing you some of the building blocks of PHP code. If your interest is piqued, consider picking up a copy of *PHP 5 For Dummies*, by Janet Valade (Wiley). And now, on to the real-world example of PHP in action.

When people visit a Web site, they like to see new and different things. One way to ensure that they do is to generate a random image on the home page. If you have ten or more images that are generated randomly, the visitor is likely to see a different image every time he visits the site. To generate a random image on a home page, follow these steps:

- 1. Rename the images and add a number suffix, starting with 1.**

We recommend using Adobe Bridge to rename the images to `IMG_`, followed by a number. You end up with a folder of images named `IMG_1.jpg`, `IMG_2.jpg`, and so on.

- 2. Open the PHP document in which you want to display a random image each time the page loads.**

- 3. Position your cursor where you want the image to appear.**

To constrain the position of the image, put the PHP code in a table row.

- 4. Enter the following code:**

```
<?php
$image = rand(1,30);
print '';
?>
```

You get a lot of bang for your buck with just a few lines of code. The first line tells the server it is parsing PHP code. The second line of code creates a variable named `$image` and sets it equal to a random number between 1 and 30 (the number of images renumbered to display randomly). The third line of code uses an HTML image tag (``) to add the random image to the page. The `$image` variable is used to append the number to the `image` filename.

- 5. Upload the file to your server and preview it in your Web browser.**

The script causes an image to be displayed randomly. (See Figure 3-4.) Refresh the browser a few times to test the script.



Figure 3-4:
Is it a bird?
Is it a plane?
No, it's a
random
image
generated
by PHP
code.

Chapter 4: MySQL and PHP

In This Chapter

- ✓ **Building a mailing list form**
- ✓ **Creating a MySQL database**
- ✓ **Transferring data to the database**
- ✓ **Reading data from the database**

Mysql and PHP are almost like a marriage made in heaven. PHP likes to send stuff to MySQL, and MySQL accepts it with open arms, er, fields. Like Batman and Robin, MySQL and PHP are a dynamic duo. So dynamic, in fact, that quite a few applications out there use the combination to good effect. Blogs, bulletin boards, and calendar applications are good examples of the dynamic duo at work. You can write a PHP script in a PHP document that sends information to and plucks data from a MySQL database. MySQL databases keep the information safely locked away until its good buddy PHP comes calling.

In case you rushed right to this chapter expecting to find out everything you need to know about MySQL and PHP, well, you almost came to the right place. To learn everything about integrating MySQL databases with PHP pages would require a bigger book. However, in this chapter, on these very pages written with tender loving care, with blisters on your authors' fingers, we show you how to use PHP and MySQL to create a mailing list from visitors who submit a form with their contact information.

You can choose from many applications for managing mailing lists. The applications enable you to easily manage a mailing list, send information to people who have opted in to the mailing list, and so on. However, if you have a client who doesn't need all the bells and whistles, why give them to him? If your client's only need is to store a limited number of names and e-mail addresses in a database and occasionally retrieve the information, you can easily accomplish this by creating a form for your client's Web site visitors, a MySQL database on your client's Web server, and then the PHP code to send and retrieve information from the database.

Creating a Mailing List Form

The first step in the process of creating a mailing list is setting up a mailing list form. In case you haven't read the chapter on creating forms, the following steps lead you through the process. It's easy. Really. The results from the form are what populates the database.

To create a mailing list form:



- 1. In your favorite HTML editor (ours is Dreamweaver), open the document into which you want to insert a mailing list form.**

Make sure you're working in Design view.

Some designers have the audacity to create a pop-up window on the home page, asking the visitor to sign up for a mailing list. Really, how can you tell if you want to sign up for a mailing list when you haven't even seen the site yet? We'll jump off the soapbox now, leaving the absolute location to you and your client.

- 2. Position your cursor where you want to insert the form.**
- 3. Choose Insert⇨Form⇨Form.**

Dreamweaver inserts the first building block for a form.

- 4. Choose Window⇨Properties.**

The Property inspector opens.

- 5. In the Action field, enter `addrecord.php`.**

This is the document you create to add the information to the database. The action tells the Web browser to launch the page when the visitor clicks the Submit button.

- 6. Position your cursor within the form (the red border) and then choose Insert⇨Table.**

The Table dialog box appears. Using a table for a form enables you to keep text and form fields segregated. For a simple e-mail collection, you need three rows and two columns.

- 7. Enter 3 in the Rows field and 2 in the Columns field. Set the other parameters to suit the page in which you're inserting the form.**

- 8. Click OK.**

Dreamweaver creates a table.

9. Enter the desired text in the first column of the first row.

Entering **Name** and **E-Mail** makes a lot of sense to us.

10. Position your cursor in the second column in the first row and then choose Form↪TextField.

Dreamweaver inserts a text field in the table cell.

11. Choose Window↪Properties.

The Property inspector appears.

12. Label the TextField Name.

The default name of the first text field in a document is Text. Highlight the default name and type the desired name in its place. In this case, type **Name**. (See Figure 4-1.)

Figure 4-1:
Setting the parameters for the text field.



13. Enter a value of 40 in the Char Width and Max Chars fields.

These values size the text field and limit the maximum number of characters users can input to 40. If your client has visitors with long names, you might want to increase these values.

14. Position your cursor in the second column of the second row and repeat Steps 10–13.

This time, name the TextField **E-Mail**.

15. Position your cursor in the first column of the third row.

16. Choose Insert↪Form↪Button.

Dreamweaver inserts a button in your form. If desired, you can change the button text in the Property inspector. Your finished form should resemble Figure 4-2.

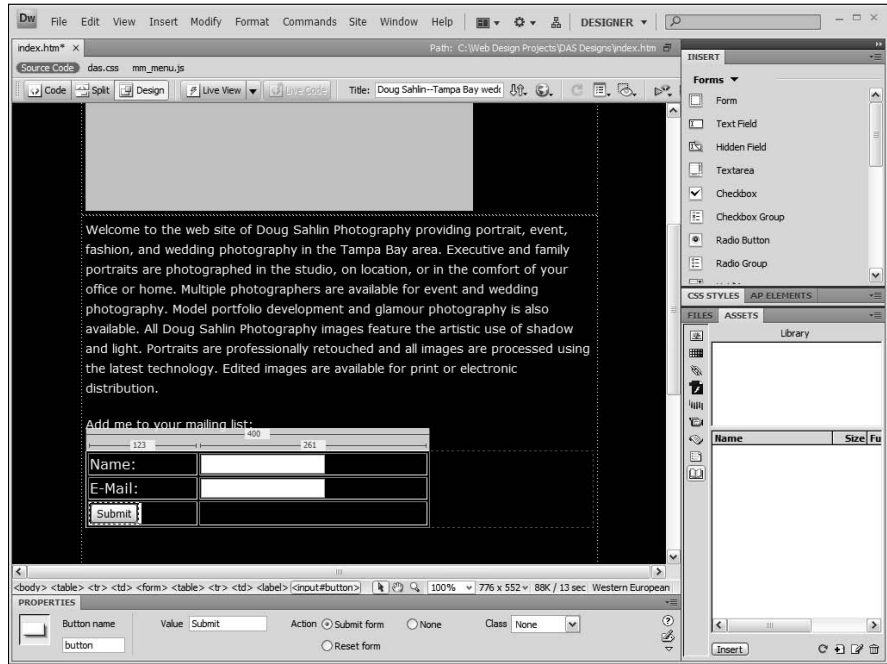


Figure 4-2:
The
completed
form.

Creating the Database

After you know what fields you want in your database, you're ready to create it. Now, we could show you a whole lot of complex code that creates a new database, but why work harder when you can work smarter? Most Web hosting services that have MySQLAdmin installed also have a little gem called PHPMyAdmin, which enables you to create databases from within the service's control panel. This dynamic duo gives you the power to define and create a database. Here's how to do it:

1. Log in to your Web server.

If you're not sure how to do this, check with your hosting service's technical support.

2. Create a new MySQL database.

Your Web hosting service probably has something like a MySQLAdmin icon. Click the icon and create a new database. You'll submit a username and password. Jot these down because you'll need them to connect to the database. Also, make sure that you choose the DBA User option, which means *database administrator*. If you're unsure, check with your Web hosting service's technical support.

3. After you create the database, launch PHPMyAdmin.

There should be an icon for this on your server’s control panel. If you’re not sure of how to launch PHPMyAdmin, check with your Web hosting service’s technical support.

4. Select the database you just created.

5. Click Operations.

The Operations section of the PHPMyAdmin dialog box appears. (See Figure 4-3.)

6. Enter a name in the Name text box (underneath Create New Table on Database).

You can have multiple tables in a database. However, for a simple database like this, you need only one table. Give the database a name that reflects its purpose. Don’t create a name that has spaces.

7. Enter a value in the Number of Fields text box.

For this database, you need three fields.

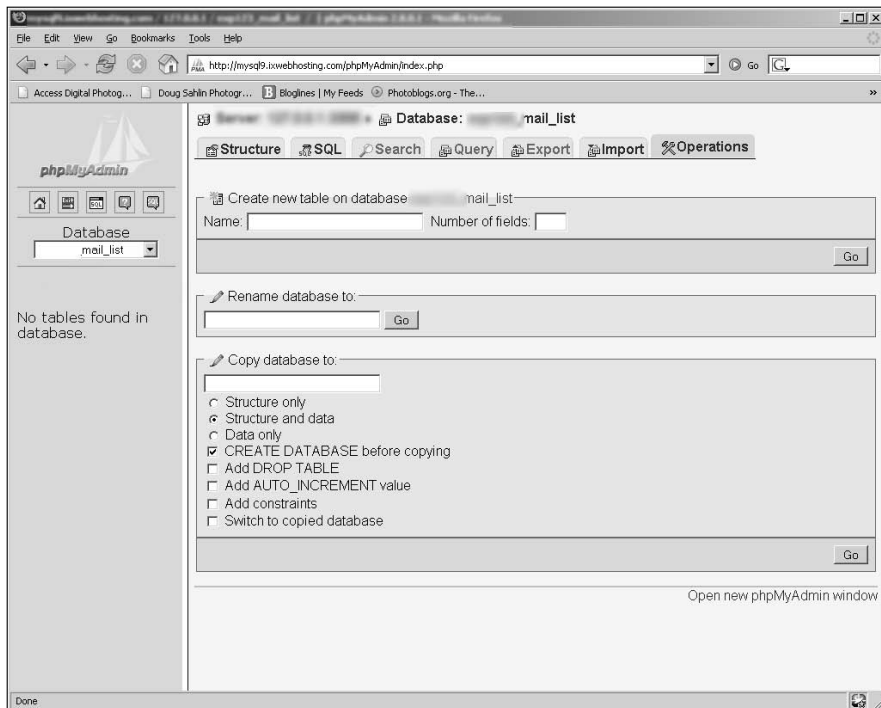


Figure 4-3:
Configuring
the
database.

8. Click the Go button.

The dialog box refreshes, and you're faced with the task of naming your fields. (See Figure 4-4.)

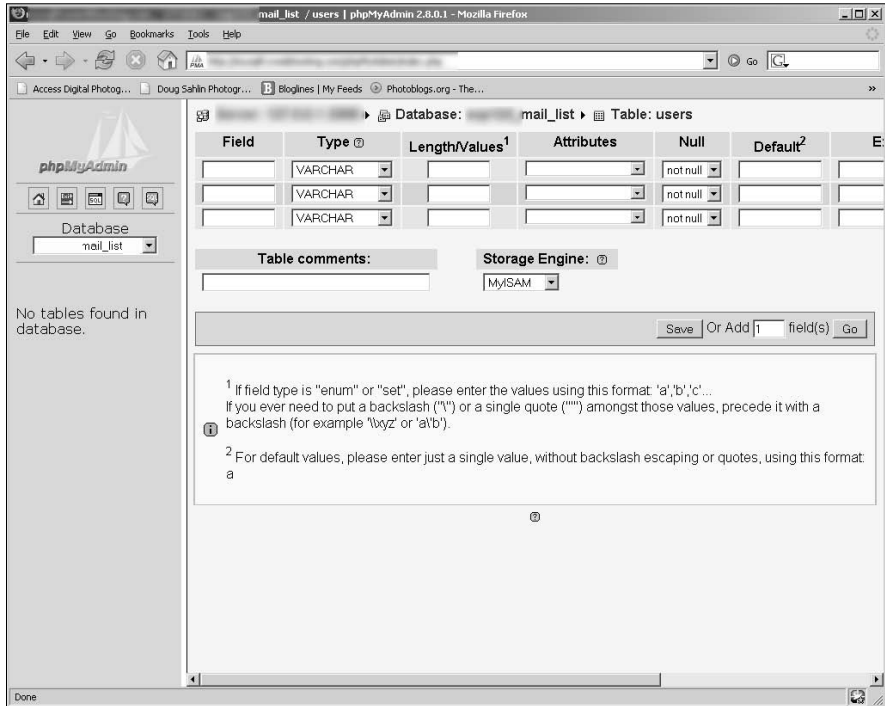


Figure 4-4:
Setting
the field
character-
istics.

9. Enter the names and characteristics for your fields.

- *First field:* Name the first Field **ID**, choose INT from the Type drop-down menu, enter **5** in the Length/Values field, choose Auto_Increment from the Extra drop-down menu, and then select the Primary radio button. The Web hosting service you're using may have different options. When all else fails, call tech support.
- *Second field:* Name the second Field **Name**, accept the default VARCHAR (*variable characters*, which means the field will accept text and numbers, which are read as text data and can't be used for arithmetic operations) option for Type, and then enter **40** in the Length/Values field.
- *Third field:* Name the third Field **E-Mail**, accept the default VARCHAR option for Type, and then enter **40** in the Length/Values field.

The second and third field names are the same names as those in the form you created. The value you enter is the same value you specified for Max Chars when you created the form. The first field generates a number every time a record is added to the database. This number is used to locate information in the database. The ID is incremented by a value of 1 every time a record is added to the database.

10. Click Save.

PHPMyAdmin creates the table and fields. Next, you create the PHP code that transfers the information from the form to the database.

Creating the PHP Code

After you have the form and the database ready to go, create the PHP code that will populate the database with e-mail addresses. You create variables for each form field and then create a connection to the database. Your code also puts the information in the database. Just follow these steps:

1. Create a new document in Dreamweaver.

Click Blank Page and choose PHP from the Page Type column.

2. Switch to Code view.

3. Create a new line after the <body> tag and then enter the following code:

```
<?php
    $Name=$_POST['Name'];
    $EMail=$_POST['EMail'];
    $connection=mysql_connect ("localhost",
    "myusername", "mypassword") or die ('I cannot
    connect to the database. ');
    mysql_select_db ("mydatabase",$connection) or die(
    "Unable to select database");
    $query = "INSERT INTO `mytable` (`ID`, `Name`,
    `EMail`) VALUES ('','$Name','$EMail')";
    mysql_query($query);
    mysql_close();
?>
```

You'll have to change the code to suit your Web server and the database information, but the first three lines of code will remain unchanged. Here's what's happening in this code:

- The first line signifies that PHP code follows.
- The next lines of code create two variables, \$Name and \$EMail, which are set equal to the names of the fields from the form you created where users enter their names and e-mail addresses.

- The line of code that begins with `$connection` creates a variable with the information needed to connect to the database. Most Web hosting services use `localhost` as the location for the database; however, some Web servers keep their MySQL databases on a separate server, in which case you'll need to enter the URL for the database server. Check with your Web hosting service's technical support staff for more information.
- The line of code that begins with `mysql_select_db` selects the database using the information from the `$connection` variable. Replace `mydatabase` with the name of your database.
- The line that begins with `$query` creates a variable with the instructions that insert the data into the database. Replace `mytable` with the name of the table into which you're inserting the form data. You might recognize the information (`'ID', 'Name', 'EMail'`). These are the fields in your database into which the information is being inserted. The information being inserted is specified by the word `VALUES`. The values being inserted are (`'' , '$Name', '$EMail'`). The single quotes are a placeholder for the ID field in the database. When the information is inserted in the database, MySQL assigns a number to the record. The ID field is the primary field in the database and is set to *auto increment*, which means that sequential numbers are assigned to the records entered: 1, 2, 3, 4, and so on.
- The line of code that begins with `mysql_query` tells MySQL that a query operation is being requested. The query performed gets instructions from the variable `$query`.
- The line `mysql_close()` ; closes the connection to the database. And if you've read Chapter 3 of this minibook, you know that `?>` ends the PHP code.

4. Save the document.

Save the document as `addrecord.php`, the same filename specified in the action of the form you created.

When you upload the HTML page with the form and the PHP file to your server, site visitors can enter their names and e-mail addresses into the form. When they click the Submit button — or whatever clever name you gave the button — their information is added to the database.

You need to add additional code to the head of the `addrecord.php` document. Otherwise, the site visitor will click the Submit button and get stuck on the `addrecord.php` page. The code you need to add is

```
<meta http-equiv="REFRESH" content="0; URL=http://www.mywebsite/thanks.htm">
```

After adding this code to the `addrecord.php` page, you need to create a new HTML document called `thanks.htm`, which is uploaded to the Web

site. The browser refreshes with this page after the code executes. Your Thank You page should contain site navigation and a message to site visitors thanking them for adding their information to the database.

Retrieving Information from a Database

Okay, after you have a database that accepts information from a Web page, how do you access it? You can choose from many commands to search a database. You can set up a form where the user enters the information he's searching for and then create the code that searches a particular field of the database and returns information that is identical to or similar to the user's request. This is a *query*.

To show you everything you can do with a database requires several chapters. To show you everything you can do with PHP and a MySQL database requires an entire book. Neither option is available, and our project editor was chomping at the bit for this chapter. To find out everything you ever wanted to know about MySQL and PHP, consider *PHP and MySQL For Dummies*, 3rd edition, *PHP & MySQL Everyday Apps For Dummies*, both by Janet Valade (both by Wiley), or *PHP & MySQL Web Development All-in-One Desk Reference For Dummies*, by Janet Valade, Tricia Ballad, and Bill Ballad (Wiley).

Now that the disclaimer's out of the way, we want to show you how to retrieve the data from the simple mailing list we show you how to create in the previous section.



If you didn't read the previous section, do so now.

To retrieve the data from the mailing list database, follow these steps:

- 1. Create a new document in Dreamweaver.**

Click Blank Page and then choose PHP from the Page Category column.

- 2. Switch to Code view.**

- 3. Create a new line after the <body> tag and then enter the following code:**

```
<table>
<tr>
<th width="120" align="Left">
Name</th>
<th width="120" align="Left">
E-Mail</th>
</tr>
</table>
<table>
<?php
$connection=mysql_connect ("localhost", "myusername", "mypassword") or
die ('I cannot connect to the database.');
```

```
mysql_select_db ("mail_list",$connection) or die( "Unable to select
database");
```

```
$query="SELECT * FROM users ORDER by name";
$result=mysql_query($query);
$num=mysql_numrows($result);
mysql_close();
$i=0;
while ($i < $num) {
$name=mysql_result($result,$i,"Name");
$email=mysql_result($result,$i,"EMail");
?>
<tr>
<td width="120" align="left" ><? print $name ?>
<td>
<td width="120" align="left" ><? print $email ?>
</td>
</tr>
<?php
$i++;
}
?>
</table>
```

4. Save the file and upload it to your server.

When you load the PHP file in a browser, it plucks all the data from the database. Figure 4-5 shows a hypothetical mailing list displayed using the code in the previous steps.

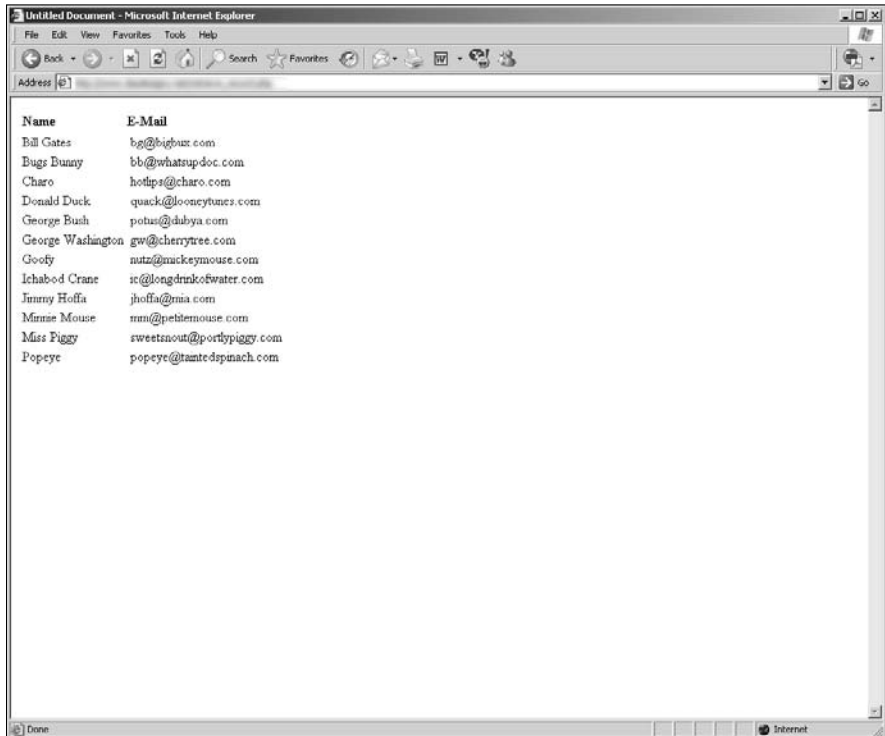


Figure 4-5:
Now that's a
mailing list.

You're probably thinking, "Yikes, that's a lot of code!" We don't leave you in a lurch, though. Read on to find out what the code means.

Listing 4-1 shows the code for the table that displays the titles for the information that will be retrieved. The titles use the `<th>` tag (table header), which boldfaces the text. The code also sizes the headers and aligns the text to the left.

Listing 4-1: The Table Titles

```
<table>
<tr>
<th width="120" align="Left">
Name</th>
<th width="120" align="Left">
E-Mail</th>
</tr>
</table>
```

In Listing 4-2, the first line of code creates a new table and the second line begins the PHP script.

Listing 4-2: Connecting to the Database

```
<table>
<?php
$connection=mysql_connect ("localhost", "myusername", "mypassword") or die ('I
cannot connect to the database. ');
mysql_select_db ("mail_list",$connection) or die( "Unable to select database");
$query="SELECT * FROM users ORDER by name";
$result=mysql_query($query);
$num=mysql_numrows($result);
mysql_close();
```

The new table stores the results of your query, which in this case is all of the names and e-mail addresses in the database. The second line of code creates a variable that contains all of the information needed to connect to the database. The line of code that begins with `mysql_select_db` creates a connection to the database, which in this case is named `mail_list`. The `$query` variable in the fourth line of code is set equal to the query that will be performed to select the information from the `users` table of the database. The results will be ordered by name. The `$result` variable in the fifth line of code is set equal to the `mysql_query` command, which queries the database using the information in the `$query` variable. The `$num` variable in the sixth line of code is set equal to the number of rows of data returned from the query. The final line of code closes the database.

Listing 4-3 shows the final lines of code. It's not as bad as it looks.

Listing 4-3: Using a while Loop to Extract the Data

```
$i=0;
while ($i < $num) {
$name=mysql_result($result,$i,"Name");
$email=mysql_result($result,$i,"EMail");
?>
<tr>
<td width="120" align="left" ><? print $name ?>
<td>
<td width="120" align="left" ><? print $email ?>
</td>
</tr>
<?php
$i++;
}
?>
</table>
```

The `$i` variable is set to 0 (zero), which, incidentally, is the ID number of the first record row returned from the database. The `while` loop will run as long as the value of `$i` is less than the variable `$num`. The next variable in the lineup is `$name`, which is set equal to the result of the first query. The variable `$i`, which is initially set to 0 (zero), is the first record that matches the search criteria and returns the data from the Name field of the database. The `$email` variable is similar but returns the result from the E-Mail field of the first record that matches the criteria. Now that you have your variables in a row, put some data in the table rows.

The next line ends the PHP code and is immediately followed by a table row, which has two cells. The first cell contains the contents of the `$name` variable, which contains the data from the Name field of the first record that matches our results. Notice the manner in which the PHP code is entered: `<? print $name ?>`. This is a shortcut when you need to insert PHP in an HTML tag. The `<?>` tag signifies the start of PHP code, and the `?>` tag ends it. The next table cell contains the contents of the `$email` variable, which contains the data from the E-Mail field of the first record that matches the query results.

And now is more PHP code. Basically, the code `$i++` is increasing the value of the variable `$i` by a value of 1. The `++` shortcut is code to increase a value by 1. The closing curly bracket (`}`) signifies the end of the instructions included in the `while` loop. The loop continues until all of the records are displayed.

Chapter 5: Additional Site Interactivity

In This Chapter

- ✓ **Considering adding a blog**
- ✓ **Letting users speak: Message boards and wikis**
- ✓ **Setting up automatic notifications: RSS feeds**
- ✓ **Keeping in touch with e-mail**

It seems as though interactivity is the buzzword when it comes to Web sites. If you design sites that aren't interactive, you're not a happening designer. And it's a well-known fact that happening designers get most of the work, and non-happening designers run the risk of being has-beens. If you've read previous chapters of this minibook, you have an idea of the type of interactivity you can create when you design pages for your clients. But wait; there's more! In fact, a whole lot more. This chapter shows you other forms of interactivity you can add to your pages. In fact, it shows you a whole lot of interactivity in this chapter.

Engaging Visitors with an Online Journal (Blog)

Blogs have become the hottest thing since sliced bread. It seems as though everyone and his little sister has a blog these days. Some of them are vanity blogs, and others contain useful information or points of view. Figure 5-1 shows a blog that contains useful information about digital photography. The creators are two best-selling authors of books on digital photography and Photoshop. The blog is interactive in that readers can leave comments on blog posts.

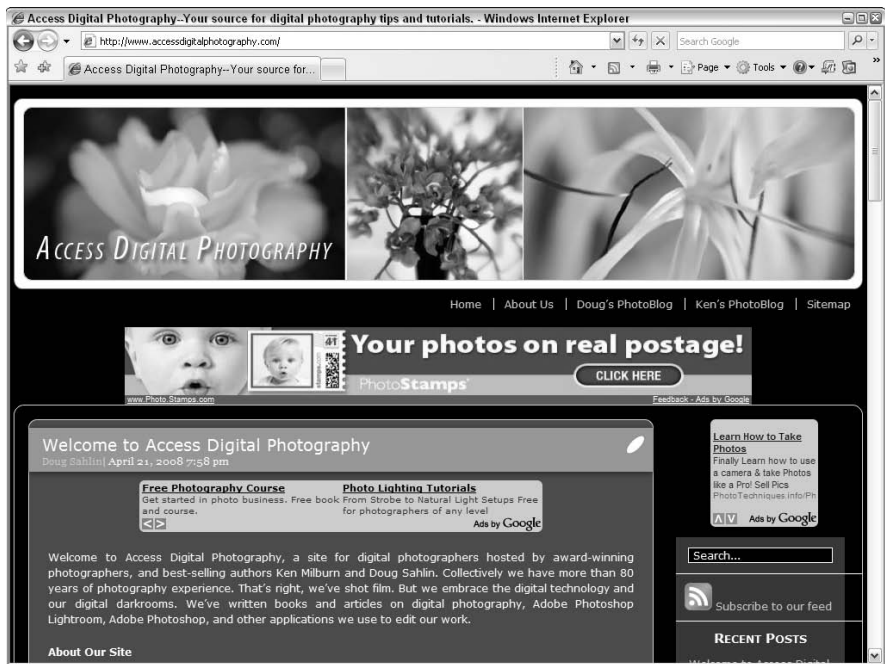


Figure 5-1:
Use blogs
to dispense
useful
information.

Savvy businesspeople also use blogs. Because the information in a blog is updated frequently, Web sites with blogs are ranked highly by search engines. If a commercial blog site is optimized for search engines, it attracts traffic that might include potential clients.

You can download two popular — and free — blog applications. One is b2evolution. You can find information about b2evolution and download the application at <http://b2evolution.net>. WordPress is another popular blog application. You can find information about WordPress and download the application at <http://wordpress.org>. You can customize both applications to suit a Web site.

Blogs are also ideal venues for distributing podcasts. Developers like to create plug-ins for blog applications. For example, the PodPress plug-in for the WordPress blog makes posting a video or audio podcast child's play. The plug-in enables the site owners to view statistics, such as how many times an episode has been downloaded and played. It also features a player. Figure 5-2 shows a podcast being hosted with a WordPress blog.

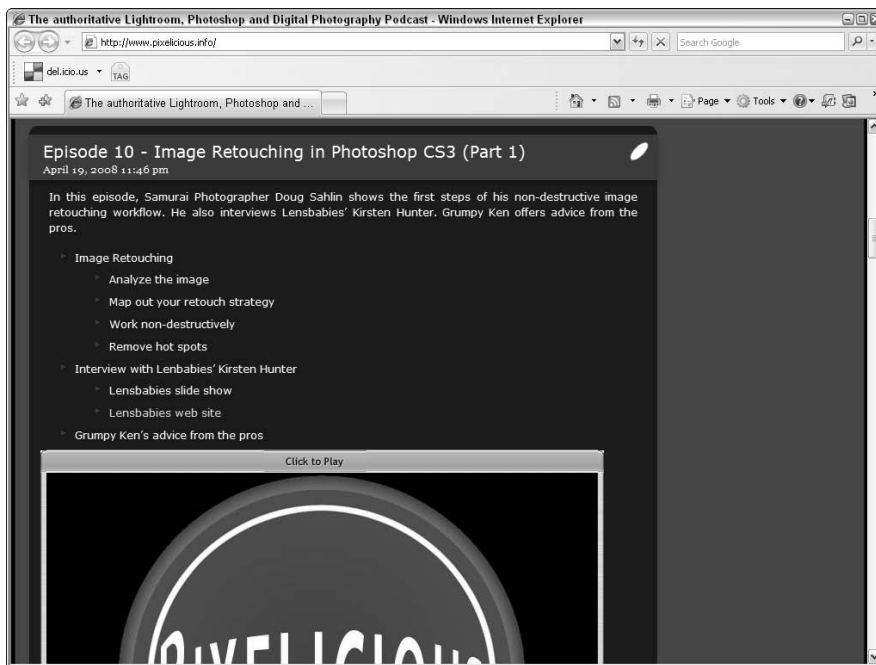


Figure 5-2: Blogs and podcasts reside happily forever after.

Book VI
Chapter 5

Additional Site
Interactivity

The aforementioned blog applications use PHP and a MySQL database. Both applications enable users to post and manage blogs using a browser-based interface. Before you add one of the aforementioned blog applications to your site (or to your client's site), make sure your server has the proper version of PHP and MySQL at your disposal. For more information on blogs, see Book IX, Chapter 1.

Initiating Online Conversations: Forums and Message Boards

If your client has a product or service that includes a user base that likes to exchange information about topics related to the product or service, an online forum or a bulletin board is an ideal addition to the main Web site. Forums give users a chance to interact with each other. Someone on your client's staff moderates the bulletin board. The moderator can add his two cents to a forum thread. Forums can create customer loyalty. Forums also turn visitors into potential customers.

A *forum* or *bulletin board* is an application that enables visitors to post information. The information in a bulletin board changes; therefore, the page needs to be generated on the fly when someone visits the site. Most bulletin board and forum applications use ASP or PHP code to generate the page and pluck the required information from a database. The application writes new forum posts to the database, which also contains subscriber information and so on. One forum application that you can download free is phpBB. The application uses PHP code combined with a MySQL database. For more information about phpBB and to download the application, visit www.phpbb.com.

The forum application enables the administrator to determine whether guests can make a post or whether only registered members can post messages in the forum. The administrator can subdivide a forum site into forums that pertain to related topics. The forum site administrator manages the forums. The administrator also grants rights to visitors of the forum site, which determine who can view posts, write posts, moderate forums, and so on.

When someone posts a new topic on a bulletin board, other visitors (or registrants, depending on the permissions allowed by the forum administrator) can respond to the post. The topic and replies are *threads*. The information in the forums is stored in a database on the server. Ken Milburn, a photographer and author, and Doug recently used phpBB to create a forum for digital photography. Figure 5-3 shows the home page of the forum.

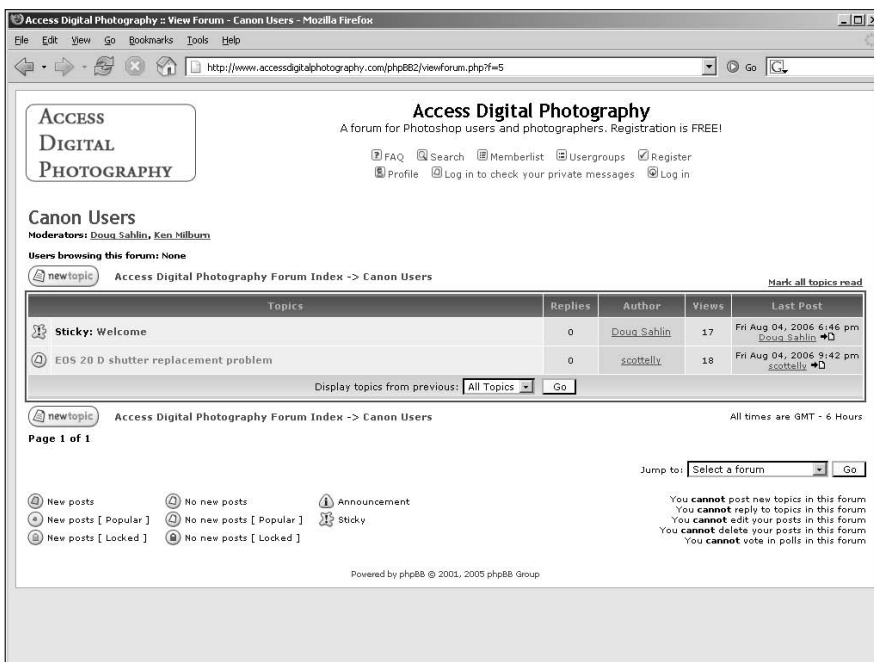
The forum site in Figure 5-3 is divided into several interest areas, including forum areas devoted to Photoshop, Painter, and specific brands of digital cameras. Site visitors can view other visitors' comments in a topic area by clicking the appropriate link on the home page. Each topic area lists the most recent posts. (See Figure 5-4.) A forum page generated with the phpBB application shows a visitor what rights she has in a list at the bottom of the page.

Another nice feature of a forum using the phpBB application is the ease of administration. After logging in, an administrator can manage every aspect of the forum using the Administration panel. From within the panel, the administrator can assign rights to user ranks, create new user ranks, delete users, manage and create forums, and so on. Figure 5-5 shows the Forum Administration section of the Administration panel.

Figure 5-3:
A phpBB forum for digital photographers.



Figure 5-4:
A topic area in a phpBB forum.



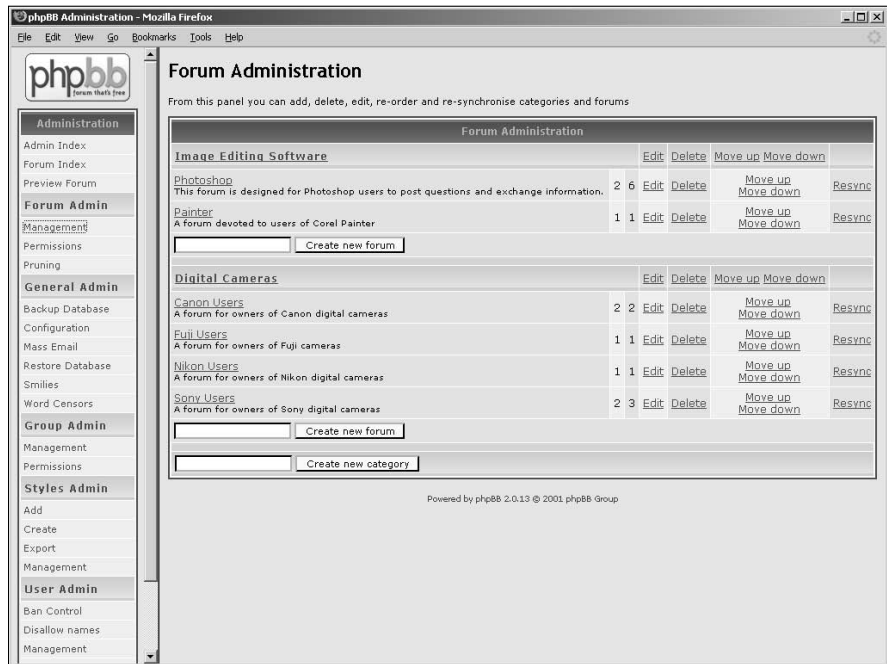


Figure 5-5:
Administering
a phpBB
forum.

Establishing an Online Community: A Wiki

A *wiki* (pronounced just like it looks, *WICK-ee*) is an online community of users who can share documents and spreadsheets, create Web pages, and more. The name *wiki* comes from *wiki wiki*, which means *quick* in Hawaiian. The software that runs a wiki enables members of the community to use a common interface to create content for the community. *Wiki* is server software that enables users to create and edit Web page content using any Web browser. Wiki software enables visitors to insert hyperlinks, using simple text syntax to create new pages and cross-links between internal pages. Users can modify the organization of wiki content as well as edit the documents. To check out a very busy and active wiki community, visit www.wikipedia.org. Figure 5-6 shows a page from a wiki site where visitors can edit the page.

The very nature of a wiki means that literally anyone who can access the wiki can edit pages within the wiki site. Some wiki communities require users to register before they can edit pages. Others require an authentication code, while others let the user freely edit the page — but they record the user's IP. You might wonder whether pages could be wiped out. The answer is yes. However, most communities are fairly civil, and pages are rarely wiped clean. Figure 5-7 shows a wiki page that's ready for editing. You can find wiki software by typing those very words in your favorite search engine. Everything you want to know about wikis can be found in *Wikis For Dummies* by Dan Woods and Peter Theony (Wiley).

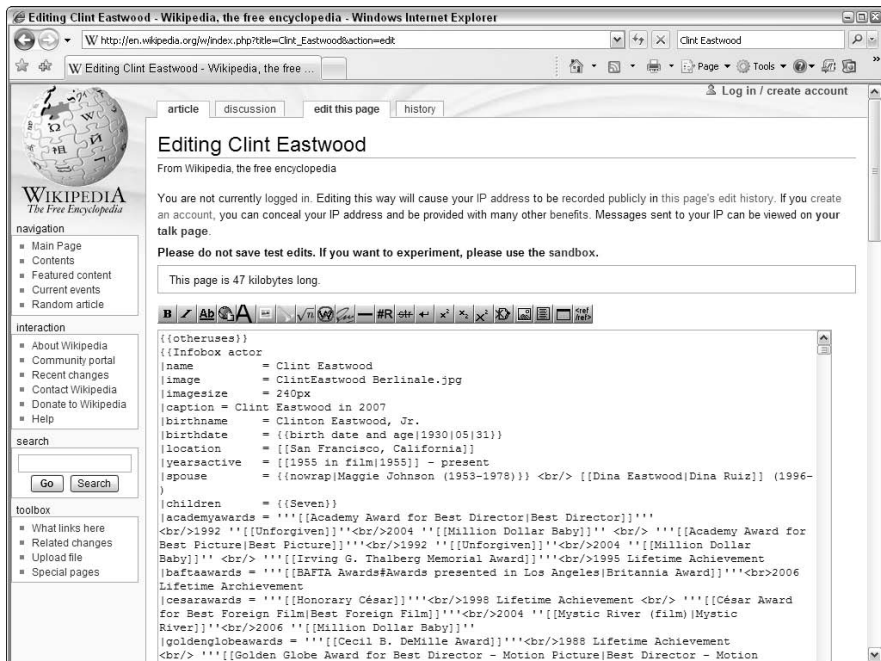
Figure 5-6:
Edit me.
Go ahead and make my day.



Book VI
Chapter 5

Additional Site
Interactivity

Figure 5-7:
Now let's see, if we boldface this and italicize that . . . yada, yada, yada.



Feeding Them Your Content: RSS

If you set up a blog or a wiki for a client, you can create an RSS (really simple syndication) feed to notify subscribers when new information is posted. Subscribers can use an online RSS feed service or download an RSS feed reader to keep track of desired RSS feeds. Either method notifies a subscriber when you update the RSS feed. RSS delivers information to RSS feed readers through an XML document, known as an *RSS feed*, a *Webfeed*, an *RSS stream*, or an *RSS channel*. The RSS feed notifies users when newly released content is added. The type of notification differs depending on the vehicle reading the stream.

Some people don't like to be hassled with auxiliary software. This type of Web site visitor prefers to do everything through a browser. When those visitors visit a site with an RSS feed, they see an RSS icon. When the icon is clicked, they see a page that contains the most recent posts. If they click the **Subscribe to This Feed** button in Internet Explorer (IE) 7. The feed can be saved in the default Feeds folder or in a folder specified by the user. After the **Subscribe Now** button is clicked, Internet Explorer checks the feed on a regular basis and notifies the user when a new post is added to the site.

The feed shown in Figure 5-8 enables the user to peruse the newest headlines or all headlines. If a user finds a headline that piques his interest, he can click the headline link to read the article from the Web site hosting the RSS feed.

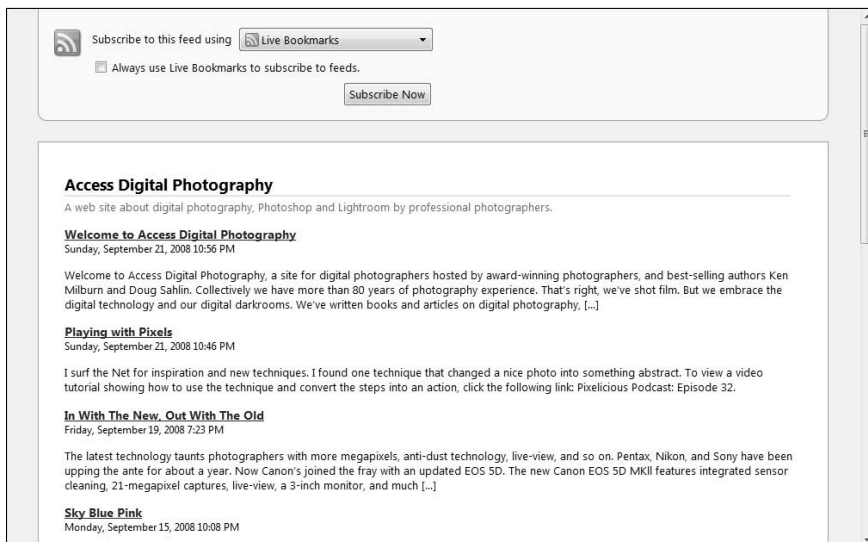


Figure 5-8:
I'm gonna
subscribe
to this
feed. Now.

The popular Firefox browser offers different options for subscribing to a feed. When someone visits a blog with a feed, an RSS icon appears to the right of the address window. If the site has an *Atom feed* — The Atom feed format uses XML data to propagate feeds from a Web site — the user can subscribe by clicking the icon. If the user prefers to use a different blog aggregator, he can specify this by choosing Tools → Options → Feeds, and then choosing the desired *feed aggregator* — also known as a feed reader. Figure 5-9 shows a podcast hosted by a blog as displayed in the Google Reader.

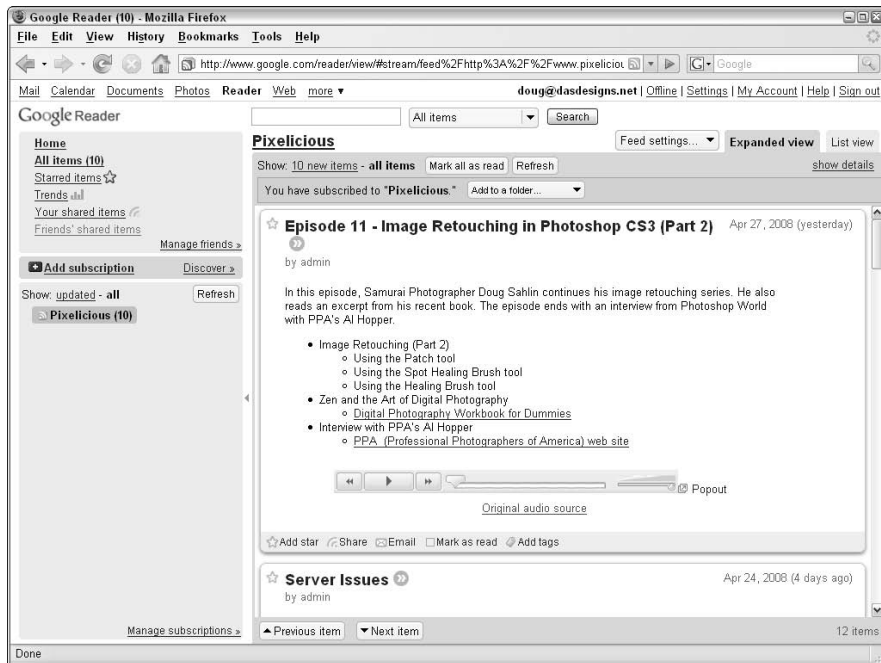


Figure 5-9:
Read me,
baby.

Using an online feed reader

Many people prefer to use online RSS feed reading services instead of downloading and installing an RSS feed reader. An *online RSS feed reading service* enables you to track feeds through something you're already familiar with: namely, a Web browser. Many online services are available. We've explored three online RSS feed reading services: Bloglines (www.bloglines.com), Google, and Pluck (www.pluck.com). If you haven't explored the wonderful world of RSS feeds, you can subscribe to each service free. Figure 5-10 shows an RSS feed as it appears in Bloglines. Clicking the name of the blog transports the user directly to the blog site in a new browser window. Clicking the title of the post displays the entire post in a new browser window from the URL of the post.

Figure 5-10: Reading a blog post, using the Bloglines online RSS reader.



You can also subscribe to a feed through Yahoo!.

Creating an RSS feed

To broadcast an RSS feed, you create an XML (eXtensible Markup Language) document. Even if you're not familiar with XML, you can create an RSS feed in no time flat with a word-processing application, such as Notepad. The document needs the following tags:

- ◆ `<rss>` — Encompasses the entire feed. It tells feed readers what version of RSS the feed is.
- ◆ `<channel>` — Houses all the information about the feed. You can have more than one channel in a feed.
- ◆ `<title>` — Displays the title of the blog.
- ◆ `<link>` — Contains the URL of the blog.
- ◆ `<description>` — Explains the purpose of the blog and the type of information in the blog posts.
- ◆ `<language>` — Chooses the language in which the blog is written.
- ◆ `<item>` — Displays the information for each blog post included in the feed. Each item needs the following tags:
 - `<title>` — Shows the title of the blog post.

- `<link>` — Adds the URL for the blog post.
- `<description>` — Provides a description of the blog post. For this tag, I cut and paste the first few sentences from the blog post.

That's pretty much it in a nutshell. You can add other bells and whistles to the post, such as the time it was created, and so on. Listing 5-1 shows a simple RSS feed for a single blog post.

Listing 5-1: A Really Simple XML Document for an RSS Feed

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
  <channel>

  <title>Doug Sahlin Photography</title>
    <link>http://www.dasdesigns.net/blog</link>
    <description>A digital photography and Photoshop blog
      with news, tips, and tutorials.</description>
    <language>en-us</language>
  <item>
    <title>Rescuing Anemic Skies</title>
    <link>http://www.dasdesigns.net/blog/?p=21</link>
    <description>I'm sure you've photographed what you thought
      was a lovely scene with an impressive cloudscape only
      to find that the sky was a little flat in the resulting
      image. Although the image is properly exposed, the details
      in the clouds are not like you remember them. There's
      a relatively easy fix to this problem in Photoshop:</
      description>
  </item>
</channel>
</rss>
```

To add an item to an RSS feed, open a browser and navigate to the URL for the blog post you're adding. Then, it's a simple matter of cutting and pasting the title, link, and description for the new item. For the description, you can use the first few sentences from the blog post. The first lines of a blog post should be interesting enough to pique a visitor's curiosity and make her want to read the rest of the post. Each post is a new item with a title, link, and description, as shown in Listing 5-2.

Listing 5-2: An RSS Feed with Multiple Posts

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
  <channel>
  <title>Doug Sahlin Photography</title>
    <link>http://www.dasdesigns.net/blog</link>
```

(continued)

Listing 5-2 (continued)

```
<description>A digital photography and Photoshop blog
with news, tips, and tutorials.</description>
<language>en-us</language>
<item>
<title>Rescuing Anemic Skies</title>
<link>http://www.dasdesigns.net/blog/?p=21</link>
<description>I'm sure you've photographed what you thought
was a lovely scene with an impressive cloudscape only
to find that the sky was a little flat in the resulting
image. Although the image is properly exposed, the details
in the clouds are not like you remember them. There's
a relatively easy fix to this problem in Photoshop:</
description>
</item>
<item>
<title>Selling Your Photos</title>
<link>http://www.dasdesigns.net/blog/?p=20</link>
<description>Next time you're at a shopping center that
has a frame shop, stop by and take a look at their
stock. Chances are they have several framed paintings or
photographs on display to showcase the various styles of
frames they sell. The shop owner probably changes the
display frequently to pique customer interest. If so, this
frame shop is an excellent place to sell your photos.</
description>
</item>
</channel>
</rss>
```

You can use Listing 5-1 as a template for your own RSS feed. As long as you create a carbon copy of the first three tags, you can fill in the blanks. In fact, to show you how nice we are, we'll leave you a template (see Listing 5-3) from which to work.

Listing 5-3: An RSS Feed Template

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
<channel>
<title></title>
<link></link>
<description></description>
<language></language>
<item>
<title></title>
<link></link>
<description></description>
</item>
</channel>
</rss>
```

After you create your feed, save it as an XML document. We call this one `feed.xml`. Upload the document to the `root` directory of your blog. You're now ready to start feeding the masses, so to speak.



For those who don't like manually typing the code shown in the previous examples, there is another way of creating an RSS feed: the free application called RssPublisher. You can download it from www.rsspublisher.com. The application consists of three tabs: Feed, Headlines, and Publish. On the Feed tab, you enter the information about your feed. On the Headlines tab, you enter the title, description, and link to each post you want to feed to the public. The Publish tab enables you to publish the feed directly to the Web site hosting the blog, to your local machine, or to both.

Many blog applications create feeds. For example, the latest version of WordPress generates a blog feed that can be accessed by IE 7. If you really want to err on the side of caution, you can use the online service FeedBurner (www.feedburner.com). After creating a FeedBurner account, you can *burn* — FeedBurner speak for adding your site to their list of feeds — a feed. This Google service is free and invaluable if you're hosting a podcast on a blog. Use the FeedBurner feed address (updated automatically from your site feed whenever you add a new post) when submitting your podcast to iTunes, and your feed will have all the required iTunes tags and other required nomenclature. As an added bonus, you can use FeedBurner to view detailed statistics about your feed, as shown in Figure 5-11.

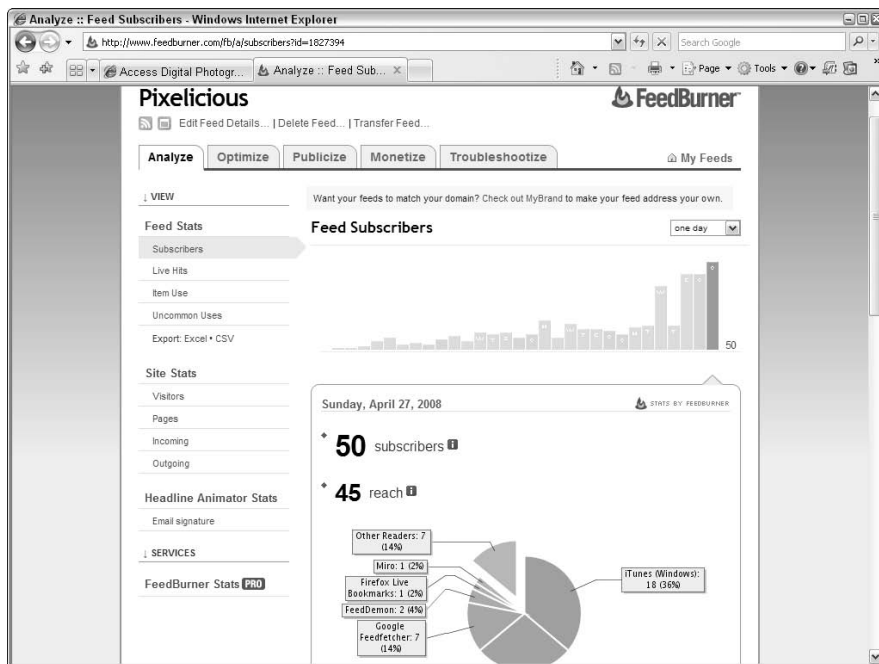


Figure 5-11: FeedBurner: The wise choice for burning a red-hot feed for iTunes.

Validating your feed

You validate parking tickets, so why not validate your RSS feed? If all RSS readers can't read your RSS feed, it's not useful. You can validate your RSS feed free by following these steps. (*Validating a feed* simply means that the XML document that designates your feed is correct and will function with feed readers and other services.)

1. **Log on to the Internet and navigate to `www.feedvalidator.org`.** (See Figure 5-12.)

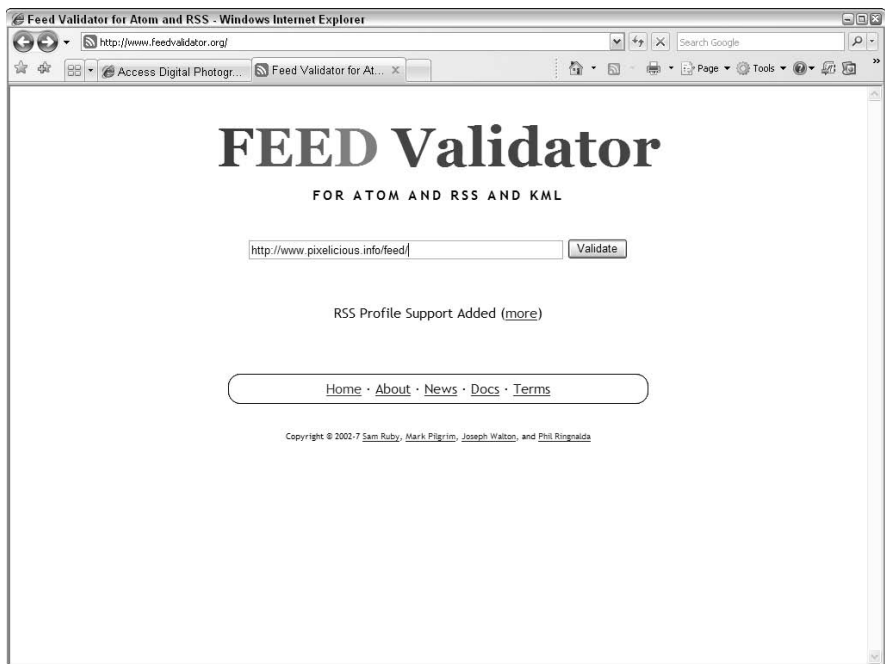


Figure 5-12:
Please
validate me;
set me free.

2. **Enter the URL to your RSS feed.**

Enter the URL plus the document name: for example, `http://www.mywebsite.com/feed.xml`. Or, enter the URL to a feed created by a blog application: `http://www.mywebsite.com/feed`.

3. **Wait.**

The feed validator checks your feed and tells you the status of your feed. Figure 5-13 shows the results when a feed is valid.

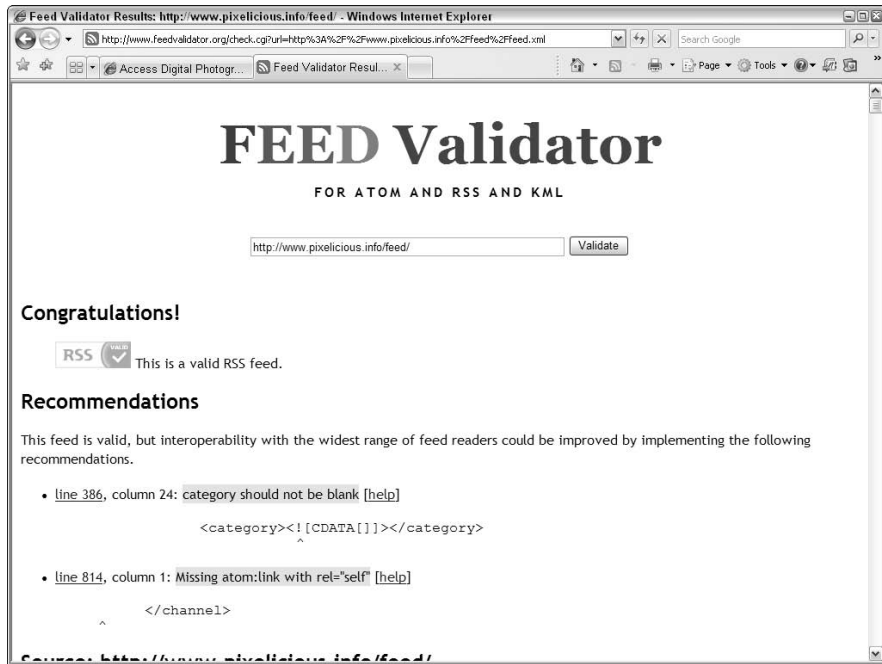


Figure 5-13:
Look Ma,
I'm valid!

If your feed isn't valid, the feed validator highlights the troublesome lines of code and lets you know what steps to take to correct the problem. You need to make sure UTF-8 encoding is mentioned in the XML document, as shown in the previous examples. However, some Web servers read the file as having ANSI encoding. If this occurs, the error will show up when you validate the feed. To correct this problem, follow these steps:

1. **Open a word-processing application such as Notepad.**
2. **Create a new document.**
3. **Enter the following text:**

```
AddType 'application/xml; charset=UTF-8' xml
```

4. **Save the file as .htaccess.**

Your word-processing application might save the file with an extension. If this is the case, rename the file prior to uploading it.

5. **Upload the file to the blog root directory.**

When you validate the blog now, the .htaccess document ensures that the XML file is delivered with UTF-8 encoding.

Getting your feed going

After you have an RSS feed, it's time to announce the fact to the world. Of course, you want to let current visitors to the blog know that it's syndicated with an RSS feed. You can do this by adding the code in Listing 5-4 to the applicable place in your blog page. This code uses three RSS feed reading services: Yahoo!, Bloglines, and Rojo. In addition, consider adding text that shows the URL to your feed. For example, you could add a line of text that reads

To add our blog to your RSS feed reader, enter the following URL: `www.mysite.com/blog/feed.xml` in the applicable field in your feed reader.

Listing 5-4: Code to Subscribe to a Blog

```
<h2>Subscribe to my blog!</h2>
<form name="subscribe">
<p>Subscribe to this blog via RSS feeds.
</p> <select name="feeds">
<option value="http://add.my.yahoo.com/rss?url=
  <?php bloginfo('url'); ?>/feed.rss">My Yahoo!</option>
<option value="http://bloglines.com/sub/
  <?php bloginfo('url'); ?>/">Bloglines</option>
<option value="http://rojo.com/add-subscription?resource=
  <?php bloginfo('url'); ?>/">Rojo</option>
</select><br /><p></p>
<input type="button" onclick="location=document.subscribe.feeds.
  options[document.subscribe.feeds.selectedIndex].
  value;" value="Subscribe">
</form>
```

If you use FeedBurner to burn your feed, you can get the proper code to include a link to subscribe to the feed from your Web site. Even though your own site might have a feed, you should use the FeedBurner feed address because FeedBurner records subscribers and visitors to your site.

Finding the place in your page code where you add the code in Listing 5-4 will take a bit of searching when you're using a blog application. After all, you didn't write the code. You should be able to figure it out by determining where you want to place the form in the blog page, by viewing it online, and then by examining the code in your HTML editor with a fine-toothed comb. When you successfully add the code to a blog, you'll see a window with a down-arrow that opens a drop-down menu with your subscription choices. Figure 5-14 shows a blog Web site hosting a podcast with buttons to subscribe to the feed and subscribe through iTunes.



Figure 5-14:
Subscribe
to my blog,
or I'll
spam you!

Book VI
Chapter 5

Additional Site
Interactivity

You can publicize a blog by submitting it to online blog readers, such as www.bloglines.com or www.pluck.com. Another alternative is to use an application that submits your blog feed to multiple sources. One such application is RSS Submit Pro by Allscoop. This works like the software you can purchase to submit Web sites to search engines. And the wonderful thing is that it's free. We've submitted two blog feeds to numerous sites with the software and have noticed an increase in the amount of traffic that comes to each blog.

The application is self-explanatory. You launch the application and then add the RSS feeds you want to submit. (See Figure 5-15.) You then select the sites to which you'd like to submit the feed. (See Figure 5-16.) When you click the Next button (the right-pointing arrow), the application submits the blog to the sites you selected while you sit back and enjoy a fresh cup of your favorite beverage. You can download Submit Pro from www.allscoop.com/tools/rss-submit.



Figure 5-15: Select the feeds you want to submit.

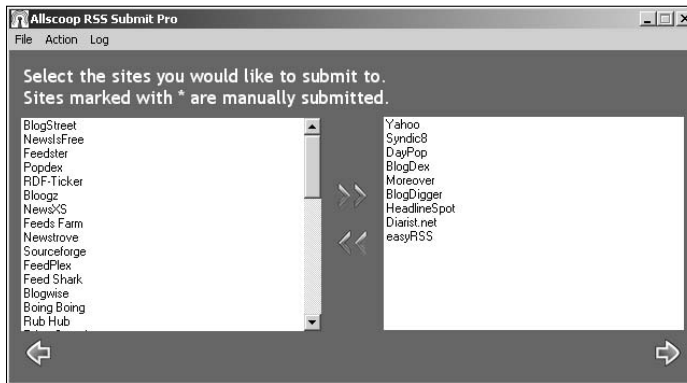


Figure 5-16: Select the sites to which you want to submit the feeds.

Branding through E-Mail

Anyone who is serious about doing business online should have an e-mail address associated with the Web site. Sure, you can conduct business by sending an e-mail with one of the free online services, but who wants to do business with `friendlyfred@hotmail.com`? In addition to the credibility factor, the free services limit the number of messages you can receive. The better solution is to set up one or more e-mail addresses using your Web server's control panel. E-mail addresses are also a wonderful way to make a Web presence seem like a huge company. The following list shows examples of addresses you can set up for a fictitious Web site called `widgetsrus.biz`:

- ◆ `john@widgetsrus.biz`
- ◆ `sales@widgetsrus.biz`

- ◆ info@widgetsrus.biz
- ◆ service@widgetsrus.biz

If your client uses an application like Microsoft Outlook to manage mail, he can easily set up accounts to receive and send messages from each e-mail address. The recipient will never be the wiser. When the business grows, your client can add additional e-mail addresses for other employees and transfer the other addresses he's currently managing to the applicable employee.

Keeping Them Up to Date: Newsletters

If your client's site changes frequently — and it should — he'll want to get the word out when new material is posted on the site. One of the easiest ways to do this is telling people who have visited the site that new information is posted. You're probably thinking that there's no way your client is going to e-mail each and every person who has stopped by his site. And you'd be right; that's way too much work.

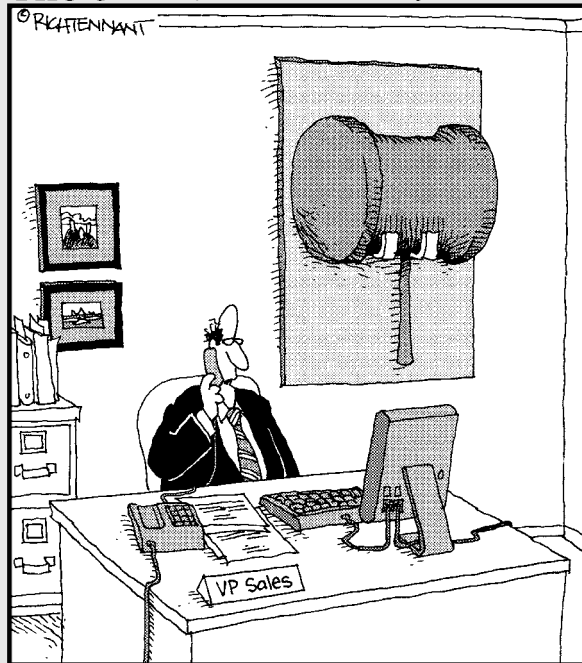
The easy way to keep visitors abreast of new information is to send them newsletters on a regular basis. When you design the site, you create a form that enables visitors to subscribe or opt in to the newsletter list. The visitor's information is stored in a database. You could create your client's monthly newsletter, but here's an easier way: an application that enables your client to create a newsletter from a template (that you design) and to send that newsletter to subscribers. The application should also give your client the option of managing the mailing list. Such applications exist. If you don't have access to one, type **e-mail list manager** in your favorite search engine.

Book VII

E-Commerce

The 5th Wave

By Rich Tennant



"It's web-based, on-demand, and customizable.
Still, I think I'm going to miss our old sales
incentive methods."

Contents at a Glance

Chapter 1: An E-Commerce Primer	565
Nailing Down E-Commerce Concepts.....	565
Adding Basic E-Commerce with PayPal.....	575
Using PayPal Website Payments Pro	578
Integrating with Credit Card Authorization Packages	579
SSL — What Is It?	581
E-Commerce Do’s and Don’ts.....	582
Chapter 2: Building an E-Commerce Site.	585
Technological Considerations	585
Usability: Thinking Like a Customer	586
Legal Considerations: Call a Lawyer	588
Creating an E-Commerce Site.....	590
Using a Shopping Cart.....	598
Selling Online	604
Chapter 3: Maintaining an E-Commerce Site	609
When Technology Breaks Down.....	609
Handling Customers	611
Search Engine Optimization (SEO) and Marketing.....	614
Figuring Out When to Upgrade	624

Chapter 1: An E-Commerce Primer

In This Chapter

- ✓ **Establishing a domain name**
- ✓ **Serving your Web catalog to customers**
- ✓ **Accepting payments with PayPal**
- ✓ **Getting paid with credit cards**
- ✓ **Using a secure server**

Your client is an entrepreneur with a product or service for sale and wants to take the business to the next level. Or perhaps you're creating a Web site for a bricks-and-mortar business that wants to kick it up a notch. Your solution for either client is quite simple: Design an interactive e-commerce site for your client. When done correctly, an e-commerce site makes it possible for your client to conduct business worldwide, 24 hours a day, 7 days a week. That's right — a store that never closes.

Of course, any venture has its pitfalls. Your goal as a Web designer is to create an attractive, interactive site that your client's customers can easily navigate. In addition to the usual bells and whistles you'll include on an interactive site, you also need to provide some method with which Web site visitors can order and pay for their wares as well as a method for your client to collect her money. In this chapter, you see basic e-commerce concepts such as setting up the site, working with credit card authorization packages, and obtaining a secure server.

Nailing Down E-Commerce Concepts

If you build it, they will come. Not! The world of e-commerce enables a merchant to branch out and present her service or product to a worldwide audience. However, if the merchant doesn't address certain issues, the site won't succeed. It often falls on the shoulders of a savvy Web site designer — and that would be you, reader — to advise your client. The following sections show you what to consider when creating an e-commerce site.

Establishing an online identity

Every Web site has a domain name — it's a law. A *domain name* is unique, like a phone number. When entered in the Address field of a Web browser and preceded by `www.`, it leads the flock to the proper Web site. The domain name for an e-commerce site should serve two functions:

- ◆ Describe the business to potential customers
- ◆ Navigate to the merchant's online catalog

Your client might already own a domain name. However, if the name doesn't fit, you must not quit even if your client thinks the domain name is the most clever thing since the invention of the fork. When you cross this “fork” in the road, recommend to your client to register a more appropriate name. After all, you can register a domain name rather inexpensively these days. In fact, domain registration is free with many Web hosting packages. (And, the client is paying you for your wisdom and foresight, too.)

So what's in a name? Everything. The domain name should be easy to remember, be short, and tell something about the business. The name should also be unique. For example, people associate the Amazon.com domain with the largest online seller of books — and other assorted sundries — in the world. A domain name of Isellbooks.com, tells the visitor about the business, but it's not unique, and it's not short.

If your client already has an established bricks-and-mortar business, use that name, or a clever derivative thereof, as the domain name. For example, if your clients are attorneys and their domain name is `carnettsmithLaw.com`, consider changing it to `Cslaw.com`. Many firms personalize a domain name by using the word “my” as part of the domain name: for example, `mycslaw.com`.

Researching domain names

As we mention earlier, a domain name is like a phone number. When a Web surfer enters the domain name into the Address field of a browser window, the browser refreshes to the Web site's home page. That's right: Domain names are unique. Each of the zillions of Web sites out there has a unique name. When you or your client is trying to come up with a unique domain name for the e-commerce site you're designing, you can quickly cut to the chase and figure out whether a domain name has already been reserved by following these steps:

1. Log on to the Internet and navigate to `www.betterwhois.com`.

You can research a domain name at most *registrars* (the companies that take your money in exchange for registering a domain name). However, you can use the free Better-Whois.com service to search all domain registrars to see whether the name has been reserved.

2. Enter the desired domain name in the Search field, as shown in Figure 1-1.



Figure 1-1:
Find out whether a domain name is available.

3. Click the Search button.
4. Enter the security code and then click Continue.

The domain registrar databases are searched. If the domain name is reserved, a message to that effect appears on the results page. If the domain is reserved, you can get further information about the domain by clicking one of the links. If the domain name is available, the Web page refreshes with a message to that effect, and you have the opportunity to register the domain name, as shown in Figure 1-2.



If the desired domain name isn't available, see whether the same name is available with a different domain name suffix. Most businesses like to register their domain as a top-level (.com) domain. However, the .net, .biz, and .info suffixes are also acceptable for e-commerce sites.



Figure 1-2:
If you like the domain name, book it Dan.



GoDaddy.com might well indeed be the granddaddy of all domain registrars. You can search for a domain at this site. If the desired domain name isn't available, the results page lists derivatives your client might want to consider.

Purchasing domain names

After you find an available and desired domain name, you can reserve the domain by purchasing it. Domain registrars are not regulated. There are many from which to choose. You can reserve a domain name for as little as \$6.95 per year. However, make sure you research the domain registrar before reserving the name with them. Many of the low-price registrars come with strings attached, or they try to sell you Web hosting along with the domain name. And the Web hosting provided by the domain registrar might not be applicable for e-commerce.

Many Web hosting services include domain registration as part of their hosting fee. Of course, your client is responsible for renewing the domain name after the registration expires. Sometimes, you can get a better deal if you register the domain for longer than one year. Like everything else on the Internet, the price of registering domains is in a constant state of flux. If you do a bit of research before registering your client's desired domain name, you'll find the best package currently available. Just type **domain registrars** into your favorite search engine and you'll have enough results to keep you busy for a few hours.



If you can get your desired domain name with the .com suffix, consider purchasing .net, .biz, .info, and .org suffixes for your domain name as well. This enables your client to prevent competitors from registering your client's domain name with other suffixes. It's also good practice to register misspellings of the chosen domain name so parasites won't create a domain with similar spellings to ride on the coattails of your client's successful site.



Just because a domain name has been registered doesn't mean that it's being used. Many domain name brokers buy names on speculation. Unused domain names can be purchased. You might be able to purchase your desired, but currently registered, domain name from a company such as BuyDomains.com or Afternic.com.

Choosing an e-commerce-friendly server

Web hosting services are a dime a dozen these days, or so it seems. Choosing the right server can mean the difference between a successful e-commerce site and one that fades into oblivion after a few hundred visitors. Web hosting services offer a variety of features. The following sections offer some sage advice for choosing a Web hosting service for your e-commerce site.

To share or not to share

Most Web hosting companies offer shared and dedicated hosting. The option you choose depends on your client's budget and how much traffic you anticipate will be generated for the e-commerce site you're designing. Of course, when it comes to traffic, your client will want the moon, Alice.

If your client is just dipping his toe into the e-commerce tidal pool, a shared server suffices nicely. When a *shared server* hosts a site, the site information is stored on a host server with several other Web sites. Shared servers are, understandably, more economical than dedicated servers. The only potential drawback with a shared server is that if your client's site (or one of the other sites on the server) starts getting a lot of traffic, visitors might not be able to access your site quickly during busy times. Or — the worst-case scenario — visitors might not be able to access the site 100 percent of the time. Another thing to consider is the speed of the server. If the shared server has a T1 line, though, it might work just fine. A *T1 line* is fiber optic and enables a fast data transfer rate.

If your client has an established, bricks-and-mortar business that's doing well and you have a stout plan for marketing the site (you do have a marketing plan, don't you?), consider paying the extra to get a dedicated server. When you host a site on a *dedicated server*, the server is dedicated to your client's site. That's right: Your client's site is the only one on the server, which means visitors get faster access — even when the Internet is busy — and they get access 100 percent of the time.



If your client's initial needs warrant using only a shared server, make sure that the Web hosting company can upgrade the client to a dedicated server without penalty when the need arises.

Getting the best deal

When searching for a service to host an e-commerce site, the best price is not necessarily the best deal. In some instances, the best price is an absolute disaster. As the Web designer, you need to be in control and tell your client which server you're going to use. So what features do you need? That, like the proverbial well, is a deep subject. However, here are a few items to consider when shopping for the quintessential, e-commerce, Web hosting service:

- ◆ **Does the Web hosting company provide shared servers or dedicated servers?** If the company provides a shared server, visitors might experience problems connecting to the Web site during peak traffic periods. Not good.
- ◆ **How much hard disk space does the Web hosting service provide?** This usually differs depending on the plan you choose. If the site you're designing is relatively small, you can get by with 300–500 megabytes (MB) of hard disk space. If the site you're designing has copious amounts of multimedia elements such as full-motion video or Flash movies, subdomains, and other bells and whistles, you might need up to 9 gigabytes (GB) of space. Choose the amount of space that best serves your client's current needs. Just make sure that the hosting service is flexible and that your client can upgrade to a different plan that provides more disk space when the need arises.
- ◆ **What is the maximum bandwidth the service provides?** For a small site, bandwidth probably won't be an issue. If your client is going big time with lots of information and lots of pages, and has a service that's in demand causing people to flock to the site in droves, her site will gobble up bandwidth quicker than Donald Trump adds to his considerable net worth. If your client throws in a podcast, you're talking about gobs of bandwidth. The good news is that Web hosting is an extremely competitive business, so most companies offer packages with 15,000GB of data transfer. Now that's a whole lotta bandwidth. If the mega bandwidth comes on a shared server package, make sure the server is lightning fast.



When you're considering a hosting company, ask the salesperson for the URLs of sites they host. Check with the site Webmasters to see whether they've encountered any problems, such as down time or slow connections.

- ◆ **What percentage of up time does the Web hosting service guarantee?** E-commerce sites need to be available 24/7. In an ideal world, there would never be power outages or problems with Internet hub connections. However, these interruptions in service are facts of life, and the best Web hosting services have redundancy built into their system to guard against down time. Choose a Web server that can guarantee you near 100 percent availability.
- ◆ **How many e-mail accounts does the Web hosting service provide?** Most Web hosting services provide 50 e-mail accounts with a hosting package. Having an e-mail account linked to a Web site adds an air of authenticity to the site. Most consumers would rather send a query to, say, an e-mail address like `info@widgetzrus.com` than to `fred@aol.com`. With 50 e-mail addresses that have your domain name, you can use generic e-mail addresses, such as `info@mysite.com` and `sales@mysite.com` — and, if needed, personal e-mail addresses such as `fred@mysite.com`.
- ◆ **Does the Web hosting service provide a user-friendly method of managing the Web site?** Most Web hosting services provide a control panel that clients use to manage the Web site, add e-mail addresses, manage databases, control FTP access to the site, and so on. If you're going to manage the account for the client, make sure you understand how to use the control panel. If your client is going to manage the site, make sure the control panel is extremely user friendly.
- ◆ **Does the Web hosting service provide support? If so, is the support by phone or online?** You need to know whether support is available 24/7 in case of an emergency.
- ◆ **Can you set up a database for the Web site?** This feature is especially important if your client requires a Web site that receives data from clients for future use. You also want to know what type of database the hosting service supports. Make sure the database is compatible with any applications you intend to use, such as a blog. If you're using a Linux server, a MySQL database can get the job done just fine.
- ◆ **Does the Web hosting package support pages with PHP code?** If so, make sure the version of PHP offered by the Web hosting service matches the version you'll be using for your code.
- ◆ **Does the Web hosting service provide Web analytics and statistics?** If so, how much information is provided? You or your client needs to know how many visits the site is getting each day, where the traffic is coming from, what keywords are being used to find your site from search engines, what the most popular entrance and exit pages are, and so on. This information is imperative when you and the client fine-tune the content of the site, or when redesigning it in the future.

- ◆ **Can you host more than one domain with the hosting package?** This information is important if the client is setting up more than one e-commerce site. Many Web hosting services enable you to host up to ten domains with one hosting package at a reasonable cost.
- ◆ **Does the hosting service enable you to create subdomains?** This feature is useful if the client wants to set up different Web sites for different aspects of the company's services. For example, you could design pages for the client's customer service and store them in a subdomain that you might name `customerservice.mye-commerce.com`.



As a Web designer, you can opt for hosting your own Web site with a Web hosting package that permits more than one domain. You can then host your client's Web sites using your Web hosting service and bill the client for the hosting service, enabling you to add additional profit to your bottom line. Many Web hosting services also have reseller packages. You can purchase Web hosting for additional domains at a reduced price. You then mark up the hosting service and bill your client.

Planning a user-friendly site

Before you begin designing the site, sit down with your client and find out what his goals are for the site. If the client wants to use the site to dispense information and collect sales leads, you need to create an online catalog and a database that records the names and e-mail addresses submitted by interested parties. You can collect this information, using an online form.

If the site is used to sell products and your client wants visitors to be able to purchase products online, you need additional information to create a successful site. Here are a few important items that must be included when designing an e-commerce Web site:

- ◆ **An About Us section:** This section gives your client credibility. Here, you include information about your client and the reasons why visitors should purchase your client's products or services.
- ◆ **A comprehensive description of your client's products or services:** This section of the site should include the features of your client's offering and information on how each feature benefits the client's potential customer. If your client is selling a product or interactive service, this section should also explain the function of each feature and its benefit to the customer.
- ◆ **A section that features testimonials and/or success stories from your client's existing customer base:** This section builds tremendous credibility for your client's product or service.
- ◆ **FAQs:** An FAQ (frequently asked question) section anticipates and answers questions or concerns potential customers might have.

When you design the site, make it easy for potential customers to gather all the information they need to make an informed buying decision. Add as many graphic elements as you need to add sizzle to the steak. Images are a must when it comes to selling a product. You should include an image with each product description. If your client's catalog is extensive, you'll display many items on each page, which equates to a small image size. If your customer's products bear close scrutiny, make sure you include links to different views of the product as well as links to larger versions of the image.

If appropriate, you can also include multimedia elements in your Web design. Video clips are great ways to explain complicated details about a product or service. And you can spice up an e-commerce site with the liberal use of Flash movies. For that matter, Flash video (a proprietary codec that is decoded with the Flash Video Player) is also an excellent vehicle for incorporating full-motion video on a Web site.



Don't overload a page with multimedia elements, especially if part of your client's intended audience accesses the Internet with dialup modems. If your client insists on multimedia elements or you feel multimedia elements are essential for the site, consider creating two versions of the site: one for high bandwidth and one for low bandwidth. Make sure that the multimedia for the low bandwidth version of the site is smaller in size, with higher levels of compression, to ensure that it loads in a reasonable amount of time.

Clean navigation is essential. If you design a site with a cluttered navigation menu with multiple choices, visitors can be confused and have a hard time figuring out what to click to get the information they need. If your client has more products or information, you can divide the site into sections. Each section is like a minisite with pertinent information to a specific part of your client's business.

Brainstorming the site

When you and your client brainstorm to create the ultimate e-commerce site, your goal is to arrive at a common vision: a marriage of your creative talent and your client's vision. During your meeting of the minds, you should strive to take the upper hand. Unless your client is familiar with Web sites and e-commerce, some of her ideas might be over the top, or simply undoable. Designing an e-commerce site is covered in detail in Chapter 2 of this minibook. When initially planning the site, consider the following factors:

- ◆ **Research e-commerce sites of businesses similar to those of your clients.** In fact, ask your client for the URLs of his fiercest competitors. Most business Web sites have a common look and feel. It's generally not a good idea to design a site that's considerably different from your client's competitors' sites. Researching your client's competitors' sites gives you an idea of what are considered standard design staples for that type of business.

- ◆ **Display your client's company name and logo prominently on each page.** The easiest way to do this is to create a banner and display it at the top of each page. Figure 1-3 shows a Web site with the client name and logo prominently displayed at the top and left side of the page. The end of the banner gently curves into the main site menu, which comprises buttons with a color that was copied from the client's logo.

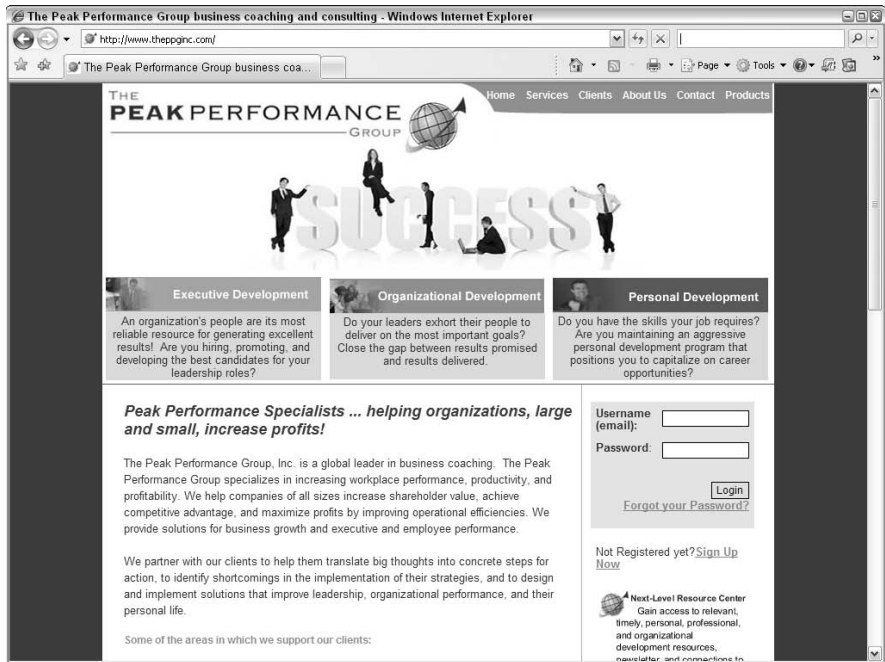


Figure 1-3:
Identifying
the site with
a banner.

- ◆ **Create a separate page for contact information.** On this page, include the client's physical address, phone number(s), fax number(s), and e-mail addresses.
- ◆ **Create a What's New section to showcase your client's newest products and other pertinent news.** This is also a great place to announce special promotions. This section should be updated frequently to encourage customers to visit the site often.
- ◆ **In addition to the main navigation menu for the site, add a text-only menu at the bottom of each page.** This information can improve the site's ranking in search engines. It also gives customers a convenient way to navigate when they scroll to the bottom of a page.
- ◆ **Add a privacy statement that clearly describes your client's policy for protecting customers' personal information.**

- ◆ **Keep your navigation simple.** Customers shouldn't have to click more than six times to go from the site's home page to checkout.
- ◆ **Don't use any trick navigation.** The navigation menu you design should be simple and should not distract the customer's attention from the reason for the site: the product or service your client is selling.
- ◆ **Use meaningful link names.** If your client suggests using obtuse (yet creative) link names, it's your job to become the voice of reason and suggest readily identifiable link names. That is, of course, unless your customer is selling an avant-garde product or service whose target audience consists of eclectic people with a creative bent.



When creating an e-commerce Web site, set up an `info@myclientsbiz.com` e-mail address. This can be a catchall e-mail for customer queries. Most Web hosting services provide more than enough e-mail addresses with each hosting package. Contact the service hosting your client's Web site for more information on setting up new e-mail accounts.

Adding Basic E-Commerce with PayPal

If your client is on a budget or isn't sure whether the product or service he is promoting will take off, PayPal is a viable alternative to hosting with a secure server and purchasing a credit card authorization package. In fact, many successful e-commerce sites use PayPal to complete transactions. PayPal is the brainchild of the folks who created eBay. Initially, PayPal was used to complete eBay auctions. The buyer could set up a free, personal PayPal account to pay the seller. Online shoppers with a personal PayPal account can pay for purchases using a debit card, credit card, or by transferring funds from the bank account associated with the PayPal account.

Online merchants can use PayPal to accept payments using either a Premier or Business account. Merchants can sign up for either account for free. The fees for transactions up to \$3,000 per month are 2.9 percent plus \$0.30 per transaction. The beauty of a PayPal account is that the merchant incurs no setup fees, no monthly maintenance fees, and no fraud-protection fees. A merchant can transfer money from his PayPal account to his bank account at any time. There is no charge for a wire transfer, which takes three to four days, depending on the bank account routing.

When you use PayPal to accept payments, you can create a custom payment page. When you create a custom payment page, it looks like the payment is being conducted with your own secure server. To create a custom page in PayPal, follow these steps:

1. **Log on to your PayPal Premier or Business account and navigate to the Profile page.**

The Profile section of the PayPal Web site is where you specify your account information, financial information, and seller preferences.

2. From the Seller Preferences menu, choose Custom Payment Pages.

The Custom Payment Page Styles section of the site appears. The default payment style is PayPal unless you already created a custom payment page. In this case, all payment page options are displayed, as shown in Figure 1-4, and a custom style is selected.

3. To create a new custom payment page, click Add.

The Edit Custom Page Style page appears, as shown in Figure 1-5.

4. Enter a name for the custom page style.

This name is added to the list of custom styles associated with your profile.

5. Enter the URL for the custom banner that will be displayed at the top of your custom payment page in the Header Image URL field.

The ideal size for the banner is 750 pixels x 90 pixels. Ideally, the banner should contain your client's logo or the name of the e-commerce site. You can easily create an image in Fireworks that loads quickly. Unless the banner contains photorealistic images, export the image in the GIF format.



You should store the image for a custom payment page on a secure server. If you don't, a warning is displayed in the customer's Web browser notifying him that the payment page contains insecure items. The URL for a secure server begins with `https://` instead of `http://`, which is used for Web pages in which the data is not encrypted.

6. If you're not using a custom banner, enter the hexadecimal value for the header background color in the Header Background Color field.

Choose a background color that matches the background color of the e-commerce page from which the payments will be made. (For more information on hexadecimal values, see Book II, Chapter 4.)

7. Enter the hexadecimal value for the header border in the Header Border Color field.

This places a 2-pixel border with the specified color around the payment page header.

8. Enter the hexadecimal value for the background color in the Payment Flow Background Color field.

This changes the background color of the page from white to the color specified. Note that certain colors, such as red and fluorescent green, are not permitted because these colors clash with the payment page warning messages, making them hard to read. Of course, don't choose a dark color because the text will be illegible as well.

Figure 1-4:
The Custom Payment Page Styles section of the PayPal Web site.

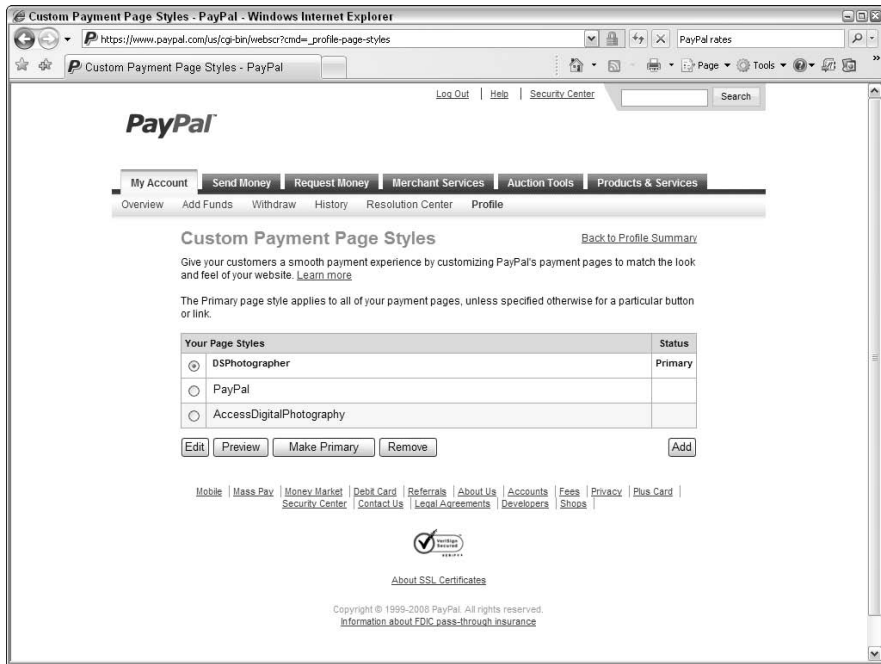
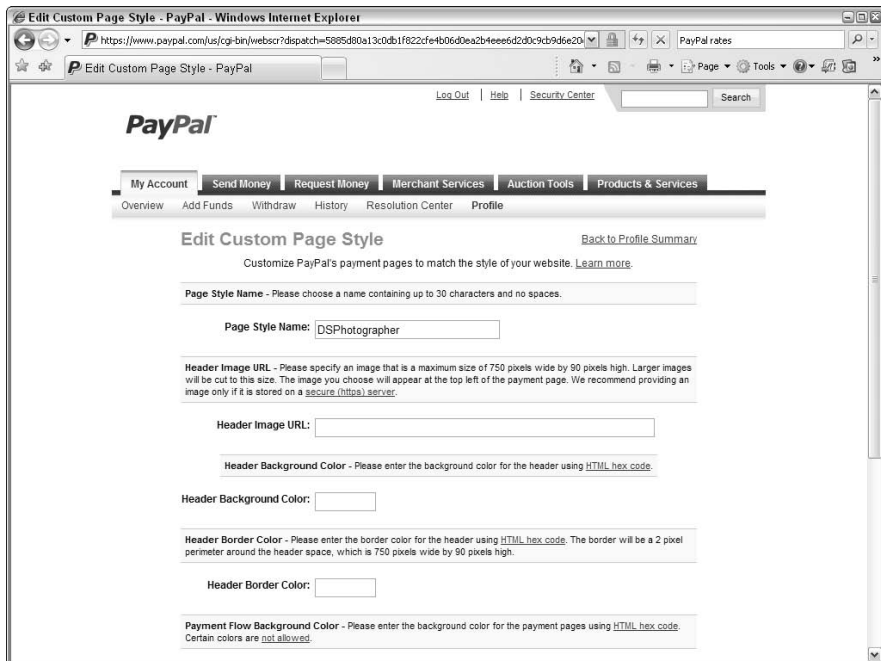


Figure 1-5:
Create a custom payment style.



9. Click Save to save the custom style and return to the Custom Payment Page Styles page.

Alternatively, you can click Preview to preview the custom style. If the style isn't to your liking, change the parameters until the custom style matches the look and feel of the e-commerce site from which the payments are being made.

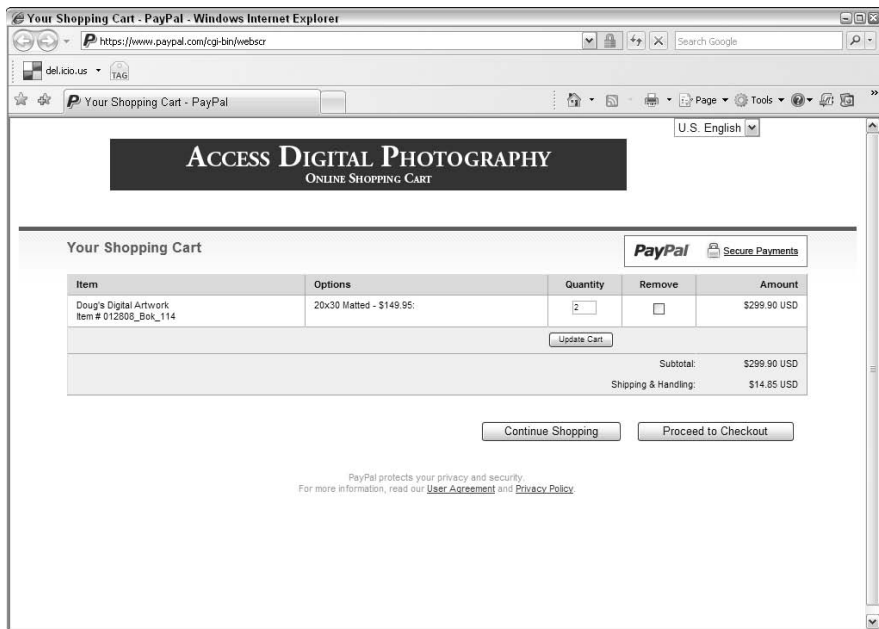
10. Select the radio button for the style you just created.

This makes the selected style the style that displays when payments are made from the e-commerce site.

11. Click the Make Primary button.

The style you created is now the default style for any PayPal transactions. Figure 1-6 shows a custom PayPal payment page.

Figure 1-6:
A custom PayPal payment page makes it appear as though the payment is being made through the site's secure server, and not through PayPal.



Using PayPal Website Payments Pro

PayPal Website Payments Pro is a package designed for e-commerce Web sites. The application integrates with existing shopping cards using a developer API. It enables you to take Visa, MasterCard, American Express, and Discover credit cards as forms of payment. In addition, a shopping cart powered with PayPal Website Payments Pro can also accept payments from

debit cards, e-checks, bank transfers, and PayPal balances. If PayPal Website Payments Pro is used to handle online transactions, your client won't need another merchant account. Here are some added benefits to using PayPal Website Payments Pro:

- ◆ No application or setup fees
- ◆ No cancellation fees
- ◆ No monthly minimum
- ◆ No reporting fees
- ◆ No gateway fees
- ◆ No long-term commitment

When a visitor uses one of these credit cards to make a purchase through a Web site using PayPal Website Payments Pro, the payment is processed in the background. Payments made using PayPal express checkout transport the visitor to PayPal's secure Web site. The landing page can be modified to match the look and feel of your client's Web site. Checkout is an innocuous three-click experience. And the money from the transaction is immediately deposited to your client's account. (Sweet.) PayPal also provides fraud protection. You can set a limit on the maximum purchase amount, country of origin, and so on. As of this writing, Website Payments Pro pricing is as shown in Table 1-1 (all amounts in U.S. dollars):

<i>Monthly Sales</i>	<i>Per Transaction</i>	<i>Per Month (Unlimited Transactions)</i>
\$0.00 to \$3,000	2.9% + \$0.30	\$30.00
\$3000.01 to \$10,000	2.5% + \$0.30	\$30.00
> \$10,000.01	2.2% + \$0.30	\$30.00
> \$100,000	1.9% + \$0.30	\$30.00

Integrating with Credit Card Authorization Packages

The traditional way to handle transactions from a Web site is using a credit card authorization package. When you integrate a site with a credit card package, the payments from the site are handled by a number of banking institutions and processes. The service charges a setup fee and a monthly maintenance fee. The maintenance fee is based on the number of transactions handled per month. A thorough review of each of the many authorization packages available is beyond the scope of this book.

Credit card authorization packages enable you to run your online business with minimum overhead. In a traditional bricks-and-mortar store, you have to hire salespeople to show customers the product, take orders, and accept payments. With a properly designed e-commerce site and credit card authorization package, these tasks are handled automatically. If the e-commerce site you're designing sells a service, you can schedule regular payments through the credit card authorization package.

Online payments are quite involved, flowing through a complex network of financial institutions and processes. The complexity is exacerbated when you accept payment from foreign countries. Fortunately, technology has simplified this process, and the proper credit card authorization package provides the gateway for this process. When a customer makes an online payment, the process occurs in the background, and the payment process occurs fairly quickly.

Solving the online payment maze

When a customer makes a payment, it's submitted through a gateway. Then the payment must go from the *issuing bank* (the institution that funds the customer's credit card) to your client's bank. The process appears seamless when a payment is made online, but there are many supporting players in what seems like a one-act play. The following list introduces the various actors in this one-act play that happens when the Checkout button is clicked:

Merchant's Online Business: The customer makes a purchase and clicks the Checkout button.

Internet Gateway: The merchant's secure Web site submits a credit card payment to the payment gateway.

Merchant's Bank Processor: The gateway provider receives the secure transaction information and hands it off to the merchant's bank processor.

Credit Card Processor Interchange: The merchant's bank processor submits the transaction information to the *credit card processor interchange*, which is a network of financial entities that manages processing, clearing, and settlement of credit card transactions.

Credit Card Issuer: The transaction is handed off to the customer's credit card issuer, which approves or declines the transaction based on the customer's available funds. If the transaction is approved, the appropriate funds are sent back through the credit card processor interchange.

Credit Card Processor Interchange: The credit card processor exchange relays the information to the merchant's bank processor.

Merchant's Bank Processor: The merchant's bank processor relays the information to the Internet gateway.

Internet Gateway: The Internet gateway stores the result and funnels the information to the merchant. The process from checkout to sending the information back to the merchant averages less than three seconds.

Credit Card Processor Interchange: The funds, less any processing fees, are transferred to the merchant's bank account — a process that takes from two to four days.

Fees for credit card authorization packages vary, depending on the institution you choose and the number of bells and whistles included with the chosen credit card authorization package.

Internet fraud: An e-commerce merchant's worst nightmare

Not unlike things that go bump in the night, Internet fraud can rear its ugly head when you least expect it. Online merchants are responsible for fraudulent credit card transactions conducted through their sites. In addition to incurring heavy penalties and fees from the credit card association, there is the matter of product costs and shipping fees incurred by your client. Fortunately, you can protect the site against credit card fraud, even with a start-up online business that has limited transactions.

The first and most obvious step in safeguarding against fraud is to choose a secure and reliable payment solution. Choose a payment package that includes standard processing and anti-fraud features such as Card Security Code (CSC) and Address Verification Service (AVS). In addition, the payment solution should have options that enable your client to upgrade to the new buyer verification systems such as MasterCard SecureCode or Verified by Visa. Your payment solution might have additional options to safeguard against fraud. Ask your representative for details.

SSL — What Is It?

The world is full of acronyms these days, and the wonderful world of e-commerce is no exception. But remember, if it weren't for acronyms, you'd have to type or write a whole lot of words. The acronym of concern for any e-commerce site is SSL, which stands for *Secure Sockets Layer*. See, we told you it was a lot of words.

In a nutshell, when a site has SSL, the transaction is encrypted and cannot be deciphered by a third party. All secure pages are listed as `https`, followed by the rest of the Web address. Users can transmit any amount of information from a secure site and know that a third party cannot decipher the information. Sending credit card information via a secure site is safer than whipping out your credit card in a local store where prying eyes — and for that matter, the salesperson — can see the information. When you

purchase goods or services from a secure site, the data is *encrypted* when submitted. When the recipient (the bean counter for your client's e-commerce site) receives the data, it's *decrypted*. The bean counter adds the money to your client's P&L (profit and loss) statement, and the goods are shipped to your client's customer.

Using a secure server

When you conduct your e-commerce using a secure server, the SSL certificate is linked to the site from which the transactions are being conducted. The SSL certificate is in your client's name and in the domain name of the e-commerce site.

Hosting an e-commerce site through a secure server is an expensive proposition. To host an e-commerce site on a secure server using an SSL certificate with 128-bit encryption costs several hundred dollars per year — and as much as one thousand dollars and up per year depending on the services included with the package. However, the less-expensive alternative of using PayPal's secure server is always an option. You might pay a bit more per transaction, but the cost of a secure server isn't added to your overhead. The other alternative is sharing an SSL certificate, if your Web hosting service provides this option.

Sharing an SSL certificate

Many Web hosting services allow their customers to share an SSL certificate. This service is included with your hosting, and therefore can be considered free. When you share an SSL certificate, you do so through a third-level domain alias, for example: `www.yourdomain.c2.yourWebhostingserver.com`. When you share an SSL certificate, you must use the domain alias in the code you use to create the buttons for your site's shopping cart. Alternatively, you can purchase a shopping cart package, which you can integrate into the site.

The problem with sharing an SSL certificate is that Internet Explorer issues a warning saying the domain name on the certificate doesn't match the domain name of the site from which the transaction is being conducted. Even so, the data is still encrypted, and the transaction is secure. This might cause a customer to back out of a transaction.

E-Commerce Do's and Don'ts

When you create an e-commerce site, your goal is to sell your merchandise — or your client's merchandise. In keeping with these goals, there are certain things you should consider as well as certain things you should avoid (such as designing an e-commerce site that looks like a board game). The following list of do's and don'ts can keep you on the straight and narrow:

- ◆ **Do make the site user-friendly.** Make sure your site navigation is easy to decipher. If your visitors need a manual to figure out how to use your site, it's not a good thing.
- ◆ **Do include a privacy statement.** If you request the visitor's contact information, make sure you include a link to your privacy statement that is readily visible on any page that requests sensitive information.
- ◆ **Make sure the site has the look and feel of other e-commerce sites in your client's industry.** Customers shy away from something that looks different from what they've come to expect. After all, no department stores look like the hip boutiques in Haight-Ashbury.
- ◆ **Avoid using buzzwords and hype, such as *best price* or *highest quality*, in the product descriptions.** These raise a red flag with many buyers.
- ◆ **Don't write the content.** Your client knows more about his business and product than you could ever hope to. After all, you wouldn't have your client write the content for your Web site, would you?
- ◆ **Spice up the pages.** Many Web sites have too much text on the page, while others rely solely on graphics. A good design has a nice mix of text and graphics.
- ◆ **Use Flash content judiciously.** If you need to add some razzle-dazzle, Flash can provide it for you, but don't make your site exclusively Flash. Search engines tend to avoid Flash content like the plague. However, you can safely add a small Flash animation to an otherwise HTML page.
- ◆ **Keep it fresh.** Make sure your client understands that an e-commerce site needs to be updated frequently in order to ensure return visitors. Either negotiate a fee for periodic revisions up front, or design the site in such a manner that your client can update it with Contribute. If you choose the latter, be sure to add extra for Contribute tutoring and the inevitable calls for advice. For more information on Contribute, see Book VIII, Chapter 2. And read all about it in *Macromedia Contribute For Dummies*, by Janine Warner and Frank Vera (Wiley).
- ◆ **Don't clutter the home page.** Some e-commerce sites look like the front page of the daily news and are way too busy to be useful. The home page of the e-commerce site should be like the cover of a book: inviting and a reason for the visitor to click a few links to see what your client has to offer.
- ◆ **Give visitors a reason to return.** Set up the site with a news section, or create a small section of the home page devoted to new products or information. Make sure this content is updated at least twice per month.
- ◆ **Don't create content that disappoints the visitors.** If your client insists on a section that promises response, such as a section that accepts customer comments, make sure your client is aware that he must live up to the promise and provide feedback.

Pumping it up with eBay

eBay is without a doubt the largest online merchant in the world. Thousands of people use eBay to auction off their unwanted toys, digital cameras, musical instruments, and so on. One man's trash is another's treasure. A NASCAR driver used eBay to auction off a helmet he had thrown at a competitor's car after being forced off the track. The proceeds went to charity. In fact, a section of eBay is devoted to selling cars that sell anywhere from a few hundred dollars to several hundred-thousand dollars.

Savvy owners of e-commerce Web sites duplicate their efforts on eBay by setting up an eBay

store. The fees for selling and setting up an eBay store are reasonable. Many bricks-and-mortar businesses that also have Web sites set up eBay stores. To drive traffic to their eBay stores, they auction off popular, fast-selling items. The eBay store features items other than those being offered for auction. Once the vendor starts racking up sales on eBay, they can add the buyers to their e-mail list and send them news about specials from their e-commerce Web site. The figure shows an eBay store run by the owners of a successful bricks-and-mortar business, who also run a successful e-commerce Web site. There's profit in redundancy.

The screenshot shows an eBay store page for 'BACKDROP OUTLET' in a Windows Internet Explorer browser. The page title is 'eBay Store - BACKDROP OUTLET: Fantasy Cloth, Props, Mottled Muslin'. The browser address bar shows 'http://stores.ebay.com/BACKDROP-OUTLET'. The page features a navigation menu with 'Home', 'eBay Stores', and 'BACKDROP OUTLET > All Categories'. A banner at the top reads 'Supplying backgrounds, props & studio accessories' and 'BACKDROP OUTLET at the lowest prices for 15 years.' Below the banner, a text block states: 'BACKDROP OUTLET IS YOUR COMPLETE SOURCE FOR ALL YOUR BACKGROUND AND STUDIO SUPPLY NEEDS... BE SURE TO CALL FOR OUR HUGE CATALOG FOR GREAT POSING AND LIGHTING IDEAS OVER 130 COLOR PAGES! 800-466-1755'. A promotional message says 'OUR EBAY STORE OFFERS 50-70% OFF OUR NORMAL CATALOG AND WEB SITE PRICES!'. A call to action asks users to click on the 'ADD TO MY FAVORITE STORES' link. The main content area shows a search bar with '800 items found in All Categories' and a list of items for sale. The items listed include 'MIRRORED PLEXIGLASS PHOTO PROP BRAND 24"x48"', 'GIANT CRAYON PHOTO PROP/BANK GREEN backdrop outlet', 'SHELL PHOTO PROP MERMAID COSTUME & FC BACKDROPS', and '9X20 MOTTLED FC BACKDROP/BACKGROUND PHOTO EXFC63'. The page footer shows 'Done' and 'Internet'.

Chapter 2: Building an E-Commerce Site

In This Chapter

- ✓ **Coping with technology**
- ✓ **Dealing with the law and privacy**
- ✓ **Using Web hosting and payment system tools**
- ✓ **Getting the word out**
- ✓ **Protecting your site**

All the good intentions in the world won't get the job done, especially when you're creating an e-commerce Web site. For an e-commerce site to be successful, it has to be usable. If potential customers can't navigate the site without a manual thick enough to be a doorstop, they'll never come back. In addition to being usable, the site needs technology in order to perform tasks such as capturing the e-mail addresses of customers for a mailing list, taking orders online, and so on. And of course, any time you're doing business, you have legal considerations. Doing business online has its own grab-bag of legal issues — and you thought you were just designing another pretty site on the 'Net. In this chapter, we show you the technological and legal considerations involved with an e-commerce site as well as how to build some of the elements for your e-commerce site.

Technological Considerations

Technology is everywhere in some shape or form. When it comes to creating an e-commerce site, consider several technological issues. We cover some of the technological issues concerning e-commerce — such as secure servers and credit card authorization — in Chapter 1 of this minibook. However, additional technological issues are involved with building the site.

The technology you use depends on the needs of your client. Perhaps your client requires a form that collects information from customers. You could use a CGI (common gateway interface) mail-forwarding script to transmit the data from customer to e-commerce client via e-mail. However, a technologically advanced method for doing this is to create PHP code that transmits the information to a database.

If you're like many Web designers, the thought of creating code causes you to break out in a cold sweat. If this is the case, you need to form an alliance with a Web developer who is as happy as a clam in saltwater when writing complex code. Like most human beings, Web developers come in varying forms of initiative, skill, and honesty. Make sure you're working with a developer who will stay the course and not turn tail at the first sign of a bug in his code. You'll also have to pad your price to the client to include the Web developer's fee and whatever profit you can reasonably add to his fee.

After you take care of the technological issues on the design side, you might think it safe to jump off the deep end and begin designing the site. Nay, nay, Nanette. Did you ever hear of *server side*? The technology you use to create your Web site must be present on the server side. You must address these considerations when you choose a hosting service for the e-commerce site. Most standard Web hosting packages include the latest version of PHP and the ability to create one or more MySQL databases. These are generally handled with a Linux-based server. However, if your site requires ColdFusion, ASP, MS SQL databases, or MS Access databases, you need to opt for a Windows NT server, which is generally more expensive.

Usability: Thinking Like a Customer

The customer is always the customer. But then again, without the customer, the snazzy e-commerce site you're designing for your client would be absolutely worthless. The trick here is to get yourself and your client into a customer frame of mind. How many Web sites have you and your client gone to and clicked out of almost immediately? Probably quite a few. When you and your client are brainstorming, bring up the topic of Web sites you absolutely refuse to visit again. Then jot down the reasons why you'd never return to them. Now that you know what you don't like, consider the following usability issues:

- ◆ **Can I easily find my way around the site?**
- ◆ **How many clicks does it take to go from the home page to checkout?**
- ◆ **Can I easily get back to where I've been?**
- ◆ **Is the navigation menu consistent on all pages?**
- ◆ **Do all pages on the site have a consistent look and feel?**
- ◆ **When I get to the bottom of the page, can I easily navigate to another page?** If the answer to this question is no, create text links at the bottom of the page that contain all the links in your navigation menu. The text links also help your rating with search engines.
- ◆ **Does each page have a descriptive title?** You wouldn't open a book in a bookstore if it didn't have a descriptive title on the cover, would you? Titles are also used by search engines to rank Web sites.



You can easily add a title to a Web page by changing the page properties in an application such as Dreamweaver.

- ◆ **Does each page have a link to the site's home page?** Many designers add a hyperlink to an image such as the company's logo and place it in the upper-left corner of the page. This is considered good practice, but Web neophytes might not be hip to this trick. Always include a navigation menu link that clearly indicates that clicking it returns the customer to the site's home page.
- ◆ **Does the site have alternative navigation?** If you're designing a large site with hundreds of pages, create a text-based site map. Include a link to the site map on each page. As an added bonus, with a site map, search engine robots have easy access to all of the site's pages.
- ◆ **Does the site have a search field?** Adding a search field makes it easy for customers to find the information they're after. You can find inexpensive — or perhaps even, free — search utilities by typing **Web site search utility** in your favorite search engine. Weed through the results to find the best solution for the site you're designing.
- ◆ **Is the most important content of each page clearly visible?** Think like a newspaper editor and put the most important content of each page at the top of that page. You can further draw attention to pertinent content through the use of a header style, which functions just like a newspaper's headline.



You can customize heading styles with a CSS (Cascading Style Sheet). See Book III, Chapter 3 for more about CSS.

- ◆ **Can a page be summed up in a glance?** Customers should be able to quickly scan each page and get an idea of the content contained within. You can use header styles, bold text (which can be created as a style, using CSS), bullet lists, or graphics to emphasize the important content on each page.
- ◆ **Is text included in an image?** Text in an image won't be visible if customers disable images in their browsers, or if visually challenged customers access the site with a screen reader. If text is needed to describe the contents of an image, create the text as a caption when designing the page in your HTML editor.
- ◆ **Does each <image> tag specify alt text?** When you add *alt text* to an `<image>` tag, the tags show up as tooltips in many browsers. An added benefit is increased visibility to search engines. Alt text is generally used to describe the image for visually impaired viewers accessing the site with a screen reader. However, you can describe the image using keywords that relate to the site.
- ◆ **Does the site enable visitors to provide feedback?** If not, consider adding feedback forms in applicable sections of your site — pages such as Contact Us or Feedback. You might also consider creating a

Feedback Forum where customers can post feedback and read feedback from other customers. The type of feedback you're looking for includes whether visitors enjoyed the site, would visit a similar site, understood the site content, were easily able to navigate the site, and so on.

- ◆ **Can your client's target audience easily understand the site content?**
If your client provides content for the site and uses technical jargon and hard-to-understand terms, become the voice of reason and tell your client to "dummy it down." Pun intended.

Legal Considerations: Call a Lawyer

When you create an e-commerce site for a client, the client is liable for the content of the site. Of course, your name is also linked to the site. Therefore, you have two concerns in that regard: your reputation and your client's reputation. One of the most important things to consider is copyright law. Don't use anything on an e-commerce site — or for that matter, any Web site — for which you or your client don't own the copyright or for which you or your client have not purchased a license. This includes photos, videos, and music. In the long run, the only way to be positively sure is to run everything past your client's legal counsel.

Other legal concerns involve the privacy of Web site customers. Every commercial Web site that collects information from customers should have a privacy statement. Then there's the issue of cookies — the software kind, not the ones that cause love handles. Speaking of edible (well, almost edible) objects that have become part of Internet lingo, there's spam. Cookies and spam, which won't be found on any menus other than at the Internet Café, are covered in upcoming sections.

Cookies that don't crumble

Cookies are small text files that are automatically downloaded to Web site visitors' computers. Cookies are used to track Web site visitors' habits, such as what sites they visit, how they found the site that issued the cookie, what pages they navigate to, and so on. They're a useful tool that shows e-commerce marketing gurus which sections of their sites are the most popular, which need updating, where Web site traffic is coming from, and so on. Cookies are also used to save passwords, site preferences, and so on. Any Web site that uses cookies must notify visitors in some shape or form that this software is being downloaded to the user's computer. Failure to do so is an invasion of the visitor's privacy.

Spam, spam, spam, spam!

Marketing a Web site is a whole different kettle of fish. One tried-and-true method of marketing used by most e-commerce sites is a mailing list. With a mailing list, e-commerce marketers can inform customers about new

products and special offers. You can send this information in the form of a classy HTML newsletter, such as the one shown in Figure 2-1. However, commercial e-mail messages can be sent only to people who *opt-in*. (In other words, the customer requests the information.) Previously, marketers could send the messages to anyone as long as the recipients had a means of *opting-out* (making the messages stop coming). If an unsolicited e-mail message is sent and that message presents a commercial advertisement, an offer to purchase a product or service, or a link to an offer to purchase a product or service, it is considered *spam*, and it is not wonderful.

The CAN-SPAM (Controlling the Assault of Non-Solicited Pornography and Marketing) Act of 2003 is an attempt to control spam. Many e-commerce marketers mistakenly think that because they purchase a list with hundreds or thousands of e-mail addresses of likely prospects, they can barrage their inboxes with all manner of advertisements and such, which contain deceiving subject lines or false headers. Sending messages of this type to multiple recipients (100 e-mails in a 24-hour period; 1,000 e-mails in a 30-day period; or 10,000 e-mails in a one-year period) is a felony. The CAN-SPAM act also regulates what merchants need to do when sending messages to recipients who have elected to receive commercial e-mails from the merchant. The message header must clearly indicate that the message is a solicitation. There must be a valid e-mail address in the message header, and there must be a means by which the recipient can unsubscribe from the list. In February 2004, the first violator of the act was arrested.



Creating HTML newsletters and advertisements can be a lucrative business for Web designers. If you decide to do this type of work for your client, you need to create an unsubscribe link in the messages. It's also a good idea to include a link to the merchant's privacy notice, plus a link to the merchant's legal notice if he has one.



Figure 2-1:
HTML newsletters requested by the customer are not spam.



All things being equal, people want to – no, scratch that – *love* to do business with their friends. All things being unequal, people will do business with their friends before they'll do business with a stranger!

"Power Pitch"
How to Develop the 30 Second Commercial that Wins Interest and Business!
Are you getting the response you need from your elevator pitch? Is it generating an emotional connection from your audience and producing follow-up interest in your business?

Creating an E-Commerce Site

After you brainstorm with the client to get his grand vision and consider usability and technology, you're ready to begin designing the site. Setting up an e-commerce site isn't rocket science, but keep in mind that some design elements are critical and unique to e-commerce sites. For example, commercial Web sites need e-mail addresses that are associated with the domain. An e-mail address with an actual domain name other than a generic ISP service, such as AOL, lends credibility to the e-commerce business. And of course, there's the matter of how to pay for the goods. In the upcoming sections, we show you how to add these components to your e-commerce design and more.

Setting up the e-mail addresses

Most visitors to e-commerce sites don't feel comfortable sending an e-mail to an address like `fredswidgets@aol.com`. It just doesn't seem very professional to have an e-mail address that's not associated with the domain from which the goods are being sold. In most instances, you set up the e-mail addresses through the Web hosting service's control panel. Each Web hosting service is different, but the following shows an example of how this works.

1. Log in to your Web hosting service's control panel.

This is a secure part of the server's site. You need a user name and password to access the control panel. This information is generally included with the e-mail confirmation after you set up a hosting account. Figure 2-2 shows an example of a control panel.

2. Click the e-mail link.

This section of the control panel enables you to add and edit e-mail addresses associated with the domain. An example of the e-mail section of a control panel is shown in Figure 2-3.

3. Click the link to add a new e-mail address.

The control panel refreshes and shows the options available for setting up new e-mail accounts. You have several options from which to choose. Your best option is to set up a mailbox. Doing so means that you or your client has to configure her e-mail client to accept messages from the new mailbox.

Your other option is to forward mail from the domain to an existing e-mail address. The problem with this option occurs when your client needs to respond to a customer. That's right, the response is from the existing e-mail address, and not the new e-mail address with the client's domain name. A response from `Jillswidgets@aol.com`? Not very professional.

Figure 2-2:
You use the control panel to manage the Web site.

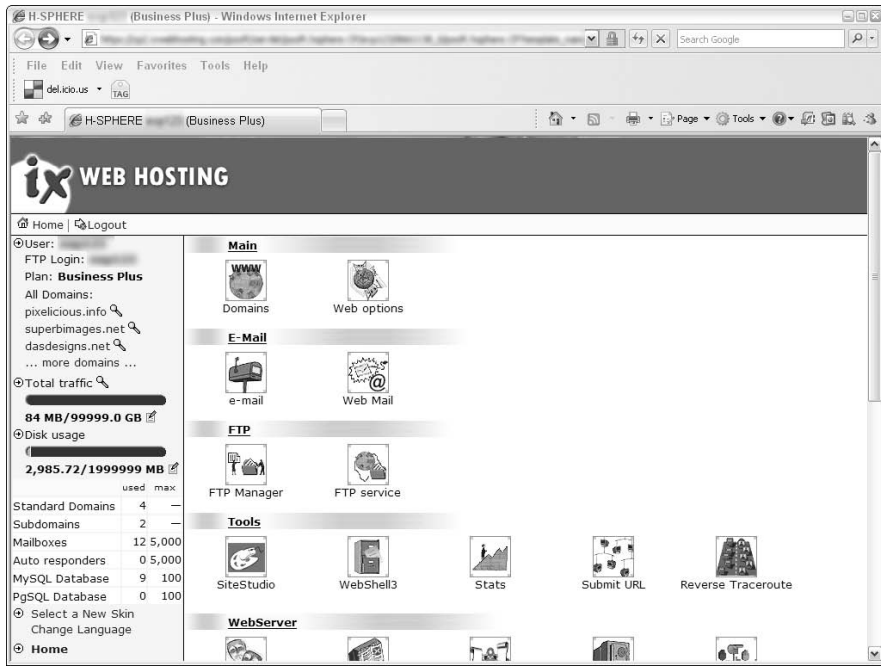
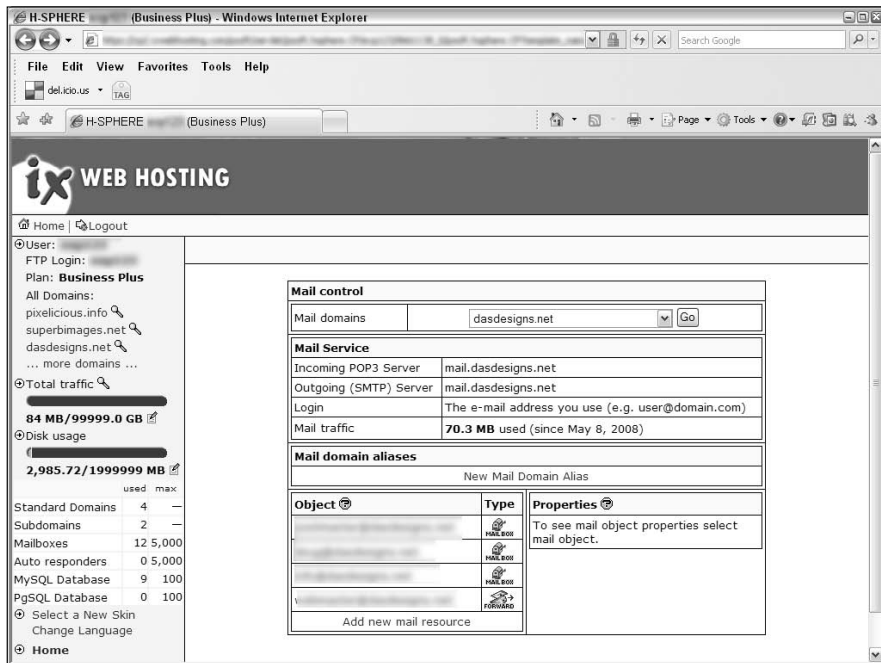


Figure 2-3:
Most hosting services enable you to configure e-mail addresses.



4. Choose the desired option.

The control panel refreshes and contains the necessary screen to set up an e-mail address and password. Figure 2-4 shows the typical screen for setting up a new e-mail address.

Figure 2-4:
Creating
a new
mailbox.

Add a new Mailbox	
Email address	<input type="text"/> @dasdesigns.net
Password	<input type="password"/> !
Comment	<input type="text"/>
	<input type="button" value="Submit"/>

5. Set up other mailboxes as needed.

Most Web hosting services enable you to set up a plethora (more than 2, fewer than 51) of mailboxes. This enables the savvy designer and client to make the company behind the Web site seem larger than it actually is. You can set up multiple mailboxes, such as *info@mywebsite.com*, *sales@mywebsite.com*, *service@mywebsite.com*, and so on. The customer never needs to know that only one person is responding to the messages.

Selling items with a PayPal account

If your client prefers to dip a toe in the e-commerce waters, she can save considerable money by accepting payments with a PayPal account. As we mention in Chapter 1 of this minibook, the rates are reasonable. All you need is a bank account and an e-mail address to set up a PayPal account. After the account is set up, you can accept payments online. Adding PayPal Add to Cart buttons to any Web page you create in Dreamweaver is a breeze, thanks to a free plug-in from WebAssist (www.webassist.com). To add PayPal Add to Cart buttons to a Web Page, do the following:

1. Navigate to www.webassist.com/professional/products/ProductDetails.asp?PID=18.

As of this writing, this is the URL from which you download the free PayPal eCommerce Toolkit 4.5.1 for Dreamweaver.

2. Download and install the plug-in.

The plug-in, *PayPal451.mxp*, is a Dreamweaver extension. By default, it's downloaded to your desktop.

3. Launch Dreamweaver and then choose **Commands** → **Manage Extensions**.

The Adobe Extension Manager CS4 dialog box appears (see Figure 2-5). This dialog box is used to manage all extensions for all Adobe CS4 applications you have installed on your computer.

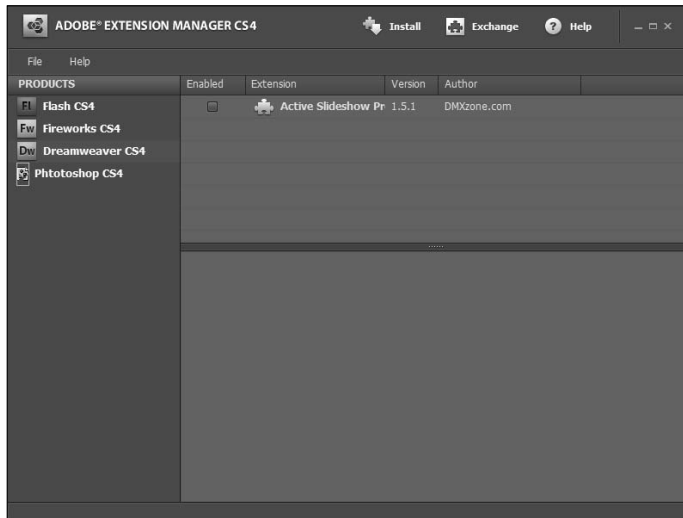


Figure 2-5: Manage extensions with the Adobe Extension Manager CS4.

4. **Click Install, navigate to and select the `PayPa1451.mxp` file, and then click Open.**

You have to accept a disclaimer that the extension is not an Adobe product and that Adobe does not warrant or support the extension. Don't be put off by this: The PayPal extension is a proven performer. Adobe is just dotting every *i* and crossing every *t*. Their lawyers wouldn't have it any other way. After you accept the disclaimer, the extension is installed.

5. **Close the Adobe Extension Manager CS4, exit, and then restart Dreamweaver.**

The extension won't work until you do.

6. **Create the page from which the customer will check out.**

You have a lot of flexibility here. Many Web designers create a page where several products are displayed along with a description. The Add to Cart button appears below the product description. Figure 2-6 shows a photographer's Web site from which images can be purchased. The portfolio consists of thumbnails that, when clicked, reveal a larger image in a separate window. Figure 2-7 shows a basic HTML page with the image. Now it's time to insert the Add to Cart buttons.

7. **Choose `Insert` ⇨ `PayPal` ⇨ `Add to Cart Button`.**

The first page of the Insert PayPal Add to Cart Button Wizard appears; see Figure 2-8.

The first time you use the PayPal extension, you have to activate it. After you download the extension, you get an e-mail receipt that has the serial number.



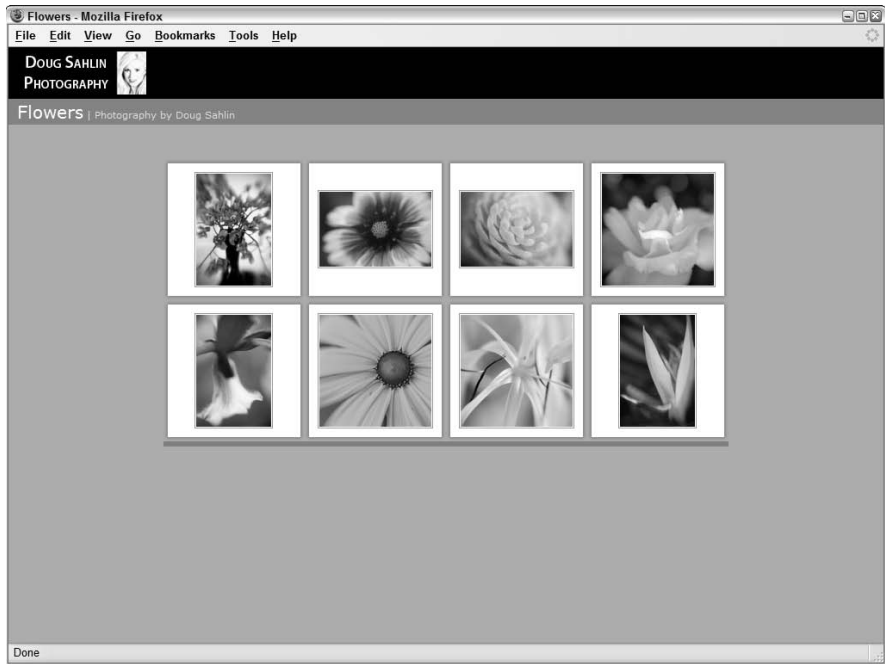


Figure 2-6:
A photo-
grapher's
portfolio
page.



Figure 2-7:
An image
that needs
an Add to
Cart button.

8. Enter the e-mail address of the PayPal account to which the sale will be credited. (See Figure 2-8.)



Figure 2-8:
Add the e-mail address for the PayPal account.

9. Click Next.

The second page of the Insert PayPal Add to Cart Button Wizard appears.

10. Choose the desired button style from the dialog box shown in Figure 2-9.

You can choose one of the preset styles, or choose a custom button. If you choose the latter option, the Button Image URL field becomes active. In this field, enter the URL where the button image is stored.



Figure 2-9:
Choose a button style.

11. Click Next

The third window of the Insert PayPal Add to Cart Button Wizard appears.

12. Enter the product information.

The product information you enter indicates which product to ship to the customer. You need to use a unique product name for each item. Select Yes or No for the options to request a shipping address from the customer and to include a comments field at checkout time. After the customer pays for the purchase, this information is sent to the e-mail address entered in Step 8. Figure 2-10 shows the dialog box for Step 3 of 5 of the Insert PayPal Add to Cart Button Wizard.

13. Click Next.

The fourth page of the Insert PayPal Add to Cart Button Wizard appears, as shown in Figure 2-11.

14. If desired, enter the URL to the success page, which usually thanks the visitor for his purchase.

This step is optional. Many designers create a Thank You page for a successful purchase.

15. Click Finish.

And that's all you need to do to add the button to one page. Figure 2-12 shows the completed page with the PayPal button added. Of course, now you have to add the button to other pages of your online store.

Figure 2-10:
Enter
product
information.

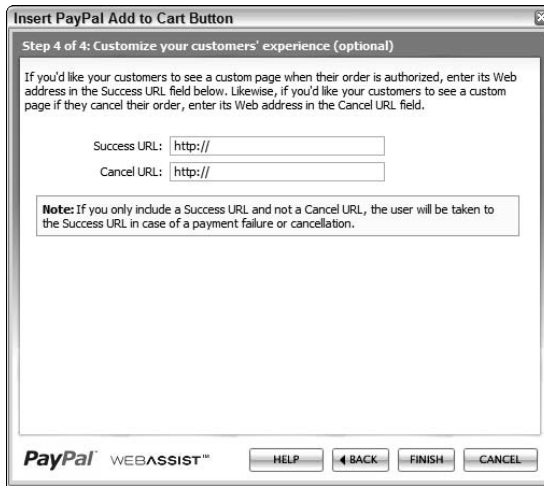


Figure 2-11: Entering the URL for a success page.

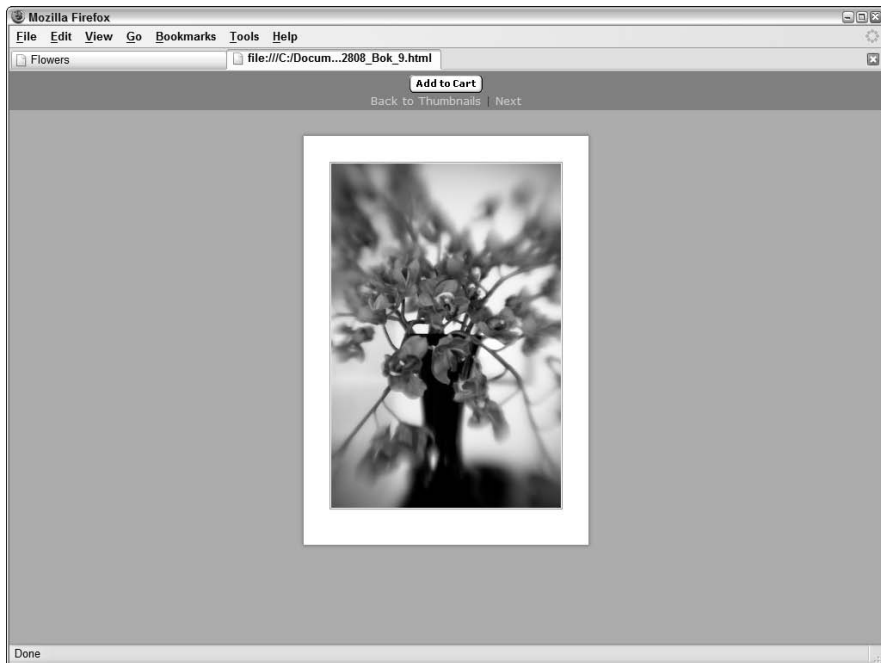


Figure 2-12: Click the button and send me the money, honey.

Using a Shopping Cart

If the site you're designing will be hosted at a secure server, integrate the items for sale with a credit card authorization package, as discussed in Chapter 1 of this minibook. However, if you attempt to integrate the items for sale with the authorization package *without* a little help from a friend, you're a card-carrying geek — also known as a Web developer. This means writing code, baby. The thought of writing code sends shivers down most Web designers' spines, but you have an alternative. Enter the shopping cart.

You can find shopping cart applications online. In fact, the service hosting the site might include a shopping cart application. Figure 2-13 shows an example of the type of features you can get with a Web hosting shopping cart application. Figure 2-14 shows a demo of an online shopping cart created using a Web hosting service's shopping cart application.

Another solution is to purchase an e-commerce shopping cart template that integrates with Web design software, such as Dreamweaver or FrontPage. A full-featured shopping cart template gives you complete control with features such as the ability to calculate shipping, integration with a product search page, the ability to add unlimited product features such as size and color, the ability to integrate with popular credit card authorization packages, and much more.

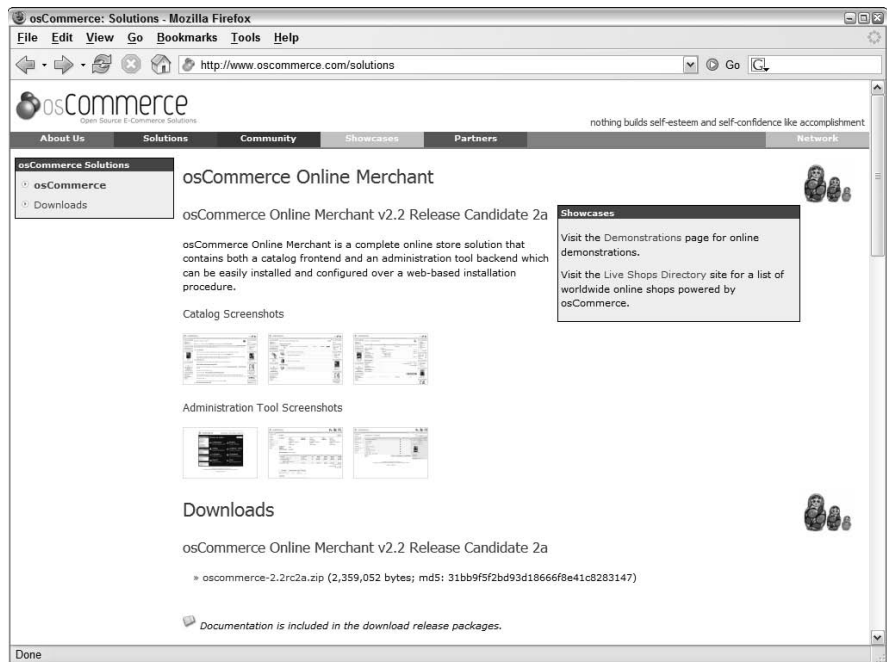


Figure 2-13:
Choosing
a shopping
cart
application.



Figure 2-14:
An online shopping cart.
Cha-ching!

You can find a plethora of shopping cart templates by going to your favorite search engine and typing the name of your Web design software followed by **shopping cart template**. When you review the results, look for a template that has all the features you need. Many templates let you design a store from scratch using a generic template that you can integrate with your existing design. These templates include the PHP or ASP code you need to create a functional shopping cart that you can integrate with a credit card authorization package. Figure 2-15 shows an e-commerce store that was created using a shopping cart template.

Creating a mailing list form

If your client wants to keep customers updated about new products, changes to the Web site, and so on, the easiest way to accomplish this is to have customers opt-in to a mailing list. You can easily do this by creating a form, such as the one in Figure 2-16.

For more information on creating forms, see Book VI, Chapter 1.

600 Using a Shopping Cart

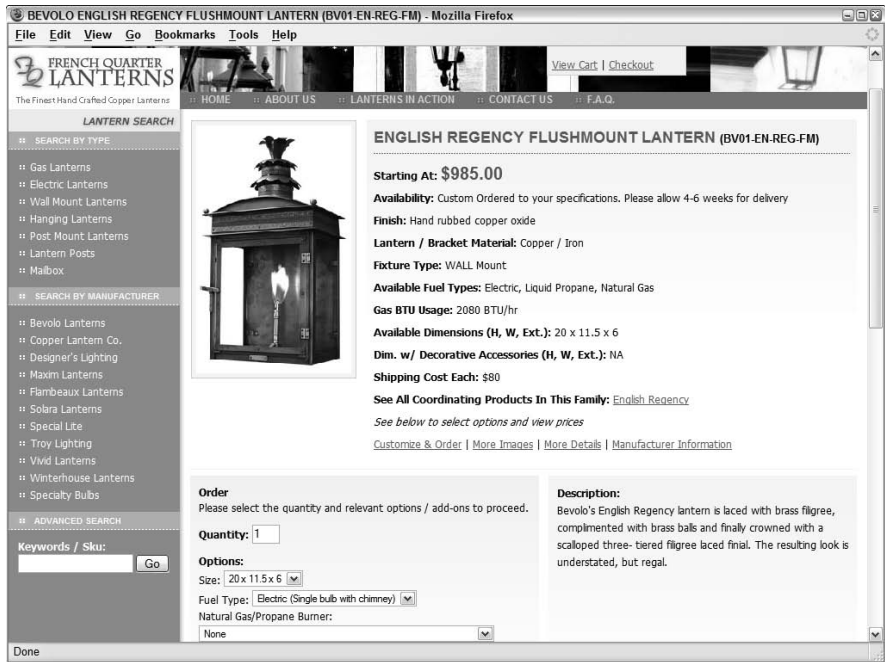


Figure 2-15:
We're only
in it for the
money.

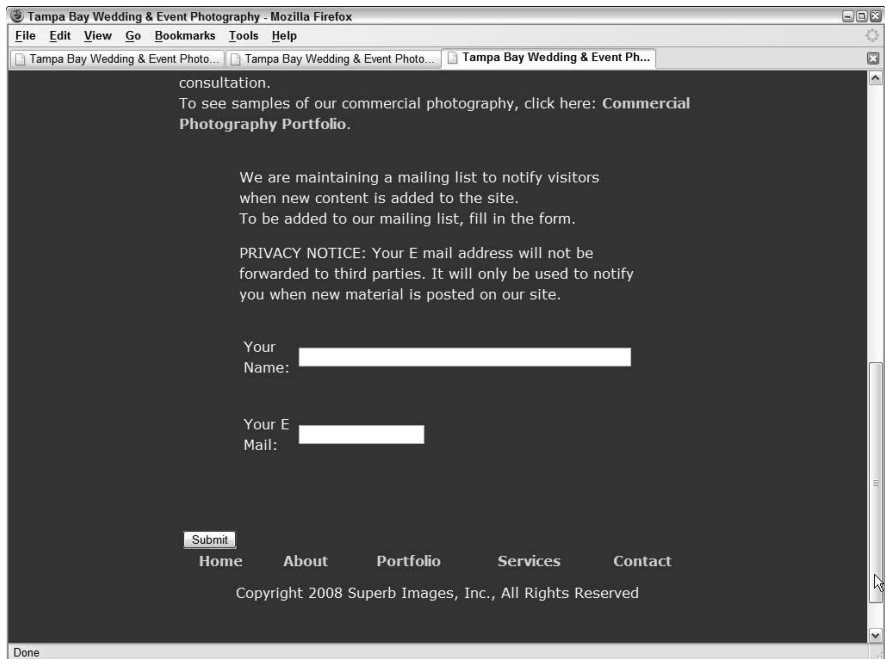


Figure 2-16:
Send
me your
answer, fill
in a form. . . .

The form itself is pretty straightforward. What makes the form tick is the form action. When the customer clicks the Submit button, a PHP page is summoned. The form action for the HTML form (see Listing 2-1) summons a PHP page that sends the form results to the database. We show you the code for the PHP page in the next section of this chapter.

Listing 2-1: Summoning the PHP Page

```
<table>
<tr>
  <td colspan="2"><form action="addRecord.php"
  method="post">
    <p>We are maintaining a mailing list to notify visitors
    when new content is added to the site. <br />
    To be added to our mailing list, fill in the form.</p>
    <p>PRIVACY NOTICE: Your E mail address will not be
    forwarded to third parties. It will only be used to
    notify you
    when new material is posted on our site. </p>
    <table width="564" border="0" cellspacing="0"
    cellpadding="5">
      <tr>
        <td width="168"><p>Your Name:</p></td>
        <td width="376"><input name="Name" type="text"
        size="60" /></td>
      </tr>
      <tr>
        <td><p>Your E Mail:</p></td>
        <td><input name="E_Mail" type="text" /></td>
      </tr>
    </table>
    <p>&nbsp;</p>
    <table width="451" border="0" cellspacing="0"
    cellpadding="5">
      <tr>
        <td width="56"><div align="center">
          <input type="submit" name="Submit"
          value="Submit" />
        </div></td>

      </form></td>
    </tr>
  </table>
```

Forwarding the information to a database

Most Web hosting services make it possible for you to quickly set up a database through their control panel. When you set up a database, you define the fields that store the information. Figure 2-17 shows a database as set up through a hosting service's control panel. Listing 2-2 shows the PHP code needed to connect to the database.

Figure 2-17:
The Web hosting service's control panel made this database possible.

The screenshot shows the phpMyAdmin interface for a database named 'mailinglist'. The table 'Mailing_list' is selected, and its structure is displayed. The table has three fields: 'ID' (int(5), primary key, auto-increment), 'Name' (varchar(75), latin1_swedish_ci), and 'E-Mail' (varchar(100), latin1_swedish_ci). The interface also shows indexes, space usage, and row statistics.

Field	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/> ID	int(5)			No		auto_increment
<input type="checkbox"/> Name	varchar(75)	latin1_swedish_ci		No		
<input type="checkbox"/> E-Mail	varchar(100)	latin1_swedish_ci		No		

Indexes: 0					Space usage	
Keyname	Type	Cardinality	Action	Field	Type	Usage
PRIMARY	PRIMARY	32		ID	Data	1,420 Bytes
					Index	2,048 Bytes
					Total	3,468 Bytes

Row Statistics	
Statements	Value
Format	dynamic
Collation	latin1_swedish_ci
Rows	32
Row length ø	44
Row size ø	108 Bytes
Next Autoindex	42
Creation	May 07, 2007 at: 05:06 AM

Listing 2-2: Forwarding the Information to a Database

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//
EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
<meta http-equiv="REFRESH" content="0;
URL=http://www.mywebsite.com/thanks.htm">
<title>Successfully Added Record</title>
</head>
<body>
<?php
$name=$_POST['Name'];
$E-Mail=$_POST['E-Mail'];
$connection=mysql_connect ("mywebhostingservice.com",
"myusername", "mypassword") or die ('I cannot connect to
the database.');
```

```

mysql_select_db ("mydatabase",$connection) or die( "Unable to
    select database");
$query = "INSERT INTO `Mailing_list` (`ID`, `Name`, `E_
    Mail`) VALUES ('','$Name','$E-Mail)";
mysql_query($query);
mysql_close();
?>
</body>
</html>

```

A full dissertation in PHP code is beyond the scope of this book. We do, however, explain how the code makes it possible to connect to the database and transfer the data. The objects with the dollar sign (\$) before them are variables. The variable \$name is set as equal to \$_POST['Name']. In other words, the code is posting the name entered by the customer. The variable \$E-Mail is set equal to the e-mail address entered by the customer on the mailing list form. The line of code that begins with \$connection makes the connection to the database. The line of code that begins with \$query inserts the data into the mailing list. For more information on PHP and MySQL databases, check out *PHP & MySQL For Dummies*, 3rd Edition, by Janet Valade, or *PHP & MySQL Web Development All-in-One Desk Reference For Dummies*, by Janet Valade, Tricia Ballard, and Bill Ballard (both published by Wiley).

Test, test, and then test again

After your e-commerce site is created, test the site to make sure everything works. Yes, we know, this is just common sense. But an e-commerce site does have items that are not found on garden-variety Web sites. The following list shows the obvious — and not so obvious — things for which you need to test:

- ◆ **Make sure all the links work.**
- ◆ **Time the pages as they download.** If any page doesn't download in less than 12 seconds, optimize the graphics until it does. Alternatively, delete graphics you don't need. If possible, strive for pages that download in 8 seconds or less.
- ◆ **Test the pages using the type of Internet connection the majority of your clients' intended audience will be using.**
- ◆ **If the site uses forms that submit information to a database, test the form to make sure the data makes the trip to the database.**
- ◆ **Test the site on real users.** While the site is still under development, invite employees of your client and your friends to test the beta version of the site. Create a customer survey form to get feedback from your beta testers and incorporate this feedback in future iterations of your design.



After securing a server for the e-commerce site, create a generic home page to let people who find the site before it goes public know what they can expect. Create a separate folder — *Test* is a logical name — on the server to store the site while you're creating it. After you finish testing the site, transfer the pages and associated folders to the `root` directory for the site.

Selling Online

After you test the e-commerce site, your client is ready to dive into the deep end of the pool and begin selling his wares. Selling online is like selling from a bricks-and-mortar operation. Your client can stock the best-looking shop in town with the finest goods, but if he doesn't promote it, the business will fail. Security is another concern. Web sites are vulnerable to hackers, scams, and so on. And then there's the matter of the customer, who is the very reason the e-commerce site exists. Getting sales is one thing, but getting customers to come back again is a different story. Not getting customers to return is another reason businesses fail.

If you build it they won't care

So you set up a fancy e-commerce site with all the bells and whistles that the law allows. And you think that just because the site's the coolest thing since sliced bread, it will be overcome with visitors who place large orders. Well, dream on, because that won't happen. No one will care about your Web site — until you give them a reason to care. And they won't have a reason to care if they can't find you. The only way you can attract traffic to your Web site is to get the word out. Many Web site owners spend copious amounts of money employing online marketing and search engine optimization companies. (These methods of promotion are covered in Chapter 3 of this minibook.) However, here are a couple of things a Web site owner can do to get the word out:

- ◆ **Send out a press release.** Many local newspapers will let you post a free press release. Lots of online companies specialize in press releases as well. Type **free press release** in your favorite search engine and you'll come up with Web sites specializing in press releases. Type **press release** and you'll find information on how to write a press release.
- ◆ **Add your URL to your business cards, letterheads, and other written literature.** Include the URL in your e-mail signature.
- ◆ **Have the site featured in online and traditional business directories.** Let your fingers do the walking and type **online business directories** in your favorite search engine. You'll end up with more than enough results.
- ◆ **Submit the site to search engines.** Many Web hosting services enable you to submit a site through their control panel. If not, navigate to your favorite search engines and look for a link that reads, "Submit a URL" or something similar. Follow the prompts to submit the site.

- ◆ **Create cross-links with other sites.** Links help a site's rank with search engines. Encourage your client to write the Webmaster of other popular Web sites asking for permission to link to their site and vice versa. This one-hand-washes-the-other form of linking draws visitors from your site to theirs and vice versa.
- ◆ **Market the site's merchandise through other Web sites.** A network of affiliates that sell similar items or services can expand the sales of any e-commerce site. The affiliate creates links to merchandise on your client's site. The affiliate gets a commission for each sale.
- ◆ **Provide free information.** If your client's site targets a certain niche market, such as photographers, free tutorials give potential customers a reason to return. For this technique to work, the information must be of value to the target group and must be updated frequently. Tell your client to visit forums associated with the niche market and answer users' questions. In the signature of the message, include the URL to the e-commerce site along with a description of the free information available.
- ◆ **If your client has promotional funds, consider using a pay per click option offered by some search engines.** The search engine company charges a fee for each click to your client's site with a particular keyword phrase. For example, if your client is a wedding photographer in Houston, the obvious keyword phrase to choose for a pay per click would be *Houston Wedding Photographer*.
- ◆ **Use search engine ads.** Many Web sites offer ads on the right side of the results page. The ads that appear in the results page are relevant to the keywords entered by the search engine visitor.
- ◆ **Advertise the site in print media such as trade publications associated with the product being sold from the e-commerce site.**

Security concerns

The Internet can be a nasty place. Web site owners have to contend with hackers, online theft, fraud, and viruses, to name a few. Any owner of an e-commerce site must be prepared for problems. In addition to these security issues, you must consider security for e-commerce transactions as discussed in the previous chapter of this minibook. Here are a few things that you can do to bolster the security of any e-commerce site:

- ◆ **Add a copyright notice to the bottom of each page.**
- ◆ **If your client is a photographer or is selling fine art online, make sure each image bears either the copyright symbol and the name of the creator or a watermark.** Furthermore, you can incorporate JavaScript on the page, which prevents visitors from right-clicking an image and downloading it to their computer.

- ◆ **To prevent high-resolution images from being downloaded, use JavaScript to implement an image swap when users mouse over the image.** The replacement image can either be a warning that copyright laws protect the image, or it can be a low-resolution version of the same image that the visitor would not want to download.
- ◆ **Copy all your client's Web site files to CD or DVD.** If the server's equipment crashes or a hacker destroys the site, you can quickly get your client up and running again by uploading the files to the server. Archival CDs or DVDs are the best bet. However, these are quite expensive. If you use them, make sure your rates reflect the added expense. Alternatively, you can use media from a quality manufacturer like Memorex, which should last for a long time.
- ◆ **Advise your client to protect business computers with firewall and antivirus software.** This prevents hackers from accessing your client's business records or sending virus-laden e-mails to your client.
- ◆ **Advise the client to install some sort of anti-spam software on the business computers.** Spammers are relentless, and your client's e-mail addresses are up for grabs after the site is live.
- ◆ **Make sure the Web hosting service has adequate protection.** Hackers often know how to access hosting service control panels. A successful attack on an e-commerce site through a control panel can be devastating. For example, the hacker can clean out your client's database.
- ◆ **Advise your client to change his password frequently and to choose one that cannot be easily guessed by competitors or hackers.** This is very important because many hosting services use a combination of the client's e-mail address as the user name and a client-generated password to gain access to the control panel. The best password is alphanumeric and at least eight characters in length.
- ◆ **To ensure security for all transactions on an e-commerce site, choose a secure server that offers 128-bit SSL (Secure Sockets Layer) encryption.**

Following through on every sale

Businesses that succeed rely on repeat customers to keep the ball rolling. It's expensive to entice new customers to use a product or service. Happy customers who use a product repeatedly and refer friends and business associates are the hallmark of a successful business. The easiest way to get customers to use a product again and again is to treat them like royalty before and after the sale. Here are some tips for following through on a sale:

- ◆ **Send each customer a thank you letter via e-mail.** The letter also serves as a confirmation of the order.
- ◆ **Send each customer a copy of the invoice via e-mail.** This is yet another confirmation of the order, and it shows customers that their business is valued.

- ◆ **Send each customer a message letting her know when her order will ship.** This is another opportunity for the e-commerce company to get its name in front of the customer.
- ◆ **Send each customer a tracking number when the order has been shipped.** Keeping customers informed is good business.
- ◆ **Invite each customer to participate in a survey a few days after the order has been received.** Asking customers to participate in a survey makes them feel more important. A discount on future orders can be used as an incentive to ensure that a high percentage of customers participate in the survey.

Chapter 3: Maintaining an E-Commerce Site

In This Chapter

- ✓ **Coping with client-side and server-side technology issues**
- ✓ **Working with customers**
- ✓ **Optimizing and marketing the site**
- ✓ **Upgrading your hosting service**

After your client's squeaky-clean e-commerce site is uploaded to the server, your job is done. Well, maybe. If your client is techno-phobic, he'll rely on you for advice when the going gets rough. And like anything else technical, the going will get rough. Believe us.

In many instances, your job as a Web designer ends when you hand off the shiny new site to your client. Your client, however, might not be the most techno-savvy person on the planet. Therefore, you have to serve another role — advisor — after the site is uploaded to the Web hosting service. In this chapter, we show you what to recommend to your client when the going gets rough.

When Technology Breaks Down

The best-laid plans go to waste, and the best-engineered technology can and will break down at the most inconvenient time. Technology comes in two forms: client side and server side. The code on your HTML pages marries client-side technology with server-side technology. Your role as Web-design guru means that you end up doing some handholding with your client when technology breaks down. The following sections deal with both client-side and server-side technology problems.

When client-side technology runs amuck

An example of client-side technology is code within your HTML pages that's executed with client-side technology, such as the Web site viewer's browser and associated plug-ins. If you did your homework, you should know what scripting works with the technology used by your client's target audience. Therefore, the only potential problem with client-side technology rests

firmly on your shoulders. If the code doesn't work, you end up looking like a jerk. The best medicine against code errors is to test, test, and then test again. After you're sure your code is up to snuff, you can add the following safeguards to ensure that your client's target audience sees exactly what you intend it to see:

- ◆ **If your Web design is optimized for a specific browser**, make sure to leave a note to that effect on the home page of the Web site. You can also include code that identifies the browser with which the user is accessing the site and redirect the user to the appropriate page that's optimized for that specific browser.
- ◆ **If you're using technology that's dependent upon DSL or cable modem connections**, make sure you include an alternate page for users that still connect with dialup modems.
- ◆ **If your Web design is optimized for a certain-size desktop**, make sure this information is clearly noted on the home page of the Web site.
- ◆ **If your design needs plug-ins**, such as the latest version of Flash Player or Adobe Reader, make sure you include a link to the site from which the browser plug-in can be downloaded.
- ◆ **If you rely on JavaScript drop-down menus or links**, here's another potential JavaScript problem that can occur. Most search engine spiders cannot follow a JavaScript link. If you use JavaScript links, create a text navigation menu at the bottom of each page for search engines to pick up on.
- ◆ **If you include mailto: links in your design**, make sure that your client's target audience uses e-mail applications that support this method of launching a client e-mail application and opening a blank message. Knowledge of your client's target audience can enable you to ascertain whether this might be a problem.
- ◆ **If you really care about what visitors think**, include a section where visitors can comment on site usability. This section can provide any red flags to usability problems related to client-side technology. Negative feedback can tell you what you need to know to revise your code so that such problems don't occur.
- ◆ **If you think that the limitations of the user's processor and memory are entirely out of your control**, you're right. In this regard, design your site to be compatible with the lowest common denominator of current computer technology.

Server-side technology, or code sleight of hand

You enhance the usability of your client's site by adding items such as blogs, forums, databases, and so on. This technology relies on the server to interpret the code in your pages to return a desired result. For example, PHP

code is commonly used for blogs and user forums. Before you create any page that relies on server-side technology, you must be sure that the service that hosts your design supports the technology. Again, the best defense against server-side technology issues is to test your code thoroughly before the site goes public.

Another server-side technology issue is the availability of the hosting service. If your client's site isn't accessible 100 percent of the time, you might need to consider changing your client's hosting service. Web server outages can be another problem. If your client's Web hosting service is hit by a natural disaster (think tornado or hurricane), your client's site is definitely down for the count. However, if your client's hosting service is out for any other reason, consider getting another server. Most Web hosting services have generators that kick in whenever the local power company has a failure. The Web hosting service should also have replacement parts to quickly repair a server that runs amok.

If your client's customers complain that the site takes too long to download and you designed the site as lean and mean as possible, the server's connection might be the problem. A good Web hosting service has *OC3 lines* (enable a data transfer of 155 megabits per second) or *DS3 lines* (enable a transfer rate of 45 megabits per second). Most Web hosting services have several incoming lines that are a combination of OC3 and DS3 lines.

Handling Customers

Customers are the lifeblood of any business, and Web design is no different. Whether you're creating a Web site for a small, mom-and-pop operation or a Fortune 500 company, you have to deal with client issues on a repeated basis. The key to working with any client is communication. Even though you might be an accomplished vocal communicator, the type of communication called for here is the written variety. Your client has to deal with customers as well. Although the following can't be considered a primer in working with customers, we do offer some sound advice in that regard.

Getting it in writing

If you're an experienced Web designer, you'll start the project after your initial consultation with the client. Your portfolio does your talking, and the client hires you based on your experience and what you say during the consultation. If you don't have too many Web sites notched on your totem pole, you'll have to first sell yourself and your skills to the client and then create a mockup for the client to peruse. Then you're hired by the client to design a Web site and look after it in sickness and in health, 'til death do you part. Oops, wrong vow. At any rate, whether you're an experienced pro or an accomplished designer earning your stripes, you need a contract to cover your posterior — and for that matter, your client's.

When you write a contract, make sure you list in detail everything you're going to do for the client. You should list every phase of your work, such as optimizing photos, scanning images, publishing the Web site, optimizing the site for search engines, uploading the site to the server, submitting the site to search engines, and so on. If you list every phase of the project, your client sees the added value in the services you're providing. The added value justifies the cost of your services and often helps you get a job when bidding against a competitor who is not as meticulous as you when it comes to contractual issues.

You should also include details such as whether you're including hosting with your design fee — and if so, for how long. Another detail you need to address is whether you'll revise the site as time goes on. Major revisions are time consuming, and minor revisions are annoyances when you have other work to do. And, of course, your client wants any revision done immediately, if not sooner. We strongly advise you to address revisions in another contract. If you take our sage advice, your revision contract should be as detailed and ironclad as your original Web-design contract.



Schedule revisions at the same time every month. This enables you to schedule the workload when it's convenient for you. Your revision contract should include a clause that says that your client is responsible for getting all materials such as revised text and images to you in a timely manner. If the client drops the ball, you're not responsible for a tardy release of an update.

You also need to address contingencies, such as the client being late in delivering material that is imperative to your design. After all, if the client is going to hold your feet to the fire on the final delivery date, you need to hold your client's feet to the fire as far as the delivery of needed material. You should also cover legal issues, such as the venue in which any disagreements that require the services of an attorney will be resolved.

Your contract should also include a payment schedule. Typically, you're paid a percentage when the client signs the contract and you receive additional payments when you've achieved certain milestones: for example, completing 50 percent of the design, completing 100 percent of the site and making it available for client review, and launching the site to the public. Make sure you include a clause that covers you in the event that the client doesn't deliver the material you need to complete the site. There will be times when you've completed the design, and all you're waiting for is the client's text to fill in the blanks. Your contract should include a clause to the effect that the applicable milestone payment is due even if your client doesn't deliver the text or other required material in a timely manner.

You also need to cover the client who waffles — ahem, you know, changes her mind. Time is money, and in case the client significantly changes the design from what she originally signed for, you need to include a clause that covers this eventuality. Bill changes at your hourly rate. You do have an hourly rate, don't you? Another factor you need to cover is additional material you need to create that's not covered in the contract, such as additional sections of the site or additional images the client supplies that you must optimize for the design. In our contracts, we add a clause that if the total scope of the site exceeds what is listed in the contract, the client will be notified so that we can negotiate an agreeable fee for the additional work. Most Web designers do extra work on an hourly basis.

Create a boilerplate contract that lists all the things you usually include with your Web design. This is a tremendous timesaver. While your prospective client is still thunderstruck with your presentation, you can open your boilerplate contract, add any needed clauses, and quickly send it to him. Better yet, keep one in your briefcase and get the client to sign on the dotted line after you make your presentation. Strike while the iron is hot, and don't give your prospective client a chance to shop your services with another designer.



To make sure you dotted all the *i*'s and crossed the *r*'s, have an attorney review your boilerplate contract.

Documenting everything

Never do business on a handshake. It always comes back to haunt you. Make sure your client signs and dates your original contract. If you need to make minor changes to the contract, cross out the original text and then write in the revision. Both you and the client need to initial the revision. If the client requests significant changes in the contract, rewrite it — or tell the client to take a hike.

Make sure all additional client requests — with the exception of really small details — are handled with a change order. Now, we know what you're thinking: That's a lot of paperwork. The change order doesn't need to be anything elaborate, though; a quick note to the client on your letterhead will suffice. Leave an area for the client to accept and date the change order. Include a SASE (self-addressed, stamped envelope) with the change order and don't do any additional work until you receive the signed change order.

You should also document the client's feedback. When you create new content and upload it to the server, send the client an e-mail requesting that she review the changes and comment on them by return e-mail. Print out the e-mail with your client's comments and put it in the client's folder. That's known as CYOP — cover your own posterior.

Handling e-commerce customers

No Web designer in his right mind should take on the task of handling e-commerce customers. After all, taking care of your own clients is a full-time job. However, if you're dealing with a client who did fall off the turnip truck yesterday, she might not have a clue as to how to handle customers. Just because your client's online shop is selling the neatest new gadget since the wheel doesn't mean she'll be able to handle the going when the going gets tough. You can provide value-added services that separate you from your competition if you can give your customer advice on how to deal with her customers.

First and foremost, your client should keep copies of all electronic communication with customers and keep fastidious records. He should meticulously document every sale. Unless your client's product can be downloaded electronically, he will have to deal with shipping. If your client is inexperienced, advise him to develop a relationship with a shipper. This makes it easier to deal with goods damaged in shipment. If your client has a good, working relationship with the shipper, it's easier to resolve damaged goods claims. Whenever we ship something, we err on the side of overkill and package the item so it will survive a 6-foot fall.

Search Engine Optimization (SEO) and Marketing

If you build it, they will not come. And that's the truth. There are more sites on the Web than Carter has little liver pills — in other words, way too many for people to care about the masterpiece you just designed for your client. If your client's site is to succeed, you have to give customers a reason to care, Bunky. Hopefully, your design and your client's content are enough to keep them at the site, but first you need to grab that herd of horses by the scruff of the neck and drag them to water. You can achieve part of the task by optimizing the site for search engines. This brings some of the horses to the pond. To search out the thoroughbreds for their sip of Perrier, though, you have to resort to more esoteric marketing techniques. Search engine optimization, like the proverbial well, is deep. We show you how to optimize your site and present some marketing techniques in the upcoming sections. These are really just the tip of the iceberg. For a heaping plateful of Web site optimization techniques, pick up a copy of *Search Engine Optimization For Dummies*, 3rd Edition, by Peter Kent (Wiley).

Optimizing the site for search engines

If you know how to properly optimize a Web site for search engines, you can make a lot of money — in fact, maybe enough money to give up your day job as a Web designer. Many visits to commercial Web sites occur as the result of users typing pertinent keywords in a search engine. Your job as a Web designer is to have your client's site show up at or near the top of the

first page of results from a keyword search. We know: easier said than done. Scores of words have been written on optimizing a site for search engines — enough to fill several books. The following are a few tips you can use to optimize your client's site:

- ◆ **Test keywords.** Brainstorm with your client and come up with a list of keywords or phrases that you think users enter into search engines to find sites similar to your client's. Test the keywords and phrases in the major search engines such as Google and Yahoo! to test your theory. Discard the phrases or keywords that don't bring up sites similar to your client's. If a keyword or phrase brings up the Web site of your client's fiercest competitor, put that keyword at the top of the list.
- ◆ **Jot down the titles of your client's competitors' Web sites.** Also include those of similar businesses that are in the top 10 percent of a search result using keywords or phrases that customers use to find sites similar to your client's. Use a variation of these titles for your client's Web site.
- ◆ **Create a keyword-rich title for each page of the Web site.** You can modify the title by changing page properties. Create a unique title with keywords that are likely to be used to find your client's site. A site's *title* is displayed in the search engine's results page. Some designers think that a series of keywords or phrases will get the job done. However, just because the site's title vaults a site near the top of the first page of search results, it won't necessarily drive traffic to the site. In addition to being keyword rich, the title must make sense and give users a reason to click through to the site.
- ◆ **Make sure that the URLs entice visitors to click through to the site.** For example, which site would you visit if searching for *telephoto lenses*?

```
www.photosuppliesrus.com/products123.asp
```

```
www.photosuppliesrus.com/telephotolenses.asp
```
- ◆ **Mirror the keywords from your title tag in the alt text of each image on the home page.** *Alt text* is displayed in text readers to describe an image. Some browsers also show alt text as a tooltip when the visitors pause the cursor over the image.
- ◆ **Mirror the keywords from your title page in the text on the home page.** The redundancy of keywords gives the site a higher ranking with search engines.
- ◆ **Create a text menu at the bottom of the page if your design uses JavaScript for menu navigation links.** Search engine robots have a difficult time following JavaScript links but can easily follow the redundant text links you place at the bottom of the page.
- ◆ **If possible, include keywords anytime you use a heading style on the home page.**

- ◆ **Add content to the meta tags in the head section of each page's HTML.** The `<meta name>` tag enables you to add a keyword-rich description of the e-commerce site. Limit the description to 250 characters, including spaces; this seems to be the limit that search engine spiders recognize. The `<meta keywords>` tag enables you to add keywords and key phrases that pertain to sites, such as the one you're creating. You can include up to 255 keywords/key phrases. Enter a comma to separate keywords and key phrases. Remember to mirror the key phrases you include in the `<title>` tag. Search engines like redundancy — to a limit.



Don't repeat a keyword more than five times in the `<meta keywords>` tag because many search engines consider this practice as spamming and may remove the site from their index.



When creating keywords for the site, include common misspellings of words in your key phrases. Remember to include the town(s) or regions in which your client does business. You might also want to consider adding all lowercase and all uppercase variations of what you and your client feel are the most popular key phrases, as many people type with all caps or all lowercase.

- ◆ **Create links to other Web sites and have them link to yours.** Many search engines increase a site's rank due to its popularity. These search engines include the number of links to your site in that criterion. Read how to do this in the following section.



Consider using a service to do your keyword research. NicheBOT (www.nichebot.com) provides a variety of tools you can use when researching keywords and key phrases.



Don't resort to trickery to try to vault your client's site to the top of the heap. In the past, Web designers repeated keywords and key phrases relentlessly beneath the regular site content. In order to make these invisible to the user, they used the same color as the background, or a color that was one decimal different. Visitors couldn't see the words, but search engine spiders could. Search engines are wise to this trickery and drop a site from their index when they discover a designer's chicanery. Other spamming techniques include adding keywords that aren't related to the site, creating multiple instances of the home page with a different URL and title, or using multiple instances of the same tag.

Finding sites to link to the e-commerce site

Reciprocal links are great, but you should also have a plethora of sites that link to your client's because of the content on your client's site. Oh, yes, your spiffy design is also a good reason for them to link to your client's site. We know what you're thinking: That's not an easy task. Well, it is if you follow these simple steps:

1. Navigate to `www.google.com`.

Google is currently one of the most popular search engines.

2. Enter a keyword that relates to your client's business.

Your search will reveal the most relevant Web sites that pertain to your client's business. As you peruse the results, look for *bona fide* Web sites, not directories of sites. Your goal is to find Web sites of your client's competitors or similar businesses.

3. Type `Link:` followed by the domain name of a Web site from the results of Step 2.

For example, to find out which Web sites are linked to Yahoo!, type **Link:yahoo.com**.

This command, followed by the domain name, returns the URLs of sites that are linked to the domain.

4. Click Search.

Google returns a list of Web sites that are linked to the domain you entered in Step 3.

5. Make a list of the domains that you'd like to link to your client's Web site.

The easiest way to accomplish this is to

- a. *Open a new document in your favorite word-processing application.*
- b. *In your Web browser, select the URL that you want to link to your client's site.*
- c. *Press `Ctrl+C` (Windows) or `⌘+C` (Mac) to copy the URL. Then, with your cursor in the word-processing application, press `Ctrl+V` (Windows) or `⌘+V` (Mac) to paste the URL into the document.*

Alternatively, if you have the full version of Adobe Acrobat, you can use the Web Capture feature to capture the entire page as a PDF document.

6. Contact the Webmaster of the domains that you'd like to link to your client's site.

Compose a simple e-mail to the Webmaster explaining that your client's site has information that would be beneficial to the visitors or the Webmaster's site. The Webmasters will ignore you, disagree, agree, or ask you to provide a reciprocal link. If the Webmaster doesn't agree to link to your site, thank him and try to find out why he refused. This information might help you fine-tune the site.



You can perform a similar search at Yahoo.com to find out which sites are linked to your client's competitors. Type **Link:** followed by the domain name in the Yahoo! search engine to find out which sites are linked to your client's competitors.

If your client uses a particular product to perform his service, ask the Webmaster of the manufacturer's Web site to link to your client's. For example, if your client is a musician and exclusively uses Gibson guitars, ask the Webmaster of `www.gibson.com` to link to your client's site.

Danger, Will Robinson! These do not compute!

Certain elements of Web design are frowned upon for commercial sites because search engines have problems with them. If possible, you should avoid these elements like the plague, but if your client insists on using elements such as framesets and Flash, take a look at the following sections, which discuss the problems and workarounds.

Optimizing sites with frames

If your client wants to get framed, you'll need a private investigator or a good attorney to untangle the mess. But we digress. Web site search engines do a notoriously bad job of indexing sites that use frames — so bad that you should insist your client shy away from frames. When a site is created with *frames*, more than one HTML document can be displayed on a page. Each document is displayed in its own frame. The problem is that when you submit the site, you're submitting the page with the code that designates how large each frame is, how many frames there are, and so on. In other words, there's no real content, such as text, images, alt text, and other items, that search engine spiders can sink their teeth into and properly index the site. If, however, frames are the only logical solution for the e-commerce site you've been hired to design, there is a workaround.

Add the `<no frames>` tag to the frameset page. This tag was originally used for browsers that didn't support frames. However, most modern browsers do support frames. The content you add within the `<no frames>` tag helps get your client's site listed. Listing 3-1 shows HTML code for a frames page with the `<no frames>` tag.

Listing 3-1: Optimizing a Site with Frames

```
<html>
  <head>
    <title>Example Framed Site</title>
  </head>
  <frameset cols="150,*">
    <frame name="contents" target="main">
    <frame name="main">
  </frameset>
</body>
```

```
    Optimized page content for search engine optimization goes
    here.
</body>
  </noframes>
</frameset>
</html>
```

Within the `<body>` tags, place your optimized content for the search engines. You can copy and paste content from the page that shows up in the main frame when the site first loads. You can include all content from a page including HTML within these tags. Use the optimization tips discussed earlier in this chapter. Be sure you include content in the `<title>` tag and the meta tags.

Jumpin' Jack Flash is not such a gas

A Flash movie is a thing of beauty — what with dancing images, flying letters, and whatnot. Many designers like to use Flash for an intro or splash page. They look cool, but search engines think they hit a major fog bank 'cause they can't see links, text, and all the other stuff they use to index a site. In spite of this, you can do a few things to increase the visibility of a Flash intro or splash page because the Flash movie is housed inside an HTML document:

- ◆ Use the `<title>` and meta tags to add information that contains keywords relevant to the site you're designing.
- ◆ Add text that will appear below the *fold* (the area below the browser status bar that viewers will have to scroll to). The text you add should be meaningful because some people might be curious and scroll down to see what's there.
- ◆ Add a text navigation menu below the fold. Search engine spiders can't see what links are in a Flash movie, but they can read the text links and use them to spider the other pages in the site.

Using search engine software

You can use other methods of optimizing your site and then submitting it to search engines. Some software applications can analyze your Web design in regards to popular search engines. The actual options depend on the application. Some applications make suggestions on what you can do to optimize the site, some generate keywords, and others provide means for submitting your site to the most popular search engines. WebPosition 4 is considered the granddaddy of optimization applications. You can download a trial version by visiting www.web-positiongold.com. You can find other

optimization packages by typing “**search engine optimization software**” in your favorite Web browser. You can also find reviews of optimization software to determine which package best suits your needs. Some applications enable you to track a given number of optimized sites. As of this writing, the standard edition of WebPosition 4 enables you to optimize and track five domains, whereas the Pro version of the application features unlimited domain support.

Marketing the site to the world

You can use many different methods to market an e-commerce site. The client usually does this after the site goes live. However, your client will probably count on you for suggestions, or you might decide to include marketing as part of your design contract. In either event, you need to know something about marketing an e-commerce site — or you have to employ the services of a professional Web site marketing company. As with any organization, Web site marketing companies come in all flavors. Your best defense against a snake-oil salesman masquerading as a professional Web site marketing company is to do some research or take advantage of some of the services you can purchase through search engines.

Using a pay per click promotion

Many search engines offer the option of a paid ad that shows up in a results page when users enter a certain keyword or key phrase. Like anything else, he who swings the biggest club gets the game. With most search engines, you specify how much you’ll pay for each click on an ad that appears when users enter a specified keyword and specify your maximum budget per month. The placement of your ad is determined on how much you bid per click versus your total budget. In other words, if your bid on a keyword and budget is higher than another advertiser’s, your ad appears before hers. If your bid is high but your budget is low, your ad appears lower in the list. To get a concise idea of how this works, visit the Google AdWords Help Center at https://adwords.google.com/support/?hl=en_US. (You can also read through *AdWords For Dummies* by Howie Jacobson. For a bigger picture, read through *Pay Per Click For Dummies*, 3rd Edition, by Peter Kent. Both books are from Wiley.)

When you take out an AdWord, you create the content for your ad. The resulting ad appears in a prominent position in the search engine’s results pages for the particular keyword(s) you purchase. Figure 3-1 shows the results page for the key phrase, *Tampa Photographer*. The paid ads are on the right side of the page.

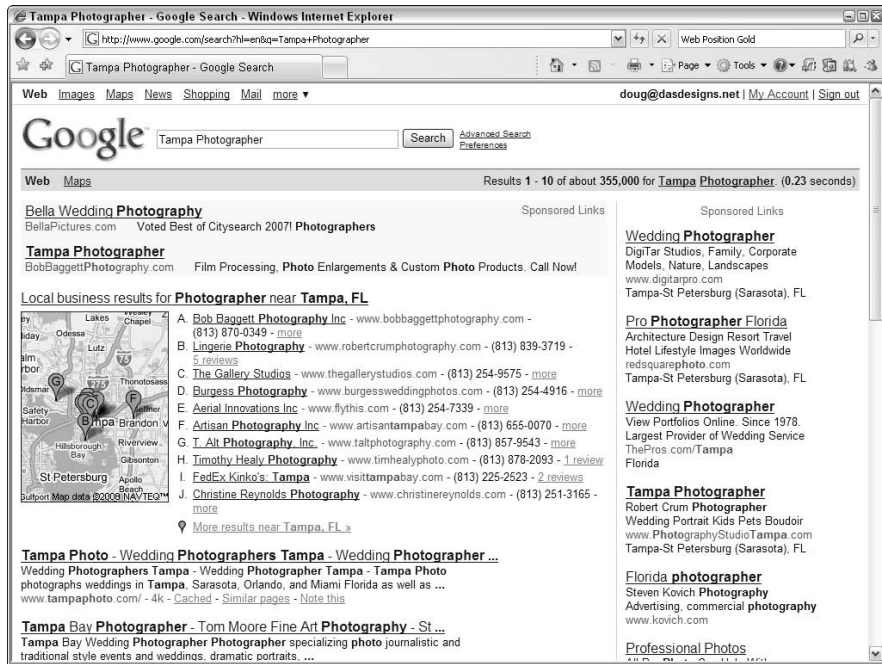


Figure 3-1:
Keyword ads can help draw traffic to an e-commerce site.

When you decide to include paid keyword ads in your online marketing campaign, you have four steps to complete before that paid keyword ad is in place:

1. Determine which keywords are relevant to your client's business.

If your client markets many products, he'll quickly go bankrupt paying for ads for each product he sells. If your client sells multiple products, the key is to break the phrases related to your client's offerings' genres. For example, if your client sells three brands of acoustic and electric guitars, a logical choice would be *acoustic guitars* and *electric guitars*. Your client pays for two key phrases instead of six.

2. Find the most popular keywords and key phrases that relate to the particular genre.

You can employ an online marketing company to do this work. If your client is on a budget, he can use one of the online tools, such as Google AdWords Keyword Tool, which you can find at <https://adwords.google.com/select/KeywordToolExternal>. Simply enter the word for which you'd like to find keywords, and the site comes up with a

page of results. Figure 3-2 shows the results for the key phrase, *Tampa Photographer*. The results page shows the key phrases used most in conjunction with searches for a photographer in Tampa, Florida. Note that the key phrase, *photographer tampa*, was used the most times when searching for a photographer doing business in Tampa. The most popular keyword or key phrase is also the most expensive. Review the list and find a keyword or key phrase that is most relevant to your client's business and will result in customers perusing his Web site. To take the Tampa photographer scenario a step further, the fourth keyword on the keyword suggestions page, *tampa florida photographer*, might be the ideal choice.



Subscription services for keyword research are also available. Wordtracker (www.wordtracker.com) and KeywordDiscovery (www.keyworddiscovery.com) are available by subscription and also feature free trials.

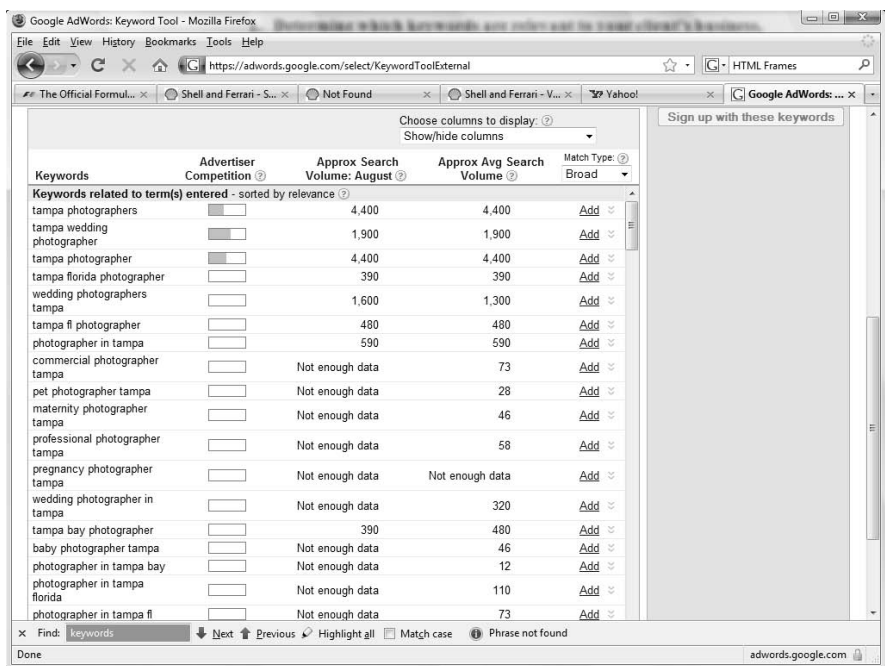


Figure 3-2: You can find suggestions for keywords and key phrases online.

3. Write the ad that will display in the results page with your client's chosen keyword or key phrase.

Create compelling ad copy that will make people want to click through to your client's site. You should also include the URL to the Web site in your ad. Even if users don't click through, they might remember the URL to the site and visit it in the future. But the most compelling ad copy in the world is wasted if the landing page to which the ad is linked is not

relevant. For example, if your ad copy is about wedding photography, the wise choice would be to link to the page about wedding photography, not the home page, on your client's site.

4. Fine-tune the ad.

Google and Yahoo! enable you to target an ad by language or location. For example, you can target an ad to appear in one or more of the following: countries, territories, regions, and/or cities. Furthermore, you can specify that the ad be available to only a specific language audience. Targeting enables you to get more bang for your buck.

The only way you'll know for sure whether the pay per click campaign is working is to monitor the site statistics. Google and Yahoo! supply statistics with their campaigns. Over time, a thorough review of statistics enables the e-commerce site owner to edit the key phrases and, if necessary, choose new ones. In addition, you can monitor the statistics log supplied by your Web hosting service to see if a campaign is effective.

Other marketing techniques

If your client has a startup company, her budget might not be able to afford a pay per click campaign. Never fear; there are other ways to get visitors to her site. The following are a few techniques you and your client can employ:

- ◆ **Use link exchanges.** As we mention previously, you can popularize a Web site by exchanging links with other sites. The easiest way to accomplish this is to solicit links from e-commerce Web sites that offer similar services but aren't in competition with your client's services. The link to your client's site appears on the other site's link page. In return, your client posts a reciprocal link to the other Web site on his links page. Many Webmasters are familiar with this valuable marketing tool and might have hundreds of links on their site.



One way you can make your client's site stand out in a crowded links page is to create a small graphic with the client's logo and URL to the site. Export the graphic from your image-editing application as a GIF file.

- ◆ **Use banner exchanges.** You can publicize your client's site by participating in a banner exchange. Your client's banner is displayed on other sites that subscribe to the service. When clicked, the banner drives visitors to your client's site. Many banner exchange programs are free. When you sign up for a banner exchange program, make sure to read the fine print. Sometimes you have to prominently display the banner exchange company's logo on your client's site in exchange for the free service.
- ◆ **Offer free content.** If your client is an acknowledged expert in her field, advise her to offer free content, such as articles, to established sites that might be frequented by visitors who would be interested in your client's product or service. In exchange for the content, the site provides a link to your client's site.

Figuring Out When to Upgrade

When you initially consult with a client, your first concern is what type of Web site you'll design to best suit your client's need. The second concern is what features your client needs from the Web hosting service. When you first consult with the client, you can get a pretty good idea of what he needs from a Web hosting service. However, if you, or your client, monitor the site statistics, you'll know when it's time to upgrade. Other hints can come from visitor feedback. This vital information tells whether the site is immediately accessible or downloads slowly.

As the site evolves, your client might need additional features that are not included in the hosting package. When you initially choose a company to host your client's site, make sure that it's flexible and will let you change hosting packages when needed. Another useful option is a money-back guarantee if the hosting service doesn't live up to its promises. Additionally, some hosting services will let you cancel the service, without penalty and for any reason, with a 30-day notice. The Web hosting business is very competitive. You can pick and choose to get the service that best suits your needs and your client's.

Upgrading to a more robust Web hosting package

When you initially design your client's site, you might be under budgetary constraints, which means you'll choose a hosting package that can take care of the client's immediate needs and nothing more. However, when the site gets off the ground, and your client's bank account can no longer be balanced with an abacus, your client might need another tax deduction or four. One way you can accomplish this is by upgrading to a better Web-hosting package. Of course, your client has to justify any additional expense to her accountant. Here are a few reasons why she should consider upgrading her Web hosting package:

◆ **Your client needs to host more domains.**

If your client expands into different ventures that need their own unique domain names, you'll have another project to create. In addition, your client will need the ability to host more domains.

◆ **Your client needs more space for the Web site.** E-commerce sites are often used to sell a multitude of products. When more products are sold, more Web pages are needed, more images are needed, and multimedia applications such as video are added, and so on. This takes up space on the server's system. A Web hosting service offers a limited amount of disk space. When your client is close to using her allotted space, it's time to upgrade to a package that offers more.

- ◆ **Your client needs features such as FTP access for customers, one or more databases, CGI (common gateway interface) forms, SSI (server-side includes), and so on.** Entry-level hosting packages generally don't offer these features. When your client needs one or more of these features, it's time to upgrade to a different hosting package.
- ◆ **Your client needs content that is updated on the server side.** Most entry-level Web hosting packages don't support server-side technology, such as ASP or ColdFusion. Additionally, if your client needs a robust database using MS SQL or MS Access, you'll need to upgrade to a Web hosting package that supports these applications.
- ◆ **Your client needs additional subdomains.** A *subdomain* is like a different Web site, but it has the same domain name. For example, if your client wants to separate his customer service from the main Web site, the subdomain would be something like *service.mydomain.com*. When your client has used his allocation of subdomains, it's time to upgrade to a more robust Web hosting package.
- ◆ **Your client needs more e-mail addresses.**

Most Web hosting services give you a finite number of e-mail addresses. If your client's needs exceed this number, you have to purchase additional e-mail addresses or opt for a more robust Web hosting package.
- ◆ **Your client exceeds his monthly data transfer allotment.** If you design a site with bells and whistles (video and audio clips), visitors are downloading more data when compared with a site with just text and graphics. When the site gets popular, the sheer number of visitors tax the data transfer allotment. Rather than paying a premium when your client exceeds his monthly data transfer, he should upgrade to a hosting package that features a higher data transfer allocation.

Each site has different requirements, and each Web hosting company is different. You should always endeavor to get the best package, based on your client's needs. When you shop for hosting space for your client, make sure you're dealing with a hosting service that is flexible and will let you upgrade your client's service at any time. The best advice we can offer in this regard is to find a reliable hosting service for your personal Web site that treats you right and has a great track record. Based on your personal and hopefully acceptable experience with the Web hosting service, you'll know you can use the company for your clients as well.

Upgrading to another server

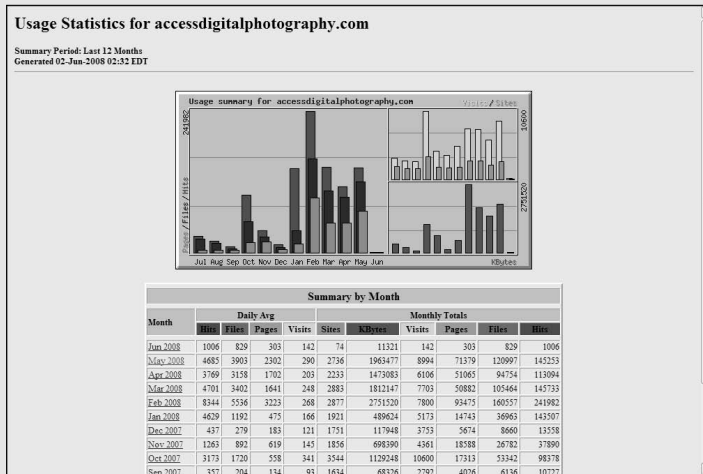
If you did your homework, the server you choose for your client's e-commerce site will perform flawlessly. As the old saying goes: "If it ain't broke, don't fix it." However, not all servers are created equal. And

sometimes, servers sign up more users than they can comfortably accommodate. When this happens, visitors experience longer download times because of server traffic, or they might not be able to connect to the site at all. If your client's site experiences any usability issues that can be attributed to the server, you either need to work out the issues with the server or upgrade to another hosting service.

Sometimes server issues can be attributed to the type of server on which your client's site is hosted. If your client's site is hosted on a shared server and visitors report usability issues such as long download times, upgrade to a dedicated server. A dedicated server costs more money. However, if the site has the traffic to justify a dedicated server and the traffic is purchasing goods or services, the additional cost is money well spent.

Book VIII

Site Management



Checking a site's usage stats.

Contents at a Glance

Chapter 1: Helping a Site Succeed	629
Planning and Incorporating Search Engine Optimization (SEO)	629
Hiring the Right Professional Help	636
Determining Whether Your SEO Is Working	640
Using Reciprocal Links to Boost the Site's Visibility	644
Promoting a Site with Traditional Marketing.....	644
Keeping Them Coming Back	645
Chapter 2: Maintaining a Web Site	649
Updating Pages without Destroying Them.....	649
Sharing the Work with Contribute	651
Managing a Site with Dreamweaver	665
Tips for Managing Tasks.....	668
Chapter 3: Expanding a Site	679
Use It or Lose It: Should You Add On or Redesign?	679
Headache-Free Expansion	683
After Launching the Expanded Site.....	689

Chapter 1: Helping a Site Succeed

In This Chapter

- ✓ **Optimizing the site for search engines**
- ✓ **Submitting the site to search engines**
- ✓ **Marketing the site**
- ✓ **Keeping the content fresh**

A successful Web site is an ever-changing work in progress. You use your skills as a Web designer to bring your client's business to life on the Web. You take the seed of your client's idea, plant it, and nurture it to fruition. However, the fruit will fall from the vine and rot unless you or your client does something to bring it to the public eye. The first step in making the site successful is optimizing the site for search engines. This step can help your client's site rank higher in the search engines. That, however, might not be enough, especially if the Web site is promoting a popular product or service. If this is the case, your client has to resort to marketing the site online. In addition to marketing the site online, your client can use conventional methods to popularize the site.

After visitors are coming to your client's site in droves, there must be a reason for them to return. That's right — you or your client needs to update it on a regular basis. After all, you wouldn't buy the same magazine month after month if the content never changed. In this chapter, we show you how to implement SEO (search engine optimization), online marketing, and other techniques to help the site succeed.

Planning and Incorporating Search Engine Optimization (SEO)

After you decide what type of content the site will have, you begin building the site in your favorite HTML editor. However, if you don't include one other important step, your client's site will be the virtual equivalent of a store that relies on word of mouth to get its customers. If your client's site is to succeed, it has to be findable — and found — by the masses. And how the masses get most of their information is by using search engines, such as Google. Be forewarned that search engine optimization has become a fairly exact science. And like any science, there's a bit of black magic involved as well. A full-course, soup-to-nuts serving of SEO tips is beyond the scope of

this book. However, in the upcoming sections, we show you a few pointers on optimizing your client's site for the search engines. (For more on SEO, pick up a copy of *Search Engine Optimization For Dummies*, 3rd Edition, by Peter Kent, Wiley.)

SEO defined

A search engine might seem like it's a fairly simple beast. You enter a keyword — or several keywords (also known as a *key phrase*) — and magically, results pages appear that list sites that are supposed to be related to the words you typed in the search engine text box. The results are listed by relevancy (whoever he is). But seriously, the sites on the pages are ranked in accordance to how closely the keywords entered by the user relate to the site.

But how is the ranking determined? How does one site magically pop to the top of the first results page, while others are lost in relative anonymity on the eighth or ninth results page? The answer lies in the skill of the Web designer or the company hired to optimize the site for search engines. Optimizing a site for search engines involves adding the right words to the HTML tags that the search engine robots analyze when adding the site to the search engine index. The most skilled SEO wizards know exactly what will improve a site's ranking with the most popular search engines. They don't resort to trickery, such as typing a bunch of keywords the same color as the background, known as *spamming the search engines*. Most search engines are aware of the old tricks. Therefore, a Web designer who knows how to optimize sites for search engines will always be in demand.

Seeing why SEO is essential

If you're a relatively inexperienced Web designer, you might wonder about all the hubbub regarding search engine optimization. Quite simply, the Web is a competitive place. Literally thousands of sites sell similar product lines, promote similar services, and have similar causes — all of them competing for Joe Public's visit, which will hopefully end with a visit to the checkout. Can you say, "Cha-ching?" We knew you could. And your client's competitors all want to be at the top of the first page of a search engine's results. If one of your client's goals is to achieve a high ranking in the search engines, it is imperative that the site be optimized, either by you or by a company that specializes in search engine optimization and marketing.

Understanding the mechanisms: Meta tags, keywords, descriptions, and alt text

Search engine optimization might seem like smoke and mirrors to the uninitiated. In reality, though, it's a matter of tweaking what's at your disposal. Search engine *robots* index a site based on the information they find in

certain tags. When a search engine robot indexes your site, it stores all the pertinent information in a massive database. When a query is entered in a search engine, it returns a site as a result when the indexed data matches the query. The more the information matches the query, the higher the site ranks. It's your job as a Web designer, or the job of the search engine optimization company, to put the right information in these tags, which will propel your client's site to the top of the results page when certain keywords are entered in the search engine's text field. The keywords or key phrases relate to businesses or organizations similar to your client's. You judiciously place these keywords in meta tags and alt tags. *Meta tags* are found in the head of the document, and *alt text* is an alternate description for an image.

What's in a name?

As we just mentioned, meta tags appear in the head of the document. With the exception of the <title> tag, visitors don't see the tags in the head of the document. Search engine robots can, however, read the information in meta tags and the title tag. Listing 1-1 shows the <title> tag and meta tags from a hypothetical coin-collecting company.

Listing 1-1: The Title and Meta Tags

```
<html>
<head>
<title>Coin Collectors Paradise--A 1-Stop resource for coin
collectors.</title>
<meta name="description" content="Coin Collector's Paradise
is a buyer and seller of rare American and foreign coins.
Coin collector books and research material are also
available.">
<meta name="keywords" content="coins, coin dealers, coin
collectors, American coins, foreign coins, buy coins, sell
coins, appraise rare coins, gold coins, rare coins, paper
money, un-circulated coins, proofs">
</head>
```

The <title> tag is prominently displayed in the Web visitor's browser. This information tells Web site visitors what they can expect to find on the site. The <title> tag is also displayed in most search engine results pages. Notice that the word *coin* is repeated twice in the title of our fictional coin collector's Web site. A key phrase `coin collector` is also listed twice in the title tag.

The first meta tag in the hypothetical HTML document is `description`. The description should also make sense as it is also displayed in a search engine's results page. Always remember that a keyword-rich description will help your site rank higher in a search engine index. A logical description

that tells visitors what they can expect to find at your client's site is more likely to result in a click-through to the site. You can have up to 250 words in your description. However, many search engine robots record only the first 30 words or so. The `description` tag for our hypothetical Web site is short, sweet, and to the point with only 23 words, yet the keyword `coin` is repeated three times, and the key phrase `coin collector` is repeated twice. If you don't add the `description` meta tag to a page, search engine results pages display the first few lines of text on the home page in lieu of a site description.

The final meta tag in the document is `keywords`. You can include up to 255 keywords and key phrases. Keywords and key phrases are separated by commas. Listing 1-1 has a total of 13 keywords and key phrases in the fictional `keywords` tag. Most search engine robots use the first 25 key phrases as part of their indexes, so there's no need to go overboard with this tag. Note that the word `coin` is used repeatedly in most of the key phrases. The only thing we might add is the state, county, and town in which the client's bricks-and-mortar business is located. And if the business were associated with a region (say, Tampa Bay, Florida), include that as well: for example, `Tampa Bay coin dealer`.



Initially, all search engines supported the `keywords` tag. This isn't the case anymore. However, the major search engines still support this tag. If you have to scrimp anywhere in your search engine optimization, this is one tag you can ignore.

Increasing site visibility with alt text

In addition to using meta tags, you can also increase the visibility of your client's site by adding alt text to each `` tag in a page. In a nutshell, *alt text* is text that you add to images. Adding alt text does several things. In some Web browsers, it shows up as a tooltip when a user positions her cursor over the image. If someone is accessing the site via a text-only browser, the tag shows up in a box that is the same dimensions as the image. For visually impaired folks viewing the page with a screen reader, the tag is displayed in lieu of the image. When search engine robots index the site, they use the information in the alt tags for indexing the site. When you add alt text to the `` tags on your site, include the keywords and key phrases used in the title, description, and keywords meta tags. The alt text should also be a logical sentence. If you jam an alt tag full of keywords and key phrases, it might cause the search engine robots to disregard the information. Worse yet, the gibberish can distract site visitors when the tooltip appears.

Improving site searchability

After you optimize a site for search engines, you might think you're ready to upload the site and submit it to search engines. But, you can still do other things to make your client's site easier for potential visitors to find.

Clean up your code

Before you submit your site to the search engines, your code must be impeccable. Most Web designers use HTML editors to create their pages, which creates fairly bulletproof code. However, if you manually modified any of the code, you run the risk of sending mixed signals to the search engine robots. Here are a few things to review:

- ◆ **If you have any JavaScript-based menus, create a text equivalent at the bottom of the page.** Search engine robots cannot follow JavaScript links to index the rest of the site.
- ◆ **If your menus are graphic icons, create a text equivalent menu at the bottom of the page.** Search engine robots cannot read the text on a menu composed of images. You can however, pepper the alt tags for each menu icon with keywords, which can be read by search engine robots.
- ◆ **If any of the code is created by using a word-processing application, check the code for redundant tags.** Dreamweaver has a Clean Up Word HTML command, which removes markup specific to the Microsoft Word application. It also cleans up font tags and a plethora of other tags created by the application.
- ◆ **Avoid adding custom styles to each document.** Instead of adding styles to the head of a document, use a CSS (Cascading Style Sheet), which will create cleaner code for the search engine robots to follow. For more information on CSS, see Book III, Chapter 3.
- ◆ **Use a unique <title> tag for every page on your site and make sure the title is relevant to the content of each page.** This enables the search engine robots to accurately index each page of your site. For example, if one page shows a map and directions of how to get to your client's site, the page title might read: *Directions to your client's bricks-and-mortar store.*
- ◆ **Add an alt tag for every image on the page.** Make sure the tag text matches the content of the page and mirrors the keywords and key phrases used in the title and meta tags.
- ◆ **Validate your code.** Most HTML editors have a command to validate or clean up the code. For example, Dreamweaver has a Clean Up XHTML command. This command removes empty and redundant tags; plus, it has an option to combine nested tags when possible. You can also validate your markup in Dreamweaver by choosing File↪Validate↪Markup.

Create a title, alt tags, descriptions, and keywords

Prior to the site being submitted to the search engines, you need to add the elements that will propel your client's Web site to the top of the desired results page. You do so by creating a unique title for each page, alt tags for

each image, a description for each page, and keywords. This is especially important if your client's site promotes multiple products or multiple services. When this is the case, you optimize the pages for each product or service as if they were individual Web sites. Therefore, the title, alt tags, descriptions, and keywords for these pages will be tailored for the product or service your client is promoting. The following steps show you how to optimize a page in Dreamweaver:

1. **Open the document you want to optimize.**
2. **Open the Property inspector.**

When you initially open a document and haven't yet selected any elements, the Page Properties button in the Property inspector is available.

3. **Click the Page Properties button.**
4. **In the Page Properties dialog box that appears, click the Title/Encoding category.**

The Page Properties dialog box refreshes to show the current title and encoding.

5. **Type the desired title in the Title text box. (See Figure 1-1.)**

The title you choose for the page is the result of your tireless research to determine the optimum keywords to optimize the page for search engines. Redundancy intended.

6. **Click OK.**

The dialog box fades from view.

7. **Choose Insert → HTML → Head Tags → Description.**

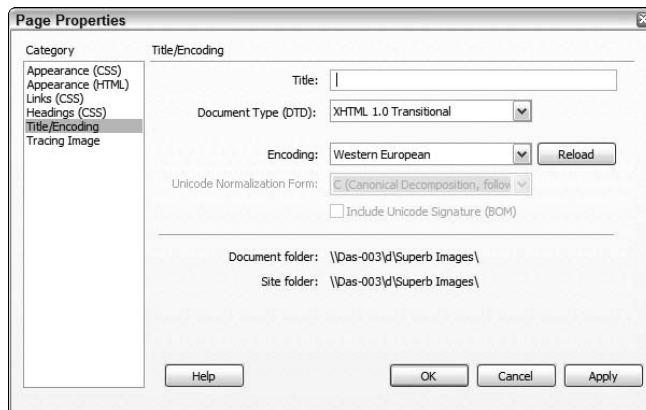
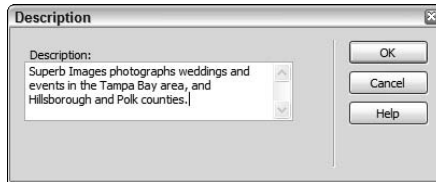


Figure 1-1:
Adding a
title to the
document.

- 8. In the Description dialog box, type the desired description in the text box. (See Figure 1-2.)**

The description you type depends on the product or service your client performs. If your client performs multiple services or sells multiple products, each page needs to have a different description.

Figure 1-2:
Adding a
description
to the Web
page.



- 9. Click OK.**

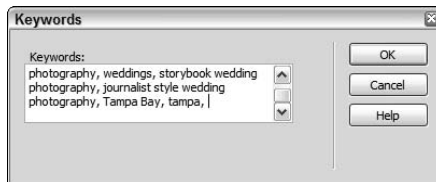
The description meta tag is added to the head of the document you're optimizing for search engines.

- 10. Choose Insert⇨HTML⇨Head Tags⇨Keywords.**

- 11. In the Keywords dialog box, type the desired keywords.**

The keywords and key phrases you enter are the result of your research of other Web sites selling products or services similar to your client's. Keywords or key phrases are separated by commas. (See Figure 1-3.)

Figure 1-3:
Adding
keywords
to the
document.



- 12. Click OK.**

The keywords are added to the head of the document you're optimizing.

- 13. Select the image to which you want to add the alt tag.**
- 14. Open the Property inspector.**

The Property inspector enables you to change many parameters of the image, including the alt tag.

15. Type the desired text in the Alt text field, as shown in Figure 1-4.

The alt tag is added to the document when you save it. Figure 1-5 shows the code of the document that was optimized in the preceding steps.

Figure 1-4:
Adding a descriptive alt tag to an image.



Figure 1-5:
The code of the optimized Web page.

```
Code Split Design Live View Live Code Title: Tampa Bay Wedding & Event
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4 <head>
5 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
6 <title>Tampa Bay Wedding & Event Photography</title>
7 <meta name="Description" content="Superb Images photographs weddings and events in the Tampa Bay
8 area, and Hillsborough and Polk counties." />
9 <meta name="Keywords" content="photography, wedding photography, event photography, weddings,
10 storybook wedding photography, journalist style wedding photography, Tampa Bay, tampa, Polk County,
11 Hillsborough County, beach weddings, coffee table wedding photo album" />
12 <link href="sl.css" rel="stylesheet" type="text/css" />
13 <style type="text/css"></style>
14 <script src="Scripts/AC_RunActiveContent.js" type="text/javascript"></script>
15 <style type="text/css" media="screen">
16 <!--
17 @import url("p7gs/p7gs_base.css");
18 body {
19 background-color: #333333;
20 margin-top: 0px;
21 }
22 -->
23 </style>
24 <script type="text/javascript" src="p7gs/p7gs_scripts.js"></script>
25 <script src="Scripts/suffix_modified.js" type="text/javascript"></script>
26 <script type="text/JavaScript">
27 <!--
28 function MM_preloadImages() { //V3.0
29 var d=document; if(d.images){ if(!d.MM_p) d.MM_p=new Array();
30 var i,j=d.MM_p.length,a=MM_preloadImages.arguments; for(i=0; i<a.length; i++)
31 if (a[i]) d.MM_p[i]=new Image(a[i]);}
32 -->
33 </script>
34 </html>
```

Hiring the Right Professional Help

If you're not a guru when it comes to search engine optimization, you or your client has to hire someone to take care of this task. If not, your client's site will be all but invisible to people using search engines to find Web sites of similar products or services. After all, if a site isn't optimized, it ends up on the fifth or sixth results page of a search — or higher if the product or service is extremely popular. For example, a Google search for *same day shipping* yields 14,300,000 results. FedEx doesn't rank number one because of its expertise in shipping the product on the same day; a team of SEO experts was employed to get this result. Still, FedEx hedged its bet, too, and has a sponsored link that appears above its number one-ranked search result.

Many companies claim to optimize Web sites for search engines. Most SEO companies use a combination of proprietary software and consultation with the client to optimize a site. The company suggests changes that the client needs to make to ensure that the site ranks near the top of the major search engines. Of course, your client's major competitors are also working to have their sites rank near the top of the major search engines. If you think SEO is a one-time shot, you're sadly mistaken. The site will need to be optimized on an ongoing basis. If you or your client thinks you can get off cheaply by hiring a company that claims they can do it in one shot, think again.

As we mention previously, lots of companies claim to be SEO experts. When choosing a company, here are a few questions to ask:

- ◆ How long has the company been doing SEO?
- ◆ Is SEO its main business or just a service offered?
- ◆ Does the company have a proven track record of getting results for their clients?
- ◆ Can the company provide testimonials and references from satisfied clients?
- ◆ Can the company clearly demonstrate a logical process by which it optimizes the site for maximum visibility in the search engines?
- ◆ Does the company offer ongoing optimization as part of its contract?
- ◆ Will the company submit the site to the major search engines?
- ◆ How often will the company resubmit the site?
- ◆ How many members does the company have on its staff?

When you're searching for a search engine optimization company, you should also find out the company's code of ethics — in other words, what steps it will take to optimize the site. If it seems as though the company might employ some non-ethical tactics to optimize the site, look elsewhere. Search engines quickly catch when any form of chicanery is used to vault a site to the top of the search engines. In fact, you can examine the code of one of the SEO company's clients and see for yourself whether anything is rotten in the state of Denmark.

Deciding between SEO and paid advertising

In addition to, or in lieu of, SEO, you can use *keyword advertising*, which shows your client's paid ad on results pages. The ad is listed in the Sponsored Links section of the results page for the keyword or key phrase your client pays for. The frequency and the page on which the ad appears is determined by the amount your client agrees to pay per click and the total amount the client has budgeted for pay per click advertising with the

search engine company. If your client allocates a large enough budget for pay per click advertising, his site will appear in the sponsored links section on several results pages. For example, FedEx Expedited shipping appears on 22 results pages for the key phrase *same day shipping*. In spite of what must be a huge keyword advertising budget, the company must employ a talented team. The page that comes up for this key phrase uses all the techniques we point out in this chapter. If you compare FedEx with the company that comes up as a sponsored link in a search for Tampa Wedding Photographer, you know there's a huge difference in budgets, as this sponsored link appears only on the first results page.

However, certain keywords and key phrases, such as *wedding photographer*, are in high demand. Therefore, the only way to ensure that your client's pay per click ad appears in a desirable position on the first or second results page is to allocate a large budget for pay per click advertising. When faced with a large pay per click advertising budget, your client will have to determine whether it's more economical to employ a top-notch SEO company or to sign up for the pay per click advertising campaign — or maybe both.

Submitting your site to search engines

After your client's site is optimized for search engines, it's time to submit the site to search engines. Some of them are highly specialized and might not be relevant to your client's type of business. There are so many search engines out there that you could go nuts trying to submit the site to multiple Web sites. Therefore, submit the site to the most popular search engines. Of course, what's popular now might be *passé* in two or three years. The following is a list of popular search engines and the URLs to the pages where you submit your site:

- ◆ **Google:** Google is considered the most popular search engine for the masses. You can submit a site to Google at

www.google.com/addurl/?continue=/addurl

- ◆ **MSN Search:** This is another popular search engine affiliated with MSN.com. You can submit your site to MSN Search at this URL:

<http://search.msn.com/docs/submit.aspx>

- ◆ **AltaVista:** AltaVista has been around for a while. It's a popular search engine that offers alternative services such as Yellow Pages, a directory, a shopping and travel section, a people-finder section, and much more. You can submit a site to AltaVista, which is powered by Yahoo! Search Technology. In order to submit a site, you must be a registered Yahoo! user. After you register as a Yahoo! user (it's free), you can submit a site at the following URL:

<http://submit.search.yahoo.com/free/request>

- ◆ **Yahoo!:** You can submit a site for inclusion in Yahoo!'s search engine at the same URL for which you submit a site to AltaVista. However, in order to guarantee that your site appears in Yahoo!'s search engine, you'll have to pay to submit the site, and then pay per click. You can find out more at this URL:

<http://searchmarketing.yahoo.com/srchsb/sse.php?mkt=us>



Many of the sites in the preceding list have specialized directories to which you can submit a site. However, a complete tutorial on submitting your site to search engines is beyond the scope of this book.

Don't resubmit your client's site to a search engine unless you made significant changes to the site or re-optimized pages using a different title or description, or you changed the keywords. Continually submit a site with no changes is considered *spamming the search engine* and can negatively affect the site's rank within the search engine.

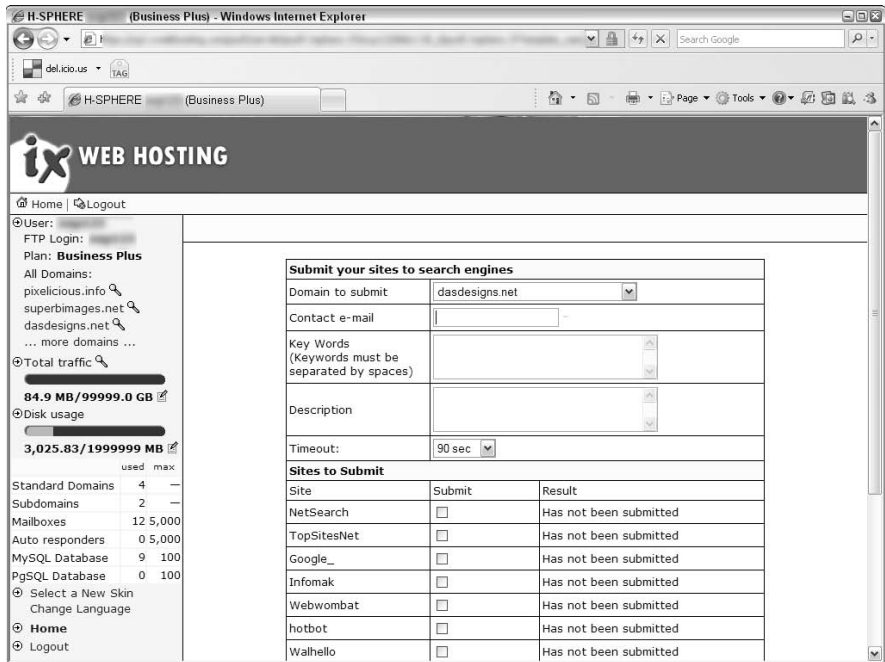
You can find services that enable you to submit your site to multiple search engines for no fee by typing **free search engine submission** in your favorite search engine. However, many of these sites will try to sell you on using their paid services or will add you to some sort of mailing list. If you decide to use one of these services to submit your client's Web site, read the fine print. **Remember:** There are no free lunches.

Your Web hosting service might offer submission to search engines as part of their package. Contact your Web hosting service's customer support for more information. If your Web hosting service offers this feature, you'll probably submit the site using its control panel, as shown in Figure 1-6.

Using a service to submit your site

If you contract an SEO company to optimize the site, chances are that the company will also submit it for you. If you optimized the site yourself and don't want to go through the hassle of submitting the site to multiple search engines, you can employ a service to submit the site for you. Like all services, site submission services come in multiple flavors and varying degrees of initiative. The price charged by the company depends on the number of search engines to which it submits the site, how often it resubmits the site, and other services included in the package such as sponsored links and so on. Most services use proprietary software to submit the site. A good search engine–submission company will also know the ins and outs of each search engine and whether the search engine has specialized directories for products and services to which it can submit your client's site.

Figure 1-6: Many Web hosting services offer search engine submission with Web hosting packages.



You can find a plethora of search engine–submission companies by typing **search engine submission service** in your favorite search engine. Just because a submission service’s site is listed at the top of a results page doesn’t mean it’s a good service. (Can you say, “Sponsored links”?) Like any other service, make sure the company is legitimate and their employees didn’t fall off the turnip truck yesterday. A reputable service has testimonials from satisfied customers. Be a savvy buyer. Don’t take the service’s word for it. See whether the sites listed in the testimonials are still active. If so, see if you can contact the person who wrote the testimonial and ask him for his opinion — off the record, so to speak.

Determining Whether Your SEO Is Working

Even though you submit a site to several search engines or hire a site-submission firm to perform this task, it takes a while to determine what effect search engine optimization is having for your client’s Web site. First and foremost, search engines don’t index a submitted site overnight. Depending on the search engine, it might take several weeks to get indexed. The submit page on each search engine usually gives you an idea of how long it takes a submitted site to be added to the search engine’s index.

If your client is getting more orders or inquiries after the site is submitted, this is one indication that SEO is working. However, there are more scientific ways to determine the impact of SEO. Most Web hosting services provide a means by which you and your client can monitor the number of visits to your site and the means by which the traffic was drawn to the site. The following sections provide useful information on Web site stats, plus what to do with the information you receive.

Reading your Web stats

Web site statistics are a wonderful thing. With most Web hosting services, stats are accessed through the Web service's control panel. In addition to being a wonderful thing, they are artistic and colorful, what with all the bar charts and pie graphs. To read site stats, log on to the site's control panel, click the applicable icon, and . . . *voilà!* Instant stats! Many stat packages show a graphical representation of site traffic. (See Figure 1-7.) In addition, you'll see other information, such as the number of hits, files, pages, and visits. With most services, you click the desired month to get more details. (See Figure 1-8.) We know what you're thinking: What does all this stuff mean? Fear not, intrepid designer! Please fast-forward to the next section for nontechnical definitions of Web stat terms.

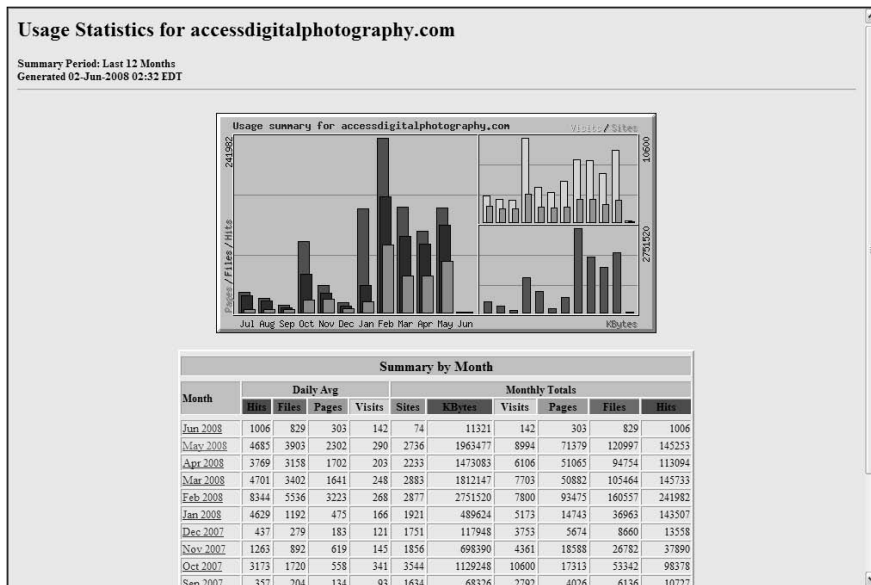


Figure 1-7: A graphical and colorful representation of site statistics.

Figure 1-8:
Getting the
lowdown on
May stats
for the site.

Usage Statistics for accessdigitalphotography.com

Summary Period: May 2008
Generated 02-Jun-2008 02:32 EDT

[Daily Statistics] [Hourly Statistics] [URLs] [Entry] [Exit] [Sites] [Referrers] [Search] [Agents] [Countries]

Monthly Statistics for May 2008		
Total Hits		145253
Total Files		120997
Total Pages		71379
Total Visits		8994
Total KBytes		1963477
Total Unique Sites		2736
Total Unique URLs		1396
Total Unique Referrers		487
Total Unique User Agents		527
	Avg	Max
Hits per Hour	195	5531
Hits per Day	4685	14680
Files per Day	3903	14274
Pages per Day	2302	10992
Visits per Day	290	383
KBytes per Day	63338	290768
Hits by Response Code		
Code 200 - OK		120997
Code 206 - Partial Content		20
Code 301 - Moved Permanently		6362
Code 302 - Found		674
Code 304 - Not Modified		15558
Code 400 - Bad Request		22

Defining Web stat terms

After you crack open the statistics for a site, you have to decipher what they mean. Unless you're a trained professional, the stats might seem like gibberish. The following list is designed to demystify Web stat terms:

- ◆ **Hits:** A *hit* occurs any time the Web server delivers a file. Hits can be misleading because several files might need to be downloaded for an individual page. For example, in addition to the HTML document for a Web page, images, a CSS, Flash movies, and so on must be downloaded as well. A single visit to any page on a site results in multiple hits. Copious amounts of hits do not justify breaking out the bubbly and patting yourself on the back.
- ◆ **Visits:** A *visit* is logged whenever a unique visit is made to the site. This is by far the best barometer you have to determine how successful your SEO efforts are. A word of caution: Some stats services tally a visit whenever a new page is accessed on the site. If this is the case, the stats service provides a listing for Unique Visits, which tells you how many visitors have accessed the site.
- ◆ **Entry pages:** This statistic gives the URL for the page on which the visitor first landed. If your client sells many products and services, this information shows you which optimized pages are doing the best job of driving visitors to the site.

- ◆ **Exit pages:** This statistic shows you which page visitors are viewing prior to exiting the site. While everybody has to leave sometime, if one page shows up more than others, it could indicate that the page isn't being received well by visitors.
- ◆ **Referrers:** A *referrer* is a Web page that contains a link to one of the pages in your client's site. Referrals might come from search engines or other sites that choose to link to you. Knowing how people are finding your client's site enables you to fine-tune the methods used to market the site.
- ◆ **Search strings:** This stat shows which keywords and key phrases the visitor used to find your client's site. If keywords or key phrases that have been used for a pay per click campaign don't show up, you know that the campaign is not yielding results with the current keywords and key phrases.

Adjusting the site and driving traffic

After analyzing your client's Web site stats, you'll have a good idea of how much traffic the site is getting and from where the traffic is coming. You'll also know which online marketing techniques are working, how well SEO is working, and how well your other marketing is working. The site stats tell you which areas of the site need revision and what other steps you can take to drive traffic to the site.

If you or your client has employed SEO, take a good look at the search strings that visitors have used to find your site. Do they match the keywords and key phrases you peppered heavily in the title, description, and keywords? If so, your efforts are working. If the search strings are obscure text in any of the head tags or actual text in the body of the page, revise the tags to reflect the keywords and key phrases that people have used to find the site — which, of course, will help drive more traffic to the site.

If you use pay per click marketing to drive traffic to the site, search strings are once again useful. If the search strings match the pay per click words you budgeted with various search engines, the campaign is doing its job. If not, adjust the pay per click campaign, budgeting the keywords and key phrases that show up most frequently in search strings. You can also use referrers to determine the effectiveness of a pay per click campaign. If the referrers are those that your client has employed a pay per click campaign with, your client's money is well spent. If not, advise your client to suspend the pay per click campaign or lower the budget.

Entry pages are another useful statistic to study when a site sells multiple products or multiple services, especially when you use marketing or SEO to draw attention to the site. The most popular entry pages tell you how effective

your SEO efforts are. If certain pages aren't being visited and these are popular products or services, optimize these pages again. If you used a pay per click campaign on the pages that aren't getting their fair share of visits, adjust the budget or budget different keywords or key phrases.

Stats on exit pages can tell you whether the content is effective. The pages that visitors exit most frequently might need to be tweaked if the pages are products or services that are vital to the client's business. In a perfect world, the most frequently used exit page would be the checkout of your client's secure server.

Using Reciprocal Links to Boost the Site's Visibility

Search engines pay special attention to the number of sites that link to your client's site. If all other parameters are equal and one site has more external sites linking to it, the site with more links ranks higher on a results page. In other words: "He who dies with the most links wins." Therefore, it's in your client's best interest to have as many sites as possible linking to his. If your client's site has interesting content and a compelling design (you did give your client a compelling design, didn't you?), it's relatively easy to get other sites to link to his site. Ask. That's right — send e-mails to Webmasters of sites that you'd like to link to your client's, telling them you like their site, would like to link to it, and would appreciate it if they'd do the same for your client's site. Keep at it until you have a slew of sites linked to your client's and then monitor the stats to see which sites are referring visitors to your client's. At the same time, monitor where your client's site appears in search engine results pages. It won't happen overnight; but in a few months, your client's site will rank higher, thanks to the power of reciprocal links.



Navigate to www.yahoo.com or www.google.com and type **Link: PopularWebsite**, replacing the phrase *PopularWebsite* with the domain of a popular Web site that sells the same product or service as your client's. The domain should be in the following format: *PopularWebsite.com* — for example, *Wiley.com*. You'll get a list of sites that link to the popular Web site. Send an e-mail to the Webmasters of these sites, requesting that they link to your client's site.

Promoting a Site with Traditional Marketing

Web designers should provide their clients with owner's manuals. Just think about the grief it would eliminate, not to mention the phone calls. In addition to telling clients that they won't get results immediately, the owner's manuals would tell clients the obvious things they can do to promote their sites.

When you write your Web site owner's manual, be sure to include the following advice for your clients: Pump that URL! You'd be surprised how many people jot down the URL of a Web site when they see it. Add it to

- ◆ Your business card and the business cards of your employees
- ◆ Every piece of stationery for your company, especially note pads and letterheads
- ◆ Every brochure and catalog you print
- ◆ Every magazine and newspaper ad you purchase
- ◆ Every television ad you purchase
- ◆ Every company vehicle and every sign
- ◆ Any other type of vehicle used to promote the business, such as billboards and ads on park benches.

Keeping Them Coming Back

After you and your client have done your best to launch an informative Web site, you'll have to think about the future. If you don't, the site can quickly lose popularity, and your client's business can dwindle. The best way to keep visitors returning to a site is to keep the content fresh. In this regard, think ahead and plan for future content. In the upcoming sections, we give you a few tips to ensure a steady stream of visitors to your client's site.

Ongoing content development

A successful Web site should have new information at least once every two weeks. That's not really as difficult as it seems. If your client is an expert in his field, he can post articles or tutorials on the Web site. The articles should be compelling enough to ensure that visitors will look forward to bi-weekly installments. The articles can include hyperlinks to the product or service the client is featuring in the article. For example, if the Web site's purpose is to sell art supplies, an article could show visitors how to mix colors. You've likely seen Web sites with a headline that gives a short introduction to an article with a Read More link at the bottom of the blurb. Savvy Web surfers know that this content is usually updated on a regular basis. If your client isn't a writer, perhaps someone on his staff can write the articles. In lieu of that, perhaps the client's vendors can create content. In lieu of that, you can find Web sites that offer free material for reprint, in exchange for an ad.



As of this writing, you can find an index of free Web content at FreeSticky (www.freesticky.com/stickyWeb).

The best way to plan for ongoing content is to know the clientele that are likely to visit the site. Knowledge of your client's intended audience is the key to planning new development. Advise your client to find interesting new stories that relate to her product or service. Include links to these articles on the client's Home or What's New page. If the client has a *blog* (a Web log, which we discuss in further detail in Book VI, Chapter 5), this is an excellent place to mention interesting news articles or product developments.

Another way to keep customers returning is to post success stories on the Home or What's New page. This takes a bit of planning on your client's part. If the client has an established bricks-and-mortar business, advise her to solicit customers for testimonials or success stories. For example, if your client is promoting a weight-loss service, before-and-after pictures are great testimonials. Advise your client to obtain a release from her customer before posting the pictures and success story. Contact your lawyer for more information on what information a model release should include. If your client is a savvy marketer, she can exchange goods or services in exchange for a good article or success story.

The key to successful content development is to do it like clockwork. It will take between four and seven visits before visitors realize that the site is updated frequently. When planning new content, your client should be on the prowl, visiting industry-specific Web sites in search of new material. Your client can set up alliances with Web sites that sell similar products or services to her intended audience. Your client can find articles on these Web sites that interest her customers, and she probably has content that is of interest to other Web sites. This information can be exchanged between sites with a short ad and hyperlink to the marketing partner's Web site.



On the home page, add some text that tells visitors the site content is updated on a regular basis and ask them to bookmark the page.

Another way to develop content for the site is to recycle. As long as the information is timely, there's no reason your client can't post an article again in another three or four months. If your client uses this tactic, though, make sure he keeps track of when the articles are posted and that he mixes them up. In other words, advise him not to repost the articles in the same order they were originally posted. Your client can also do a bit of creative cutting and pasting to create a new article from information that's already been posted on the site.

Creating content to keep them coming back

In addition to creating articles and tutorials, you can employ other devices to ensure that visitors want to return. This type of content is the fun and informative stuff. The following are a few time-honored techniques that can be used to have visitors flocking to the site on a regular basis:

- ◆ **Add short-term specials to the home page.** The specials can be on new products or services. Make sure the special is really a value and not just an attempt to get rid of old inventory. If the visitor perceives the short-term special as a value, he's likely to purchase the product or service. Or even if he doesn't make the purchase, the seed is planted, and the visitor is likely to return when the special expires to see what's offered next.
- ◆ **Provide tutorials.** If your client has a service that includes instructional material, consider giving away a short tutorial in PDF format. When you give something away, make sure you ask for something in return — say, the visitor's e-mail address so you can inform her of the next, free tutorial.
- ◆ **Give away free stuff.** If your client's business has promotional material, such as t-shirts or baseball caps, give the materials away to every 25th visitor who registers contact information on the Web site.
- ◆ **Create a What's New page and update it frequently.**
- ◆ **Create an online newsletter.** Have visitors register their e-mail addresses, and notify them when the newsletter is updated.
- ◆ **Hold a contest related to your client's products or services.** The grand prize can be a product or a discount. The visitor submits his contact information when registering for the contest.
- ◆ **Create fun, interactive pages.** The pages can be quizzes related to your client's products or services. You can create content like this in Flash, which we cover in Book V, Chapter 1.

Chapter 2: Maintaining a Web Site

In This Chapter

- ✓ **Updating Web pages**
- ✓ **Using Adobe Contribute**
- ✓ **Managing the site with Dreamweaver**
- ✓ **Tricks and tips for maintaining Web sites**

You've worked hard to create a compelling Web site for your client. For the site to succeed, though, someone needs to update the content. Updating or maintaining a Web site is usually not included in a designer's contract because the amount of work varies, depending on whether the client needs a major overhaul on a section of the site or the update is simply a matter of making some changes on an existing page. If your client has contracted you to maintain the site, you can do so by using an HTML editor, such as Adobe Dreamweaver CS4. Or perhaps, your client is relying on you to do the major maintenance and his staff will be responsible for the "grunt" work. If this is the case, people who have little or no experience in Web design will edit your pages. We cover both scenarios in this chapter, including some tips and tricks to make the job easier.

Updating Pages without Destroying Them

Clients change their minds when you least expect it. It's one of Murphy's Laws — Rule #327, most likely. So what are you to do when your client comes to you with a bunch of updates and then decides — after you created them — that he doesn't like them and wants to revert to the old ones? Well, if you already modified the pages, you're up the proverbial creek without the proverbial paddle. Whenever a client presents us with the first set of revisions, we do the following in Dreamweaver:

- 1. Choose Window⇨Files.**

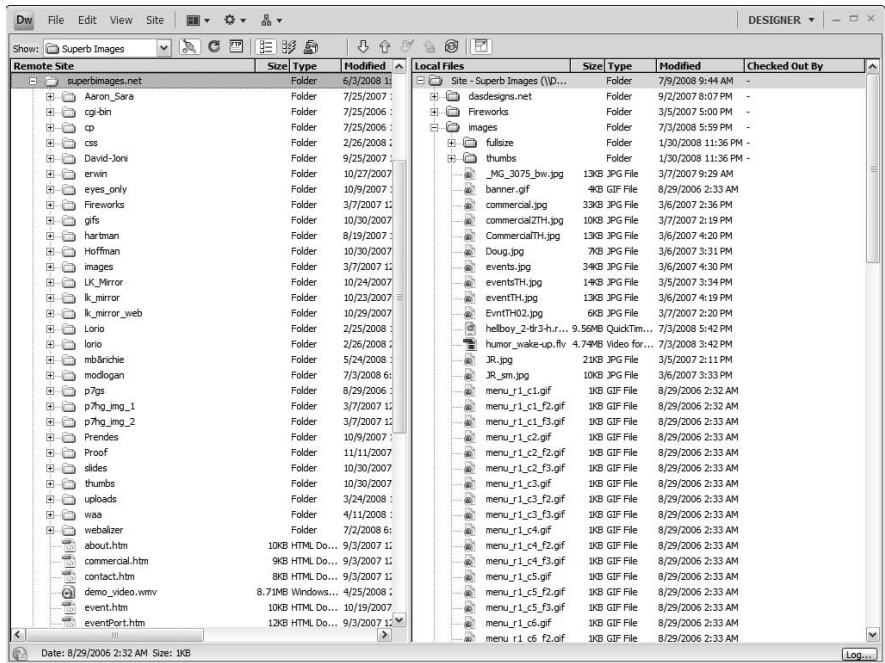
This opens the Files panel for the Web site on which you're working.

- 2. Click the Expand to Show Local and Remote Sites button.**

3. Click the Connects to Remote Host button.

Dreamweaver connects to the remote host and displays all files and folders on the remote host. (See Figure 2-1.)

Figure 2-1:
You can manage files on your local machine and the remote host.



4. Right-click (Windows) or Control-click (Mac) the remote host root directory and choose Create New Folder from the contextual menu.

Dreamweaver creates an unnamed folder.

5. Change the name of the folder to Archive.

This is where you'll store the original HTML files for the Web site.

6. Select all the original HTML files for the site, including any CSS (Cascading Style Sheets) files and JS (JavaScript) files.

7. Choose Edit→Copy.

Dreamweaver copies the files to the Clipboard.

8. Select the Archive folder.

9. Choose Edit→Paste.

Dreamweaver pastes the original HTML files into the Archive folder.

10. Create a subfolder in the Archive folder named Images.

This step isn't necessary if you're using the same images in the revised pages. If, however, you're uploading new images with the same file names, archive the original images so they'll be available if the client ever decides to revert to the original pages.

11. Select all the images in the site's Images folder.

You are storing your images in a separate folder, aren't you?

12. Choose Edit→Copy.

Dreamweaver copies the images to the Clipboard.

13. Select the Images folder.

14. Choose Edit→Paste.

Dreamweaver pastes the images into the folder.

With the original files and images safely archived, you can modify the Web site. To revert to the old page, delete the revised file and then drag the page from the Archive folder, which replaces the deleted revision. Do the same with the archived images, and everything is as it was. When you do the next update, create a subfolder of the Archive folder, and name it Revision_1. Store the files from the previous revision in there. Archiving the latest three revisions should be sufficient unless you have a client with a photographic memory. In case of server problems, consider keeping a copy of the archive on your local machine or on a CD or DVD. Or, if you're revising a site that's slightly smaller than Rhode Island, archive the files to an external hard drive.

Sharing the Work with Contribute

If your client's personnel will be responsible for some of the updates, chances are they won't know how to use a sophisticated application like Dreamweaver. However, they will likely be familiar with Office applications, such as Microsoft Word. The goal of the people who created Adobe Contribute CS4 was to create an application that anyone familiar with Office applications could use to update a Web site. Adobe Contribute CS4 is part of the CS4 Web Design Premium suite, and may also be purchased separately. The application can be used to browse to a Web page. (See Figure 2-2.) If the user has a connection to the Web page, she can edit the page with the click of a button. That's right: The page can be edited on your computer, just like you were editing a Word document. When you're satisfied with the changes, click Publish, and the edited page is uploaded to your server. One of the other cool features about Contribute is that the application enables the user to roll back to a previous version of the page, which is a blessing if one of your client's staff really messes up a page while editing it.

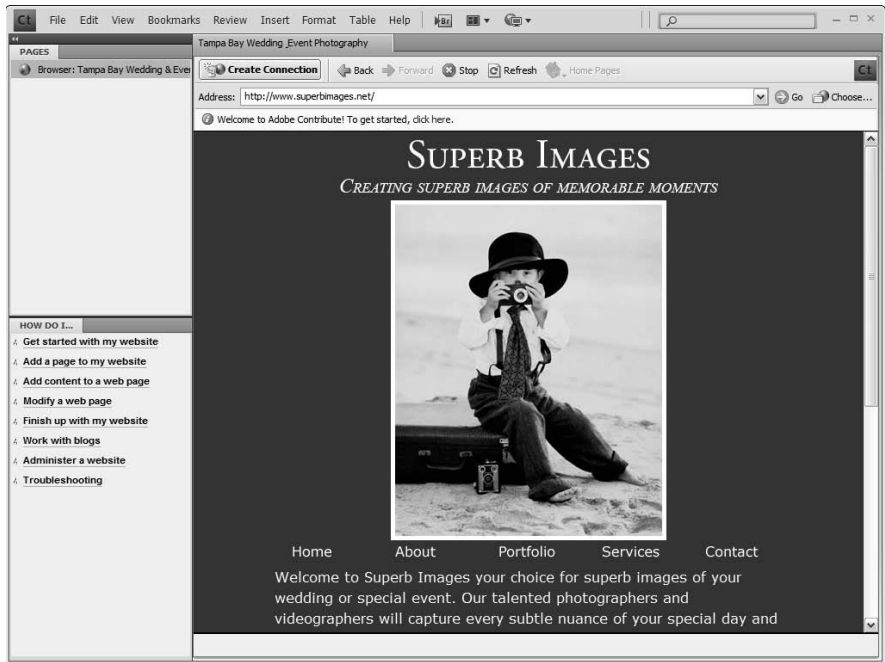


Figure 2-2:
You can use
Contribute
as a
browser.

Creating a connection

Before you can edit a Web site with Contribute, you must make a connection to it. This is similar to defining a site in Dreamweaver, but different enough to warrant some instruction. To create a connection to a Web site with Contribute, follow these steps:

- 1. Use Contribute to navigate to the home page of the site to which you want to make a connection.**
- 2. Click the Create Connection button.**

The first page of the Connection Wizard appears. (See Figure 2-3.)

- 3. Click Next.**

The second page of the Connection Wizard appears. (See Figure 2-4.)

By default, this page shows the URL to which you have navigated. If this isn't the home page, enter the URL to the home page in the text field. Alternatively, you can click the Browse button and browse to the site's home page.

- 4. Click Next.**

The third page of the Connection Wizard appears.

Figure 2-3:
The
Connection
Wizard
makes a
cameo
appearance.



Figure 2-4:
Get ready
to be
connected.



5. Choose the method by which you'll connect to the Web site from the drop-down menu. (See Figure 2-5.)

If the site is a remote host, your choice is FTP (File Transfer Protocol). If you're configuring Contribute for an intranet, the choice is Local/Network.

After you make a choice, the Connection Wizard reconfigures.

6. Enter your credentials.

Figure 2-6 shows the Connection Wizard as configured for an FTP connection with FTP server, user name, and password entered.

7. Click Next.

Contribute tests the connection. (See Figure 2-7.) After verifying the connection, the Connection Wizard reconfigures.

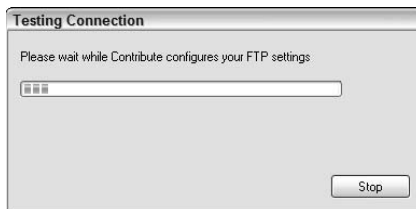
Figure 2-5:
There's a method to this connection madness.



Figure 2-6:
You can't get anywhere without credentials.



Figure 2-7:
Testing:
1, 2, 3, 4.



8. Enter the root folder in which the Web site files are stored. (See Figure 2-8.)

Alternatively, you can click the Choose button to display folders on the remote server.

Figure 2-8:
Your test is successful. Enter your root folder, please.



9. Click Next.

Contribute tests upload speed and performs other diagnostic tests on the remote server. After performing the tests, the Connection Wizard reconfigures, as shown in Figure 2-9. On this page, you enter your user information. The information that Contribute needs is your name and e-mail address. The user name and e-mail address is how Contribute identifies users who are allowed to edit the Web site.

Figure 2-9:
Who are you?
Who, who are you?



10. Enter your user information and then click Next.

The Connection Wizard reconfigures to show the Summary page, which summarizes the information you submitted in the previous steps. If the information is accurate, go to the next step. If not, click the Back button as needed to navigate to the page that contains the information that you need to change.

11. Click Done.

The connection information is stored on the remote server. The Create Connection button becomes the Edit Page button.

Congratulations! You made a connection to your client's Web site, which can now be edited with Contribute. Your next step is to administer the site and determine whether you'll accept the role of administering the site from this point on, or transfer this role to one of your client's lackeys, um, personnel.

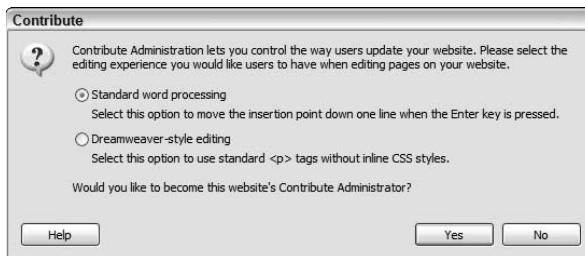
Administering the Web site

After making a connection to a Web site, you *administer* the site. In essence, you set permissions for the people who will edit the site. When you set permissions, you can limit the type of editing each team member can do, as well as set limits on the size of images and so on. To administer a Web site to which you've made a connection, do the following:

1. Choose **Edit** → **Administer Websites** and then choose a Web site to which you made a connection.

All sites to which you've been the first to connect to show up on a submenu of the Administer Website menu. After you choose a Web site, a dialog box appears asking you what type of editing you want for the site and whether you accept the role of administrator. (See Figure 2-10.)

Figure 2-10:
Will you be my administrator?



2. Choose an editing option by selecting the applicable radio button.

We strongly advise you to choose the Standard Word Processing option because it's the easiest for most office personnel to learn.

3. Click Yes to become administrator of the site.

Alternatively, you can choose No and bestow the honor on someone else.

After you make a choice, the Administer Website dialog box appears, as shown in Figure 2-11. The Users and Roles section of the dialog box appears by default. This shows all users who have a connection to the Web site and the roles they play. If you select a user role and click the Edit Role Settings button, you can set permissions for this type of user. The options are largely self-explanatory. You can limit the sizes of images that are added to pages, the type of editing allowed, the type of code Contribute writes when a user from the role group edits a page, and so on. You can also create new roles or remove roles from this section of the Administer Website dialog box.

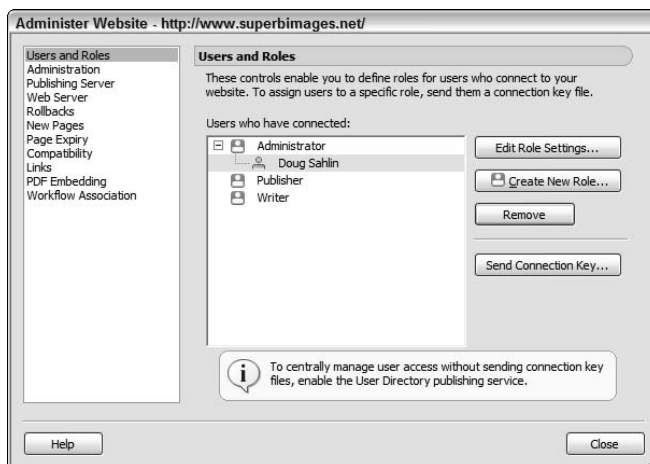


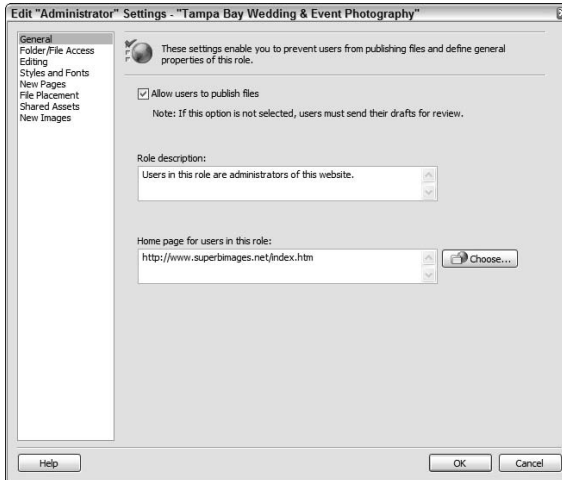
Figure 2-11:
Administer
me, baby!

4. Click Edit Role Settings.

The dialog box reconfigures, as shown in Figure 2-12. By default, Contribute has three roles. The first person who connects to the site assumes the role of administrator. The roles are defined as follows:

- **Administrator:** Users who oversee the site. More than one person can have administrator rights. If you're teaching your client how to use Contribute, set yourself up as an administrator. That way, when your client panics — and he will — you can log on to show him where he went wrong.
- **Writer:** Users in this role can edit Web pages, but cannot publish them to the site.
- **Publisher:** Users in this role can edit Web pages and publish them to the site.

Figure 2-12:
Preparing
to edit role
settings.



5. Accept the default settings or edit them.

As administrator, you obviously want the option to publish files. When you edit roles for other users, though, you can prevent them from publishing edited files by restricting them to a specific page from which they can begin editing.



You modify settings for any role by selecting the role and then clicking Edit Role Settings. If you decide that three aren't enough, click Create New Role and follow the prompts to name the new role and define role permissions.

6. Modify the other sections of the settings dialog box.

You can edit the following sections:

- *Folder/File Access:* The options in this section let users in the role you're editing edit files in any folder, or you can specify the folders from which users can edit files. You can also disable deleting files and enable users to remove rollback versions of pages stored on the site.
- *Editing:* Here, you specify how much editing users in this role can do, specify paragraph spacing options, and other editing options.
- *Styles and Fonts:* In this section, specify whether users in this role can edit styles, apply styles, apply fonts and sizes, change formatting (use boldface, italics, or underline), and change font color and background color.
- *New Pages:* In this section, you specify whether users in this role can create a blank page, use built-in starter pages, and create new pages by copying existing pages. You can also specify which pages can be used as the basis for a new page as well as whether to use Dreamweaver templates that have been uploaded to the server.

- **File Placement:** Here, specify the file placement rules for the following: images, style sheets, Microsoft and PDF files, and all other placed files. File placement rules enable you to specify the folder in which files of type are saved. You can also add, edit, or delete file placement rules.
- **Shared Assets:** This section enables you to specify which files on the site can be shared by users with this role setting.
- **New Images:** This section enables you to specify whether Contribute Image editing can be used when inserting files, and also to specify the maximum dimensions for files that are uploaded to the site (Contribute resizes them when they exceed the dimensions you specify) as well as the maximum quality of JPEG images. You can also enable an option to reject images that exceed the maximum file size you specify.

7. Click OK to apply the settings.

When you change settings, they are uploaded to the server. You need to repeat this option to specify role settings for site Publishers and Writers. After changing settings, you're back to the Administer Website dialog box.

8. Click Administration and then click Set Administrator Password.

The Change Administrator Password dialog box appears, as shown in Figure 2-13.

Figure 2-13:
Password,
please.

9. Enter a password and then confirm the password.

10. Click OK.

The Change Administrator Password dialog box closes. The password you entered has to be entered any time a user tries to choose the role of administrator when making a connection to the Web site.

11. Click Rollbacks.

The Rollbacks section of the Administer Website dialog box appears. This section of the dialog box allows you to enable rollbacks. When you enable *rollbacks*, Contribute makes a copy of a page as soon as a user clicks the Edit Page button. A user can roll back to a previous version of the page, which is a handy option if the page gets messed up eight ways

to Sunday or the boss decides he doesn't like the employee's handiwork. The default number of rollbacks is three. Enter a higher number if you think the client will need it.

12. Set options in other sections of the Administer Website dialog box as needed.

The other sections of the Administer Website dialog box are easy to understand. A detailed explanation of each section is beyond the scope of this book.

13. Click Close.

Your settings are uploaded to the server.

Sending a connection key

If your client's personnel are techno-phobic, they probably know nothing about FTP credentials and all that stuff. In fact, if your client's personnel are really young, they might think FTP are the initials of a rapper. But we digress. When you're using Contribute with nontechnical people, you send them a connection key. The connection key is their gateway to Contribute nirvana, or something like that. A *connection key* is a no-brainer way to connect to a Web site. To send a connection key, follow these steps:

1. Choose Edit→Administer Websites and then choose the Web site you want to administer.

The Users and Roles section opens by default.

2. Click the Send Connection Key button.

The Connection Key Wizard appears. (See Figure 2-14.)



Figure 2-14:
Let's get
somebody
connected.

3. Accept the default option to send your current settings, and then enable the Include My FTP Username and Password check box.

When the user receives the connection key, the user name and password are encrypted; the recipient won't be able to decipher this information. Alternatively, you can customize the connection settings for other users, which means you can give them a unique user name and password.

4. Click Next.

The Role Information page of the Connection Key Wizard appears. (See Figure 2-15). If the person you're sending the connection key to will only be editing pages, choose Writer. Or, if the person will also be creating new pages, choose Publisher. Alternatively, you can choose to add another person to become an administrator. This option is handy if you and your client will jointly administer the site.

5. Choose a role and then click Next.

The Connection Key Information page of the Connection Key Wizard appears. (See Figure 2-16.)

6. Choose the manner in which you're going to send the connection key.

You can send a connection key via e-mail, choose to save the connection key to your local machine for distribution via floppy disk, or upload it to a secure folder on a local server. The wizard recommends that you don't send the key via Web-based e-mail, which is an excellent suggestion because it isn't secure. If you're dealing with a local client, save the key to your local machine, transfer it to a Flash drive, and then deliver it to the client when you're showing her personnel how to use Contribute. If your client is distant, send the floppy disk via the most convenient method. Send the password under separate cover.



Figure 2-15:
Time for a
little role
playing.



Figure 2-16:
How are you
sending the
connection?

7. Enter the password that the recipient will need to unlock the connection key and then confirm the password.

8. Click Next.

The Summary page of the Connection Key Wizard appears.

9. Review the information for accuracy. If something is incorrect, click the Back button to navigate to the page where you need to change the information.

10. Click Done.

If you specified saving the key to the local machine, you're prompted for a location in which to save the key. If you already have a floppy disk loaded, save the key to your floppy drive. If you've specified sending the key via e-mail, your default e-mail application opens. Fill in the blanks and let the e-mail fly, Guy.

Editing Web pages

After you have your client's Web editing team up and running with Contribute, he can begin editing pages. Editing pages in Contribute is as simple as navigating to the site and then to the page that needs to be edited. The following steps show you some of the basic editing tasks you can perform with Contribute. A full tutorial on all the features of Contribute is beyond the scope of this book. If you need to know more information about Contribute, refer to the online help. The online help can also show you how to create new pages with Contribute. To edit a page in Contribute, follow these steps:

1. Launch Contribute.

When you launch Contribute, the application connects to Web sites to which you made a connection. The sites are listed on the Contribute start page. (See Figure 2-17.)

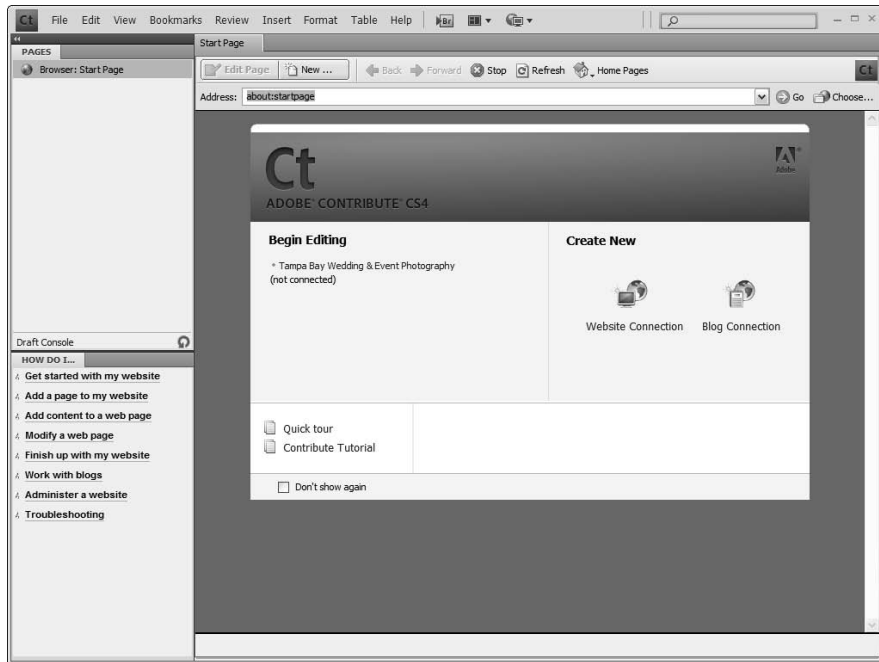


Figure 2-17:
You have
connections.

2. Click the site you want to edit.

Contribute navigates to the site's home page. Alternatively, you can enter the URL to the page you want to edit after Contribute loads.

3. Click Connect.

After navigating to a page on a site to which you made a connection, the Edit Page button appears. (See Figure 2-18.)

4. Use the site navigation menu to navigate to the page you want to edit and then click the Edit Page button.

Contribute creates a draft of the page. (See Figure 2-19.) At this stage, you can edit any part of the page that isn't locked. If you set up the site by creating templates in Dreamweaver, you should have locked areas — such as navigation menus — to prevent users from tampering with your hard work. While in draft mode, you can change text, do minimal image editing, change image properties, and so on. Notice the toolbars at the top of the page. You can use the buttons on them to insert images, create links, and insert tables. Notice the tools for formatting text. They look quite similar to what you find in most word-processing applications.

If a page needs extensive editing, click the Save for Later button. When you do this, Contribute saves the draft to your local machine. You can edit any draft by clicking the desired page in the Pages panel. After editing the page offline, click the Publish button to upload the edited page to the server.





Figure 2-18: Okay. Roll your sleeves up and click the Edit Page button.

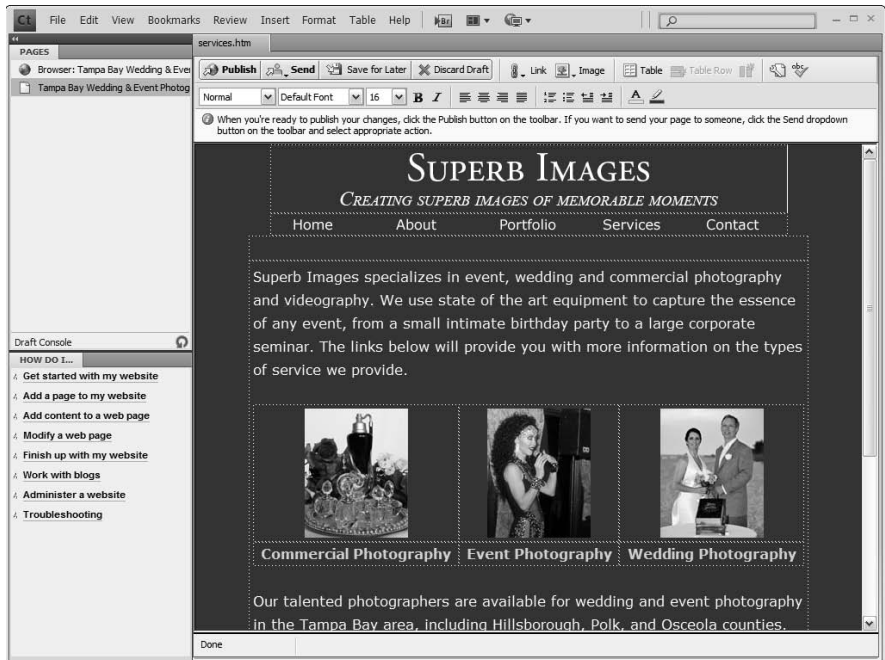


Figure 2-19: Draft Web pages are convenient, but they will never replace draft beer.

5. After editing the page, click the Publish button.

Contribute publishes the edited page to the server. If you enabled rollbacks, Contribute saves the previous version of the page in case you're not satisfied with your edits and want to revert (*rollback*, in Contribute-ese) to a previous version of the page. As we mention earlier, the default number of rollbacks is three.

Managing a Site with Dreamweaver

If you and your staff have been contracted to manage your client's site, the obvious application for editing is the application with which the pages were created. Dreamweaver is a robust application that enables the savvy Web designer to create a way-cool Web site and edit it as well. The following sections show you how to edit your pages in Dreamweaver.

Editing new pages

The beauty of editing pages in a full-featured HTML editor like Dreamweaver is that you can edit the pages as well as the underlying code. When you edit pages in Dreamweaver, you can work in one of three modes: Design, Code, or the chameleon Split mode, which enables you to view the page and code at the same time. Now, how cool is that? To edit pages in Dreamweaver, follow these steps:

1. In the Files panel, select Local View from the View drop-down menu and then select the site you want to edit from the Site drop-down menu.

Dreamweaver displays the files associated with the site. (See Figure 2-20.)

2. Double-click the page you want to edit.

Dreamweaver opens the page in Design mode. (See Figure 2-21.) Editing pages in Dreamweaver is pretty straightforward. The Property inspector is your friend in Dreamweaver. You use it to change links, CSS styles, images, and so on.

3. Edit the page as needed.

If you need to modify the code, click the Code button to display the page in Code view. (See Figure 2-22.)

If you need to view both the code and the page at the same time, click the Split button to display the page in Split view. When working in Split view, you can select items in the Display pane, and the underlying code is displayed in the Code pane. (See Figure 2-23.)

4. After editing the page, click the Put button, which looks like an up-pointing arrow.

Dreamweaver uploads the edited page to the server.

Figure 2-20:
The Files panel is your friend.

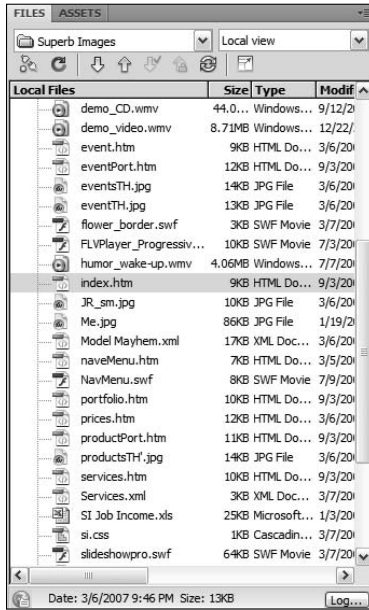


Figure 2-21:
Editing pages in Dreamweaver is, well, a dream.



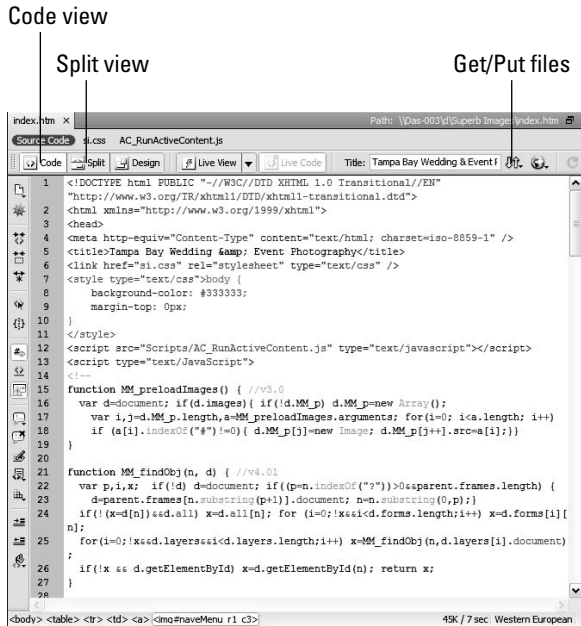


Figure 2-22: Oh, my. This page has exposed its code.

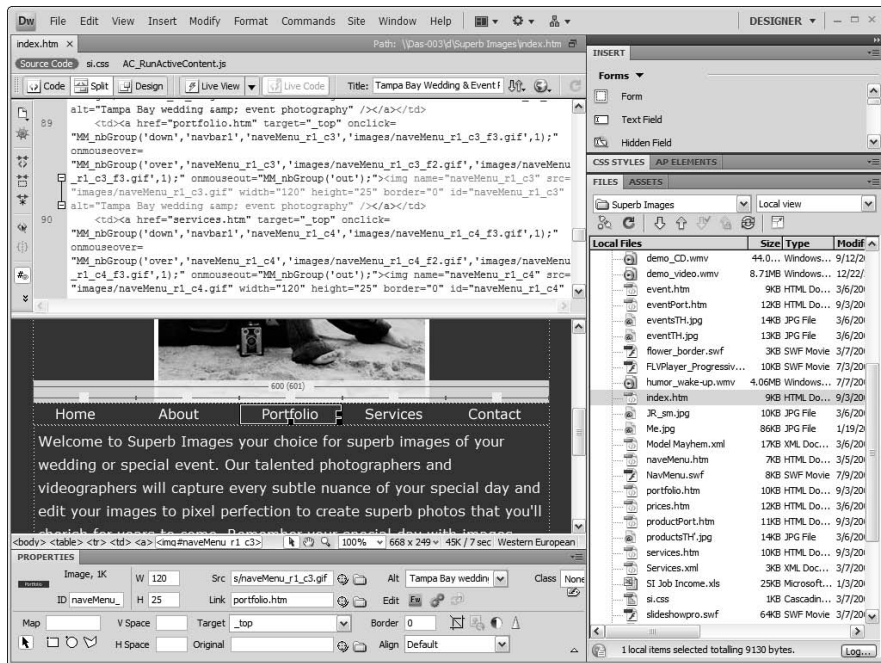


Figure 2-23: Lie down on the couch and tell me more about this split existence.

Templates in Dreamweaver can save your sanity

If your client changes her mind frequently, you might find yourself frequently creating new pages for her Web site. When this occurs, you turn on the meter and start charging your client your going rate for revisions. But why reinvent the wheel? We mean, after all, the new pages will have common elements such as the site banner, the footer, the navigation menu, and so on. Rather than creating these items from scratch every time you create a new page, you can save yourself a lot of time by creating templates when

you create the site. If you create a template for every possible page that you need to create, making a new page is a snap. Choose File→New and then click the Templates tab. Choose the desired template and go to town. You'll save yourself a lot of time by working with templates, as you don't need to re-create the common items that are used in the site. **Hint:** There's no need to pass the savings on to the client. For more information on creating templates, read through Book III, Chapter 6.

Archiving pages

You work hard to create compelling Web sites for your clients. If you follow our sage advice in the “Updating Pages without Destroying Them” section of this chapter, you archive pages on the server. But what happens if the server has a major malfunction (known as a *hissy fit* to your nontechnical client) and all your client's files are erased? Most Web servers have some kind of redundancy, which enables them to restore your client's files. But in the case of a natural disaster, the computers on which the backup files are kept might be toast. You probably have the latest version of your client's pages on your local machine. However, hard drive failure isn't out of the question. To prevent problems, we always archive a Web site to CD after we publish it. If you have sophisticated CD-burning software, you can set up a multisession disc and update the disc whenever you change a page. In either worst-case scenario (a server meltdown or a local machine hard drive meltdown), you still have everything you need to restore your client's Web site on a CD. Oh, yeah, we should mention that CDs are prone to failure as well. Make a copy of the CD and store it in a safe place. Better yet, invest in some archival CDs or DVDs and use them to archive the final version of your client's site. Archival CD or DVD media isn't cheap, but the discs have a life expectancy of over 30 years. It's always better to be safe than sorry.

Tips for Managing Tasks

Many of your tasks as a Web designer are repetitive. At times, you have to enter the same bit of code; other times, you use items that are the same, but different. When you perform repetitive tasks on a regular basis, *Snippets* are a Web designer's best friend. And for items like a site copyright notice that changes once yearly, you can't beat a Library item. Another way you can streamline your workload is with Assets. *Assets* are items, such as Flash

movies and images that you use when designing a site. Assets have their own panel and are neatly organized in folders by content type, ready to use in an existing page or a new page. The upcoming sections show you how to streamline your work when creating or editing pages.

Simplifying repetitive tasks with Snippets

Did you ever wish you could do away with the boring tasks in your everyday life? We mean, wouldn't it be wonderful to step in front of a mirror, push a button, and get your face washed, followed by a close shave and a quick trim? Well, modern grooming technology hasn't gotten that far, but Web design has. If you find yourself doing repetitive tasks in Dreamweaver, such as inserting the same image or the same bits of code in a page multiple times, you'll love Snippets. With Snippets, you can save code that you'll be using frequently in the Snippets panel. You can then use the Snippet in any Web page you create.

Creating Snippets

One task that many e-commerce Web designers do over and over is insert Buy Now links and the associated code into Web pages. Doug's photography Web site sells prints that are different sizes. The transactions are handled through PayPal. He created a table for the popular sizes, with the price information and necessary PayPal code to carry out the transaction. The following steps show how to record PayPal code as a Snippet:

- 1. Open the page that contains the code you want to save as a Snippet.**

You can record any combination of code — even code to insert images, create hyperlinks, and so on. Just make sure that any Assets such as images will be present on the site in which the Snippet code will be used and that the relative path is the same.

- 2. Click the Code button.**

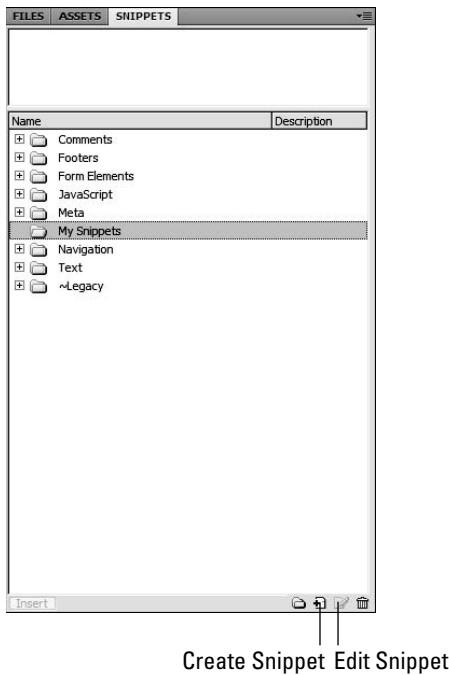
When you create a Snippet, you select the code associated with the task you want to convert to a Snippet. You can create a Snippet in Designer mode, but when you're recording several lines of code, it's hard to ensure you selected everything in Designer mode. If, however, you're recording a single object, such as text with a hyperlink, you can accomplish this in Designer mode. Otherwise, we suggest that you work in Split mode to ensure that all the needed code is selected.

- 3. Select the desired code.**

- 4. Choose Window⇨Snippets.**

The Snippets panel opens. The Snippets panel consists of several preset Snippets that ship with Dreamweaver. You can also create folders for your own Snippets. We store our Snippets in a folder, oddly enough, named My Snippets. (See Figure 2-24.)

Figure 2-24:
The Snippets are not related to The Muppets.



5. Select the folder in which you want to store the Snippet.
6. Select the code you want to convert to a Snippet.
7. Click the New Snippet icon.

The Snippet dialog box appears.

8. Name the Snippet (see Figure 2-25) and then click OK.

The Snippet is added to the desired folder.

Using Snippets

After you amass a panel full of spicy Snippets, you can liberally sprinkle them on the new pages you create or the pages you edit. And you do so with the simple click of a button. To use a Snippet in a Web page, do this:

1. Create a new page or open an existing page where you want to add a Snippet.

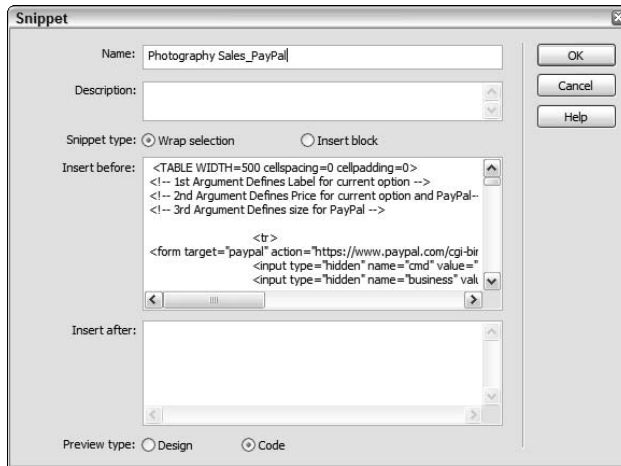


Figure 2-25:
We hereby christen thee, *Joe Snippet*.

2. Choose Window → Snippets.

The Snippets panel opens.

3. Navigate to the folder that contains the Snippet you want to use.

4. Drag the desired Snippet onto the page.

You can do this in any mode. If you decide to add a Snippet while working in Code mode, make sure you position your cursor in the proper place. Otherwise, you run the risk of messing up the page. Of course, you can always undo inserting the Snippet.

Figure 2-26 shows a page in Split mode where the PayPal Buy Now Snippet was added to a page. The only thing that you'll need to change is the PayPal item number, but that's a small price to pay for not having to enter all that code.

The Snippets panel is a powerful feature that can save you lots of time. In addition to recording your own Snippets, you'll find several useful preset Snippets neatly tucked away in folders. If you don't like JavaScript because of all the code work, check out the JavaScript folder in the Snippets panel. When you start using some of these gems, your clients will think you're a genius. Don't let on that you got by with a little help from a friend.



To edit a Snippet, select it and then click the Edit Snippet button at the bottom of the Snippets panel. This opens the Snippet dialog box. Edit the code as needed and then click OK to save the edited Snippet.

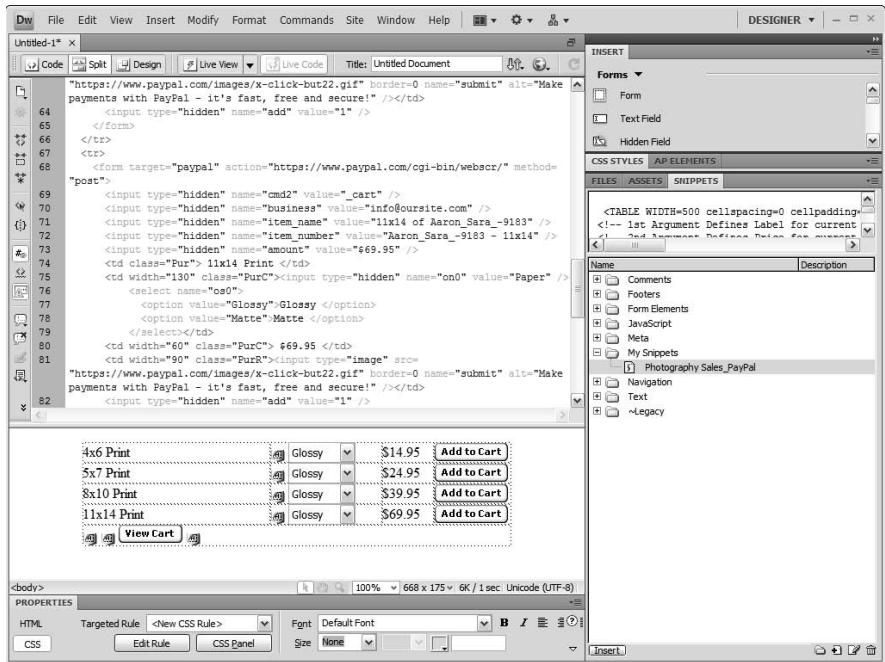


Figure 2-26:
Instant Snippets. Just add water.

Taking advantage of your assets

Everything's got assets, including the Web pages you design. Whenever you define a site and begin creating pages, the images, colors, Flash movies, hyperlinks, and so on are saved in the Assets panel. Whenever you want to use an Asset on a new page, all you need to do is grab it by the scruff of the neck and drop it on the page. What could be simpler? And if you really, really like an Asset and think you'll use it on other pages for the site you're creating, you can save it as a favorite. To use Assets, follow these instructions:

1. Choose Window → Assets.

The Assets panel opens. The panel has several icons on the left side, which, when clicked, display all the Assets in the categories. When you click an item such as an image, the image is displayed in the window at the top of the Assets panel. (See Figure 2-27.)

2. Select the desired Asset and drag it onto the page.

To use an Asset on any Web site, select the Asset and then click the Add to Favorites icon. To access the favorites for a Web site, click the Favorites icon to reveal the site favorites for that category.

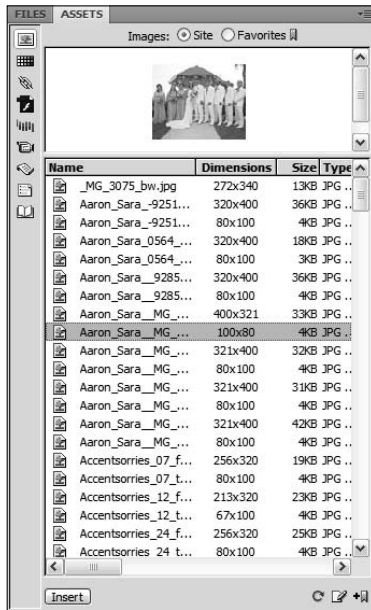


Figure 2-27:
Hot, roasted Assets; click an icon to get your hot, roasted Assets.

Using the Dreamweaver Library

The Dreamweaver Library is a wonderful thing. Anything you add to the Library can be used on any page you create within the site. The really cool thing about Library items is that when you edit them, you have the option of updating the pages that already contain the Library item, which is a perfect solution if you have a copyright or anything else that's date sensitive and needs to be updated in multiple pages.

Creating Library items

Consider creating a Library item anytime you have an object that will be used in multiple Web pages. The beauty of a Library item is that you can use it on any Web page in any Web site. To create a Library item, follow these steps:

1. **Open the Web page that contains the item you want to convert into a Library item.**

We include a copyright notice on most of the Web sites we create. When we finish creating the home page, we add the copyright notice to the bottom of the page and then convert it to a Library item.

2. Choose **Window**⇨**Assets** and then click the **Library** icon (that looks like an open book).

The Assets panel opens to the Library section. (See Figure 2-28.)

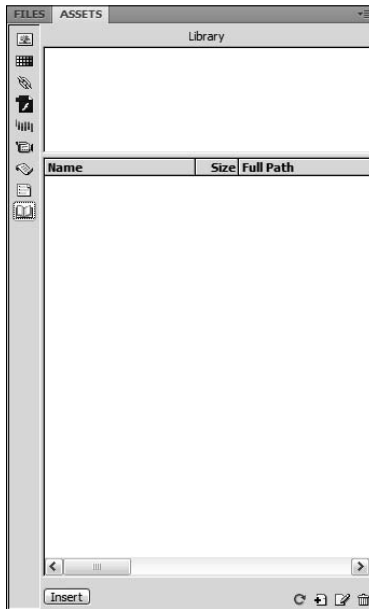


Figure 2-28:
The Dreamweaver Library has no librarian.

3. Select the item you want to convert to a Library item.

When you select an item, such as a menu with links — or in this case, a copyright notice with HTML tags — it helps to view the page in Split mode. This ensures that you select the object and the tags that format the object. In the case of the copyright notice, we also want to include the tags that center the text and format it as paragraph text. (See Figure 2-29.)

4. Drag the item from the Web page into the Library when working in Design view.

Alternatively, you can click the New Library Item icon. The item appears in the Library as Untitled.

If you add an item to the Library that's formatted with a CSS, a warning dialog box appears telling you the item might not look the same on other pages because the CSS formatting information is not copied with the item. In the case of text, the formatting defaults to the HTML formatting for the text object.



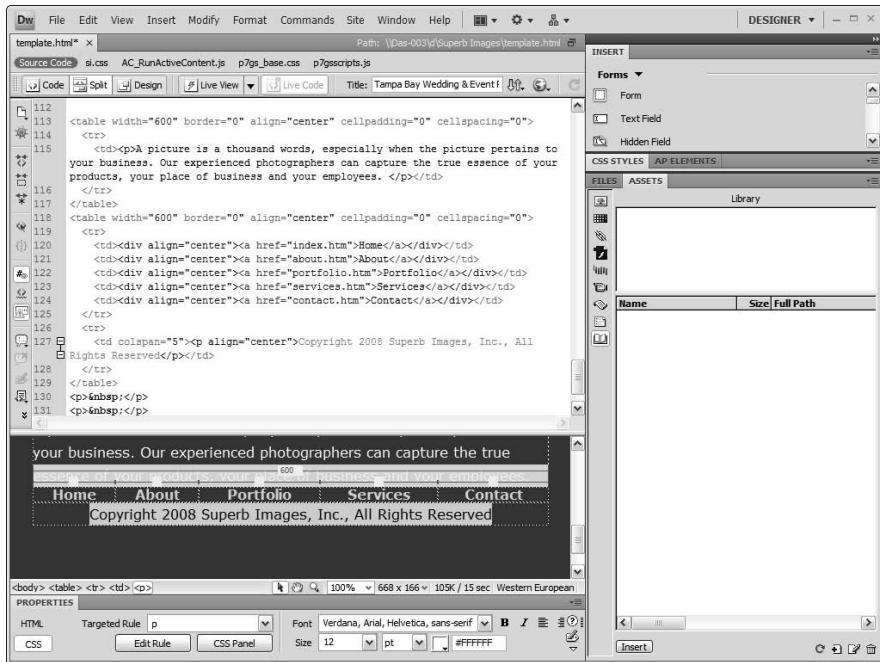


Figure 2-29: This code is going to the Library.

5. Enter a name for the item.

We use a descriptive name that's easy to decipher when using the item in another Web page. For example, if you're managing multiple Web sites and each one has a copyright notice, precede the copyright notice with the client's name. Figure 2-30 shows a copyright notice as it appears in the Library.

Using Library items

If you set up your Library items when you create the first page of a site, you can easily add it to other pages in the site. To add a Library item to a Web page you're editing, follow these steps:

1. Position your cursor where you want the Library item to appear.

If the page includes a lot of formatting tags or code specific to items on the page, view the page in Split mode to ensure that the Library item is placed in the correct position.

2. Choose **Window** ⇨ **Assets** and then click the **Library** icon.

The Assets panel opens to the Library.



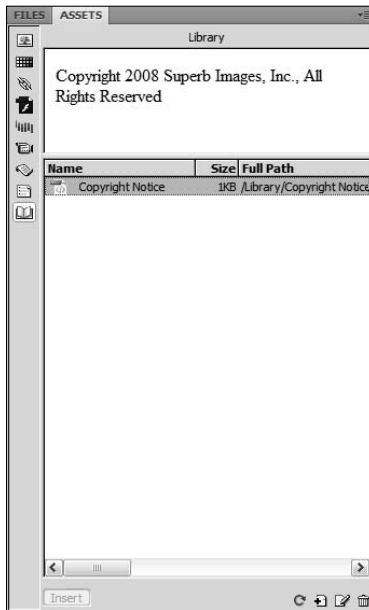


Figure 2-30:
A copyright notice in the Dreamweaver Library.

3. Drag the Library item onto the page.

The Library item is added to the page. Text items appear as a white rectangle in Display mode (see Figure 2-31) but display perfectly in a browser.

Editing Library items

You can edit a Library item at any time. When you edit a Library item, any pages that use the Library item can be automatically updated, which is a tremendous timesaver. To edit a Library item, follow these steps:

1. Choose **Window** ⇨ **Assets** and then click the **Library** icon.

The Library section of the Assets panel appears.

2. Select the item you want to edit.

3. Click the **Edit** icon that looks like a pencil on a piece of paper.

The Library item appears in the main window of the Dreamweaver workspace. (See Figure 2-32.)

4. Edit the file.

5. Choose **File** ⇨ **Save**.

The Update Library Items dialog box appears. (See Figure 2-33.)

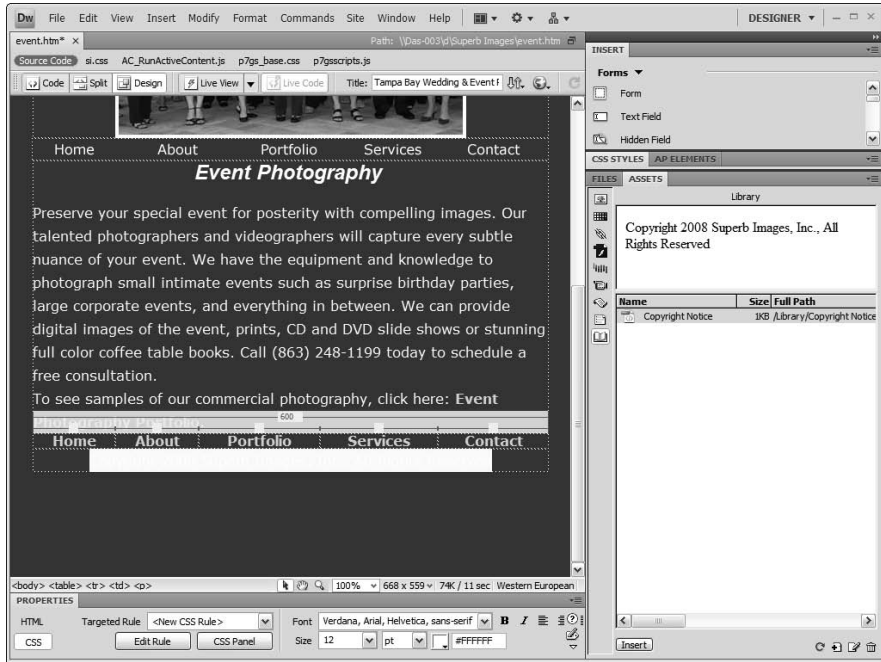


Figure 2-31:
A Library
item is
added to
the page.

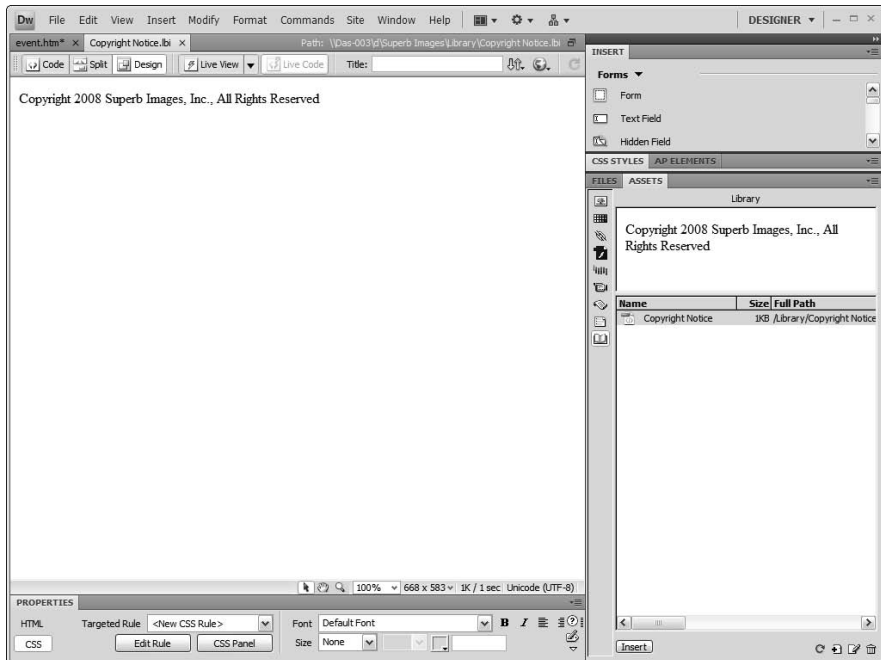
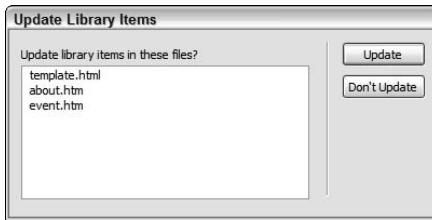


Figure 2-32:
A Library
item ready
for editing.

Figure 2-33:
Whoopee!
I'm gonna
update
some pages.



6. Click the Update button.

All pages that contain the Library item are updated. Alternatively, click the Don't Update button if you don't want to update the pages.

Chapter 3: Expanding a Site

In This Chapter

- ✓ **Deciding between expanding or adding on**
- ✓ **Starting anew from scratch**
- ✓ **Refurbishing an existing design**
- ✓ **Publicizing the relaunched site**

Like a toddler using a bike with training wheels, there comes a time when your client needs to make the next step, knock off the training wheels, and ride with the big boys. When this blessed event occurs, you have a decision to make: namely, whether to

- ◆ Update the existing design.
- ◆ Start from scratch and redesign.

Naturally, many factors can be involved in your decision, such as how long the site has been public or how many changes your client is requesting. After you make your decision, you have to craft the revision or redesign so you don't shut down the client's current site. If you're in an expansion frame of mind, this chapter shows you the factors to consider when expanding the site. Doug takes the lead in this chapter, showing you the steps he took to update his own site.

Use It or Lose It: Should You Add On or Redesign?

You designed a wonderful Web site for a client; you uploaded the design and went on your merry way, designing sites for other clients. Several months later, the client comes back to you, and guess what? You have another gig. He wants you to do more work on the site. The Web site is very important to your client, and he wants you to add information about his new products and services. You meet with the client, and he lays out the grand vision for the site expansion. Now, you're faced with an important decision. Do you add on to the existing site or redesign it from scratch?

Your value as a Web designer is your experience, the sites you've designed since you created the last iteration of your client's site. You've learned more, experienced more, and have grown as a designer. After conversing with your client, you'll know whether it makes more sense to add on to the existing content — inherently the least expensive option — or to put your expertise and creativity into a total redesign.

Adding on to a site

Your client has given you some cool new stuff that he wants included on the site, and it's time to expand his Web site. An obvious option for you, as a designer, is to simply add on to the site. This route allows you to keep the look and feel of the current Web site with a virtually seamless expansion (and relatively pain-free for the client and his business). After all, your client's visitors — who are familiar with the features — are coming to a place they know. Consider the following indicators as good signs to know that adding on to an existing site is your best option:



- ◆ **The site is less than a year old and doesn't look outdated when compared with the competition.**

Quick fix: If the site sports outdated features (such as a Flash intro or beveled buttons), remove them. Like beehive hairdos, these features have gone the way of the dodo. If a Flash intro resides on a page (such as `default.htm` or `index.htm`) and links to the home page of the site, simply rename the home page to `default.htm` or `index.htm`. Poof! The Flash intro is gone.

- ◆ **You can do the expansion without a massive renovation to the existing navigation menus.**
- ◆ **Your client wants you to only add additional features, such as a forum or *blog* (Web log).**

No brainer. You can easily accomplish this by using the existing site framework.
- ◆ **The current site has been received well by the client's customers, and the expansion can be accomplished by using the existing site framework.**
- ◆ **You can expand the current site without incurring usability issues or otherwise confusing visitors.**
- ◆ **You can redesign the site if it is warranted and the cost doesn't exceed your client's budget.**

You did ask the client what her budget is, didn't you?

Redesigning the site from scratch

Redesigning a site from scratch is almost the same as designing a new Web site. The obvious advantage is that you've already worked with the client, so you know her foibles and follies. Figure 3-1 shows my Web site prior to redesign. Web design is still a major part of my business. However, I now work in conjunction with other Web designers and developers, so my personal

site focuses on my photography. My Web design work comes from word of mouth and projects with other designers and developers. When I redesigned the site for the last edition of this book, white text on black background was fashionable. When I revised the site for this edition of the book, I decided to go with a simple background and a Flash slide show of images. The current content is black text on a white background. I added a little color with a red logo of my initials. The page background is a neutral gray, which is fitting for a photographer's Web site. The redesigned Web site is shown in Figure 3-2.

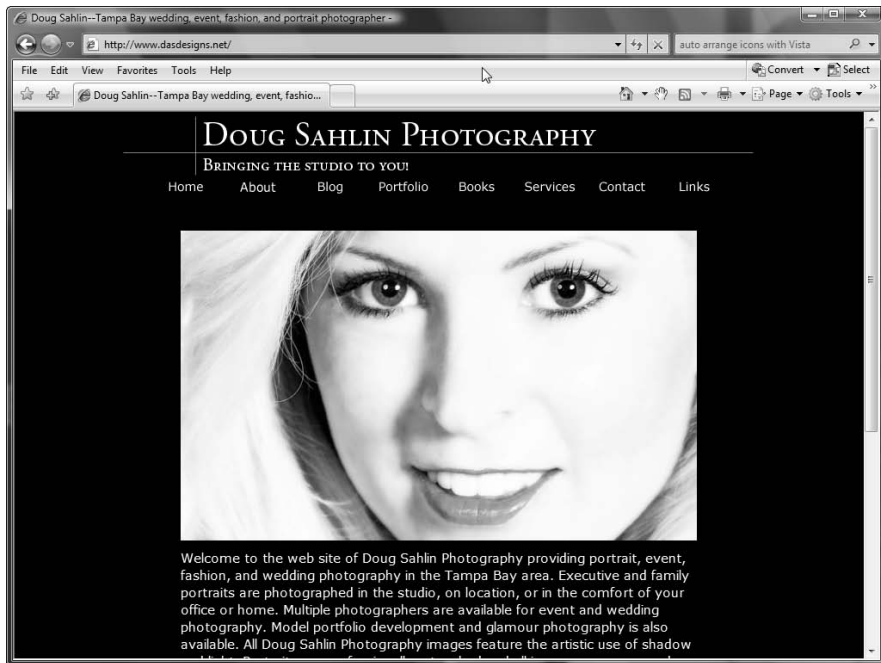


Figure 3-1:
Bend me,
shape me,
any way you
want to.

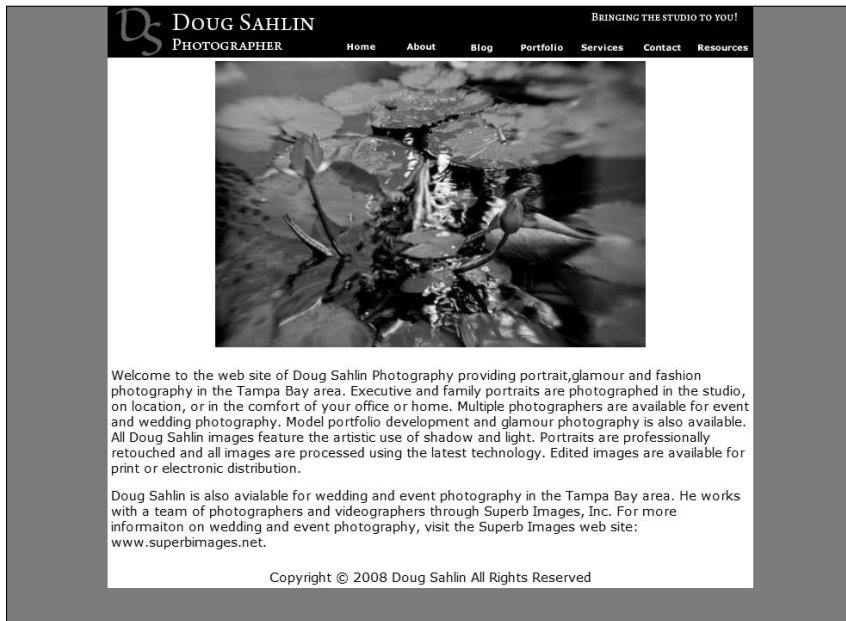
Here are some pretty good indicators for deciding that a site needs to be redesigned instead of just updated:

- ◆ **The site is more than one year old and appears dated.**
- ◆ **The site isn't up to snuff with Web sites of your client's competitors.**

You can either ask your client to send you the URLs to the sites of his competitors, or do your own research for which you charge your hourly rate.

- ◆ **The site no longer accurately reflects your client's main focus.**

Figure 3-2:
The URL is the same, but the content has been changed to protect the innocent.



- ◆ **An expansion will cause usability issues such as cluttered menus, hard-to-follow navigation, and confusing content.**
- ◆ **You already made substantial changes and added new features prior to the client's expansion request.**

Upside: Redesigning the site from scratch enables you to use the latest technology and create a state-of-the-art redesign for your client. This should strongly be considered if the site is over a year old, technology has taken a quantum leap, and your client's budget can stand the burden of a total redesign.

- ◆ **The current Web site can't easily accommodate your client's expansion plans.**
- ◆ **Your client's intended audience is accessing the Internet with faster connections.**

Quick fix: Incorporate streaming Flash video, audio, and other interactive features.



Your client might expect a price break because you previously did work for her. If a price break or discount comes up in the course of discussion, you have to put your foot down and tell the client that a redesign from scratch involves just as much work as a new site design. In fact, if it's been more than a year since the initial site design, your price should be higher.

Headache-Free Expansion

When you expand a site, everything must go off without a hitch. Whether you revise the existing site or redesign the site, the finished product must be error free. This is even more important when you're expanding a site because visitors are used to surfing the current site — which, if you did your homework and tested the original design — is error free. (Ahem. You did test the original site, didn't you?)

If the site is a redesign, you must consider other things before jumping in with both feet and cranking out new Web pages. In the upcoming sections, I give you some food for thought when it comes to redesigning a site. I also show you how to make sure the site expansion goes off without a hitch.

Things to consider when redesigning

When you redesign a site that's been up for a while, you do so because the client has new material to show site visitors and because the existing site is getting a bit long in the tooth. When you redesign a site, you need to put the same effort into the redesign as when you created the original site for the client. You need to see what's out there. In other words, do your research and view sites of businesses or organizations that are similar to your client's. Just because the first site you designed for the client was state of the art doesn't mean that a redesign that's simply a rehash of the original design will draw "oohs" and "aaahs" from people who visit the new site.

While you're doing your research, you should also check out the current sites of your client's competition. If your client is in a dog-eat-dog business, chances are that his competitors' sites have significantly changed since you viewed them while designing the original site. And, for that matter, your client's top competitors might have changed since you did the original design. Check with your client to see who his top competitors are. Also ask your client which of his competitors' sites he likes the best and why. Then incorporate these features with your creative talent to design a site with a similar look and feel with your own unique spin.



If your client is in a dog-eat-dog business, she should not wear Milk Bone undergarments.

Here's the logical workflow for a site redesign that includes the phases described in the following sections:

1. Consultation
2. Review
3. Analysis

4. Planning
5. Implementation
6. Testing

Consultation

Meet with your client to get the scope and breadth of the client's expansion. Get as much information as you can from the client. This meeting is very similar to your initial consultation with the client except that you know her better now and can talk frankly with the client based on your previous experience.

- ◆ **Find out everything about the client's current needs.** What are her goals for the revised site? Also, ask her to think about future expansion.
- ◆ **Discuss the previous site.** Your client is bound to have feedback from site visitors. Ask her which pages and sections work and which ones don't. The site stats can also give you an idea of which pages are popular and which ones are not.
- ◆ **Address the client's intended audience.** Is the intended audience the same as when you did the original site, or is the audience different or expanded based on the new information that will be incorporated in the redesign?
- ◆ **Address your client's current competitors.** Is she competing against different businesses now, or does the list of the client's competitors expand along with the Web site expansion?
- ◆ **Determine your client's timetable for going live with the redesign.** You need to incorporate her timetable with the other phases of the project and your current workload. If the timetable isn't reasonable, become the voice of reason and tell the client what the timetable will be based on the scope and breadth of the redesign.

Review

Review the client's existing site in conjunction with the desired expansion. During this review, you're looking for answers to these questions:

- ◆ **What does the client's current site say to the intended audience?**

When reviewing the information, make note of which information is concurrent with your client's new goals, and which is not.
- ◆ **Will the intended audience change based on the new information you'll incorporate with the client's current site?**

If the audience will change, consider what content from the old site is appropriate for the new site and also whether you need to change material based on the demographics of the new audience.

◆ **Can you incorporate the current graphics into the redesign?**

You might need to change the navigation menu or site banner based on your client's new goals.

◆ **Has the client changed her logo?**

If so, incorporate the new logo and colors with your redesign.

◆ **Can you improve navigation and the organization of information?**

If the client's audience complained about any items on the old site, now is the perfect time to change them.

◆ **Which pages are the most popular and which pages have the fewest hits?**

You can get this information from the site stats. If the pages that have the fewest hits contain important information, you have to tweak these pages during the redesign.

◆ **Which pages have been bookmarked by site visitors?**

You have to incorporate a version of bookmarked pages in the new design, and these pages must have the same URL. Your Web site statistics may include this statistic. Your client can also use third-party companies to track statistics, or your client can sign up for Google Analytics. Google Analytics doesn't show bookmarked pages, but it will give you extensive information about which pages are visited most frequently, how long visitors stay on the site, and so on. As of this writing, the service is free. For more information, visit www.google.com/analytics/.



You want to include some familiar items in order to provide some continuity to people who visited the original version of the site. Two of the most common items in a design are the banner and the footer. If possible, include both in the redesign so your client's Web site visitors won't think they landed on the wrong Web site.

Analysis

Analyze the information you gleaned while reviewing the site and then incorporate that with the content that the client wants incorporated with the redesign. Determine whether this information can be incorporated with that presented on the client's current Web site or whether you need to create new pages for the material.

Analyze the client's intended audience. Has it changed since your first design for the client? Has the technology used by your client's intended audience changed since your first design? Has Web technology changed since your first design? The answers to these questions shape your redesign. You also need to know what bandwidth your client's audience is using to access the Internet, their average computer screen desktop size, and so on.



A redesign is more than simply rehashing material from the old site. Although you do want to keep some continuity, analyze the material that the client is presenting to you for the redesign. Unless you're forgetful or have more business than you know what to do with, you're bound to remember the content of the old site. If not, analyze the content of the current site thoroughly before examining the new content proposed by the client. After reviewing the new content, come up with a plan to marry the old with the new.

Planning

This stage of the redesign is similar to creating a mockup for client review. Incorporate all the information you gathered to create a user-friendly navigation system that combines the old and new. After you jump this hurdle, consider how you'll organize the information on each page. The information you got while consulting with the client, reviewing the client's existing pages, and analyzing the other important factors in the redesign shows you which information is the most relevant in the redesign. This information must be easily accessible by your client's intended audience.

You have to address other issues in your redesign, such as the following:

- ◆ **Whether the current site's server has enough features to implement your redesign:** For example, if your redesign incorporates a database, does the server have provisions for creating a database? Alternatively, can you upgrade the features on the server side to incorporate the new features in your redesign?
- ◆ **What software is needed:** For example, if the client asks you to include a site-wide search engine, you need to find the applicable software — and include the cost plus markup of the software in your revision.
- ◆ **How much of the old can be wed with the new:** Consistency is important to Web site visitors. Therefore, you should keep some of the old images — and, if at all possible, the site banner. When you plan the redesign, keep the client's intended audience in the forefront of your mind. Your redesign incorporates the client's goals with the intended audience's expectations. In the viewer's eyes, the redesigned site should have value-added services.



First off, archive the original Web site. You might be able to incorporate some of the material in the original design in future revisions of the redesigned site. If the site's Web server allows you to store data, you can archive the old site in a folder on that server. If not, download the old site and save it to DVD or to an external hard drive.

- ◆ **What technology your client's intended audience is using:** Do they have the necessary browser and plug-ins to access the technology you plan on using in the redesign? If not, use a different technology or add links to sites where site visitors can download the plug-ins needed to view the redesigned site. In the long run, your best bet is to plan the redesign around the lowest common denominator; in other words, go with the oldest technology used by your client's intended audience.

Alternatively, you can create two versions of the site: one for the techies who have the latest software plus fast Internet access, and another version for the lowest common denominator. Use a gateway page with links to direct visitors to the site that matches their technology. A *gateway page* describes the resources needed to view each iteration of the site. For example, you can create a Flash site for visitors who have the Flash Player installed on their computer and an HTML site for those who don't. The gateway page has a link to each section.

- ◆ **Whether you incorporated future expansion in the redesign:** Make sure your navigation menus are laid out so that you can easily incorporate additions to the Web site. Use templates to simplify the construction of each page. Templates will also be indispensable when incorporating future expansion into the redesigned site.



Include search engine optimization (SEO) in your redesign. If you don't, the client's expanded site will be virtually invisible in search engine results pages. See Book VIII, Chapter 1 for more about SEO.

Implementation

At this stage of the process, you start with a clean slate in your HTML editor and incorporate your planning to create the actual pages. This is no different than creating a new Web site. You design a *home page* (a brief overview of your client's business, service, or organization) that's compelling enough to draw visitors to the site. Then you must lay out the ensuing pages in a logical manner with the most important information at the top of the page. The main pages in any section of the site should be short, simple, and to the point. Pages that are deeper in a section can have more text.



If you're incorporating Assets (such as images or multimedia files) from the old site, download these to a folder on your computer so you can quickly access them when needed.

Testing

Finally, test the site to ensure a smooth transition from the old to the new. If the site is small, you can test most of it on your local machine. Or, if the site uses ASP, ColdFusion, or PHP, you can test the pages with a testing server on your local machine. At the risk of being blatantly obvious, check everything: navigation menus, links, interactive content, and so on. Another option you can choose is doing an online beta test, as outlined in Book III, Chapter 10. After testing the site, evaluate your findings and tidy up any loose ends or faulty code. Then you're ready to go live.

Trauma-free site expansion

Whether you're expanding existing pages or redesigning the site from scratch, your task is a lot easier if you follow certain steps during the

project. Most of this is common sense, but even the best designer might forget a step or take things for granted because he's worked with the client before. Well, the long and the short of it is to *not take anything for granted*. Otherwise, one of Murphy's Laws will come into play when you least expect it. At any rate, here are a few things to consider when expanding a Web site:

- ◆ **Write a proposal for the expansion project.** After you and the client come to terms with the scope and breadth of the project, prepare a proposal and get the client to sign the proposal before you begin work.
- ◆ **Present a mockup to the client for approval.** Just because you've worked with the client before doesn't mean you know her like the back of your hand. You might be in for a nasty surprise if you assume that she'll fall in love with your expansion, sight unseen.
- ◆ **Use existing pages as the basis for the site expansion.** When you begin the site expansion, download the old pages from the Web hosting service. In a worst-case scenario, you may need to create a new navigation menu to incorporate new sections of the site. You can create a new menu in Fireworks and insert it in the proper place in your HTML pages.
- ◆ **Archive the original Web site so you can refer to the old content if you need to.** If the current site is relatively small, download the site to a folder on your computer.
- ◆ **Maintain links and the page's listing in a search engine index.** When revising a page that's currently on the site, begin with the document you downloaded from the Web hosting service. Change the page as needed and save it, using the same name. This keeps all your links intact and preserves the page's listing in a search engine index.
- ◆ **Keep your client abreast of the project.** If you're uploading your work while you do it, tell the client when the revised work is posted and get the client to sign off on the work. If you're working on a total redesign, upload the content to a different folder at the Web hosting service and ask your client to review and comment on the work you've done so far.
- ◆ **Test, test, test — and then test again.** Make sure everything on the pages you're creating or revising works perfectly. That includes links and any code. A site expansion should be handled with the same kid gloves as a new design.

In fact, if you have the luxury of time and the scope of the expansion is massive, consider doing a full-blown beta test, as outlined in Book III, Chapter 10. In lieu of a beta test, use your friends, employees, or the client's employees to test the revised site.

- ◆ **Upload pages *only after* securing the client's approval.** Never upload pages to the root directory of the Web hosting service until the client agrees that the work is to her liking and after you test the pages to ensure that no problems exist. Your best bet is to upload the new pages to a different folder on the server and send the URL to the client and anyone else who is testing the pages.

- ◆ **Set up a testing server.** If you're creating ColdFusion, PHP, or ASP pages, set up a testing server on your local machine. That way, you can test the pages while you create them.
- ◆ **Work smarter, not harder.** Use the features of your HTML editor to modify the previous work you did for the client as part of the site expansion. If you're using Dreamweaver, read the nearby sidebar, "Making Dreamweaver work for you," to find out how you can whip through a site expansion without breaking a sweat.

After Launching the Expanded Site

You might think the work is done after you upload the revised or redesigned site to the server. Nay, nay, Nanette! Your work — or your client's work — has just begun. Sure, you posted something new (and hopefully exciting) for the world to see. However, like anything else, nobody will know what you did unless you tell them about it. That's right: You've got it, so flaunt it. The following is a list of things you or your client should do to publicize a revised or redesigned site:

- ◆ **Send a press release to the local media.** The press release should include the reason why the client changed the Web site. Don't forget to include information about the client and the client's business or organization. Be sure to include the URL to the site in the press release. If your client enjoys a worldwide audience, several services are available on the Internet that you can use to create a press release. You can find a service that suits your needs by typing "**Internet Press Release**" in your favorite search engine.
- ◆ **Notify members of your client's mailing list that the site has new content.**
- ◆ **Send a newsletter to existing clients telling them about the expanded Web site.**
- ◆ **Request link exchanges.** If your client's expanded Web site contains information pertinent to Web sites not currently linked to the site, request a link exchange.
- ◆ **Include an invitation to visit the expanded Web site in all the client's printed media, including advertisements.**
- ◆ **Re-register the site with the search engines.**
- ◆ **Use other online marketing techniques, such as pay per click ads, to draw visitors to the expanded site.** *Pay Per Click Search Engine Marketing For Dummies*, by Peter Kent (Wiley) can help you (and your client) bring a steady flow of potential customers to your client's site.

Making Dreamweaver work for you

You already created a lot of material for the site you're about to expand. Whether you're revising the old site or redesigning it, there's no sense in reinventing the wheel. The Assets panel is chock-full of items you used on the old site: images, Flash movies, URLs, and so on. To use them in the revision or redesign, simply open the Assets panel and then drag the Asset into the new or revised page. Snippets can also be a great timesaver, especially if the site uses a lot of code. If you added items to the Library (such as footers and copyright notices), you

can use these in the site expansion as well. And last, but not least, consider recycling templates you used when creating the site. If it's a total redesign, you might be able to get by with revising templates. To revise a template, open the document as a DWT file. You can then edit locked regions of the template and make other necessary changes to suit the revision. As an added bonus, when you save the template, you have the option of updating all pages that were designed with the template.



After you launch the redesigned or revised site, pay careful attention to the site statistics to determine the overall success of the expanded site. Notice which pages are getting the most visits and from which pages visitor exit the site. If viewers are exiting the site from pages that contain important new content, you need to revise the pages. The information you learn from site statistics will drive future changes, revisions, and redesigns.



Convince your client to add an online survey with your redesign or revision. The survey should ask visitors what they like and dislike about the redesigned site. If the survey is long, consider including a special offer as a reward for submitting the completed survey. The information garnered from the survey can help you and the client gauge the overall success of the site. You can also require the visitor's e-mail address, which can then be added to the client's mailing list.

Book IX

Case Studies

The 5th Wave

By Rich Tennant



"Evidently he died of natural causes following a marathon session animating everything on his personal Web site. And no, Morganstern – the irony isn't lost on me."

Contents at a Glance

Chapter 1: Blog Site	693
Examining Blog Software and Services	693
Creating a Web Site from a Blog	707
Chapter 2: Podcasting Site	711
About Podcasts	711
Creating the Podcast Site	712
Recording Hardware	714
Sound-Editing Software	717
Chapter 3: Photographer/Portfolio Site	721
Defining Goals for the Site	721
Designing the Site	722
Chapter 4: Foundation Site	731
Meeting with the Client	731
Meeting the Client's Needs	732
Creating the Assets	733
Putting It All Together in Dreamweaver	735

Chapter 1: Blog Site

In This Chapter

- ✓ **Knowing how blogs work**
- ✓ **Adopting the b2evolution blog application**
- ✓ **Running multiple blogs with WordPress**
- ✓ **Using a blog hosting service**
- ✓ **Incorporating a blog with an existing site**

A *blog*, for those of you who have been couch potatoes instead of 'Net potatoes, is short for *Weblog*. A blog is like an online diary. You post your thoughts, photos, or information of interest to your viewing audience and then collect comments from people who read your blog. Kids use blogs, and so do politicians. Blogs come in many different varieties. There's the *vanity blog*, which is put up by someone who absolutely thinks it's the coolest thing since wireless pointing devices but only fills the blog with vacuous text. And there's the *serious blog*, put up by someone who has important information to convey to his Web site visitors. In this chapter, we explore two pieces of blog software and explore a blog used by two book authors to display tutorials and information about digital photography and image-editing applications. You can use the information in this chapter to create your own blog or to create blogs for your clients.

Examining Blog Software and Services

If you type **blog software** into your favorite search engine, you get pages of results to explore. Much of the most popular blog software and many blog templates are free.

Before actually showing you some blogs, we tell you a bit about how they work. When you add a blog to your Web site, you add PHP pages to your Web site. The pages enable you to display your blog online and accept comments from visitors to your blog. You post entries to your blog from the administration pages. From the administration pages, you can also delete pesky comments or comments you don't consider suitable for your blog. When you post an entry to your blog or a user posts a comment, the information is stored in a MySQL database on the server. When someone visits a blog site, the PHP page plucks the information from the database and displays it.

b2evolution

The b2evolution blog software, which is free, features an administration section that enables the user to create blog posts and also manage comments posted by users. The b2evolution blog is multilingual and supports multiple users and multiple blogs. You can download different blog *skins* (user interfaces) to change how the blog looks to visitors. If you're conversant with PHP code and CSS (Cascading Style Sheets), you can modify the blog to match the look and feel of other Web pages on the site. In the following sections, we show you some of the features of a b2evolution blog.

Introducing the b2evolution application

The b2evolution blog is powered by PHP and a MySQL database. As of this writing, the current version of b2evolution is b2evolution 2.4.5, which is also known as Monument Valley. You can download b2evolution at <http://b2evolution.net>. Follow the links to the Download section.

In order to run b2evolution, your Web server needs PHP 4.3 or above and MySQL 3.23 or above. The designers of the application claim that it works best with Linux with an Apache server. Everything on the blog is done online through a Web browser. b2evolution has been tested with the following browsers: Firefox 1.0+, Internet Explorer 6.0, and Konqueror 3.5+.

The b2evolution application has an impressive manual, which you can view at its Web site. You can see the full manual at http://manual.b2evolution.net/Main_Page. The manual covers every conceivable question about the application, including how to change the look of the blog, how to create blog entries, how to manage multiple blogs, and so on. The downloaded application comes with installation instructions that link to the b2evolution Web site and manual.

Installing b2evolution

If your Web server has the required features (listed in the preceding section) to run b2evolution, you're ready to install b2evolution on your server. You can find detailed instructions with the application. The following steps are condensed, but they give you an idea of how simple it is to install the application on your server.

1. Download and unzip the b2evolution application.

Detailed installation instructions can be found here:

`http://manual.b2evolution.net/Install_b2evolution`

All the necessary files you need to run the application are downloaded into a folder named Blogs.

2. Create a new MySQL database on your server.

You can easily create a MySQL database through your Web hosting service's control panel or a server application called phpMyAdmin. When you create the database, you're prompted for a name, user name, and password. You don't have to configure the database because the b2evolution application handles that task for you.

3. Upload the Blogs folder to your server via an FTP (File Transfer Protocol) application, such as CuteFTP or FileZilla.

When you unzip b2evolution, all the files required to run the application appear in the Blogs folder.

4. Change permissions on several pertinent files.

Using your FTP application, change the permission of files that configure the blog, enable uploading of image files when creating blog entries, and so on. Detailed instructions on which files need permission changes are in the complete instructions.

5. Install b2evolution on the server:*a. Access the Install folder.*

On a typical installation, you can find this at www.mywebsite.com/blogs/install/.

The installation dialog boxes appear.

*b. Follow the prompts to enter your MySQL database name, user name, and password.***6. After entering your database information, choose New Installation, click Go, and then follow the prompts.**

The installation software writes the config files and populates the database. During the installation, you receive a username and password, which you use to log in to the blog. If desired, after logging in to the blog for the first time, you can change the default password.

7. After you install b2evolution on your server, click the Log In link.

This takes you to the blog's log in page, where you're prompted for your user name and password. Before you can actually post to the blog, though, you have to change a few files.

8. Change a few permissions on the server and make some changes to the config.php file.

Most FTP applications give you the option to change permissions. The blog installation instructions tell you what changes to make.

After making these changes, you're ready to go blogging.



Posting a blog entry in b2evolution

After you install and set up a b2evolution blog, you (or your client) can begin posting entries to the blog. When you post entries to a b2evolution blog, you do so through the Back Office, which you access by logging in as administrator and then clicking the Log into Backoffice button. Several tabs in the Back Office enable you to perform myriad blogging tasks. The Back Office is well documented in the online manual. It's also fairly intuitive — almost as easy as writing an e-mail. If your client is familiar with working in a Web browser, you'll have him up and blogging in no time flat.

When you create a blog post, you enter the information in the Write section of the Back Office. You use the various buttons to add hyperlinks, images, and so on. (See Figure 1-1.)

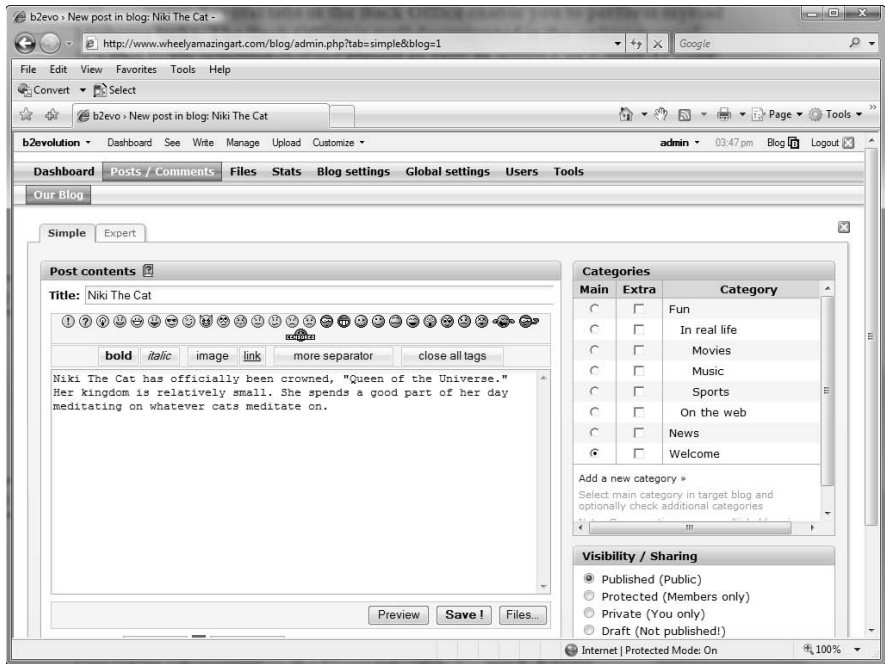


Figure 1-1:
Post an
entry to
the blog.

After writing a blog entry, you can preview it (see Figure 1-2) before clicking the Save button. After you upload the post, visitors to the blog can read it and add comments.

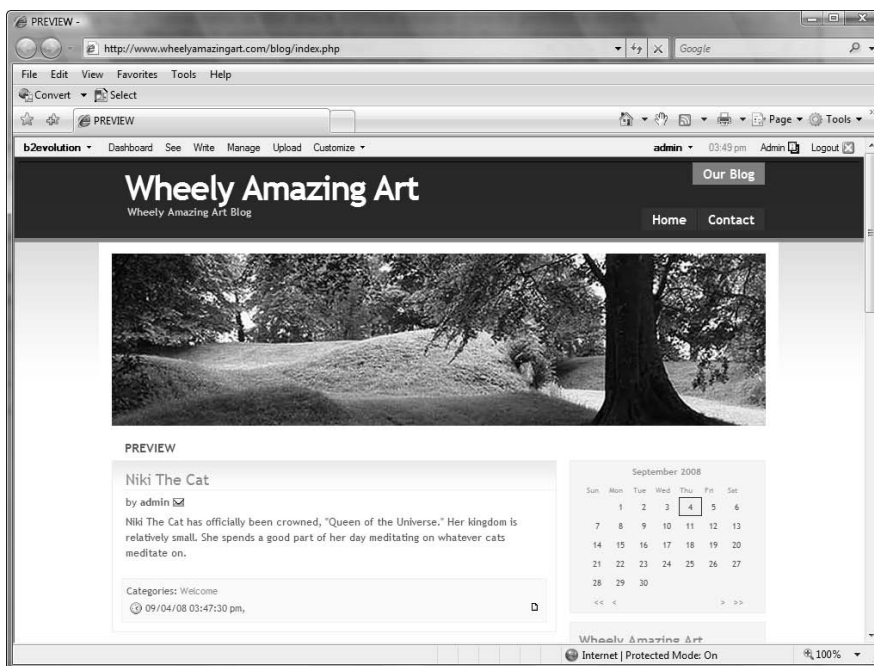


Figure 1-2:
Preview the
blog prior to
posting it.

Managing a b2evolution blog

You manage a b2evolution blog through the Back Office. In the Back Office, you configure your blog and add other users to the blog. If you're maintaining the blog for your client, you're the administrator, and you set up other users on the Users tab. (See Figure 1-3.)

After you have the blog up and running, you can manage posts as well as user comments. You can edit previous posts, delete posts, and delete comments that you deem inappropriate for the blog. You do this on the Posts/Comments tab of the Back Office. (See Figure 1-4.)

WordPress

Another very popular blog application is WordPress. Like b2evolution, WordPress is free. The application is feature rich. After you set up a blog on a server, you (or your client) can manage everything online through a Web browser. WordPress supports multiple blogs and multiple users. If desired, you can have users create profiles with WordPress. Depending on the template you use, visitors to the blog can leave comments about each post.

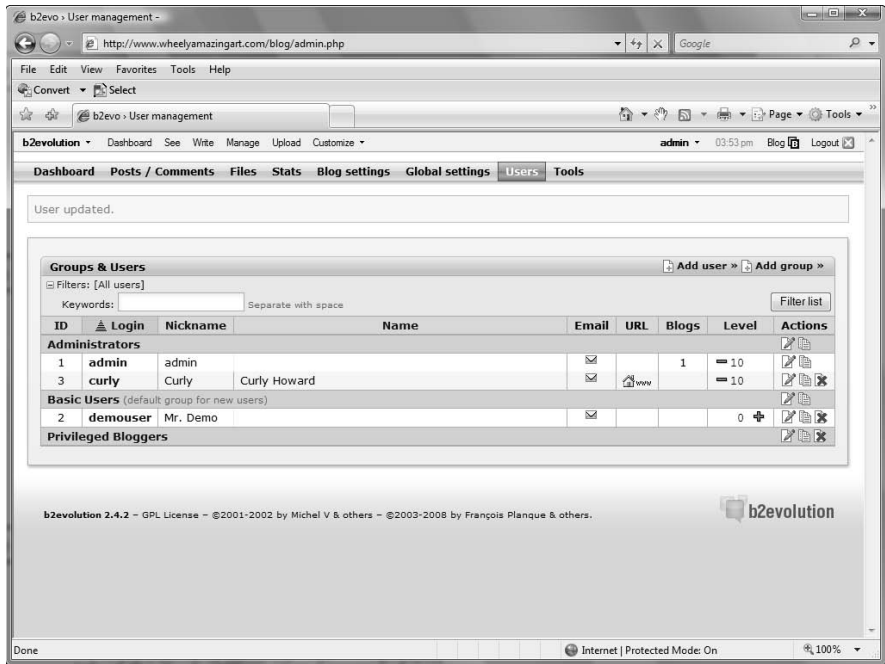


Figure 1-3:
Set up other
users in the
Back Office.

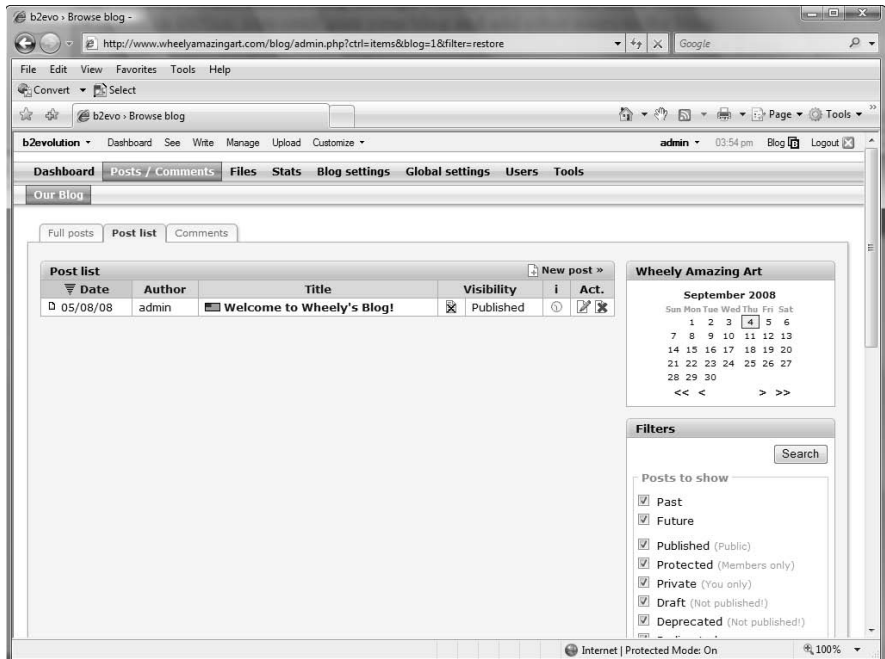


Figure 1-4:
Manage
blog posts
here.

About WordPress

WordPress also uses a combination of PHP code and a MySQL database to display and maintain the blog. As of this writing, the current version of WordPress is WordPress 2.6.1. You can download WordPress at <http://wordpress.org/download>.

To run WordPress, your Web server needs PHP 4.3 or above and MySQL 4.0 or above. The designers of the application claim it works best with an Apache or a LiteSpeed server.

WordPress comes with installation instructions. Although the application doesn't have a full-fledged manual, you can find impressive documentation on the application at http://codex.wordpress.org/Main_Page. The WordPress Web site also features an impressive support section (<http://wordpress.org/support>) that covers many installation issues as well as how to modify templates.

Installing WordPress

If your server has the necessary features (listed in the preceding section) to run WordPress, you're ready to install the application. Detailed installation instructions can be found here:

http://codex.wordpress.org/Installing_WordPress

To give you an idea of how easy it is to install the application, here's a condensed version of the installation process.

1. Download the application and unzip the files.

The files needed to run the application are downloaded into a folder called WordPress.

2. Create a new MySQL database on your server.

You can easily create a MySQL database through your Web hosting service's control panel or a server application called phpMyAdmin. When you create the database, you're prompted for a name, user name, and password.

3. Open `wp-config-sample.php` in a word-processing application.

We recommend using an application like Notepad to make your changes. More sophisticated word-processing applications use smart quotes (they look like this: “ ’ ”), which will cause errors when used to designate the database name, user name, and password.

4. Change the database information.

Modify the file by changing the default name of the database, user name, and password.

5. Save the file as `wp-config.php`.

6. Upload the files to a folder on your server.

Create a folder on your server. The logical name for that file is Blog. Upload the files to this folder.

7. Install the application on your server.

To install the application, navigate to the `install.php` file on your server. The default location is `www.mywebsite.com/blog/wp-admin/install.php`.

8. Follow the prompts to install the application.

The installation dialog box prompts you for a name for your blog and your e-mail address. After you supply that information, WordPress automatically fills in the table for your database. The application generates a user name and password.

9. Write down your user name and password.

You need these to log in to your blog, administer it, and post entries. If you're not crazy about the user name or password, you can change one or both of them when you administer the blog.

10. Log in to your blog.

You'll find a `wp-login.php` link at the bottom of the installation dialog box. Click the link to begin blogging.

Modifying a WordPress blog

After you install a WordPress blog, you can modify the blog to suit the Web site to which it linked. You can modify an existing theme (skin) or upload a new theme to your server. The downloaded version of WordPress has two themes. However, WordPress is a popular application, and many developers have created themes for the application. To modify the look and feel of your WordPress blog with a theme, follow these steps:

1. Open your Web browser and navigate to `http://wordpress.org/extend/themes`.

The WordPress Theme Directory Web page appears. (See Figure 1-5.)

2. Navigate to a theme you like.

As of this writing, there are more than 200 themes. You can speed up your search by displaying themes that have color schemes similar to the site to which they'll be linked. You can sort by color by enabling one or more check boxes.

3. Click the thumbnail to preview the theme in your browser.

The test site appears in a new browser window. (See Figure 1-6.)

Figure 1-5:
Select
a new
theme for a
WordPress
blog.

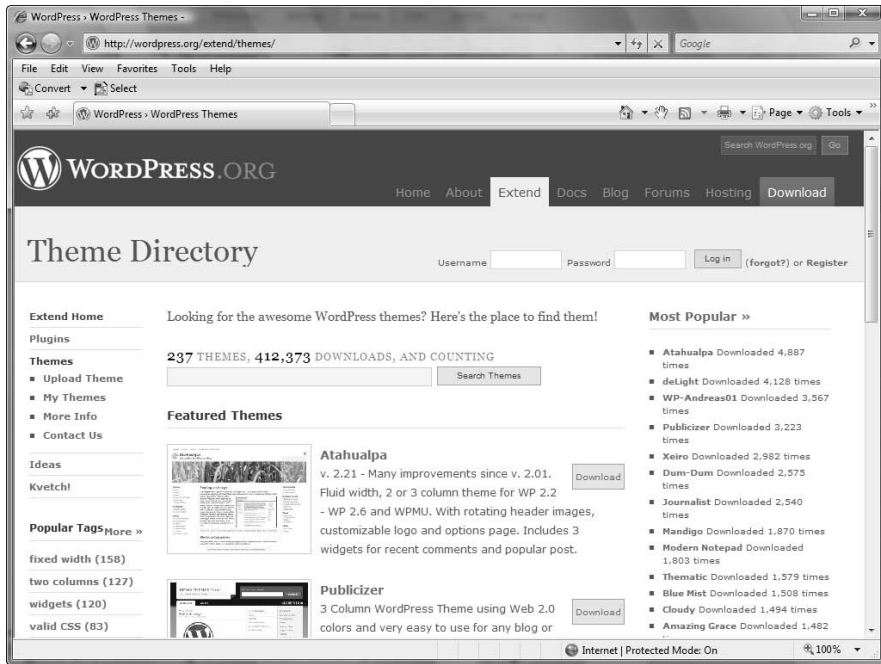
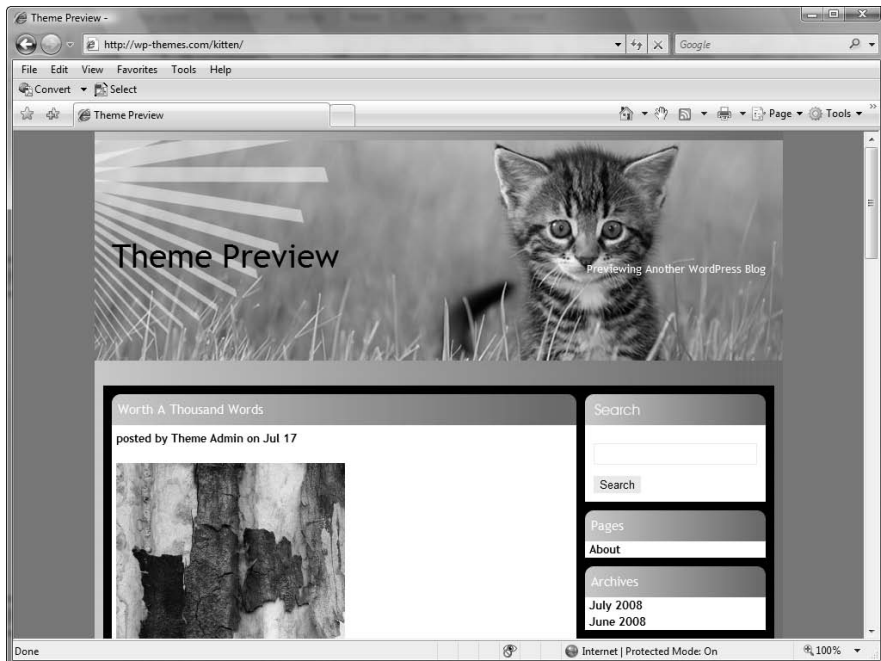


Figure 1-6:
Preview a
WordPress
theme.



4. When you find a theme you'd like to use, click the Download link and then follow the prompts to download the file to your desktop.

5. Unzip the file.

6. If desired, modify the theme in your HTML editing application.

You'll find tutorials on modifying themes at this URL:

```
http://codex.wordpress.org/
Blog_Design_and_Layout#Themes_and_Templates
```

7. Upload the Themes folder to the following directory on your server:

```
www.mywebsite.com/blog/wp-content/themes
```

After uploading a Themes folder to this directory, you can change the look of your blog.

8. Log in to your blog.

When you log in as administrator, the Dashboard section of your blog appears, which is where you administer the blog, write new posts, add new users, and so on. (See Figure 1-7.)

9. Click the Presentation tab.

Your Web browser refreshes and displays the themes you uploaded to your server. (See Figure 1-8.)

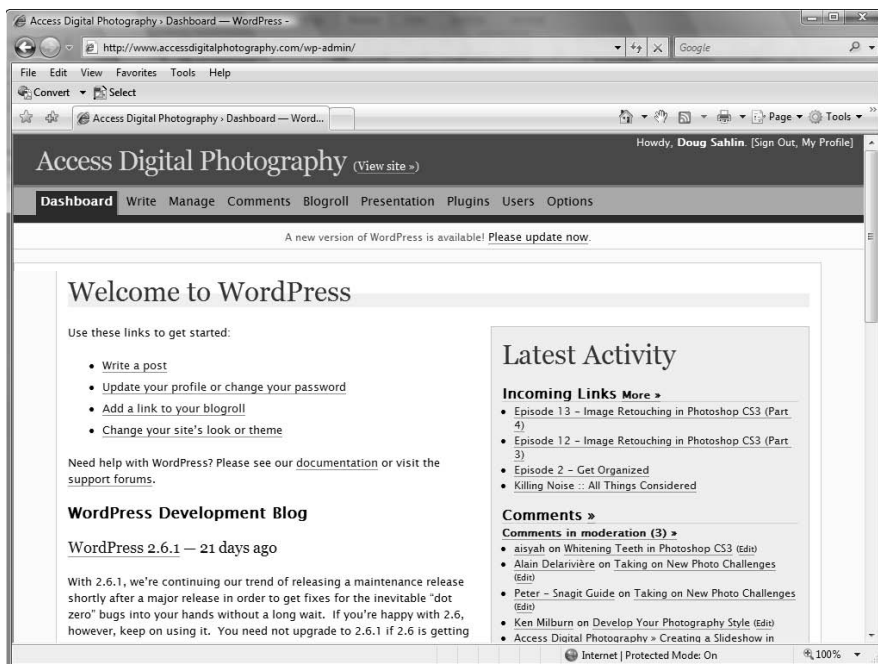


Figure 1-7:
The
WordPress
Dashboard.

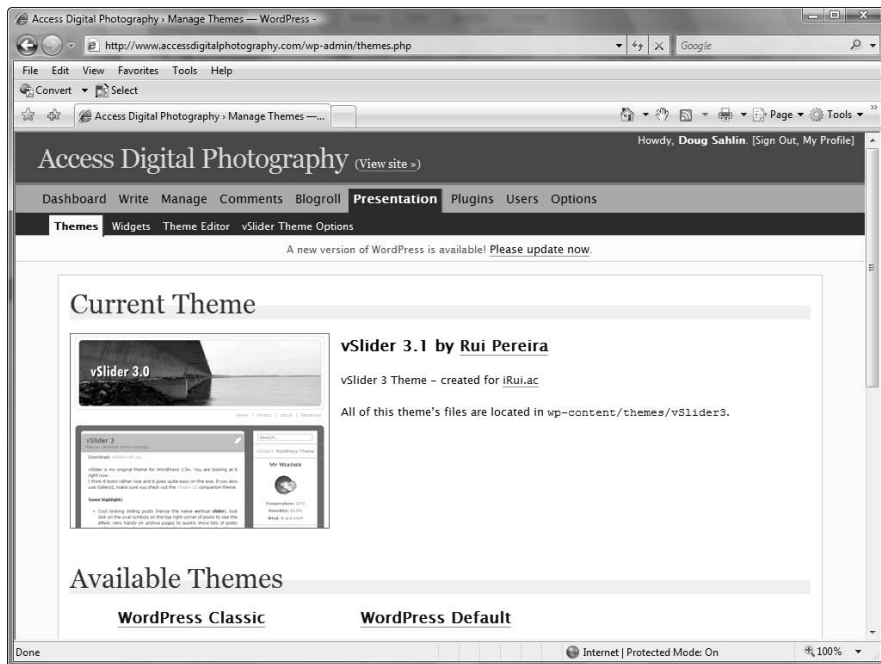


Figure 1-8:
Choose a
theme from
the uploads.

10. Select the desired theme.

The previously used theme is replaced.

Posting a WordPress blog entry

Blogs are all about getting the word out. WordPress makes it easy for you to post entries to your blog. You can add images and hyperlinks when posting a message to your blog. To add an entry to your blog, follow these steps:

1. Log in to your WordPress blog.

When you log in as administrator, WordPress displays the Dashboard, which enables you to administer your blog.

2. Click the Write tab.

Your Web browser refreshes to the Write page of your blog.

3. Enter the desired information.

When you post an entry to your blog, you can format the text and add images to the post. You can also add hyperlinks by clicking the Link icon. This opens a dialog box that enables you to enter the URL and determine whether the Web page opens in the same browser window or in a new one. Figure 1-9 shows a post being written. Figure 1-10 shows the post as it appears on the blog.

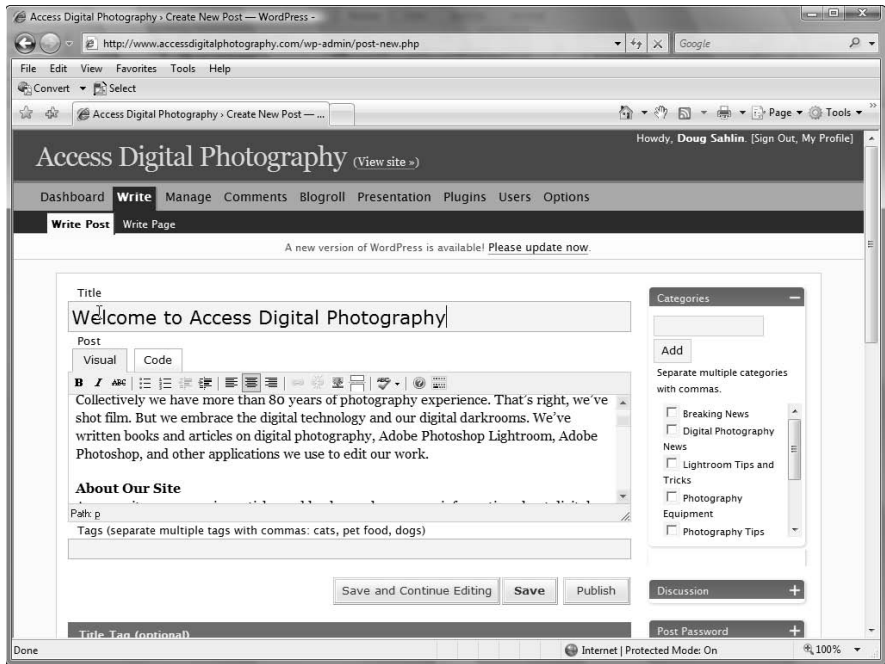


Figure 1-9:
Writing
a post.



Figure 1-10:
The entry,
as it
appears on
the blog.



If you or your client posts information on several different topics, you can create a category for each topic. This makes it easier to manage your blog.

Managing a WordPress blog

If you're setting up a blog with multiple authors and multiple categories, you can easily manage the blog through the WordPress Dashboard. You can also edit or delete posts. To edit posts in a WordPress blog, follow these steps:

1. Log in to your WordPress blog.

When you log in as administrator, WordPress displays the Dashboard, which enables you to administer your blog. If you have other people creating content for the blog, users with editor status can also edit blog entries.

2. Click the Manage tab.

Your browser refreshes to show all entries posted to the blog. (See Figure 1-11.)

3. Click the Edit link for the post you want to edit.

WordPress opens the entry on the Write tab.

4. Edit the message and then click the Save button.

WordPress publishes the edited post.

Access Digital Photography (View site »)

Dashboard Write **Manage** Comments Blogroll Presentation Plugins Users Options

Posts Pages Uploads Categories Files Import Export Title Tags Ads

A new version of WordPress is available! Please update now.

Posts

Search terms... Status... Author... Month... Category...

Any Any Any All

Filter »

ID	When	Title	Categories	Comments	Author			
194	2008-05-30 4:58:21 am	Welcome to Access Digital Photography	Breaking News, Digital Photography News	0	Doug Sahlin	View	Edit	Delete
256	2008-05-29 10:54:27 pm	How to Shop for a Tripod	Photography Equipment	0	Ken Milburn	View	Edit	Delete
253	2008-05-05 9:49:20 am	Taking on New Photo Challenges	Digital Photography News	2	Ken Milburn	View	Edit	Delete

Done Internet | Protected Mode: On 100%

Figure 1-11:
Manage posts here.

You can also add categories to the blog, modify comments, and do other tasks on the Manage tab. You can add links to the blog through the Blogroll tab (in this version of WordPress; the Links tab in other versions), plus add and manage users through the Users tab.

Exploring other options: Blogger.com

If your client doesn't have PHP capability on his Web server but still wants to go blogging, many online sources can host a blog. The online blog is linked to the blogger's site. Google acquired Blogger.com in 2003. Google has made improvements to the service, and it now features the capability to upload images with blog posts. You don't have to download any software; you do everything online. Blogger.com users can choose from a variety of templates.

Hosting a blog with Blogger.com is a three-step process. You create an account, name your blog, and then choose a template. After choosing a template, you or your client can begin posting words of wisdom — or a reasonable facsimile thereof. Figure 1-12 shows Doug's blog at blogspot.com.

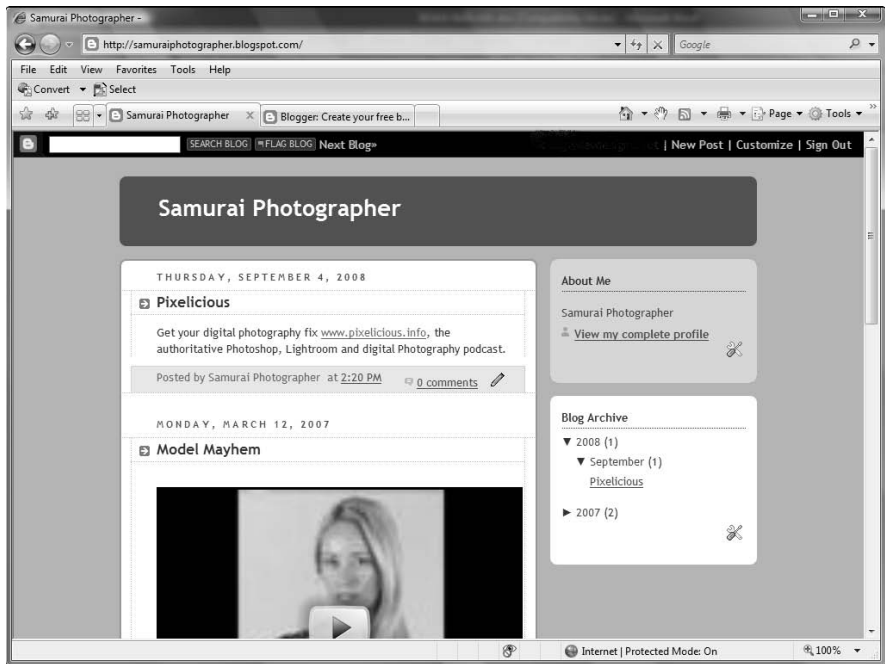


Figure 1-12:
A free blog
linked to
an author's
Web site.

Creating a Web Site from a Blog

When Doug and a fellow photographer decided to create a Web site to post digital photography, Photoshop, and Photoshop Lightroom tutorials, they decided to use a WordPress blog for the site. The blog makes it possible for site contributors who don't know HTML to add tutorials to the site. After setting up the blog, Doug added links to the blog that enabled viewers to visit different pages and other linked blogs. He also signed up for a Google AdWords account to monetize the site. Google AdWords pays the site owner when an ad is clicked through. The blog is shown in Figure 1-13.

Managing spam

The biggest problem with blogs is spam. When users find a blog, they think it's a place where they can post anything with links to everything from counterfeit Rolex watches to porn Web sites. Your first defense against spam in a WordPress blog is two-pronged:

- ◆ Require that an administrator approve a comment before it's posted.
- ◆ Require that the person posting the comment add her name and e-mail address.



Figure 1-13:
Create a full-fledged Web site from a blog.

Another option is for WordPress to notify an administrator by e-mail when a comment is posted. You choose these options in the Discussion section of the Options tab (see Figure 1-14).

Another powerful option for preventing spam is to use one of the built-in plug-ins: namely, the Akismet plug-in. This plug-in searches posted comments for any known spam words (think Viagra), and quarantines the posts for a site administrator to review. You need to set up an account at WordPress.org, which is free. When you enable the Akismet plug-in, you're prompted to enter your WordPress API (Application Programming Interface) key, which you can retrieve by logging into your account at WordPress.org. Trust us: As your blog becomes more popular, it will become a target for spammers. The Akismet plug-in does a remarkable job of thwarting spam, as shown in Figure 1-15.

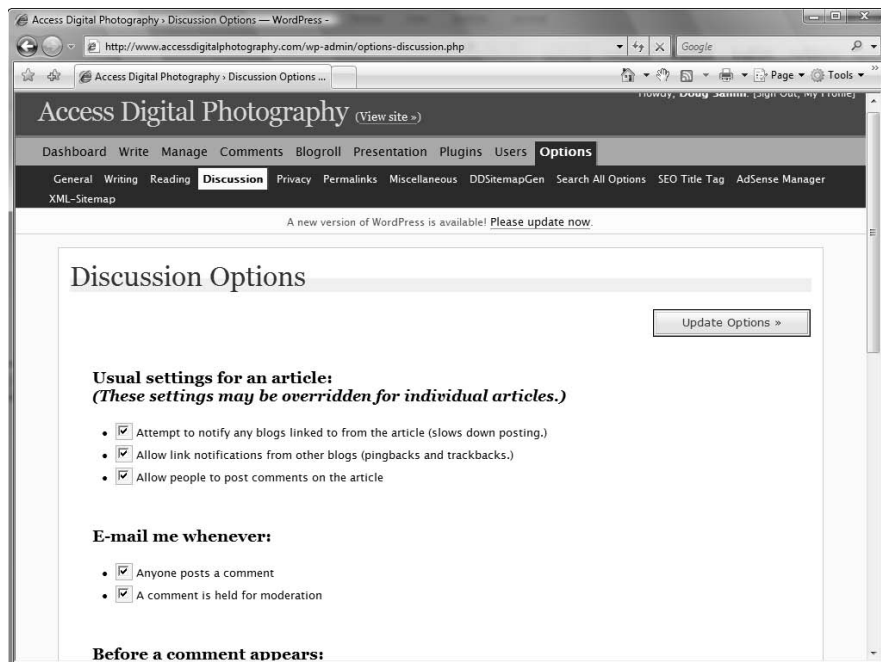
Extending WordPress

When Doug set up the site, he searched the 'Net for plug-ins to extend the blog. He wanted the blog to be recognized by search engines, easily administer Google AdWords, create a sitemap, and much more. He found the plug-ins here:

<http://wordpress.org/extend/plugins/>

Figure 1-16 shows the sitemap automatically generated and updated with a WordPress plug-in.

Figure 1-14:
Set user
commenting
options
in the
Discussion
section.



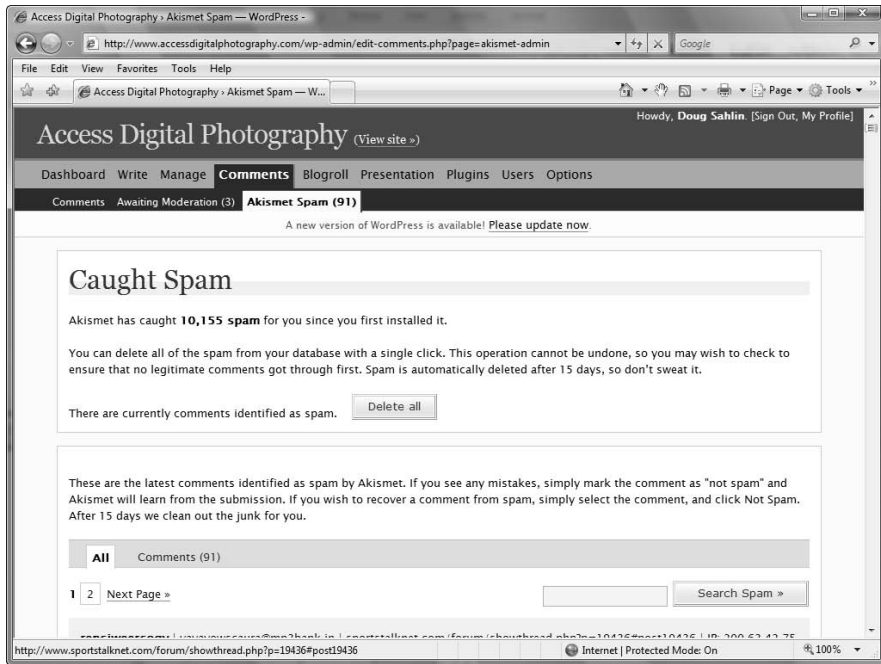


Figure 1-15:
Put a stop to spam on your blog.

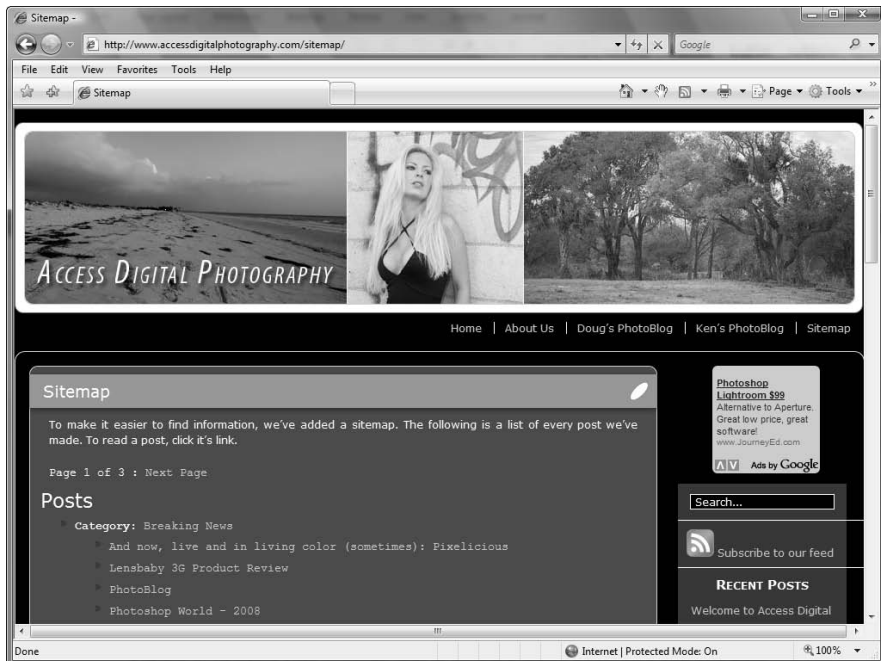


Figure 1-16:
Make life easier with WordPress plug-ins.

Photoblogs

WordPress is an extremely versatile application. With the right template, you can get the look you're after. Photographers like to post their work online, but many aren't conversant with HTML editing. If you're contacted by a photographer who wants to be seen online, consider the one-two punch of a regular blog that's linked to a *photoblog*. You can find photoblog templates for the WordPress application by typing **WordPress photoblog template** in your favorite search engine. Figure 1-17 shows Doug's photoblog, which is linked to www.accessdigitalphotography.com.

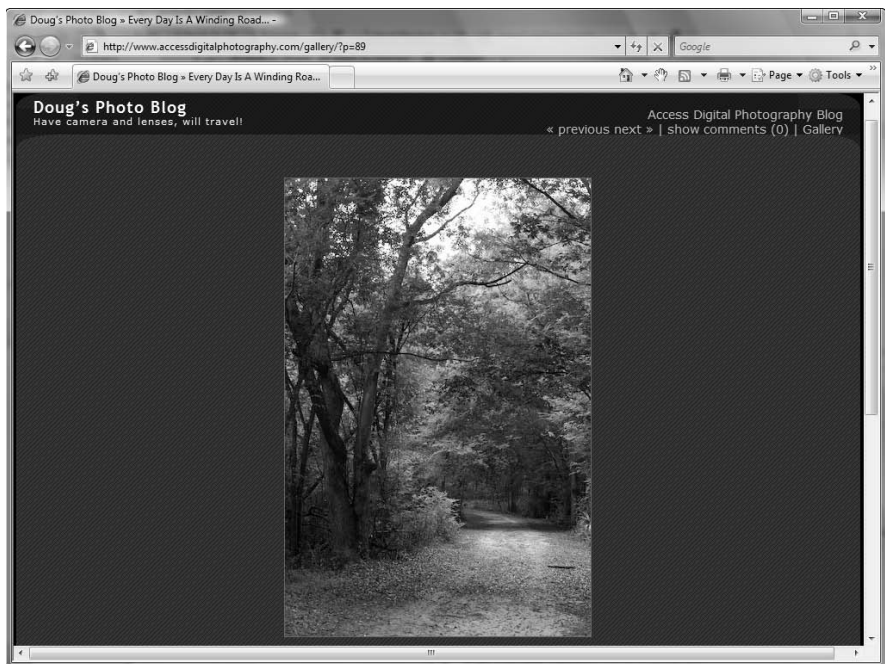


Figure 1-17:
A photoblog
that's easy
on the eyes.

Chapter 2: Podcasting Site

In This Chapter

- ✓ **Pondering the podcast**
- ✓ **Creating a user-friendly podcast site**
- ✓ **Checking out recording equipment**
- ✓ **Perusing recording software**

Podcasts are very popular. A *podcast* is an Internet broadcast that anyone with a computer can access. Podcasters run the gamut from people who have something that they want to get off their minds, to aspiring entertainers who want to be noticed, to corporations that want to educate their clients. If you have a client who wants to get the word out with a podcast, you need to create a site that's user friendly and that can be maintained by someone who may not have a lot of technical knowledge about creating and updating Web sites.

Doug loves photography and wanted to create a podcast to educate fellow photographers and share his enthusiasm about the subject and the applications he uses to edit his work. In this chapter, we dissect Doug's podcast site <http://www.pixelicious.info>.



This is a book about Web design, not about podcasting. However, many Web designers create podcasts to inform clients. If you're interested in creating your own podcast and want to read more, we can recommend two books: *Podcasting For Dummies*, 2nd Edition, by Tee Morris, Chuck Tomais, and Evo Terra, and *Expert Podcasting Practices For Dummies*, by Tee Morris, Evo Terra, and Ryan Williams; both published by Wiley.

About Podcasts

Think of a podcast as on-demand information or entertainment in an audio or a video format. Some people think that podcasts are associated with the iPod device, which can be used to view or listen to podcasts. However, podcast actually stands for Personal On-Demand (broad)cast. As the name implies, the medium can be listened to whenever the end user wants to. As this chapter is being written, the 2008 Democratic Convention and 2008 Republican Convention have concluded. People who want to hear what the

candidates have to say but didn't get to watch the broadcasts can download the speeches in the form of podcasts from iTunes, which is undoubtedly the most popular software for listening to digital media. The application is cross-platform, and can be downloaded free from the Apple Web site (www.apple.com/itunes/download). After installing the application on a computer, a user can download podcasts and other digital media directly from the iTunes store. Audio podcasts are saved as MP3 files, and video podcasts are saved as MP4 files.

Creating the Podcast Site

When Doug decided to do a podcast, he knew that at some point in time, he'd need other photographers to chip in and post podcasts. Therefore, the site had to be user friendly for contributors who didn't know how to post podcasts to the Web. The solution was a WordPress blog. Doug tweaked a template, and created a header and logo in Photoshop. He modified the template to include the header (see Figure 2-1).

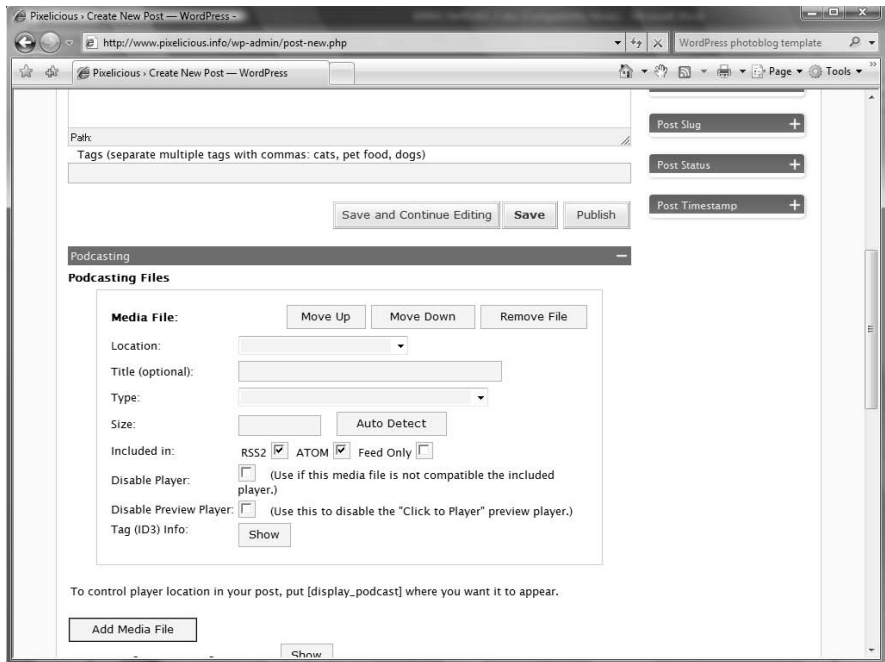


Figure 2-1:
Podcasting
for fun and
profit.

While still in the planning stages, Doug read many books about podcasting. These books showed him what he needed to play audio and video media and incorporate them in a post. The user-friendly application Doug decided on is PodPress (www.mightyseek.com/podpress). PodPress is a free plug-in that makes child's play of adding podcasts to a blog post. The application is added to the WordPress plug-ins folder at the blog Web site. Post authors can write the show notes for the blog, and then click the Add Media File button in the Podcasting section. (Refer to Figure 2-1.)

After selecting a media file that's been uploaded to the site, the plug-in can detect the file size and duration of the podcast. The user can then use the default image for the player or modify the URL to an image that's been uploaded to the site (see Figure 2-2).

PodPress also supplies a player for the podcast. This free plug-in saves hours of work. The plug-in also makes it possible to monitor the number of downloads for each podcast. Figure 2-3 shows a published post with an embedded podcast.



Figure 2-2: Add a player for the podcast and detect the file size and duration.



Figure 2-3:
Click to play
the podcast.

Recording Hardware

If you're going to record, get the best microphone you can afford. If you're recording voice, you don't need a mixer or preamp. Several microphones are available that can operate directly from one of your computer's USB ports. Here are a couple of solutions that will give you good sound quality without breaking the bank.

Zoom H2

The Zoom H2 microphone is a handheld recording powerhouse with four microphones in an incredibly small package. The device can record 16- or 24-bit sound files with a frequency of 96/48/44.1 kHz uncompressed in the WAV format or in the MP3 format with a sample rate of up to 320 Kbps.

The device enables you to change the sound format, frequency, and sample rate via menu commands. Buttons on the front of the recorder enable the user to switch microphone configurations. The unit can record a 90-degree pattern using two microphones from the front of the unit, which is ideal for recording your own voice; a 120-degree pattern using two microphones from the rear of the unit, which is ideal for recording a group of people; or four microphones for 360-degree surround sound.

The incredible sound quality is a great way to incorporate interviews in your podcast or to create a podcast from a remote location, such as a convention or show. The Zoom H2 can also double as a USB microphone. Simply plug in the supplied USB cable to the microphone, use a menu command to connect the H2 as an audio input device, and then use your computer recording software to record your podcast. In fact, Doug has used the Zoom H2 to record podcasts. For field recordings, the unit uses SD (SanDisk) memory cards.

The unit ships with a 512MB SD card, ear buds, a small tripod, windscreen, USB cable, power adapter, and a handle that can be used to connect the device to a microphone stand. As of this writing, the list price of the Zoom H2 (see Figure 2-4) is \$199.

Find online information for the Zoom H2 at

www.samsontech.com/products/productpage.cfm?prodID=1916

Blue Microphones Snowball

Professional podcasters and radio personalities use condenser microphones to get sweet rich vocals. However, most condenser microphones need phantom power to work. The Snowball, from Blue Microphone (see Figure 2-5), is white and round (just like a snowball), but it's powered by your computer's USB port.



Figure 2-4:
Use the
Zoom H2
for field and
computer
recordings.



Figure 2-5:
The Snowball is a condenser microphone ideally suited for computer recording.

The Snowball is a professional-quality, condenser microphone with two polar patterns: *omnidirectional* (360 degree) and *cardioid*, which records sound from the front of the microphone. The cardioid pattern is ideal for podcasting when you have to deal with background noise such as computer fans. There is also a cardioid pattern with a -10 decibel (dB) attenuator if you're recording in a noisy environment. The microphone records 16-bit sound with a sample rate of 44.1 kHz.

The microphone is plug and play. Connect the Snowball to a microphone stand using the thread mount at the bottom of the unit, connect the USB cord to the microphone, and then to your computer. It's that simple. The Snowball works with either Macintosh (OS X; USB 1.0 or 2.0; with a minimum of 64MB RAM) or Windows (XP Home Edition or XP Professional; USB 1.0 or 2.0; with a minimum of 64MB RAM) OS. As of this writing, the Snowball has a list price of \$159.

Find more online about the Snowball at

www.bluemic.com/modules.php?op=modload&name=Products&file=index&prod_id=18

Blue Microphones Snowflake

The Snowflake from Blue Microphone Snowflake is an ideal recording solution for road warriors. It's small. Just toss it in your laptop case, and it's readily available when you need to record on the road. You can place the microphone on a flat surface or clip it to your laptop. The Snowflake is a professional-quality, condenser microphone in an incredibly small package. The

microphone has a cardioid pattern, which means the device picks up sound in front of the microphone. With this microphone, you record 16-bit sound with a sample rate of 44.1 kHz.

The microphone is plug and play with most modern operating systems. Connect a USB cable from your computer to the microphone, and you're ready to record. When you need to pack the microphone away, coil the USB cord, place it in the base, and close it.

You can also separate the base and attach the microphone to a laptop computer (see Figure 2-6). This handy microphone is about the same size as a classic iPod and works with either Macintosh (OS X; USB 1.0 or 2.0; with a minimum of 64MB RAM) or Windows (XP Home Edition or XP Professional; USB 1.0 or 2.0; with a minimum of 64MB RAM) OS. As of this writing, the Snowflake has a list price of \$79.

For more information on the Snowflake, visit

http://www.bluemic.com/modules.php?op=modload&name=Products&file=index&prod_id=21



Figure 2-6:
The
Snowflake,
a road
warrior's
best friend.

Sound-Editing Software

After picking up a good microphone, you need software to capture your podcast. Sound-editing software is also useful when you need to edit files supplied by other contributors to your podcast. In this section, we show you three applications Doug uses: Adobe Audition, Sony Sound Forge, and Sony Acid Music Studio.



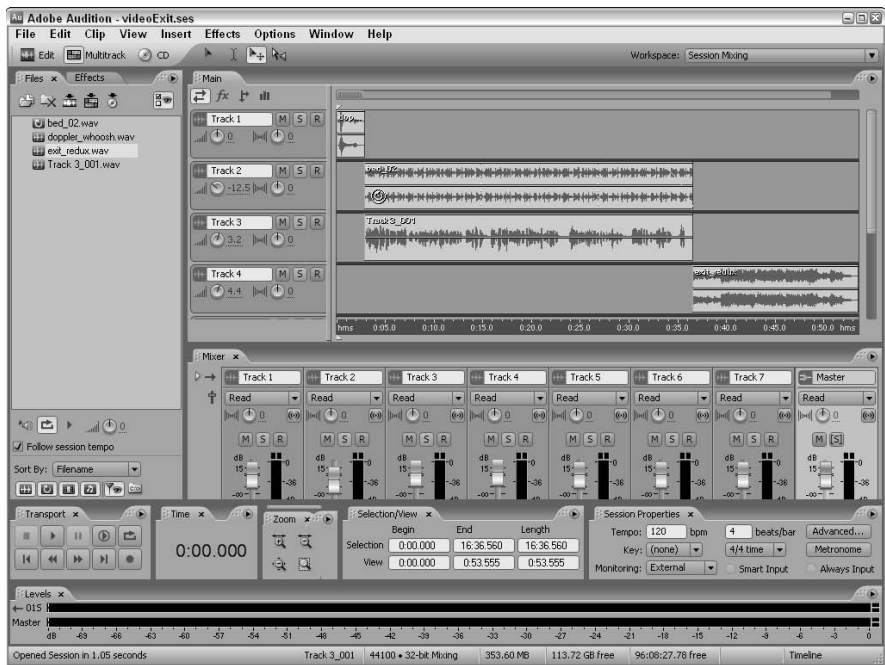
When you record, make sure you use the VU meters that are available in most applications, and make sure the VU meters don't go into the red zone. When the meters go into the red, the peak of the waveform is flat, which means the sound is clipped and distorted.

Adobe Audition and Sound Booth

Adobe Audition (www.adobe.com/products/audition) is a full-featured, professional recording and sound-editing application. You can record multiple tracks with the application, clean up existing audio, and edit your podcasts. The software features several options. You can work in multitrack mode or edit an individual track. The software has tools you use to analyze a sound and much more. You can also restore noisy and clipped files. Doug uses Audition to record and edit Pixelicious (www.pixelicious.info). The workspace (see Figure 2-7) is intuitive if you've edited sound before. As of this writing, Adobe Audition lists for \$349.

Adobe Audition may be more than you need. In that case, you might want to consider another Adobe application called Sound Booth (www.adobe.com/products/soundbooth), which is targeted toward people with little or no experience in sound editing. As of this writing, the application lists for \$199.

Figure 2-7:
Edit and
record
sound with
Adobe
Audition.



Sony Sound Forge

Sony Sound Forge is another powerful sound-editing application. Sony Sound Forge 9 features multitrack editing. The application also makes it possible for you to record and edit a podcast. As of this writing, Sony Sound Forge 9 retails for \$299. Figure 2-8 shows the workspace for Sony Sound Forge 7.

Find more online at

www.sonycreativesoftware.com/soundforge

Sony Acid Music Studio (Windows only)

Sony Acid Music Studio is an application that enables you to create soundtracks by using music sample loops. The application supports multiple tracks and recording. You can find it at

www.sonycreativesoftware.com/musicstudio

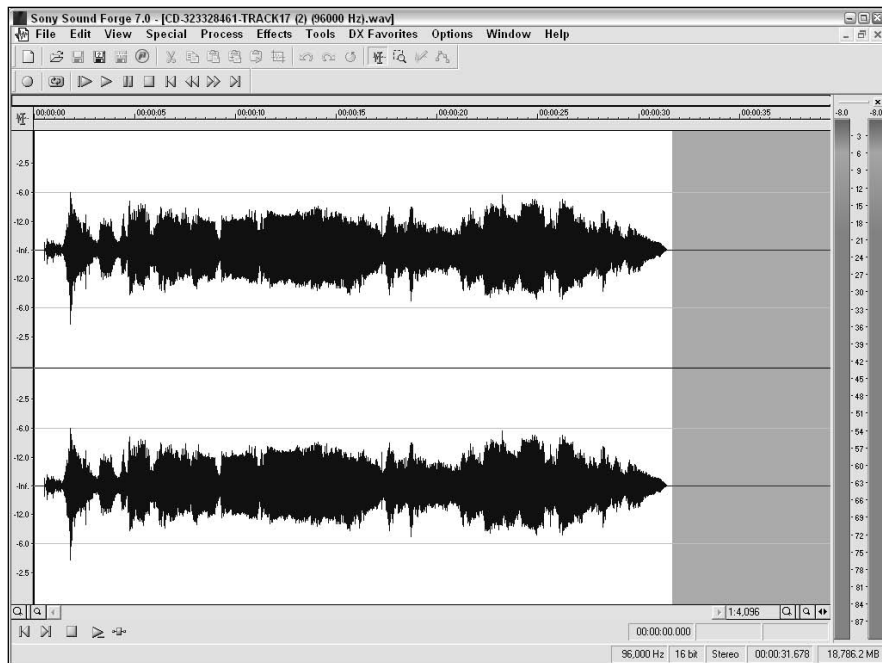


Figure 2-8: Record, edit, and repair sound files with Sony SoundForge.

When you assemble a song in Acid, the loops are seamlessly mixed to create your soundtrack. Doug used Sony Acid to create the background music for the intro and exit of his podcast. As of this writing, Sony Acid Music Studio, which ships with over 3,000 loops, sells for \$54.95. You can purchase additional loops for most musical instruments in a wide variety of genres from Sony online at

www.sonycreativesoftware.com/loops

Chapter 3: Photographer/ Portfolio Site

In This Chapter

- ✓ Knowing what you want
- ✓ Creating a navigation menu and thumbnail page
- ✓ Doing preliminary work in Fireworks
- ✓ Creating the portfolio

Every client has special needs, and photographers are no exception. The old saying that a picture is worth a thousand words has never been truer than when a photographer chooses to use a Web site for promotion. Photographers need to include information about themselves and their services. By far, though, the most important part of a photographer's Web site is the portfolio section. We can share some first-hand insight into this topic because when Doug isn't occupied building Web sites and writing books, he's also a professional photographer. In this chapter, Doug takes the lead and dissects a photographer's Web site — in fact, he dissects the Web site for his own wedding and event photography company, Superb Images, Inc.

Defining Goals for the Site

If a Web site doesn't have a reason for existence, it's just another domain name taking up space on a server. When I decided to turn the photography skills that I'd spent most of my life learning into a source of income, I realized I needed to have a Web presence. Superb Images is comprised of two talented photographers — including the author of this chapter. Before incorporating Superb Images, we researched the domain name and found that superbimages.net was available. We secured Web site hosting, reserved the domain name, and incorporated the company. This chapter gives you an idea of the thought process that went into creating the site, plus the actual construction of the site. The following is a list of goals I wanted to accomplish with the Web site, including creating

- ◆ A clean, uncluttered design to showcase galleries of photography that highlight the strengths of the company
- ◆ A simple, easy-to-understand navigation system
- ◆ Fast-loading pages
- ◆ Portfolio pages of thumbnail images that the viewer can click to reveal full-size images
- ◆ Multiple portfolio pages of similar images
- ◆ An About page to give viewers information about the company and its photographers
- ◆ A page that details the service the company offers, including wedding photography
- ◆ A Contact page
- ◆ A Links page

Designing the Site

After I defined the goals for the site, I was ready to design the site. I reviewed many sites of local photography companies and bookmarked those that matched my design goals. Then, it was just a matter of relying on my creativity to come up with a Web site that's immediately identifiable as a photographer's Web site — but with my own spin.

Creating the navigation menu in Fireworks

I knew how many buttons I would need for the navigation menu. I wanted a horizontal navigation menu, but the number of buttons exceeded the available space for a clean, horizontal layout. Therefore, I decided to use the Fireworks pop-up menu feature for the subpages in the portfolio section of the site. When I designed the menu in Fireworks, I added the URL in the Link text field in the Property inspector. The basic menu is shown in Figure 3-1.

After laying out the main menu, it was time to add the pop-up menu. My first step was to define the links in the Content tab of the Pop-up Menu Editor. (See Figure 3-2.) Thanks to the flexibility of Fireworks, I was also able to define the URL for each link.

After defining the content of the pop-up menu, I specified the parameters in the Appearance, Advanced, and Position tabs of the Pop-up Menu Editor. I decided to use HTML to generate the pop-up links.

Figure 3-1:
Roll up for
the magical
mystery
menu.

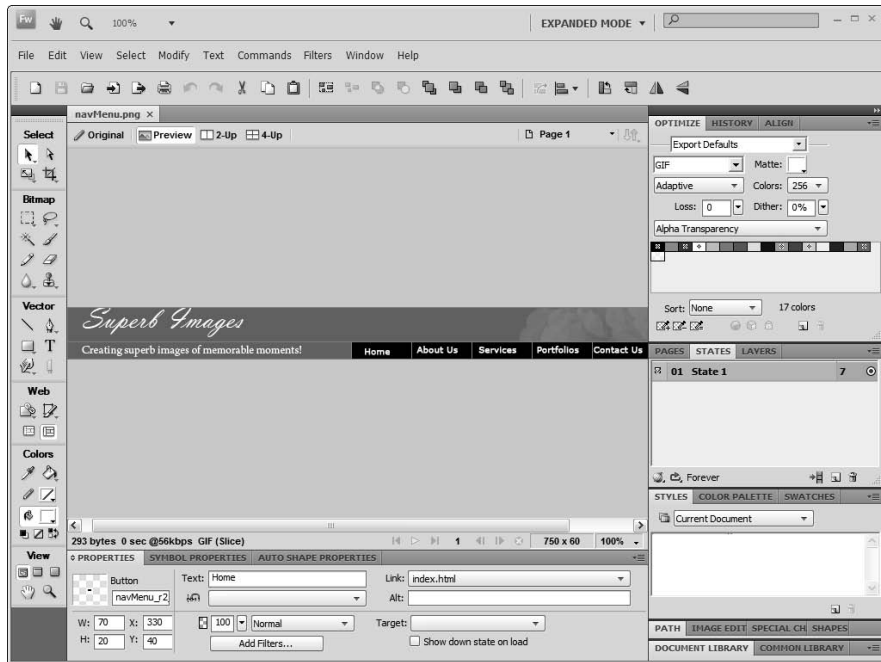
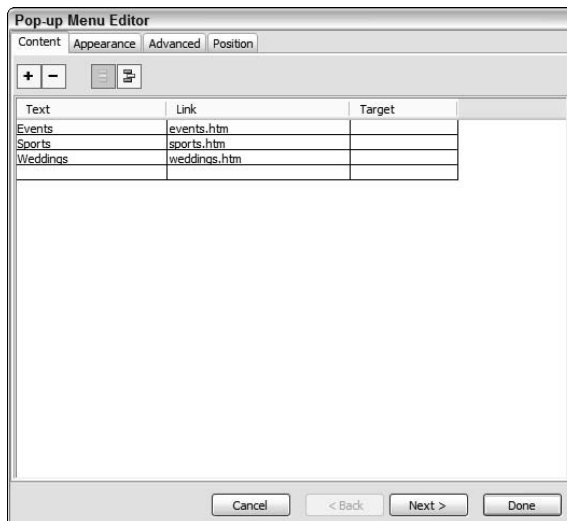


Figure 3-2:
Defining
the pop-up
menu.



I exported the finished menu from Fireworks as HTML and images. This option creates HTML code for the menu, which neatly houses the buttons in a table. The menu, complete with HTML, can be inserted into another HTML document.

Optimizing images in Fireworks

I'm a photographer who uses Photoshop extensively for image editing, touchup, and so on. Fireworks CS4 is a remarkably versatile application in the hands of an adept designer. Although it doesn't have the image-editing features of Photoshop, Fireworks makes it easy for a Web designer to optimize a folder of images for display in a photographer's portfolio. Prior to optimizing images for a Web portfolio, I use Adobe Bridge to rename images and add other metadata, such as copyright information and keywords. If you're a photographer and want to know more about workflow in Adobe Photoshop CS2, including Adobe Bridge, pick up a copy of *Photoshop CS2 for Digital Photographers Only*, by Ken Milburn and Doug Sahlin (Wiley).

I create uncompressed copies of the original images in the Web site Images folder and rename them to reflect the portfolio in which they'll appear. For example, if I'm renaming images for a portfolio called Events, I end up with a folder of images named Events_1, Events_2, and so on. Or if the images are for a specific event, I'll create a four-character abbreviation of the event. For example, images for a classic car show at Lake Mirror would be renamed LKMR_1, LKMR_2, and so on. I then use the Fireworks Batch Process command to optimize the images in JPEG format and resize them for the Web. (See Figure 3-3.)

When you resize images, make sure you check the Only Scale Documents Currently Larger Than Target Size; otherwise, you're asking Fireworks to increase the size of pixels, which leads to image degradation. For a photographer's portfolio, image quality is an important factor. As a rule, I use a quality value of 60 as the minimum. The size to which you optimize the image depends on the desktop size used by your intended audience. After optimizing the images, you can delete the copies of the originals.

Creating thumbnails in Fireworks

One of the goals of my site was to have fast-loading pages. Therefore, I decided to display thumbnails of the images that a viewer can click to reveal full-size images in another browser window. You can easily resize a folder of images by using the Fireworks Batch Process command. When I process thumbnails in Fireworks, I use the original images and batch process using the Export, Scale and Rename options. I use the same image quality (60) as I do for the full size images. I add a suffix `_th` to the filename so as not to overwrite the original image. (See Figure 3-4.)

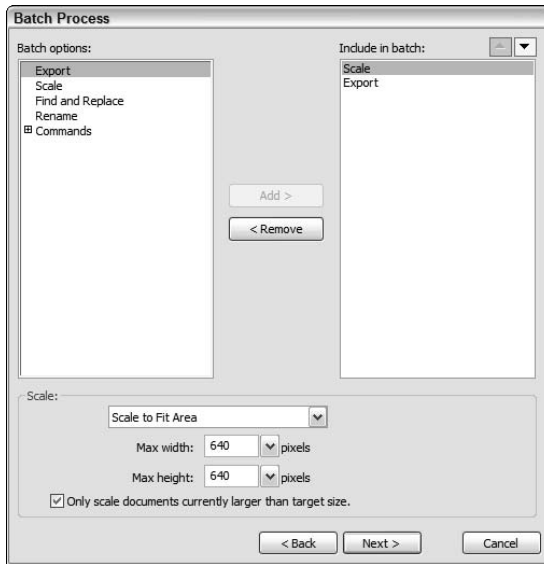


Figure 3-3:
Optimize me,
Fireworks.

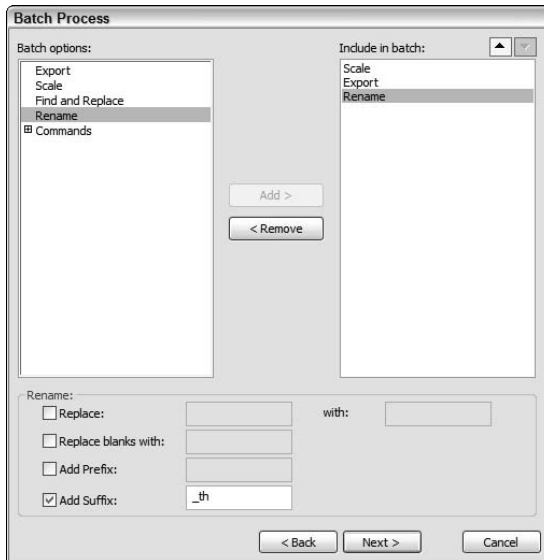


Figure 3-4:
It's so easy
when you
use batch
processing.

Adding `_th` to the filename tells me the images are thumbnails, which makes it a breeze to insert them (rather than the full-size images) into the gallery page. For more information on using Fireworks to optimize images, refer to *Adobe Creative Suite 3 Web Premium All-in-One For Dummies*, by Damon Dean, Andy Cowitt, Jennifer Smith, and Christopher Smith (Wiley).

Creating the other assets

For this Web page, I incorporate the banner as part of the navigation menu. But there are other times when designing a site that it makes more sense to separate the banner from the menu. In these cases, I use Photoshop to create the site banner and the original image on the home page. I use the Text tool to create the banner text, using a legible font like Adobe Garamond. I often use the Character panel to format the text with the small caps option.

I also created a slideshow in Flash CS4 for the home page. I debated long and hard whether to include background music. I did decide against including background music for the main photography site, but I might include it for a separate wedding site, which is still under construction while this chapter is being written. The slideshow was published as a Flash SWF file without HTML. I used Dreamweaver to insert the file into the home page.



If you're not proficient in Flash, there are lots of Dreamweaver plug-ins you can use to create compelling slide shows. DMX Zone makes a plug-in called 3D Image Flow Gallery, that creates a slideshow similar to an iPod cover flow.

Putting it together in Dreamweaver

After creating the assets in Fireworks, Flash, and Photoshop, it was time to put it all together in Dreamweaver. I began by defining the site. With that task out of the way, I was ready to create a template that I could use on all pages.

I used the Insert⇨Image Objects⇨Fireworks HTML command to add the pop-up menu at the top of the page. I followed up by creating an editable region in which I could add text. To finish the template, I added a text menu at the bottom of the template because search engines can't read the links in the Fireworks pop-up menu, which is generated by JavaScript. When finished, I saved the document as a template. For more information on using Dreamweaver CS4 to create a site, refer to Chapters 5 and 6 of Book III.

With template in hand, I could create most of the content for the site. Because I used a template, I didn't have to re-create the navigation menu and other content common to the site. Sweet. I also created a Cascading

Style Sheet (CSS) to modify the default styles to suit the site and create new styles. I added the Flash slideshow was added to the HTML document by using Dreamweaver Insert→Media→SWF command. Because I have a pop-up menu on the home page — which when popped up, displays directly over the Flash slideshow — I added the `wmode` parameter in the Property inspector and set its value to `transparent`. The finished home page as displayed in Firefox is shown in Figure 3-5.



Creating the portfolio pages

To create the portfolio pages, I created a new document using the site template. I modified the template by deleting the image that was used on all other pages (an editable region), replacing it with a title — in this case, Wedding Photography. I then created a table to house the thumbnails. Each cell in the table was created as an editable region. The modified template was saved with a different name. I used the template to create the portfolio pages.

Each thumbnail is a link to a page that contains a larger version of the image. The Open Browser behavior (Window→Behaviors) is used to open the link in another browser window. The behavior also enables you to specify the size of the new browser window, which (in this case) is just slightly larger than the image. Figure 3-6 shows the portfolio page as it appears in the Firefox Web browser. Note the larger version of the image that appears when the visitor clicks the applicable button.

Showing off without snootiness

I wanted to do something special for the portfolio pages. While surfing for inspiration, I saw an image gallery plug-in — Image Gallery Magic — that just blew my socks off. As soon as I saw a gallery created with this plug-in, I just knew I had to use it to create the portfolio pages for my Web site. You can find it at the Project VII Web site:

www.projectseven.com/products/galleries/lightshow/index.htm



As of this writing, the plug-in retails for \$95 and is compatible with Windows and Macintosh versions of Dreamweaver CS4. If you're doing a Web site for a photographer — or, for that matter, any client who wants to display a large number of images with a touch of class — add the cost of this plug-in to your proposal. Your client will be pleased with the end result, and you can use the plug-in for other clients.



Figure 3-6:
The
completed
portfolio
page.

Image Gallery Magic enables you to create new pages using one of six gallery presets. Alternatively, you can add a Image Gallery Magic gallery to an existing page, which is what I did on my site. (See Figure 3-7.) The application dovetails nicely with Fireworks, which handles the creation of the thumbnails and full-size images. The application enables you to scale the images to fit within a desired size, specify the caption under each photo, and so on.



Figure 3-7:
An Image
Gallery
Magic photo
portfolio.

Chapter 4: Foundation Site

In This Chapter

- ✔ Meeting with the client
- ✔ Defining site goals
- ✔ Managing the assets
- ✔ Creating the site

Doug receives many referrals by word of mouth. Many referrals are nothing more than somebody trying to pick a designer's brain, after which the designer then attempts to design the site. We've dealt with many of those, and have a sixth sense that enables us to figure out when a referral is the real deal or a waste of time. Web designers are too busy to deal with the latter. Remember that the quality of a referral is almost always equal to the quality of the referrer. A valued colleague of Doug's referred a woman who was setting up a foundation to help the families of firefighters lost in the line of duty. Her son lost his life in a tragic line of duty accident. This chapter discusses some of the special needs for that client and how they were met.

Meeting with the Client

Doug asked the client to meet him at a local restaurant with Wi-Fi capabilities. That way, he could bring his laptop computer so they could look at some of his previous work as well as some sites that the client liked. She told Doug more about the foundation she was creating, and they explored some of the sites she liked. After discussing a color scheme for the site, Doug made a few sketches in his notebook and jotted down the client's needs, which were as follows:

- ◆ An easy to navigate site with a green-and-yellow color scheme
- ◆ A way to communicate with families who had lost loved ones in the line of duty
- ◆ An events page
- ◆ A site that could be maintained by the client
- ◆ A page from which people can contact the foundation
- ◆ A page with images and biographies of the foundation's board of directors



At this stage, Doug made a couple of suggestions, including a links page, where the site could display the logos of sponsors with links to their sites.

Using a combination of incoming and outgoing links also helps boost the site's visibility with search engines.

After Doug had a firm grip on the client's needs, the conversation turned to Web hosting and a domain name. The client didn't have a domain name, but she knew that she wanted a .org domain, using the initials of her foundation — FFSF — for the name of her foundation, the Fallen Firefighter Survivors Foundation. Doug immediately checked online, but unfortunately, the domain name was not available. When the client told Doug the foundation would be nationwide, he suggested using FFSFUSA.org instead, which was available. After talking with the client, Doug knew the server side requirements, and he suggested a couple of hosting services.



You can use a domain registrar like GoDaddy (www.godaddy.com) to check availability of a domain name. An alternative is www.betterwhois.com.



When you register a domain, anybody can research the domain name, and find out quite a bit of information. If you want to remain anonymous, don't give out any personal information (such as your name and address) when registering the domain. When at all possible, just use the company name and a P.O. box.

Meeting the Client's Needs

After the initial consultation with the client, Doug drafted a detailed proposal, including his fee. While reviewing the client's requirements and the sites of other nonprofit foundations, he remembered the selfless acts of heroism that the NYFD demonstrated during and after the terrorist attacks on the World Trade Center on September 11, 2001. After recalling these memories, he decided this would be a cause to which he could donate his time.

There were two options for communicating with site visitors: a forum or a blog. A forum can get huge and be difficult to manage. Doug suggested using a blog, which the client accepted when she saw how easy it was to maintain. Using a blog makes it possible to frequently update content, which makes a site rank higher with many search engines. Site visitors can comment on blog posts, and the site owner or designer can set up the blog so that all comments must first be approved by a blog administrator before they're made public. This prevents posting derogatory comments, spam, or links to porn sites. Using a WordPress blog also features a powerful anti-spam plug-in — Akismet — which quarantines comments with copious links as well as dubious content (such as words related to porn sites).



When you decide to add a blog to a Web site, make sure that your Web server's operating system supports the blog type you choose. As of this writing, Windows servers and WordPress blogs don't work and play well together. If you want to add a blog to a Windows server, b2evolution will get the job done.

While Doug was gathering assets for the site, the foundation owner registered the domain and secured a host for the site. The site owner did not have HTML-editing software, and one of her requirements was the ability to easily update the site. Doug suggested that she download a trial version of Contribute, and he sent her a connection key to the site, after uploading some test pages to the site. After a few minutes of instruction, the client got the knack of working with Contribute, and Doug had met another major requirement. As of this writing, Contribute CS4 retails for \$199.

After doing preliminary research and conferring with the client, Doug decided that the initial version of the site menu required using seven buttons: Home, About Us, Director Bios, Event Calendar, Contact Us, Blog, and Links.

Creating the Assets

The client sent the content to Doug as it became available. A local graphic artist was commissioned to create the logo for the foundation. Doug stored the client-supplied assets (text documents and images) in the site's folder on his computer.

When he had enough information to begin creating assets, Doug put some thought into the colors for the site. The site owner initially wanted green, yellow, and black, but decided black was a tad somber. She asked Doug for a solution. He ended up using a wonderful tool — Kuler (<http://kuler.adobe.com>) — from the Adobe Web site. The Kuler home page shows some of the most recent color schemes created by Kuler users. You can also use the site to create your own color scheme by logging in with your Adobe ID and then using the site to create your own color scheme (see Figure 4-1). As of this writing, you can set up an Adobe ID and use Kuler at no charge. Kuler is also part of Adobe CS4 and Flash CS4.

The client wanted to use a photo of a serene forest as part of the banner. She chose one of Doug's landscape photos. The image was incorporated with the site name and a placeholder for the logo in Photoshop CS4 (see Figure 4-2). The banner was exported as a JPEG file, using the Save for Web & Devices command.

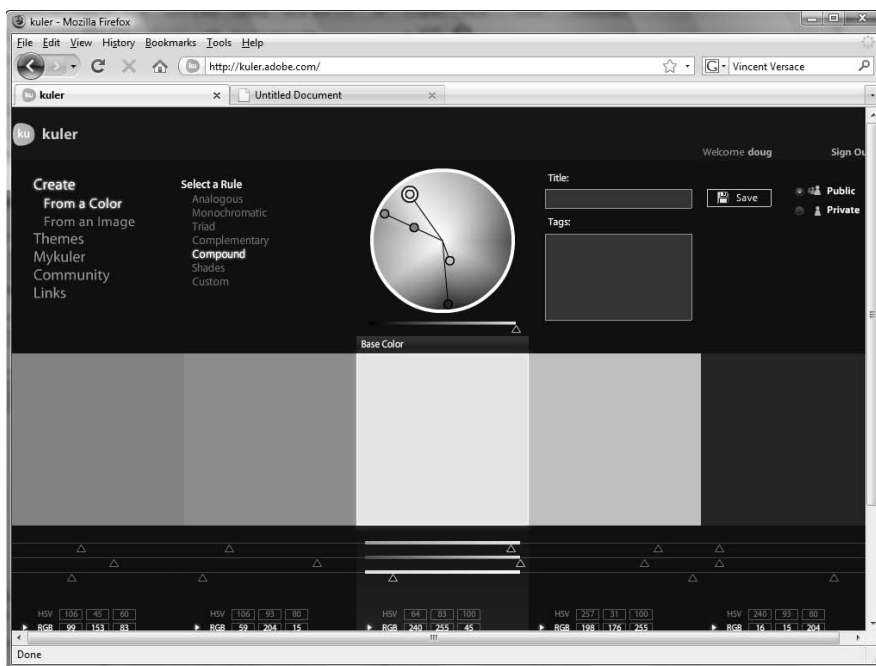


Figure 4-1: Creating a cool color scheme in Kuler.

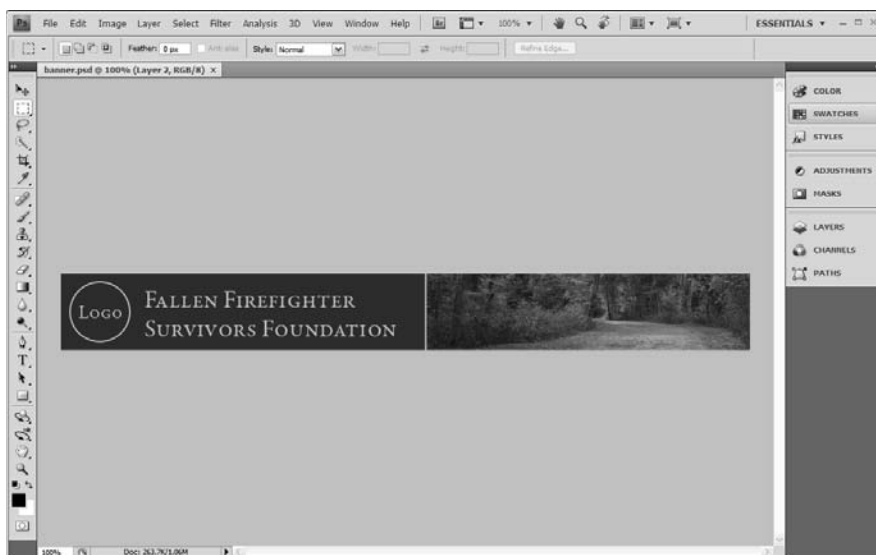


Figure 4-2: Creating the banner in Photoshop CS4.

Fireworks was used to optimize the images for the site. The images were all in one folder, which makes it easy to use Fireworks batch-processing to export, scale, and rename the images. When you use the batch process export option, you can choose one of the presets or you can choose Custom to fine-tune export settings for the site. Doug also used Fireworks to create the navigation menu (see Figure 4-3). The navigation menu was saved in the Fireworks default PNG file format, which makes it possible to edit the menu if the client requests changes. After saving the document, the menu was exported as HTML and images. With the work in Fireworks completed, it was time to put it all together in Dreamweaver.

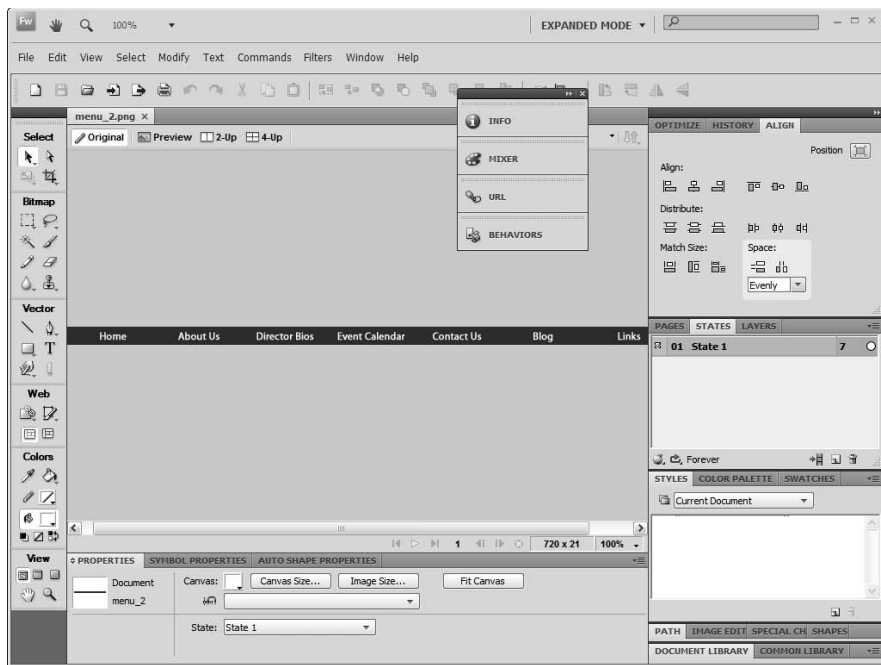


Figure 4-3:
Creating the navigation menu.

Putting It All Together in Dreamweaver

The first step in completing the site is to create a site definition. This information includes the folder in which the site files are stored, the image folder, and the server information. Defining a site is covered in Book III, Chapters 5 and 6. After defining the site, Doug created an HTML document that would serve as the site template. He inserted the banner at the top of the page, and then imported the navigation menu he created in Fireworks. The next step

was to create an editable region for the home page, and to save it as a template (see Figure 4-4). A CSS sheet was created for text formatting and then was attached to the template. The template was used as a basis for the other pages. As of this writing, Doug is waiting on more content to flesh out the site and upload it to the server.

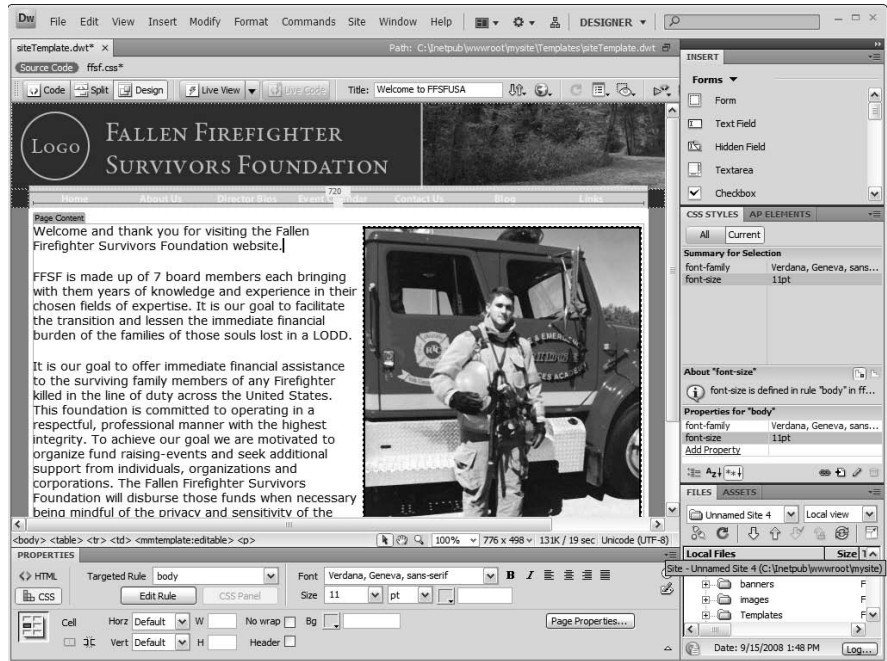


Figure 4-4: Putting it all together in Dreamweaver.

Index

Symbols and Numerics

- == (equal) comparison operator, 526
- != (not equal) comparison operator, 526
- 3D Rotation tool (Flash), 395
- 3D Transform tool (Flash), 395
- 9-Slice Scaling tool (Fireworks), 334

A

- <a> tag, 175
- accessibility (Web site)
 - alternate content, 290–291
 - checking accessibility with Dreamweaver, 295–296
 - color-coding, 291–292
 - creating accessible Web pages, 293–295
 - defined, 83
 - fixing mistakes, 297
 - orientation tools, 293
 - overview, 289–290
 - pictures, 292
 - running reports, 296–297
- accessories, 131–132
- accountants, 30
- ActionScript, 422
- active, 176
- Active Server Pages. *See* ASP (Active Server Pages)
- Active Slideshow Pro
 - adding slides to slideshows, 461–465
 - adding sound to slides, 465–468
 - adding text to slides, 465–468
 - cost, 455
 - creating slide shows, 457–469
 - editing slide shows, 468–469
 - installing extension, 456–457
 - Ken Burns effect, 456
 - launching, 458–461
 - overview, 455
- Active Slideshow Pro dialog box, 459
- ADA (Americans with Disabilities Act), 290
- adding
 - e-commerce with PayPal, 575–578
 - elements to forms, 483–494
 - files to Web pages, 419–420
 - Flash audio to Web pages, 424–428
 - Flash video to Web pages, 447–448
 - functionality to buttons, 416–418
 - Photoshop to graphics toolbar, 340
 - QuickTime Player to Web pages, 449–451
 - RealMedia Player to Web pages, 431–433
 - to sites, 680
 - slides to slide shows, 461–465
 - sound to Web pages, 427–428
 - text and sound to slides, 465–468
 - Windows Media Player to Web pages, 433–435, 451–453
- additive color system, 104
- adjustment layers, 263
- administering Web sites
 - with Contribute, 656–660
- Adobe Audition, 427, 718–719
- Adobe Bridge, 373
- Adobe Contribute, 39, 234
- Adobe Creative Suite 4 (CS4), 133–134
- Adobe Customer Showcase Web site, 316
- Adobe Dreamweaver. *See* Dreamweaver (Adobe)
- Adobe Fireworks. *See* Fireworks (Adobe)
- Adobe Flash. *See* Flash (Adobe)
- Adobe FreeHand, 181
- Adobe GoLive, 141
- Adobe Kuler Web site, 323
- Adobe Media Encoder, 446
- Adobe Photoshop. *See* Photoshop (Adobe)
- Adobe Premiere Pro, 140
- Adobe Web site, 138, 185
- Advanced button (Photoshop), 188
- advertising
 - compared with Search Engine Optimization (SEO), 637–638
 - keyword, 637
- AdWords Help Center (Google), 620
- AIFF (Audio Interchange File Format), 423
- alert box, 250
- alerts, 244
- Alien Skin Software Web site, 377
- Align tool (Photoshop), 192
- Allscoop's RSS Submit Pro, 559

- <alt> tag, 112, 633–636
 - alt text, 290, 630–631
 - AltaVista search engine, 638
 - alternate content, 290–291
 - Americans with Disabilities Act (ADA), 290
 - analytics tools, 40
 - anchor points, 220
 - and logical operator, 526
 - animations
 - creating in Flash, 404–406
 - defined, 139
 - overview, 292
 - using Flash, 420–421
 - Apple
 - Final Cut Pro, 139–140
 - QuickTime multimedia software, 185
 - QuickTime video format, 437
 - Web site, 185
 - applet, 284
 - application IIS object, 500
 - archiving pages, 668
 - arranging page-level information, 98
 - artifacts, 109
 - artwork
 - adjustment layers, 263
 - banner photos, 260
 - blend modes, 263–264
 - clipping masks, 260–261
 - exporting from
 - Fireworks, 366
 - filters, 262–263
 - optimizing in Fireworks, 362–365
 - overview, 259–260
 - ASP (Active Server Pages)
 - creating, 501–506, 506–513
 - creating rotating ad banners, 515–517
 - defining site in
 - Dreamweaver, 502–506
 - displaying random images, 513–514
 - IIS objects used within, 500
 - overview, 496–497, 499–500
 - setting up Web sites, 502
 - variables, 507–511
 - VBScript functions, 511–513
 - working with IIS, 501–502
 - ASP/ASP.NET server side programming language, 278–279
 - Aspect Ratio (Photoshop), 188
 - ASPError IIS object, 500
 - assembling menus in Flash, 414–416
 - assets
 - compiling in
 - Dreamweaver, 726–727
 - creating, 726, 733–735
 - overview, 672–673
 - Assets panel (Dreamweaver), 238–240
 - assigning content development tasks, 41
 - Atome feed, 551
 - audience
 - of client, 44
 - defining, 9–11
 - multimedia choices for, 62
 - audio
 - equipment, 131
 - formats, 423–424
 - Audio Interchange File Format (AIFF), 423
 - Audition (Adobe), 427
 - Auto Shape tools (Fireworks), 337
 - Avid Web site, 140
-
- ## B
-
- b2evolution blog
 - application installing, 694–695
 - managing blogs, 697
 - overview, 544, 694
 - posting entries, 696–697
 - back-end systems
 - bandwidth, 108
 - defined, 28, 60
 - background contents, 87
 - backups, 128–129
 - bandwidth, 108, 570
 - banner graphics
 - components, 369–370
 - creating, 367–369, 367–370
 - logos, 370
 - banners
 - banner exchanges, 623
 - creating rotating ad, 515–517
 - overview, 76
 - placing photos on, 260
 - Bare Bones Software
 - BBEdit, 142
 - base styles, 169
 - behaviors (Dreamweaver)
 - creating pop-up messages, 250–251
 - defined, 229, 243
 - editing via Behaviors panel, 250
 - Extension Manager, 249
 - installing from Adobe Exchange Web site, 249
 - Open Browser Window, 246–249
 - overview, 243–246

Behaviors panel
 (Dreamweaver), 250
 beta tests, 302–303, 304
 Bind tool (Flash), 396
 bitmap images
 compared with vector
 images, 182, 183
 defined, 330
 Black and White tool
 (Flash), 397
 blank keyframes, 398
 Blank Page starter file
 (Dreamweaver),
 205–206
 blend modes, 120, 188,
 263–264
 blog software/services
 b2evolution, 694–697
 Blogger.com, 706
 overview, 693
 WordPress, 697–706
 Blogger.com Web site, 706
 Bloglines RSS feed reading
 service, 551
 blogs
 creating Web sites from,
 707–710
 overview, 543–545
 photoblogs, 710
 as site content, 40
 Blue Microphone Snowball,
 715–717
 Blue Microphone
 Snowflake, 717–718
 Blur tool (Fireworks),
 335–336
 body of HTML document,
 151
 <body> tag, 145
 Bone tool (Flash), 396
 books, 322
 borders, 178

 tag, 152

Braille devices, 67
 brainstorming, 14, 573–575
 branding through E-mail,
 560–561
 breadcrumbs, 85, 293
 Bridge (Adobe), 373,
 386–387
 browser
 checking compatibility,
 226
 simulation programs, 129
 software, 142–143
 using, 26
 Brush tool
 Fireworks, 335
 Flash, 396
 Photoshop, 194
 budget planning, 14–15
 building. *See also* creating
 HTML forms, 476–483
 tables, 154–156
 bulleted lists, 153
 bulletin board, 546
 burn, 555
 Burn tool (Fireworks), 336
 Burns, Ken, 456
 Button Editor (Fireworks),
 345–346
 buttons
 adding functionality in
 Flash, 416–418
 Advanced (Photoshop),
 188
 creating in Fireworks,
 345–351
 creating in Flash, 410–414
 creating multistate,
 350–351
 creating simple two-state,
 346–350
 Get (Dreamweaver),
 215–216
 Preview in Browser
 (Dreamweaver), 216

Put (Dreamweaver),
 215–216
 Refresh in Design View,
 216
 Submit, 477–478

C

call to action, 101, 372
 camcorder, 130
 camera, 130
 CAN-SPAM (Controlling
 the Assault of Non-
 Solicited Pornography
 and Marketing) Act of
 2003, 589
 capturing digital video,
 439–440
 card readers, 132
 Cascading Style Sheets. *See*
 CSS (Cascading Style
 Sheet)
 CGI (common gateway
 interface), 282
 CGI/Perl server side
 programming language,
 282–283
 <channel> tag, 552
 check box groups, 487–488
 check boxes, 486–487
 Checkbox Group dialog
 box, 488
 checking
 browser compatibility,
 226
 code in Dreamweaver, 215
 in/out files, 211
 site accessibility with
 Dreamweaver, 295–296
 checklist, Web site
 publishing, 299–302
 Circular Hotspot tool
 (Fireworks), 338

- class selectors, 171–172
- cleaning up code, 632
- client
 - customers, 54
 - defining voice, 43–44
 - developing look, 45–46
 - logo, 369
 - meeting needs of, 732–733
 - meetings, 731–732
 - mockups, 361–362
 - needs of, 38–40
 - visitors needs, 36–37
- client side
 - defined, 273
 - scripts, 243
 - technology issues, 609–610
- client side programming
 - languages
 - Java, 284–285
 - VBScript, 285–286
 - clipped sound, 436
 - clipping masks, 260–261
 - closing tag, 146
 - CMYK, 103–104
- code
 - checkers (Dreamweaver), 215
 - cleaning up, 632
 - commenting, 168
 - creating PHP, 537–539
 - hard, 478
 - overview, 474–476
 - snippets, 238
 - validating in
 - Dreamweaver, 225–226
 - validators, 214, 215
 - viewing helpers, 214–215
- code samples
 - check boxes, 487
 - conditional statements, 275, 276, 523–525
 - connecting to databases, 541
 - drop-down lists, 491–492
 - event handlers, 495
 - extracting data, 542
 - forwarding information to databases, 602–603
 - functions, 277–278
 - hidden fields, 484
 - loops, 276, 277, 527
 - mailing lists, 601
 - password text fields, 483
 - radio buttons, 489
 - radio groups, 490
 - RSS feeds, 553–554, 558
 - site optimization, 618–619
 - table titles, 541
 - textarea form objects, 485
 - title and meta tags, 631
 - variables, 274, 275, 521–522
 - VBScript, 286
- Code view (Dreamweaver), 212–213
- coding
 - paragraphs to HTML
 - document, 152
 - whole Web page, 156–158
- collaborations, 15
- Collapsed view (Dreamweaver), 211–212
- color
 - adding to CSS file, 121–122
 - CMYK, 104
 - coding, 84, 291–292
 - hexadecimal, 104–105
 - managing with custom panels in Photoshop, 257
 - non-Web safe, 319–321
 - RGB, 104
 - rollover, 218
 - shift, 107
 - Web-safe, 106–108, 319–321
- Color Picker, 164
- Color Profile (Photoshop), 188
- Color Reduction Algorithm (Photoshop), 381–382
- color schemes
 - established, 105–106
 - new, 116
- ColorSchemer application, 323
- ColorVision, 127–128
- comments, 159
- common gateway interface (CGI), 282
- comp image, 114
- company name, 369
- comparison operators, 525–526
- compiling
 - assets in Dreamweaver, 726–727
 - foundation site in
 - Dreamweaver, 735–736
- complex information, 65
- compressing sound, 436
- compression
 - application, 440
 - lossless, 328
- computer, 126–127
- computer monitor, 127–128
- Comstock Images Web site, 317
- concatenated, 511
- conditional statements
 - code sample, 275, 276
 - defined, 275
 - overview, 523–526
- connection keys, 660–662
- connections, 652–656
- consistency, 73–75
- container tag, 146
- content
 - adding to site, 40–41
 - avoiding problems, 99–100
 - developer/development, 25–26, 29, 41, 645–646

- file management, 92–93
- large amounts of, 48–49
- owners, 20
- structure, 96–97
- using personas to develop, 53–54
- context-sensitive panel, 193, 218
- Contribute feature
 - administering Web sites, 656–660
 - creating connections, 652–656
 - editing Web pages, 662–665
 - overview, 39, 651–652
 - sending connection keys, 660–662
- cookies, 41, 588
- copyright
 - e-commerce, 605
 - holder, 114
 - issues, 37–38
 - protecting, 318–319
 - respecting, 316–318
 - for sound, 424
- Corel Painter, 342–343
- Corel products, 181
- Create New Symbol dialog box (Flash), 410–411
- Create Web Photo Album dialog box (Dreamweaver), 388
- creating
 - accessible Web pages, 293–297
 - <alt> tags, 633–636
 - animations in Flash, 404–406
 - ASP pages in
 - Dreamweaver, 506–513
 - assets, 726, 733–735
 - banner graphics, 367–370
 - blank keyframes, 398
 - buttons in Fireworks, 345–351
 - buttons in Flash, 410–414
 - client mockups, 361–362
 - clipping masks, 260–261
 - connections with
 - Contribute, 652–656
 - content for returning visitors, 646–647
 - CSS document, 168–175
 - databases, 534–537
 - descriptions, 633–636
 - e-commerce sites, 590–597
 - Flash audio, 424–427
 - Flash documents, 400–403
 - Flash presentations, 421–422
 - Flash video, 442–447
 - frames, 398
 - images in Corel Painter, 342–343
 - keyframes, 398
 - keywords, 633–636
 - layers in Photoshop, 188–189
 - library items, 241–242, 673–675
 - mailing list forms, 532–534, 599–601
 - multistate buttons in Fireworks, 350–351
 - navigation menus, 351–353, 409–410, 722–724
 - objects in Flash, 403–404
 - page layouts with wire frames, 253–259
 - pages from templates in Dreamweaver, 236
 - photo galleries, 385–389
 - Photoshop workspaces, 186–187
 - PHP code, 537–539
 - PHP pages, 520–521
 - placeholder text, 88–89
 - podcast sites, 712–714
 - pop-up menus, 354–359
 - pop-up messages in Dreamweaver, 250–251
 - portfolio pages, 727
 - printable style sheets, 69
 - profile, 11
 - promotional graphics, 371–374
 - prototype, 117–121
 - reusable graphic templates, 359–360
 - rotating ad banners, 515–517
 - RSS feeds, 552–555
 - scope document, 12–13
 - simple forms in
 - Dreamweaver, 478–482
 - simple two-state buttons, 346–350
 - sites with prebuilt page designs, 230–233
 - slide shows, 457–458
 - snippets, 669–670
 - templates, 234–235, 234–238
 - thumbnails in Fireworks, 724–726
 - titles, 633–636
 - troubleshooting plan, 22
 - Web sites from blogs, 707–710
 - Web sites with
 - Dreamweaver, 221–224
 - wire frame, 85–89
- credit card authorization packages, 579–581
- Credit Card Issuer, 580
- Credit Card Processor Interchange, 580, 581
- Crop tool
 - Fireworks, 335
 - Photoshop, 192–193

- cross-links, 605
 - cross-platform, 127
 - CSS (Cascading Style Sheets)
 - adding new colors, 121–122
 - benefits of, 161–163
 - borders, 178
 - checking source, 163–164
 - commenting code, 168
 - creating with, 23–24
 - defined, 161
 - embedding styles, 166–167
 - external style sheets, 167
 - headline, 176–177
 - inline styles, 166
 - layouts, 158–159
 - padding and margins, 177–178
 - setting class selectors, 171–172
 - setting default selectors, 169–170
 - setting ID selectors, 172–175
 - specifying text parameters with, 324
 - styling links, 175–176
 - using with other tools, 164–165
 - CSS Style Sheet starter files (Dreamweaver), 206
 - CSS Styles panel (Dreamweaver), 219–220
 - CSS Zen Garden Web site, 159
 - customers
 - of client, 44, 54
 - handling, 611–614
 - serving, 22–23
-
- D**
-
- databases
 - creating, 534–537
 - forwarding information to, 602–603
 - retrieving information
 - from, 539–542
 - as site content, 40
 - deck, 131
 - Deco tool (Flash), 396
 - dedicated server, 569
 - default selectors, 169–170
 - defining
 - sites in Dreamweaver, 502–506
 - variables, 275
 - Web stats, 642–643
 - deleting layers in Photoshop, 189
 - delivery method, 64–67
 - <description> tag, 552
 - descriptions
 - creating, 633–636
 - defined, 630–631
 - page overview, 77–80
 - Design view (Dreamweaver), 213–214
 - design (Web site). *See also* Fireworks (Adobe); Photoshop (Adobe)
 - basics, 83–85
 - color charts, 322–323
 - colors, 319–321
 - concept to completion process, 361–362
 - copyright, 316–319
 - creativity tools, 322–323
 - finding ideas, 315–316
 - font issues, 323–324
 - image formats, 325–328
 - print to Web, 324–325
 - type issues, 323–324
 - destructive tool, 194
 - detaching library items, 243
 - developing
 - section-level information architecture, 97–98
 - Web sites, 159–160
 - dialog box (Dreamweaver)
 - Create Web Photo Album, 388
 - Insert FLV, 447
 - List Values, 492
 - New Document, 205
 - Open Browser Window, 247
 - Parameters, 430, 432, 435, 450, 453
 - Preferences, 217, 294
 - Radio Group, 491
 - Remote Info, 223
 - Reports, 297
 - Site Definition, 222, 503–505
 - Synchronize, 228
 - dialog box (Fireworks), Image Size, 375–376
 - dialog box (Flash)
 - Create New Symbol, 410–411
 - Document Properties, 402
 - Duplicate Symbol, 415
 - Import Video, 443
 - New Document, 401
 - Publish Settings, 419
 - Sound Properties, 426
 - dialog box (general)
 - Active Slideshow Pro, 459
 - Checkbox, 488
 - New, 86
 - Save for Web & Devices, 94
 - dialog box (Photoshop)
 - New, 187
 - Save for Web & Devices, 268–269, 378–379
 - digital still camera, 130–131
 - digital video
 - capturing, 439–440
 - editing, 441
 - encoding, 440–442
 - overview, 438–439
 - pitfalls, 453–454
 - Digital Video For Dummies* (Underdahl), 439
 - displaying random images, 513–514
 - Distort tool (Fireworks), 334
 - dithering, 108

- <div> tag, 173
- docking floating panels in Flash, 408
- Document Properties dialog box (Flash), 402
- Document title (Dreamweaver), 215
- document type definition (DTD), 148, 232
- Document window (Dreamweaver), 212
- documenting customer issues, 611–613
- documents
 - creating Flash, 400–403
 - creating navigation menu, 409–410
 - scope, 12–13
 - slicing Photoshop, 264–267
 - source, 160
- Dodge tool (Fireworks), 336
- domain name
 - defined, 566
 - purchasing, 568–569
 - researching, 566–568
 - suffix, 567, 569
- dots per inch (dpi), 182
- downloading
 - Alien Skin Blow Up, 377
 - b2evolution, 694
 - programming scripts, 284
 - SWF studio V3.5, 422
- dpi (dots per inch), 182
- Dreamweaver (Adobe)
 - adding Flash video to Web pages, 447–448
 - adding QuickTime Player to Web pages, 429–431, 449–451
 - adding RealMedia Player to Web pages, 431–433
 - adding Windows Media Player to Web pages, 433–435, 451–453
 - Assets panel, 238–240
 - behaviors, 243–251
 - Behaviors panel, 250
 - Blank Page starter files, 205–206
 - checking site accessibility with, 295–296
 - Code view, 212–213
 - code-viewing helpers, 214–215
 - Collapsed view, 211–212
 - compiling foundation site in, 735–736
 - creating ASP pages in, 506–513
 - creating pages from templates, 236
 - creating photo galleries, 387–389
 - creating PHP pages, 520–521
 - creating pop-up messages, 250–251
 - creating Property Inspector links, 218
 - creating simple forms, 478–482
 - creating sites with prebuilt page designs, 230–233
 - creating templates, 234–235
 - creating templates for site maintenance, 234–238
 - creating Web sites with, 221–224
 - CSS Styles panel, 219–220
 - defining sites in, 502–506
 - Design view, 213–214
 - Document window, 212
 - DTD (document type definition), 148, 232
 - editable regions, 235
 - editing new pages with, 665–667
 - Expanded view, 208–211
 - Extension Manager, 249
 - file management, 215–216
 - Files panel, 208
 - Find and Replace, 66
 - home page, 233
 - Insert panel, 220
 - library items, 238, 240–243
 - managing sites with, 665–668
 - New Document dialog box, 205
 - Open Browser Window behavior, 246–249
 - overview, 141, 184, 201–202
 - Page from Sample starter files, 206–207
 - Preferences dialog box, 217
 - Preview in Browser, 216
 - Property Inspector (panel), 218
 - publishing Web site with, 227–228
 - Refresh in Design view, 216
 - running reports in, 296–297
 - setting page properties, 218–219
 - Split view, 214
 - templates, 668
 - terminology, 204
 - testing Web pages, 225–226
 - Title field, 215
 - updating templates, 236–238
 - uploading content with, 305–308
 - using to insert accessible elements, 293–295
 - validating code, 215, 225–226
 - View options, 216
 - visual aids, 216
 - Welcome screen, 202–204
- Dreamweaver Developer Center Web site, 220

- Dreamweaver Library
 - creating items, 673–675
 - editing items, 676–678
 - overview, 673
 - using items, 675–676
- drop-down lists, 491–494
- Duplicate Symbol dialog box (Flash), 415
- duplicating layers in Photoshop, 188–189
- dynamic multimedia, 61
- dynamic Web pages
 - making with server side technology, 496–497
 - using, 59–61, 65

E

- eBay, 584
- e-commerce sites
 - adding basic e-commerce with PayPal, 575–578
 - brainstorming, 573–575
 - choosing servers, 569
 - client-side technology issues, 609–610
 - cookies, 588
 - creating mailing list forms, 599–601
 - creating sites, 590–597
 - credit card authorization packages, 579–581
 - eBay, 584
 - e-mail address setup, 590–592
 - establishing online identity, 566
 - follow up, 606–607
 - forwarding information to database, 602–603
 - handling customers, 611–614
 - legal considerations, 588–589
 - maintaining, 609–626
 - marketing, 604–605
 - overview, 565
 - PayPal Website Payments Pro, 578–579
 - planning user-friendly sites, 572–573
 - purchasing domain names, 568–569
 - researching domain names, 566–568
 - Search Engine Optimization (SEO), 614–623
 - Secure Sockets Layer (SSL), 581–582
 - security, 605–606
 - selling items with PayPal account, 592–597
 - selling online, 604–607
 - server-side technology issues, 610–611
 - shared servers, 569–570
 - shopping carts, 598–604
 - spam, 588–589
 - technological considerations, 585–586
 - testing, 603–604
 - tips, 582–583
 - upgrading, 624–626
 - usability, 586–588
 - Web hosting, 570–572
- editable regions, 235
- editing
 - behaviors via Behaviors panel, 250
 - digital video, 441
 - Library items, 676–678
 - new pages with Dreamweaver, 665–667
 - slide shows, 468–469
 - sound, 427
 - Web pages with Contribute, 662–665
- educational use, 114
- e-learning materials, 52–53
- Electric Rain Swift 3D, 139
- elements, 483–494
- Ellipse tool (Fireworks), 337
- elseif statement, 525
- E-mail
 - accounts, 571
 - branding through, 560–561
 - newsletters, 40
 - setting up addresses, 590–592
 - using, 26
- embedding
 - QuickTime Player in Web pages, 429–431
 - styles, 166–167
- emphasizing important information, 81
- key points, 46–47
- empty tags, 146
- encoding
 - digital video, 440–442
 - Flash video, 442–447
 - video with Adobe Media Encoder, 446
- Entertainment Weekly Magazine's 25 Best Music Web Sites*, 316
- entry pages, 642
- == (equal) comparison operator, 526
- Eraser tool
 - Fireworks, 335
 - Flash, 397
 - Photoshop, 194–195
- establishing online identity, 566
- evaluating ideas, 14
- event handlers, 494–496
- exit pages, defined, 643
- Expanded view (Dreamweaver), 208–211
- expanding sites, 679–690

Export Area tool
(Fireworks), 335
exporting
 artwork from Fireworks,
 366
 defined, 63
 graphics in Photoshop,
 267–270
extending WordPress,
708–709
eXtensible HyperText
 Markup Language
 (XHTML), 149
Extension Manager
 (Dreamweaver), 249
extensions, 249
external audience, 10–11
external style sheets, 167
Eyedropper tool
 Fireworks, 338
 Flash, 397
 Photoshop, 105

F

facelift, 116
Fair Use Doctrine Web
 site, 38
fast-loading, 36
feed aggregator, 551
feed reader, 551–552
feedback
 giving, 18–19, 31–32
 receiving, 17–18
 Web site, 303
FeedBurner Web site, 555
fields
 hidden, 484
 password, 483–484
File Transfer Protocol.
 See FTP (File Transfer
 Protocol)
files
 adding to Web pages,
 419–420

Blank Page Starter
 (Dreamweaver),
 205–206
checking in/out, 211
CSS Style Sheet starter
 (Dreamweaver), 206
Flash Projector, 421
Frameset starter
 (Dreamweaver),
 206–207
Get command, 210
graphics format, 108–111
management, 92–94
managing in
 Dreamweaver, 215–216
organization consistency,
74
organizing supporting,
160
Page from Sample starter
 (Dreamweaver),
 206–207
Photoshop, 190
publishing in Flash, 419
Put command, 210
read-only, 217
remote, 210
repository, 210
revising, 93
size, 109
Files panel (Dreamweaver),
208–212
FileZilla FTP client Web
 site, 308
Fill Color tool
 Fireworks, 339
 Flash, 397
filters, 262–263
Final Cut Express HD,
139–140
final version, 93–94
Find and Replace
 (Dreamweaver), 66
finding sites to link
 e-commerce site,
 616–618

Fireworks (Adobe)
 Button Editor, 345–346
 creating buttons, 345–351
 creating multistate
 buttons, 350–351
 creating navigation
 menus, 351–353,
 722–724
 creating pop-up menus,
 354–359
 creating reusable graphic
 templates, 359–360
 creating simple two-state
 buttons, 346–350
 creating thumbnails in,
 724–726
 exporting artwork, 366
 interface, 332–339
 optimizing artwork,
 362–365
 optimizing images, 724
 overview, 135–136,
 181, 329
 PNG file format, 331
 Property inspector, 333
 raster images, 329–331
 resizing images, 375–376
 toolbar, 333–339
 using, 24–25
 vector images, 329–331
Flash (Adobe)
 ActionScript, 422
 adding files to Web pages,
 419–420
 adding functionality to
 buttons, 416–418
 assembling menu, 414–
 416
 creating animations,
 404–406
 creating buttons, 410–414
 creating documents,
 400–403
 creating navigation menu
 document, 409–410
 creating objects, 403–404

- Flash (Adobe) (*continued*)
 - creating presentations, 421–422
 - docking floating panels, 408
 - interface, 394
 - modifying workspace, 408–409
 - multimedia software, 185–186
 - overview, 138–139, 181, 393
 - panels, 398–399
 - Property inspector, 399–400
 - publishing files, 419
 - resizing panels, 408
 - setting preferences, 407–408
 - Timeline, 397–398
 - Tools panel, 394–397
 - undocking panels, 408
 - using as animation tool, 420–421
 - workspace setup, 406
 - Flash audio
 - creating, 424–427
 - format, 423
 - Flash interactive
 - content, 53
 - Flash movies, 619
 - Flash Projector file, 421
 - Flash video
 - adding to Web pages in Dreamweaver, 447–448
 - creating, 442–447
 - encoding, 442–447
 - overview, 438
 - Flip4Mac's Windows Media Components for QuickTime, 185
 - fly-out menu, 333
 - fold, 301
 - follow-up, 606–607
 - font-family method, 111
 - fonts
 - issues, 323–324
 - overview, 82
 - using on Web sites, 111–112
 - footer, 80
 - for logical operator, 526
 - for loop, 527
 - </form> tag, 477
 - format of graphics, 108–111
 - formatting date and time, 512
 - forms
 - adding elements, 483–494
 - building HTML, 476–483
 - check box groups, 487–488
 - check boxes, 486–487
 - creating in Dreamweaver, 478–482
 - defined, 476
 - drop-down lists, 491–494
 - form tags, 477
 - hidden fields, 484
 - mailing list, 532–534, 599–601
 - password fields, 483–484
 - radio buttons, 489
 - radio groups, 489–491
 - as site content, 40
 - submit buttons, 477–478
 - testing, 483
 - text fields, 477
 - textarea form objects, 485–486
 - Forms toolbar, 478
 - forums, 545–548
 - forwarding information to databases, 602–603
 - foundation site
 - client meeting, 731–732
 - compiling in Dreamweaver, 735–736
 - creating assets, 733–735
 - meeting client needs, 732–733
 - frame rate, 393
 - frames
 - creating, 398
 - optimizing sites with, 618–619
 - frameset, 206
 - Frameset starter files (Dreamweaver), 206–207
 - fraud, Internet, 581
 - free content, 623
 - Free Transform tool (Flash), 395
 - Freeform tool (Fireworks), 337–338
 - FreeHand (Adobe), 181
 - FreeSticky Web site, 645
 - front end, 60
 - FTP (File Transfer Protocol)
 - defined, 209
 - uploading pages with, 308–310
 - Full Screen with Menus mode (Fireworks), 339
 - Full Screen mode tool (Fireworks), 339
 - functions
 - overview, 277–278
 - VBScript, 511–513
-
- G**
-
- generating random images in PHP, 528–529
 - Get button (Dreamweaver), 215–216
 - Get command (Dreamweaver), 210, 305
 - GIF (Graphics Interchange Format), 325–326
 - optimizing artwork, 362–365
 - overview, 110–111
 - giving feedback, 18–19, 31–32

global navigation, 76–77
 global variables, 275
 GoDaddy.com Web site, 568, 732
 Google AdWords Help Center, 620
 Google Analytics, 40
 Google search engine, 638
 gradient
 fill, 195
 linear, 196
 radial, 196
 Gradient tool
 Fireworks, 339
 Photoshop, 195–198
 Gradient Transform tool (Flash), 395
 graphics
 bandwidth, 108
 banner graphics, 367–370
 creating banner, 367–370
 creating promotional, 371–374
 file formats, 108–111
 file management, 93
 keeping fresh, 41–42
 making transparent, 270–271
 optimizing in Photoshop, 267–270
 photo gallery, 385–389
 photos, 374–385
 promotional graphics, 371–374
 raster, 329–331
 replacing, 117
 tablet, 132
 toolbar, 340
 vector, 329–331
 Graphics Interchange Format. *See* GIF (Graphics Interchange Format)

graphics software
 Adobe Fireworks, 135–136
 Adobe Illustrator, 136–137, 180–181
 Adobe Photoshop, 135, 180
 selecting, 179–182
 groups
 check box, 487–488
 radio, 489–491

H

<h2> tag, 177
 H264/MPEG-4 video format, 438
 Hand tool
 Fireworks, 339
 Flash, 397
 Photoshop, 199
 Handheld, 67
 handling customers, 611–614
 hard-code, 478
 hard disk space, 570
 hardware. *See* recording hardware
 <head> tag, 145, 149
 heading of HTML document, 151–152
 headlines
 overview, 76
 styling, 176–177
 help, professional, 636–637
 hexadecimal colors, 104–105, 164
 hidden fields, 484
 Hide Hotspots and Slice tool (Fireworks), 338
 hiding
 Fireworks tools, 339
 layers in Photoshop, 189

hiring
 accountant, 30
 content developer/writer, 29
 lawyer, 30
 professional help, 636–637
 project manager, 28
 Web designer/new media designer, 28
 Web developer/programmer, 28–29
 Webmaster/host, 30
 hits, 642
 home pages, 233
 HomeSite, 142
 hover, 176
HTML 4 For Dummies (Tittel and Burmeister), 164
 HTML (HyperText Markup Language)
 bandwidth, 108
 body, 151
 building forms, 476–483
 bulleted and numbered lists, 153
 coding paragraphs, 152
 coding Web page, 156–158
 creating with, 23–24
 defined, 145
 document, 145–147
 document type definition, 148
 head tag, 148–151
 headings, 151–152
 tables, 154–156
 whole page, 153
 versus XHTML, 149
 HyperText Markup Language (HTML). *See* HTML

1

- ID selectors, 172–175
- ideas
 - brainstorming/evaluating, 14
 - selling, 15
- IIS objects
 - used within ASP pages, 500
 - working with, 501–502
- Illustrator (Adobe), 136–137
 - graphics software, 180–181
 - overview, 25
- IM (instant messaging), 26
- image formats
 - GIF (Graphics Interchange Format), 325–326
 - JPEG (Joint Photographic Experts Group), 326–327
 - PNG (Portable Network Graphics), 328, 331
- Image Gallery Magic, 728
- image rollover, 229
- Image Size dialog box (Fireworks), 375–376
- images
 - bitmap compared with vector, 182, 183
 - creating in Corel Painter, 342–343
 - displaying random, 513–514
 - enlarging with Alien Skin Blow Up, 377
 - generating in PHP, 528–529
 - optimizing in Fireworks, 724
 - optimizing GIF, 325–326
 - preparing for Web with Photoshop, 341–342
 - raster, 329–331
 - resizing, 182
 - resolution, 375
 - sizes, 182, 375
 - splash, 375
 - vector, 183, 329–331
 - Web site, 292
- implementing marketing, 311–312
- Import Video dialog box (Flash), 443
- improving site
 - searchability, 632
- incorporating Search Engine Optimization (SEO), 629–636
- increasing site visibility
 - with alt text, 632
- InDesign Place
 - command, 374
- industry standard, 133
- information
 - architecture, 96
 - emphasizing
 - important, 81
 - page-level, 98
 - presenting, 49–52
 - section-level, 97–98
- Ink Bottle tool (Flash), 396
- inline styles, 166
- Insert FLV dialog box (Dreamweaver), 447
- Insert panel (Dreamweaver), 220
- inserting accessible elements with Dreamweaver, 293–295
- installing
 - Active Slideshow Pro, 456–457
 - b2evolution blog
 - application, 694–695
 - behaviors from Adobe Exchange Web site, 249
 - Windows IIS 7.0, 502
 - WordPress blog
 - application, 699–700
- instant messaging (IM), 26
- instructional video,
 - including in Web site, 52
- integrating with credit card authorization packages, 579–581
- interactive technologies
 - conditional statements, 275–276
 - functions, 277–278
 - loops, 276–277
 - overview, 273
 - variables, 274–275
- interactivity
 - adding elements to forms, 483–494
 - blog, 543–545
 - building HTML forms, 476–478
 - check boxes, 486–488
 - code, 474–476
 - creating simple forms in Dreamweaver, 478–482
 - drop-down lists, 491–494
 - E-mail, 560–561
 - event handlers, 494–496
 - forums, 545–548
 - hidden fields, 484
 - JavaScript, 494–496
 - making dynamic pages with server side technology, 496–497
 - message boards, 545–548
 - newsletters, 561
 - overview, 473–474
 - password fields, 483–484
 - radio buttons, 486, 489–491
 - RSS feeds, 550–560
 - testing forms, 483
 - textarea form objects, 485–486
 - wiki, 548–549
- interface
 - Fireworks, 332–339
 - Flash, 394

- internal stakeholders, 9–10
- Internet
 fraud, 581
 Gateway, 580, 581
- Internet resources
 Active Slideshow Pro, 455
 Adobe, 138, 185
 Adobe Audition, 718
 Adobe Customer Showcase Web site, 316
 Adobe Kuler, 323
 Alien Skin Software, 377
 AltaVista search engine, 638
 Apple, 185
 Avid, 140
 b2evolution, 544, 684
 blog publicity, 559
 Blogger.com, 706
 Bloglines RSS feed reading service, 551
 Blue Microphone Snowball, 717
 Blue Microphone Snowflake, 717
 ColorSchemer, 323
 Comstock Images, 317
 CSS Zen Garden, 159
 development practices, 159–160
 Dreamweaver Developer Center, 220
 Electric Rain, 139
Entertainment Weekly Magazine's 25 Best Music Web Sites, 316
 Eric Meyer, 159
 FeedBurner, 555
 FileZilla FTP client, 308
 FreeSticky, 645
 GoDaddy.com, 568, 732
 Google search engine, 638
 interactive, 473
- iStockphoto Web site, 317
 Jupiterimages, 317
 KeywordDiscovery, 622
 Kuler, 733
 Microsoft, 142, 185
 Microsoft Visual SourceSafe, 160, 2005
 MSN Search search engine, 638
 NicheBOT, 616
PC Magazine's Top Web Sites 2007, 316
 phpBB, 546
 Pixelicious, 718
 Planetizen Top 10 Web Sites of 2007, 316
 Pluck RSS feed reading service, 551
 PodPress, 713
 RssPublisher, 555
 search engines, 638–639
 Sony Acid Music Studio, 719–720
 Sony Sound Forge, 719
 Sorenson Squeeze, 441
 Sound Booth, 719
 Submit Pro, 559
 SWF Studio V3.5, 422
 Sypder series, 127–128
Time Magazine's 50 Best Web sites 2007, 316
 Toon Boom, 139, 185
 United States Access Board, 290
 U.S. Copyright Office, 38
 Web Marketing Association WebAward, Best Real Estate Web Sites, 316
 WebAssist, 592
 WordPress, 544, 699, 708
 Wordtracker, 622
- World Wide Web Consortium (W3C), 67, 147, 225, 232, 290
 Yahoo! search engine, 639
 YSlow, 300
 Zoom H2 microphone, 715
- interpolation, 182
 issuing bank, 580
 iStockphoto Web site, 317
 <item> tag, 552
 iterative, 9
-
- ## J
-
- Java applet, 285
 Java client side programming language, 284–285
 JavaScript
 overview, 494–496
 server side programming language, 280–281
 JPEG (Joint Photographic Experts Group), 326–327
 optimizing artwork, 365
 overview, 109–110
 Jupiterimages Web site, 317
-
- ## K
-
- key phrase, 630
 key points, 46–47
 keyframes, 398
 keyword advertising, 637
 KeywordDiscovery Web site, 622
 keywords, 77–80, 630–631, 633–636
 kick-off meeting, 15–16
 Knife tool (Fireworks), 338
 Kuler Web site (Adobe's), 323, 733

L

landscape, 375
<language> tag, 552
languages.
 See programming languages
Lasso tool
 Fireworks, 335
 Flash, 395
 Photoshop, 190–191
launching Active Slideshow Pro, 458–461
lawyer
 hiring, 30
 use of, 115
layers
 adjustment, 263
 defined, 188
 mask, 195
Layers panel (Photoshop), 188–189
layout
 considerations, 58
 controlling in CSS, 158–159
 wire frame, 85–89
legal considerations,
 e-commerce sites,
 588–589
Library. *See* Dreamweaver Library
library items
 creating, 241–242
 defined, 238
 detaching, 243
 overview, 240–241
 updating, 242
 using, 242
license of stock images, 113
lights, 132
Line tool
 Fireworks, 336
 Flash, 396
linear gradient, 196

<link> tag, 552, 553
links
 defined, 175
 exchanges, 312, 623
 making with Property Inspector in Dreamweaver, 218
 reciprocal, 644
 sponsored, 312
 styling, 175–176
List Values dialog box (Dreamweaver), 492
lists, drop-down, 491–494
Live Code (Dreamweaver), 215
live site, 204
Live View (Dreamweaver), 215
loading custom panels in Photoshop, 257
local site, 204
local variables, 275
logical operators, 526
logos, 370
loops
 for, 527
 overview, 276–277, 527
 while, 527
lossless compression, 328

M

Macintosh platforms, 26
Magic Wand tool
 Fireworks, 335
 Photoshop, 190–191
Magnify tool (Photoshop), 199
mailing list forms, 532–534, 599–601
maintaining
 e-commerce sites, 609–626
 maintenance agreement, 32
 Web sites, 649–678

making. *See also* building; creating
dynamic pages with server side technology, 496–497
graphics transparent, 270–271
links with Property Inspector in Dreamweaver, 218
managed rights, 114
managing
 blogs in b2evolution, 697
 colors with custom panels in Photoshop, 257
 files in Dreamweaver, 215–216
 project, 21–22
 sites with Dreamweaver, 665–668
 spam from blogs, 707–708
 team, 31
 WordPress blogs, 705–706
manipulating styles in Dreamweaver, 219
marching ants, 191
margin
 customizing, 177–178
 settings, 177–178
marketing
 e-commerce sites, 604–605, 620
 implementing, 311–312
 promoting sites with traditional, 644–645
 redesigned sites, 689–690
 RSS feeds, 558–560
 yourself, 27
Marquee tool
 Fireworks, 335
 Photoshop, 190
meeting
 client needs, 732–733
 kick-off, 15–16

menus
 assembling in Flash, 414–416
 creating Navigation, 351–353
 creating pop-up, 354–359
 fly-out, 333
 navigation, 722–724
 Merchant's Bank Processor, 580
 Merchant's Online Business, 580
 message boards, 545–548
 messages, pop-up, 244
 meta tags, 148, 630–631
 metadata, 148
 Microsoft
 FrontPage, 141
 Visual SourceSafe 2005 Web site, 160
 Visual Web Developer 2008 Express Edition, 142
 Web site, 142, 185
 Windows Media multimedia software, 185
 middleware, 60
 milestones, 12
 mind mapping, 361
 mission of client, 44
 modes, blend, 188, 263–264
 modifying
 Flash workspace, 408–409
 WordPress blogs, 700–703
 monitors, 127–128, 182
 Move tool (Photoshop), 192
 MP3 audio format, 423
 MSN Search search engine, 638
 multimedia
 for all users, 64
 bandwidth, 108
 considerations, 58, 66–67
 element choices, 61–64
 using in project, 26–27

multimedia software
 Adobe Flash, 138–139
 Adobe Premiere Pro, 140
 Apple Final Cut Pro, 139–140
 Electric Rain Swift 3D, 139
 Final Cut Express HD, 139–140
 selecting, 185–186
 Toon Boom Studio/
 Toon Boom Studio Express, 139
 MySQL
 creating database, 534–537
 creating PHP code, 537–539
 mailing list forms, 532–534
 retrieving information from database, 539–542

N

naming home pages, 233
 navigation menus
 creating, 351–353
 creating in Fireworks, 722–724
 defined, 351
 documents, 409–410
 Navigator panel (Photoshop), 199–200
 nesting, 154
 .NET Framework server side programming language, 279–280
 New dialog box (Photoshop), 86, 187
 New Document dialog box
 Dreamweaver, 205
 Flash, 401
 newsletters, 561
 NicheBOT Web site, 616
 9-Slice Scaling tool (Fireworks), 334
 No Color tool (Flash), 397

No Stroke or Fill tool (Fireworks), 339
 non-Web safe colors, compared with Web safe colors, 319–321
 Notepad, 183–184
 != (not equal) comparison operator, 526
 null, 274
 numbered lists, 153

O

ObjectContext IIS object, 500
 object-oriented programming (OOP), 279, 280
 objects
 creating in Flash, 403–404
 textarea form, 485–486
 tag, 153
 online catalog, 40
 online feed reader, 551–552
 online identity, 566
 online journal, 543–545
 online selling, 604–607
 OOP (object-oriented programming), 279, 280
 opacity, 188, 263
 Open Browser Window behavior, 246–249
 Open Browser Window dialog box (Dreamweaver), 247
 opening tag, 146
 operators
 comparison, 525–526
 logical, 526
 opt-in, 589
 optimizing
 artwork in Fireworks, 362–365
 GIF images, 325–326
 graphics in Photoshop, 267–270

- optimizing (*continued*)
 - images in Fireworks, 724
 - sites with frames, 618–619
 - sites for search engines, 614–616
 - techniques, 377
- Options tool (Flash), 397
- Or logical operator, 526
- ordered list, 153
- organizing sites, 360
- orientation tools, 293
- Oval Marquee tool (Fireworks), 335
- Oval Primitive tool (Flash), 396
- Oval tool (Flash), 396
- overlays, 459

p

- <p> tag, 166
- padding
 - customizing, 177–178
 - defined, 170
 - settings, 177
- Page from Sample starter files (Dreamweaver), 206–207
- page-level information, 98
- pages
 - archiving, 668
 - banner, 76
 - creating portfolio, 727
 - description/keywords, 77–80
 - editing with Dreamweaver, 665–667
 - footer, 80
 - global navigation, 76–77
 - headline, 76
 - layout considerations, 58
 - prebuilt, 229
 - title, 77
 - updating, 649–651

- Paint Bucket tool
 - Fireworks, 339
 - Flash, 397
 - Photoshop, 195–198
- palettes, Web-safe, 320–321
- panels (Dreamweaver)
 - Assets, 238–240
 - Behaviors, 250
 - CSS Styles, 219–220
 - Files, 208–212
 - Insert, 220
 - Property Inspector, 218
 - Validation, 225–226
- panels (Flash)
 - docking, 408
 - overview, 398
 - Property inspector, 399–400
 - resizing, 408
 - Tools, 394–397
 - undocking, 408
- panels (Photoshop)
 - Layers, 188–189
 - Navigator, 199–200
 - Text, 198
- paragraphs, coding in HTML document, 152
- parameters, 248
- Parameters dialog box (Dreamweaver), 430, 432, 435, 450, 453
- password fields, 483–484
- Path Scrubber tool (Fireworks), 338
- Path Selection tool (Photoshop), 199
- pay per click, 311, 605, 620–623
- PayPal
 - adding basic e-commerce with, 575–578
 - selling items with account through, 592–597
 - Website Payments Pro, 578–579

- PC Magazine's Top Web Sites 2007, 316
- PDF documents, 53
- Pen tool
 - Fireworks, 336–337
 - Flash, 395
 - Photoshop, 199
- Pencil tool
 - Fireworks, 335
 - Flash, 396
 - Photoshop, 194
- personas, 11, 53–54
- photo galleries
 - creating, 385–389
 - selecting photos, 385–387
- photoblogs, 710
- photographer/portfolio site
 - goals, 721–722
 - site design, 722–729
- photos
 - optimization techniques, 377
 - placing on banners, 260
 - resizing, 374–376
 - Save for Web & Devices command in Photoshop, 377–385
 - selecting for photo galleries, 385–387
- Photoshop (Adobe)
 - adding to graphics toolbar, 340
 - adjustment layers, 263
 - Advanced button, 188
 - artwork, 259–264
 - banner photos, 260
 - blend modes, 263–264
 - Brush tool, 194
 - clipping masks, 260–261
 - color management, 257
 - Color Reduction
 - Algorithm, 381–382
 - creating banner graphics, 367–369

- creating layout wire frame, 86–89
 - Crop tool, 192–193
 - default workspace, 186–187
 - Eraser tool, 194–195
 - Eyedropper tool, 105
 - files, 190
 - filters, 262–263
 - Gradient tool, 195–198
 - graphics software, 180
 - Hand tool, 199
 - Lasso tool, 190–191
 - layers, 188–189, 191
 - Layers panel, 188–189
 - Magic Wand tool, 190–191
 - Magnify tool, 199
 - making graphics
 - transparent, 270–271
 - Marquee tool, 190
 - Move tool, 192
 - Navigator panel, 199–200
 - optimizing graphics, 267–270
 - overview, 135
 - Paint Bucket tool, 195–198
 - Path Selection tool, 199
 - Pen tool, 199
 - Pencil tool, 194
 - preparing images for Web with, 341–342
 - Save for Web & Devices command, 377–385
 - Selection tools, 190–191
 - Shape tool, 198
 - Slice Select tools, 264–267
 - Slice tool, 194, 264–267
 - slicing documents, 264–267
 - Text panel, 198
 - Text tool, 198
 - tools, 189
 - tools panel, 189
 - using, 24
 - Vector tools, 198–199
 - wire frames, 253–259
 - work area, 187–188
- PHP (PHP HyperText PreProcessor), 497
 - conditional statements, 523–526
 - creating code, 537–539
 - creating database, 534–537
 - creating pages, 520–521
 - creating PHP code, 537–539
 - generating random images, 528–529
 - loops, 526–527
 - mailing list forms, 532–534
 - overview, 519
 - retrieving information
 - from database, 539–542
 - server side requirements, 520
 - variables, 521–523
 - PHP server side
 - programming language, 281–282
 - phpBB Web site, 546
 - pictures. *See* images; photos
 - Pixelicious, 718
 - pixels, 182, 330
 - pixels per inch (ppi), 182
 - Place command (InDesign), 374
 - placeholder
 - defined, 16–17
 - text, 88–89
 - Planetizen Top 10 Web Sites of 2007, 316
 - planning
 - budget, 14–15
 - establishing workflow, 31
 - questions, 10
 - revising original, 16–19
 - Search Engine Optimization (SEO), 629–636
 - setting goals, 12
 - timelines, 14–15
 - troubleshooting, 22
 - user-friendly sites, 572–573
- Pluck RSS feed reading service, 551
 - PNG (Portable Network Graphics), 328, 331
 - podcast sites
 - creating, 712–714
 - overview, 711–712
 - recording hardware, 715–718
 - sound-editing software, 718–720
 - Podcasting For Dummies* (Morris, Tomais & Terra), 711
 - PodPress Web site, 713
 - point-and-click, 29
 - Pointer tool (Fireworks), 333
 - polling, 36
 - Polygon tool (Fireworks), 337
 - Polygonal Hotspot tool (Fireworks), 338
 - Polygonal Lasso tool (Fireworks), 335
 - Polygonal Slice tool (Fireworks), 338
 - Polystar tool (Flash), 396
 - pop-up menus
 - creating, 354–359
 - defined, 354
 - pop-up messages
 - creating in Dreamweaver, 250–251
 - defined, 244
 - Portable Network Graphics (PNG), 328, 331
 - portfolio sites. *See* photographer/ portfolio site
 - posting
 - blog entries in
 - b2evolution, 696–697
 - blog entries in WordPress blogs, 703–705
 - ppi (pixels per inch), 182

- prebuilt pages
 - creating sites with, 230–233
 - defined, 229
 - Preferences dialog box (Dreamweaver), 217, 294
 - preferences (Flash), 407–408
 - preparing images for Web with Photoshop, 341–342
 - presentations (Flash), 421–422
 - press releases, 604
 - Preview in Browser button (Dreamweaver), 216
 - preview image, 114
 - print to Web, 324–325
 - printable materials, 67–69
 - printable style sheets, 69
 - printers
 - as needed tools, 129–130
 - relationship with image size, 182
 - product highlighting, 47–48
 - production files, 94–95
 - profiles, 11
 - programmers, 28–29
 - programming languages (client side)
 - Java, 284–285
 - VBScript, 285–286
 - programming languages (server side)
 - ASP/ASP.NET, 278–279
 - CGI/Perl, 282–283
 - JavaScript, 280–281
 - .NET Framework, 279–280
 - PHP, 281–282
 - programming scripts, 284
 - progressively downloading, 64, 109
 - projects
 - handing off to client, 32–33
 - handling solo, 26–27
 - hiring manager, 28
 - management, 21–22
 - reasons for getting started, 13–15
 - promoting sites with traditional marketing, 644–645
 - promotional graphics, 371–374
 - properties, setting in Dreamweaver, 218–219
 - property, 169
 - Property Inspector (Dreamweaver)
 - making links with, 218
 - panel, 218
 - setting page properties, 218–219
 - Property Inspector (Fireworks), 333
 - Property Inspector (Flash), 399–400
 - prosumer, 131
 - protecting copyright, 318–319
 - prototype, 117–121
 - publications, 322
 - Publish Settings dialog box (Flash), 419
 - publishing (Web sites)
 - beta test, 302–303, 304
 - checklist, 299–302
 - files in Flash, 419
 - getting feedback, 303
 - getting opinions, 302
 - going live, 304–310
 - marketing, 311–312
 - requesting link exchanges, 312
 - submitting site to search engines, 310–311
 - tweaking design, 304
 - uploading content with Dreamweaver, 305–308
 - uploading pages with FTP client, 308–310
 - Web sites in Dreamweaver, 227–228
 - purchasing domain names, 568–569
 - Put button (Dreamweaver), 215–216
 - Put command, 210, 305
-
- ## Q
-
- QuickTime audio format, 423
 - QuickTime Player
 - adding to Web pages in Dreamweaver, 449–451
 - embedding in Web pages, 429–431
 - QuickTime/QuickTimePro (Apple) multimedia software, 185
-
- ## R
-
- radial gradient, 196
 - radio buttons, 489
 - Radio Group dialog box (Dreamweaver), 491
 - radio groups, 489–491
 - raster images, 329–331
 - reading Web stats, 641–642
 - read-only files, 217
 - Really Simple Syndication.
See RSS (Really Simple Syndication)
 - RealMedia
 - audio format, 424
 - Player, 431–433
 - video format, 438
 - receiving feedback, 17–18
 - reciprocal links, 644
 - recording hardware
 - Blue Microphone Snowball, 715–717
 - Blue Microphone Snowflake, 717–718
 - podcast sites, 715–718
 - Zoom H2, 715
 - Rectangle Hotspot tool (Fireworks), 338

- Rectangle Primitive tool (Flash), 396
 - Rectangle tool
 - Fireworks, 337
 - Flash, 396
 - Red Eye Removal tool (Fireworks), 336
 - redesigning
 - defined, 116
 - existing site, 19–20
 - sites from scratch, 680–682
 - Redraw tool (Fireworks), 337
 - reference, file management, 92
 - referrers, 643
 - Refresh in Design View
 - button (Dreamweaver), 216
 - registrars, 566
 - remote files, 210
 - Remote Info dialog box (Dreamweaver), 223
 - remote site, 204
 - reordering layers in Photoshop, 189
 - Replace Color tool (Fireworks), 336
 - reports, running with Dreamweaver, 296–297
 - Reports dialog box (Dreamweaver), 297
 - request IIS object, 500
 - requesting link exchanges, 312
 - researching
 - domain names, 566–568
 - software, 134
 - Reshape Area tool (Fireworks), 338
 - resizing
 - images, 182
 - panels in Flash, 408
 - photos, 374–376
 - resolution
 - defined, 330
 - image, 375
 - respecting copyright, 316–318
 - response IIS object, 500
 - retrieving information from databases, 539–542
 - revising plans, 16–19
 - RGB color, 104
 - rights managed, 114
 - rollover
 - buttons, 173
 - color, 218
 - defined, 136
 - royalty-free, 114
 - <rss> tag, 552
 - RSS (Really Simple Syndication)
 - creating feeds, 552–555
 - marketing, 558–560
 - online feed reader, 551–552
 - overview, 550–551
 - validating feeds, 556–557
 - RSS Submit Pro (Allscoop), 559
 - RssPublisher Web site, 555
 - Rubber Stamp tool (Fireworks), 336
 - running reports with Dreamweaver, 296–297
 - rush fee, 15
-
- S
- sans serif fonts, 111
 - Save for Web & Devices command (Photoshop), 377–385
 - Save for Web & Devices dialog box (Photoshop), 94, 268–269, 378–379
 - Scale tool (Fireworks), 334
 - scanners, 129–130
 - scope creep, 13, 32
 - scope document, 12–13
 - scope of variables, 275
 - screen CSS, 68
 - screen readers, 67
 - scripts
 - creating with, 23–24
 - downloading programming, 284
 - Search Engine Optimization (SEO)
 - banner exchanges, 623
 - compared with paid advertising, 637–638
 - defined, 29, 630
 - determining effectiveness, 640–644
 - finding sites to link to e-commerce sites, 616–618
 - Flash movies, 619
 - free content, 623
 - incorporating, 629–636
 - link exchanges, 623
 - marketing, 620
 - optimizing site for search engines, 614–616
 - optimizing sites with frames, 618–619
 - overview, 614
 - pay per click promotion, 620–623
 - planning, 629–636
 - search engine software, 619–620
 - search engines
 - ads, 605
 - optimizing sites for, 614–616
 - software, 619–620
 - spamming the, 639
 - submitting sites to, 310–311, 638–639
 - Web sites, 638–639
 - search strings, 643
 - secondary navigation, 77
 - section-level information architecture, 97–98
 - secure server, 582
 - Secure Sockets Layer (SSL), 581–582
 - security, 605–606

- selecting
 - graphics software, 179–182
 - layers in Photoshop, 188–189
 - multimedia software, 185–186
 - photos for photo galleries, 385–387
 - servers, 569
- Selection tool
 - Flash, 395
 - Photoshop, 190–191
- selector, 169
- selling
 - ideas, 15
 - items with PayPal
 - account, 592–597
 - online, 604–607
- sending connection keys
 - with Contribute, 660–662
- SEO. *See* Search Engine Optimization (SEO)
- serif, 323
- serif fonts, 111
- server IIS object, 500
- server side
 - defined, 273, 586
 - requirements for PHP, 520
 - scripts, 243
- server side programming
 - languages
 - ASP/ASP.NET, 278–279
 - CGI/Perl, 282–283
 - JavaScript, 280–281
 - .NET Framework, 279–280
 - PHP, 281–282
- server side technology
 - defined, 499
 - issues, 610–611
 - making dynamic pages
 - with, 496–497
- server space, 63
- servers
 - choosing, 569
 - dedicated, 569
 - defined, 204
 - shared, 569–570
 - testing, 210
 - upgrading, 625–626
- Session IIS object, 500
- session variables, 279
- Set Default Stroke/Fill
 - Colors tool (Fireworks), 339
- setting
 - class selectors, 171–172
 - Flash preferences, 407–408
 - goals, 12
 - ID selectors, 172–175
 - page properties in
 - Dreamweaver, 218–219
- setting up
 - e-mail addresses, 590–592
 - Flash workspace, 406
 - Web sites with ASP pages, 502
- Shape tool (Photoshop), 198
- sharing
 - servers, 569–570
 - SSL certificate, 582
- Sharpen tool (Fireworks), 336
- shopping cart, 598–604
- Show Hotspots and Slice tool (Fireworks), 338
- showing Fireworks
 - tools, 339
- signoffs, 31
- Site Definition dialog box (Dreamweaver), 222, 503–505
- site redesign
 - analysis, 685–686
 - considerations, 683–684
 - consultation, 684
 - implementation, 687
 - overview, 683, 687–689
 - planning, 686–687
 - review, 684–685
 - testing, 687
- sites. *See also* Web pages
 - accessibility, 289–297
 - adding on to, 680
 - adjusting, 643–644
 - administering with
 - Contribute, 656–660
 - banner, 260
 - creating from blogs, 707–710
 - creating prototype, 117–121
 - creating with
 - Dreamweaver, 221–224
 - creating with prebuilt
 - page designs, 230–233
 - defining in Dreamweaver, 502–506
 - development practices, 159–160
 - expanding, 679–690
 - finding to link e-commerce sites, 616–618
 - foundation, 731–736
 - improving searchability, 632
 - increasing visibility with
 - alt text, 632
 - index, 293
 - interactive, 473
 - maintaining, 649–678
 - management, 571
 - managing with
 - Dreamweaver, 665–668
 - map, 49, 71, 293
 - marketing redesigned, 689–690
 - new color scheme, 116
 - optimizing with frames, 618–619

- optimizing for search
 - engines, 614–616
- organizing, 360
- promoting with traditional marketing, 644–645
- publishing, 299–312
- publishing in
 - Dreamweaver, 227–228
- redesign, 19–20
- redesigning existing, 19–20
- redesigning from scratch, 680–682
- replacing graphics, 117
- setting up with ASP pages, 502
- submitting to search engines, 638–639
- submitting with services, 639–640
- usability, 287–289
- using fonts, 111–113
- size, images, 375
- Skew tool (Fireworks), 334
- Slice Select tool (Photoshop), 264–267
- Slice tool
 - Fireworks, 338
 - Photoshop, 194, 264–267
- slicing Photoshop documents, 264–267
- slide shows
 - creating, 457–458
 - editing, 468–469
- slides
 - adding to slide shows, 461–465
 - adding text/sound, 465–468
- Smudge tool (Fireworks), 336
- snippets
 - creating, 669–670
 - defined, 450
 - overview, 669
 - using, 670–672
- software
 - Adobe Fireworks, 135–136
 - Adobe Flash, 138–139
 - Adobe Illustrator, 136–137
 - Adobe Photoshop, 135
 - Adobe Premiere Pro, 140
 - Apple Final Cut Pro, 139–140
 - blog, 694–706
 - browser, 142–143
 - Electric Rain Swift 3D, 139
 - Final Cut Express HD, 139–140
 - graphics, 179–182
 - multimedia, 185–186
 - researching, 134
 - search engine, 619–620
 - sound-editing, 718–720
 - Toon Boom Studio and Toon Boom Studio Express, 139
 - Web-Design, 140–142, 183–184
- solopreneur, 45
- Sony Acid Music Studio, 719–720
- Sony Sound Forge, 427, 719
- Sony Vegas 7.0, 441
- Sorenson Squeeze Web site, 441
- sound
 - adding RealMedia Player to Web pages, 431–433
 - adding to slides, 465–468
 - adding Windows Media Player to Web pages, 433–435
 - audio formats, 423–424
 - clipped, 436
 - compressing, 436
 - copyright issues, 424
 - creating Flash audio, 424–427
 - editing, 427
 - embedding QuickTime Player in Web pages, 429–431
 - overview, 427–428
 - tips, 436
 - Sound Booth, 718–719
 - Sound Forge (Sony), 427
 - Sound Properties dialog box (Flash), 426
 - sound-editing software
 - Adobe Audition, 718–719
 - Sony Acid Music Studio, 719–720
 - Sony Sound Forge, 719
 - Sound Booth, 718–719
 - source
 - checking in CSS, 163–164
 - control, 211
 - documents, 160
 - Source Code (Dreamweaver), 215
 - spacer graphics, 177
 - spam, 588–589, 707–708
 - spamming the search engine, 639
 - specifying text parameters with CSS, 324
 - splash image, 375
 - Split view (Dreamweaver), 214
 - sponsored links, 312
 - Spray Brush tool (Flash), 396
 - Spyder series Web site, 127–128
 - SSL certificate, 582
 - SSL (Secure Sockets Layer), 581–582
 - stakeholders, 20
 - Standard Screen mode tool (Fireworks), 339

Starter Page category
 (Dreamweaver), 207
 statements, conditional,
 523–526
 static Web pages
 defined, 280
 using, 59, 64–65
 statistics tools, 40
 stock images, 113–115
 storage, 128–129
 storage cards, 132
 Stroke Color tool
 Fireworks, 339
 Flash, 397
 structure
 content, 96
 starting, 91–92
 studio set up, 132–133
 styles
 of client, 44
 external sheets, 167
 manipulating in
 Dreamweaver, 219
 styling
 headlines, 176–177
 links, 175–176
 stylus, 132
 submit buttons, 477–478
 Submit Pro Web site, 559
 submitting
 site to search engines,
 310–311, 638–639
 sites with services,
 639–640
 Subselection tool
 Fireworks, 334
 Flash, 395
 subtractive color system,
 104
 supplementary navigation,
 77
 supporting files, 160
 Swap Colors tool (Flash), 397
 Swap Stroke/Fill Colors tool
 (Fireworks), 339

SWF studio V3.5, 422
 Synchronize dialog box
 (Dreamweaver), 228

T

T1 line, 569
 table data, 156
 table heading, 156
 table row, 156
 tables, building for HTML
 document, 154–156
 tag line, 369
 tags
 </form>, 477
 <a>, 175
 <alt>, 112, 633–636
 <body>, 145

, 152
 <channel>, 552
 <description>, 552
 <div>, 173
 <h2>, 177
 <head>, 145, 149
 <item>, 552
 <language>, 552
 <link>, 552, 553
 , 153
 , 153
 <p>, 166
 <td>, 156
 <th>, 156
 <tr>, 156
 , 153
 tangents, 330
 tapes, 131
 target audience
 defined, 10
 multimedia choices for, 62
 task management tips
 assets, 672–673
 Library, 673–678
 overview, 668–669
 Snippets, 669–672

<td> tag, 156
 team members
 accountant, 30
 content developer/
 writer, 29
 lawyer, 30
 managing, 31
 project manager, 28
 Web designer/new media
 designer, 28
 Web developer/
 programmer, 28–29
 Webmaster/host, 30
 technology. *See*
 also interactive
 technologies
 client-side issues, 609–610
 considerations for
 e-commerce sites,
 585–586
 issues, 609–611
 server-side, 499, 610–611
 templates (Dreamweaver)
 creating, 234–235
 creating pages from in
 Dreamweaver, 236
 creating for site
 maintenance, 234–238
 defined, 229
 editable regions, 235
 overview, 668
 updating, 236–238
 templates (general)
 creating reusable graphic,
 359–360
 for prototype, 118–121
 as site content, 41
 terminology, Web site
 creation, 204
 terms of use, 114
 tertiary navigation, 77
 testing
 e-commerce sites, 603–604
 forms, 483
 server, 204, 210

- traps in usability, 84
 - Web pages in
 - Dreamweaver, 225–226
 - text
 - adding to slides, 465–468
 - box, 478, 479
 - strings, 511
 - text field
 - defined, 478, 479
 - forms, 477
 - Text panel (Photoshop), 198
 - Text tool
 - Fireworks, 337
 - Flash, 396
 - Photoshop, 198
 - textarea form objects, 485–486
 - <th> tag, 156
 - 3D Rotation tool (Flash), 395
 - 3D Transform tool (Flash), 395
 - thumbnails, creating in
 - Fireworks, 724–726
 - tiling background, 50
 - Time Magazine's 50 Best*
 - Web sites 2007, 316
 - Timeline (Flash), 397–398
 - timelines, planning, 14–15
 - tips
 - e-commerce, 582–583
 - sound, 436
 - <title> tag, 552, 553, 631
 - titles
 - creating, 633–636
 - overview, 77
 - tooggling between Fireworks
 - tool, 339
 - toolbar
 - adding Photoshop to
 - graphics, 340
 - Fireworks, 333–339
 - tools (general)
 - accessories, 131–132
 - backups, 128–129
 - cameras and camcorders, 130
 - computer, 126–127
 - computer monitor, 127–128
 - destructive, 194
 - digital still cameras, 130–131
 - graphics tablet, 132
 - printers, 129–130
 - scanners, 129–130
 - storage, 128–129
 - video camera, 131
 - Tools panel (Flash), 394–397
 - tools (Photoshop)
 - Align, 192
 - Brush, 194
 - Crop, 192–193
 - Eraser, 194–195
 - Gradient, 195–198
 - Hand, 199
 - Lasso, 190–191
 - Magic Wand, 190–191
 - Magnify, 199
 - Marquee, 190
 - Move, 192
 - Navigator panel, 199–200
 - overview, 189
 - Paint Bucket, 195–198
 - Path Selection, 199
 - Pen, 199
 - Pencil, 194
 - Selection, 190–191
 - Shape, 198
 - Slice, 194, 264–267
 - Slice Select, 264–267
 - Text, 198
 - vector, 198–199
 - tools (Web sites),
 - Orientation, 293
 - Toon Boom Studio and
 - Toon Boom Studio Express, 139, 185
 - <tr> tag, 156
 - trial version, 134
 - triggers, 354
 - tripod, 132
 - troubleshooting plan, 22
 - TTY, 67
 - tweaking Web site design, 304
 - type issues, 323–324
-
- ## U
-
- tag, 153
 - undocking panels in Flash, 408
 - United States Access Board
 - Web site, 290
 - unordered list, 153
 - up time, 571
 - updating
 - library items, 242
 - pages, 649–651
 - templates in
 - Dreamweaver, 236–238
 - upgrading
 - e-commerce sites, 624–626
 - servers, 625–626
 - Web hosting, 624–625
 - uploading
 - content using
 - Dreamweaver, 305–308
 - pages with FTP client, 308–310
 - U.S. Copyright Office Web
 - site, 38
 - usability
 - defined, 83
 - e-commerce sites, 586–588
 - overview, 287–289
 - testing, 16
 - user groups, 134
 - user preferences,
 - multimedia choices for, 62
 - using
 - library items, 242, 675–676
 - Snippets, 670–672

U

validating
code in Dreamweaver, 215, 225–226
RSS feeds, 556–557
Validation panel (Dreamweaver), 225–226
variables
defined, 274
PHP, 521–523
sample code, 274, 275
session, 279
using, 507–511
VBScript client side
programming language, 285–286
VBScript functions, 511–513
VCR, 131
vector images
compared with bitmap images, 182, 183
defined, 183
overview, 329–331
Vector Path tool (Fireworks), 337
vector tools (Photoshop), 198–199
version control
defined, 211
using, 159–160
video
adding Flash video to Web pages, 447–448
adding QuickTime Player to Web pages, 449–451
adding Windows Media Player to Web pages, 451–453
camera, 131
codec, 440
creating Flash video, 442–447
digital, 438–442

digital video pitfalls, 453–454
encoding with Adobe Media Encoder, 446
encoding Flash video, 442–447
Web video formats, 437–438
View Source technique, 163
views (Dreamweaver)
Code, 212–213
Collapsed, 211–212
Design, 213–214
Expanded, 208–211
options, 216
Split, 214
visitors, 36–37, 176, 642
visual aids (Dreamweaver), 216
voice of client, 43–44
VTR, 131
VU meters, 436

W

watermark, 114
Web designer, 28
Web developer, 28–29
Web hosting, 570–572, 624–625
Web Marketing Association
WebAward, Best Real Estate Web Sites, 316
Web pages. *See also* sites
adding files, 419–420
adding Flash audio, 424–428
adding Flash video in Dreamweaver, 447–448
adding QuickTime Player in Dreamweaver, 449–451
adding RealMedia Player in Dreamweaver, 431–433

adding sound, 427–428
adding Windows Media Player in Dreamweaver, 433–435, 451–453
banner, 76
coding, 156–158
creating accessible, 293–297
description/keywords, 77–80
dynamic, 59–61
editing with Contribute, 662–665
embedding QuickTime Player, 429–431
footer, 80
global navigation, 76–77
headline, 76
static, 59, 280
testing in Dreamweaver, 225–226
title, 77
uploading with FTP client, 308–310
Web safe
colors compared with non-Web safe colors, 319–321
defined, 106
Web site maintenance
administering Web site, 656–660
assets, 672–673
Contribute, 651–665
creating connections, 652–656
Dreamweaver Library, 673–678
Dreamweaver templates, 668
editing Web pages, 662–665
managing with Dreamweaver, 665–668

- sending connection key, 660–662
- Snippets, 669–672
- task management tips, 668–678
- updating pages, 649–650
- Web sites
 - Active Slideshow Pro, 455
 - Adobe, 138, 185
 - Adobe Audition, 718
 - Adobe Customer Showcase Web site, 316
 - Adobe Kuler, 323
 - Alien Skin Software, 377
 - AltaVista search engine, 638
 - Apple, 185
 - Avid, 140
 - b2evolution, 544, 684
 - blog publicity, 559
 - Blogger.com, 706
 - Bloglines RSS feed reading service, 551
 - Blue Microphone Snowball, 717
 - Blue Microphone Snowflake, 717
 - ColorSchemer, 323
 - Comstock Images, 317
 - creation terminology, 204
 - CSS Zen Garden, 159
 - Dreamweaver Developer Center, 220
 - Electric Rain, 139
 - Entertainment Weekly Magazine's 25 Best Music Web Sites*, 316
 - Eric Meyer, 159
 - FeedBurner, 555
 - FileZilla FTP client, 308
 - FreeSticky, 645
 - GoDaddy.com, 568, 732
 - Google search engine, 638
 - iStockphoto Web site, 317
 - Jupiterimages, 317
 - KeywordDiscovery, 622
 - Kuler, 733
 - Microsoft, 142, 185
 - Microsoft Visual SourceSafe, 160, 2005
 - MSN Search search engine, 638
 - NicheBOT, 616
 - PC Magazine's Top Web Sites 2007*, 316
 - phpBB, 546
 - Pixelicious, 718
 - Planetizen Top 10 Web Sites of 2007, 316
 - Pluck RSS feed reading service, 551
 - PodPress, 713
 - RssPublisher, 555
 - search engines, 638–639
 - Sony Acid Music Studio, 719–720
 - Sony Sound Forge, 719
 - Sorenson Squeeze, 441
 - Sound Booth, 719
 - Submit Pro, 559
 - SWF Studio V3.5, 422
 - Sypder series, 127–128
 - Time Magazine's 50 Best Web sites 2007*, 316
 - Toon Boom, 139, 185
 - United States Access Board, 290
 - U.S. Copyright Office, 38
 - Web Marketing Association WebAward, Best Real Estate Web Sites, 316
 - WebAssist, 592
 - WordPress, 544, 699, 708
 - Wordtracker, 622
 - World Wide Web Consortium (W3C), 67, 147, 225, 232, 290
 - Yahoo! search engine, 639
 - YSlow, 300
 - Zoom H2 microphone, 715
- Web stats
 - defining, 642–643
 - reading, 641–642
- Web video formats, 437–438
- WebAssist Web site, 592
- Web-design software, 140–142, 183–184
- Web-Design Studio, 132–333
- Webinars, 54
- Webmaster, 30
- Web-safe colors, 106–108
- Web-safe palette, 320–321
- Welcome screen (Dreamweaver), 202–204
- while loop, 527
- wiki, 548–549
- Windows Media Components for QuickTime (Flip4Mac), 185
- Windows Media Player
 - adding to Web pages in Dreamweaver, 433–435, 451–453
 - audio format, 424
 - multimedia software, 185
- Windows Media Video format, 438
- Windows platforms, 26
- wire frame
 - creating, 85–89
 - creating page layouts with, 253–259
 - defined, 71
- word processing, 26
- WordPress blog
 - application, 544
 - extending, 708–709
 - installing, 699–700
 - managing blogs, 705–706
 - modifying blogs, 700–703
 - overview, 697–699
 - posting blog entries, 703–705
 - Web site, 708

Wordtracker Web site, 622
work area
 Fireworks, 332–333
 Photoshop, 187–188
workflow
 establishing with
 client, 27
 establishing during
 planning, 31
working file, 93
workspace
 default Photoshop,
 186–187
 modifying Flash, 408–409
 setting up Flash, 406

World Wide Web
 Consortium (W3C)
 Web site, 67, 147, 225,
 232, 290
writer, 29
WYSIWYG, 140

X

XHTML (eXtensible
HyperText Markup
Language), 149

Y

Yahoo! search engine, 639
YSlow Web site, 300

Z

Zoom H2 microphone,
 715–716
Zoom tool
 Fireworks, 339
 Flash, 397

BUSINESS, CAREERS & PERSONAL FINANCE

Accounting For Dummies, 4th Edition*

978-0-470-24600-9

Bookkeeping Workbook For Dummies†

978-0-470-16983-4

Commodities For Dummies

978-0-470-04928-0

Doing Business in China For Dummies

978-0-470-04929-7

E-Mail Marketing For Dummies

978-0-470-19087-6

Job Interviews For Dummies, 3rd Edition*†

978-0-470-17748-8

Personal Finance Workbook For Dummies*†

978-0-470-09933-9

Real Estate License Exams For Dummies

978-0-7645-7623-2

Six Sigma For Dummies

978-0-7645-6798-8

Small Business Kit For Dummies, 2nd Edition*†

978-0-7645-5984-6

Telephone Sales For Dummies

978-0-470-16836-3

BUSINESS PRODUCTIVITY & MICROSOFT OFFICE

Access 2007 For Dummies

978-0-470-03649-5

Excel 2007 For Dummies

978-0-470-03737-9

Office 2007 For Dummies

978-0-470-00923-9

Outlook 2007 For Dummies

978-0-470-03830-7

PowerPoint 2007 For Dummies

978-0-470-04059-1

Project 2007 For Dummies

978-0-470-03651-8

QuickBooks 2008 For Dummies

978-0-470-18470-7

Quicken 2008 For Dummies

978-0-470-17473-9

Salesforce.com For Dummies, 2nd Edition

978-0-470-04893-1

Word 2007 For Dummies

978-0-470-03658-7

EDUCATION, HISTORY, REFERENCE & TEST PREPARATION

African American History For Dummies

978-0-7645-5469-8

Algebra For Dummies

978-0-7645-5325-7

Algebra Workbook For Dummies

978-0-7645-8467-1

Art History For Dummies

978-0-470-09910-0

ASVAB For Dummies, 2nd Edition

978-0-470-10671-6

British Military History For Dummies

978-0-470-03213-8

Calculus For Dummies

978-0-7645-2498-1

Canadian History For Dummies, 2nd Edition

978-0-470-83656-9

Geometry Workbook For Dummies

978-0-471-79940-5

The SAT I For Dummies, 6th Edition

978-0-7645-7193-0

Series 7 Exam For Dummies

978-0-470-09932-2

World History For Dummies

978-0-7645-5242-7

FOOD, HOME, GARDEN, HOBBIES & HOME

Bridge For Dummies, 2nd Edition

978-0-471-92426-5

Coin Collecting For Dummies, 2nd Edition

978-0-470-22275-1

Cooking Basics For Dummies, 3rd Edition

978-0-7645-7206-7

Drawing For Dummies

978-0-7645-5476-6

Etiquette For Dummies, 2nd Edition

978-0-470-10672-3

Gardening Basics For Dummies*†

978-0-470-03749-2

Knitting Patterns For Dummies

978-0-470-04556-5

Living Gluten-Free For Dummies†

978-0-471-77383-2

Painting Do-It-Yourself For Dummies

978-0-470-17533-0

HEALTH, SELF HELP, PARENTING & PETS

Anger Management For Dummies

978-0-470-03715-7

Anxiety & Depression Workbook For Dummies

978-0-7645-9793-0

Dieting For Dummies, 2nd Edition

978-0-7645-4149-0

Dog Training For Dummies, 2nd Edition

978-0-7645-8418-3

Horseback Riding For Dummies

978-0-470-09719-9

Infertility For Dummies†

978-0-470-11518-3

Meditation For Dummies with CD-ROM, 2nd Edition

978-0-471-77774-8

Post-Traumatic Stress Disorder For Dummies

978-0-470-04922-8

Puppies For Dummies, 2nd Edition

978-0-470-03717-1

Thyroid For Dummies, 2nd Edition†

978-0-471-78755-6

Type 1 Diabetes For Dummies*†

978-0-470-17811-9

* Separate Canadian edition also available

† Separate U.K. edition also available

Available wherever books are sold. For more information or to order direct: U.S. customers visit www.dummies.com or call 1-877-762-2974. U.K. customers visit www.wileyurope.com or call (0) 1243 843291. Canadian customers visit www.wiley.ca or call 1-800-567-4797.



INTERNET & DIGITAL MEDIA

AdWords For Dummies

978-0-470-15252-2

Blogging For Dummies, 2nd Edition

978-0-470-23017-6

Digital Photography All-in-One

Desk Reference For Dummies, 3rd Edition

978-0-470-03743-0

Digital Photography For Dummies, 5th Edition

978-0-7645-9802-9

Digital SLR Cameras & Photography For Dummies, 2nd Edition

978-0-470-14927-0

eBay Business All-in-One Desk Reference For Dummies

978-0-7645-8438-1

eBay For Dummies, 5th Edition*

978-0-470-04529-9

eBay Listings That Sell For Dummies

978-0-471-78912-3

Facebook For Dummies

978-0-470-26273-3

The Internet For Dummies, 11th Edition

978-0-470-12174-0

Investing Online For Dummies, 5th Edition

978-0-7645-8456-5

iPod & iTunes For Dummies, 5th Edition

978-0-470-17474-6

MySpace For Dummies

978-0-470-09529-4

Podcasting For Dummies

978-0-471-74898-4

Search Engine Optimization For Dummies, 2nd Edition

978-0-471-97998-2

Second Life For Dummies

978-0-470-18025-9

Starting an eBay Business For Dummies, 3rd Edition†

978-0-470-14924-9

GRAPHICS, DESIGN & WEB DEVELOPMENT

Adobe Creative Suite 3 Design Premium All-in-One Desk Reference For Dummies

978-0-470-11724-8

Adobe Web Suite CS3 All-in-One Desk Reference For Dummies

978-0-470-12099-6

AutoCAD 2008 For Dummies

978-0-470-11650-0

Building a Web Site For Dummies, 3rd Edition

978-0-470-14928-7

Creating Web Pages All-in-One Desk Reference For Dummies, 3rd Edition

978-0-470-09629-1

Creating Web Pages For Dummies, 8th Edition

978-0-470-08030-6

Dreamweaver CS3 For Dummies

978-0-470-11490-2

Flash CS3 For Dummies

978-0-470-12100-9

Google SketchUp For Dummies

978-0-470-13744-4

InDesign CS3 For Dummies

978-0-470-11865-8

Photoshop CS3 All-in-One Desk Reference For Dummies

978-0-470-11195-6

Photoshop CS3 For Dummies

978-0-470-11193-2

Photoshop Elements 5 For Dummies

978-0-470-09810-3

SolidWorks For Dummies

978-0-7645-9555-4

Visio 2007 For Dummies

978-0-470-08983-5

Web Design For Dummies, 2nd Edition

978-0-471-78117-2

Web Sites Do-It-Yourself For Dummies

978-0-470-16903-2

Web Stores Do-It-Yourself For Dummies

978-0-470-17443-2

LANGUAGES, RELIGION & SPIRITUALITY

Arabic For Dummies

978-0-471-77270-5

Chinese For Dummies, Audio Set

978-0-470-12766-7

French For Dummies

978-0-7645-5193-2

German For Dummies

978-0-7645-5195-6

Hebrew For Dummies

978-0-7645-5489-6

Ingles Para Dummies

978-0-7645-5427-8

Italian For Dummies, Audio Set

978-0-470-09586-7

Italian Verbs For Dummies

978-0-471-77389-4

Japanese For Dummies

978-0-7645-5429-2

Latin For Dummies

978-0-7645-5431-5

Portuguese For Dummies

978-0-471-78738-9

Russian For Dummies

978-0-471-78001-4

Spanish Phrases For Dummies

978-0-7645-7204-3

Spanish For Dummies

978-0-7645-5194-9

Spanish For Dummies, Audio Set

978-0-470-09585-0

The Bible For Dummies

978-0-7645-5296-0

Catholicism For Dummies

978-0-7645-5391-2

The Historical Jesus For Dummies

978-0-470-16785-4

Islam For Dummies

978-0-7645-5503-9

Spirituality For Dummies, 2nd Edition

978-0-470-19142-2

NETWORKING AND PROGRAMMING

ASP.NET 3.5 For Dummies

978-0-470-19592-5

C# 2008 For Dummies

978-0-470-19109-5

Hacking For Dummies, 2nd Edition

978-0-470-05235-8

Home Networking For Dummies, 4th Edition

978-0-470-11806-1

Java For Dummies, 4th Edition

978-0-470-08716-9

Microsoft® SQL Server™ 2008 All-in-One Desk Reference For Dummies

978-0-470-17954-3

Networking All-in-One Desk Reference For Dummies, 2nd Edition

978-0-7645-9939-2

Networking For Dummies, 8th Edition

978-0-470-05620-2

SharePoint 2007 For Dummies

978-0-470-09941-4

Wireless Home Networking For Dummies, 2nd Edition

978-0-471-74940-0

OPERATING SYSTEMS & COMPUTER BASICS

iMac For Dummies, 5th Edition

978-0-7645-8458-9

Laptops For Dummies, 2nd Edition

978-0-470-05432-1

Linux For Dummies, 8th Edition

978-0-470-11649-4

MacBook For Dummies

978-0-470-04859-7

Mac OS X Leopard All-in-One Desk Reference For Dummies

978-0-470-05434-5

Mac OS X Leopard For Dummies

978-0-470-05433-8

Macs For Dummies, 9th Edition

978-0-470-04849-8

PCs For Dummies, 11th Edition

978-0-470-13728-4

Windows® Home Server For Dummies

978-0-470-18592-6

Windows Server 2008 For Dummies

978-0-470-18043-3

Windows Vista All-in-One Desk Reference For Dummies

978-0-471-74941-7

Windows Vista For Dummies

978-0-471-75421-3

Windows Vista Security For Dummies

978-0-470-11805-4

SPORTS, FITNESS & MUSIC

Coaching Hockey For Dummies

978-0-470-83685-9

Coaching Soccer For Dummies

978-0-471-77381-8

Fitness For Dummies, 3rd Edition

978-0-7645-7851-9

Football For Dummies, 3rd Edition

978-0-470-12536-6

GarageBand For Dummies

978-0-7645-7323-1

Golf For Dummies, 3rd Edition

978-0-471-76871-5

Guitar For Dummies, 2nd Edition

978-0-7645-9904-0

Home Recording For Musicians For Dummies, 2nd Edition

978-0-7645-8884-6

iPod & iTunes For Dummies, 5th Edition

978-0-470-17474-6

Music Theory For Dummies

978-0-7645-7838-0

Stretching For Dummies

978-0-470-06741-3

Home:Dummies

DUMMIES.COM The Online Resource for the Rest of Us!

SEARCH

MY ACCOUNT SHOPPING CART HELP

Everyday Computing Advanced Computing The Internet At Home Health Mind & Spirit Making & Managing Money Sports & Leisure Travel Beyond the Classroom

Win the Ultimate Back-to-School Prize Package!

oTips: Subscribe to our free newsletters. Sudoku/Puzzles: Catch the Fever! Free Sudoku and Kakuro Puzzles each day! Podcasts: Whether reading a Dummies book or listening to a Dummies podcast, when you get Dummies, you get results. RSS: It's all free to you so what are you waiting for? Dummies Deals: At a store near you.

Featured Books & Articles

Focus on Football

How Football Began

Understanding Football's Downs, Yardage, and Stuff

How the Football Season is Set Up

Understanding Football's Defensive Line

Scoring in Football

Resource Center

iPhone

Office2007/Vista

Adobe CS3

Basic Math & Pre-Algebra For Dummies

SEARCH

Around the World View other Dummies editions

United Kingdom

Canada

Australia

Translations

What's a Gigabook?

Win a \$500 Gift Card

Dummies Biographies

Spring Cleaning

Featured Product

Wine For Dummies Champagne Hites and Book Set

More Info >

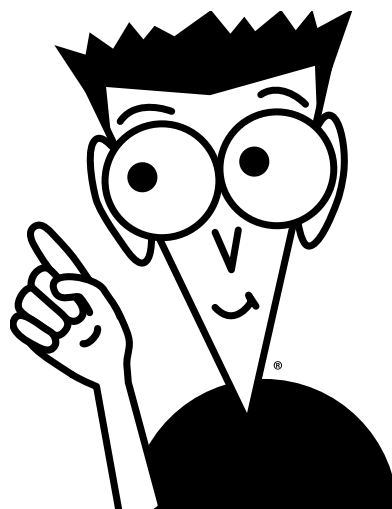
Product Shop

- Consumer Electronics
- Crafts & Sewing Patterns
- Culinary
- and more >

About Dummies | Sign Up for eTips | Sign Up for RSS Feeds | Custom Publishing | Contact Us

Get smart @ dummies.com®

- Find a full list of Dummies titles
- Look into loads of FREE on-site articles
- Sign up for FREE eTips e-mailed to you weekly
- See what other products carry the Dummies name
- Shop directly from the Dummies bookstore
- Enter to win new prizes every month!

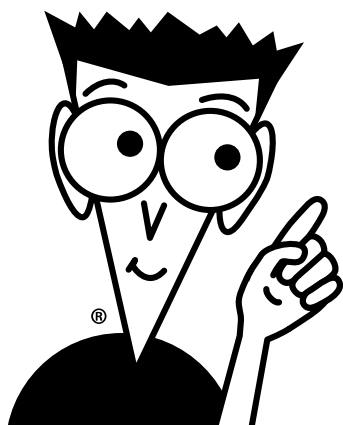
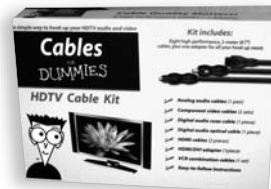
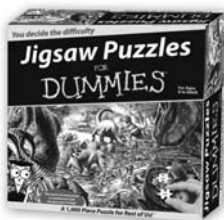
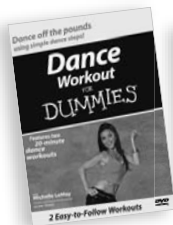
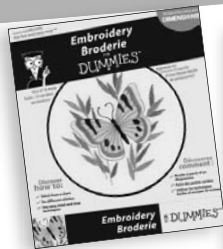


* Separate Canadian edition also available

† Separate U.K. edition also available

Available wherever books are sold. For more information or to order direct: U.S. customers visit www.dummies.com or call 1-877-762-2974. U.K. customers visit www.wiley.co.uk or call (0) 1243 843291. Canadian customers visit www.wiley.ca or call 1-800-567-4797.

Do More with Dummies Products for the Rest of Us!



**DVDs • Music • Games • DIY
Consumer Electronics • Software • Crafts
Hobbies • Cookware • and more!**

Check out the Dummies Product Shop at www.dummies.com for more information!



Plan, build, and maintain a Web site that does exactly what you need!

Want to launch a Web site but don't know where to begin? This book is your answer! There's a minibook covering each step of the process, from deciding what your site should do to working with HTML, using Dreamweaver, incorporating Flash creations, and keeping your site on the cutting edge. There are even case studies to give you great ideas!

- **Start off right** — plan your site, decide whether you need a Web team, and create relevant content
- **Design basics** — develop your site design, work with wire frames, and organize behind-the-scenes files that make your site work
- **Choose your tools** — select the right hardware and software and learn to create pages with Dreamweaver®
- **Get graphic** — discover how to create cool graphics for your site with Photoshop® and Fireworks®
- **Live up your site** — create interest with Flash® animations, slideshows, video, and sound
- **Audience participation** — make your site interactive with JavaScript®, PHP, ASP, and MySQL®
- **Make your site pay** — build an e-commerce site that's user friendly, legally sound, and secure
- **Ongoing success** — keep your site fresh and up to date

Doug Sahlin is a media expert who has written on topics ranging from Web building to digital photography to YouTube. **Claudia Snell** is a professional digital media producer who has worked on Web sites large and small. They are coauthors of the previous edition of this bestseller.



Open the book and find:

- How to use the latest versions of Dreamweaver, Flash, Photoshop, and Fireworks
- Inspiration for great content and graphics
- How to create a Web page from a Photoshop file
- Tips for designing effective layouts
- Ways to build navigation menus with Flash
- The workhorse graphics your site needs
- How to maintain an e-commerce site
- When to consider a complete redesign

Go to **dummies.com**®
for more!

For Dummies®
A Branded Imprint of



\$34.99 US / \$37.99 CN / £22.99 UK

ISBN 978-0-470-38541-8



