

PLANNING THE SITE

When you complete this chapter, you will be able to:

- ◆ Create a site specification
- ◆ Identify the content goal
- ◆ Analyze your audience
- ◆ Build a Web site development team
- ◆ Create conventions for filenames and URLs
- ◆ Set a directory structure
- ◆ Create a site storyboard

A good Web site design requires a detailed initial planning phase. Before starting to code your site, pick up a pencil and paper and sketch out your site design. Creating the stylistic conventions and conceptual structure of your site beforehand saves time during development. This chapter walks you through planning and building a framework for your site, resulting in less recoding when you actually sit down at the computer.

CREATE A SITE SPECIFICATION

What are your objectives for building a Web site? You may want to increase communication among employees, gain visibility, provide a service, attract new customers, or simply show the world you can code XHTML. Because properly maintained Web sites take a lot of work, make sure you have valid and achievable goals for your site.

Start your planning by creating a **site specification**; this is the design document for your site. If you completed the Hands-on Projects and Case Project at the end of Chapter 1, you created a basic draft of a project proposal. You can use some of that information in your site specification. After you read this chapter, you will be able to answer a number of additional questions about your site. You can return to the site specification as you build your site to help maintain your focus. If you are providing Web site design to a client, the site specification is the first document they see that establishes the basic site design. You can visit the Online Companion Web site for this book to look at some sample site specification documents. Answer the following questions in your site specification:

- Why are you building the Web site? Write a two- or three-paragraph mission statement that briefly states the site's goals. What do you or your company or organization hope to gain from creating and maintaining a Web site?
- How will you judge the success of the site? What are the measuring factors you can use to assess the effectiveness of the site?
- Who is the target audience? What characteristics does the audience share, and how can you find out more about them?
- What are the limiting technical factors affecting your site?

IDENTIFY THE CONTENT GOAL

Consider carefully what type of site you are building. What you want the Web site to accomplish and what your users want from your site may differ. For example, site designers are often concerned with the visual aspects of a Web site, such as the quality of the graphics and the use of animation. Your users probably care more about how quickly they can find information. Adopt your users' perspective. Think about the type of content you are presenting and look to the Web for examples of how best to present it. The following types of Web sites demonstrate ways to focus your content. You can view samples of each type of site on the Online Companion Web site for this book.

- *Billboard*—These sites establish a Web presence for a business or commercial venture. In many cases they are informational and offer no true Web-based content, acting as an online brochure rather than offering Web-based interaction. Many businesses build this type of site first and then slowly add functions such as online ordering and product demonstrations as they become more comfortable with the medium.

- *Publishing*—Most major newspapers and periodicals now publish both to print media and to the Web. These Web sites are some of the most ambitious in breadth and depth of content, often containing multiple levels of information with many page templates. Many publishing sites use special software to create Web pages using content from the same databases that produced the paper-based versions. This allows their authors to write the article once but have it published to multiple destinations, such as the daily newspaper and the Web site.
- *Portal*—Portals act as a gateway to the Web and offer an array of services including searching, e-mail, shopping, news, and organized links to Web resources. Many of the major search engines have been converted into portals to attract more users. These sites are often heavy with advertising content, which is their main source of revenue.
- *Special interest, public interest, and nonprofit organization*—These sites include news and current information for volunteers, devotees, novices, a specific audience, or the general public. Public-service Web sites contain links, information, downloadable files, addresses, and telephone numbers that can help you solve a problem or find more resources. Nonprofit organizations can state their manifesto, seek volunteers, and foster a grassroots virtual community.
- *Blog*—Short for “Weblog,” a blog is a personal Web page that reflects the personality and interests of the author. No matter what your interest, a community of *bloggers* (blog authors) on the Web is devoted to it. Blog Web design varies greatly, reflecting the skills (or lack thereof) of the author. Take the time to visit some blogs and discover the wide range of their expression and design. You can easily find blogs by using a blog directory such as www.blogwise.com or www.blogarama.com.
- *Virtual gallery*—The Web is a great place to show off samples of all types of art and design. Photographers and artists can display samples of their work; musicians can post audio files of their songs; writers can offer sections of text or complete manuscripts. However, keep in mind that any copyrighted material you display on a Web site can be downloaded to a user’s machine without your permission. Software companies such as Digimarc (www.digimarc.com) offer digital watermarking technology that lets artists embed digital copyright information in their electronic files as a deterrent to piracy of proprietary content. This information cannot be seen or altered by the user.
- *E-commerce, catalog, and online shopping*—The Web has become a viable shopping medium that continues to expand as more users improve their Internet access and learn to trust the security of online commerce. Web commerce already has begun to compete successfully with traditional retailing, offering many advantages over mail-order shopping, such as letting the customer know immediately whether an item is in stock. Other types of commerce on the Web include stock trading, airline ticketing, and auctions. Many software

vendors offer turnkey systems that can be integrated with existing databases to speed the development of a commerce site. A good e-commerce site provides users with quick access to the item they want, detailed product descriptions, and easy, secure ordering.

- *Product support*—The Web is a boon to consumers who need help with a product. Manufacturers can disseminate information, upgrades, troubleshooting advice, documentation, and online tutorials through their Web site. Companies that provide product support information on the Web often find that the volume of telephone-based customer support calls decreases. Software companies especially benefit from the Web; users can download patches and upgrades and use trial versions of software before they buy.
- *Intranet and extranet*—An **intranet** is a smaller, limited version of the Internet on a company's private local area network (LAN) accessible only to those who have access to their network. Many companies have telecommuting employees who need access to company policies, documentation, parts lists, pricing information, and other materials. These employees can be reached via an **extranet**, which is a part of the private intranet extended outside the organization via the Internet. Many organizations mandate a particular browser for employee use, making the Web designer's job a little easier, because they only have to code and test for one browser.

ANALYZE YOUR AUDIENCE

If possible, analyze your audience and produce an **audience definition**, a profile of your average user. If you are building a new site, work from your market research, look at sites with content similar to yours, and try to characterize your average user. If you have an existing user base, contact your typical users and try to answer the following questions:

- What do users want when they come to your site?
- How can you initially attract users and entice them to return?
- What type of computer and connection speed does your typical visitor have?

Obtaining answers to these questions is especially difficult when your medium is the Web. Though your users may fit no common profile, there are a few ways you can gather information about them. One way is to include an online feedback form in your site. Figures 3-1 and 3-2 show a portion of an online survey recently used on the IBM Life Sciences Web site.

IBM Life Sciences : User Survey - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print

Select a country

IBM Life Sciences >

IBM Life Sciences

User survey

Your opinion counts

Thank you for visiting the IBM Life Sciences Web site!

In our on-going effort to improve our site and its usefulness to you, we would appreciate your feedback about your experience on our site.

For questions 1-7, please use the following scale to rate your overall satisfaction with the following aspects:

Very Dissatisfied 1 2 3 4 5 6 7 Very Satisfied

Your privacy

1. Graphical appeal
2. Page layout
3. Navigation
4. Organization of the information
5. Value of the information
6. Amount of information
7. Ease in finding information

Figure 3-1 User feedback form part 1

IBM Life Sciences : User Survey - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Links >>

7. Ease in finding information

8. When you navigate through a Web site using a site map, which is your preferred method of locating information?

By index

By category

9. What will be your primary purpose for visiting the IBM Life Sciences Web site in the future?

10. Did you expect this site to contain information about a topic that was not covered? If so, what was that topic?

Additional comments:

Figure 3-2 User feedback form part 2

The survey asks users about their experiences visiting the Web site. It uses both scaled and open-ended questions to elicit a variety of responses from the user concerning the visual and information design of the site.

If you cannot survey your users, or if you feel you are not getting good survey results, try to adopt a typical user's perspective as you define your audience. Here are some questions to consider:

- Who are the typical members of your audience? Are they male or female? Do they have accessibility issues? What is their level of education? What is their reading and vocabulary level? What is their level of technical aptitude?
- Why do people come to your site? Do they want information? Do they want to download files? Are they looking for links to other Web sites?
- Do you have a captive audience, such as a base of loyal customers who want up-to-date information? Are you designing for an intranet, where users are employees of an organization?
- If users are unfamiliar with the site, will they know what you offer?
- How often will users return to your site? Why would they come back?
- What computing platform do your users use? What is their typical connection speed? What type of browser do they use? If you are on an intranet, is there a standard for browsers, connection, and screen resolution?
- Whose skills do you need to build the site? Who will create the graphics, code the pages, and write the text? Do you have the talent and economic resources that you need? Will the results meet the expectations of your users?

Refine your content and presentation even after your site is built and running. Continue soliciting user feedback to keep your site focused and the content fresh.

Identify Technology Issues and Constraints

Make your best effort to identify any technological factors—limitations or advantages—that members of your audience share. As you read in Chapter 1, you have to make assumptions about the user's browser, connection speed, operating system, and screen resolution. If you think your user is the average person browsing the Web, you may have to adopt settings that represent the lowest common denominator to satisfy the widest audience. If you feel that you have a primarily high-tech or computer-savvy audience, higher resolution or connection speed may apply. If you are designing an intranet site, you may have the luxury of knowing your users' exact operating systems and browser versions. Whatever the particulars, make sure to design at an appropriate level, or you risk losing visitors.

Identify Software Tools

Determining the software requirements for your Web site is important during the planning process. Try to choose software that matches the complexity and needs of your site so that you do not end up with a tool that is either underequipped or overspecialized. Simple Web sites, including many student sites, can be built with text editors such as Notepad or SimpleText. As your site and skills grow, you can move up to more robust tools such as Macromedia Dreamweaver (www.macromedia.com), Microsoft FrontPage, or Adobe GoLive (www.adobe.com). These tools offer complete coding, design, and site management capabilities. You may also need graphics tools (discussed in Chapter 8), database software, and online credit and shopping programs, based on the needs of the different members of your Web site team, as described in the next section.

One popular type of software is **shareware**, programs that you can download and use for a trial period. Users can then register the software for a relatively small fee compared to commercially produced software. Because shareware is usually developed by individuals or small software companies, registering it is important to support future development efforts. One of the most popular and commonly used shareware programs is WinZip, from Nico Mak Computing, Inc., at www.winzip.com. WinZip lets you work with .zip archive files, the PC standard for file compression and archiving. If you are sending or receiving files via e-mail, you need WinZip to compress and uncompress them. If you have a Macintosh, you can use Stuffit to compress your files. Stuffit Deluxe and Stuffit Lite, created by Aladdin Systems, Inc., are available in shareware versions at www.aladdinsys.com. If you are a PC user and someone sends you a Stuffit file, you can expand it with Aladdin's Expander program, which is available free of charge at the Aladdin site. Shareware programs are also available to help you with Web site development. Two great shareware sites that have WinZip as well as hundreds of other programs are Shareware.com (www.shareware.com) and Tucows.com (www.tucows.com).

BUILD A WEB SITE DEVELOPMENT TEAM

Although one person can maintain small Web sites, larger sites require groups of people filling a variety of roles. Of course, the line between these roles can be blurred, and many aspects of site design require collaboration to solve a problem. The following are examples of the types of talent necessary to build a larger, well-conceived site.

- *Server administrators*—Get to know and appreciate the technical people who run your Web server. They take care of the sticky technical issues such as firewalls, ports, internal security, file administration, and backup procedures. Consult with them to determine your Web site's default filename and directory structure. They also can generate reports detailing how many visitors your site is attracting, where the visitors are coming from, and what pages they like best.

- *XHTML coders*—These are the people responsible for creating the XHTML code, troubleshooting the site, and testing the site across different operating systems and Web browsers. Most XHTML coders are now using XHTML editing programs to create code, but any self-respecting XHTML author knows how to open the XHTML file in a text editor and code by hand. Knowing how to work directly with the code frees you from dependency on one particular authoring tool and makes you more desirable to companies hiring XHTML authors.
- *Designers*—Designers are the graphic artists responsible for the look of the site. They use graphic design software, such as Adobe Photoshop or Macromedia Fireworks. Designers are responsible for the page template design, navigation icons, color scheme, and logos. If your site uses photographic content, the designers are called upon to prepare the photos for online display. They might also create animations and interactive content using Macromedia Flash.
- *Writers and information designers*—Writers prepare content for online display, including hypertext information and navigation paths. Additionally, many writers are responsible for creating a site style guide and defining typographic conventions, as well as consistency, grammar, spelling, and tone. They also work closely with the designers to develop page templates and interactive content.
- *Software programmers*—Programmers write the programs you need to build interaction into your site. They may write a variety of applications, including Common Gateway Interface (CGI) scripts, Java scripts, and back-end applications that interact with a database. Commerce sites especially need the talents of a programming staff.
- *Database administrators*—The people who are responsible for maintaining the databases play an important role in commercial Web sites. They make sure that your data is accessible and safe.
- *Marketing*—The Marketing Department can generate content and attract visitors to the site.

CREATE CONVENTIONS FOR FILENAMES AND URLS

Before you sit down at the keyboard, plan the filename conventions for your site. Find out from your system administrator what type of operating system your Web server uses. Typically you develop your Web site locally on a PC or Macintosh and upload the files to the Web server as the last step in the publishing process. If the Web server runs a different operating system from your local development system, any filename or directory structure inconsistencies encountered in transferring your files to the server may break local URL links.

Naming Files

A filename's maximum length, valid characters, punctuation, and sensitivity to uppercase and lowercase letters vary among operating systems, as described in Table 3-1.

Table 3-1 Filenaming conventions

Operating System	Filename Conventions
ISO 9660 Standard	The filename consists of a maximum of eight letters followed by a period and a three-letter extension. Allowed characters are letters, numbers, and the underscore (_)
DOS and Windows 3.x (FAT file system)	The same as ISO 9660 but with the following additional characters allowed: \$ % ' ` - @ ^ ! & [] () #
Microsoft Windows/NT, NTFS, and Windows 95 VFAT, Windows 98 FAT32, Windows 2000 NTFS, Windows XP NTFS	Maximum 255 letters, all characters allowed except \ / * " < >
Macintosh	Maximum 31 letters, all characters allowed except the colon (:)
UNIX	Maximum 255 letters, all characters allowed except the forward slash (/) and spaces

Case Sensitivity

If you have an image file named `Picture.gif` for example, and you reference that file as ``, the image is displayed properly on a Macintosh or Windows machine. On a UNIX server, however, the image does not load properly because UNIX is case sensitive; `"Picture.gif"` and `"picture.gif"` are recognized as two different files. It is best to use lowercase letters for all filenames, including filenames in your XHTML code.

Character Exceptions

Like case sensitivity, character use also is incompatible between operating systems. For example, the filename `my stuff.htm` is valid on a Windows PC or Macintosh, but not on a UNIX machine because of the space in the filename. If you transfer a Web site containing `my stuff.htm` to a UNIX server, the links to the file will not work. As another example, the filename `<section2>.htm` is valid on a Macintosh or UNIX machine, but the file would not be recognizable to a Windows NT server because the `<>` characters are not allowed. It is best when naming your files to leave out special characters such as `<`, `>`, `/`, `\`, `&`, `*`, and blank spaces.

File Extensions

You must use the correct file extensions to identify your file to the browser. XHTML text files must end in .htm or .html; whichever you choose, set it as a standard convention for your site. Be careful to add this extension when you are working in Notepad, which defaults to saving as .txt. You also must correctly identify image file formats in the file extensions. Joint Photographic Experts Group (JPEG) files must end in .jpg or .jpeg; Graphics Interchange Format (GIF) files must end in .gif; and Portable Network Graphic (PNG) files must end in .png.

Solving the Filename Dilemma

The best way to overcome the restrictions of case sensitivity, character exceptions, and file extensions is to use the convention specified by the International Standards Organization (ISO) for all your files. This convention (often called 8.3, pronounced “eight-dot-three”) specifies a maximum of eight characters followed by a period and a three-character extension. Allowed characters are letters, numbers, and the underscore character. Here are some examples of 8.3 filenames:

- mypage.htm
- chap_1.htm
- picture1.jpg
- logo.gif

If you use the 8.3 filenames convention on your development system, you will have fewer filename problems when you transfer your files to the Web server, regardless of the server’s operating system. By sticking with this filename format, you ensure that your files can be transferred across the greatest number of operating systems. Do not forget to use lowercase characters and omit special characters from your filenames to maximize compatibility.

The Default Main Page Name

Every Web site has a default main page that appears when the browser requests the directory of the site rather than a specific file. The URL for such a page always includes a trailing (forward) slash, as in *www.mysite.com/*. In this instance the Web server provides the index file, which usually is named *index.htm*. Windows NT, however, defaults to an index filename of *default.htm*, and other servers may be set to other names such as *main.htm* or *home.htm*. Before you start coding, check with your system administrator to verify the correct main page filename.

Using URLs

Although you may know that URLs are the addresses you type into your browser to access a site, you may not realize that there are two types of URLs: complete and partial.

Complete URLs

A **Uniform Resource Locator (URL)** is the unique address of a file's location on the World Wide Web. A **complete URL** includes the protocol the browser uses, the server or domain name, the path, and the filename. Figure 3-3 shows an example of a complete URL.

```
<a href="http://www.yoursite.com/business/trends/laptop.htm">
```

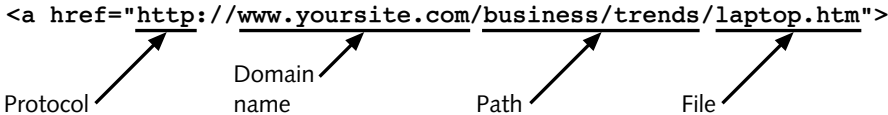


Figure 3-3 Parts of a complete URL

In this example, *http* is the protocol, and *www.yoursite.com* is the domain name. The path shows that the destination file, *laptop.htm*, resides in the *business/trends* folder. Use complete URLs in your XHTML code when linking to another server.



TIP

When you are browsing the Web, you do not need to enter the protocol because the browser defaults to `http://`. However, when creating links in your code, you must always include the protocol with a complete URL; otherwise, the browser does not know how to connect to the location you specify.

Partial URLs

Use a partial URL when you are linking to a file that resides on your own computer or server. **Partial URLs** omit the protocol and domain or server name, and specify the path to the file on the same server. Files that reside in the same directory need no path information other than the filename. The following code shows an example of a partial URL.

```
<a href="laptop.htm">link text</a>
```

SET A DIRECTORY STRUCTURE

When you complete your site, you publish your files on the Web by transferring them to a Web server. A typical Web server has a user area that contains folders for each user; your files are stored in your user area, and files from other Web sites are stored in their user areas. The directory structure of the Web server affects the format of your site's URL.

Figure 3-4 shows a typical Web server directory structure. If you do not register a domain name for your site, you will have a URL that reflects your path in the public area of the Web server. A user enters the following address in the browser to access User2's Web site: `www.websserver.com/user2/`.

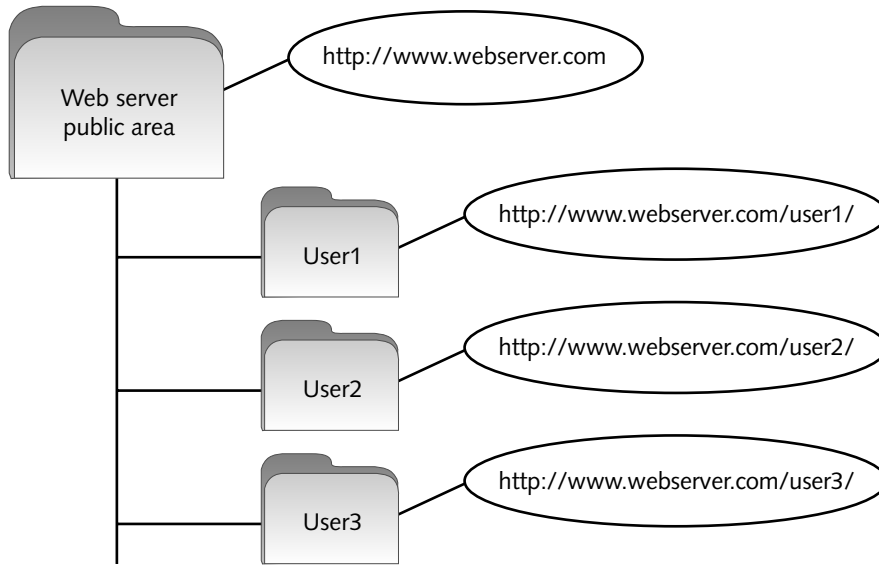


Figure 3-4 Typical Web server directory structure

A domain name is an alias that points to your actual location on the Web server, as shown in Figure 3-5. User2 has purchased the domain name *www.mysite.com*. The actual path to User2's content has not changed, but the visitor to the site sees only the domain name. Now User2 can advertise the Web site with a URL that's easy to remember.

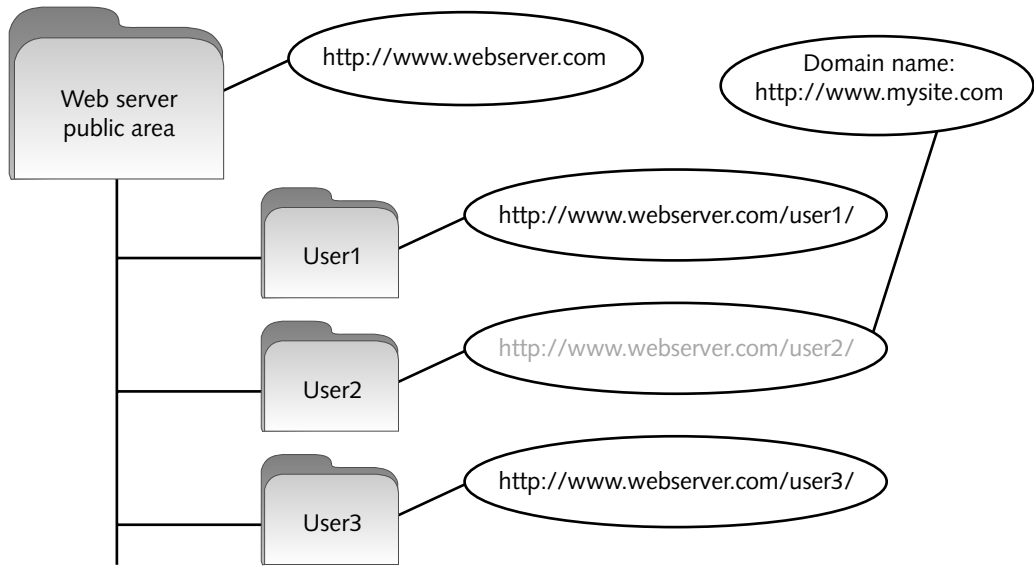


Figure 3-5 Domain name hides the actual path

Relative versus Absolute Paths

You will probably build your Web site on a computer that is different from the computer that hosts your site. Keep this in mind when you are designing the directory and file structure. Because your files will be transferred to another computer, any URLs you specify to link to other pages in your site must include paths that are transferable. This is why you should never specify an absolute path in your partial URLs. An absolute path points to the computer's root directory, indicated by a leading (forward) slash in the file path:

```
/graphics/logo.gif
```

If you include the root directory in your partial URLs, you are basing your file structure on your development machine. If the files are moved to another machine, the path to your files will not apply, and your site will include links that do not work because the browser cannot find the files.

Relative paths tell the browser where a file is located relative to the document the browser currently is viewing. Because the paths are not based on the root directory, they are transferable to other computers.

Building a Relative File Structure

The easiest way to ensure that all your path names are correct is to keep all of your XHTML and image files in the same directory. Because all files are kept together, the only information you need to put in the `src` or `href` attribute is the filename itself. In Figure 3-6, User2 has simplified the directory structure. To reference the file `logo.gif`, User2 adds the following code in one of the XHTML files:

```

```

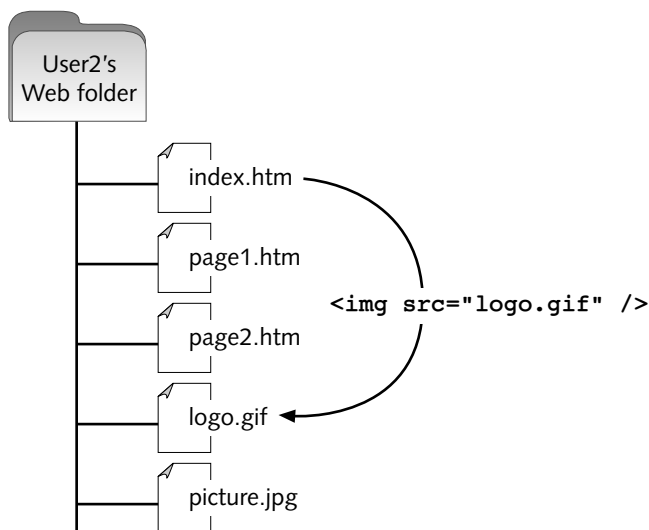


Figure 3-6 Simplified single folder file structure

The simple directory structure shown in the preceding example is fine for a small Web site, but as your site grows you may want to segregate different types of content into separate folders for ease of maintenance. Take a look at the relative file structure for User2's Web site as depicted in Figure 3-7. Notice that User2's Web folder contains three XHTML files and one subfolder named images, which contains the graphics and pictures for the Web site.

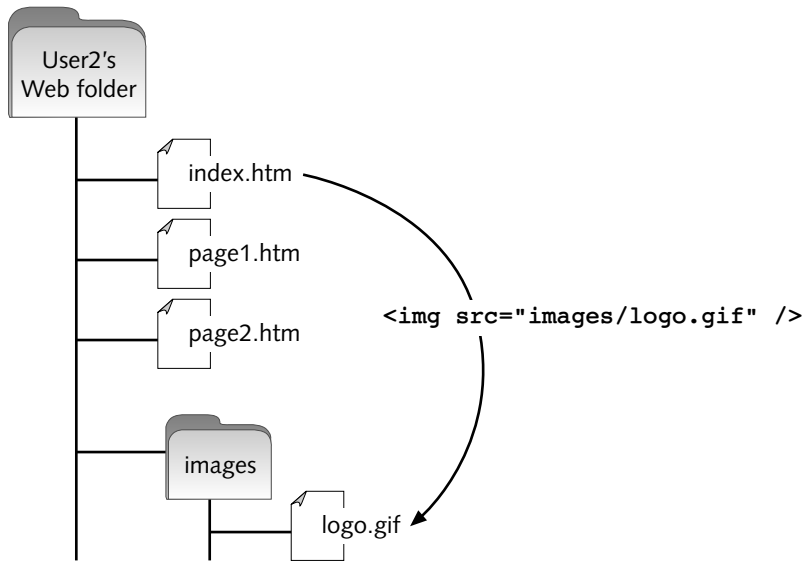


Figure 3-7 Basic relative file structure

To include the image file `logo.gif` in `index.htm`, User2 adds the following code to `index.htm`:

```

```

The path in the `src` value tells the browser to look down one level in the directory structure for the `images` folder and find the file `logo.gif`. The path to the file is relative to the file the browser is viewing. This type of relative file structure can be moved to different machines; the relationship between the files does not change, because everything is relative within the Web folder.

User2's Web site may need a more segregated directory structure, as shown in Figure 3-8. In this example, common files such as the index and site map reside in the top-level folder. Multiple subfolders contain chapter and image content. Two linking examples are illustrated in this figure:

- *Example 1*—To build a link from `page1.htm` (in the `chapter1` folder) to `index.htm`, use `../` in the path statement to indicate that the file resides one level higher in the directory structure, as shown in the following code:

```
<a href="../index.htm">Home</a>
```

- *Example 2*—To include the image file `logo.gif` in `page1.htm`, use `../` to indicate that the file resides in the `images` folder, which is one level higher in the directory structure, as shown in the following code:

```

```

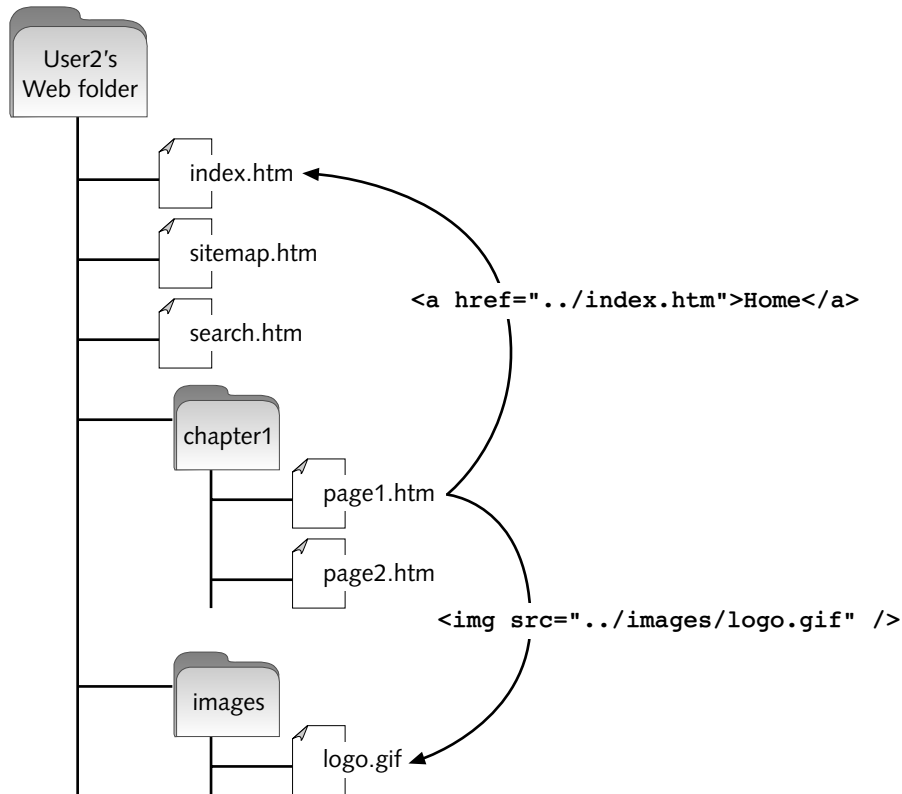


Figure 3-8 More segregated relative file structure

CREATE A SITE STORYBOARD

Plan your site by creating a storyboard flowchart that shows the structure and logic behind the content presentation and navigation choices you offer. You can sketch your site with paper and pencil or create it using flowcharting software. Sometimes it is helpful to use sticky notes or cards to plan the structure visually. This method lets you easily move pages from one section or level to another. Whichever method you choose, this preliminary planning step is one of the most important in planning your site. You can move pages and whole sections of content freely, plan navigation paths, and visualize the entire site. This is the stage at which to experiment and refine your designs. Once you have started coding the site, it is much more difficult and time consuming to go back and make major changes. Remember to adhere to the file naming conventions for each of your pages.

Organize the Information Structure

Think about your users' information needs and how they can best access the content of your site. How should your information design map look? Review the sample structures provided in this section and judge how well they fit your information. Your design may incorporate several different structures, or you may have to adapt the structures to your content. Each sample structure is a template; you may have more or fewer pages, sections, topics, or links. You may choose to use bidirectional links where only single-direction links are indicated. Use these examples as starting points and design from there.

Linear Structure

The linear information structure, illustrated in Figure 3-9, guides the user along a straightforward path. This structure lends itself to booklike presentations; once into the content, users can navigate backward or forward. Each page can contain a link back to the main page if desired. Pages may also contain links to a related subtopic. If the users jump to the subtopic page, they can return only to the page that contains the subtopic link. This structured navigation returns them to the same point in the content path.

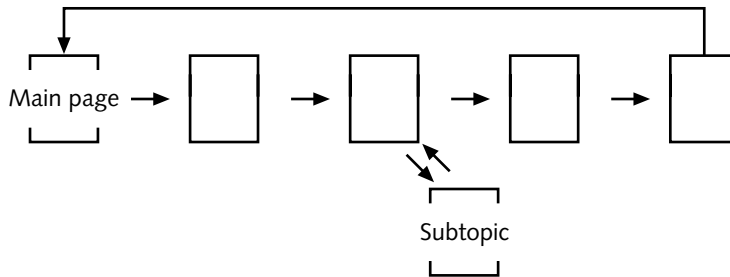


Figure 3-9 Linear structure

Tutorial Structure

The tutorial structure illustrated in Figure 3-10 is perfect for computer-based training content such as lessons, tutorials, or task-oriented procedures. The tutorial structure builds on the simple linear structure in Figure 3-9. The user navigates through the concept, lesson, and review pages in order. Because the lessons use hypertext, users can leave the lesson structure and return at any time. They also can choose the order of lessons and start anywhere they wish. Notice that the table of contents, index, and site map pages are linked to—and from—all pages in the course. Within each lesson users can navigate as necessary to familiarize themselves with the content before they review. This structure can be adapted to fit content needs; for example, the group of pages in the illustration could be one section of a larger training course.

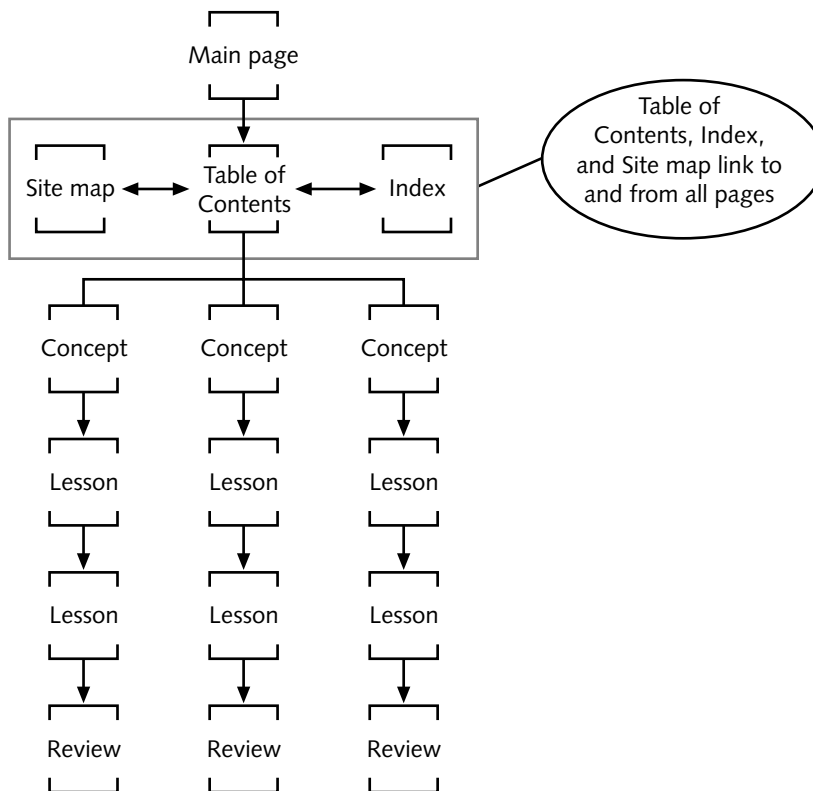


Figure 3-10 Tutorial structure

Web Structure

Many smaller Web sites follow the Web-type content structure illustrated in Figure 3-11, which is nonlinear, allowing the user to jump freely to any page from any other page. If you choose to use this type of content structure, make sure that each page includes clear location information and a standardized navigation bar that not only tells users where they are, but where they can go.

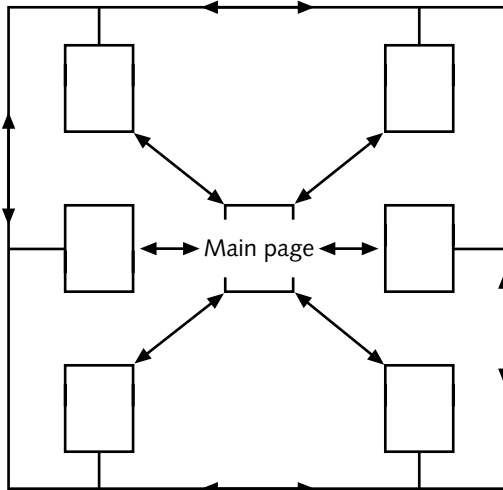


Figure 3-11 Web structure

Hierarchical Structure

The hierarchical structure illustrated in Figure 3-12 is probably the most commonly used information design. It lends itself to larger content collections because the section pages break up and organize the content at different levels throughout the site. Navigation is primarily linear within the content sections. Users can scan the content on the section page, and then choose the content page of their choice. When they finish reading the content, they can return to the section page. The site map allows users to navigate freely throughout the site. A navigation bar on each page lets the user jump to any section page, the main page, and the site map.

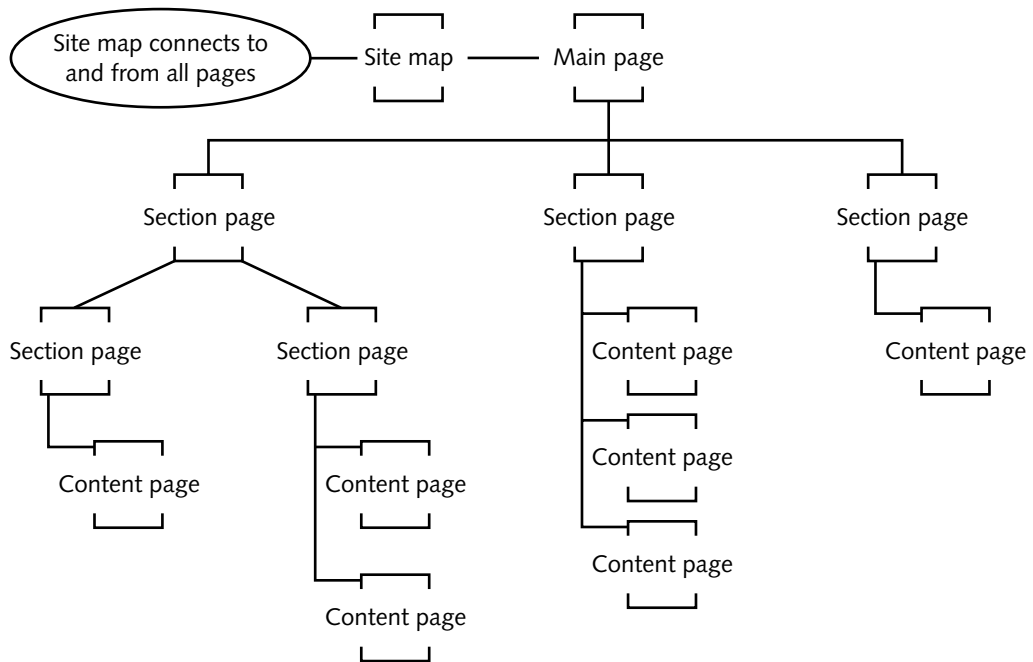


Figure 3-12 Hierarchical structure

Cluster Structure

The cluster structure illustrated in Figure 3-13 is similar to the hierarchical structure, except that every topic area is an island of information, with all pages in each cluster linked to each other. This structure encourages exploration within a topic area, allowing the user to navigate freely through the content. All pages contain a navigation bar with links to the section pages, main page, and site map.

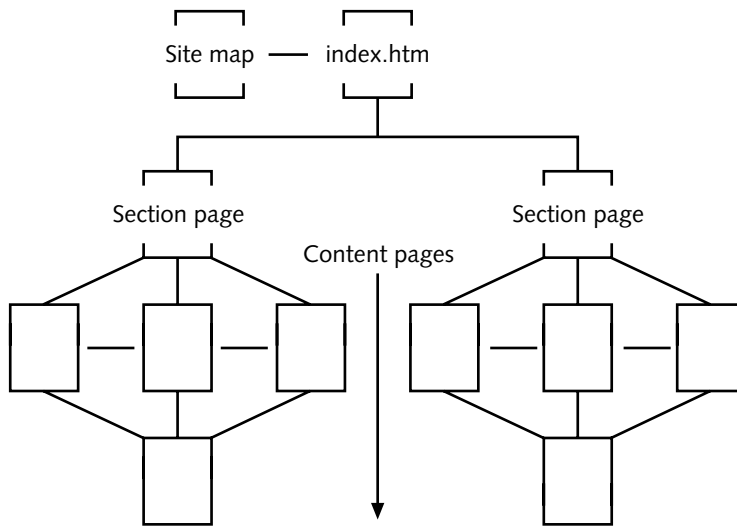


Figure 3-13 Cluster structure

Catalog Structure

The catalog structure illustrated in Figure 3-14 is ideally suited to electronic shopping. The user can browse or search for items and view specific information about each product on the item pages. Users can add items to their shopping cart as they shop. When they are finished, they can review the items in their shopping cart and then proceed to checkout, where they can enter credit card information and finalize the order.

3

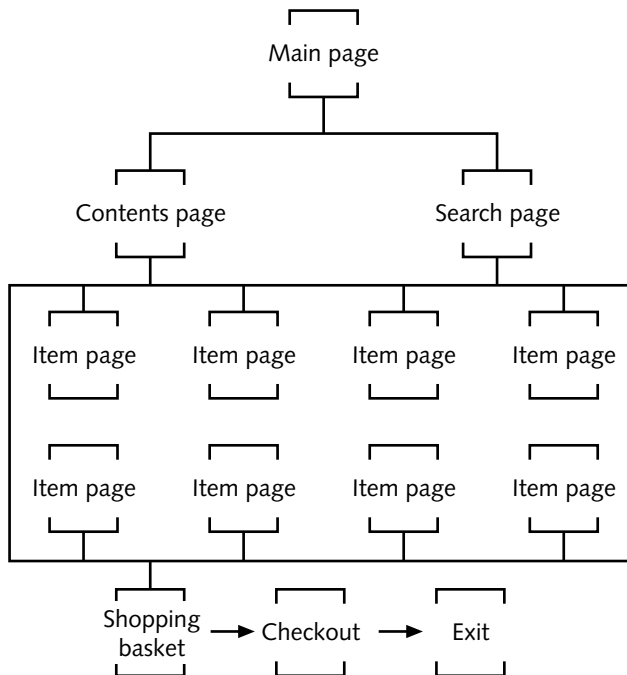


Figure 3-14 Catalog structure

This type of Web site requires back-end data transaction processing to handle the shopping cart tally, process credit card information, and generate an order for the warehouse. Businesses that want to set up an electronic commerce site can purchase ready-made commerce software packages or develop their own from scratch.

CHAPTER SUMMARY

A successful Web site is the result of careful planning. The steps you take before you actually start coding the site save you time, energy, and expenses in the long run. Remember these guidelines for successful planning:

- Start with pencil and paper; your ideas are less restricted and you can easily revise and recast without recoding.
- Write a site specification document. You will find it invaluable as a reference while building your site.
- Identify the content goal by adopting your users' perspective and learning what they expect from your site.
- Analyze your audience and create an audience profile. Focus your site on the users' needs and continue to meet those needs by adapting the site based on user feedback.
- An effective site is most commonly the result of a team effort. Leverage different skill sets and experience to build a Web site development team.
- Plan for successful implementation of your site by creating portable filename conventions. Build a relative file structure that can be transferred to your Web server without a hitch.
- Select a basic information structure for your site and then manually diagram it, customizing it to the needs of your site.

REVIEW QUESTIONS

1. List three technology constraints that can affect the way a user views your Web site's content.
2. Consult your Web server administrator when you need to determine the _____ and _____ for your site.
3. Name two inconsistencies that can cause broken links when you upload your files to a Web server.
4. List three characteristics of filenames that vary by operating system.
5. The international standard for filenames often is called _____.
6. Which computer operating system is case sensitive?
7. Rename the following files so that they are compatible across all operating systems:
My file.htm _____
case:1.htm _____
#3rdpage.htm _____

8. What is the default home page filename for a Web site?
9. What are the two types of URLs?
10. What are the four parts of a complete URL?
11. What type of URL links to another server?
12. What type of URL links within a server?
13. What affects the format of the URL for your Web site?
14. What is the benefit of purchasing a domain name?
15. What symbol indicates an absolute path?
16. Why should you never specify an absolute path in partial URLs?
17. What is the benefit of building a site with relative paths?
18. Files that reside in the same directory need only the _____ to refer to each other.
19. List two benefits of diagramming your site before you start coding.

HANDS-ON PROJECTS



1. Browse the Web and find a site you like. Write a brief statement of the Web site's goals.
2. Browse the Web and find Web sites that fit the following content types:
 - a. Billboard
 - b. Publishing
 - c. Special interest
 - d. Product supportWrite a short summary of how the content is presented at each Web site and describe how each site focuses on its users' needs.
3. Browse the Web and find a site that does not contain a user survey form. Write a 10- to 15-question user survey that you would use on the site. Tailor the questions to the site's content and goals.
4. Find a billboard-type Web site. Write an analysis of the site that includes functions and features you would add to extend the site's effectiveness for its users.
5. Visit www.winzip.com and download the latest evaluation version of WinZip. If you have a Mac, visit www.aladdinsys.com and download the shareware version of Stuffit. Practice using the software to archive and compress multiple files into a single file.

6. Browse the Web to find examples of the following site structures and describe how the content fits the structure. Think about how the chosen structure adds to or detracts from the effectiveness and ease of navigation of the site. Determine whether the site provides sufficient navigation information. Print examples from the site and indicate where the site structure and navigation information are available to the user.
 - a. Linear
 - b. Hierarchical
7. Browse the Web to find a site that uses more than one structure type and describe why you think the site's content benefits from multiple structures. Consider the same questions as in Hands-on Project 3-6.
8. Are there other structure types that are not described in this chapter? Find a site that illustrates a different structure content. Create a flowchart for the site and determine how it benefits from the different structure type.

CASE PROJECTS



CASE PROJECTS

Write a site specification for the site you defined in Chapters 1 and 2. Include as much information as possible from the project proposal you completed at the end of Chapter 1. Make sure to include a mission statement. Determine how you will measure the site's success in meeting its goals. Include a description of the intended audience. Describe how you will assess user satisfaction with the site. Include technological issues that may influence the site's development or function.

Prepare a detailed flowchart for your site using the preliminary flowchart you created at the end of Chapter 2. Create a filename for each page that matches the ISO 9660 standard. Indicate all links between pages. Write a short summary that describes the flowchart. Describe why you chose the particular structure, how it suits your content, and how it benefits the user.