

Chapter 18

Ten Ways to Drag a Site Out of the Stone Age

In This Chapter

- ▶ Deciding whether it makes sense to update an old site
 - ▶ Getting strict with code syntax
 - ▶ Replacing formatting with CSS
 - ▶ Validating and previewing
-

Time marches on, hemlines rise and fall, and pant legs get skinny, wide, and skinny again. Hang on to your clothes long enough and they're bound to come into style again. But what about your old Web sites?

In this chapter, we make some general suggestions and give you a few tips, in case you want to turn an old clunker into the lean, clean, CSS-driven pages that conform to today's Web design standards. Many of these suggestions come from professional Web designers who went through an overhaul or two and lived to tell about it.

Here's a word of caution before you rev up your Sawzall power tool: Just as remodeling a house requires a more-than-basic knowledge of building structure, wiring, and plumbing, remodeling a Web site requires a more-than-basic knowledge of HTML markup and CSS syntax. In other words, when you expose all the wires, you need to know the difference between the red ones and the black ones.

If you feel that you're not yet up to the task, read a book or two, follow a tutorial or two, and watch and learn as you build new sites with Expression Web. It takes time to acquire new skills, and even more time to find how to apply them.

Note: Although we present these tips in a somewhat logical sequence, every Web page is different, with different problems and levels of needed updates. Use the tips in this chapter as guidelines rather than as step-by-step procedures.

Decide Whether the Site is Worth Updating

This is probably the most crucial consideration, and one that shouldn't be taken lightly. Figure out what's wrong with the site. Does it work? Is the basic design good or a complete disaster? Which parts of the site are still useable? How long ago was the site written or updated? Does it employ any CSS styles, or is all the formatting crammed into the site's HTML? How big is the site?

You may decide that it's time for a completely new design, in which case it makes more sense to pillage your site's text and pictures and build a new design from the ground up. In fact, most of the designers we surveyed preferred to start fresh rather than clean up old code.

Back It Up and Keep It Safe

If you decide to overhaul a site, make a backup copy of the site in its "before" state and keep it in a safe place. This strategy is a generally good practice for *all* your Web sites, and it gives you a fallback position if you decide to bail out along the way and revert to the old site.

Add a Doctype

The doctype, or *document type declaration*, tells the browser which version of HTML to expect in the Web page so that it knows how to display it correctly. The HTML version in widest use today is XHTML Transitional, and it's the default version for new HTML pages created in Expression Web. If you want to make your page's code toe the line, you need to give it a standard to measure up to. Adding a doctype to a page that doesn't have one is a simple procedure when you're using the Expression Web Code Snippets feature. We talk about doctypes and Code Snippets in Chapter 14.

Run the Compatibility Checker

After you add a doctype to your Web page, you can run the Expression Web Compatibility Checker. It looks at the doctype declaration and then checks each line of code for compliance with that version of HTML. Then it tells you what's not up to snuff and gives you some information about why. It's a bit like taking a pretest for the LSAT (Law School Admission Test) to find out what you don't know before you shell out the big bucks for the real test. Depending on what the Compatibility Checker turns up, you may decide to forge ahead or ditch the overhaul. We cover using the Compatibility Checker in Chapter 12.

Fix the Worst Code Offenders

Unlike dress codes, which have become more relaxed with the passage of time, XHTML syntax is uptight and precise compared to its HTML predecessor. Whereas HTML didn't care about whether a heading tag was written as `<H1>` or `<h1>`, XHTML runs a tight ship and requires code to look just so. Converting your page's code is a tedious but necessary job if you want it to measure up. You can read up on how the folks at W3 Schools updated their page code at www.w3schools.com/xhtml/xhtml_howto.asp. Also check out HTML Tidy, a free utility for cleaning up your code, at <http://sourceforge.net/projects/tidy>.

Remove Formatting from HTML Tags

Yep. All those `` tags and their attributes have to go. (Depending on when your page was created, you may have a lot of them.) While you're at it, replace all the `` tags with `` and the `<i>` tags with ``. The Replace command may well become your new best friend.

Check Your Content

After cleaning out the code, take a look at your page's content. Does the page have good heading hierarchy and document structure? Does it follow a logical sequence of information? Are all the parts tagged with the proper HTML elements? Take a look at Chapter 14 for more information about structured content.

Add Alternative Text to All Pictures

The World Wide Web Consortium requires that all pictures (art, photos — anything that appears in your page courtesy of the `` tag) come with alternative text (or *alt text*) so that a visitor whose browser doesn't display pictures knows what's supposed to be displayed in that spot. Check your `` tags to make sure they contain the attribute `alt="text"`; if it's missing, add a meaningful word or blurb that adequately describes the picture. If the image is a button or heading, use the text that appears on the button or in the heading.

Replace Formatting with CSS Styles

Even if you choose to keep your page's structure in a table-based layout, using CSS to format your content gives a Web page a leg up on compatibility and compliance. For some sites, this may be where you stop. If you're feeling ambitious and your CSS skills are up to par, forge on and try your hand at using CSS for positioning, too. Chapter 7 covers the basics of creating, applying, and modifying CSS styles. Chapter 8 tackles using CSS for positioning elements on the page. Create an external style sheet and move your styles to it — we tell you how in Chapter 9.

Seek Validation

After all your hard work, measure your progress by running the Compatibility Checker again. Then check your page against the W3C code and CSS validators. You may have overlooked a few things, which these tools will most likely catch. (Find the HTML validator at <http://validator.w3.org> and the CSS validation tool at <http://jigsaw.w3.org/css-validator>.)

Don't forget to check your links, and, at the risk of sounding like a jukebox with only one song, preview, preview, and preview again, in as many browsers as you can get your mouse on! We show you how to preview your pages in Chapter 2.