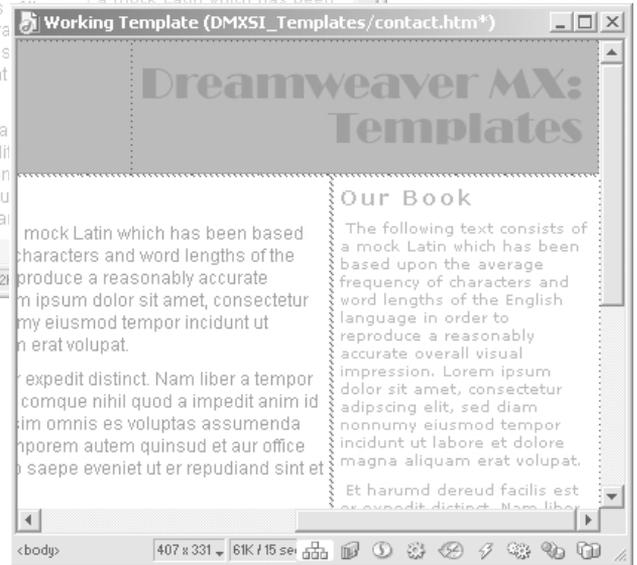




# Chapter 4

## Removing Dreamweaver Markup



Before and After.

*Dreamweaver Markup* refers to those comments Dreamweaver inserts into templates, template child pages, and library items that allow Dreamweaver to control the various

template regions discussed in Chapter 3, “All About Templates and Regions,” and library items discussed in Chapter 2, “Library Items Revealed.”

When you are building a template or using library items in Dreamweaver MX, HTML comments are embedded in the code for internal use by Dreamweaver, depending on the design-time methods you use. You may want to remove this markup before mounting the pages to the production server. In this chapter, we explore the techniques available and discuss their caveats.

## Why Would You Want to Remove DMX Markup?

As discussed previously in Chapters 2 and 3, the DW/DMX template and library item markup is 100% proprietary, meaning that only DW/DMX can make appropriate use of it. You may want to remove DMX markup for various reasons, which include but are not limited to these:

- Bandwidth considerations
- Troubleshooting purposes
- Unique modifications
- Ability to conceal templates and libraries

The sections that follow explore these issues in more detail.

### Bandwidth Considerations

Template and library markup can add a substantial amount of weight (bytes) to each document. By removing markup, you are reducing the page weight and therefore the bandwidth required for your site, which translates into faster load times for your individual pages and less expensive hosting costs if you’re charged by bandwidth usage. This reduction is directly proportional to the number of template regions and library items on each page. A page with multiple repeat regions and multiple template parameters (particularly in a nested template) can become quite “bulky” due to the presence of this markup for each region and parameter.

### Troubleshooting Purposes

Maybe you’re having problems with a specific behavior that works in the template but not in the child page. Detaching the child page from the template would allow you to work 100% with the file without worrying about adding editable regions or editable tag attributes. Perhaps the troubleshooting is going to require help from an outside source

such as a newsgroup member; in order for them to assist you efficiently, you should provide them with a copy of the child page and a detached child page so that they can work with the files in both forms to assist you. This admittedly is a very special case, and in most circumstances you would want to leave the markup intact to assist with locating the problem area more easily.

### Unique Modification

On occasion, you might have a single page that you want to make unique, but still based on the original template. Detaching the page from the template will allow you to customize that unique page. This particular reason isn't so compelling anymore, given DMX's enhanced template capabilities.

### Conceal the Use of Templates and Libraries

It's logical that you might build a site based on templates and library items for your own ease of maintenance. By detaching the child pages, you conceal the fact that you used a template to build the site.

## Why Would You Want to Leave DMX Markup?

On the other hand, there are valid reasons to leave DMX's markup on the page. These include, but are not limited to, the following reasons:

- Remote co-workers
- Bandwidth
- Troubleshooting purposes

The following sections take a closer look at these considerations.

### Remote Co-Workers

It's likely that you will eventually work collaboratively to build a site, if you aren't already. By leaving the markup intact on the page, it is easy for your remote or local co-worker(s) to make further modifications to the page without affecting the template or the updating functionality of DMX.

### Bandwidth

The target market for the current site is the primary determinant with regard to Bandwidth concerns. Web developers are fortunate in that high-speed Internet services

are becoming much more widespread and less expensive to use. Nevertheless, the population of dial-up users is not negligible in most demographics and is worthy of consideration. Depending on how extensively markup is used on the various pages of the site, you might be required to balance its removal against any productivity impact. Making markup removal the last operation prior to publishing the site may alleviate this concern.

**Note**

A modem making a connection at 56K has a theoretical maximum throughput of about 52,000 bits per second (due to telephone line restrictions). Some of this throughput is consumed with a “handshake” between the sending and receiving modems, although more of it is consumed by network congestion (lost and retransmitted packets). In practice, the typical 56K dialup connection can sustain a throughput of about 32,000 bits per second. With an 8-bit byte, this is equivalent to 4,000 bytes per second, or 4K/sec.

Most usability experts agree that a 10-second download time for a given page (or a page’s engaging content) is the desired target to prevent a site’s visitors from “clicking through,” or leaving the site. This means that the target weight for any particular page should be 40–45K to meet this standard.

It is worth mentioning that corporate users are usually less concerned about bandwidth because most are on high-speed LANs and/or Internet connections. Therefore, in a corporate environment the internal documentation provided by the markup would be more beneficial than the bandwidth saved.

## Troubleshooting Purposes

If you’re having problems with a child page, leaving the markup intact could help you determine whether the issue is with a marked up region, thereby pointing you to the problem area (such as a specific library item or template region). Using the markup as a code commenting technique is ill-advised and not a replacement for good code commenting strategies, a discussion of which is beyond the scope of this book.

## Methods of Removing Markup

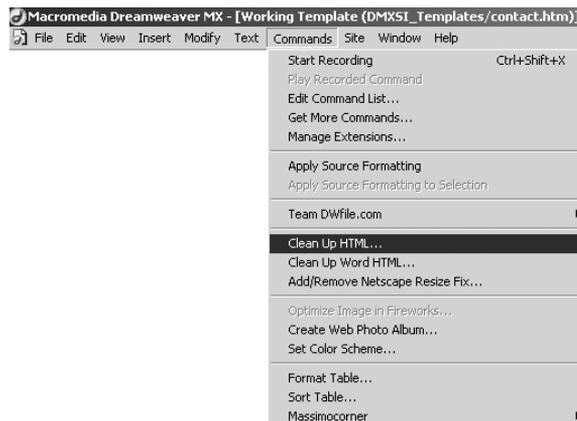
As with many functions in DMX, there are different ways to remove markup. However, unlike other processes, each method of removing template markup is unique in what it actually does with the page and/or site.

**Warning**

Exercise caution using any of these methods. It is all too easy to inadvertently overwrite good pages with bad ones and to lose the template/library item update capability in a site. Unless you are careful with the options, an unintended export operation could overwrite finished pages! So choose your options carefully, and always perform a backup of your site prior to using these markup removal techniques.

## Commands > Clean Up HTML

The first method of markup removal you need to learn is the Clean Up HTML command. To access this menu item (disabled if no document is open), select Commands > Clean Up HTML menu item to activate the Clean Up HTML dialog box (see Figure 4.1). Exercise 4.1 works through how to use this menu item and illustrates what it does on a templates child page.



**Figure 4.1** Clean Up HTML Menu.

You could use this method if you wanted a level of control to delete all template markup and do a code check at the same time, all while working on a single page that is based on a template. This method is usable only with the currently open and active document.

You cannot use this method if you desire your whole site to be affected. If you want to export your whole site removing markup, then please read the section entitled “Export without Markup,” later in this chapter.

Be careful of the following events that might occur when using this method:

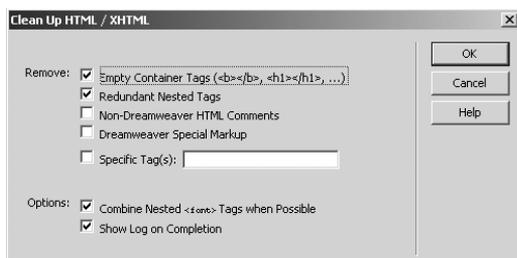
- This option works with an open and active document only.

- Depending on your Clean Up HTML dialog box settings, you could potentially disable functionality of the page.
- If you overwrite your template child page, it is difficult to get it back.
- The affected open and active document is not saved automatically, so no changes are written to the site until you save the file. (This is actually also a benefit because it allows you a level of error checking.)

#### Exercise 4.1: Using Cleanup HTML to Remove Dreamweaver Markup

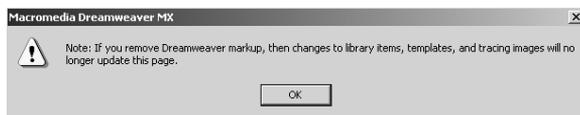
In this exercise, you'll see how to use the Clean Up HTML command by examining a practical example.

1. Make sure that you have the required site defined in DMX and that Dreamweaver is running—if not, then please see the instructions provided in the section entitled “Defining the Site” in the Section I, “Reusable Page Elements,” or start Dreamweaver if your site is already defined.
2. Open the document named **contact.htm**, and then select the Commands > Clean Up HTML menu item. The Clean Up HTML dialog box should display (see Figure 4.2).



**Figure 4.2** The Clean Up HTML Dialog box.

3. Place a checkmark in the checkbox beside Dreamweaver Special Markup. When you do this, a warning dialog box is displayed, as you can see in Figure 4.3.



**Figure 4.3** Dreamweaver Special Markup Warning.

4. Click OK to close the warning box, and then click the OK button when you are satisfied with your selections. Dreamweaver will do its clean-up routine and clean the items you chose in step 3. When DMX is finished, it displays a

Summary Report telling you how many tags it removed from the active document (see Figure 4.4).

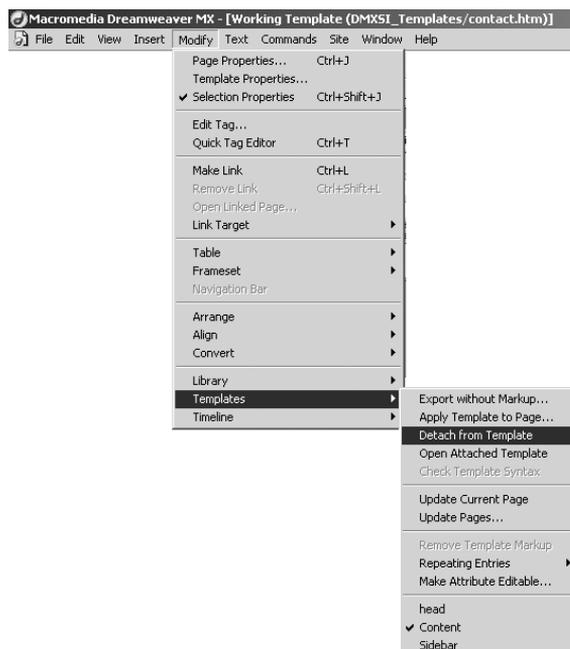


**Figure 4.4** Clean Up Summary Report.

5. Save the changed file as **contact\_clean\_a.htm**, close the document, and leave DMX open for the next exercise. The document is now detached from templates and library items completely and will not be updated further by modifications to the templates or library items.

## Modify > Templates > Detach from Template

To access this menu item (it is disabled if no document is open or if the open document is not attached to a template), select Modify > Templates > Detach From Template to activate the detachment routine (see Figure 4.5). Exercise 4.2 works through how to use this menu item and what it does on a template's child page.



**Figure 4.5** Detach From Template Menu.

Similar to the Clean Up HTML command, this method also works for only the currently open and active document. There are no dialogs to complete, no warning prompts to read and acknowledge—it's instantaneous and unforgiving. Unlike the Clean Up HTML method, library items are left untouched by this command.

You cannot use this method if you desire your whole site to be affected. This method removes only template related markup—it will not remove library item markup.

Note the following warnings (as stated earlier) when using this method:

- This method works with only the active open document.
- You potentially could disable functionality of the page.
- If you overwrite your template child page, it is difficult to get it back, as you will see in Chapter 5, “Miscellaneous Template-Related Functions.”
- The affected open and active document is not automatically saved, so no changes are written to the site until you save the file. (This actually is also a benefit because it allows you a level of error checking and control.)

#### **Exercise 4.2: Detaching a Child Page from Its Template**

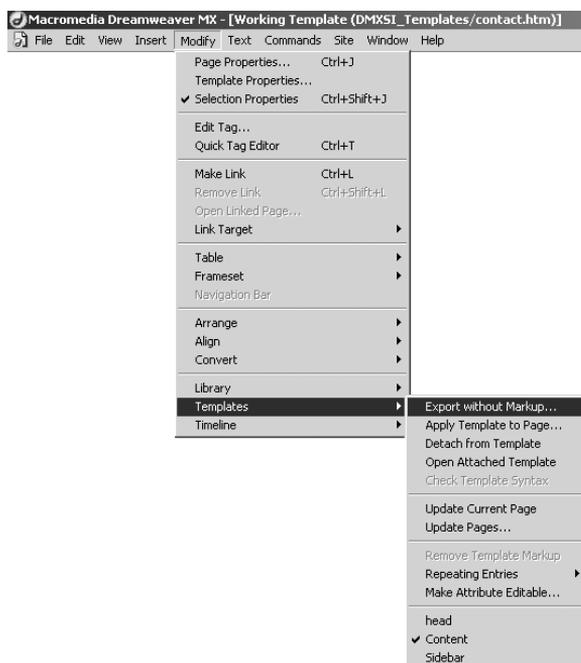
In this exercise, we will illustrate the methods we feel are the best to use for detaching pages from templates

1. It's assumed that you have the required site defined in DMX and that Dreamweaver is running; if not, then please see the instructions provided in the section entitled “Defining the Site” in the Section I, or start Dreamweaver if your site is already defined.
2. Open the document named `contact.htm`. Select the Modify > Templates > Detach From Template menu item. DMX goes through its detach routine and provides no notification that it is finished other than the document becomes active again—although you can see that it has been detached because the editable regions disappear.
3. Save the changed file as **`contact_clean_b.htm`**, close the document, and leave DMX open for the next exercise. The document is now detached from the template completely and will not be updated further by modifications to the template. However, library items remain attached to the document and will be affected by library item updates.

#### **Modify > Templates > Export Without Markup...**

There might be times when you would want to remove all template markup from an entire site. This section will explain how to do this.

To access this menu item (disabled if no document is open and active), select **Modify > Templates > Export Without Markup** to activate the Export Without Markup dialog box (see Figure 4.6). Exercise 4.3 works through how to use this menu item and what it does on a Templates child page.



**Figure 4.6** Export without Markup Menu.

You could use this method if you wanted a level of control to delete all template markup on the entire site, or if you changed files only since last export and you don't want to affect your local site files. This method is used for exporting changed files (since last export) or all files (overwriting what was previously exported).

Do not use this method if you want to detach only a single page from the template. Do not use this method if you want library item information to be removed, because this method removes only template markup and not library markup.

Keep the following things in mind when using this method:

- This method is a site-wide method, so do not use it if you only want to detach a single page from the template or a select group of files.
- Markup other than that used by the template function of DMX is not removed by this method.

- If you previously exported your site and are happy with several files, using this method could overwrite those files if you are not extremely careful setting the Dialog options.
- A new Site Definition must be created to use DMX's built-in FTP Client to send the files to the server.

### Exercise 4.3: Exporting the Site without Dreamweaver Markup

Here's where we discover how to use this site-wide operation. In this exercise, we will put all that we have learned about Export Without Markup into practical use through a generalized example.

1. Begin with the site defined and Dreamweaver open.
2. Open the document named contact.htm. Select the Modify > Templates > Export Without Markup menu item. The Export Site Without Template Markup dialog box is displayed (see Figure 4.7).



**Figure 4.7** Export Without Markup dialog box.

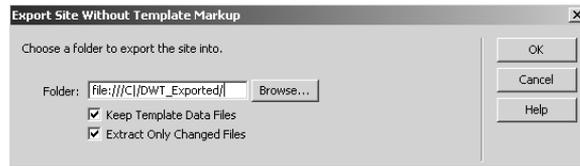
In this dialog box, you'll see the following options:

- **Folder.** This entry must be a folder completely outside the current Dreamweaver Site Definition; otherwise, an error message will be displayed. The folder selection information is retained by the site once it is defined.
  - **Keep Template Data Files.** This option creates XML template content files and saves them in the export folder in which the source file is located. These XML files can be used to import template data into a template-based page and can be used as a boilerplate for creating other template child pages, as you will learn in Chapter 6, "Building the Hot Cool Toys Base Template." This information is retained by the site.
  - **Extract Only Changed Files.** This option extracts only those files that have changed since the last extraction. If an extraction has not occurred, this option extracts all child pages. This information also is retained by the site.
3. In the dialog box, fill in the fields as defined (see Figure 4.8):

**Folder:** C:\DWT\_Exported

**Keep Template Data Files:** Enabled

**Extract Only Changed Files:** Enabled



**Figure 4.8** Export Without Markup dialog (Completed).

4. Depress the OK button, and Dreamweaver performs its magic magic by exporting the pages with all template markup removed.



#### Note

You would need to configure a second Site Definition to point to the exported site folder C:\DWT\_Exported if you wanted to put these files to the server using Dreamweaver.

5. Close the document and leave DMX open for the next exercise. The site has now been detached from templates, possibly updated. The exported site will not be updated further by modifications to the templates or library items; however, the original site that you exported from will continue to be updated by changes to the template and/or library items.

## Summary

In this chapter, you have learned various ways to detach a file or group of files from the template that created them. There are times when it's necessary or desirable to remove DMX's template and/or library markup. DMX offers you various solutions for your use, each method being unique in and of itself. Sometimes you decide that the best solution is to leave the markup on the pages. Should you desire a single page cleaned or a whole site, the choice is yours. ■