- Introduction to Dreamweaver extensions
- Using the Server Behavior Builder (SBB)
- Distributing and installing your Extensions
- Extension references and recommendations
Extending Dreamweaver MX

One of the great things about Dreamweaver MX is its extensibility. The base program is an extremely powerful tool, but being able to add our own extra functionality to it by building extensions is what really gives us the ability to make the program sing as well as dance. The idea of building extensions is to speed up and simplify the process of writing code. Dreamweaver MX comes with lots of prebuilt extensions but they might not fit your particular situation, which is where extension building comes in.

There are many developers out there already producing some quite amazing extensions. There has been a growing market for commercial quality extensions for some time now and this can only be a good thing in terms of pushing back the boundaries of what is possible. Just a selection of the more common types of extensions that you can create to add functionality to Dreamweaver MX are:

- Behaviors
- Objects
- Commands
- Server Behaviors
- Panels

In this chapter we will build two simple Server Behaviors. The first will make it easy for us to maintain the paragraph formatting that our data contains when retrieved from a database. It will also include some simple validation that will prevent the code from throwing an error on the web page if there isn't any relevant data in the database. The second will conditionally show or hide an area depending upon there being database content to show in that area. If there isn't any data to show, we'll hide the area.

The order in which we go about building an extension is not set in stone, but there is an obvious progression that, if followed, will make your extension building life far easier than if you try to go about it another way. First we will write the final version of the code so we can test it to make sure it works exactly as intended. Then we will launch the Server Behavior Builder (SBB) and begin adding the code into there, along with the required parameters so that the SBB can build the extension interface and write the code necessary to run the Server Behavior interface and insert the correct code into our document.