In this chapter, we'll see

- using partial templates
- rendering into the page layout
- updating pages dynamically with AJAX and rjs
- highlighting changes with Script.aculo.us
- hiding and revealing DOM elements
- working when JavaScript is disabled

Chapter 9

Task D: Add a Dash of AJAX

Our customer wants us to add AJAX support to the store. But just what *is* AJAX?

In the old days (up until a year or two ago), browsers were treated as really dumb devices. When you wrote a browser-based application, you'd send stuff down to the browser and then forget about that session. At some point, the user would fill in some form fields or click a hyperlink, and your application would get woken up by an incoming request. It would render a complete page back to the user, and the whole tedious process would start afresh. That's exactly how our Depot application behaves so far.

But it turns out that browsers aren't really that dumb (who knew?). They can run code. Almost all browsers can run JavaScript (and the vast majority also support Adobe's Flash). And it turns out that the JavaScript in the browser can interact behind the scenes with the application on the server, updating the stuff the user sees as a result. Jesse James Garrett named this style of interaction *AJAX* (which once stood for *Asynchronous JavaScript and XML* but now just means *Making Browsers Suck Less*).

So, let's AJAXify our shopping cart. Rather than having a separate shopping cart page, let's put the current cart display into the catalog's sidebar. Then, we'll add the AJAX magic that updates the cart in the sidebar without redisplaying the whole page.

Whenever you work with AJAX, it's good to start with the non-AJAX version of the application and then gradually introduce AJAX features. That's what we'll do here. For starters, let's move the cart from its own page and put it in the sidebar.