blink

By [Heidi Adkisson](#)

This past June, I attended the Usability Professional Association conference—the theme for which was Patterns: Blueprints for Usability. It provided the opportunity to hear a number of different perspectives on user interface patterns—and I presented my own thoughts on the topic as a conference presenter.

My relationship with user interface patterns goes back to the year 2000, when I was developing the topic for my master's thesis. The research I did—examining 75 leading e-commerce sites for design similarities—was closely related to the idea of patterns. However, for a number of reasons, I did not structure the research as patterns. The main reason was that my research focus was identifying the range of solutions to a given design problem. Patterns seemed more suited to documenting (in great detail) idealized or archetypical solutions.

The promise of patterns is alluring—the idea that you can solve a particular design problem by looking up the solution in a pattern library. In theory, designers use a pattern library to document user interface solutions, which are subsequently re-used by other designers. The very detailed, structured pattern description provides complete information about what the pattern is, when to use it, and why it works. There are a number of pattern libraries available on the web including welie.com, the Yahoo pattern library, and Designing Interfaces.

Despite the promise of patterns, as a working designer I have found the practical usefulness of patterns limited. One problem stems from the "look up" model of use that patterns assume. If I am grappling with a design problem and want to look up the solution in a pattern library, I need to know how to find what I'm looking for. For simple, well-known problems I can probably take a good guess at the name of the pattern and look it up that way. But of course if it's a simple, well-known problem I'm probably not grappling with it in the first place. For complex or unique problems—the types of problems experienced designers do grapple with—it can be very difficult to pin a label on the problem that you are trying to solve. In my experience, time is usually

better spent working to solve the problem than navigating through a pattern library in search of a ready-made solution.

However, I don't discount the idea of documenting and analyzing user interface solutions. In fact, I advocate quite the opposite. The research I did for my master's thesis convinced me of the value of spending time understanding others' design solutions. The difference is I advocate documenting design solutions primarily for your own use—not as a resource for others to draw on. What I recommend is creating and maintaining a personal design library.

The personal design library contains design examples that are particularly useful to you. What you may find useful could vary based on the type of design work you do and your level of experience (junior vs. more senior). For example, I wouldn't collect an example of global navigation with "flyout" submenus, because this is a solution I am well-acquainted with. However, I did collect the example below, which shows an interesting variation on this type of navigation. (The flyout displays a few top options with a link to the entire set of options—particularly handy when the amount of sub-options is large.)

I don't describe the solution using the structure of a pattern, which, with its prescriptive categories, can be intimidating (and time-intensive) to complete. Instead, I use a more conversational style, explaining why I captured the example and my analysis of it (both good and bad). Bad examples can be particularly illuminating. For example, if something in a system trips me up, I'll stop, take a capture of it, and then think through why the design was a problem for me. (I have also collected examples from usability studies I have conducted.) The key is to keep the documentation for each example brief and concise. This makes it easier to maintain your design library and maximizes the chances you'll actually keep it up to date.

The Blink Design Library is an example of a personal design library, which is publicly accessible. We chose to use make it public so others could see how ours is set up to start their own. To create ours we used blogging software that supports tagging (categorization) of posts. A flat vs. hierarchical categorization scheme is another way to simplify the process of documenting examples.

In short, developing a personal design library can have a very high return on the time invested: you will improve your design skills by systematically examining solutions—both good and bad. While user interface patterns all share a worthy goal—to document solutions in a way they can be re-used by others—practicing designers will likely find a personal design library a more effective avenue to building design skills.