

Spring, 2006



01001100 01010100 01000101 01010010
LTER DataBits
 Information Management Newsletter of
 The Long Term Ecological Research Network
 01001100 01010100 01000101 01010010

Designing Interfaces

- *Shaun Haber (PAL/CCE)*

J. Tidwell. Designing Interfaces. O'Reilly Media, 2005

We have all used plain, sterile, and sometimes ugly looking web applications before. These web apps may work well, but where's the fun in using them? Sometimes it's worth taking that extra step to give your application a "sexy" look. By doing so, the user receives an overall greater experience. A look at sites like [Flickr](#) and [Gmail](#) shows that not only are these great services, they also present visually pleasing and intuitive user interfaces.

In *Designing Interfaces*, author Jennifer Tidwell explores the many aspects of user interface design. This book is analogous to *Design Patterns* (see my other Good Read) in that it documents various interface design patterns, culminating in a wealthy repository of visual ideas for the interface designer. Additionally, Yahoo! has recently created its own online library of visual design patterns (<http://developer.yahoo.com/ypatterns/index.php>). These resources are extremely helpful for improving the front-end of any web application by promoting ideas for a more intuitive and aesthetic user interface.

Design Patterns: Elements of Reusable Object-Oriented Software

- *Shaun Haber (PAL/CCE)*

E. Gamma, R. Helm, R. Johnson, J. Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1995.

Let's say we want to build a new web application for querying and analyzing ecological data. A service like this sounds pretty great, so where should we start? Assuming we already have a working infrastructure (web server, database, etc.), it's time to get dirty in the code. However, we should first answer a couple more questions: How do we best structure our code so to maximize code reuse and minimize duplicated efforts? What best practices and frameworks exist for promoting optimal code design?

Design Patterns is the first book to document object-oriented design patterns, providing programmers with a rich repository of reusable frameworks and preventing them from having to rediscover these

solutions themselves. The four authors worked together to discover, analyze, and provide working examples of 23 previously undocumented design patterns. Essentially, they captured these patterns as abstract ideas, making it easier for developers to design, document, and share their code. (For the programmers, these patterns include the Flyweights, Factory Method, and Singleton class).

Although this book was written before the explosion of the web (it is geared for general software development), everything in it is applicable for building a web service. Be forewarned though: This book is fairly technical, and requires a background in computer science and/or object-oriented programming. That being said, with this book in our library, we can comfortably build any new software with efficiently designed code.