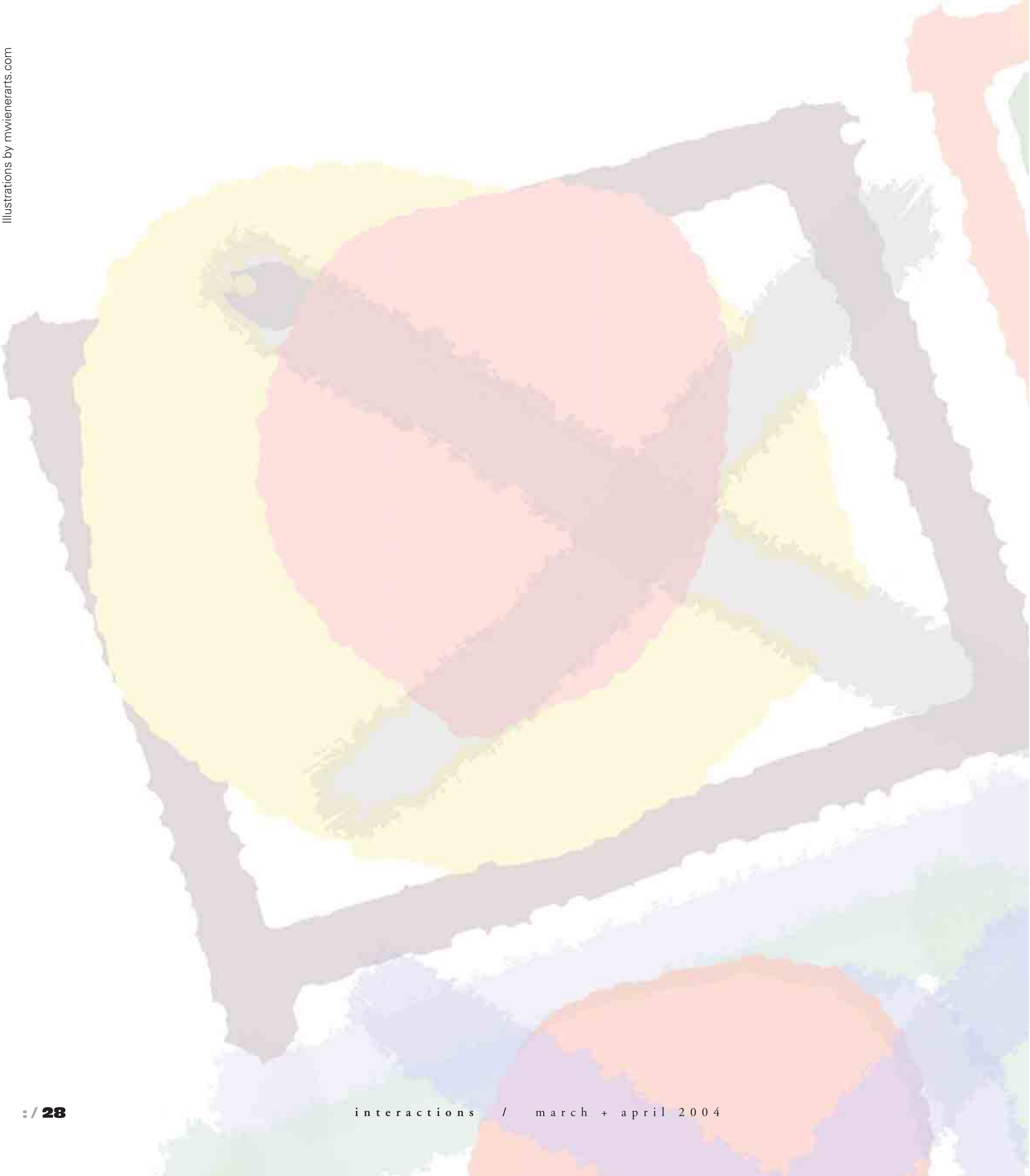
Illustrations by mwienerarts.com



DESIGN PATTERNS ENABLE PEOPLE TO ENVISION THE FUTURE: NOT ONLY DEVELOPERS, BUT ALL STAKEHOLDERS.

Patterns within Patterns

By Aaron Marcus
President
Aaron Marcus and
Associates, Inc. (AM+A)
www.AmandA.com

Introduction to Design Patterns

In past essays of "Fast Forward," I have written about the computer-human interaction (CHI) community being a gathering of tribes from many disciplines and incorporating wisdom from diverse fields. One example of cross-disciplinary fertilization is the adaptation of architectural design patterns to user-interface (UI) design patterns. As a former faculty member in a university's school of architecture and urban planning during the late 1960s and early 1970s, I was aware of Christopher Alexander's [1, 2] pioneering work at the University of California/Berkeley in establishing underlying paradigms for built form that are more general than specific solutions or standards but more specific than principles and guidelines.

There were other practitioners of the time, like the Princeton University's Urban Research Center, exploring "user-centered design" (we called it "participatory design" even then), and I had wondered in the first decade of SIGCHI whether CHI community professionals would rediscover architectural and urban planning concepts, terms, and principles that could be adapted to UI development. They did. In fact, the process is currently under way, thriving, and gaining adherents...and Alexander's writings about design patterns have gained new currency in the CHI community.

Design patterns are formalized descriptions of proven concepts that express nontrivial solutions to some design challenge. Design patterns consist of a set of contexts, common challenges (called problems), descriptions, enumeration of forces on the general resolution of the forces, and their impact on the final solution. In the original theory, architectural design patterns were relevant at all scales of built form, from parts of rooms to buildings, neighborhoods, and entire urban areas. In UI theory, design patterns are applicable for all user types, applications, platforms, content, and markets. The objective of using design patterns is to increase the quality of well-designed UIs with improved usability, usefulness, and appeal.

As one group of theorists put it, a primary goal of patterns is to create an inventory of solutions to help UI designers resolve challenges that are recurring and difficult [9]. Another theorist commented that patterns provide good solutions to common design challenges within specified contexts by describing invariant qualities of all those solutions [11]. An important idea is that design patterns capture the essence of a general solution in a way that enables other people to design, evaluate, and build it more easily and successfully.

So, what are the benefits of using UI design patterns? Well, for one thing, they are supposed to supplement existing documentation by providing background reasoning for the solution, which guidelines and standards may not. Of special importance is the ability of design patterns to serve as a means for exchanging ideas among different disciplines and stakeholders, making it possible for engineers, designers, evaluators, marketers, funders, and, yes, even users, to communicate more effectively with each other. Design patterns capture knowledge, promote reuse, and require only hours to apply, even though they may have taken many years to create. As such, they are valuable intellectual property.

Pattern Parts

To understand patterns better, consider the way their basic parts are described:

- *Name:* The name or title of a pattern expresses its benefit (like "Non-Stop Shopping" for a commercial Web site). The name should be easy to remember and use. Keep in mind that others, including you, will be looking through large collections of patterns searching for something that may have an unusual moniker and you might have difficulty finding the right one if the name is obscure or too witty.

- *Context:* The context identifies the user's specific goal or primary goals. The context must be focused, clear, precise, and easy to comprehend.
- *Problem:* This part of the description captures the objectives or intent of stakeholders, especially the user. The challenges facing the developers as well as the users can be included. This part describes the designer's goals.
- *Forces:* Any context is made from a set of forces. This part describes those relevant concerns and how they inter-relate. A design pattern resolves conflicts among these forces to reduce stress and to provide a general configuration that can be recognized, used, and reused. Poorly designed patterns fail to contain these forces, and tension, or unresolved forces, tends to spill into neighboring patterns within a complex field or set of hierarchically related patterns.
- *Solution:* This part describes how to realize the desired outcomes.

If this terminology sounds a little confusing, or mystical, well, it is. Alexander describes qualities of form that seem quasi-religious or cultish and are perhaps culture-biased. As a result, some professionals criticize the theory. However, his goals are to create forms that make people feel more active, happy, and satisfied. To that extent, it sounds somewhat like achieving usability, usefulness, and appeal.

An Example

To give a concrete example of design patterns from a Web UI context, consider how to move from one page to another:

- *Name:* Paging controls
- *Context:* A list may contain too many items to fit on one page. An example: the results of a search.
- *Problem:* Users need a way to browse through a long list of items.
- *Forces:* The number of items that can be returned may be limited by system performance. Users need to directly access positions within the list.
- *Solution:* Group items into pages. Provide paging controls above and below the list. Rationale: Dividing a list into shorter, manageable pages makes it easier to view and navigate the constraints. Users are given ways to navigate the list easily.

Notice that the design pattern does not provide detailed descriptions or advice about a specific solution with a particular browser, query application, and specific contents. Rather, the design pattern seeks to allow a reader to understand essential components and relationships.

Pattern Patter

Design patterns are associated with events and spaces. A typical architectural pattern was the seat of a bay window, which was comfortable for sitting and reading, or watching the world go by. What repeats in the patterns are not specific elements, but relationships—among the seat, the window, the light, the size of all elements, the location within a room, and so on.

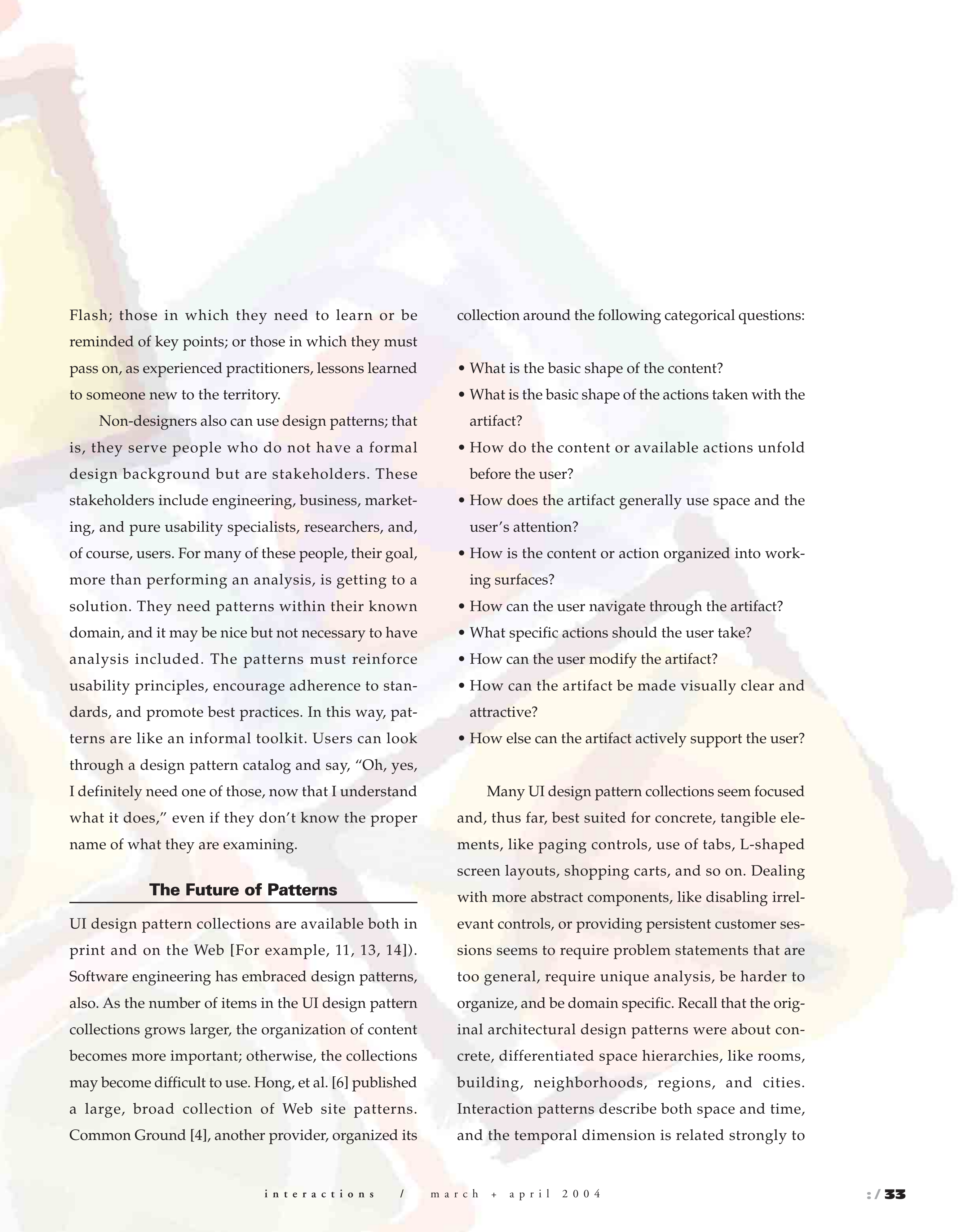
Design patterns are different from design standards and from traditional UI documentation. They are solutions to recurring challenges for a specific context. They do contain information about users, context, and tasks and therefore can serve as guides for usability, usefulness, and appeal. They also point to other patterns in a hierarchical network of patterns. They intended to support, not replace, other forms of documentation.

To design collections of design patterns is itself a challenge. There should not be too many, making it difficult to find the right kind. They should be organized well, and with as few conflicts in their approach, nomenclature, and descriptions, as possible. They should not point to a specific solution; that is *not* their purpose. They do require some experience to apply well, but with rapid development of technology and markets, they enable new techniques and constructs to be covered that are not adequately documented in the "classic" texts. In general, design patterns are discovered, not invented, not created by fiat or committees. They result from commonality in many examples of actual use that work well or feel right or are judged successful not only by professional juries but by users in the marketplace.

In a large, well-organized collection, design patterns can work together as a kind of language, with a rich set of signs and a grammar for combining them.

One special quality of design patterns is that they have the capacity—in a large, well-organized collection—to work together as a kind of language, that is, a rich set of signs and a grammar for combining them. A subset can even define a style, like southern Greek island houses versus northern California Victorian houses versus northern North American igloos. Variations will exist, even within a subset. Yet, these design patterns, related patterns, subsets, and supersets provide a rich set of ways in which human experience can be enabled. In our profession, that means human communication and interaction is enabled.

Design patterns have value for both designers and non-designers. Design patterns serve people who have a basic understanding of design principles and, in a professional sense, are "responsible for" or "own" the design. Professional designers may need to find references when designing for an unfamiliar context outside their immediate domains. Challenging contexts include those in which they may lack experience in particular topics, like searching; those with the constraints of particular platforms or medium, like handheld devices or Macromedia



Flash; those in which they need to learn or be reminded of key points; or those in which they must pass on, as experienced practitioners, lessons learned to someone new to the territory.

Non-designers also can use design patterns; that is, they serve people who do not have a formal design background but are stakeholders. These stakeholders include engineering, business, marketing, and pure usability specialists, researchers, and, of course, users. For many of these people, their goal, more than performing an analysis, is getting to a solution. They need patterns within their known domain, and it may be nice but not necessary to have analysis included. The patterns must reinforce usability principles, encourage adherence to standards, and promote best practices. In this way, patterns are like an informal toolkit. Users can look through a design pattern catalog and say, “Oh, yes, I definitely need one of those, now that I understand what it does,” even if they don’t know the proper name of what they are examining.

The Future of Patterns

UI design pattern collections are available both in print and on the Web [For example, 11, 13, 14]). Software engineering has embraced design patterns, also. As the number of items in the UI design pattern collections grows larger, the organization of content becomes more important; otherwise, the collections may become difficult to use. Hong, et al. [6] published a large, broad collection of Web site patterns. Common Ground [4], another provider, organized its collection around the following categorical questions:

- What is the basic shape of the content?
- What is the basic shape of the actions taken with the artifact?
- How do the content or available actions unfold before the user?
- How does the artifact generally use space and the user’s attention?
- How is the content or action organized into working surfaces?
- How can the user navigate through the artifact?
- What specific actions should the user take?
- How can the user modify the artifact?
- How can the artifact be made visually clear and attractive?
- How else can the artifact actively support the user?

Many UI design pattern collections seem focused and, thus far, best suited for concrete, tangible elements, like paging controls, use of tabs, L-shaped screen layouts, shopping carts, and so on. Dealing with more abstract components, like disabling irrelevant controls, or providing persistent customer sessions seems to require problem statements that are too general, require unique analysis, be harder to organize, and be domain specific. Recall that the original architectural design patterns were about concrete, differentiated space hierarchies, like rooms, building, neighborhoods, regions, and cities. Interaction patterns describe both space and time, and the temporal dimension is related strongly to principles of human behavior. The subjects of cognitive psychology, persuasion, customer service, and so on may inevitably involve principles that are not so neatly hierarchical and thus may be harder to describe in the conventions of patterns. Nevertheless, attempts are being made to broaden patterns to all aspects of what I have defined in the past as primary UI components: metaphors, mental models, navigation, interaction, and appearance.

The nomenclature of patterns may become more complex in the process and may involve some of the typical elements of large bodies of content: version and status; authors or editors; summaries; introductions for readers; glossaries of terms; and varieties of pattern names, including technical or scientific, popular, and alternate professional.

As UI design pattern catalogs become ubiquitous and enlarge to become giant compendia, in order to remain useful across disciplines—one of their key strengths—they must remain domain specific and, in the users' language, not overly formal. Remaining so may enable UI design pattern collections to become a *lingua franca* by which CHI professionals from many tribes can communicate, enabling all interested parties to smoke a peace pipe and achieve harmonious, desirable solutions that benefit users while achieving other stakeholder objectives.

REFERENCES

1. Alexander, C., et al. *A Pattern Language; Towns, Buildings, Construction*. Oxford University Press, New York, 1977.
2. Alexander, C. *A Timeless Way of Building*. Oxford University Press, New York, 1977. ISBN 0195024028.
3. Borchers, Jan. *A Pattern Approach to Interactive Design*. Wiley, New York, 2001. ISBN 0-471-49828-9.
4. Common Ground UI patterns, 1999. Available at www.mit.edu/~jtidwell/common_ground.html
5. Gang of Four Software Design Patterns: Gamma, Johnson, Helm, Vlissades, 1995, http://hillside.net/patterns/DPBook/GOF.html
6. Hong, Jason I., Landay, James, and Van Duyne, Douglas. *The Design of Sites: Patterns, Principles, and Processes for Crafting a Customer-Centered Web Experience*. Addison-Wesley, Reading, 2002. ISBN 0-201-72149X.
7. Lafreniere, www.gespro.com/lafrenid/patterns.html
8. Lafreniere, D., and Granlund, A. Describing and Using Patterns for UI Design. Usability Professionals Association (UPA) annual conference, UPA Extended Proceedings.
9. Loureiro, K., and Plummer, D. (1999). *AD Patterns: Beyond Objects and Components*. Research Note # COM-080111, Gartner Group.
10. Patterns and software, www.enteract.com/~bradapp/docs/patterns-ointro.html
11. Tidwell, Jenifer. Common Ground: A Pattern for Human-Computer Interface. www.mit.edu/~jtidwell/interaction_patterns.html
12. UI design process patterns, http://c2.com/ppr/ui.html
13. UI interaction patterns, www.pliant.org/personal/Tom_Erickson/InteractionPatterns.html
14. Welie Web design patterns (Amsterdam collection), www.welie.com