[Tom's Home Page]
[Professional] [Life, Fun, &c] [Tell Me...]
[Bookmarks] [Publications List] <and many papers and essays>

Supporting Interdisciplinary Design: Towards Pattern Languages for Workplaces

Thomas Erickson
snowfall@acm.org
Apple Research Labs
(Now at: IBM T.J. Watson Research Center)

I am concerned with design, particularly with the issue of how to design systems that mesh gracefully with the practices and activities of particular workplaces. This concern gives rise to a number of questions: How do designers come to understand a new workplace? How do they get a sense of the sorts of activities that occur within it, and with which their design must coexist? How do designers avoid or minimize disruptions caused by the inevitable changes a new system will introduce? How do they figure out how to design things that are useful, and not just usable?

Workplace research is a vital part of any answer to these questions. However, from my vantage point as a practitioner of design, it seems very unlikely that a thorough research phase will become a standard part of design practice. Systems design and implementation typically takes place under considerable limitations of time and resources; it seems unlikely that this will change. To me, this suggests a clear conclusion: we need ways of allowing the results of workplace studies to be reused in new and different situations.

But in what form should the results of workplace studies be presented? This is a difficult problem because most designers are not versed in the disciplines and assumptions that underlie workplace research; workplace studies are not accessible to those who need them the most. The problem is compounded by the fact that those involved in systems design lack *any* core discipline. Systems are being designed by computer scientists, anthropologists, psychologists, visual designers, and industrial designers; furthermore, the advent of new technological domains such as multimedia and virtual spaces are creating roles for architects, interior designers, musicians, and film makers. Even end users (who are actually no more homogenous in their backgrounds than 'designers') are involved in design.

I suggest that we are faced with a problem of representation. We need ways of representing knowledge about the workplace so that it is accessible to the increasingly diverse set of people involved in design. Ideally, we want not only to represent workplace knowledge, but to provide a framework within which it can be discussed, explicated, extended, and generalized. In the absence of a shared discipline or conceptual framework, I believe that this means that knowledge must be embodied in a concrete, recognizable form, in the terms of the design's target domain: the workplace. This brings us to pattern languages, which, I suggest, deserve serious consideration as a representational mechanism for workplace design.

Pattern Languages

The concept of a pattern language has been developed by Christopher Alexander and his colleagues in architecture and urban design [1, 2]. In brief, a pattern language is a network of patterns of varying scales; each pattern is embodied as a concrete prototype, and is related to larger scale patterns which it supports, and to smaller scale patterns which support it. The goal of a pattern language is to capture patterns in their contexts, and to provide a mechanism for understanding the non-local consequences of design decisions.

Alexander's pattern language is more than an engaging theory. It has been in use for two decades, and has been applied to the design of everything from rooms to communities (see [5] for references to a number of published accounts). Alexander's approach has a relatively small following among architects, many of whom are hostile to his methods or alienated by his rhetoric. Most of its use appears to be by those outside the mainstream of the architecture profession - designer-builders, and people with no design training who simply wish to expand or build their own houses, either by themselves or in collaboration with pattern-language-friendly architects [12]. A piece of corroborating evidence is that Alexander's book of design patterns, *A Pattern Language* [2], continues to be a best selling architecture book after two decades on the market, even though it is available only as a rather expensive hard back.

Over the last decade the pattern language approach has attracted a lot of interest in the field of object oriented software design. There is an active software patterns community with an annual conference, mailing lists, and web sites (see the patterns home page [13]). Software patterns have been receiving increasing attention from mainstream computer science, with a special issue of *The Communications of the ACM* on software patterns [14] joining the growing number of books on the subject (e.g., [4, 6, 7]). In contrast, the approach discussed in this chapter is very much in its exploratory phase; to my knowledge, no workplace pattern languages have been published.

It is important to note that the approach discussed in this chapter differs from those taken both by Alexander and the software design community. The principle difference is that this chapter emphasizes the use of pattern languages as a descriptive device, a *lingua franca* for creating a common ground among people who lack a shared discipline or theoretical framework. In contrast, both Alexander and the software patterns community tend to use patterns more prescriptively. The software patterns community focuses on using patterns to capture accepted practice and support generalization; Alexander's central concern is using patterns to achieve the ineffable "quality without a name," which characterizes great architecture. But these are differences of emphasis; in spite of my focus on patterns as a *lingua franca,* generalization, reuse of design knowledge, and increased quality of design all seem to me to be possible and desirable outcomes of pattern languages for workplaces.

The Alexandrian Pattern Language

Alexander's pattern language is described in two books, one of which provides the theory and usage model [1], and the other a language for planning and building which consists of a network of 253 patterns [2]. The patterns range in scale from a pattern for the distribution of towns and cities down to a pattern for walls. The patterns are loosely connected across scales: any given pattern typically points to smaller scale patterns which can support it, and larger scale patterns in which it may participate. For example, the *Identifiable*

Neighborhood pattern (aimed at creating neighborhoods with their own sense of place) invokes smaller scale patterns such as *Street Cafe*, *Individually Owned Shops*, and *Corner Grocery*, and participates in larger scale patterns such as *Mosaic of Subcultures* that specify characteristics of communities.

The pattern language is *not* intended to be a book of patterns that is followed by rote. It is actually a meta-language which is used to generate languages for particular sites. For any particular situation a subset of existing patterns is selected; in addition, designers modify existing patterns and create new patterns that reflect the culture, environment, history, customs and goals of the site's location and inhabitants. These patterns - old, modified, and new - form a site-specific language which is used to guide reflection and discussion about the relationships among the site, the proposed design, and the activities of the inhabitants.

Each pattern is presented in a standard form. For example, the *Street Cafe* pattern [2] begins with its name, number, and a photograph of a sidewalk cafe. The first paragraph describes some of the larger scale patterns of which it is part (e.g., *Activity Node*). Next is a statement of the essence of the pattern which illustrates the various forces responsible for the existence and nature of the pattern. This is followed by an often lengthy rationale section which describes the background of the pattern, evidence for its validity, ways in which the pattern can be manifested, etc. For *Street Cafe*, the rationale ranges from a careful description of the experience of being in a cafe, to an analysis of the elements of successful street cafes, to a discussion of a survey on the role played by student-oriented cafes in educational settings. *Street Cafe* concludes with a diagram and short statement describing how to implement the key features discussed in the rationale, and with a paragraph describing smaller scale patterns (e.g., *Opening to the Street*, *A Place to Wait*, and *Sitting Wall*) which may be used to strengthen this pattern.

Pattern Languages as Representational Systems

Alexander's pattern language has a number of attributes that make it an interesting candidate for a *lingua franca*.

> **Concrete Prototypes**: Alexandrian patterns are embodied as concrete prototypes, rather than abstract principles. Every pattern comes with a name (generally sufficient to evoke an image), a picture of an archetype of the pattern, and a diagram of how it is implemented. This is true for patterns at large scales (e.g., *Agricultural Valleys; Ring Roads*) and at small scales (e.g., *Thick Walls; Child Caves*). Whereas abstract principles require users of the principles to understand some conceptual framework, and to be able to map the principles onto their domain of concern, the concrete prototypes in pattern languages make direct contact with the user's experience. Anyone who has experience with the situation can quickly understand, discuss, and contest Alexandrian patterns.
>
> **Grounded in the Social**: Another characteristic is that the patterns tend to focus on the interactions between the physical form of the built environment, and the way in which that inhibits or facilitates various sorts of personal and social behavior within it. Thus, *Street Cafe* emphasizes the importance of the cafe being located along a busy path because this facilitates casual meetings, people watching, etc. This linkage between the design components and usage reinforces the concrete, grounded nature of the pattern language.
>
> **Expresses Values**: Alexander's pattern language is not value neutral. Patterns such as *Individually Owned Shops, Bike Paths and Racks, Farmhouse Kitchen, and Old Age Cottage* all manifest particular values in their names alone (and more explicitly in their rationales). While this aspect of *A Pattern Language* can alarm those who mistakenly believe it to be a universal

prescription for architecture, I see its ability to explicitly express values as part of its representational power, and as yet another way the language becomes grounded for its users.

Towards a Workplace Pattern Language

An important aspect of Alexander's pattern language is that its aim is to support the design of the built environment. In contrast, a pattern language for describing workplaces has to embrace considerably more than the physical architecture. What else should such a pattern language include? What would it look like? How would it be used? The best way to answer such questions would be to develop a workplace pattern language - however, this is far beyond the scope of this chapter. Nevertheless, I believe it is instructive to make a small beginning in order to get a feel for what a language might be like and how it might be used.

An Example: A Consulting Firm

My point of departure will be a study of a design consulting firm by Bellotti and Bly [3]. The company works with many clients at the same time, and provides a wide range of services. Its designers and engineers work on project-oriented teams (often more than one at a time), which form and reform as new projects begin and old ones end. The company culture encourages kibitzing and informal collaborations across team boundaries. The consulting firm is geographically distributed, and it is not unusual for the members of a project team to be located in different buildings, or even in different cities. While Bellotti and Bly did a general study of the company, their report focused on what they called "local mobility" (the tendency of workers to be away from their desks) and its impact on collaboration with local and remote coworkers. What I'm going to do is recast some of their results into pattern language form.

In recasting the results of this study as patterns, a couple of issues immediately arise. First, it turns out that some of the patterns from Alexander's architectural pattern language are relevant. For example, the Alexandrian patterns *The Flow Through Rooms* and *Office Connections* discuss the way in which the interconnection of rooms and the traffic through them can facilitate or inhibit spontaneous interaction - something that is an issue in Bellotti and Bly's study. Thus, a workplace pattern language might be able to build on some of the work that Alexander and his colleagues have already done. However, it is also clear that new patterns need to be added to describe the consulting firm offices; for example, a pattern language for this site would need to include patterns for *Model Shop*, *Central Scanning Station*, and *Open Plan Offices*, all spaces which play important roles in the life of this workplace.

In addition to new patterns describing spatial components of the workplace, new patterns are needed to describe two other entities: activities and roles. Activities range from formal events like a *Client Presentation*, to informal activities such as *Kibitzing*. Roles found in this workplace include *Manager*, *Engineer*, *Designer*, *Receptionist*, and so on. As with spatial patterns, patterns for roles and activities would describe the role or activity itself, its context and rationale, and its relationship to other patterns.

A Sketch of a Consulting firm Pattern Language

Let's look at some of the patterns that could describe the consulting firm. Because space is limited, I'll summarize the patterns; normally a pattern takes up several pages, and has a well-defined canonical structure. After describing a few patterns, and alluding to others, I'll discuss some of the ways in which a consulting firm pattern language might be used.

The largest scale pattern might be called *Consulting Firm*: its goal would be to characterize the consulting business by describing the various forces which shape it. Thus the pattern would depict the firm's need to act quickly and flexibly to get, keep, and complete projects, balanced with its need to do this with a relatively fixed set of human resources, and limited amounts of time and materials. *Consulting Firm* would also describe the multiple clients, simultaneous projects, loose teams, and informal collaborations that occur in the firm. It would end with pointers to the smaller scale patterns which support *Consulting Firm.*

These patterns would include:

> **Maintaining Mutual Awareness**
> Bellotti and Bly observed that it was important for designers and engineers to keep up-to-date with what was going on, regardless of the projects with which they were involved. This practice helped the company bring a wide range of expertise to bear on problems, and was a good counterbalance to the potential inflexibility of project-oriented teams. *Maintaining Mutual Awareness* is supported by a number of smaller scale activity patterns, ranging from *Blanket Email* (the custom of addressing email messages with questions or answers to large groups) to *Kibitzing*, to what one engineer called *Doing a Walkabout* (i.e. wandering through the work areas just to see what others were up to). *Maintaining Mutual Awareness* was also supported by spatial patterns such as *Open Offices, Model Shop* and *Central Scanning Station*.
>
> **Locally Mobile Workers**
> This pattern captures the fact that many engineers and designers spend considerable time away from their desks, but are still in the general vicinity. On the one hand, workers are pulled away from their desks by the need to get access to immobile shared resources such as *Model Shop* and *Central Scanning Station*, or to find local coworkers with whom they need to collaborate. On the other hand, they are pulled back to their desks by the need to use desk-based personal resources (PCs, telephones, voice mail) or to collaborate with remote colleagues. Thus, the engineers and designers spend considerable time moving about, a fact which - as Bellotti and Bly note - has considerable consequences for how they accomplish their work. Clearly to the extent that locally mobile workers encounter others (determined, in part, by *Open Offices*), this pattern supports *Maintaining Mutual Awareness*, at least for coworkers in the same locale.

Another pattern that goes hand in hand with *Locally Mobile Workers* is:

> **Receptionist as Hub**
> The mobility of many workers produces a need for coordination, a way for one person to locate another when the need arises. In this workplace Bellotti and Bly discovered that the receptionist played an important role in keeping track of people. This arose because her location and continuous presence at the entrance enabled her to observe the arrivals and departures of people. This was facilitated by her role as the conference room coordinator, which resulted in her being aware of the time, location, and composition of meetings. Finally, some employees, recognizing her role as *de facto* coordinator, had adopted the practice of informing her of their anticipated whereabouts.

Many other patterns could be described - *Open Offices*, *Shared Resources*, *Kibitzing*, *Blanket Email,* and *Doing a Walkabout* - but this is enough to give a flavor of what the consulting firm pattern language would be like.

Discussion

Let's stop and look at what's emerging here.

First, even in this fragment of a pattern language, we're seeing the basic characteristics of the Alexandrian language for built environments. We see the growth of a set of interrelated patterns at multiple scales: *Maintaining Mutual Awareness* is supported by *Locally Mobile Workers*, *Kibitzing* and *Blanket Email*, and *Locally Mobile Workers*, in turn, is supported by *Receptionist as Hub.* These patterns also focus on the interplay between the social and the physical, as in *Receptionist as Hub* being facilitated by the need for a person to coordinate the conference rooms. We also see that this language embodies values: *Kibitzing* and *Doing a Walkabout* are positive patterns that are supported by this firm's culture; it is easy to imagine workplaces in which such activities would be frowned up.

Now let's look at some of the ways a consulting firm pattern language might be used. One use could be to better anticipate the impact of new systems on the consulting firm. For instance, imagine that the consulting firm is considering the purchase of an automatic scheduling system. While the first order effects of this might be to reduce the receptionist's workload and to provide employees with more immediate access to room scheduling, *Receptionist as Hub* suggests that such a change might have less beneficial second order effects: reducing the receptionist's involvement in meeting scheduling might well reduce her effectiveness in keeping track of employees whereabouts. Similarly, switching from open offices to closed ones would clearly have impact *Maintaining Mutual Awareness* and *Kibitzing*. The point here is not that these patterns and their relationships should be used to reject or approve changes, but rather that they can be used as a language for discussing changes and reflecting on their possible impacts.

Another use of the consulting firm pattern language would be as a starting point for understanding a different type of workplace. Patterns from the consulting firm language can be sought in other types of workplaces. For example, a design team charged with implementing a new system in a hotel might, beginning with the consulting firm language, investigate the extent to which *Locally Mobile Workers* was present. If so, the workplace pattern language developed for consulting firms becomes a rich source of questions and hypotheses: What are the forces that sustain the *Locally Mobile Workers* pattern in this context - are they similar to those in the consultancy domain? In the consulting firm language, *Locally Mobile Workers* is supported by *Receptionist as Hub* - how do workers in the hotel domain achieve coordination? And so on.

Finally, we can see how, over time, the consulting firm pattern language might be generalized. For example, the design team studying the hotel might find that hotel workers achieve coordination primarily by means of shared objects, like bulletin boards and work sheets, rather than by relying on a person. The design team might devise a *Coordination Object* pattern that captures their findings, and also the findings of other researchers (for example, other coordination objects in the literature are flight strips in air traffic control [8], navigation charts in ship piloting [10], and time tables in control rooms for the London Underground [9]). Over time we might see the development of a large scale pattern, *Activity Coordination* that can be supported by either artifacts (*Coordinating Objects*) or by social roles (*Coordinator*).

Concluding Remarks

Design is becoming increasingly interdisciplinary. Neither 'designers' nor 'end users' are homogenous groups; they lack common disciplines, practices, and conceptual frameworks. All that we can realistically expect those involved in design to share is access to the situation for which they are designing. As a consequence, pattern languages, with their emphasis on embodying design knowledge as a network of concrete prototypes, have the potential to serve as a *lingua franca* for workplace design.

One question that might be posed is what advantages pattern languages offer over the more traditional means of presenting the results of workplace studies. In the course of this chapter, I've suggested the following:

- Patterns, especially smaller scale patterns, are more concrete, more tightly bound to the situation at hand, and thus more accessible to an audience that lacks a common disciplinary framework.
- Recasting the results of workplace studies as a network of interrelated patterns results in the modularization of workplace knowledge, and thus makes it easier to take a subset of a pattern language and apply it to a new type of workplace.
- This modularization also makes pattern languages more amenable to generalization across workplaces.

However, the most important advantage is a side effect of the form of representation. I believe that patterns are most likely be created, developed, and utilized by an audience that is largely distinct from those who read and write research papers. The fact is that the communicative goals of pattern languages and research papers are quite different. The values that underlie research, and lead to the publication or rejection of workplace studies, have to do with accuracy, careful examination and analysis of particular cases, and new contributions to the literature. However design teams faced with a new workplace and a short schedule have considerably more pragmatic needs. They need information that can be mastered quickly, applied to new situations, and used as a basis for dialog with their users. While researchers may, with full justice, be wary of the simplification and generalization encouraged by casting workplace research as patterns - this is precisely what design teams need. If pattern languages can assist design teams in communicating effectively with their users, noticing connections between activities and artifacts that would have been otherwise missed, or simply decrease the time between encountering a workplace and being able to ask useful questions, they will be a boon to design.

Acknowledgments

Thanks are due to Victoria Bellotti, Paul Dourish, Beki Grinter, Don Norman, Lucy Suchman, John Thomas, Mike Tschudy, and David Warren for comments on and discussions of various aspects of this paper. I am indebted to two architects, Dale Mulfinger and Robert Boylin, who spent considerable time providing me with practicing architects' perspectives on Alexander's pattern language, as well as helping me to understand its reception and use by layfolk and by professional architects. Finally, discussions in the CHI '97 Workshop on Interaction Pattern Languages deepened my understanding of these issues.

References

1. Alexander, C. A (1979) *Timeless Way of Building*. New York: Oxford University Press.

2. Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I., & Angel, S. A (1977) *A Pattern Language*. New York: Oxford University Press.

3. Bellotti, V. & Bly, S. (1996). Walking Away from the Desktop Computer: Distributed Collaboration in a Product Design Team. *Proceedings of CSCW '96*.

4. Coplien, J. O. (1995) A Generative Development-Process Pattern Language. In *Pattern Languages of Program Design*. (eds. J. O. Coplien and D. C. Schmidt). Reading, Mass: Addison-Wesley, pp. 183-237.

5. Fromm, D. & Bosselmann, P. (1985) Mexicali Revisited: Seven Years Later. *Places*, Vol. 1, No. 4, pp. 78-91. New York: The Design History Foundation.

6. Gabriel, R. P. (1996) *Patterns of Software: Tales from the Software Community*. New York: Oxford University Press.

7. Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995) *Design Patterns: Elements of Reusable Object-Oriented Software*. Reading, Mass: Addison-Wesley.

8. Harper, R., Hughes, J.A. and Shapiro, D.Z. (1991) Working in Harmony: An Examination of Computer Technology in Air Traffic Control. In J. M. Bowers & S.D.Benford, (eds.). *Studies in Computer Supported Cooperative Work: Theory, Practice and Design*. Amsterdam: Elsevier, pp. 225-234.

9. Heath, C. C. and Luff, P. (1991) Collaborative Activity and Technological Design: Task Coordination in London Underground Control Rooms. *Proceedings of E-CSCW 91* (Amsterdam, Sept. 24-27 1991), pp. 65-80.

10. Hutchins, E. (1990) The Technology of Team Navigation. *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work* (ed. J. Gallagher). Hillsdale, NJ: Lawrence Erlbaum, pp. 191-220.

11. Luff, P., Heath, C., and Greatbatch, D. (1992) Tasks-in-interaction: Paper and Screen Based Documentation in Collaborative Activity. *CSCW 92 Proceedings* (November, 1992), pp. 163-170.

12. Mulfinger, D. (1996) Personal communication, in a conversation on July 12, 1996, Minneapolis, Minnesota, U.S.

13. Patterns Home Page. http://st-www.cs.uiuc.edu/users/patterns.

14. Schmidt, D., Fayad, M., Johnson, R. (eds.) 1996. Special Issue on Software Patterns. *Communications of the ACM*, Vol. 39, No. 10. New York: ACM Press.

[Tom's Home Page]
[Professional] [Life, Fun, &c] [Tell Me...]
[Bookmarks] [Publications List] <and many papers and essays>