

Patterns or Claims: Do they help in communicating design advice?

George Abraham

College of Information Science and Technology
Drexel University
Philadelphia, PA USA
george.abraham@ischool.drexel.edu

Michael E. Atwood

College of Information Science and Technology
Drexel University
Philadelphia, PA USA
atwood@drexel.edu

ABSTRACT

It has been asserted that patterns or claims will help capture and communicate interaction-design advice. Both structures attempt to provide advice in context along with the justifications for fit. These properties aim to make patterns or claims more concrete and comprehensible to novice designers than design guidelines. However, empirical work evaluating these promises is lacking. This research presents a controlled study that examines the value of structuring design advice as patterns or as claims. Patterns and claims seem different given their respective roots in architecture and design rationale. They also differ in their emphasis when capturing design decisions; patterns emphasize capturing a problem-solution pair in a certain context, whereas claims focus on capturing the positive and negative implications to a design decision. The findings from the study suggest it may be promising to combine the claim and pattern structures and that such a structure may facilitate discussions of design trade-offs.

Author Keywords

Pattern, claim, controlled-study, context, trade-offs, advice

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION

How should HCI researchers and practitioners communicate design advice to their clients, end-users, or to each other? For a discipline with a strong belief in an iterative design-evaluate cycle, how advice gleaned from one evaluation cycle feeds into the next design cycle is clearly important.

It has been argued that HCI design knowledge is traditionally presented in a format that is comprehensible only to members of the HCI community [32]. Given the inter-disciplinary nature of design, sharing design knowledge may be critical. Some have theorized that the concrete and contextual nature of advice presented as patterns or claims will facilitate comprehension [15] [32],

and sharing. Patterns or claims can also possibly serve as middle ground between high level guidelines or principles and concrete instances of design.

Patterns and claims share at least two goals in common: (1) reuse and (2) communicating design advice. It is unclear, however, whether patterns or claims actually add value to the design process. At present, there is no clear empirical evidence demonstrating the benefits of capturing advice as pattern or claims. This study corrects this deficiency by exploring whether expressing design advice using the pattern or claim-structure influences its perceived effectiveness.

This study differs from past research in two ways:

- It evaluates possible benefits of structuring design advice in a pattern or claim form rather than the efficacy of using an existing pattern collections or claims library
- It studies what impact such a structure may have on sharing design advice.

By focusing on structure of advice, we aim to explore the benefits and challenges surrounding communication and reusability of design advice. This will enable us to extend the current discussion on patterns or claims by specifying how we might articulate design advice.

This paper is organized as follows. First, we provide a brief introduction to patterns and claims and their inherent structure. After this we offer an overview of empirical work examining the value of patterns or claims to communication. A comparison of pattern and claim structure is then shown to highlight the affinities and differences between the two approaches. We then describe the methodology used to compare the two structures followed by implications of our findings to future work in this area.

PATTERNS AND CLAIMS IN HCI

This section provides an introduction to patterns and claims to describe how their structures have been described in the literature.

Patterns in HCI

Patterns in HCI draw their inspiration from patterns in architecture, introduced by Alexander et al. [1]. The HCI community has proposed that interaction patterns may serve as a common language for users and designers to voice their ideas and as a possible format to communicate

OZCHI 2009, November 23-27, 2009, Melbourne, Australia.
Copyright the author(s) and CHISIG
Additional copies are available at the ACM Digital Library
(<http://portal.acm.org/dl.cfm>) or ordered from the CHISIG secretary
(secretary@chisig.org)
OZCHI 2009 Proceedings ISBN: x-xxxxx-xxx-x

interaction design knowledge to those interested in HCI [3] [5] [15].

However, there is no universally accepted definition for a pattern in HCI; different researchers use different definitions. Therefore, we attempt to stay as close as possible to the definition proposed by Alexander et al [1]. The pattern structure brings together three elements- a description of the recurring design problem, a solution showing a configuration that brings into balance the “forces,” and a definition of the context in which the problem-solution pair holds true. For example, an ALCOVE pattern describes a spatial configuration (solution) that balances opposing forces of community and privacy (problem) in large communal rooms (context).

Claims in HCI

Claims analysis involves identifying the positive and negative implications on usability, or use, stemming from a design choice. A scenario of use involving the task and artifact forms the basis for this reflective exercise. Carroll and Kellogg [8] argue that HCI artifacts present opportunities for building theory in HCI. Such efforts may serve as a middle ground between concrete instances of use and abstract theories in HCI. They assert that “HCI artifacts embody psychological claims about contexts-of-use” [id, p.8]. Claims analysis highlights design trade-offs when making a design choice. Carroll and Rosson [9] describe the basic claim structure as follows:

“(Artifact feature or technique) CAUSES (desirable psychological consequence) BUT MAY ALSO CAUSE (undesirable psychological consequence)” [id., p.193].

For example, a flashing window icon on the task bar of a computer will alert user that a background process needs attention, *but may also distract the user from the current window or task* (it is possible for a feature to have more than one positive or negative consequence).

To be accurate, a claim is an evaluative statement expressed as a positive or negative design implication. For the sake of clarity and convenience, we shed the term claims-analysis. Instead, we adopt claim-advice to refer to design advice that is presented as a usage scenario along with possible pros and cons (claims) of implementing a design decision.

EMPIRICAL WORK ON PATTERNS AND CLAIMS

A frequent argument offered for both patterns and claims is that they are helpful in communicating design advice. Before evaluating this promise, we review related research in the section below.

Dearden and Finlay's review of patterns and pattern languages in HCI reveal clearly that there is insufficient evidence demonstrating benefits of patterns or pattern languages, either to the design process or the designed product [14]. We have seen several recent studies exploring this question in the area of ubiquitous computing [11] [29], participatory design [12] [13], teaching HCI [4] [22], and information retrieval [34]. Among them, the controlled studies [11], [22], [29], [34]

focused on the efficacy of using patterns for improving the quality of the design product. The evidence from these studies is not conclusive as to whether patterns help. In fact, these studies show no clear advantage of using patterns to design interactive systems. Nevertheless, these studies all argue that patterns may help communication during the design process, and should be explored further.

Studies discussing the direct benefits of reusing claims to design are similarly lacking. A survey of empirical work on claims can be broadly classified under three categories: discussions on the theory behind claims and its application [7] [9]; using claims analysis as part of the scenario-based design (SBD) framework [10] [28]; and how to organize claim library to promote reuse of claims . In terms of value to the design process, Carroll and Rosson [9] point to projects where a scenario-based claims analysis was applied with success. A detailed discussion on using claims analysis as part of teaching HCI using case studies and SBD is also available [30]. These studies, however, do not to evaluate how claims communicate design knowledge in situations when claims are reviewed outside the context from which they were derived. Nor do they show clear empirical support for the value of claims reuse in design. That is, how an archived claim will help a designer with a design problem at hand. There are studies reported that evaluate findability of claim in a claims library. These studies seem targeted at understanding how to organize claims to promote access or findability [16] [17] [26], and not on assessing the usefulness of advice presented in or as claims.

In summary, we did not find much evidence of controlled studies involving claims reuse or patterns that evaluate the impact on communicating design advice. Discussion have mainly focused on the form and organization [6][27]. We believe improving access to design advice is important, but it's equally, if not more, important to examine whether the design advice is perceived as usable and effective first. Without evidence showing that articulating design advice as patterns or claims facilitates sharing and comprehension of design knowledge, the value of organization schemes, however well-designed and argued, is unclear.

From what we have seen, discussions of patterns or claims seem united in purpose, but except for a brief comparison between patterns and claims in [14], research on these two approaches rarely intersect. We too see patterns and claims share a lot in common. Both:

- Emphasize delivering design advice situated in the context-of-use
- Share a common goal of abstracting from successful solutions for design problems, and extending HCI design knowledge.
- Are aimed at facilitating communication of design advice by making design advice usable. Discuss design trade-offs

COMPARING PATTERNS AND CLAIMS

Patterns and claims are arguably similar in many respects. Both describe the context in which the design decision will be deployed. Both describe the solution that this

decision implements. They differ in that while patterns explicitly address the problem to be solved, claims do not. Rather, claims focus on the advantages and disadvantages of the solution while patterns consider only the advantages.

There are three areas in which patterns and claims differ – (1) in whether they are problem-driven (patterns) or solution-driven (claims), (2) how exactly they define context and (3) how explicitly they present the rationale for design decisions.

Problem-driven or Solution-driven

Pattern descriptions tend to include a specific problem statement that highlights the constraints present in the design space [14]. These constraints stem both from the task (e.g., shopping for furniture on the web) and the task context (i.e., aspects unique to furniture shopping, such as coordinating colors or size). However, we have seen patterns in HCI replace this problem statement with a generic task description, which states what the user is attempting to do. For example, the problem solved by an accordion menu is that an user needs to find an item in the main navigation.

A *claim* usually does not include an explicit problem statement [14]. Rather, a usage scenario motivates a claim [31]. For example, in discussing reuse of claims such as “virus warning through informational pop-ups” [16] or “rare event monitor” [31], a usage scenario sets the context for discussing pros and cons of adopting this solution. The object of claim-advice seems to be presenting a balanced view for a particular solution given a usage context.

In this sense, patterns can be argued as problem-centric versus claims, which seem solution-centric. This distinction may be important considering that when designers are looking for advice, it's usually to find a way out of a problem for which they don't have a solution. So describing the problem explicitly may enable a comparison of the problem at hand with that described in the pattern. Expressing trade-offs as part of the solution may be equally important as this will enable designers to evaluate the solution itself.

Definition of Context

Solutions in patterns are abstracted from multiple instances of a common problem. Patterns attempt to include a range of contexts where the problem-solution pair holds true. There is no set rule about how many solutions or situations need to be considered before declaring it a pattern. Reviewing patterns in HCI will show an explicit discussion of context under a “use when” section describing some leading aspects of situations where this solution is relevant. However, we did not find any recommendations on how context needs to be articulated, nor did we find any clear strategies used in discussing context.

Claims do not explicitly focus on describing context. A usage scenario (task-artifact) situates a claim, without which a claim loses its concreteness and context. In terms of scoping the context-of-use, claims do not attempt to be

exhaustive, but instead resort to considering the leading aspects of the usage scenario and how the intended solution will affect the user.

Conveying Design Rationale

Pattern descriptions include justifications as to why it is a good solution. The supporting rationale tends to be sourced empirically (i.e., the solutions need to exist). While most pattern descriptions include rationale, it is not clear what or how it should be recorded. The core pattern structure does not require an emphasis on negative consequences of implementing this solution or limitations. Example instances of the solution (i.e., screenshots) are included along with the advice to reinforce that these solutions indeed exist.

The rationale aspect in *claims* seems more fleshed out than that in patterns, and more detailed instructions describe what to capture [9]. Unlike patterns, claims describe both the positive and negative consequences on the use or usability. The basis for claims can be analytical or empirical. Furthermore, claims tend to present a verified solution rather than generalizing from multiple successful solutions.

Patterns and Claims: Review Summary

Given what we now know about patterns and claims, the contribution of this research lies in exploring how these similarities and differences contribute to the effectiveness of design advice. Rather than examining whether participants in our study produced patterns or claims, we looked at how these structures help explain and communicate interface design decisions.

In summary, this study explores the following two research questions: (1) What impact does the advice structure used have on communicating interaction design advice? (2) How does advice structure impact the time required to prepare design advice?

METHOD

To evaluate whether the advice structure impacts the effectiveness of design advice, a between-subjects design was used to randomly assign 57 participants into three groups: pattern-derived structure, claim-derived structure, and control (no specified structure). All participants were undergraduate students from an information and technology school, and had completed coursework in HCI. Participants were familiar with the concept of ease-of-use, design principles, and heuristics (completing at least an introductory course in HCI was a pre-requisite for the study).

All participants were asked to adopt the role of a designer and recommend a menu design for a room planner interface. Participants received a simple version of a room planner (Figure 1). The instructions stated that a room planner will enable users to add and remove furniture from an on-line catalog to a room plan, and allow them to try out different arrangements.

To focus the participants on capturing design advice, instead of asking each of the participants to come up with their own menu design, they received a sheet with a

randomized ordering of three menu options: *Directory-tree*, *Drop-down*, and *Accordion*. These options were selected because they are fairly common in interactive systems. These options were presented side-by-side and included a brief set of features describing the menu. For example, the *Directory-tree* menu listed the following:

- Expand/Collapse different sections using the +/-
- Multiple sections can be opened at a time
- Product information and image displayed inside each menu
- Fixed height for the menu
- Scroll (outside) to view the open catalog.

The other two menus also listed five features customized for the menu type (i.e., for the *Accordion* menu it read that clicking on the arrows open/closes the section).

Participants were asked to rank order the three menu options and then to explain their top pick. Ranking was introduced to prompt participants to consider all the three options. Depending on the assigned condition, participants were given additional instructions; participants in the *pattern* and *claims* condition were asked to address three questions when explaining their menu choice (see Table 1).

All participants were asked to explain their design decision as if they were teaching someone else to make a similar choice, and such that others reviewing this design advice may not see the menu options they rejected. Their recommendations were rated for quality by two expert judges (each having over 20 years experience in HCI education and research; an approach adopted by other controlled studies on patterns). No absolute measures exist to evaluate design advice because no representation is complete or sufficient in itself to fit all purposes; nonetheless, the following four criteria were adapted from other studies [20] [24] [25]:

- *Context of Use Described* or how well does the design advice describes the context of design advice (i.e., usage scenario, specific examples)
- *Rationale Expressed* or whether there were justifications present for calling the menu option a good fit
- *Usefulness* or how well does the design advice address the fit between the design choice and the problem context (i.e., relevancy of rationale)
- *Overall Quality* or in general how good is the prepared design advice

In addition to these four measures, time taken for completing the exercise was considered as an additional dependent measure.

The recommendation or design advice was rated on these four criteria using a 7-point Likert-type scale. For each recommendation, the ratings from the two judges were averaged. Treating the rating data as interval-type, intraclass correlation coefficient (ICC) was used to assess reliability of ratings between the two judges.



Figure 1. Participants recommend one menu option, from the following three, to be used on the room-planner interface for users to peruse the furniture catalog and make their selections: a) Directory-Tree; b) Drop-Down; c) Accordion.

Table 1. Additional instructions provided to the participants based on the assigned condition.

| |
|---|
| <p>Pattern Condition</p> <p>What is the user trying to do?</p> <p>How will the recommended menu option be used?</p> <p>Explain why it's the best option in this given situation?</p> |
| <p>Claims Condition</p> <p>How will the recommended menu option be used?</p> <p>Advantages of using this design choice?</p> <p>Disadvantages of using this design choice?</p> |

FINDINGS

Inter-rater reliability (ICC) was 0.81 overall, which showed a good level of consistency.

Of all the dependent measures (context, rationale, usefulness, overall quality, and time), analysis of variance (ANOVA) of the rating data found the differences in *usefulness* to be significant, $F(2, 18) = 6.783$, $p < 0.0125$. The p value was set at .0125 level as part of Bonferroni's correction (Fields, 2005) to account for type-II error. A post-hoc analysis using Scheffe's post-hoc criterion showed that the advice prepared using the claim structure received higher mean ratings than either the patterns or control groups. Table 2 below shows the F and p values from ANOVA for the other variables found not significant.

The advice structure did not affect the time required to prepare recommendation. The overall quality between the three was also not significant.

Table 2. Results from ANOVA for each variable

| Measure | F Value | p Value |
|-----------|------------------|----------|
| Context | F (2, 54) = .52 | p = .596 |
| Rationale | F (2, 54) = .859 | p = .429 |
| Quality | F (2, 54) = .74 | p = .48 |
| Time | F (2, 54) = .76 | p = .44 |

Table 3. Descriptive statistics showing means(M) and std. deviations (SD); * significant at p<0.05. A 7-pt Lickert type scale was used.

| Criteria | Results in (M, SD) format | | |
|-------------|---------------------------|-------------|-------------|
| | Pattern | Claim | Control |
| Context | 3.00, 1.07 | 2.71, 1.13 | 2.68, 0.96 |
| Rationale | 3.50, 0.88 | 3.84, 1.29 | 3.39, 1.08 |
| Usefulness* | 3.13, 0.99 | 3.95, 1.12 | 2.79, 0.85 |
| Quality | 3.18, 0.87 | 3.34, 0.90 | 3.00, 0.83 |
| Time (mins) | 12.73, 5.14 | 13.15, 5.07 | 11.14, 5.63 |

Usefulness of Advice

The advice structure had an impact on the usefulness of design advice. Claim structure performed significantly better than the pattern or control. Advice was rated more useful based on how many relevant implications were considered. To distinguish usefulness from rationale, rationale assessed whether reasons were given for picking a particular menu (evaluative comments) whereas usefulness evaluated the content of the cited implications.

Compared to the pattern and control condition, claim advice contained a significantly higher number of unique design implications, $F(2, 54) = 6.32, p = .003$. If we look at the advice structures provided to the participants, pattern structure differed from the claim structure in two ways. First, it asked participants to discuss “what the user was trying to do”. Secondly, the claim structure specifically asked participants to consider the disadvantages of their design choice. Pattern condition asked participants to describe why the suggested menu was appropriate in the given situation. Participants in the control condition did not specify any set structure, and were asked to explain why they would recommend a menu option.

The claim advice seems to have benefitted from a discussion of disadvantages. A subsequent content-analysis of prepared advice grouped the evaluative comments under ten types of design implications that were considered by the participants (e.g., visibility, efficiency). These represented a set of design issues considered by the participants as a whole. We found that

by discussing advantages and disadvantages of their design choice, the participants in the claim condition also considered a significantly higher number of design implications. The number of unique design issues had a significant and strong correlation with usefulness ($r = .686$), rationale ($r = .559$), and overall quality ($r = .598$) ratings, at $p < .05$ level, which seem to corroborate what we discovered from expert judging. Context ratings were weakly correlated with number of unique design issues ($r = .224$) and was not significant either.

Rationale Presented in Design Advice

Expert ratings did not reveal a significant difference between the three conditions when expressing rationale. In other words, the structure did not impact how much rationale was provided. In examining how participants argued for their design decision, we found that the claim condition elicited significantly higher number of evaluative comments (positive+negative) compared to the control condition or pattern condition, $F(2, 54) = 6.37, p = .002$. An example positive evaluative comment was stated as, “This design choice will allow the user to know all of their options at any one point in time (multiple categories can be open at once). The user can scroll through the entire open category at once.” The pattern and control condition almost rarely discussed negative implications of design choice. The difference in the number of evaluative negative comments was significant, $F(2, 54) = 114.882, p = .001$.

We observed that the claim structure prompted participants to be more reflective about their recommendation, both resulting in a higher number of evaluative comments, and as we saw earlier, lead to consideration of more design implications. This leads us to theorize that it may help to make explicit negative implications of design decisions in design advice.

Context of Use in Advice

The difference in the mean ratings for context of use based on advice structure was not significant. And while the context-of-use ratings was found to be higher for the pattern condition, this margin was not significant. Examining the content of design advice showed that participants in both pattern and claim condition discussed their design more than did the control group (c.f. table 4 on word-count). The difference in mean word count was significant between patterns-control and claims-control, $F(2, 54) = 4.27, p = .02$.

The content of advice when compared on aspects that may be commonly associated with context (e.g., user, task) showed that pattern advice discussed context related aspects more than the control group. The number of words describing context also showed a strong correlation with context-of-use expert ratings ($r = .509, p = .001$).

Table 4. Means and Standard Deviation for #Words used in the three conditions.

| Condition | Mean | Std. Deviation | N |
|-----------|----------|----------------|----|
| Pattern | 200.3684 | 78.09838 | 19 |
| Claims | 200.4211 | 72.79752 | 19 |
| Control | 143.1579 | 56.26649 | 19 |
| Total | 181.3158 | 73.63791 | 57 |

DISCUSSION

Controlled studies in the past explored whether exposure to a set of patterns impacted the resulting quality of the interface [11][29][34]. The results from these studies were not conclusive in favor of patterns. Nevertheless, it was argued that patterns may help communicate design knowledge; and therein lies the real value of patterns. A similar argument can be found in the claims-reuse literature where claims were promoted as a usable and comprehensible form of interaction design advice [32].

This controlled study explored the promise of communication by examining whether presenting design advice using a pattern-derived or claim-derived structure influenced the effectiveness of advice. A pattern-derived structure suggested a problem-solution-context form whereas the claim-derived structure suggested participants to document advice as a usage scenario, pros and cons of the design decision.

In the following discussion we discuss our findings with respect to usefulness, rationale, and context of use, and what these findings mean for pattern or claim research for sharing design advice.

Case Against “Natural” Design Advice

It appears that the pattern structure, as presented, did not offer a strategy for participants to easily identify and explore possible design implications. Given the freedom, participants displayed a tendency to “sell” their idea, and in the process were less critical. The control condition also displayed a similar tendency.

Offering only positive evidence, in the pattern and the control-advice, can be argued as being a “natural” strategy for communicating design advice. The findings, however, show that “natural” may be not be enough. The pros and cons approach in claim structure seems to have helped participants in considering more number of design implications, and in the process considered more design issues. This forces us to reconsider the value of discussing cons in design advice.

Despite the argued affinities between design in HCI and in architecture, pattern-advice for interactive systems may require a different treatment. Alexander’s [1] version of patterns does not explicitly address negative implications. Does that imply interaction design can afford to do the same? We can think of at least two reasons why this is the

case for architecture: (1) we may be implicitly aware of the cons or residual forces, through our lived-in experiences, and pointing them out may be redundant; (2) A pattern contains a solution that helps bring the forces into balance, such the net forces are zero, which may be interpreted as no negative implications.

Lived-in Experiences and Interactive Systems

We believe cons, or negative implications, may prove to more helpful in advice about interactive systems since we do *not* have the benefit of lived-in experience. And hence, may be unaware of the negative implications. Furthermore, it’s highly unlikely that any design solution is perfect enough such that there are no negative implications. Patterns in collections like Tidwell’s [33] collection seem to acknowledge this, and try to address negative implications in the pattern description. However, the impact of describing negative implications has not garnered much attention in the pattern literature. Even the standardized format suggested for formatting patterns, pattern language mark-up language (PLML), while being extensive, does not acknowledge the possible value of cons to communicating design advice. We feels negative implications may be just as important to communicating design advice, as is describing context or sharing a “good” solution.

Sufficiency of Context in Advice

The challenge for capturing context is to balance specificity of advice with generalizability. While an exhaustive specification of context may lend concreteness to design advice, it may also impede the designers’ ability to critique this advice in reuse situations. Moreover, an exhaustive specification is impossible. One primary argument favoring pattern or claims has been the presence of context in design advice. In the study, pattern-advice and claim-advice did not differ much when discussing aspect related to context.

It’s even more challenging to articulate context separate from the design solution (or problem) itself. Originally, patterns were about a recurring problem and a good solution discussed *in* context. However, over the years pattern collections have inadvertently pushed an agenda for discussing problem, solution, *and* context. The evidence of this lies in the inclusion of context in “use-when” or “why” sections in pattern descriptions. In doing so, it gives the impression that the problem and solution can be abstracted from the context, and separating the context is a good thing. The design advice prepared by the participants showed that it was more natural to pair each design implication with the context of use. In fact, the design implication lost its meaning without specifying the context of use alongside.

We found that using a pros and cons approach prompted participants to implicitly describe leading aspects of context. For example in stating a positive implication the participant states, “the +/- button on a directory-tree menu will be familiar to users’ based on their prior experience to indicate more or less items.” In offering such a positive evaluative comment, the participant implicitly touched upon aspects of the interaction context like users’

experience level. The pattern structure by itself did not provide much guidance on describing context. The discussions on patterns in HCI too have eluded providing any recommendations on how to approach discussing context. This is not a critique of any particular pattern collection, but more a reflection on the state of pattern research. We found that drawing the attention of our participants to explicitly account for context did not make a significant difference. We found that the claim structure seems to have achieved what a pattern-advice was meant to be, describing advice *in* context.

CONCLUSIONS AND FUTURE WORK

Patterns and claims both present themselves as viable options for sharing design advice, but empirical work examining their similarities and differences is very limited. Furthermore, discussions on patterns and claims rarely overlap. This controlled study found that design advice prepared using the claims structure was perceived to be more useful.

We plan to pursue future work in following directions: (1) exploring a new formalism for sharing design advice; (2) expand the discussion on patterns or claim as a complimentary approach to existing design methods, where the emphasis is on learning from design trade-offs.

Rather than declaring pattern or claim structure as a clear winner, we believe future work in this area should explore a hybrid structure, a new formalism, for sharing design advice. This new structure proposes articulating design advice as problem and solution *in* context, and then explicating the pros and cons of applying the solution. The concept of including a problem description is borrowed from the pattern concept. The solution can be presented usage scenario that situates the design artifact in use. And finally, adopt a pros and cons structure for articulating trade-offs. Describing the problem in context will allow readers to evaluate the solution against the problem itself, and specifying the pros and cons of the solution will afford readers an opportunity to evaluate the solution itself.

More importantly, asking whether patterns or claims communicate design advice better than guidelines because they are concrete and include context may be the wrong question! Patterns or claims are not just context-rich versions of guidelines, but are more about describing trade-offs in design. This latter contribution has been under-appreciated. Given that design problems are messy (a set of intersecting design issues), good design is about making the right trade-offs. We do not yet have a formal way to learn about design from design (trade-offs). We believe patterns and claims may offer a way to capture and share successful design trade-offs in context.

REFERENCES

- Alexander, C.: The timeless way of building. Oxford University Press, New York (1979)
- Alexander, C., Ishikawa, S., Silverstein, M. et al.: A pattern language: Towns, buildings, construction. Oxford University Press, New York (1977)
- Bayle, E., Bellamy, R., Casaday, G. et al.: Putting it all Together: Towards a Pattern Language for Interaction Design: A CHI 97 Workshop. SIGCHI Bulletin, 30 (1998)
- Borchers, J.: Teaching HCI Design Patterns: Experience from Two University Courses. Minneapolis, MI (2002)
- Borchers, J.: A pattern approach to interaction design. 1st edn. Wiley (2001)
- Borchers, J.: CHI Meets PLoP: An Interaction Patterns Workshop. SIGCHI Bull., 32 (2000) 9-12
- Carroll, J. M., & Rosson, M. B.: Deliberated Evolution: Stalking the View Matcher in Design Space. Design Rationale: Concepts, Techniques, and Use, (1996)
- Carroll, J. M., Kellogg, W. A., Rosson, M. B.: The Task-Artifact Cycle. (1991) 74-102
- Carroll, J. M., & Rosson, M. B.: Getting Around the Task-Artifact Cycle: How to make Claims and Design by Scenario. ACM Trans.Inf.Syst., 10 (1992) 181-212
- Chin, G., Rosson, M., Carroll, J.: Participatory Analysis: Shared Development of Requirements from Scenarios. (1997) 162-169
- Chung, E., Hong, J., Lin, J. et al.: Development and Evaluation of Emerging Design Patterns for Ubiquitous Computing. (2004) 233-242
- Dearden, A., Finlay, J., Allgar, E. et al.: Using Pattern Languages in Participatory Design. Proceedings of the Participatory Design Conference, (2002) 104-112
- Dearden, A., Finlay, J., Allgar, L. et al.: Evaluating Pattern Languages in Participatory Design. Conference on Human Factors in Computing Systems, (2002) 664-665
- Dearden, A., & Finlay, J.: Pattern Languages in HCI: A Critical Review. Human-Computer Interaction, 21 (2006) 49-102
- Erickson, T.: Lingua Francas for Design: Sacred Places and Pattern Languages. (2000) 357-368
- Fabian, A., McCrickard, D. S., Felton, D. et al.: Designing the Claims Reuse Library: Validating Classification Methods for Notification Systems. Proceedings of the 42nd annual Southeast regional conference, (2004) 357-362
- Fabian, A., Wahid, S., Bhatia, S. et al.: Creating an Interactive Learning Environment with Reusable HCI Knowledge. (2006) 2314-2321
- Henninger, S., & Corrêa, V.: Software Pattern Communities: Current Practices and Challenges. (2007)
- Horner, J., & Atwood, M. E.: Design Rationale: The Rationale and the Barriers. Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles, (2006) 341-350
- Jeffries, R.: Usability Problem Reports: Helping Evaluators Communicate Effectively with Developers. Usability inspection methods table of contents, (1994) 273-294
- Karsenty, L.: An Empirical Evaluation of Design Rationale Documents. Proceedings of the SIGCHI conference on Human factors in computing systems: common ground, (1996) 150-156
- Kotze, P., Renaud, K., Biljon, J. v.: Don't do this-- Pitfalls in using Anti-Patterns in Teaching Human- Computer Interaction Principles. Computers & Education, 50 (2008) 979-1008

23. McCrickard, D. S., Chewar, C. M., Somervell, J. P. et al.: A Model for Notification Systems Evaluation- Assessing User Goals for Multitasking Activity. *ACM Transactions on Computer-Human Interaction*, 10 (2003) 312-338
24. Molich, R., Jeffries, R., Dumas, J.: Making Usability Recommendations Useful and Usable. *Journal of Usability Studies*, 2 (2007) 162-179
25. Molich, R., Hornbaek, K., Scott, J.: Recommendations on Recommendations. (2007) 1921-1924
26. Payne, C., Allgood, C., Chewar, C. et al.: Generalizing Interface Design Knowledge: Lessons Learned from Developing a Claims Library. *Information Reuse and Integration, 2003.IRI 2003.IEEE International Conference on*, (2003) 362-369
27. Perspectives on,H.C.I.Patterns Concepts and Tools
C.H.I.Workshop report: Perspectives on HCI Patterns:
Concepts and Tools. *CHI 2003 Workshop Report*. (2003)
28. Rosson, M., & Carroll, J.: Scenario-Based Design. (2003) 1032-1050
29. Saponas, S., Prabaker, M., Abowd, G. et al.: The Impact of Pre-Patterns on the Design of Digital Home Applications. (2006) 189-198
30. Somerveir, J., Chewar, C., McCrickard, D.: Making a Case for HCI: Comparing Materials for Case-Based Teaching. *Frontiers in Education, 2004.FIE 2004.34th Annual*, (2004) T3H
31. Sutcliffe, A., & Carroll, J.: Designing Claims for Reuse in Interactive Systems Design. *International Journal of Human-Computer Studies*, 50 (1999) 213-241
32. Sutcliffe, A.: On the Effective use and Reuse of HCI Knowledge. *ACM Trans. Comput. -Hum. Interact.*, 7 (2000) 197-221
33. Tidwell, J.: *Designing interfaces: Patterns for effective interaction design*. O'Reilly Media, Inc (2005)
34. Wania, C. E.: Examining the Impact of an Information Retrieval Pattern Language on the Design of Information Retrieval Interfaces. (2008)