

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/228979455>

Analyzing the HCI Design Pattern Variety

Article · March 2010

DOI: 10.1145/2371736.2371745

CITATIONS

8

READS

178

2 authors:



Christian Kruschitz

Alpen-Adria-Universität Klagenfurt

6 PUBLICATIONS 39 CITATIONS

[SEE PROFILE](#)



Martin Hitz

Alpen-Adria-Universität Klagenfurt

144 PUBLICATIONS 1,102 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Software for Statistical Analyses [View project](#)



Modelling and Simulation Environments [View project](#)

Analyzing the HCI Design Pattern Variety

Christian Kruschitz
Department of Informatics-Systems
University of Klagenfurt
Klagenfurt, Austria
chris@isys.uni-klu.ac.at

Martin Hitz
Department of Informatics-Systems
University of Klagenfurt
Klagenfurt, Austria
hitz@isys.uni-klu.ac.at

ABSTRACT

Human-Computer Interaction (HCI) design patterns are an often used tool for developing user interfaces. They render the communication among stakeholders more efficient and allow for a faster design of user interfaces. However, today there exists a vast amount of patterns written by many different authors, published on Web repositories, in scientific papers, and books. This causes the form or structure of the patterns to vary according to the authors' preferences. This paper presents the results of a survey that analyses the structure and relationships of HCI design patterns from 21 different design pattern resources.

Categories and Subject Descriptors

H.5.2 [User Interfaces]: Theory and methods; H.5.2 [User Interfaces]: Style Guides

General Terms

Design Pattern, Formalization, Human-Computer Interaction, Pattern Structure, Usability, Survey, Standardization

1. INTRODUCTION

We investigated a selection of 21 HCI pattern languages and collections that were published through Web repositories, scientific papers and books between 1996 and 2007. Our main goal of this survey was to analyze the similarities in the pattern form in order to specify a unified HCI design pattern format that should be used as the basis of XPLML (eXtended Pattern Language Markup Language). An unification of the pattern structure should be achieved to exploit the full reuse potential of patterns, which does not constrain pattern authors in their work but supports pattern users by easing the process of understanding and instantiating patterns to specific design problems[12].

Four books, eight scientific papers and nine online resources were examined. Three out of four books deal with pattern

languages [5, 24, 10] and one is a pattern collection [21]. Two papers describe a pattern language [9, 15, 27] and five papers portray pattern collections [6, 7, 18, 20, 25]. Finally, nine pattern collections were found online [11, 13, 16, 17, 23, 22, 28, 29, 30], the most important resource for design pattern users, because of the easy access and high availability.

The survey is divided into seven subcategories which are described in more detail. These categories are:

- Publication Year/Publication Medium
- Arrangement of Design Patterns
- Device Categories for HCI Design Patterns
- HCI Design Pattern Domains
- HCI Design Pattern Categorization
- HCI Design Pattern Structure
- Pattern Relationships

From the vast amount of design patterns available, we have chosen the above-mentioned repositories and pattern sources because we believe that they reflect the current state of efforts in the HCI design pattern community well.

2. DEFINITION

We are talking from pattern languages, design patterns and pattern collections. To make it clear what we mean with this terms we provide a short definition of each of them.

2.1 HCI Design Pattern

An HCI design pattern describes a recurring user interface design problem together with a proven solution. An HCI design pattern, in the following referred to as "pattern" or "design pattern", has a well defined form, which is dependent on the individual author's preferences. A pattern form should be used consistently across a pattern language or pattern collection. This makes it easier for pattern users to understand the problem, context, and solution of a pattern throughout a pattern collection/language. The pattern itself, when it is a part of a collection or a pattern language, may have references to other patterns.

2.2 Pattern Catalogue/Collection

Patterns stored in so-called DESIGN PATTERN CATALOGUES or COLLECTIONS are categorized to support faster navigation within the repository. However, they show almost no relationships among each other and thus do not form a fully interconnected system. The catalogue/collection contains several patterns that stand alone and have no connections to predecessor or successor patterns. Furthermore, such a collection usually does not completely cover a specific application domain.

2.3 Pattern Language

In contrast to a pattern catalogue / collection, a “pattern language” is a complete collection of patterns for a given family of design problems in a given domain. A pattern language describes problems by means of high-level design patterns, which are solved by low-level design patterns. The design patterns are connected through relationships, so that they constitute a network.

In a pattern language, the “words” are the patterns, while the connections between patterns represent the “rules of grammar” which are situated in the pattern itself. When words and rules of grammar are combined, a “sentence” is generated. Sentences can be built in many different forms when the rules are followed. So there is not only one path through a pattern language, it offers several possibilities to solve a design problem. A good example is “The Design of Sites” by van Duyne et al., a pattern language that allows designers to articulate an infinite variety of Web designs [24].

3. SURVEY RESULTS

3.1 Publication Year / Publication Medium

The cornerstone of design patterns as a tool of knowledge was laid back in the late 1970s when the mathematician and architect Christopher Alexander published several books [1, 3, 2] in which he proposed the concept of design patterns and pattern languages. Ward Cunningham and Kent Beck have adopted this principle to object-oriented programming (OOP) and user interface (UI) implementation in 1987 [4, 19]. They presented five patterns for designing window-based user interfaces in Smalltalk.

In Human-Computer Interaction, the start of the design pattern era was when Coram et al. [7] published the first design patterns of a pattern language for user-centered interface design (see Fig. 1). The objective of this design pattern language was to provide high-level patterns with which user interface designers could build graphical user interfaces which are pleasurable and productive to use.

In the following years several other pattern collections and pattern languages were published. The publishing activity in scientific papers recently slowed down because of the publication of four seminal books in the HCI design pattern community. The first in 2001 [5], two books in 2003 [10, 24] and one in 2005 [21]. At the same time, pattern writers have focused on developing repositories on online platforms. Due to the hypermedia characteristics and the 24/7 availability of data on the Internet, it is much easier to reference to other patterns and disseminate patterns across the HCI community. A further benefit of online resources

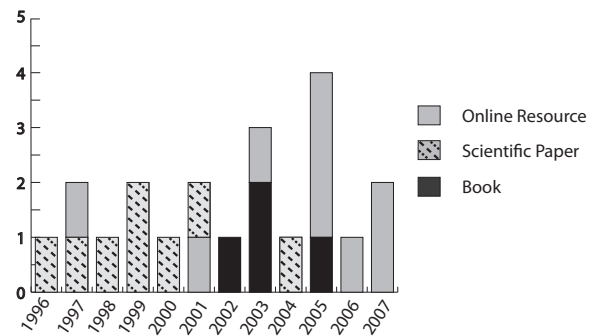


Figure 1: Year and Type of HCI Design Pattern Publications

is that users can contribute in writing and improving HCI design patterns using Web 2.0 features, for example, *van Welie's Online Repository* [29] and the *Yahoo!Design Pattern Library* [30] allow users to post comments on a design pattern, while it is not allowed to make changes directly to specific design patterns.

3.2 Arrangement of Design Patterns

Design patterns can be arranged in pattern collections or languages (cf. Section 2). The latter connect design patterns to an interconnected network whereas collections do not.

Five pattern languages were published in books consisting of 10 to 90 interconnected design patterns, and three were published in scientific papers. Due to the spatial limitations of scientific papers, these pattern languages are composed of 8 to 9 design patterns.

Beside pattern languages, 15 design pattern collections were found. One was published in a book, five were published in scientific articles and 9 were published through online repositories. The book consists of 94 design patterns, the scientific papers of 5 to 45 patterns and the online resources of 18 to 130 patterns.

A reason for the predominance of books as a publication media for complete pattern languages – besides the volume aspect – may be the scientific incentive for the intensive and time consuming research necessary to discover and describe a pattern language. A disadvantage of casting patterns in books, however, is the updating process. It is difficult to add to or improve already published pattern languages, whereas the nature of design patterns mandates to update them on a regular basis since interaction techniques change over time due to the invention of new hardware and novel interaction methods. Therefore, several books come with companion websites where pattern language updates are published regularly (e.g., [21], [24]).

Many design pattern collections can be found online, because of the easy updating and dissemination of patterns to a large audience. Some of these resources are, however, not maintained very well - last updates have occurred years ago - perhaps a typical fate of short term academic projects. Beside these not maintained online collections there are a

Domain	Book	Scientific Paper	Online Resource
Web User Interface Design	2	2	7
User Interface Design (Desktop Applications)	2	5	1
Interactive Exhibits	-	1	-
Software Design (UI Related Programming)	-	1	-
Hypermedia Applications	-	1	-
Ubiquitous Computing	-	1	-

Table 1: Domain and Publication Medium of HCI Design Pattern Languages / Collections

few which are updated regularly. These are the *Little Spring Design – Mobile UI Design Resources* [13], *UI Patterns – User Interface Design Pattern Library* [23], *Welie.com – Patterns in Interaction Design* [29] and *Yahoo!Design Pattern Library* [30]

3.3 Device Categories for HCI Design Patterns

Several design patterns are applicable for specific hardware devices. Our research showed that most design patterns have been discovered for desktop and handheld/smartphone applications. Handheld/smartphone applications differ from desktop applications in the fact that the display and input devices available are smaller and cannot be used as easy as those from desktop applications. Therefore, different interaction methods are used.

In most cases it was not explicitly mentioned which design pattern applies to which platform. Normally, design patterns should be written in a platform independent manner, but hardware limitations as mentioned before force specific interaction methods. When no specific platform is mentioned to which a design pattern applies, the forces, the context, the problem, and the solution element of the design pattern constrain the platform where the pattern can be implemented. To make an HCI design pattern more effective, the author should mention the hardware platform in the head of the pattern.

3.4 HCI Design Pattern Domains

Each domain has its specific forces. Thus, forces should be resolved in a domain specific way. Therefore, HCI design patterns are written for a specific domain they can be applied to. Our survey showed that the predominant domains are web design and interface design for desktop applications. Web design patterns were almost exclusively published online, while interface design patterns for desktop applications mostly appear in scientific papers (see Table 1).

3.5 HCI Design Pattern Categorization

Patterns are grouped according to the basic idea they address. As shown in Table 1, prominent domains are Web user interface design and interface design for desktop applications. Table 2 and Table 3 are showing which informal categorizations design pattern authors have used to subdivide the design patterns in the respective domain. These categories are the most popular but should not be considered as a formal standard categorization in the field of HCI design patterns. Research approaches in categorizing

HCI design patterns are mentioned in the literature [8, 14, 26].

These categorization schemes differ in a few elements. The basis of the Web user interface design categorization (Table 3) is the interface design for desktop application categorization (Table 2) without the elements *Visibility*, *Data Representation* and *Natural Mapping* but enriched by *Site Genres*, *E-Commerce*, *Optimization* and *Accessibility*. Due to the nature of the Web design domain, these supplements are necessary to complete the solution space.

3.6 HCI Design Pattern Structure

The content elements of a design pattern can vary due to the preferences of a pattern author. Several authors are using the Alexandrian form to describe discovered patterns in a structured way. However, the Alexandrian form is not taken as the preferred structure in all design patterns and therefore many different pattern forms and content elements exists in the HCI design pattern domain.

We have analyzed the design patterns according to their structure and their content elements. Table 4 shows all discovered content elements and which author is using which content elements. The pattern structure was divided into 4 parts, namely the HEAD, the BODY, ADDITIONAL INFORMATION and REFERENCES. To understand each of the content elements described in Table 4 we have briefly described each of them below.

The minimal set of common content elements throughout all analyzed design patterns can be used as a basis of a “standard” HCI design pattern form. These are:

- HEAD
 - Pattern Name
 - Sensitizing Image
 - Short Problem / Content Description (Summary)
- BODY
 - Context
 - Forces
 - Solution
 - Example / Pattern Instance Gallery
- REFERENCES
 - Related Patterns

Category	Description
Visibility	How to design something so that the user knows immediately how to use it just by looking at it
Feedback	Describes how feedback should be generated if a task is being correctly or incompletely completed.
Natural Mapping	Creates a clear relationship between what the user wants to do and the mechanism for doing it.
Content Organization	Information architecture and application structure.
Navigation	How to get around efficiently in the application.
Layout	Shows how to layout application screens for a satisfying result.
Data Representation	Techniques how to represent large data sets.
Getting Input from User	Provides appropriate input methods.
Search	Describes how search methods can be incorporated into the application.
Accessibility	Techniques and methods to adopt the website for people with disabilities.

Table 2: Categorization of Design Patterns in the Interface Design for the Desktop Application Domain

Category	Description
Site Genres	Describes various site genres, e.g., News Site, Personal Homepage, Shopping Site, Information Site, etc.
E-Commerce	Shows methods which can be used to enrich websites with e-commerce functions.
Optimization	Technical advices to speed up the website.

Table 3: Additional Categories of Design Patterns in the Web User Interface Domain

These elements can be considered as a mandatory set of content elements of a well-defined HCI design pattern. There is enough information to understand the problem, context, and solution of the addressed design pattern. Beside these basic elements, authors should have the possibility to add their own elements to enrich the pattern's content with useful information for easier implementation of the pattern. The following resources are using the minimal set of content elements together with others:

- Ian Graham, "A Pattern Language of Web Usability", [10],
- Douglas van Duyne, "The Design of Sites", [24],
- Carol Stimmel, "Hold Me, Thrill Me, Kiss Me, Kill Me", [20], and
- "User Interface Design Patterns", [22].

Below we give a description of each of the content elements mentioned in Table 4.

3.6.1 HEAD

The HEAD or introduction paragraph of the pattern gives the pattern user a short overview of the problem which the pattern addresses. A short context description and an image which shows a successful solution to the addressed problem are placed at the top of the pattern structure. The image (or sensitizing image) ensures that the pattern is remembered more easily. The description of the problem and context

must be as short as possible while it must give as much information as possible to get a rough idea of the patterns problem space. Along with this information some metadata is also placed in the head of the pattern. Below there is a short description of each of the content elements which were found in the head of the analyzed HCI design patterns.

PATTERN NUMBER

The number uniquely identifies a pattern within a pattern language/collection. It is useful for referencing. An alphanumerical code is used when it should encode the categorization of the pattern as well.

PATTERN NAME

The pattern name is the reader's first "contact" to the pattern proper and - beside the sensitizing image - the most important cue to remember the pattern. Therefore, it should be chosen wisely to give a significant hint to the content of the pattern. A unique name should be used which is easy to remember and to use for unambiguous communication within design documents, meetings and other situations.

ALTERNATIVE PATTERN NAME

Also known as "AKA". Indicates the alternative names of a pattern.

RATING/RANKING

This item indicates how the author or the pattern users are rating the pattern. It should help pattern users to decide if they can use it without worries or should rather consider another pattern.

SENSITIZING IMAGE

“A picture is worth more than a thousand words”. Beside the name, the sensitizing image creates the user’s first impression of the design pattern. A good screenshot or - even better - sketch can help to grasp the idea behind the pattern’s solution much faster. If the solution has a dynamic character, a short animation may be appropriate to demonstrate it to the user.

SHORT PROBLEM/CONTENT/CONTEXT DESCRIPTION

This content element should give a short overview of the problem, content and/or the context of the pattern. For a more detailed description of the *problem* and/or *context*, the CONTEXT and the DETAILED PROBLEM DESCRIPTION content element in the BODY part of the pattern should be used.

AUTHOR NAME

The author name designates the contact person and writer of the pattern.

PATTERN CLASSIFICATION/GROUP

Pattern in languages/collections are grouped according to a common underlying idea. (see PATTERN NUMBER)

CREATION DATE

Shows when the pattern was first created.

LAST REVISION DATE

Together with CREATION DATE the LAST REVISION DATE content element is very useful to show when the pattern was revised. Ideally all revision dates are published so the user knows how often and when a pattern was updated since patterns tend to change over time due to the invention of new hardware and interaction methods.

HARDWARE

Shows on which device the solution of a design pattern can be implemented.

LEVEL

Indicates if it is a high-, medium- or low-level pattern. High-level patterns describe problems and solutions in a very abstract way. These patterns lead to more detailed patterns such as medium- and low-level patterns. Low-level patterns are the most detailed, describing e.g. the function of a certain interaction widget such as an action button.

3.6.2 BODY

After a user has decided to use a pattern, the BODY section offers the user detailed information about the problem, forces, context, solution, as well as many more additional information for a better understanding of the design idea addressed. The information provided in the BODY section extends the information of the HEAD section and is more detailed. The BODY of a pattern consists of the following (unordered) content elements.

CONTEXT

Beside the problem and the solution content element, the context is essential for the understanding of a design pattern. This element makes a pattern distinct from a style guide or guideline document. It shows designers the preconditions in which situation the problem and its solution occur, and thus

defines the applicability of a particular pattern.

DETAILED PROBLEM DESCRIPTION

This element describes the problem the pattern solves. A detailed analysis of the problem and background information is provided to clearly understand the design problem.

FORCES

This element discusses the forces and constraints relevant to the pattern and how they conflict and/or interact with each other. Forces help the user to better understand the problem and the connection to the context.

SOLUTION

The SOLUTION addresses the reuse of recurring design practices and how to resolve the forces discussed in the FORCES element. It is written in a way that the designer has an idea how to resolve those forces in an efficient way. But it is not like a guideline where you only need to follow the instructions step-by-step to get a solution. Therefore the designer’s creativity is necessary to produce a good solution.

RATIONALE

The RATIONALE element describes why the pattern works. It describes how and why the current pattern resolves its forces. It goes deep into the mechanisms which are used to get the forces into harmony. In other words, it is a proof of concept.

DIAGRAM

Sometimes diagrams or sketches are used to summarize the solution of a pattern. It is not a working instance of a pattern, but it rather gives the user another view of the solution and it supports the design decision of a UI engineer. When the solution has a time dimension, a storyboard may be a better tool to demonstrate the solution.

RESULTING CONTEXT

After resolving the forces of the current pattern, it builds a new context for other patterns. This content element discusses the resulting context and which patterns may be applied next.

EXAMPLES

Links and screenshots or working instances of the design pattern’s solution are presented in this content element. It is good practice to show many different solutions, to give the UI designer a better understanding of how to implement the pattern’s solution.

KNOWN USES

It gives information where to find good, already implemented solutions on different platforms.

COUNTER EXAMPLES

Shows bad design in the context of the current design pattern’s problem/solution approach. Usually a link and short description or screenshot to the faulty design is provided.

3.6.3 Additional Information

These elements do not fit in the two sections above, but enrich the pattern with more information. The elements

shown below are used by a few authors. Actually, there is no order of the content elements.

RELATED IDEAS / LITERATURE

This element comprises references to basic literature and/or ideas regarding the interaction mechanisms the design pattern describes.

REFERENCES TO IMPLEMENTATIONS

Authors provide direct links to good solutions. Because of the nature of a link, it is used in online resources. It supports the element **EXAMPLES** in the **BODY** part of the design pattern.

CODE EXAMPLES

Whereas code examples are very often used in software engineering patterns, in HCI they are not so popular because of the many different possibilities to implement an interaction mechanism. Sometimes there are code examples provided for better understanding of the pattern's solution. This content element is provided only in online resources.

ACCESSIBILITY

It shows how to extend the pattern solution in such a way that the application can be accessed by people with disabilities.

3.7 Pattern Relationships

References are essential when working with pattern languages. They must be incorporated into the design pattern structure. Authors must take care to link the patterns in the right way and order so that they can build a network. In pattern collections references to other patterns are not so essential because there are many pattern which are not connected to another pattern. References are also used to indicate which other patterns can be applied after having implemented a certain pattern.

During our survey it was interesting to find out that many authors as inter-pattern relationships used association and aggregation. Specialization was used by two authors.

A new relationship is also pointed out. It is named the "anti-association" connection. This connection is similar to the association but it references to an anti-pattern. An anti-pattern describes, in contrast to a design pattern, a problem statement with a bad solution. But only one author used an anti-pattern to show the user how not to solve a problem [20]. This is an interesting concept since it can show user interface designers common pitfalls to learn from.

REFERENCES IN TEXT

This relationship indicates if references to other design patterns are made within the **PROBLEM**, **FORCES**, **SOLUTION**, etc. content elements. The benefit of referencing design patterns in such a way is that when a problem occurs that the design pattern does not solve, the user can be guided to more appropriate patterns without having to search the design pattern for the **RELATED PATTERNS** content element.

RELATED PATTERNS

Unlike **REFERENCES IN TEXT**, no references were made in the design pattern itself. The **RELATED PATTERNS** content

element encapsulates all references to other patterns. Usually, it includes references to lower-level patterns.

SPECIALIZATION

Specialization of a design pattern means to add more attributes to it. The **SEARCH** pattern, for example, only provides a basic search mechanism. To extend the concept of this pattern a more specialized pattern is generated. It inherits the attributes from **SEARCH** and adds new ones to fulfill the purpose of advanced searching. This leads to the specialized pattern **ADVANCED SEARCH** (see Fig. 2).

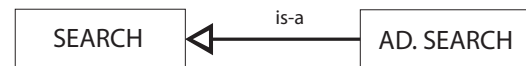


Figure 2: Design Pattern Specialization

AGGREGATION

When a pattern consists of more than one sub-pattern, an aggregation relationship is used to connect them. As an example, consider the **SHOPPING CART** pattern in Fig. 3. This pattern consists of many sub-patterns, like **LIST BUILDER**, **WIZARD**, etc. After applying these patterns in a way suggested by **SHOPPING CART**, the problem and forces of **SHOPPING CART** are solved.

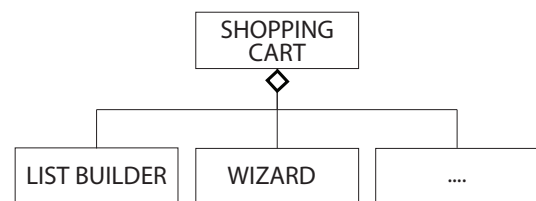


Figure 3: Design Pattern Aggregation

ASSOCIATION

An association (see Fig. 4) between design patterns is a unspecific connection between them. When referencing patterns, the words "related to" and "similar to" are often used to indicate an association with another design pattern.



Figure 4: Design Pattern Association

ANTI-ASSOCIATION

Anti-Association is similar to Association. It is a connection to an anti-pattern. It shows how a pattern should not be implemented. This is a good way to demonstrate common pitfalls and bad design solutions.

4. CONCLUSION

This survey shows in which way HCI design pattern authors are writing design patterns. Most HCI design patterns are

published through websites or Web portals. Basically a good idea, but it is also necessary to maintain the site to provide up-to-date information, since over time, a design pattern may change because of the invention of new interaction mechanisms. Therefore, patterns published online should be updated regularly. Only a few portals are maintaining their pattern collections, one reason being possibly the lack of tool support.

Structure and organization of patterns vary due to their authors' preferences. Thus, there is no consensus on how patterns should be formulated and categorized in order to provide appropriate information to produce good interface design. Many authors are using the Alexandrian form, possibly because it was the first form used to encapsulate design knowledge. So pattern authors have transferred the Alexandrian pattern structure to software engineering and then to the HCI domain. It seems necessary, however, to identify out the most important elements for HCI design patterns to better support the work of HCI designers and pattern authors, respectively. This survey has analyzed the most frequently used content elements. But to propose a unified pattern form for the HCI domain, workshops and discussions in the HCI community are necessary to develop an generally acceptable basic HCI design pattern structure.

Finally, during our research on freely accessible HCI design patterns it was interesting to find out that most patterns are written for Web design issues. In the last years Web design was one of the hottest issues in UI design, where UI designers have adapted many principles of ordinary desktop interface design to the special needs of websites.

5. FUTURE WORK

With the results of this survey and a study of categorization schemes in usability literature we will be able to propose a classification scheme or a taxonomy for HCI design patterns. It should help pattern authors to identify overlapping design patterns and define patterns according to their problem group or scope. For pattern users a taxonomy is useful when searching for solutions to a specific design problem and alternative solutions can be found more easily.

6. ACKNOWLEDGMENTS

We would like to thank Prof. Dinesha from the Indian Institute of Technology Bombay, for his suggestions and discussions on this topic which helped us to improve the quality of this paper.

7. REFERENCES

- [1] C. Alexander. *The Oregon Experiments*. Oxford University Press, 1975.
- [2] C. Alexander. *The Timeless Way of Building*. Oxford University Press, 1979.
- [3] C. Alexander, S. Ishikawa, and M. Silverstein. *A Pattern Language*, volume 2. Oxford University Press, New York, 1977.
- [4] K. Beck and W. Cunningham. Using Pattern Languages for Object-Oriented Programs. In *OOPSLA 87 workshop on the Specification and Design for Object-Oriented Programming*, 1987.
- [5] J. Borchers. *A Pattern Approach to Interactive Design*. Software Design Patterns. Wiley, 2001.
- [6] E. S. Chung, J. I. Hong, J. Lin, M. K. Prabaker, J. A. Landay, and A. L. Liu. Development and Evaluation of Emerging Design Patterns for Ubiquitous Computing. In *DIS '04: Proceedings of the 5th conference on Designing interactive systems*, pages 233–242, New York, NY, USA, 2004. ACM.
- [7] T. Coram and J. Lee. Experiences - A Pattern Language for User Interface Design. 1996. Available at: <http://www.maplefish.com/todd/papers/Experiences.html>.
- [8] S. Fincher and P. Windsor. Why patterns are not enough:some suggestions concerning an organising principle for patterns of UI design. In *CHI '2000 Workshop on Pattern Languages for Interaction Design: Building Momentum*, 2000. <http://www.cs.kent.ac.uk/people/staff/saf/patterns/chi00.pdf>.
- [9] A. Garrido, G. Rossi, and D. Schwabe. Pattern Systems for Hypermedia. In *Pattern Languages of Programming 1997*, 1997.
- [10] I. Graham. *A Pattern Language of Web Usability*. Addison-Wesley, 2003.
- [11] Hypermedia Design Patterns Repository. Online. Available at: <http://www.designpattern.lu.unisi.ch/index.htm>, Accessed on December 27, 2009.
- [12] C. Kruschitz. XPLML: a HCI pattern formalizing and unifying approach. In *CHI EA '09: Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, pages 4117–4122, New York, NY, USA, 2009. ACM.
- [13] Little Spring Design - Mobile UI Design Resources. Online. Available at: http://patterns.littlespringsdesign.com/index.php/Main_Page, Accessed on December 27, 2009.
- [14] M. J. Mahemoff and L. J. Johnston. Pattern Languages of Usability: An Investigation of Alternative Approaches. In J. Tanaka, editor, *APCHI 98 Proceedings*, pages 25–31. IEEE Computer Society, Los Alamitos, CA, 1998.
- [15] M. J. Mahemoff and L. J. Johnston. The Planet Pattern Language for Software Internationalisation. In *Pattern Languages of Programs 1999 Proceedings*, Monticello, IL, 1999.
- [16] PatternCube - Design Pattern Portal. Online. Available at:www.patterncube.com, Accessed on May, 2008.
- [17] Patterns for Personal Web Sites. Online. Available at:<http://www.rdrop.com/~half/Creations/Writings/Web.patterns/index.html>, Accessed on December 27, 2009.
- [18] K. Perzel and D. Kane. Usability Patterns for Applications on the World Wide Web. In *Pattern Languages of Program Design 1999 Proceedings*, 1999.
- [19] R. Smith. Panel on Design Methodology. In *OOPSLA '87: Addendum to the proceedings on Object-oriented programming systems, languages and applications (Addendum)*, pages 91–95, New York, NY, USA, 1987. ACM.
- [20] C. L. Stimmel. Hold Me, Thrill Me, Kiss Me, Kill Me: Patterns for Developing Effective Concept Prototypes.

In *Pattern Languages of Program Design 1999 Proceedings*, Monticello, IL, 1999.

- [21] J. Tidwell. *Designing Interfaces*. O'Reilly, 2005.
- [22] UI Patterns - User Interface Design Pattern Library. Online. Available at: <http://ui-patterns.com/>, Accessed on December 27, 2009.
- [23] User Interface Design Patterns. Online. Available at: <http://www.cs.helsinki.fi/u/salaakso/patterns/>, Accessed on December 27, 2009.
- [24] D. K. van Duyne, J. A. Landay, and J. I. Hong. *The Design of Sites : Patterns, Principles, and Processes for crafting a Customer-Centered Web Experience*. Addison-Wesley, 2003. Website <http://www.thedesignofsites.com/>.
- [25] M. van Welie and H. Traettenberg. Interaction Patterns in User Interfaces. In *7th. Pattern Languages of Programs Conference*, 2000.
- [26] M. van Welie and G. C. van der Veer. Pattern Languages in Interaction Design: Structure and Organization. In *Human Computer Interaction - INTERACT 2003*, pages 527–534. IOS Press, 2003.
- [27] W. C. Wake. Patterns for Interactive Applications. In *Pattern Languages of Programm Design 1998 Proceedings*, 1998. Available at: http://jerry.cs.uiuc.edu/~plop/plop98/final_submissions/P44.pdf.
- [28] Web Patterns - A UC Berkeley Resource for Building User Interfaces. Online. Available at: http://groups.ischool.berkeley.edu/ui_designpatterns/webpatterns2/webpatterns/home.php, Accessed on May, 2008.
- [29] Welie.com - Patterns in Interaction Design. Online. Available at: <http://www.welie.com>, Accessed on December 27, 2009.
- [30] Yahoo! Design Pattern Library. Online. Available at: <http://developer.yahoo.com/ypatterns/>, Accessed on December 27, 2009.

	BOOKS				SCIENTIFIC PAPERS								ONLINE RESOURCES								
HEAD	[10]	[24]	[5]	[21]	[25]	[18]	[7]	[27]	[9]	[20]	[15]	[6]	[29]	[30]	[11]	[16]	[23]	[17]	[13]	[22]	[28]
Pattern Number	✓	✓	✓	✓	-	-	-	-	✓	-	✓	✓	-	-	-	-	-	-	-	-	-
Pattern Name	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Alternative Pattern Name	✓	-	-	-	✓	-	-	-	-	-	-	-	✓	-	✓	-	-	-	-	-	✓
Rating/Ranking	✓	-	-	-	✓	-	-	-	-	-	-	-	✓	-	✓	-	-	-	-	-	✓
Sensitizing Image	✓	✓	✓	✓	-	-	✓	-	-	✓	-	-	-	✓	-	-	-	-	-	✓	✓
Problem-Context Summary	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	-	✓	✓	-	✓	✓	✓	✓	✓
Author Name	✓	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	✓	✓	-	-	-	✓
Pattern Classification	-	✓	-	-	✓	✓	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-
Creation Date	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-
Last Revision	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	✓	✓	-	-	-	✓	-
Hardware	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-
Pattern Level	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BODY																					
Context	✓	✓	-	✓	✓	-	-	✓	-	✓	-	✓	✓	-	-	✓	✓	-	✓	✓	✓
Problem Description	✓	✓	✓	-	-	-	✓	-	-	-	-	✓	-	-	-	-	-	-	-	-	-
Forces	✓	✓	✓	-	-	✓	✓	✓	✓	✓	✓	-	-	-	✓	-	-	-	-	-	-
Solution	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓
Rationale	-	-	-	✓	✓	✓	✓	-	-	✓	✓	-	✓	✓	-	✓	-	✓	✓	✓	-
Diagram	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Resulting Context	✓	✓	✓	-	-	✓	-	-	-	✓	✓	-	-	-	✓	-	-	-	-	-	-
Examples	-	-	✓	✓	-	✓	-	✓	-	-	✓	-	✓	✓	✓	✓	✓	-	✓	✓	✓
Known Uses	-	-	-	-	✓	-	-	-	✓	-	-	-	-	-	✓	-	-	-	-	-	-
Counter Examples	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-
ADD. INFO.																					
Related Ideas / Literature	-	-	-	-	-	-	-	✓	-	-	✓	✓	✓	-	-	-	-	-	-	-	-
Ref. to Implement.	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-
Code Examples	-	-	-	-	-	-	-	-	-	-	-	-	✓	✓	-	-	-	-	✓	-	✓
Accessibility	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-
REFEREN.																					
Ref. in Text	✓	✓	-	?	-	-	-	-	-	✓	✓	-	✓	-	-	-	-	-	✓	✓	-
Related Patterns	-	-	✓	-	✓	✓	✓	✓	✓	-	-	✓	-	✓	✓	✓	✓	✓	-	-	-
TYPES OF REF.																					
Specialization	-	-	-	-	-	-	-	-	-	-	-	✓	✓	-	-	-	-	-	-	-	-
Aggregation	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	-	-	-	-	-	-	-	-
Association	✓	✓	-	✓	-	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	-	-	✓	✓	✓	-
Anti-association	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-

Table 4: HCI Design Pattern Content Elements