

Building a Pattern Repository: Benefitting from the Open, Lightweight, and Participative Nature of Wikis

Michael Weiss

Carleton University, Ottawa, Canada
weiss@sce.carleton.ca

Aliaksandr Birukou

University of Trento, Trento, Italy
birukou@dit.unitn.it

Abstract

The growing number of patterns creates the problem of finding the right pattern for a given design problem. As one reply to this problem, several pattern collections have been captured in the form of pattern repositories. In this paper we motivate the use of wiki for building a pattern repository, namely their openness, lightweight nature and participative architecture. These properties allow the formal structure of the repository to emerge in response to needs of pattern authors and users. At the same time, current wiki frameworks, built on extensible architectures provide a rich set of features that facilitate the rapid implementation of a repository. It appears that such a lightweight approach outweighs the benefits of a more customized, special-purpose framework – at least initially, until there is a consensus on the repository structure, and a community of users has evolved.

1. Introduction

Patterns are a means for disseminating best design practices. They distill instances of tacit experience into explicit design knowledge that cuts across specific uses, a process described in detail in [1]. Once made explicit, these patterns become part of the vocabulary that developers use to describe their designs. However, the knowledge of applying the patterns (i.e., selecting, using and reflecting on pattern use) becomes itself new tacit knowledge [1]. This knowledge can be codified in the form of a pattern language, a set of patterns that link to each other, which guides developers through the application of the patterns in the language.

When developers apply a pattern, they benefit from the experience of the pattern's author. However, the growing number of patterns creates a problem of find-

ing the right pattern for a given design problem.¹ As one reply to this problem, several pattern collections have been captured in the form of pattern repositories. In addition to links between patterns, repositories provide sophisticated search mechanisms, often built around a meta-model of the patterns and the pattern language, as well as sometimes a model of the application domain. Examples of repositories include:

- Portland Pattern Repository
<http://c2.com/ppr>
- Interaction Design Patterns Library
<http://www.welie.com/patterns>
- Computer-Mediated Interaction Patterns
<http://moskau.pi6.fernuni-hagen.de:3000>

What issues should the design of a repository address? What are the needs of pattern authors and pattern users? To what degree, for instance, should the structure of the repository be predesigned? Social issues also need to be addressed, besides such technical issues, such as how can we create and sustain a community of users around the repository?

This paper describes the initial results of an effort initiated at a EuroPLoP 2007 Focus Group on Pattern Repositories [3] to build a pattern repository that, rather than formalizing the repository structure upfront, allows it to emerge from use. Also, our focus was on building functionality that added value to users. The effort was guided by the technical and social issues outlined above, and by the desire to forge a community of developers of pattern repository frameworks. In the

¹E.g., The Patterns Almanac [2], an effort to index and link the patterns documented in the literature up to the year 2000, lists more than 1200 patterns. Between 1994 and 1999, on average, 200 patterns were published each year, suggesting there will be at least around 2800 patterns by this year (2007).

following, we motivate why we used a wiki as the initial platform for our repository. We then present details from our initial prototype and conclude the paper.

2. Requirements

Our desire was to create a system that in a short time provided a minimal set of value-added features:

- Authors capture pattlets and their relationships
- Authors and non-authors (we view them as co-producers of the content) can “tag” patterns
- Patterns are organized by pattern collections and into pattern languages via their relationships
- Pattern descriptions contain metadata, rather than the full pattern description (less effort for authors, easier to scan, kind of information not easily accessible if “drowned out” by a full description)

3. Wiki as Repository Platform

3.1 Technical Benefits

- Openness (i.e., the pattern domain model is not pre-defined, but emerges from use)
- Many complex features for “free” through a plug-in architecture (i.e., can add features as extensions such as tagging, access control, inter-wiki links, ...)
- Access control (i.e., can log user access)

3.2 Social Benefits

- Lightweight syntax (i.e., no steep learning curve for an idiosyncratic system)
- Participative architecture (i.e., both authors and non-authors can contribute to pattern descriptions)

4. Prototype

Figure 1 shows the navigation interface of the pattern repository.² Patterns can be browsed by collection or pattern name, as well as by tag. Users categorize the patterns by tagging them. The tag cloud shows the frequency of each tag across the patterns.

An example pattern view is shown in Figure 2. It consists of pattlet, links to other patterns, and tags.

5. Conclusion

Design of the pattern repository was guided by three principles: openness, lightweight nature, and participa-

²<http://www.patternforge.net/wiki>

You can also [Browse the Patterns](#) or [Pattern Collections](#) in this pattern repository.

Patterns can be tagged using categories. Browse patterns by tags:



Figure 1. Navigation of the pattern repository

In Bed with the Enemy

How can a company avoid being late introducing new products and, on top, bearing the risk of developing the wrong type of product? Therefore, pool complementary assets from other companies.

- A “not invented here” culture is a common obstacle to collaboration. Often, this can be overcome by establishing an incentive system to [Reward Collaboration](#).
- Internally developed product ideas may not receive the attention they deserve. Collaboration can be an enabler for these ideas to be adopted via an [Endorsement from Outside](#).
- [Innovation Happens Elsewhere](#) (Goldman and Gabriel, 2005) describes how the results of external product development can be reintegrated into a company’s own products.

Categories: [Pattern](#) | [Collaboration](#) | [Business](#) | [Innovation](#)

Figure 2. Pattern view: pattlet, relationships, and tags

tive architecture. All of these are well supported by using a wiki. To date the repository has been used to capture a subset of the patterns from EuroPLoP 2007 as entered by the authors, and we will receive contributions from the VikingPLoP 2007 participants.

Acknowledgement

The following people contributed to the ideas behind the repository: Andreas Fiesser, Axel Schmolitzky, Birgit Zimmermann, Christian Graf, Christian Kohls, Diethelm Bienhaus, Leon Welicki, Marty Kauhanen, Miłos Kudelka, Ondrej Lehecka, and Till Schümmer, and the discussants at the focus group (notably Allan Kelly, Andreas Rueping and Frank Buschmann).

References

- [1] May, D., and Taylor, P., Knowledge Management with Patterns, CACM, 46(7), 94-99, 2003.
- [2] Rising, L., *The Patterns Almanac*, Addison-Wesley, 2000.
- [3] Weiss, M., Birukou, A., and Giorgini, P., EuroPLoP 2007 Focus Group on Pattern Repositories, CFP, 2007, http://www.patternforge.net/wiki/index.php?title=EuroPLoP_2007_Focus_Group.