Third International Workshop on Sharing and Reusing Architectural Knowledge (SHARK 2008)

Paris Avgeriou
University of Groningen,
The Netherlands
paris@cs.rug.nl

Patricia Lago
VU University Amsterdam,
The Netherlands
patricia@cs.vu.nl

Philippe Kruchten
University of British Columbia,
Canada
pbk@ece.ubc.ca

Abstract

The shift of the software architecture community towards architectural knowledge has brought along some promising research directions. In this workshop we discuss the issues that lead to the application of architectural knowledge in research and industrial practice; ongoing research and new ideas to advance the field. In its previous editions we examined the state of the art and practice, future challenges and trends. This third edition will discuss, among others, architectural knowledge as perceived by different research communities, including requirements engineering, service-oriented computing and international standardization.

Categories and Subject Descriptors

D.2.11 [**Software Architectures**]

General Terms:

Design, Documentation, Standardization

Keywords:

Architectural Knowledge

1. Theme and goals

The Architecture Knowledge (AK) community is comprised of both researchers and industrial practitioners that are involved in a wide variety of fields, disciplines and application domains [1]. The SHARK workshop is a meeting place for this community, which has grown and matured over the last three years. In the first year, the workshop explored the current state of the art and practice in the field [2], while in the second year, SHARK dealt with a research agenda for the future [3].

This year the focus is on ongoing research & practices and emerging approaches of AK in a broader context. The workshop aims to bring together researchers and practitioners of both the software architecture field, and other communities working on related fields, to provide a more interdisciplinary and multi-faceted perspective on the definition of AK and on the mechanisms to use it. This way we hope that people from different fields with a common interest on AK can join forces and help to explore the problem-solution space and shape the AK research agenda.

The subject of architectural knowledge is truly a multi-disciplinary domain across software engineering, software architecture and knowledge management. Several of the accepted papers represent the viewpoint of other fields and application domains, e.g. embedded systems, SOAs and SPLs. In order to bring further insight from other disciplines, keynote speakers have been invited to provide the AK perspective of Requirements Engineering and IEEE/ISO standardization of architecture documentation.

ICSE'08, May 10–18, 2008, Leipzig, Germany.
ACM 978-1-60558-079-1/08/05.

2. Accepted Papers

We selected 13 papers to be presented in the workshop [4] that we believe will help stimulate discussion and further research. To give a taste of the current state of the art and practice in this domain, we present here short summaries of the accepted papers.

Nakagawa and Maldonado propose a set of UML views combined with domain-specific ontologies to describe reference architectures. A case study applying the views to the testing domain shows how these views can be used to drive the development of a testing tool for architecture compliance. *Espinoza et al.* focus on the limitations of a de-facto standard UML profile for embedded systems' modeling. They discuss how to help Embedded Systems architects to formally evaluate architecture tradeoffs. Design decisions and rationale in the form of analytical advices, trade-off parameters and annotations about non-functional properties are used in a model-driven engineering approach.

Henrickson *et al.* propose an approach to capture rationale in Software Product Line engineering, and tie it to the PL architecture as well as the modeling environment. For this purpose they propose the change set concept to capture rationale on three levels: individual architectural elements within a change set, the change set itself (to document the composition reasoning), and to relationships for product selection. This approach aims to achieve better support for traceability and product derivation.

Mattmann *et al.* focus on the knowledge about architecture connectors as first-class elements in systems handling highly voluminous data transfers. They present a framework made of data-intensive software connectors (DISCO) to support the architect's decision-making process. The framework captures AK in connectors profiles, and uses distribution scenarios to guide the architect in deciding on the most appropriate connectors.

Le Goaer *et al.* deal with the evolution of component-based architectures by capturing the necessary architectural knowledge in so-called 'evolution styles'. These are organized in catalogs and described in terms of a header publishing the evolution interface and behavior, and the competence separating the realizing implementation. A prototypical implementation of an 'evolution

shelf' supporting classification, storage and retrieval of evolution styles is discussed.

Falessi *et al.* present an empirical study regarding the impact of documenting design decisions and rationale on the quality of the results of the architecting process. They confirm that different types of architecture decision-making tasks require to document significantly different types of architectural knowledge elements, and further show which types of knowledge are considered "valuable" for which use cases. Another form of empirical study is the systematic literature review, conducted by de *Boer and Farenhorst.* They examine how architectural knowledge is defined and how the different definitions in use are related. In this way, it provides an overview of how AK is currently addressed and perceived by the community.

Erfanian & Shams Aliee define two ontologies to enrich ATAM architecture assessments with architectural knowledge. Based on the main knowledge entities used in ATAM, the first ontology codifies the semantics between tactics and quality attributes, and the second ontology builds upon the first to document the architecture while assessments are in progress. This way relevant architecture design decisions are made explicit and reused to guide evaluations.

Gu & Lago highlight the importance of process decisions in the area of Service Engineering and SOA, and discuss how reference models aimed at capturing AK can be reused for this emerging discipline.

M. Ali Babar introduces a way to support AK sharing in the emerging field of distributed and global IT environments. He highlights the coming of age of traditional workflow-based tools and envisages virtual communities of architects collaborating through work spaces governed by flexible rules and adapting to teamwork dynamics. From a different angle, Clerc addresses the need for solutions for managing architecture knowledge in global distributed teams. He presents a study of the solutions already proposed in the requirements engineering field, and translates the relevant ones into architectural knowledge management strategies. An initial industrial validation shows both the applicability and the contribution of such approach.

Zdun *et al.* propose an inexpensive way to document architecture knowledge through an architecting method that focuses on applying patterns and modeling them through primitives. The method results in the explicit documentation of a number of significant design decisions in the architecture models, and resolves possible inconsistencies between multiple architectural. The paper demonstrates this approach for two architectural views, Component-and-Connector and Process Flow, in the context of the process-driven SOA domain. Also in the patterns realm, *Bortis & van der Hoek* propose the concept of pre-patterns as a means to document architectural knowledge. They argue that pre-patterns are a closer realization of Christopher Alexander's original approach of patterns in Urban Architecture rather than the Object-Oriented Design Patterns. Therefore they propose pre-patterns as an effective means of conveying important design knowledge.

References

[1] Lago, P.; Avgeriou, P.; Capilla, R. & Kruchten, P. Wishes and Boundaries for a Software Architecture Knowledge Community WICSA '08: Proceedings of the Seventh Working IEEE/IFIP Conference on Software Architecture (WICSA 2008), IEEE Computer Society, 2008, 271-274

[2] Lago, P. & Avgeriou, P. First workshop on sharing and reusing architectural knowledge SIGSOFT Softw. Eng. Notes, ACM Press, pp. 32-36, 2006, 31.

[3] Avgeriou, P.; Kruchten, P.; Lago, P.; Grisham, P. & Perry, D. Architectural knowledge and rationale: issues, trends, challenges SIGSOFT Softw. Eng. Notes, ACM Press, 2007, 32, pp. 41-46

[4] SHARK web site: http://www.cs.rug.nl/~paris/SHARK2008