Diagrammatic Modeling of Architectural Decisions

Andrzej Zalewski and Marcin Ludzia

Warsaw University of Technology, Institute of Automatic Control and Computational Engineering, Warsaw, Poland
a.zalewski@ia.pw.edu.pl

Abstract. The paper presents a semi formal model of architectural decisions referred to as Maps of Architectural Decisions (MAD). In a form of a diagram they represent the most important components of architectural decisions (concerns, possible choices, constraints etc.) as well as logic of architectural decision making, i.e. dependencies between architectural decisions. This increases the level of formalism of the architectural decisions documentation, improves its readability and makes architectural knowledge gathered during the decision making process easier to comprehend, share and maintain.

1 Introduction

Modeling of architectural decisions is a focal issue of software architecture researchers. According to the classical paper by [1] they should include the following attributes: addressed issue, considered decision variants (positions), requirements, constraints, decision made, rationale (argument), implications. This provides a structured form of textual documentation of architectural decisions. The drawbacks of textual documentation have already been fully investigated in the software engineering discipline: it is error prone thus often inconsistent and ambiguous, difficult to analyse and verify, inefficient in presenting complex concepts.

In the software engineering discipline, such drawbacks have often been resolved by increasing the level of model's formality according to the scheme: textual (informal) – semi-formal (diagrammatical) – formal (mathematical) model. Increasing the level of formality of the models of architectural decisions seems to be an important research challenge. The most important developments towards further formalization of architectural decision models were so far:

– A tool called 'Archium' supporting architectural decision making has been presented in [3]. It concentrates on documenting the decision making process: the decisions are modeled as text records, however, their dependencies are modeled in the form of dependency graph. The tool supports traceability, basic completeness verification, detection of superfluous decisions;
– Knowledge management techniques based on ontologies have been used for managing architectural decisions in [2], however, these ideas have not been fully prover in practice yet;

R. Morrison, D. Balasubramaniam, and K. Falkner (Eds.): ECSA 2008, LNCS 5292, pp. 350–353, 2008.

– Extension of Tyree's templates to support decision makers collaboration and knowledge reuse – proposed in [4];
– Attribute decision graph presented in [5] in fact models the process of making architectural decisions driven by satisfying predefined attributes.

Maps of Architectural Decisions is another approach to the challenge of formalizing architectural decisions.

2 MAD – Models and Notation

MAD consists of two basic models: Architectural Decision Relationship Diagram (ADRD) and Architectural Decision Problem Map (ADPM), for an example – see fig. 2. The ADRD represents the set of identified architectural problems (concerns) and dependencies between them. The ADPM diagram is used to model all the important details of a given architectural decision problem and consists of the following objects: Architectural Decision Problem (ADP), Decision Variant (DV), Constraint (DC), Requirement (DR), Decision maker/Stakeholder and Connector (Constraint-Variant Connector or Variant-Problem Connector). The summary of MAD notation has been shown in fig. 1.

Each of the objects in ADRD or ADPM is additionally characterized with a set of attributes (they have been omitted due to the limit of space).

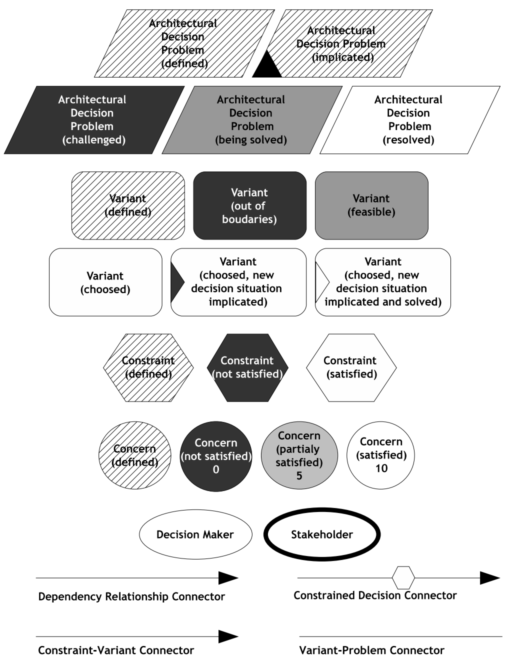

Fig. 1. Maps of Architectural Decisions – notation summary

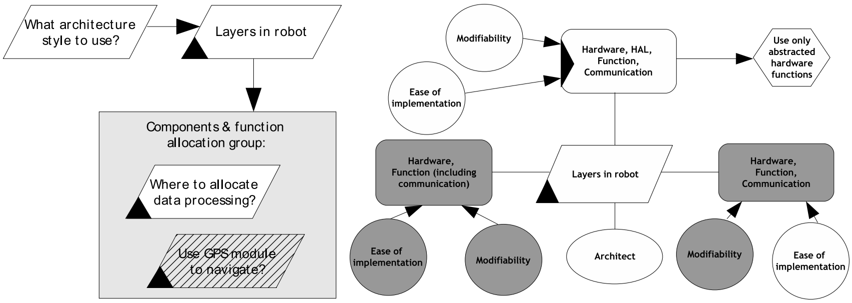

Fig. 2. MAD models for architectural decisions concerning architecting robot control system

The application of MAD has been illustrated on an example of robot control system presented in [6] – see fig. 2. We can comprehend from the ADRD, that there were three decision problems resolved so far and the fourth decision problem was added.

Each of the architectural decision problems can be modelled in detail with ADPM. An example decision problem is shown in fig. 2 (to the right). The central object of this diagram is an ADP object. These objects should first be defined in ADRD. The model of architectural decision problem consists of i) a set of considered solution variants modelled with DV objects attached to the ADP object; ii) Constraints and Requirements – instantiated from the dictionary of constraints and requirements (each Decision Variant should be assigned its own Constraints/Requirement set of instances, the constraints and requirements objects should be linked to Decision Variant by Constraint and Requirement Connector (CRC) with an arrowhead pointing the variant object), iii) Stakeholders involved in decision making process represented by an Decision Maker/Stakeholder object.

Architects evaluate all the Constraints and Requirements for each of the Decision Variants. The constraints and requirements may turn out to be satisfied, not satisfied or partially satisfied (requirements only). The variants, that do not satisfy all the constraints are marked as "Out of boundaries" and excluded as from the set of potential choices in the decision making process.

Variants meeting the requirements and satisfying the constraints should be marked as "Feasible". Architect chooses one of the "Feasible" variants and marks it as "Chosen". If chosen DV necessitates making some other decision, it is reflected as a triangle on the left side of a DV object. Additionally, architect should consider if there are any new constraint introduced by the decision. If so, the constraint should be defined in a diagram and linked with a variant object by CRC with an arrowhead directed to the Constraint object. The constraint will be automatically added to the constraints dictionary.

The entire architecture design is finished and properly documented, when all defined decision problems are described by ADPM models and all the decisions have been made and none of them is in challenged state (does not require reconsidering because of some other decision, which has been made).

3 Conclusion Further Research

The paper has been devoted to the presentation of a novel approach to the modeling of architectural decisions and the process of architectural decision making. The MAD models can be perceived as a tool for "mind-mapping" architectural decisions. Their unique feature is that they document the internal structure of architectural decisions in the form of a diagram. The notation assists both making an individual architectural decision – depicting all the important components on a single diagram – as well as documenting the whole process of architectural decision making – showing the dependencies between architectural decisions in the form of a diagram. As the level of formalism has been increased an automated checking similar to those described in [3] should be possible to implement.

Further research should include:

- interfacing the MAD and traditional modeling approaches (views, ADL's);
- predefining the dictionaries of the attributes of diagram's objects;
- providing formal definitions of diagrammatic models and their semantic;,
- defining the desired properties of the graphical models (consistency, completeness, others) and methods of their analysis;
- developing software tools supporting architecting with MAD.

References

1. Tyree, J., Akerman, A.: Architecture Decisions: Demystifying Architecture. IEEE Software (March/April 2005)
2. Kruchten, P.: An Ontology of Architectural Design Decisions in Software-Intensive Systems. In: Proc. of 2nd Groningen Workshop on Software Variability, Rijksuniversiteit Groningen, pp. 54–61 (2004)
3. Jansen, A., et al.: Tool Support for Architectural Decisions. In: Proceedings of the Working IEEE/IFIP Conference on Software Architecture (WICSA 2007). IEEE, Los Alamitos (2007)
4. Zimmermann, O., et al.: Reusable architectural decision models for enterprise application development. In: Overhage, S., Szyperski, C.A., Reussner, R., Stafford, J.A. (eds.) QoSA 2007. LNCS, vol. 4880, pp. 15–32. Springer, Heidelberg (2008)
5. Schwanke, R.W.: Layers, Decisions, Patterns, Styles, and Architectures. In: Proceedings of the Working IEEE/IFIP Conference on Software Architecture (WICSA 2001), pp. 137–147. IEEE, Los Alamitos (2001)
6. Gueorguiev, A., et al.: Design, Architecture and Control of a Mobile Site-Modeling Robot. In: Proceedings of the IEEE 2000 International Conference of Robotic and Automation, San Francisco. IEEE, Los Alamitos (2000)