Message from the QRASA 2016 Chairs

Welcome to the workshop on Qualitative Reasoning about Software Architecture (QRASA 2016), held at WICSA&CompArch 2016 in Venice, Italy. The quality properties that software systems aim to satisfy are key to the success of the systems. For many quality properties, methods and techniques have been developed for assessing these properties, such as performance, reliability, availability and some more. Several important architectural properties elude such quantitative assessment techniques, notably maintainability, modifiability, portability, extensibility, and interoperability. Having approaches for better assessing these properties would help in various aspects of managing software development, one particular application being managing technical debt.

A lot of work exists in the area of metrics for maintainability, for example. However, metrics suffer from several drawbacks. First, it may be unclear whether that metric is valid, i.e. whether it really measures the quality property of interest. Second, it remains difficult to interpret metrics (‘What does a maintainability of 4.5 mean?’). Third, it may be too expensive to collect the data for computing the metric (e.g. building a fully-fledged reliability model might not be cost-efficient for many systems). Thus, quantification is not always a good choice. Qualitative approaches could be a complementary approach for (automatically) making sense of this quantitative data.

In this workshop, we aimed to bring together researchers and practitioners who are interested in discussing, investigating and creating qualitative approaches for assessing and in general reasoning about these architectural properties. These proceedings contain four papers that our program committee selected for the workshop and that were presented at the workshop. Additionally, we had one invited presentation.

Leo Pruijt presented his and his co-workers experience on rule-type-based reasoning on architecture violations in an industrial project. In this project, they checked the conformance of a semantically rich modular architecture, which contains different types of conformance rules on different levels of abstraction, such as layers, subsystems, and components) against a code-base. As a result, they found that considering different rule types aids reasoning on the severity of the detected violations (first paper in these proceedings).

Nadeem Abbas presented his and his co-workers work of enhancing an architectural reasoning framework (eARF) by a reasoning mechanism that compares the design with requirements. As an input, quality attribute scenarios are expressed as networks of timed automata, so that properties can be checked (third paper in these proceedings).

In her invited talk titled “A Framework for Decision-Making of Architectural Refactoring”, Terese Besker presented a framework that shall help the architect to identify parts of the system to refactor in order to reduce architectural debt. In particular, she asked how software architects reason on what to refactor and how this reasoning could be supported by a framework.

In her presentation, Maryam Razavian and her co-workers proposed a reflective approach to software design decision making. Maryam suggests applying the Two-Minds-model from design thinking to architectural design and presents what kind of reflective questions software architects could ask. After an experiment with students, initial findings suggest that making students ask reflective questions helps them to challenge decisions in a more structured way (second paper in the proceedings).
Finally, Gianantonio Me presented his and his co-workers initial results on a lightweight theory to explain the interaction of patterns/styles and quality attributes. They base their theory on a systematic survey of a pilot set of primary studies (fourth paper in the proceedings). Following his presentation, the audience discussed that the context of a software system also strongly influences how a pattern or a style affects quality attributes. Consequently, to predict the effect of introducing a pattern or style, the context, such as the workload of the system, needs to be taken into account. For quality attributes that can be quantitatively assessed, such as performance and reliability, prediction techniques exist that support parametrized models and take context into account.

In the afternoon session, workshop participants brainstormed on what are the questions we should be looking at when considering qualitative reasoning about software architecture. We grouped the emerging questions into the following themes: Context, modelling, knowledge and reuse, metrics, the process of reasoning, and use cases and practicality of qualitative reasoning. The detailed discussion notes of this session can be found on the workshop web site at http://qrasa.ipd.kit.edu/results/.

Putting together QRASA 2016 was a team effort. We thank the authors for providing the content of the program. We are grateful to the program committee, who worked very hard in reviewing papers and providing feedback to authors. We also thank the WICSA&CompArch’16 organization team for their excellent work in coordinating the federated events and making possible a great conference.

Michel Chaudron, Bedir Tekinerdogan, and Anne Koziolek

Program Chairs