Towards Bridging the Gap between Goal-Oriented Requirements Engineering and Compositional Architecture Development

Sebastian Herold*, Andreas Metzger, Andreas Rausch*, Heiko Stallbaum

Software Systems Engineering
Institute for Computer Science and Business Information Systems (ICB)
University of Duisburg-Essen, Germany
www.sse.uni-due.de

* = affiliated with TU Clausthal
Agenda

- Motivation
- The REDSAA Approach
- Conclusions and Perspectives
Motivation
Characteristics of RE and AD

- RE and AD activities are **strongly interrelated** and are continuously and iteratively performed:
  - RE → AD: the architecture of a software system must satisfy its requirements
  - AD → RE: architectural constraints might prohibit certain requirements to be realized, implying a change to the initial requirements
  - RE → AD: additional requirements might be discovered during the development process, leading to changes in the architecture.

- Design **architectural decisions taken early** in this iterative process are the most crucial ones
  - they are very hard and costly to change later
Motivation
Observed Deficits

- Early AD activities in practice are mostly guided by the intuition and the experience of the software architects
  - this includes deciding on the coarse-grained architecture of the software system

→ Consequences:
  - activities tend to be unsystematic
  - results tend to be hard to predict or plan
  - activities are often difficult to be performed by a novice to architectural design
Motivation

Existing Solutions

- **Nuseibeh**
  - Twin-Peaks-Model: iteratively refine requirements and the architecture
    - analysis of the architectural alternatives & support for selecting from the alternatives is not provided explicitly

- **Chung et al.**
  - NFR models for deciding on best arch. alternative
    - does not state how to actually derive an architecture alternative from the given requirements

- **Baum et al.**
  - Deriving architecture from requirements
    - very detailed level of the requirements → better suited for later development stages

- **Perry et al.**
  - Architectural Prescriptions developed from KAOS models
    - not explicitly supporting potential reuse of architectural entities & reasoning on architectural alternatives
Agenda

- Motivation
- The REDSAA Approach
- Conclusions and Perspectives
The REDSAA Approach
Basic Ideas

- REDSAA =
  Requirements-driven Design and Selection of Architecture Alternatives

- Idea 1) Support the early reasoning on architectural alternatives by employing early RE models:
  
  ➔ Goal-oriented RE
  
  - leads to alternative architectural solutions for achieving the goals
  - architectural alternatives fulfill (satisfice) the goals differently
    - choice between alternatives can be based on a “profile” of goal satisfaction
  - understanding goals of the stakeholders early on will lead to software systems that better match the expectations of the stakeholders
The REDSAA Approach
Basic Ideas

- Idea 2) Establish **links between RE and AD** by employing formalized, coarse grained architectural quality requirements (AQRs)

  - AQRs are defined from **architectural drivers** identified in the **GRL models**
    - e.g., “sales system should be optimized for fast input”
  - Reusable, coarse grained **architectural entities** (components / patterns / styles) are augmented with **how well they contribute to AQRs** in general
    - e.g., “fat client positively contributes to high performance for input/output”

**Note:** this is an “improvement” over the solution proposed in the paper (→ “generic architectural drivers“)
The REDSAA Approach

Three Main Steps

1. **Requirements Engineering**
   - Identification of architectural drivers and their architectural requirements

2. **Architectural Design**
   - Composition of architectural alternatives and derivation of architectural assurances

3. **Goal Model (GRL)**
   - Feedback to goal model based on identified architectural alternatives

**Reusable Architectural Entities**
- C1, C2, C3, C4, C5

A. Metzger – SHARK@ICSE, Minneapolis, 2007
The REDSAA Approach
1) Identify Arch. Drivers & Arch. Requirements

- a) Identify those elements of a GRL model that **significantly impact on the architecture** of the system to be developed ➔ **architectural driver**

- b) Those architectural drivers are **mapped to AQRs** by stating their “requirements” on the quality characteristics of the architecture
The REDSAA Approach
1) Identify Arch. Drivers & Arch. Requirements

- 1a)

**A-Driver 1:** soft-goal “adaptability of sales process” is only achieved when “flexible sales system to be used” is realized.

**A-Driver 2:** soft-goal “low personnel costs” is only achieved when “sales system be optimized for fast input” is realized.
The REDSAA Approach
1) Identify Arch. Drivers & Arch. Requirements

- 1b)

<table>
<thead>
<tr>
<th>performance</th>
<th>usability</th>
<th>reliability</th>
<th>variability</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>+</td>
<td>O</td>
<td>++</td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>performance</th>
<th>usability</th>
<th>reliability</th>
<th>variability</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>++</td>
<td>++</td>
<td>+</td>
<td>O</td>
<td>...</td>
</tr>
</tbody>
</table>
The REDSAA Approach
2) Identify Architecture Alternatives

- Derive architecture alternatives by "composing" reusable architectural entities and determining how well they satisfy the AQRs
  - Only compositions that satisfy the AQRs will be used for input to further decision making process
The REDSAA Approach
2) Identify Arch. Alternatives

Reuseable Architectural Entities

<table>
<thead>
<tr>
<th>SOA</th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
</tr>
</thead>
<tbody>
<tr>
<td>OOA</td>
<td></td>
<td></td>
<td></td>
<td>AQR1 &gt; O</td>
<td>n/a</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td>AQR2 &gt; O</td>
<td>n/a</td>
<td>+</td>
<td>n/a</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>AQR3 = O</td>
<td>n/a</td>
<td>+</td>
<td>n/a</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td>AQR4 = O</td>
<td>n/a</td>
<td>-</td>
<td>n/a</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>
3) Feedback to RE

- For each architectural alternative, the **concrete goal satisfaction is determined**

**Potential results:**

- **a) Goal satisfaction profile**, which can be used during the actual decision making process (together with stakeholders)
  - e.g. by considering different priorities of the goals

- **b) Refinement and identification of additional goals**
  - this includes the refinement of contributes links (e.g., more fine-grained specification of whether + or -)
### The REDSAA Approach

#### 3) Feedback to RE

- **a)**

<table>
<thead>
<tr>
<th></th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
</tr>
</thead>
<tbody>
<tr>
<td>AQR1</td>
<td>n/a</td>
<td>+</td>
<td>n/a</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>AQR2</td>
<td>n/a</td>
<td>+</td>
<td>n/a</td>
<td>+</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>AQR3</td>
<td>n/a</td>
<td>+</td>
<td>n/a</td>
<td>-</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>AQR4</td>
<td>n/a</td>
<td>-</td>
<td>n/a</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

**Prio 1**

**Prio 2**

**Choice**
The REDSAA Approach

3) Feedback to RE

- b)
Agenda

- Motivation
- The REDSAA Approach
- Conclusions and Perspectives
Conclusion and Perspectives

- The REDSAA-Approach will...
  - more closely and systematically integrates and aligns RE and AD
  - allow identifying and reasoning on architectural alternatives *early*
  in the development process

- “Open Issues”
  - what language concepts are needed for formalizing the AQRs?
    - what is a suitable taxonomy for quality characteristics?
    - what are the building blocks of a coarse grained architecture
      (styles/patterns/components/…)?
  - what are the specific actions to be performed by the requirements engineers and architects?
  - what is the right level of formalization for the approach to be used in practice?
References


