

Automated Requirements Traceability

Requirements Engineering II

Daniel Pfeifer, Stefan Hutter, Sebastian Müller

02.11.2009

How can traceability information be
generated by software?

- Use **Best Practices** to define the requirements
- **Formalize** the **natural language** requirements
(Information retrieval)
- **Map** low-level elements to high-level
elements

Best Practices I

- A suitable **trace granularity** must be defined for each artifact type
- A well-defined **project glossary** should be used. Consistency in term usage improves traceability.
- The requirements should fulfill **generally accepted qualities** of good specification. E.g. unambiguous, complete, consistent, identifiable, etc.

Best Practices II

- The individual requirements should be embedded in a **meaningful hierarchy** to strengthen the semantic
- The semantic gap between the different stakeholders of the project should be close. Define **intradomain synonyms**
- Create **rich content** - user and system requirements should have some common terms.

Formalize natural language requirements

- The **system analyzes and indexes** the incoming documents
- To improve the result of Indexing one can use:
 - Thesaurus (car, vehicle, vw golf)
 - Incorporating user feedback
 - Stopword removal (that, the, and...)
 - Stemming
 - Filtering

Map low-level to high-level elements

- Automatically **generate** the Requirements Traceability Matrix (**RTM**)
- If a user queries for a specific requirement, the automated systems analyzes the incoming query and uses matching/ranking algorithms to determine which documents are similar to the query representation.

Can and should humans be replaced for defining traceability?

- It's **not** an **automated** but an **interactive** process:
 - A human being has to define the requirements in natural language first of all
 - The software provides the user with its best guess
→ the user has to examine all hits and sort out false negatives
 - The user should give a feedback so that the system can „train“

Automated Traceability in Research and Practice

- Research: analyzing and formalization of natural language, e.g. semantic web. Automated traceability to check humans work against algorithms.
- Practice: Practice: rarely used e.g. to create traceability afterwards.
- The system needs training material. False Positives/negatives require human review.