REQUIREMENTS ENGINEERING
LECTURE 2016/2017

Dr. Jörg Dörr

Requirements Specifications & Standards

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

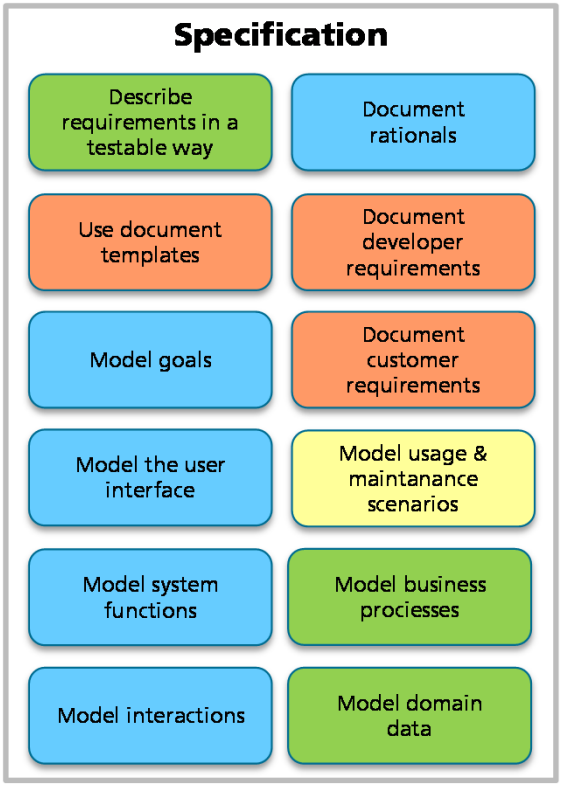
AG·SE

Fraunhofer IESE

AGENDA

- Standards & Templates
- Natural Language Requirements
- Specification with Conceptual Models
- Suitable Models for different Aspects

Recommended Specification Practices



Requirements Specification

- The activity of specifying requirements
- The document in which the specified requirements are contained

Requirements Specification

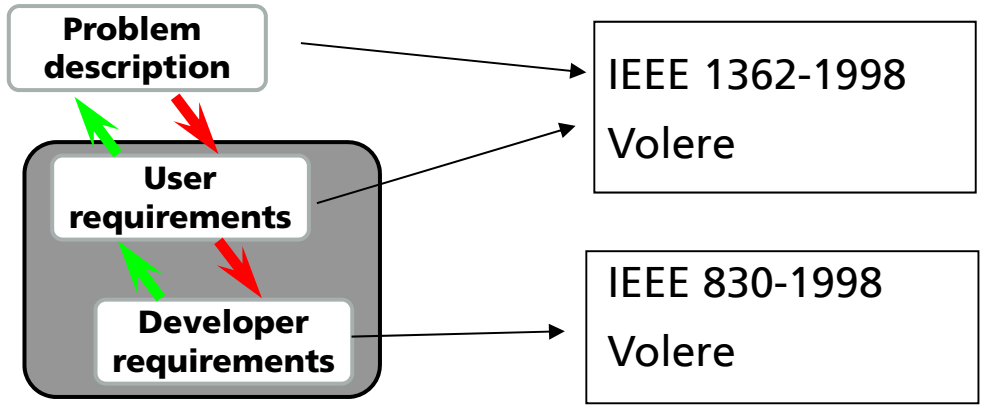
STANDARDS & TEMPLATES

Requirements Document Standards (1)

- Provide Templates
  - present a document outline for a requirements specification document (including a short content description for each chapter)
  - help to structure requirements documents
- Several Standards for Requirements Documents exist:
  - IEEE Standard 1362-1998 Guide for Information Technology – System Definition – Concept of Operations Document
  - IEEE Standard 830-1998 Recommended Practice for Software Requirements Specifications
  - Volere Template (James & Suzanne Robertson, Atlantic Systems Guild) http://www.systemsguild.com/GuildSite/Robs/Template.html

Requirements Document Standards (2)

- Standards tackle different levels of abstraction:
  - Problem Clarification
    - IEEE 1362-1998
    - Volere Template
  - Basis for Development
    - IEEE 830-1998
    - Volere Template



Requirements Document Standards (3)

- Templates basically contain sections to describe:
  - System context
    - Business Processes
    - Stakeholders
    - Rationale (Why is the software developed)
  - Organizational requirements
    - Constraints
    - Standards
  - Project Information
    - Cost and Effort Information
    - Risk
  - Functional requirements
    - What should the system do!
  - Non-functional requirements
    - How good should the system do its job.

Introducing three Standards

- IEEE Standard 1362-1998 Guide for Information Technology – System Definition – Concept of Operations Document
- IEEE Standard 830-1998 Recommended Practice for Software Requirements Specifications
- Volere Template (James & Suzanne Robertson, Atlantic Systems Guild)

Volere Template

- Developed by James & Suzanne Robertson (The Atlantic Systems Guild)
- Presents a template that may be used to specify user requirements as well as developer requirements
  - some template sections describe very detailed information about the system while other sections are very high level (developer vs user)
  - some template sections can be used for a developer audience as well as a user audience.

  In these cases either the used notation is the key differentiator or the information contained in the user document is refined in the developer section

- Available online: http://www.volere.co.uk/template.htm

Volere Template Overview (1)

- Project Drivers
  1. The Purpose of the Product
  2. Client, Customer and other Stakeholders
  3. Users of the Product
- Project Constraints
  4. Mandated Constraints
  5. Naming Conventions and Definitions
  6. Relevant Facts and Assumptions
- Functional Requirements
  7. The Scope of the Work
  8. The Scope of the Product
  9. Functional and Data Requirements

Volere Template Overview (2)

- Non-functional Requirements
  10. Look and Feel Requirements
  11. Usability Requirements
  12. Performance Requirements
  13. Operational Requirements
  14. Maintainability and Portability Requirements
  15. Security Requirements
  16. Cultural and Political Requirements
  17. Legal Requirements

Volere Template Overview (3)

- Project Issues
  - 18. Open Issues
  - 19. Off-the-Shelf Solutions
  - 20. New Problems
  - 21. Tasks
  - 22. Cutover / Migration to new product
  - 23. Risks
  - 24. Costs
  - 25. User Documentation and Training
  - 26. Waiting Room
  - 27. Ideas for Solutions

IEEE-1362 Template

- Developed by IEEE
- Presents a template that may be used to specify user requirements
- The template describes
  - current situation (without system)
  - justification for change (why new system)
  - description of proposed system (high level)

IEEE-1362 Template Overview (1)

- Title page
- Revision chart
- Preface
- Table of contents
- List of figures
- List of tables
- 1. Scope
  - 1.1 Identification
  - 1.2 Document overview
  - 1.3 System overview
- 2. Referenced documents

IEEE-1362 Template Overview (2)

- 3. **Current system** or situation
  - 3.1 Background, objectives, and scope
  - 3.2 Operational policies and constraints
  - 3.3 Description of the current system or situation
  - 3.4 Modes of operation for the current system or situation (e.g. active, maintenance, emergency)
  - 3.5 User classes and other involved personnel
  - 3.6 Support environment
- 4. **Justification** for and nature of **changes**
  - 4.1 Justification of changes
  - 4.2 Description of desired changes
  - 4.3 Priorities among changes
  - 4.4 Changes considered but not included

IEEE-1362 Template Overview (3)

5. Concepts for the **proposed system**
    - 5.1 Background, objectives, and scope
    - 5.2 Operational policies and constraints
    - 5.3 Description of the proposed system
    - 5.4 Modes of operation
    - 5.5 User classes and other involved personnel
    - 5.6 Support environment
6. **Operational scenarios**
7. Summary of impacts
    - 7.1 Operational impacts
    - 7.2 Organizational impacts
    - 7.3 Impacts during development

IEEE-1362 Template Overview (4)

- 8. Analysis of the proposed system
  - 8.1 Summary of improvements (new capabilities, deleted capabilities, improved performance)
  - 8.2 Disadvantages and limitations
  - 8.3 Alternatives and trade-offs considered
- 9. Notes
  - Appendices
  - Glossary

IEEE-830 Template

- Developed by IEEE
- Presents a template that may be used to specify developer requirements (some times it is partially used to describe user developer requirements as it contains parts that are on a higher level)
- The template describes
  - overview of the system
  - justification for change (why new system)
  - description of proposed system (high level)

- In addition the template provides **characteristics for a good software requirements specification document**

IEEE-830 Template Overview (1)

- 1. Introduction
  - 1.1 Purpose
  - 1.2 Scope (Name, General System Description, Benefits)
  - 1.3 Definitions, acronyms, and abbreviations
  - 1.4 References
  - 1.5 Overview

IEEE-830 Template Overview (2)

- 2. Overall description

  2.1 Product perspective: System interfaces, user interfaces, HW interfaces, SW interfaces, Communications Interfaces Memory constraints

  2.2 Product functions

  2.3 User characteristics

  2.4 Constraints

  2.5 Assumptions and dependencies

IEEE-830 Template Overview (3)

- 3. Specific requirements
  - 3.1 External interfaces
  - 3.2 Functions
  - 3.3 Performance requirements
  - 3.4 Logical database requirements
  - 3.5 Design constraints
  - 3.6 Standards compliance
  - 3.7 Software system attributes
    - Reliability
    - Availability
    - Security
    - Maintainability
    - Portability
- Appendixes

Refined in ISO 9126 / ISO 25010

Product Quality (ISO 9126/DIN 66272)

Functionality
- Adequacy
- Security
- Precision of calculation
- Interoperability
- Conformity with standards

Reliability
- Maturation
- Fault tolerance
- Recovery

Usability
- Comprehensibility
- Learnability
- Operability

Efficiency
- Time response
- Consumption

Changeability
- Analyzability
- Modifiability
- Stability
- Verifiability

Portability
- Adaptivity
- Installability
- Conformity with standards
- Replaceability

Types of Quality Requirements (ISO 25010)

- Quality in Use (relative to human use)
  - Effectiveness
  - Efficiency
  - Satisfaction
  - Freedom of Risk
  - Context Coverage
- Product Quality (intrinsic)
  - Functional Suitability
  - Performance Efficiency
  - Compatibility
  - Usability
  - Reliability
  - Security
  - Maintainability
  - Portability

Standards Summary

- Standards provide means to structure requirements documents
  - better overview
  - higher readability
  - overall raise in understandability
- Standards indicate what should be the content of a requirements specification
  - provides mean to raise completeness
- Standards do not indicate HOW to specify different parts or HOW to guarantee the characteristics of a good document
  - no support to choose notation to specify a certain section
  - no support in how to achieve for example completeness ore traceability