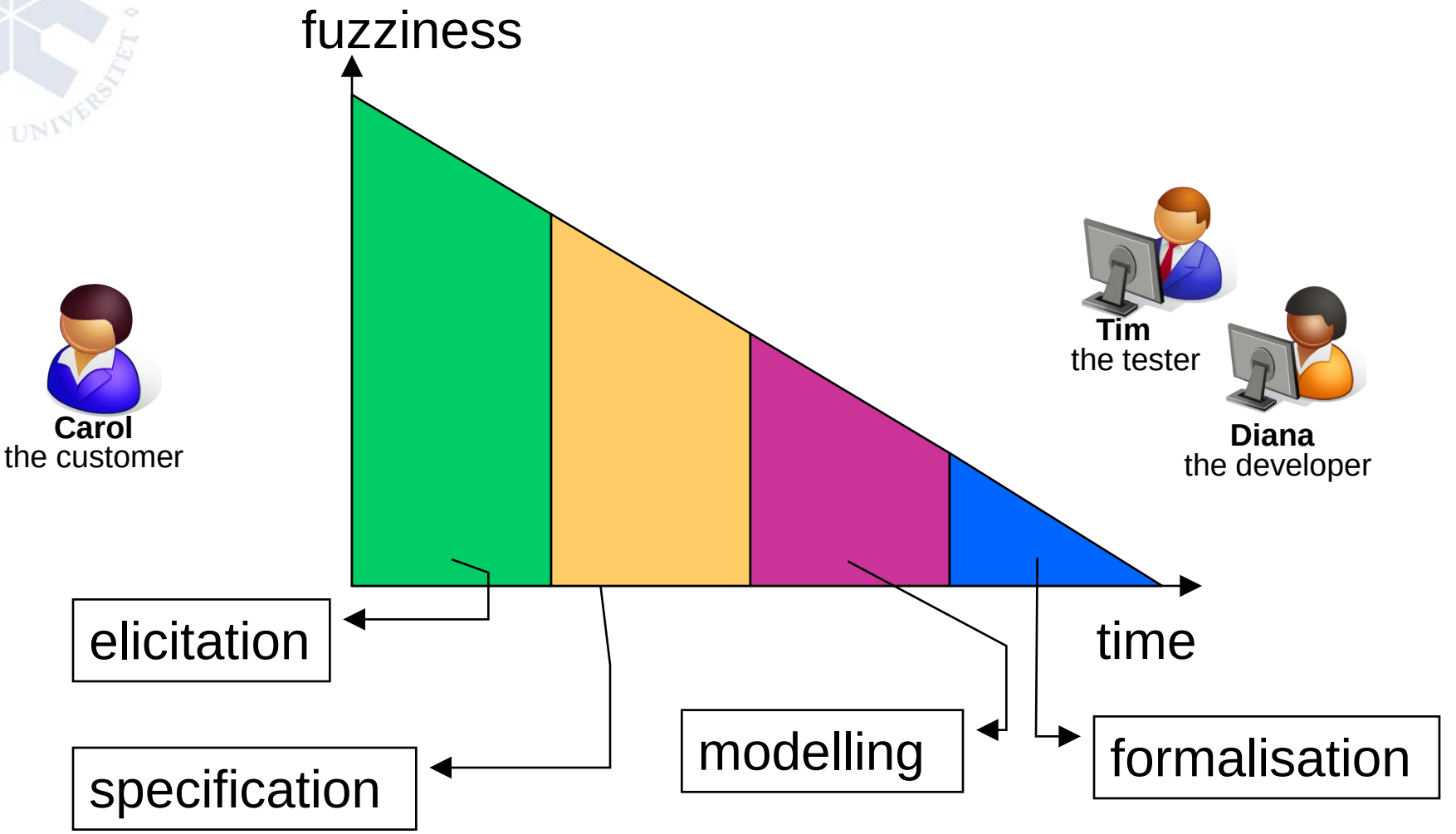
The role of requirements in the life-cycle

Kristian.Sandahl
@liu.se



Part I
Requirements Elicitation

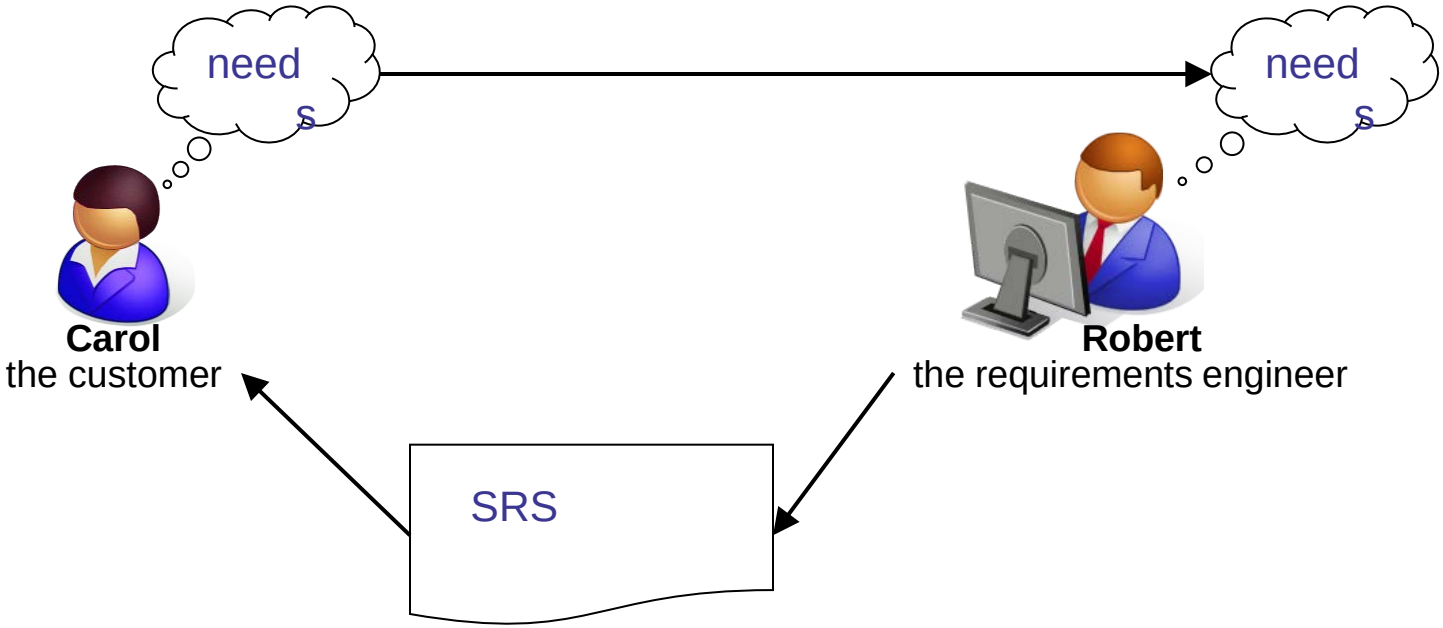
Part II
Requirements Analysis

Part III
Requirements Specification

Linköpings universitet

Advice towards a good specification

Kristian.Sandahl
@liu.se



- There is no perfect specification, but you can write a good one
- The RS, or SRS avoids many misunderstandings
- The RS is of special importance in outsourcing programming

SRS contents

Kristian.Sandahl
@liu.se

1 Introduction
- 1.1 Purpose
- 1.2 Scope
- 1.3 Definitions, acronyms and abbreviations
- 1.4 References
- 1.5 Overview

2 Overall description
- 2.1 Product perspective
- 2.2 Product functions
- 2.3 User characteristics
- 2.4 General constraints
- 2.5 Assumptions and dependencies
- 2.6 Lower ambition levels

3 Specific requirements

3.1 Interface requirements
- 3.1.1 User interfaces
- 3.1.2 Hardware interfaces
- 3.1.3 Software interfaces
- 3.1.4 Communication interfaces

3.2 Functional requirements

3.3 Performance requirements

3.4 Design constraints

3.5 Software system attributes

3.6 Other requirements

4 Supporting information
- 4.1 Index
- 4.2 Appendices

Individual requirements

Kristian.Sandahl
@liu.se

Requirement #: Requirement Type: Event/use case #:

Description:

Rationale:

Source:
Fit Criterion:

Customer Satisfaction: Customer Dissatisfaction:
Dependencies: Conflicts:
Supporting Materials:
History:

Volere

Copyright © Atlantic Systems Guild

Requirements specification

Kristian.Sandahl
@liu.se

Requirements are:

- Numbered
- Inspected
- Prioritised
- Unambiguous
- Testable
- Complete
- Consistent
- Traceable
- Feasible
- Modifiable
- Useful for:
  - operation
  - maintenance
  - customer
  - developer
  - ….

Define a standard document structure

Kristian.Sandahl
@liu.se

Benefits:

- Readers can reuse knowledge from previous RSs in understanding
- Writers' checklist
- Tools can be adapted to generate RSs

Costs:

- Finding the right standard
- Configure variants
- Periodically review standard
- Developers can have a bad attitude against standards

Explain how to use the document

Kristian.Sandahl
@liu.se

There are many readers of a RS:

- Customers
- Managers
- Software engineers
- Testers
- Maintenance staff
- Technical writers
- Subcontractors

- Part of introduction
- Types of reader
- Technical background needed
- Sections for different readers
- Sections skipped 1st time
- Order of section
- Dependence between section

Takes an hour to write

Include a summary of the requirements

Kristian.Sandahl
@liu.se

- Better than forward references
- Focus attention on critical and prioritised requirements
- Map to find specific requirements

- Highlight most important requirements in a list
- Table of classification
- Graphic presentation with relations
- Per chapter basis
- Though for large number of requirements

Make a business case for the system

Kristian.Sandahl
@liu.se

- Helps understanding
- Helps change assessment
- Special document, section or part of introduction
- Requires that top management have an agreement

Define special terms

Kristian.Sandahl
@liu.se

- Readers and writers might have their own meaning of terms
- Requirements engineer develops a jargon that need to be explained
- Use a glossary, start with a standard one, adapt and maintain
- Highlight terms in the text that can be found in the glossary

Use a data dictionary

Kristian.Sandahl
@liu.se

- A glossary for variables and terms in diagrams
- Often well-supported in tools
- Often forgotten in student-RSs
- Needs maintenance and adherence
- Can develop into an ontology => massive information exchange, search and checking

- Name of entity
- Aliases
- Type
- Description
- Rationale
- Constraints
  - Units
  - Tolerance
  - Value ranges
  - Error values
- Relations
- Links

Lay out the document for readability

Kristian.Sandahl
@liu.se

- Many, many readers justify the investment
- Meanwhile, use your standard templates of your word processor and common sense
- It is worthwhile to buy professional training for newly hired personnel

Help readers find information

Kristian.Sandahl
@liu.se

- Create table of contents
- Create index
- Easy to find support for automatic generation
- Human-made indices are still better

Make documents easy to change

Kristian.Sandahl
@liu.se

- Requirements will be changed
- Quite easy with tools
- Paper-based specifications needs some thinking:
  - Loose-leaf binders
  - Change bars
  - Short, self-contained chapters
  - Refer to labels, not pages

Summary - What have we learned today?

Kristian.Sandahl
@liu.se

- Elicitation is a very human-centered phase
- A written specification is read far more often than it is written
- Use-cases describe the mainstream flow of event