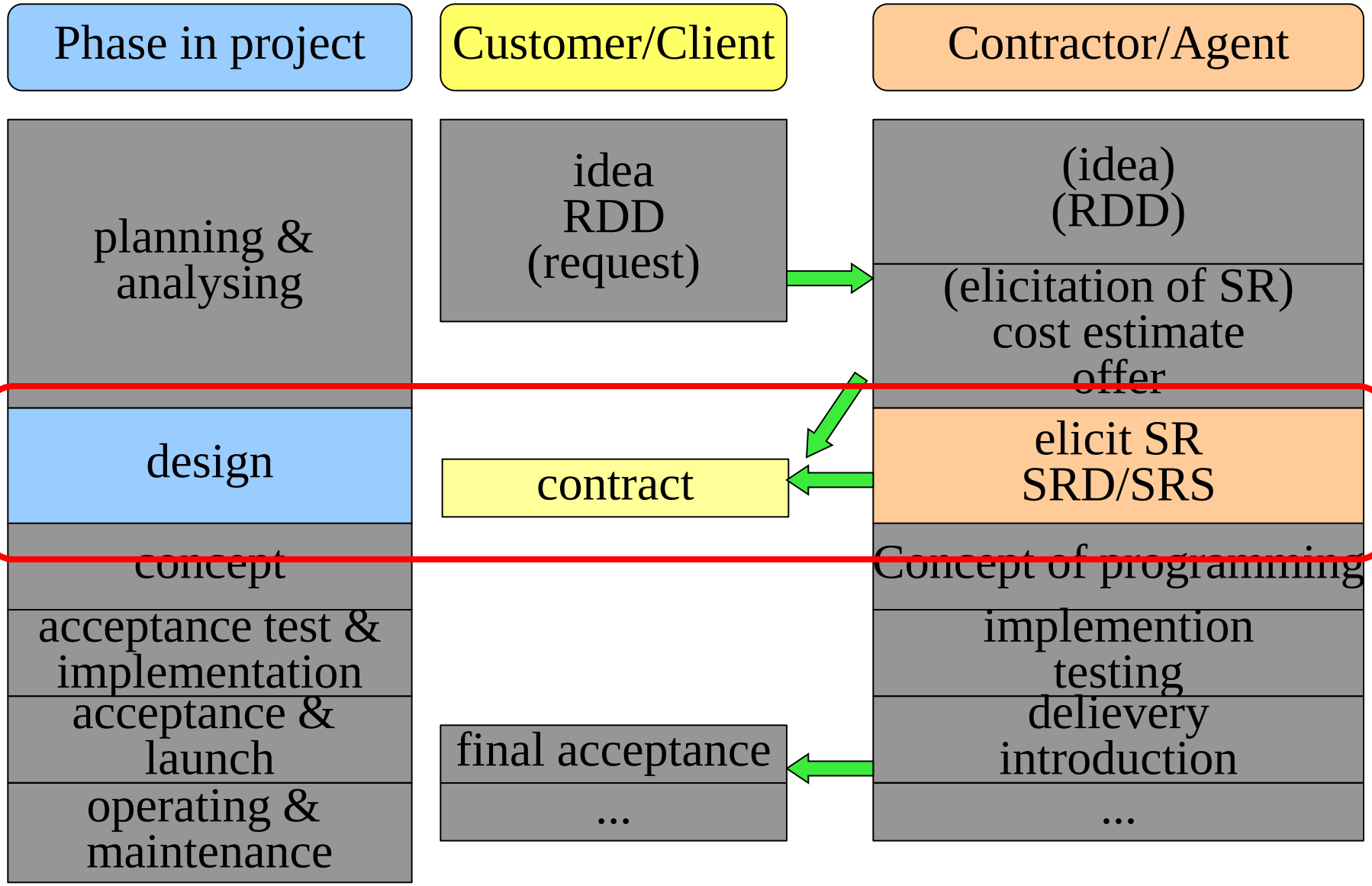
tgm
Die Schule der Technik

Cluster: software development

Content: design

Dr. Walter Rafeiner-Magor
09.11.2012



RDD: requirements definition document
SR: system requirements
SRD: system requirements definition
SRS: software requirements specification

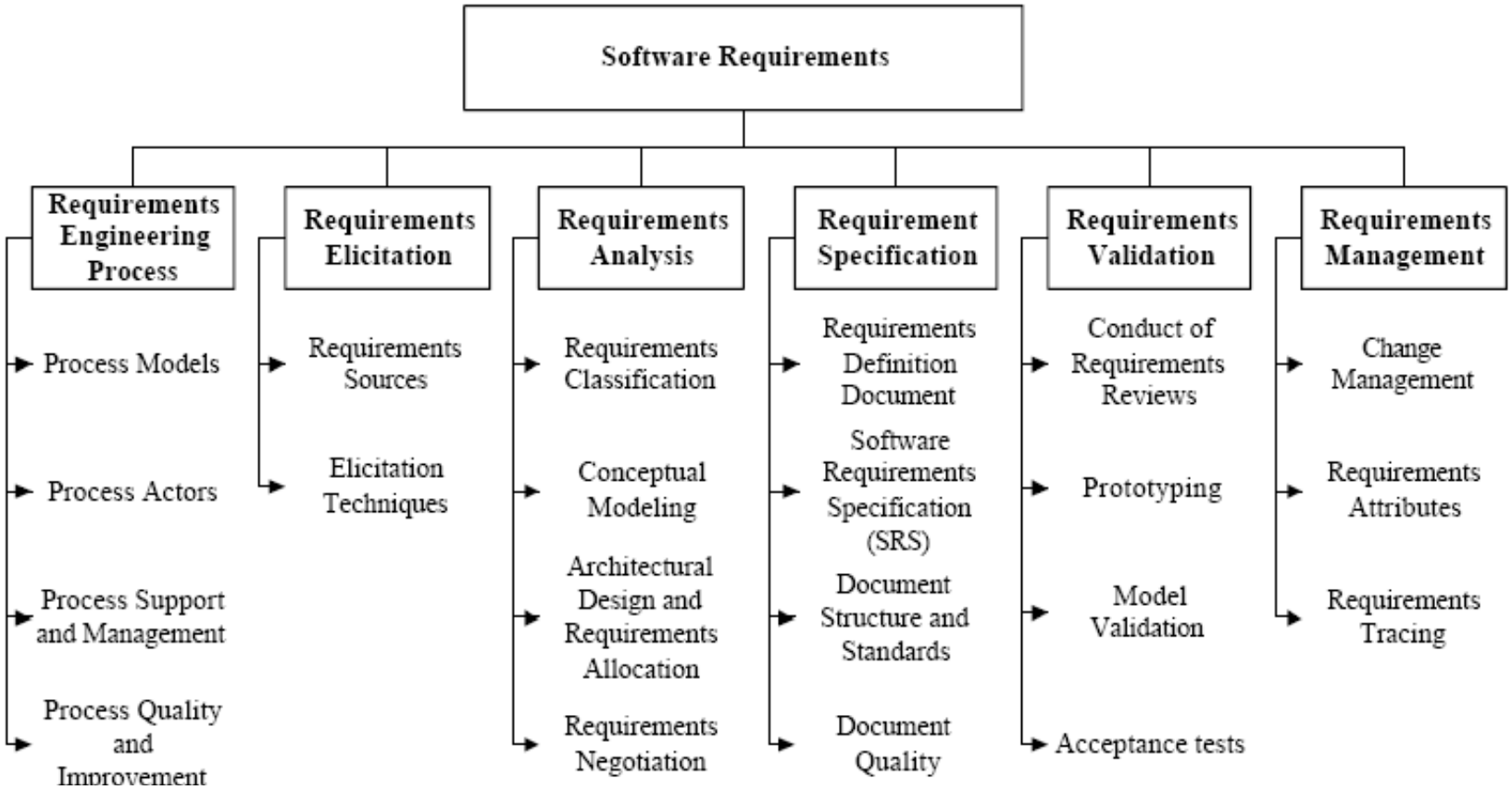
The hardest single part of building a software system is deciding precisely what to build. […] No other part of the work so cripples the resulting system if done wrong; no other part is more difficult to rectify later.

Frederick P. Brooks

Design: What's to do?

- Requirements elicitation
- Requirements analysis
- Software requirements specification (SRS)
- Requirements validation
- Requirements management

Software requirements



Requirements

- functional capability
  - capability of the new system (RDD)
- quality capability:
  - constraints
    - performance (response time, thruput)
    - accuracy
    - reliability
    - safety
    - maintainability

Requirements: elicitation
let‘s get the requirements

- **all parties concerned**(who have an interest?)
  - initiator
  - Marketing department
  - specialist departments
  - user
- **as is situation**
  - legacy system
  - current forms
  - active processes

Source:H Balzert *Lehrbuch der Software-Technik I* (Spektrum Akademischer Verlag 2001).

Requirements: elicitation
How can we get it?

- active searching for requirements
  - objectives (critical success factors)
  - expert knowledge(domain knowledge)
  - technical environment
  - organisational environment

Source :H Balzert *Lehrbuch der Software-Technik I* (Spektrum Akademischer Verlag 2001).

Requirements: elicitation
let‘s talk about ?!

»We haven ´t got time to write a requirements specification; just build us a system.«

If you ever hear this from your customers, ask them how everyone will know that the work is done if there are no requirements. A pile of e-mail messages, scribbled ideas, and two-year-old meeting notes does not replace a structured SRS [Software Requirements Specification] and system models. *Show the customers examples of what you mean by a specification and explain the value of the document*, but don't let your team proceed without creating an SRS and supporting models.

Source:KE Wiegers Creating a Software Engineering Culture (Dorset House 1996) 99-100

Requirements elicitation
a fatal approach!

Include every requirement requested by your customers in the specification document.
It doesn´t matter how often a proposed feature would be used, how much it will cost to implement, what impact it will have on other parts of the system, or whether it is technically feasible.
The customer is always right; if he says he needs it, he must know what he is talking about.
Priorities are not necessary – we can do it all (eventually).

Source:KE Wiegers Creating a Software Engineering Culture (Dorset House 1996) 99-100

Requirements: elicitation techniques

- interviews
- case scenarios
- prototyping
- direct observation

Source:H Balzert *Lehrbuch der Software-Technik I* (Spektrum Akademischer Verlag 2001).

Requirements: elicitation observation
basic conditions

- client is the expert!
- case scenario in detail
- little preparation
- implementation is time-consuming

Source: HR Beyer, K Holtzblatt “Apprenticing with the customer” CACM 38:5 (1995) 45–51.

Requirements: analysis result

useful for

- find and solve inconsistencies
- Find system interfaces and system borders
- Make the translation from system requirements to software requirements!

Requirements: analysis
requirements classification

different dimensions:

- functional capabilities and quality capabilities
- capabilities for the system and business process
- to obtain priority
- define the scope
- stability (Requirements Drift Ø ca. 1% a month)

Source: Requirements Drift: C Jones „Our worst current development practices“ IEEE Software 13:2 (1996) 102–104.

Requirements: analysis

Make a concept!

- helps to understand the problem
- No solution concept!
- to represent the reality
- Different possibilities and techniquies:
  - data flow diagram
  - control flow diagram
  - constitutional diagram

object oriented approach for the model UML