The hardest single part of building a software system is deciding precisely what to build. No other part of the conceptual work is as difficult as establishing the detailed technical requirements, including all the interfaces to people, to machines, and to other software systems. No part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later. (Brooks, 1987)

Requirements Engineering 2009

Roman Bednarik

bednarik@cs.joensuu.fi

Introduction

- What we should learn in this course about RE
  - The role of requirements in software development
  - The system – development view and the user view, and their intersection
  - How to collect, report and manage requirements
  - What is the state-of-art of RE research
  - Users' point of view and developers' pov

Organization

- The course contains
  - Working in group of 3 – 4 people
  - Reviewing the work of others and giving feedback
  - Surveying literature
  - Presenting in seminars
  - Becoming an RE expert
  - Understanding that there is no single right way

Contents

- 2 projects 50%
  - A - Literature survey on a given RE topic
  - B - Requirements specification for a given system
- Presentation of A, compulsory, 10%
- Home weekly X tasks, 24%
  - Incl. feedback on B of some other group, 5%
- Final exam – very hard!, but only 16%!

~~Lectures~~

- Not really lectures as you know
- Seminars with presentations and discussion
- Schedule will be available soon online:
  - http://cs.joensuu.fi/pages/bednarik/RE2009
  - Subscribe to the RSS feed

Demos

- Tuesday and Thursday, 10-12, B181
- Demo teacher: Minnamari Naumanen
- Home tasks need to be delivered BEFORE the demo, in order to get points
- X task are compulsory

Key aspects of successful sw development

- What are they?

Key aspects of successful sw development

- What are they?

- Resources
- User input and involvement
- Effective management and support
- **Clearly defined, complete requirements**
  - Numerous sw engineering studies show this repeatedly

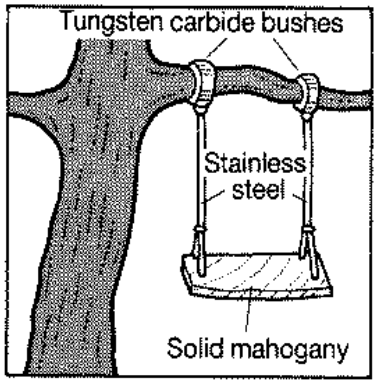
Project Success Factors

1. User Involvement 15.9%
2. Executive Management Support 13.9%
3. Clear Statement of Requirements 13.0%
4. Proper Planning 9.6%
5. Realistic Expectations 8.2%
6. Smaller Project Milestones 7.7%
7. Competent Staff 7.2%
8. Ownership 5.3%
9. Clear Vision & Objectives 2.9%
10. Hard-Working, Focused Staff 2.4%
11. Other 13.9%

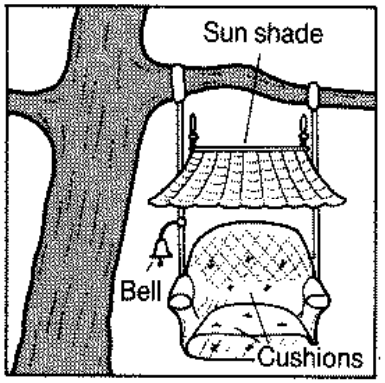
standishgroup

Other factors and risks

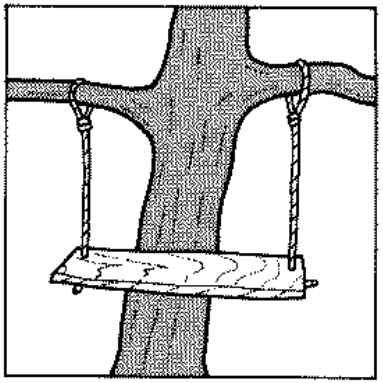
- Lack of commitment – management and user
- Misunderstanding requirements
- End user expectations
- Scope change, new technology
- Expertise
- Requirements froze/change
- Low method use



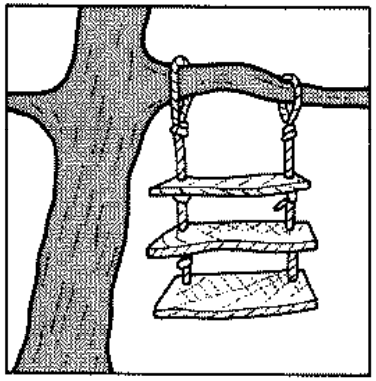
What Product Marketing specified



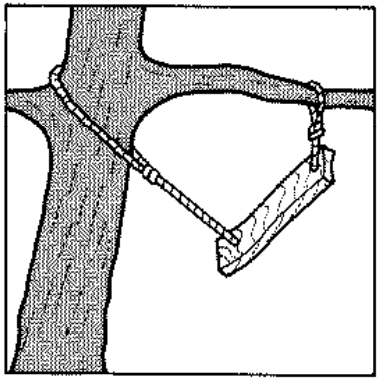
What the salesman promised



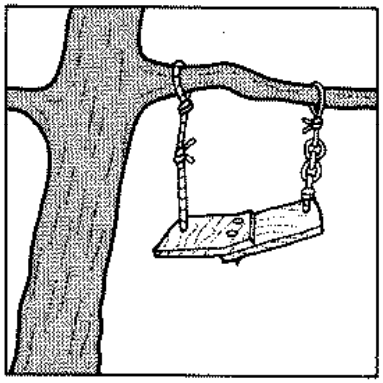
Design group's initial design



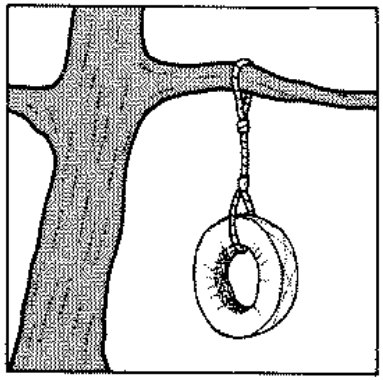
Corp. Product Architecture's modified design



Pre-release version



General release version



What the customer actually wanted

Principles Of Software Engineering Management, Tom Gilb , 1988, p.45

How Projects Really Work (version 1.5)

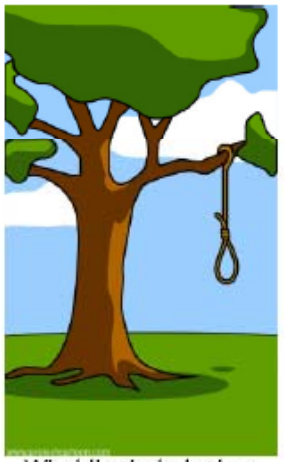
Create your own cartoon at www.projectcartoon.com

How the customer explained it

How the project leader understood it

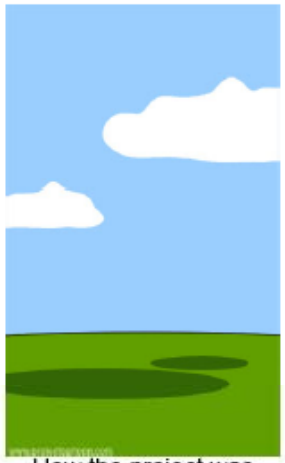
How the analyst designed it

How the programmer wrote it

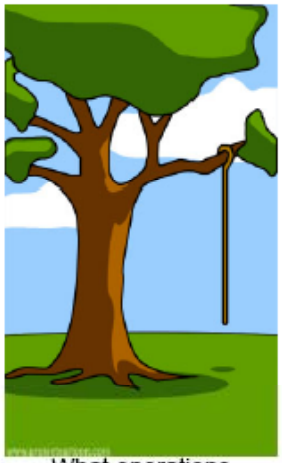


What the beta testers received

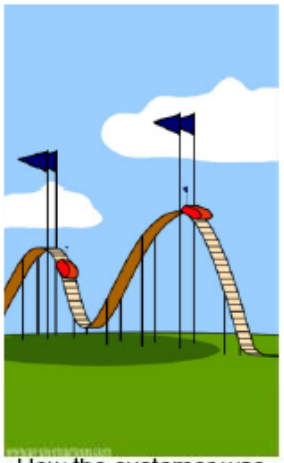
How the business consultant described it



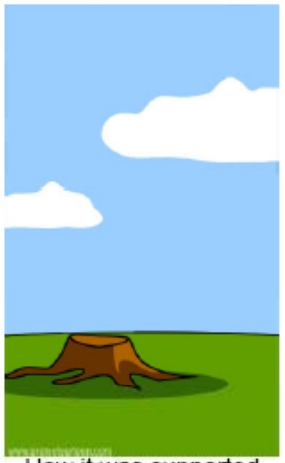
How the project was documented



What operations installed



How the customer was billed



How it was supported

What marketing advertised

What the customer really needed

How Projects Really Work (version 2.0)

Create your own cartoon at www.projectcartoon.com

How the customer explained it

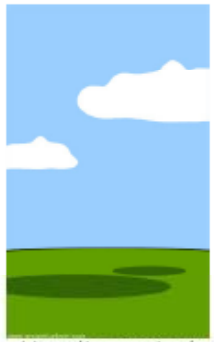
How the project leader understood it

How the analyst designed it

How the programmer wrote it

What the beta testers received

How the business consultant described it



How the project was documented

What operations installed

How the customer was billed

How it was supported

What marketing advertised

When it was delivered

What the customer really needed

What the digg effect can do to your site

The disaster recover plan

Requirements engineering

- Is one of the toughest parts in development
  - Everybody knows requirements are important
  - Few know how to do it right – 75% of companies surveyed in 1996 had some problems related to RE
- If we do not agree what to build, how do we build?
  - Analogy with building a house

Why to RE

- We try to separate problem from the design and from the solution
- We need to establish benchmarks, metrics, and views about the problem to select the right approach

- But if we desing something, that something will affect us => and maybe the problem too

So let's ask again, why RE?

- To minimize costs
  - Software is everywhere, a major part of many small (microwave) and big (airliner) systems
  - Serious consequences of failures
  - Rework/fixing costs – 50 x the price of fix during the requirements specification
  - Costs, costs, costs…
- To build systems that have purpose, value

What are they then?

- Requirements are some simplification and approximation of the problem
  - The solution, if matching the requirements, then fullfills the problem to a certain degree
- Engineering of requirements
  - Systematic transformation of customers needs into complete, consistent, precise, verifiable, formal, …., …., specifications

The Standard definition IEEE 610:1990

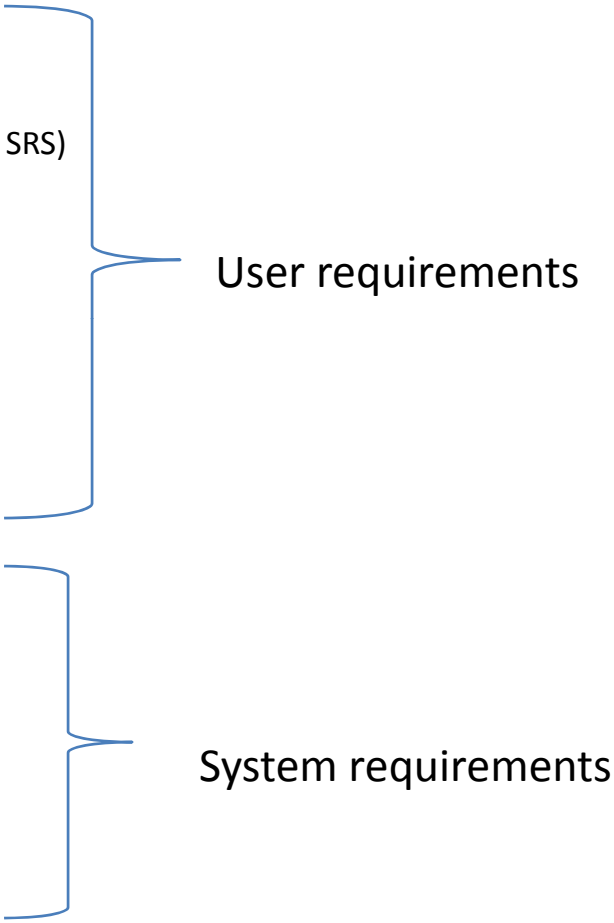
- (1) A condition or capacity needed by a **user to solve** a problem or achieve an objective.
- (2) A condition or capability that must be met or possessed by a **system** or system component **to satisfy** a contract, standard, specification, or other formally imposed documents.
- (3) A documented representation of a condition or capability as in (1) or (2).

Types of requirements

- User-requirements
  - What services will be provided? What goals can be solved? What are the limitations?
- System requirements
  - Detailed, specific requirements on functions, services, limitations of the implementation

Example: IEEE 830:1993

- Introduction (Section 1 of the SRS)
  - Purpose (1.1 of the SRS)
  - Scope (1.2 of the SRS)
  - Definitions, acronyms, and abbreviations (1.3 of the SRS)
  - References (1.4 of the SRS)
  - Overview (1.5 of the SRS)
- Overall description (Section 2 of the SRS)
  - Product perspective (2.1 of the SRS)
  - Product functions (2.2 of the SRS)
  - User characteristics (2.3 of the SRS)
  - Constraints (2.4 of the SRS)
  - Assumptions and dependencies (2.5 of the SRS)
  - Apportioning of requirements (2.6 of the SRS)
- Specific requirements (Section 3 of the SRS)
  - External interfaces
  - Functions
  - Performance requirements
  - Logical database requirements
  - Design constraints
  - Software system attributes
  - Organizing the specific requirements
  - Additional comments

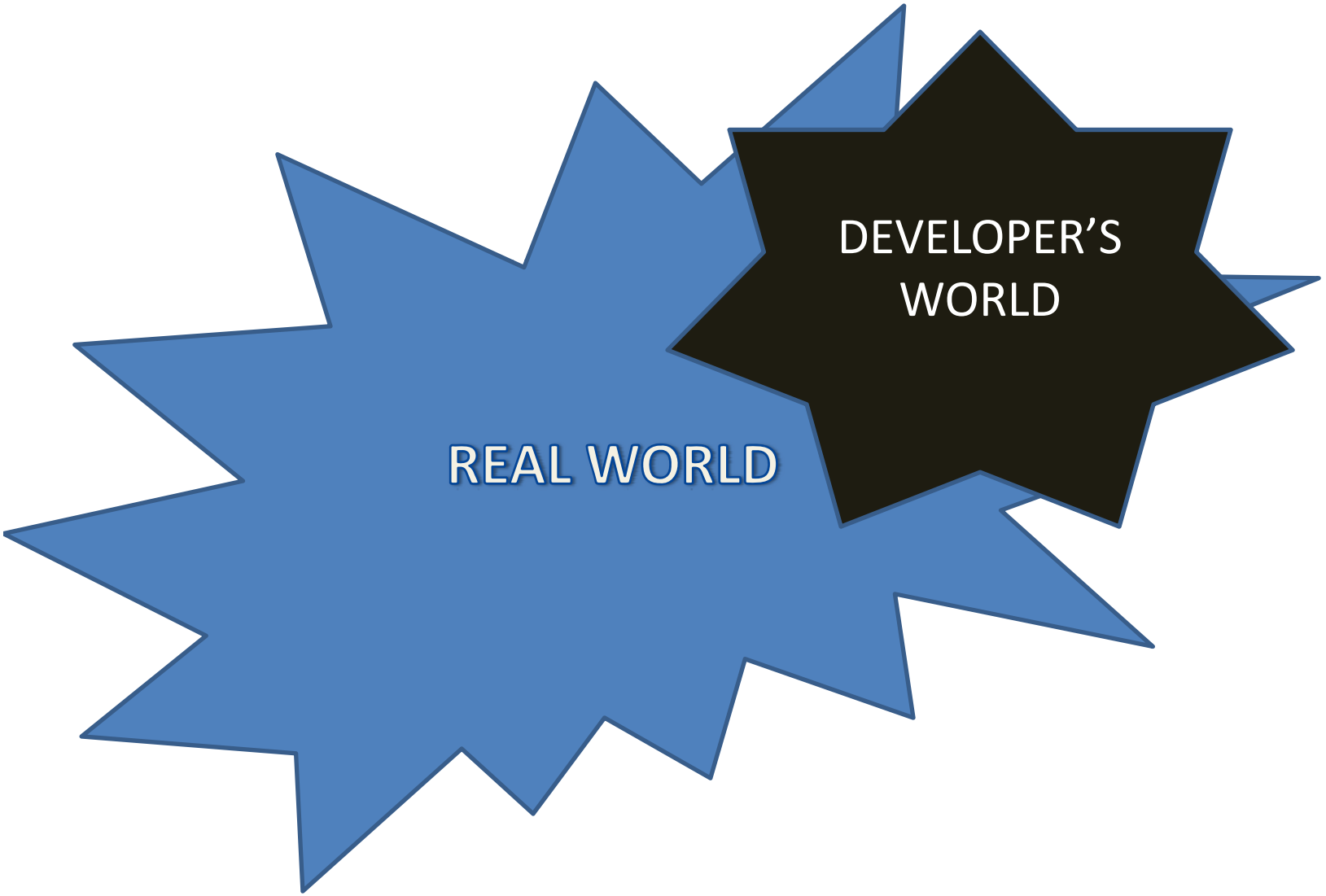


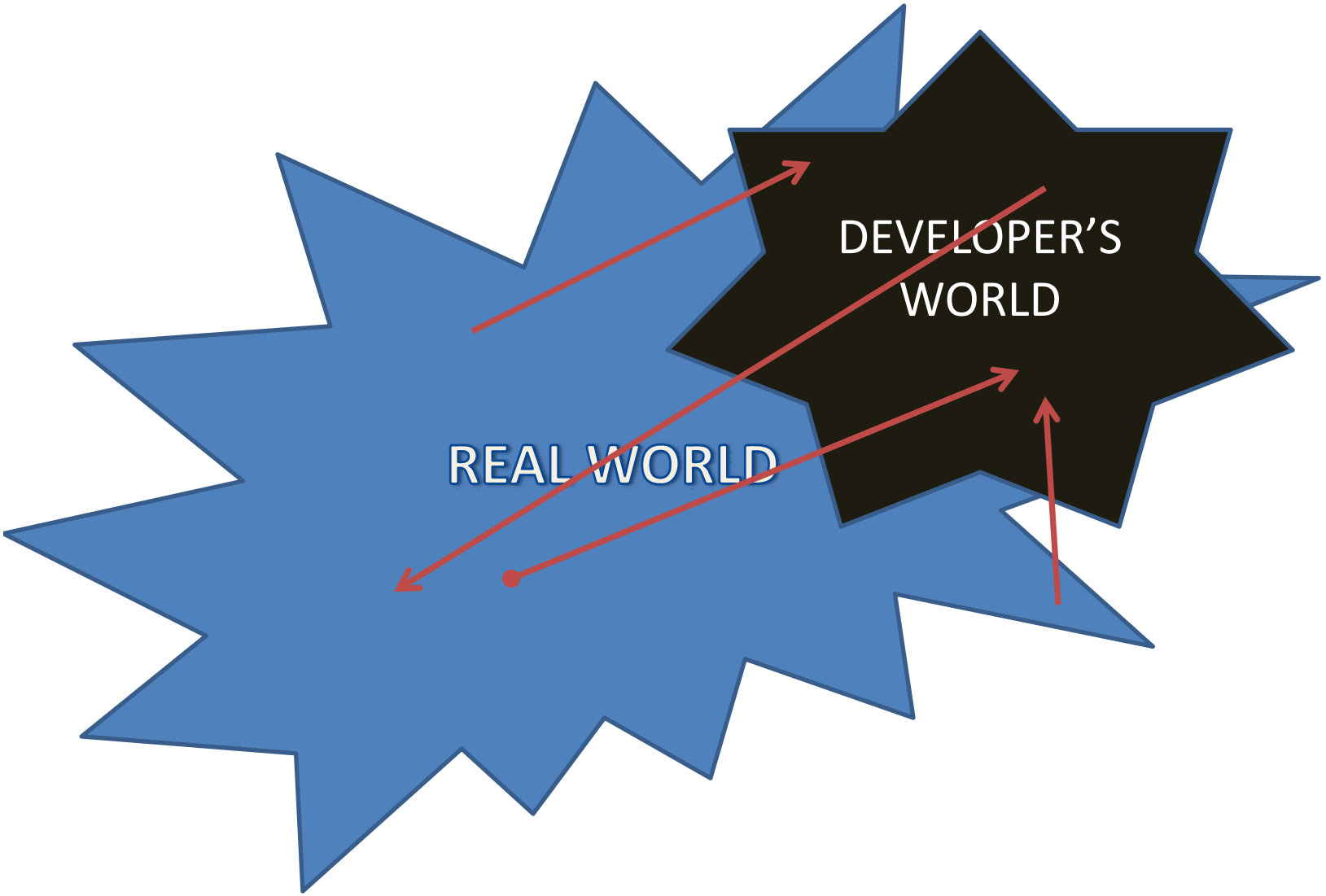
IEEE Recommended Practice for Software Requirements Specifications

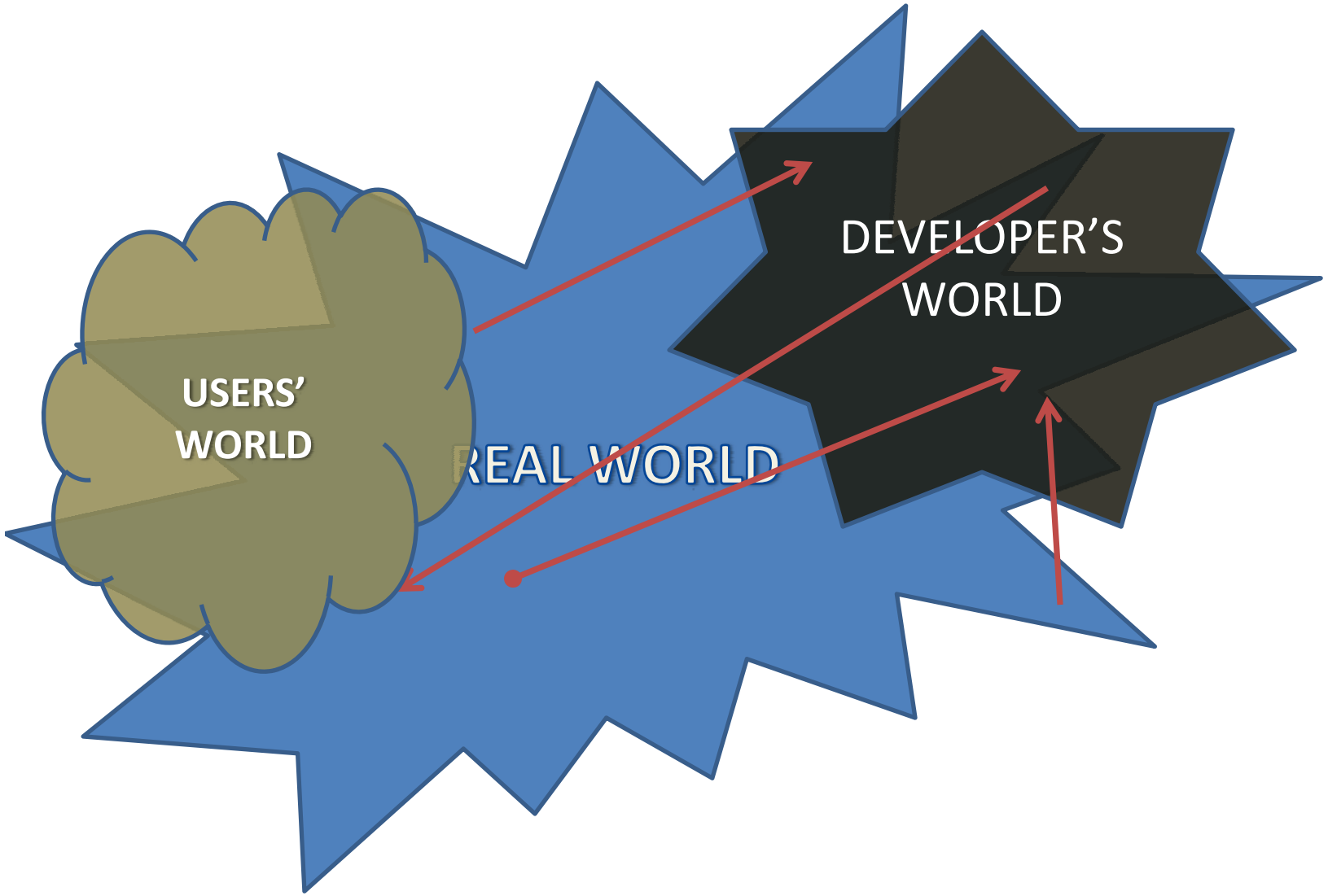
Requirements should be:

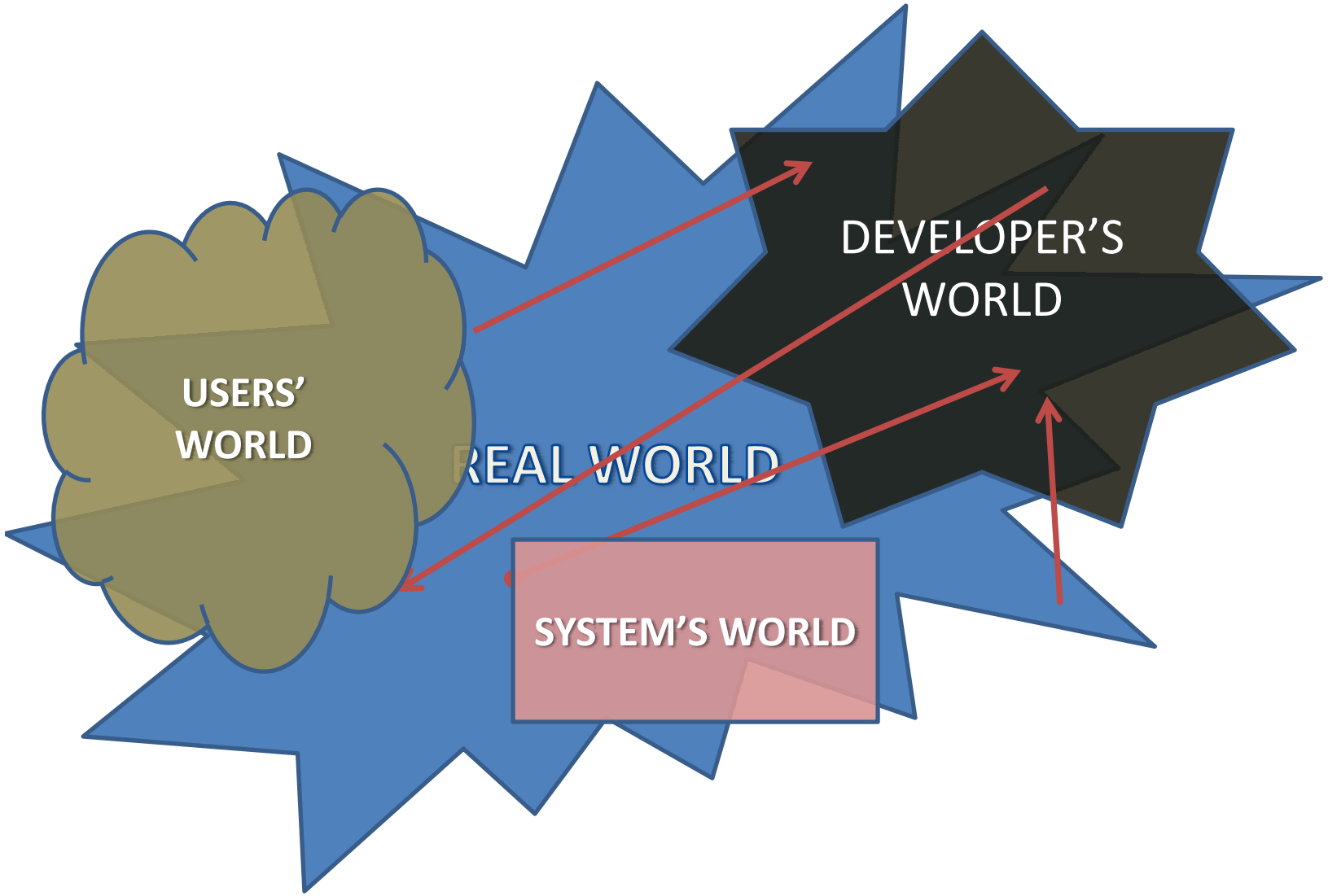
- Written in way that whole development can be based on them – developers understand. “Could I design and implement the system based on this document?”
- Customers and users understand
  - But what do these mean in practice?
- IEEE: Correct, Unambiguous (Specific), Complete, Consistent, Verifiable, Modifiable, Traceable

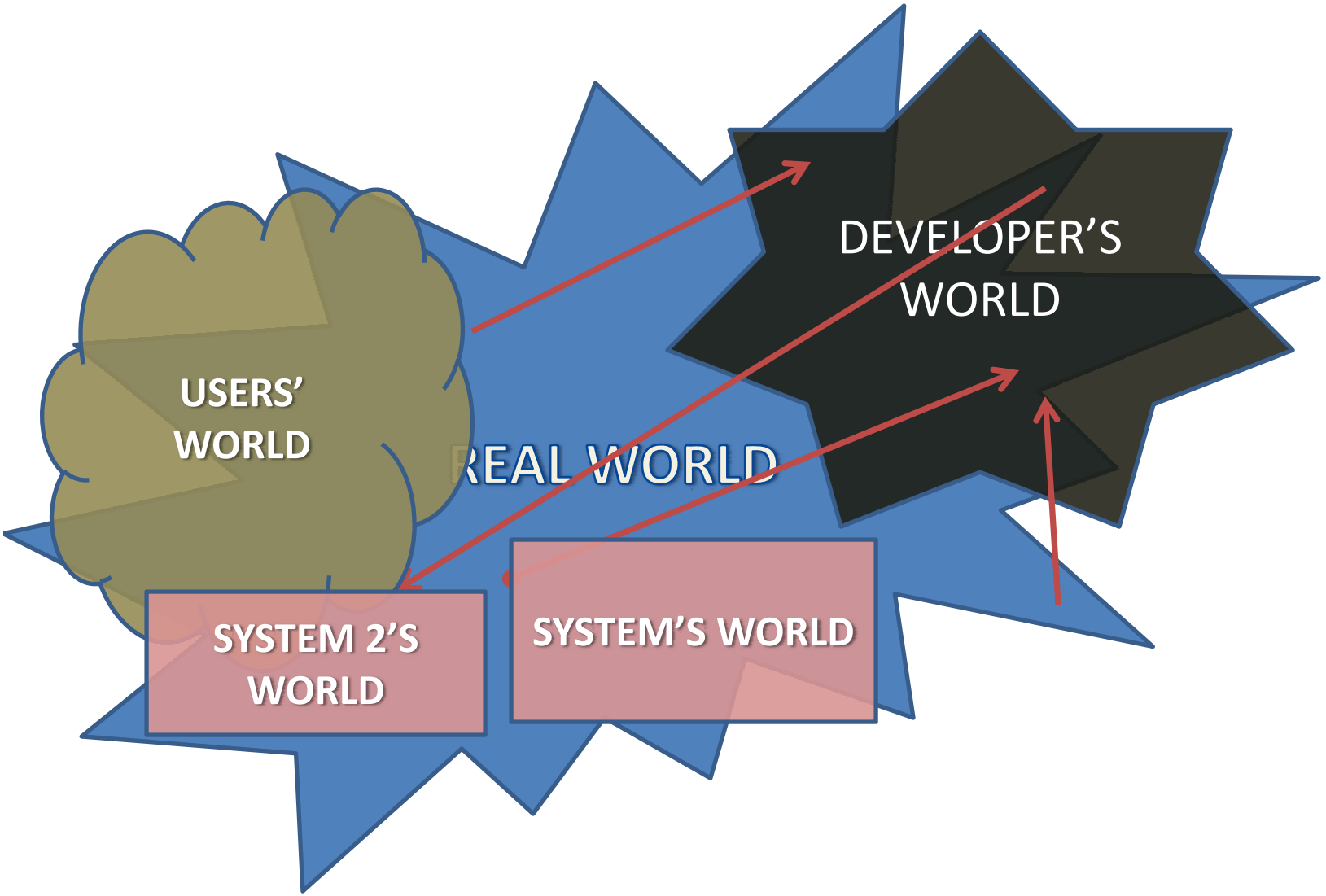
The system should be fast and easy-to-use.
There should be no more than 2 bugs in each unit.
User should be able to complete the transaction in less than 20 seconds.
At least 80% of users complete 95% of transactions in less than 20 seconds
First time users and users with low experience complete … without an error and their user experience will be …

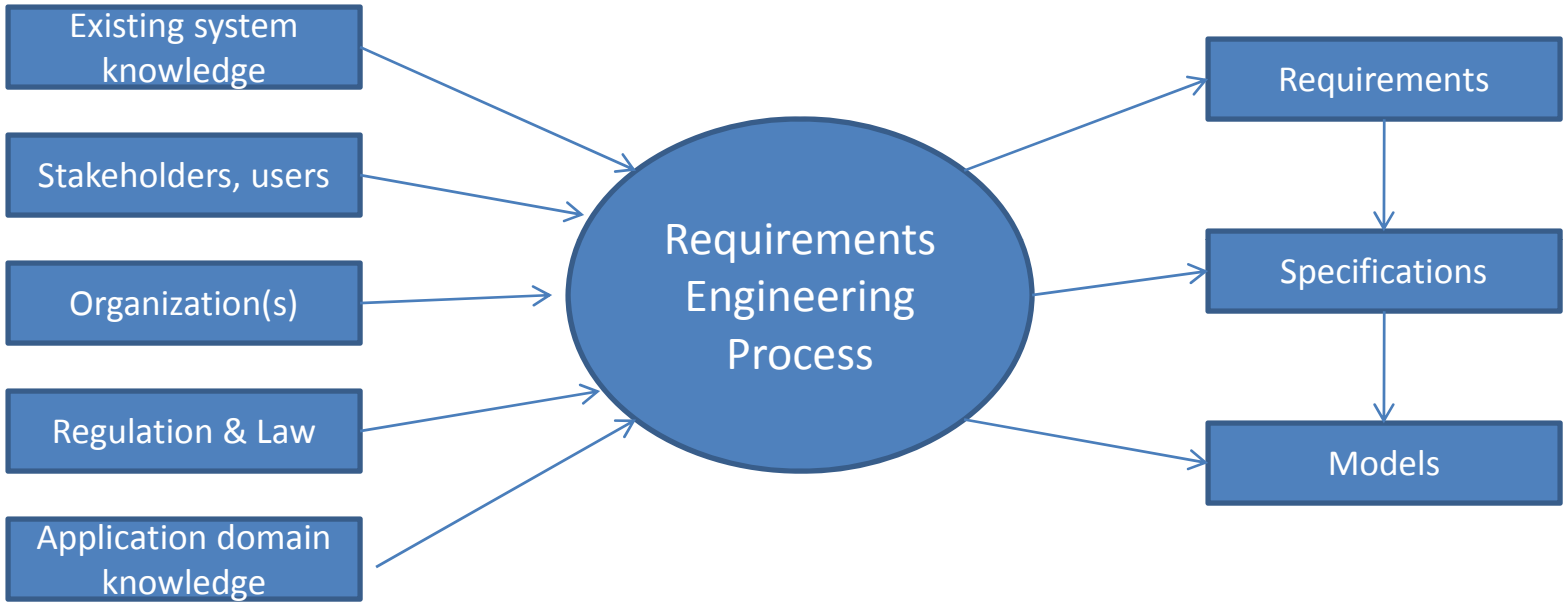

DEVELOPER'S
WORLD
REAL WORLD


DEVELOPER'S
WORLD
REAL WORLD


DEVELOPER'S
WORLD
USERS'
WORLD
REAL WORLD


DEVELOPER'S
WORLD
USERS'
WORLD
REAL WORLD
SYSTEM'S WORLD

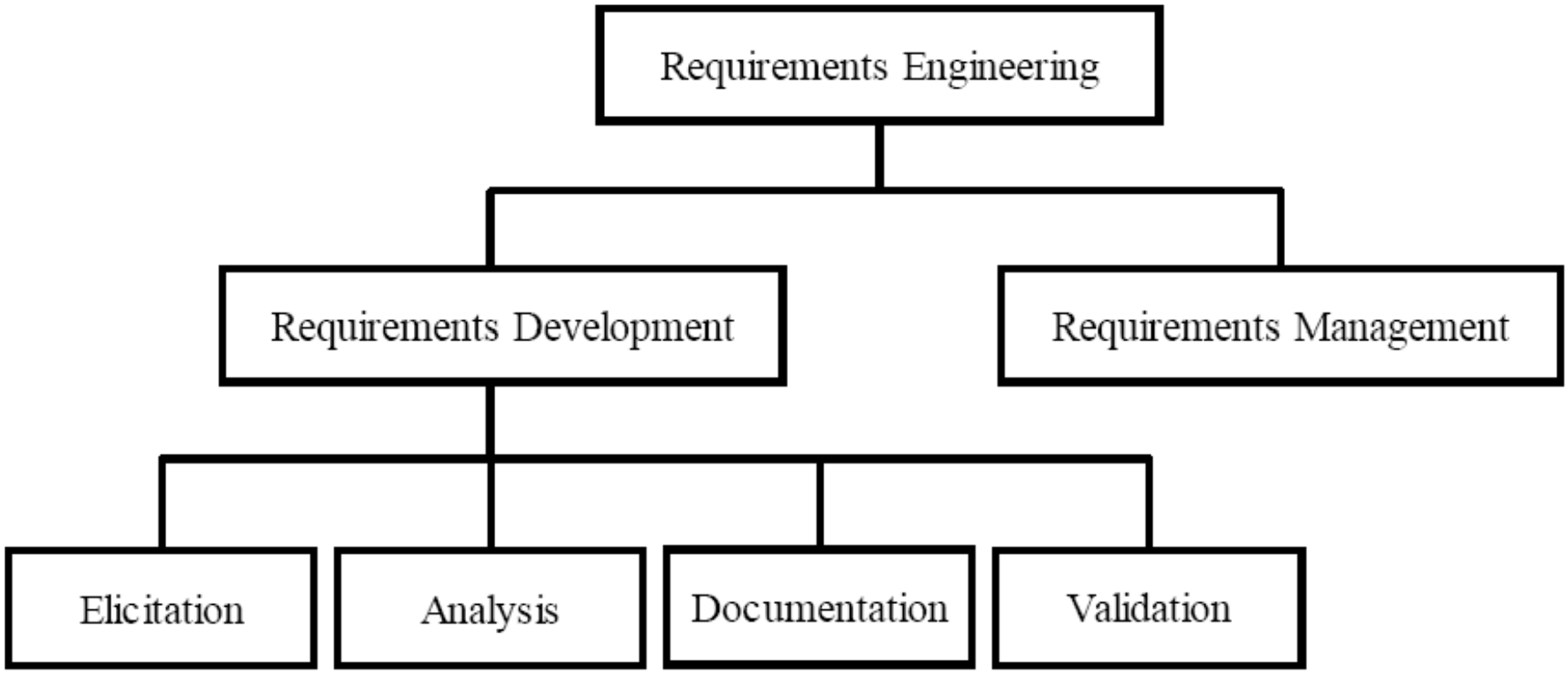

DEVELOPER'S WORLD
USERS' WORLD
REAL WORLD
SYSTEM 2'S WORLD
SYSTEM'S WORLD



Kotonya and Sommerville, 2002

Examples of inputs

Existing system knowledge	The systems works with MySQL v. abc – xyz and all browsers supporting JavaScript.
Stakeholders, users	The user can view all previous orders.
Organization(s)	The organization is using Windows Exchange and the new system will be compatible with it.
Regulation & Law	The materials will be processed in a way compatible with the Environmental Protection Act (86/2000)
Application domain knowledge	Each item will have a unique id, books will have a 10 digit ISBN.



Nikula, 2004, p. 28