

Chapter 2

Introduction to Requirements Management

- Software requirements
- Requirements management
- The problem domain
- The solution domain

What is a Software Requirement?

It is a software capability that

- is needed by the user to solve a problem to achieve an objective, and
- must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documentation

What is Requirements Management (RM)?

- A systematic approach for
 - eliciting,
 - organizing, and
 - documenting

the requirements of the system, and

- a process that establishes and maintains **agreement** between the customer and the project team on the changing requirements of the system.

Key Concepts in RM

- The ability to **elicit** the requirements from users and stakeholders is a crucial skill.
- Since hundreds, if not thousands, of requirements are likely to be associated with a system, it's important to **organize** them.
- **documenting** the requirements is necessary to support effective communication among the various stakeholders. The requirements have to be **recorded in an accessible medium**: a document, a model, a database, or a list on the whiteboard.

Example of RM activities

- Which project team member is responsible for requirement # 125, (analyze, modify, change, ..etc)
- If requirement # 125 is modified what other requirements will be effected?
- Which part of the software will satisfy requirement #125 .. who is going to that ..how do we test it?

Formal Requirements Management

Organized and formal processes of requirements management can be found in

- Capability Maturity Model (CMM)
- ISO 9000 for quality management standards

RM for all Types of Software Applications

- **IS/IT:** Information systems and other applications developed for use within a company, e.g., the payroll system of a certain company
- **ISV:** Software developed and sold as commercial products., e.g. MS Word, Excel
 - Companies developing this type of software are referred to as independent software vendors (ISVs).
- **Embedded applications:** Software that runs on computers embedded in other devices, machines, or complex systems, e.g. software in cell phones, automobile

The Road Map

- **Problem domain** is related to the
 - Stakeholder needs

- **Solution domain** is related to the
 - Features of the system
 - Software requirements

The Problem Domain

- Most successful requirements journeys begin with a trip to the land of the problem.
- **This problem domain** is the home of real users and other stakeholders, people whose needs must be addressed in order for us to build the perfect system.
- These users have **business or technical problems** that they need our help to solve.
- Therefore, it becomes our problem to understand **their problems**, in **their culture** and **their language**, and to build systems that meet **their needs**.

Stakeholders Needs

- It is also our responsibility to understand the needs of users and other stakeholders whose lives will be affected by our solution.

Moving Toward the Solution domain

- A definition of a system in terms of the **features** of the system and **the software requirements** that will drive its design and implementation.

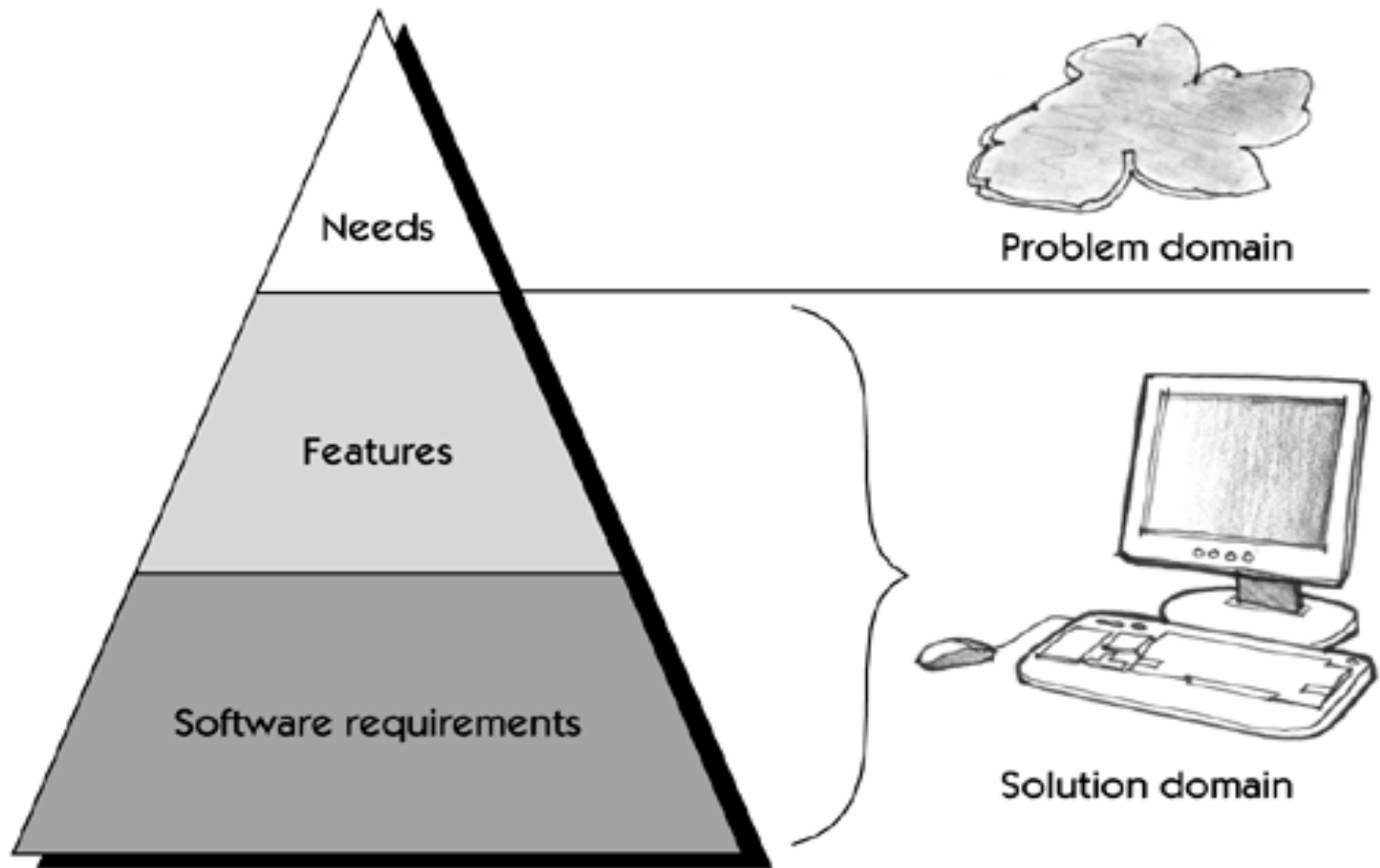
Features of the System

- **A feature** is a service provided by the system that fulfills one or more stakeholder needs.
- Simple descriptions, in the user's language, that we will use as labels to communicate with the user how our system addresses the problem.
- Examples:
 - "The car will have power windows."
 - "The program will allow Web-enabled entry of sales orders."

Software Requirements

- Once we have established the feature set and have gained agreement with the customer, we move to defining the more specific requirements needed in the solution.
- Then we can be certain that the system we develop will deliver the features we promised.
- Since the features address one or more stakeholder needs, we will have addressed those needs directly in the solution.

Overview of the Problem Domain and the Solution Domain



Key points

- A **requirement** is a capability that is imposed on the system.
- **Requirements management** is a process of systematically **eliciting, organizing, and documenting requirements** for a complex system.
- Our challenge is to **understand users' problems** in their culture and their language and to build systems that meet their needs.
- A **feature** is a service that the system provides to fulfill one or more stakeholder needs.