PRAXIS®

Introduction to Requirements

Requirements Engineering and Management Seminary

Module II

Goals

- At the end of the course, participants will be able to:
  - Understand the definition of requirement.
  - Describe functional and non-functional requirements.
  - Distinguish between customer and system requirements.
  - Describe the concept of traceability.
  - Describe the relation between requirements and software development activities.
  - Understand the importance of requirements to business goals.

Agenda

- » Definition of requirement
- » Types of requirements
- » Levels of requirements
- » Traceability
- » Requirements and software development lifecycle
- » Business goals and system requirements

Introduction (1)

- Requirements are the primary metric to measure the success of a system development.
- Many software project failures are attributed to requirements engineering issues:
  - Poorly documented requirements.
  - Requirements that are impossible to satisfy.
  - Requirements that fail to meet the users needs.
  - Requirements creep (gradual inclusion of unanticipated, undocumented and poorly considered requirements).
- Errors that occur at requirements stages turn out to be the most difficult and costly to fix.

Introduction (2)

- Benefits of appropriate requirements definition:
  - Cost savings.
  - Shorten the software development lifecycle.
  - Get a system that meets business goals.
  - Boost the team's productivity.
  - Reduce rework and conflicts arising from unclear and ambiguous requirements.

Definition of Requirement

- A requirements is defined as “a condition or capability to which a system must conform”.

- Attributes of a requirement:
  - Complete.
  - Clear.
  - Properly stated.
  - Consistent.
  - Unique.
  - Verifiable.
  - Traceable.

Types of Requirements (1)

- Functional requirements
  - Specify actions that the system must be able to perform, without taking physical constraints into consideration.
  - Are often described in a use-case model and in use cases.
  - Specify the input and output behavior of a system.
- Non-Functional requirements
  - Describe attributes of the system.
  - Describe attributes of the system environment.
  - Some of these may be captured in use cases.
  - Are often described in supplementary specifications.

Types of Requirements (2)

- A complete definition of functional requirements and non-functional requirements may be packaged together to define a software requirements specification for a particular system.
- The FURPS+ Model is used to describe the major types of requirements with subtypes.

Types of Requirements (3)

- FURPS+ Model means:
  - Functionality.
  - Usability.
  - Reliability.
  - Performance.
  - Supportability.

- "+" reminds to include such requirements as:
  - Design requirements.
  - Implementation requirements.
  - Interface requirements.
  - Physical requirements.

Types of Requirements (4)

- Functionality
  - Functional requirements may include:
    - Feature sets.
    - Capabilities.
    - Security.
- Usability
  - Usability requirements may include such subcategories as:
    - Human factors.
    - Aesthetics.
    - Consistency in the user-interface.
    - Online and context-sensitive help.
    - Wizards and agents.
    - User documentation.
    - Training materials.

Types of Requirements (5)

- Reliability
  - Reliability requirements to be considered are:
    - Frequency and severity of failure.
    - Recoverability.
    - Predictability.
    - Accuracy.
    - Mean time between failure (MTBF).

- Performance
  - A performance requirement imposes conditions on functional requirements.

Types of Requirements (6)

- Performance (cont….)
  - For a given action, it may specify performance parameters for:
    - Speed.
    - Efficiency.
    - Availability.
    - Accuracy.
    - Throughput.
    - Response time.
    - Recovery time.
    - Resource usage.

Types of Requirements (7)

- Supportability
  - Supportability requirements may include:
    - Testability.
    - Extensibility.
    - Adaptability.
    - Maintainability.
    - Compatibility.
    - Configurability.
    - Serviceability.
    - Installability.
    - Localizability (internationalization).

Types of Requirements (8)

- Design requirement
  - A design requirement, often called a **design constraint**, specifies or constraints the design of a system.

- Implementation requirement
  - An implementation requirement specifies or constrains the coding or construction of a system.
  - Examples are:
    - Required standards.
    - Implementation languages.
    - Policies for database integrity.
    - Resource limits.
    - Operation environments.

Types of Requirements (9)

- Interface requirement
  - An interface requirement specifies:
    - An external item with which a system must interact.
    - constraints on formats, timings, or other factors used by such an interaction.
- Physical requirement
  - A physical requirement specifies a physical characteristic that a system must possess.
  - Examples are:
    - Material.
    - Size.
    - Shape.
    - Weight.
  - It can be used to represent hardware requirements, such as the physical network configurations required.

Levels of Requirements (1)

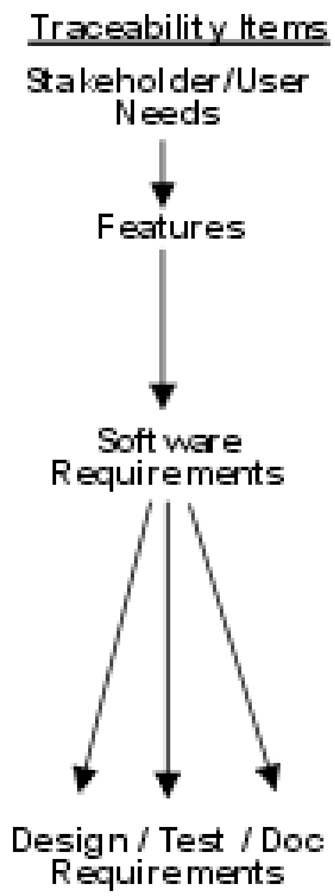
- Customer requirements
  - Are the customer and end users needs.
  - Are expressed in common language.
- System requirements
  - Are the requirements that the system must conform to satisfy the customer and end user needs.
  - Are expressed in technical language.
- To define the system means to translate and organize the understanding of stakeholder needs (customer requirements) into a meaningful description of the system to be built (system requirements).

Levels of Requirements (2)

» Levels of requirements help to separate the different levels of abstraction and purposes of the requirements.

» Levels of requirements allow to perform effective requirements management.

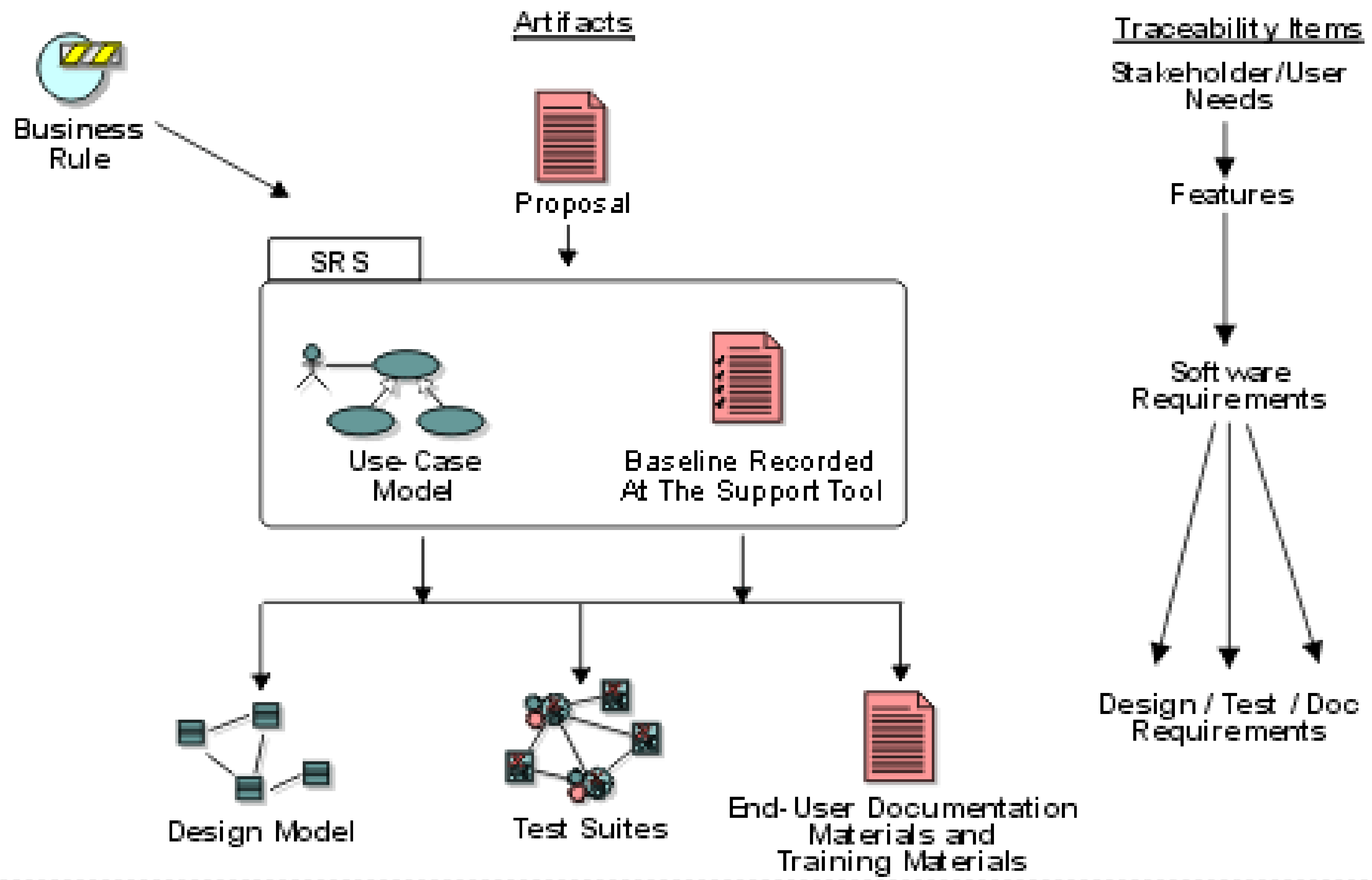
» All customer requirements must be traced to system requirements.



Traceability (1)

- Traceability is the ability to trace a project element to other related project elements, especially those related to requirements.
- Project elements involved in traceability are called **traceability items**.
- Typical traceability items include:
  - Different types of requirements.
  - Analysis and design model elements.
  - Test artifacts (test suites, test cases, etc.).
  - End-user support documentation and training material.

Traceability (2)



Traceability (3)

- The purpose of establishing **traceability** is to help:
  - Understand the source of requirements.
  - Manage the scope of the project.
  - Manage changes to requirements.
  - Assess the project impact of a change in a requirement.
  - Assess the impact of a failure of a test on requirements (i.e. if test fails the requirement may not be satisfied).
  - Verify that all requirements of the system are fulfilled by the implementation.
  - Verify that the application does only what it was intended to do.

Traceability (4)

- Traceabilities may be set up to help answer the following sample set of queries:
  - Show me user needs that are not linked to product features.
  - Show me the status of tests on all use cases in iteration #n.
  - Show me all supplementary requirements linked to tests whose status is untested.
  - Show me the results of all tests that failed, in order of criticality.
  - Show me the features scheduled for this release, which user needs they satisfy, and their status.

Requirements and Software Development Lifecycle

» Requirements are the primary input for **Analysis & Design** activities.

» **Test** activities validate the system against requirements.

» Requirements are used in the definition of the **Test** mission, and in the subsequent test and evaluation activities.

» Requirements are important input to the **Project Management** activities.

Business Goals and Requirements

- It is necesary to understand the purpose of the business and how it works.
- Business goals specify the purpose of the business.
- Requirements must support business goals.
- Business goals must guide the definition of requirements.
- Business goals serve as criteria for requirements completeness.

PRAXIS®

» The end