

The essential software requirements

Chapter 1

Types of Requirements (pg. 7)

Terms (definitions given on next slide):

- Business requirement
- Business rule
- Constraint
- External interface requirement
- Feature
- Functional requirement
- Nonfunctional requirement
- Quality attribute
- System requirement
- User requirement

Types of Requirements (pg. 7)

TABLE 1-1 Some types of requirements information

Term	Definition
Business requirement	A high-level business objective of the organization that builds a product or of a customer who procures it.
Business rule	A policy, guideline, standard, or regulation that defines or constrains some aspect of the business. Not a software requirement in itself, but the origin of several types of software requirements.
Constraint	A restriction that is imposed on the choices available to the developer for the design and construction of a product.
External interface requirement	A description of a connection between a software system and a user, another software system, or a hardware device.
Feature	One or more logically related system capabilities that provide value to a user and are described by a set of functional requirements.
Functional requirement	A description of a behavior that a system will exhibit under specific conditions.
Nonfunctional requirement	A description of a property or characteristic that a system must exhibit or a constraint that it must respect.
Quality attribute	A kind of nonfunctional requirement that describes a service or performance characteristic of a product.
System requirement	A top-level requirement for a product that contains multiple subsystems, which could be all software or software and hardware.
User requirement	A goal or task that specific classes of users must be able to perform with a system, or a desired product attribute.

Requirement

“A statement of a customer need or objective, or of a condition or capability that a product must possess to satisfy such a need or objective. A property that a product must have to provide value to a stakeholder.”

Glossary, page 601

Relationships among Types of Requirement Information (pg. 8)

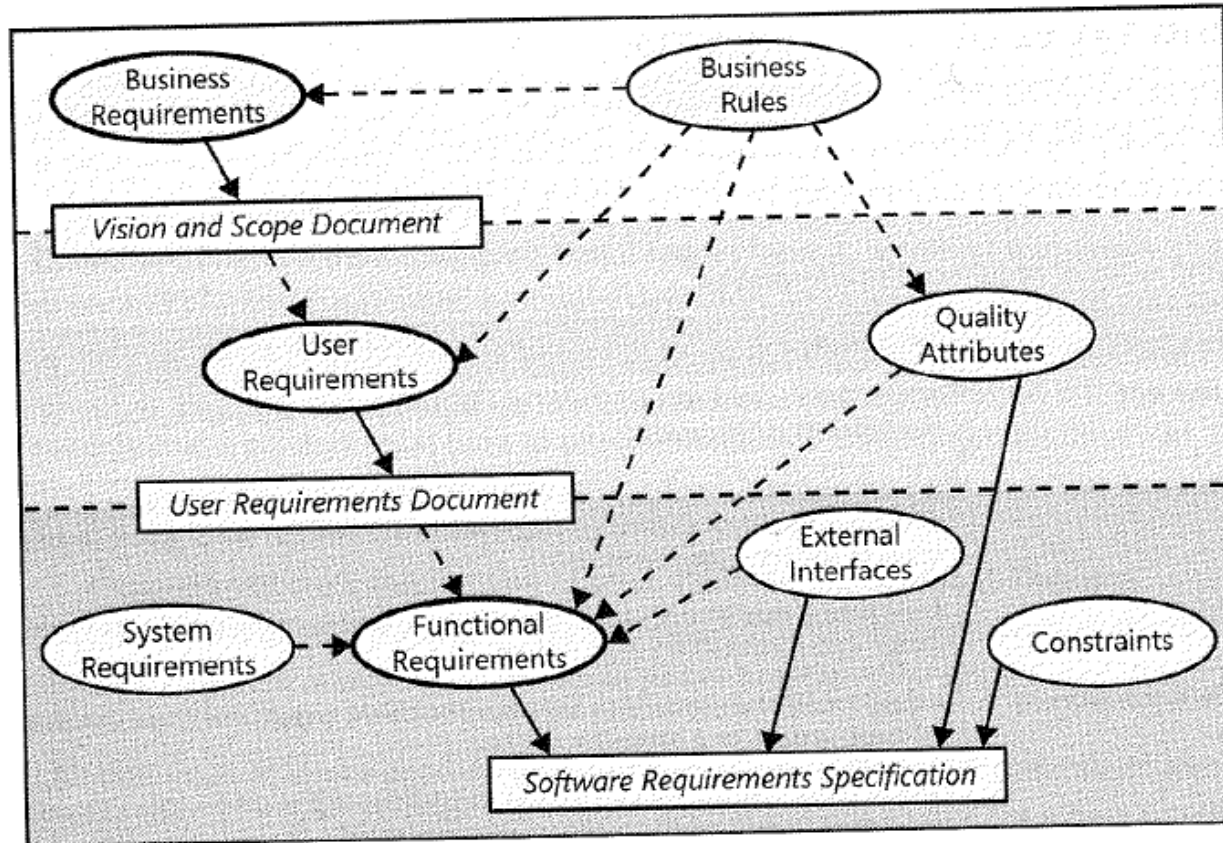


FIGURE 1-1 Relationships among several types of requirements information. Solid arrows mean "are stored in"; dotted arrows mean "are the origin of" or "influence."

Relationships among Features, User and Functional Requirements (pg. 11)

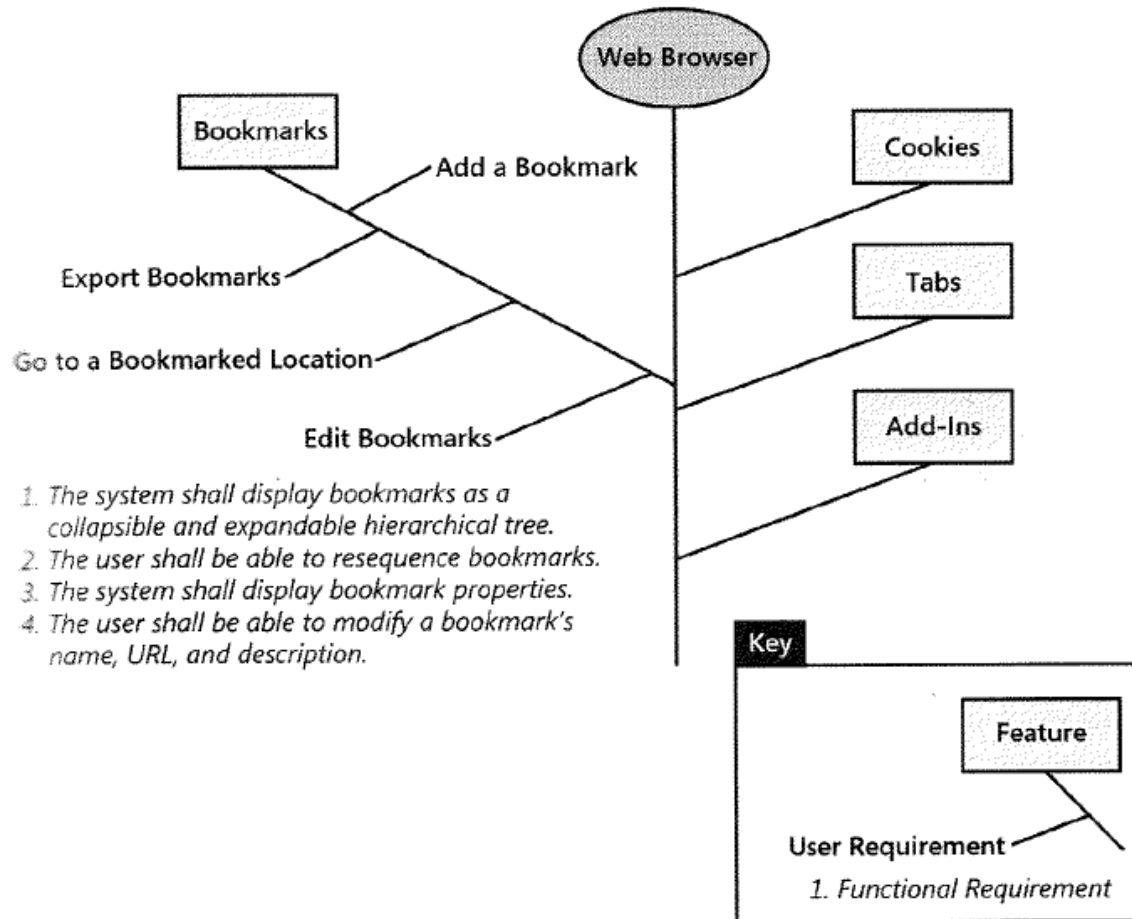


FIGURE 1-2 Relationships among features, user requirements, and functional requirements.

Stakeholders Participating in Requirements Development (pg. 12)

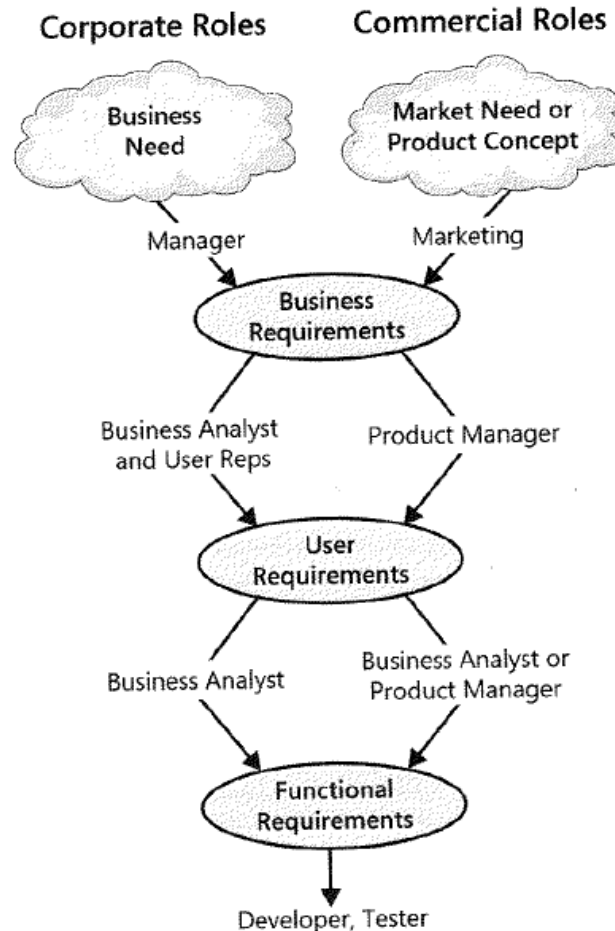


FIGURE 1-3 An example of how different stakeholders participate in requirements development.

Subdisciplines of Requirements Engineering (pg. 15)

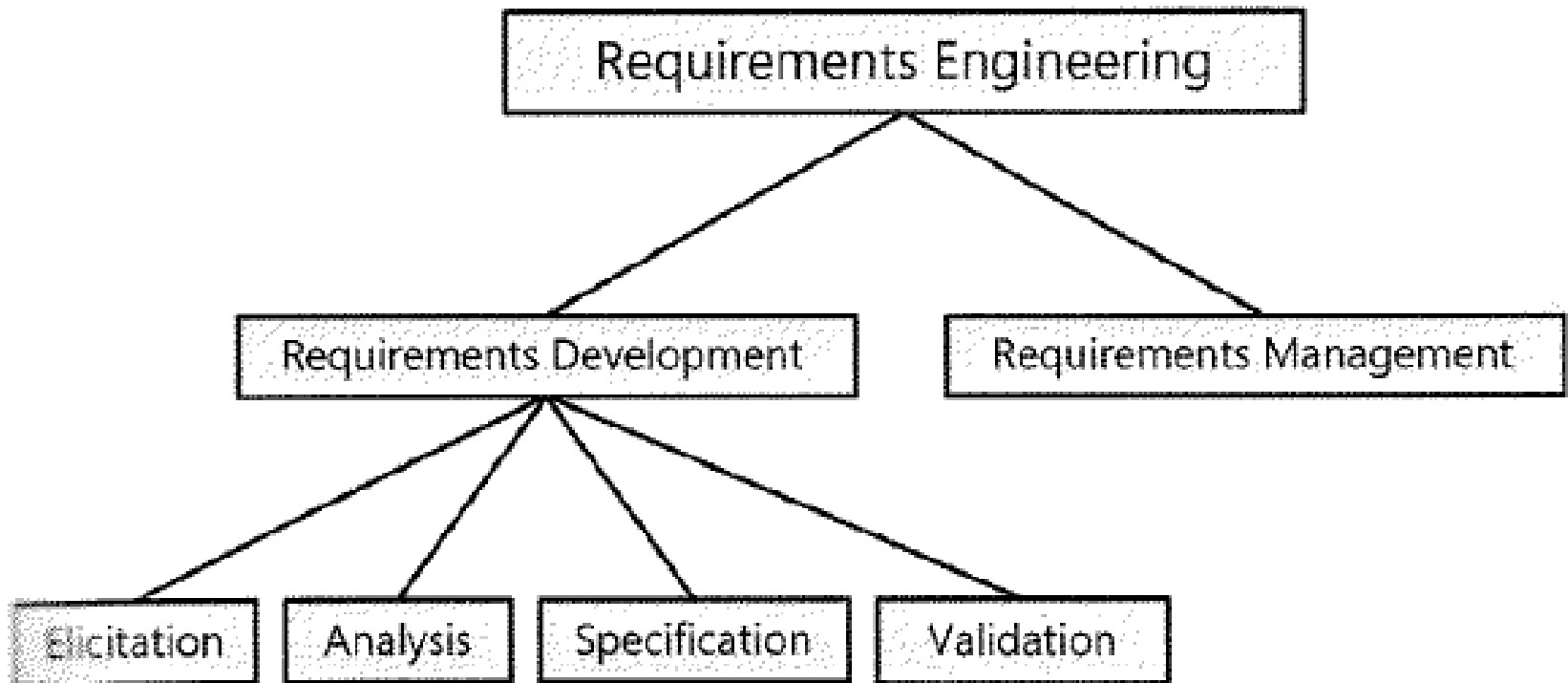


FIGURE 1-4 Subdisciplines of software requirements engineering.

Boundary Between Development and Management (pg. 18)

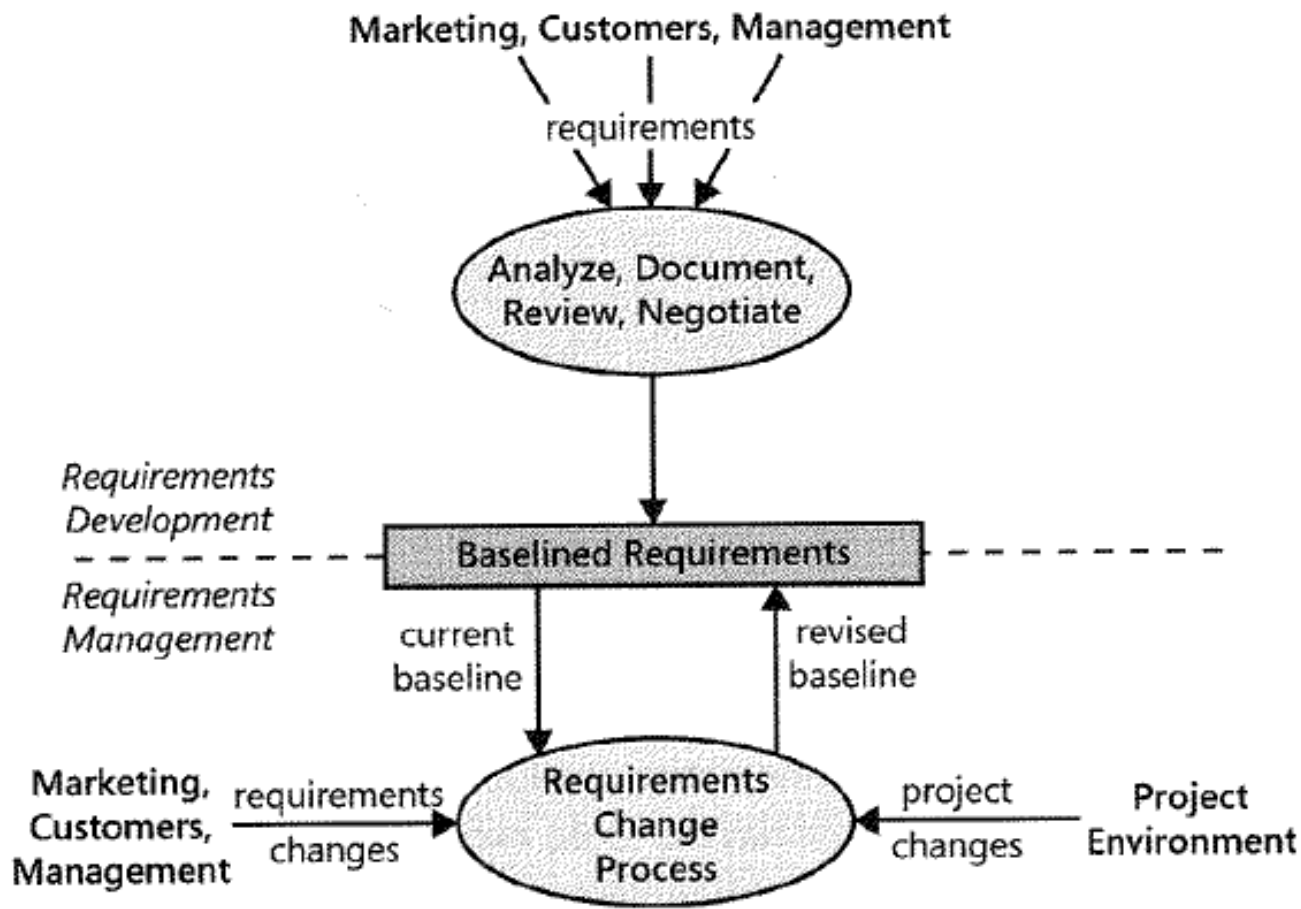


FIGURE 1-5 The boundary between requirements development and requirements management.

Common Requirements Risks (pg. 20-22)

Common requirements risks:

- Insufficient user involvement
- Inaccurate planning
- Creeping user requirements
- Ambiguous requirements
- Gold plating
- Overlooked stakeholders

Benefits of Doing Requirements Engineering Tasks (pg. 20-22)

Benefits:

- Fewer defects in requirements and in the delivered product
- Reduced development rework
- Faster development and delivery
- Fewer unnecessary features and unused features
- Lower enhancement costs
- Fewer miscommunications
- Reduced scope creep
- Reduced project chaos
- Higher customer and team member satisfaction
- Products that do what they're supposed to do