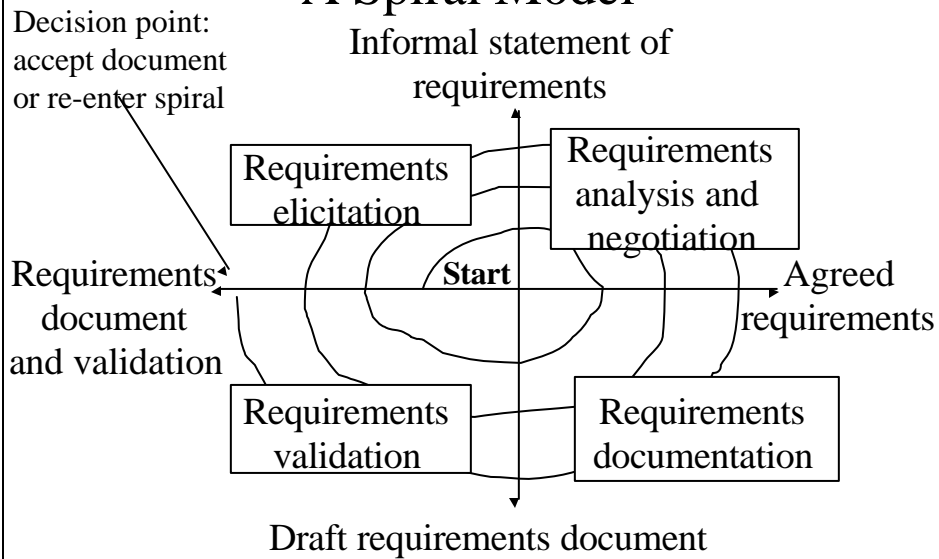
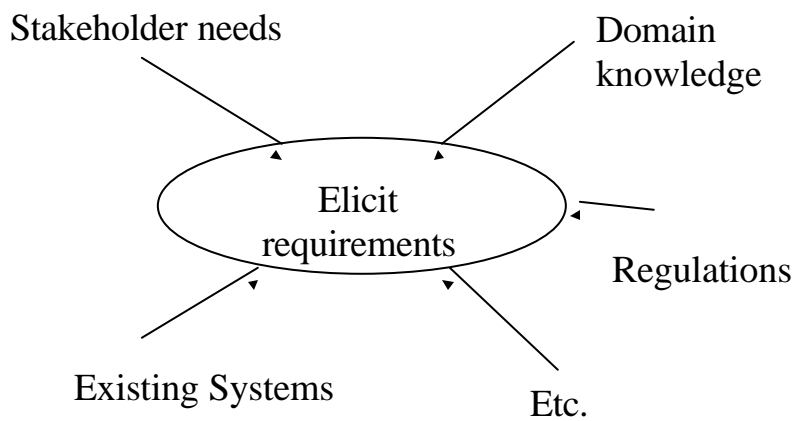


Requirements Engineering Process-- A Spiral Model



Eliciting Requirements



Requirements--What are They?

- A requirement is a description of a system feature, capability, or constraint.
- Requirements generally focus on *what* a system should do, rather than *how* it should do it.
- A requirement should be “testable”.
- Classes of Requirements:
 - functional
 - nonfunctional (constraints)
- Requirements Priority
 - essential (“shalls”)
 - highly desirable (“shoulds”)
 - desirable but low priority

Types of Requirements

- Physical environment
- Interfaces
- Users and human factors
- functionality
- performance
- documentation/training
- data
- resources
- security
- reliability
- portability
- maintenance
- etc.

Elicited Requirements--An Example

- Consider a library automation system. Here are some examples of possible elicited requirements:
 - The system shall maintain records of all library materials including books, serials, newspapers and magazines, video and audio tapes, reports, and CD-ROMS.
 - The system shall allow users to search for an item by title, author or ISBN.
 - The system's interface shall be via a web browser.
 - The system shall support 20 transactions per second.
 - The system shall be easy to use.

Analyzing/Negotiating Requirements

- Potential problems with elicited requirements
 - inaccurate
 - incomplete
 - ambiguous
 - inconsistent
 - unnecessary
 - infeasible
 - untestable
- Identified problems with elicited requirements should be negotiated with stakeholders

Documenting Requirements

- Requirements Documents go by various names:
 - requirements document
 - specification document
 - system requirements specification (SRS)
- Sometimes there will be two documents
 - requirements definition--customer-oriented
 - requirements specification--developer oriented

Typical Elements of a Requirements Document

- Listing of functional requirements
- Listing of nonfunctional requirements
- interface requirements
- hardware specification
- essential application domain information
- system models
 - conceptual model
 - use-case analyses
 - basic architectural model

Expressing Requirements

- Informal Specification
 - natural language
 - ad hoc models/diagrams
- Formal Specification
 - mathematically based notations/description languages
 - Well defined modeling techniques
 - Petri nets
 - FSMs
 - UML

Requirements Validation

- Validation is the process of establishing that the requirements specification is accurate, consistent, and complete with respect to the stakeholder's needs.
- Note that it is not possible to formally validate a requirements specification with respect to the stakeholders' expectations
- The most common form of requirements validation is a Requirements Review.
 - Formal meeting similar to a walkthrough
 - review team made up of various stakeholders

Informal Requirements Validation

- General problems to look for
 - incompleteness
 - ambiguity
 - violation of standards
 - redundancy
 - inconsistency (conflict)
 - traceability
 - is this requirement traceable to one or more stakeholder needs
 - testability