Requirements validation

Requirements validation

- It is the process of checking that requirements define the system that the customer really wants.
- Requirements error costs high so validation is important
  - Fixing a requirements error after delivery may cost up to 100 times the cost of fixing an implementation error.

Requirements checking

- **Validity:** Does the system provide the functions which best support the customer's needs?
- **Consistency:** Are there any requirements conflicts?
- **Completeness:** Are all functions required by the customer included?
- **Realism:** Can the requirements be implemented given available budget and technology
- **Verifiability:** Can the requirements be checked?

Requirements validation techniques

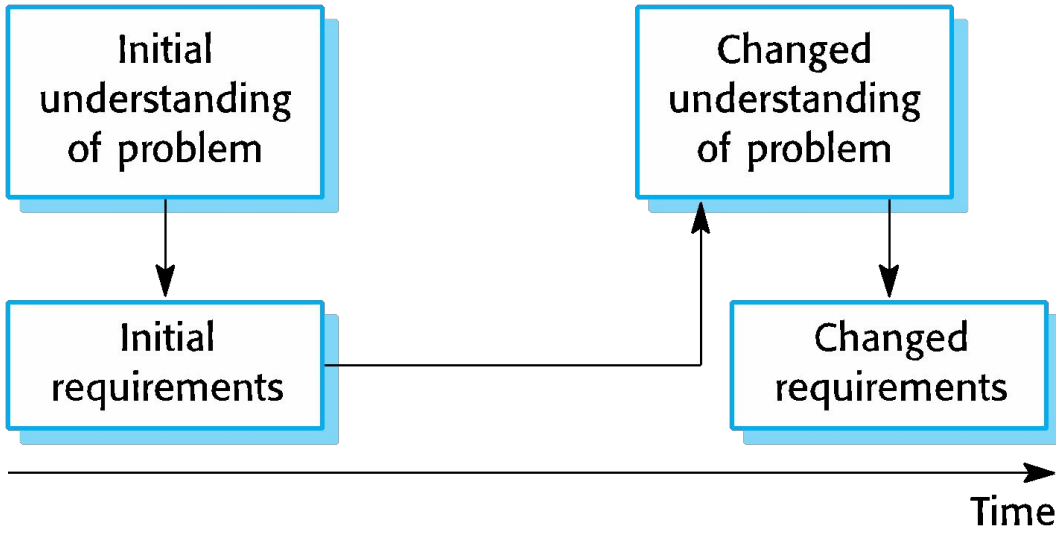
- Requirements reviews
  - Systematic manual analysis of the requirements.
- Prototyping
  - Using an executable model of the system to check requirements.
- Test-case generation
  - Developing tests for requirements to check testability.

Requirements change

Changing requirements

- **The business and technical environment of the system always changes after installation.**
  - New hardware may be introduced, it may be necessary to interface the system with other systems, business priorities may change (with consequent changes in the system support required), and new legislation and regulations may be introduced that the system must necessarily abide by.
- **The people who pay for a system and the users of that system are rarely the same people.**
  - System customers impose requirements because of organizational and budgetary constraints. These may conflict with end-user requirements and, after delivery, new features may have to be added for user support if the system is to meet its goals.

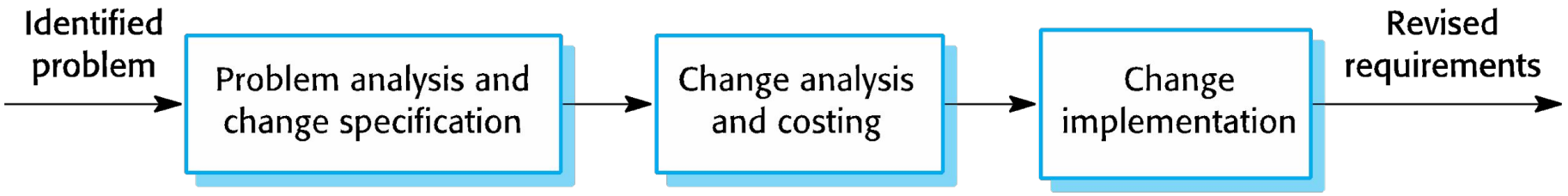
Requirements evolution



Requirements management

- Process of managing changing requirements during the requirements engineering process and system development.
- New requirements emerge as a system is being developed and after it has gone into use.
- Need to keep track of individual requirements and maintain links between dependent requirements so that you can assess the impact of requirements changes. You need to establish a formal process for making change proposals and linking these to system requirements.

Requirements change management



Requirements change management

- Deciding if a requirements change should be accepted
  - *Problem analysis and change specification*
    - During this stage, the problem or the change proposal is analyzed to check that it is valid. This analysis is fed back to the change requestor who may respond with a more specific requirements change proposal, or decide to withdraw the request.
  - *Change analysis and costing*
    - The effect of the proposed change is assessed using traceability information and general knowledge of the system requirements. Once this analysis is completed, a decision is made whether or not to proceed with the requirements change.
  - Change implementation
    - The requirements document and, where necessary, the system design and implementation, are modified. Ideally, the document should be organized so that changes can be easily implemented.

Summary

- **Software requirements:** Services that the system should provide and constraints on its operation and implementation.
- **Types of requirements**
  - User and system
  - Functional and non-functional
- **Requirements engineering process**
  - Feasibility study; elicitation and analysis; specification; validation; and management

Structured Vs Object-Oriented Paradigm

	Structured	Object-Oriented
Methodology	SDLC	Interative/Incremental
Focus	Process	Objects
Risk	High	Low
Reuse	Low	High
Suitable for	Well defined project with stable use requirements	Risky large projects with changing user requirements

Structured Vs Object-Oriented Analysis

- **Structured:** Uses DFD's, Decision Table/Tree and E-R diagrams.
- **Object-Oriented:** Use case and object model.
- **Use Case Model:** UML use cases and activity diagrams.
- **Object Model**: Find classes and their relations. Uses sequence and state machine diagrams. Object to ER mapping.