

Chapter-4

Requirements Engineering

Prepared by Asst. Prof. Bal Krishna Subedi
Tribhuvan University

Topics covered

- Functional and non-functional requirements
- The software requirements document
- Requirements specification
- Requirements engineering processes
- Requirements elicitation and analysis
- Requirements validation
- Requirements management

What is a functional requirement?

Functional requirements define the basic system behavior. Essentially, they are **what** the system does or must not do, and can be thought of in terms of how the system responds to inputs. Functional requirements usually define if/then behaviors and include calculations, data input, and business processes.



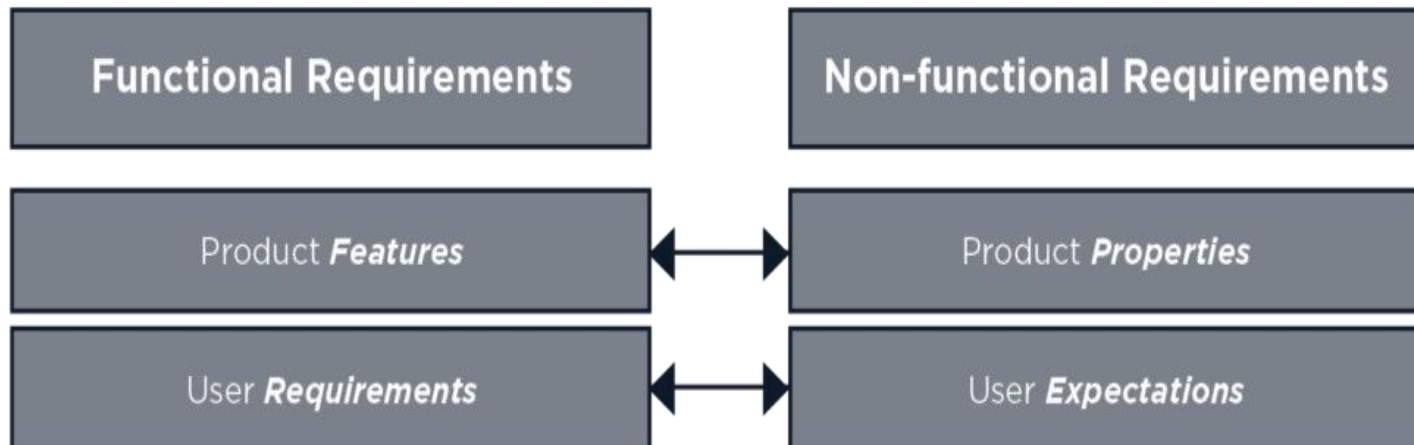
Functional requirements are features that allow the system to function as it was intended. Put another way, if the functional requirements are not met, the system will not work. Functional requirements are product **features** and focus on user **requirements**.

What is a non-functional requirement?

While *functional* requirements define what the system does or must not do, *non-functional* requirements specify **how** the system should do it. Non-functional requirements do not affect the basic functionality of the system (hence the name, *non-functional* requirements). Even if the non-functional requirements are not met, the system will still perform its basic purpose.

If a system will still perform without meeting the non-functional requirements, why are they important? The answer is usability. Non-functional requirements define system behavior, features, and general characteristics that affect the user experience. How well non-functional requirements are defined and executed determines how easy the system is to use, and is used to judge system performance. Non-functional requirements are product **properties** and focus on user **expectations**.

There are more examples below, but for now, think about a functional requirement that the system loads a webpage after someone clicks on a button. There should be a related non-functional requirement specifying how fast the webpage must load. Without it, the user experience and perception of quality are at risk if they are forced to wait too long, even though the functional requirement is fully met.



What is Software Requirement Specification - [SRS]?

A software requirements specification (SRS) is a document that captures complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineering phase.

Qualities of SRS:

- .Unambiguous
- .Complete
- .Correct
- .Consistent
- .Ranked for importance and/or stability
- .Verifiable
- .Modifiable
- .Traceable

Types of Requirements:

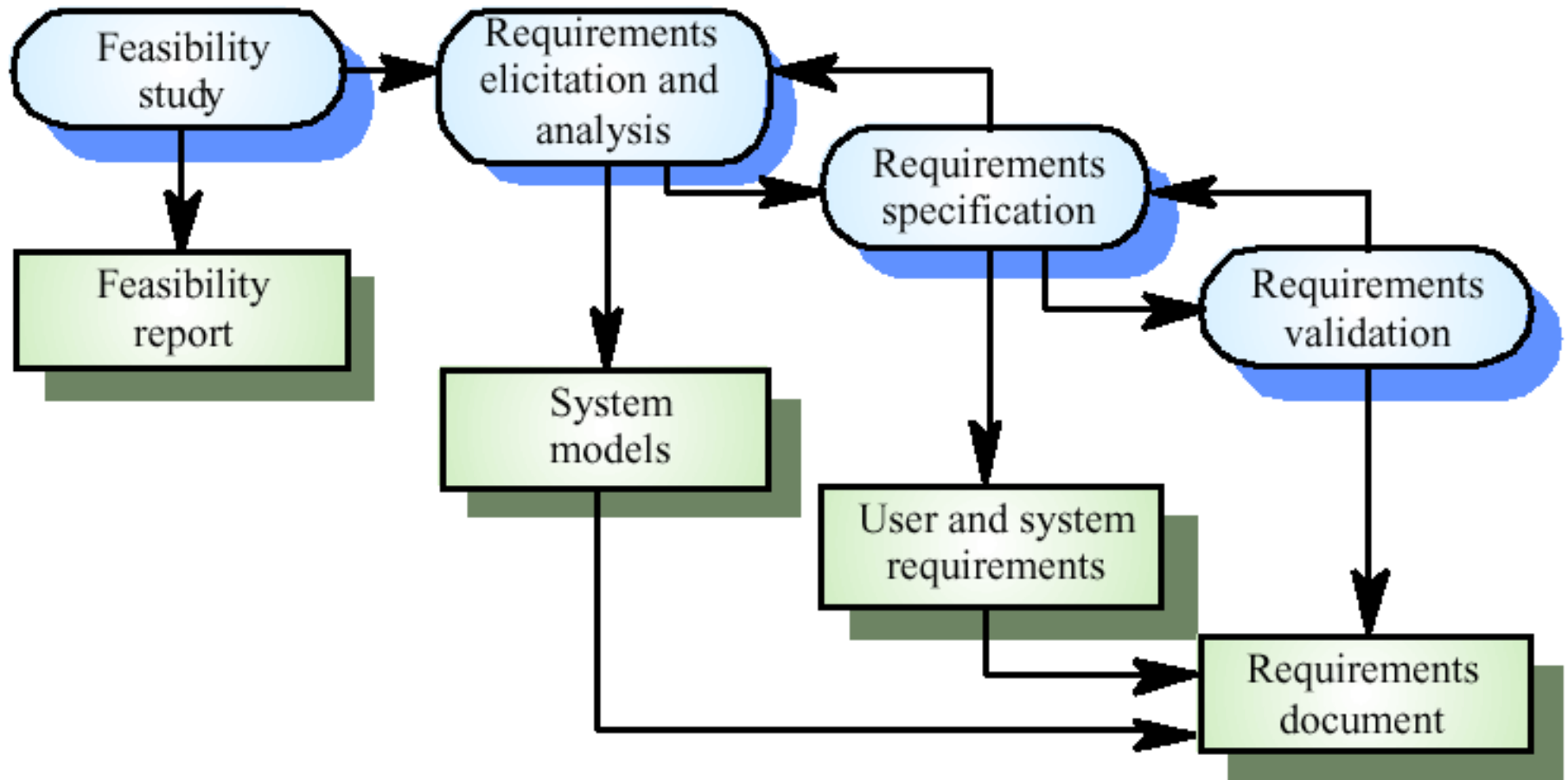
The below diagram depicts the various types of requirements that are captured during SRS.



Requirements engineering processes

- The processes used for RE vary widely depending on the application domain, the people involved and the organisation developing the requirements
- However, there are a number of generic activities common to all processes
 - Requirements elicitation
 - Requirements analysis
 - Requirements validation
 - Requirements management

The requirements engineering process



Feasibility studies

- A feasibility study decides whether or not the proposed system is worthwhile
- A short focused study that checks
 - If the system contributes to organisational objectives
 - If the system can be engineered using current technology and within budget
 - If the system can be integrated with other systems that are used

Feasibility study implementation

- Based on information assessment (what is required), information collection and report writing
- Questions for people in the organisation
 - What if the system wasn't implemented?
 - What are current process problems?
 - How will the proposed system help?
 - What will be the integration problems?
 - Is new technology needed? What skills?
 - What facilities must be supported by the proposed system?

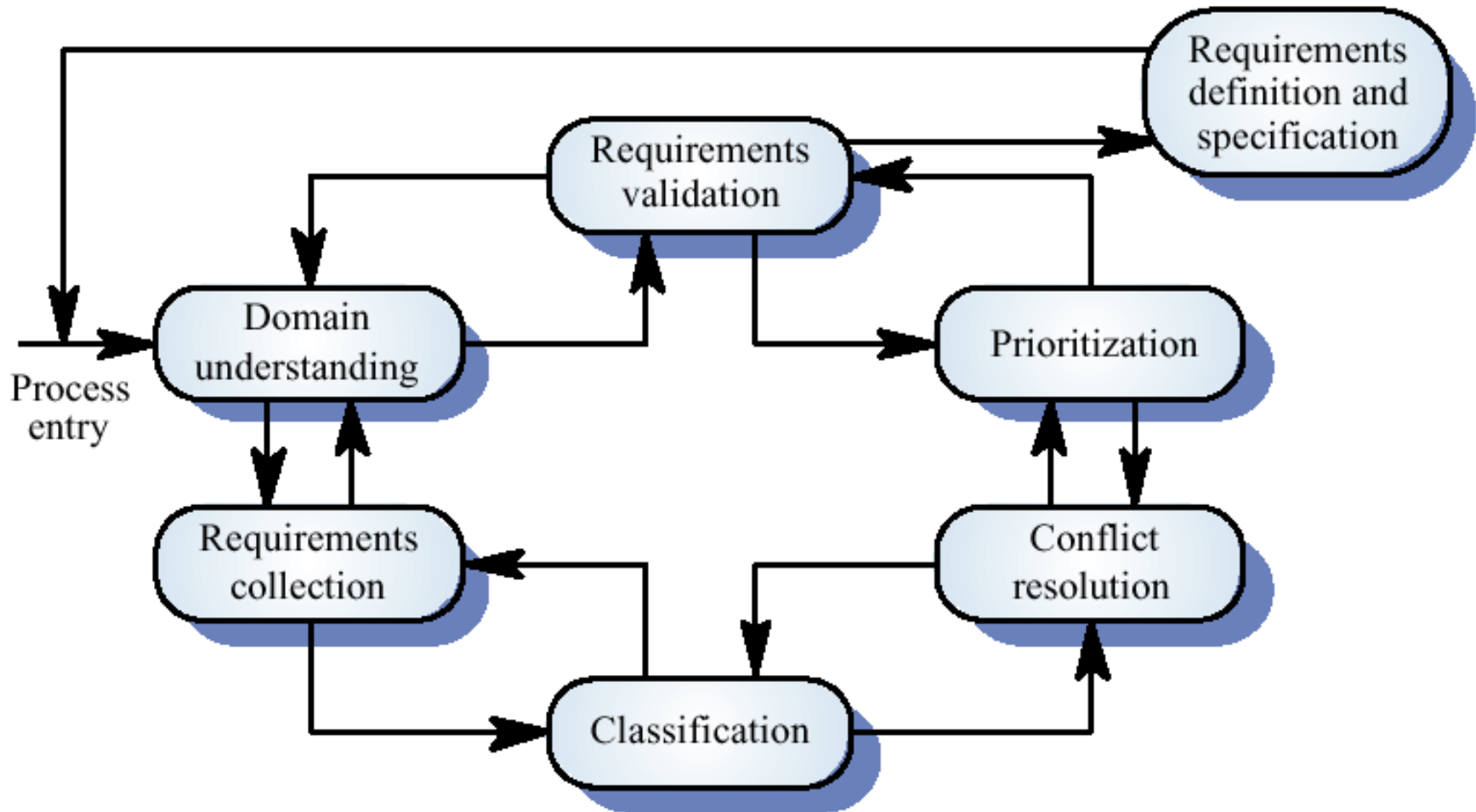
Elicitation and analysis

- Sometimes called requirements elicitation or requirements discovery
- Involves technical staff working with customers to find out about the application domain, the services that the system should provide and the system's operational constraints
- May involve end-users, managers, engineers involved in maintenance, domain experts, trade unions, etc. These are called *stakeholders*

Problems of requirements analysis

- Stakeholders don't know what they really want
- Stakeholders express requirements in their own terms
- Different stakeholders may have conflicting requirements
- Organisational and political factors may influence the system requirements
- The requirements change during the analysis process. New stakeholders may emerge and the business environment change

The requirements analysis process



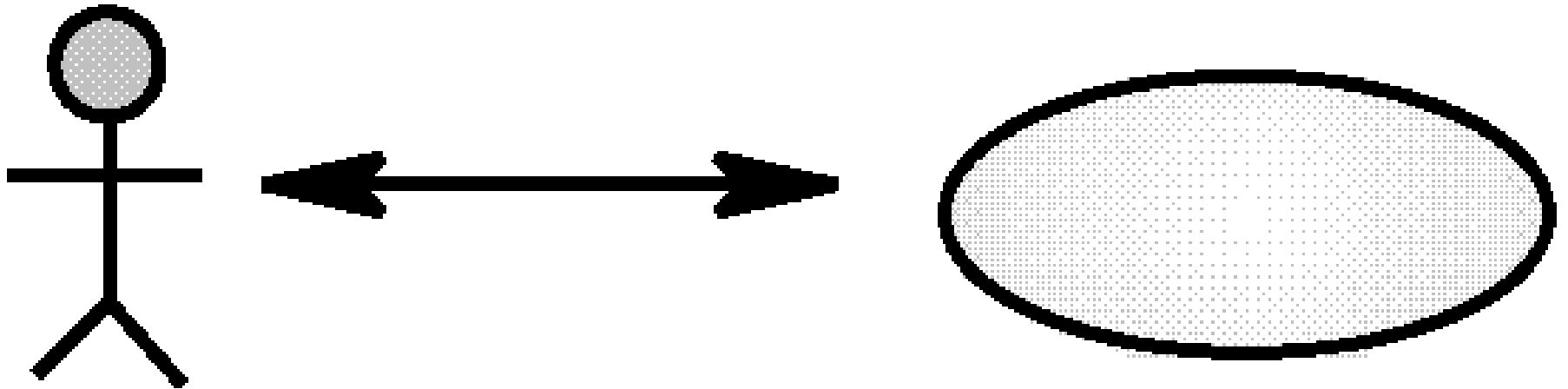
Process activities

- Domain understanding
- Requirements collection
- Classification
- Conflict resolution
- Prioritisation
- Requirements checking

Use cases

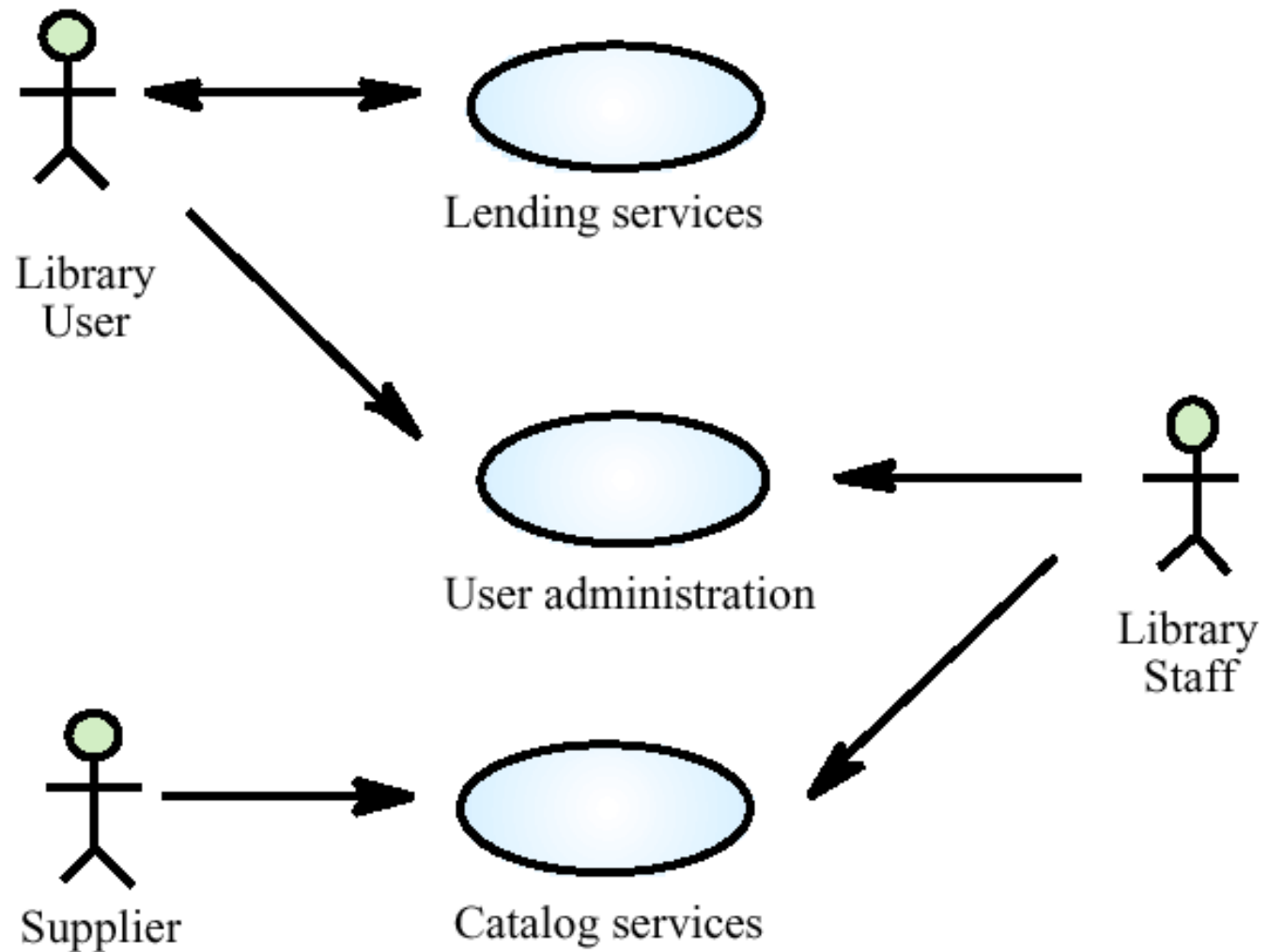
- Use-cases are a scenario based technique in the UML which identify the actors in an interaction and which describe the interaction itself
- A set of use cases should describe all possible interactions with the system
- Sequence diagrams may be used to add detail to use-cases by showing the sequence of event processing in the system

Lending use-case

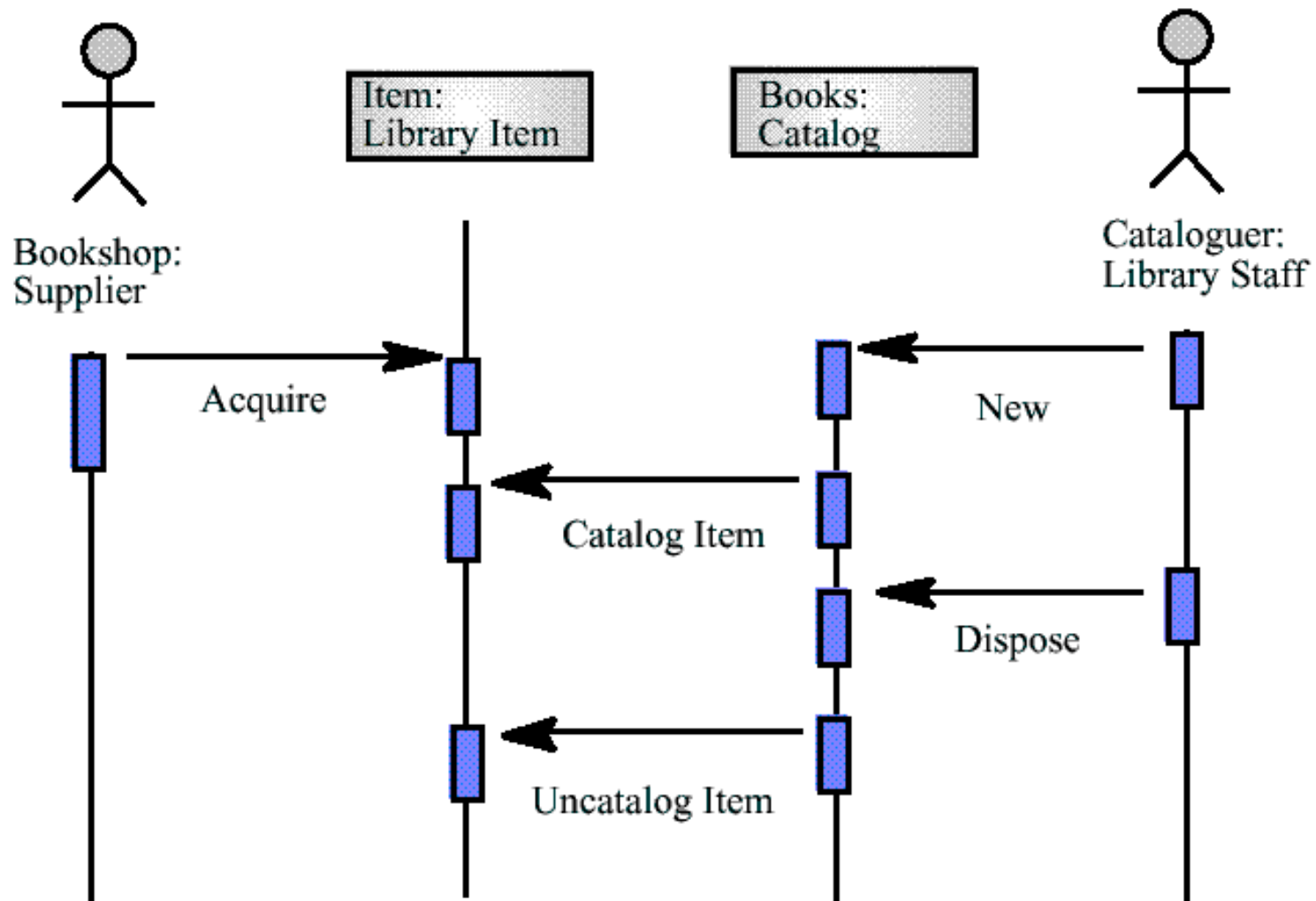


Lending services

Library use-cases



Catalogue management



Social and organisational factors

- Software systems are used in a social and organisational context. This can influence or even dominate the system requirements
- Social and organisational factors are not a single viewpoint but are influences on all viewpoints
- Good analysts must be sensitive to these factors but currently no systematic way to tackle their analysis

Example

- Consider a system which allows senior management to access information without going through middle managers
 - Managerial status. Senior managers may feel that they are too important to use a keyboard. This may limit the type of system interface used
 - Managerial responsibilities. Managers may have no uninterrupted time where they can learn to use the system
 - Organisational resistance. Middle managers who will be made redundant may deliberately provide misleading or incomplete information so that the system will fail

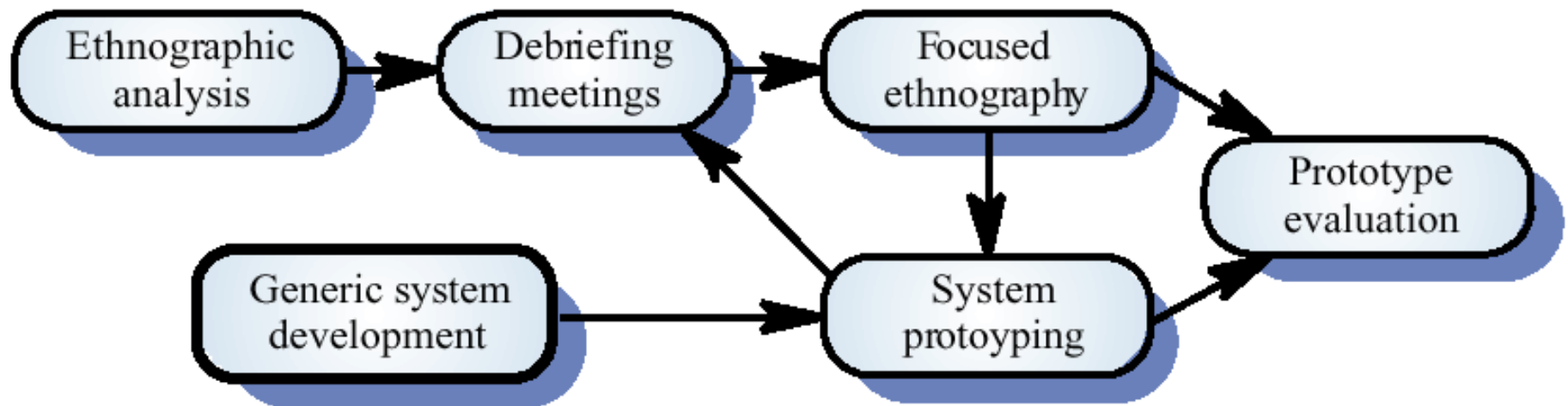
Ethnography

- A social scientists spends a considerable time observing and analysing how people actually work
- People do not have to explain or articulate their work
- Social and organisational factors of importance may be observed
- Ethnographic studies have shown that work is usually richer and more complex than suggested by simple system models

Focused ethnography

- Developed in a project studying the air traffic control process
- Combines ethnography with prototyping
- Prototype development results in unanswered questions which focus the ethnographic analysis
- Problem with ethnography is that it studies existing practices which may have some historical basis which is no longer relevant

Ethnography and prototyping



Scope of ethnography

- Requirements that are derived from the way that people actually work rather than the way I which process definitions suggest that they ought to work
- Requirements that are derived from cooperation and awareness of other people's activities

Requirements validation

- Concerned with demonstrating that the requirements define the system that the customer really wants
- Requirements error costs are high so validation is very 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. Covered in Chapter 8
- Test-case generation
 - Developing tests for requirements to check testability
- Automated consistency analysis
 - Checking the consistency of a structured requirements description

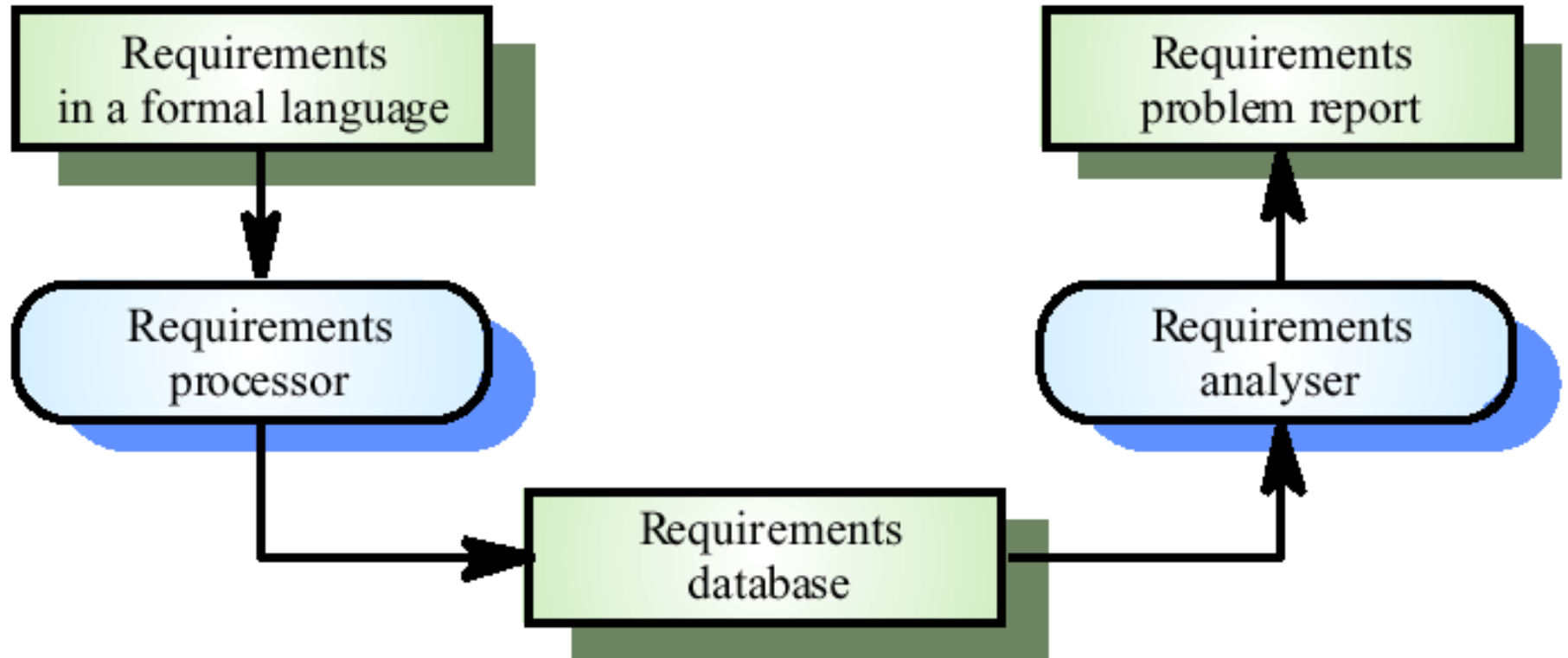
Requirements reviews

- Regular reviews should be held while the requirements definition is being formulated
- Both client and contractor staff should be involved in reviews
- Reviews may be formal (with completed documents) or informal. Good communications between developers, customers and users can resolve problems at an early stage

Review checks

- Verifiability. Is the requirement realistically testable?
- Comprehensibility. Is the requirement properly understood?
- Traceability. Is the origin of the requirement clearly stated?
- Adaptability. Can the requirement be changed without a large impact on other requirements?

Automated consistency checking



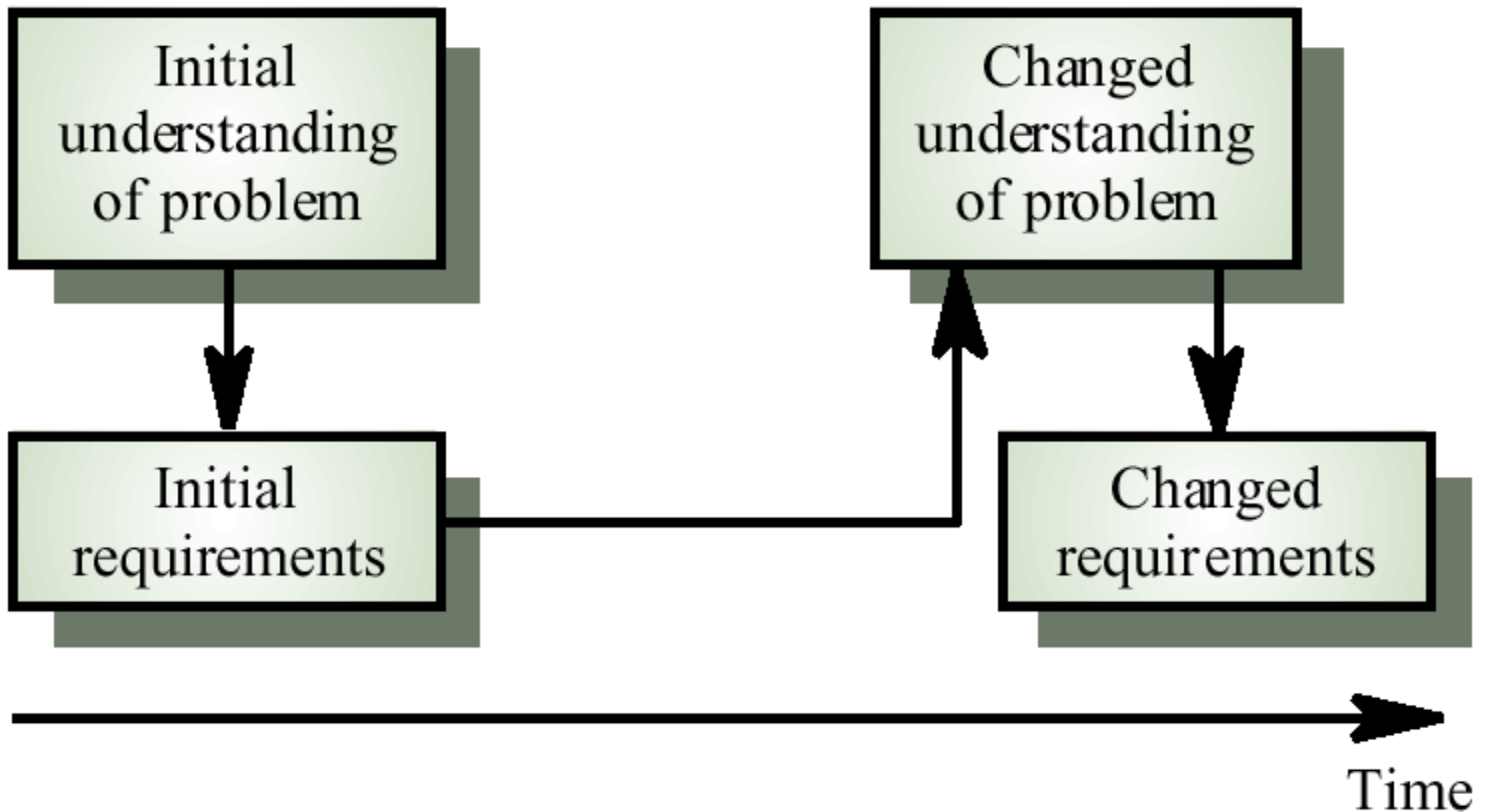
Requirements management

- Requirements management is the process of managing changing requirements during the requirements engineering process and system development
- Requirements are inevitably incomplete and inconsistent
 - New requirements emerge during the process as business needs change and a better understanding of the system is developed
 - Different viewpoints have different requirements and these are often contradictory

Requirements change

- The priority of requirements from different viewpoints changes during the development process
- System customers may specify requirements from a business perspective that conflict with end-user requirements
- The business and technical environment of the system changes during its development

Requirements evolution



Enduring and volatile requirements

- Enduring requirements. Stable requirements derived from the core activity of the customer organisation. E.g. a hospital will always have doctors, nurses, etc. May be derived from domain models
- Volatile requirements. Requirements which change during development or when the system is in use. In a hospital, requirements derived from health-care policy

Classification of requirements

- Mutable requirements
 - Requirements that change due to the system's environment
- Emergent requirements
 - Requirements that emerge as understanding of the system develops
- Consequential requirements
 - Requirements that result from the introduction of the computer system
- Compatibility requirements
 - Requirements that depend on other systems or organisational processes

Requirements management planning

- During the requirements engineering process, you have to plan:
 - Requirements identification
 - How requirements are individually identified
 - A change management process
 - The process followed when analysing a requirements change
 - Traceability policies
 - The amount of information about requirements relationships that is maintained
 - CASE tool support
 - The tool support required to help manage requirements change

Traceability

- Traceability is concerned with the relationships between requirements, their sources and the system design
- Source traceability
 - Links from requirements to stakeholders who proposed these requirements
- Requirements traceability
 - Links between dependent requirements
- Design traceability
 - Links from the requirements to the design

CASE tool support

- Requirements storage
 - Requirements should be managed in a secure, managed data store
- Change management
 - The process of change management is a workflow process whose stages can be defined and information flow between these stages partially automated
- Traceability management
 - Automated retrieval of the links between requirements

Requirements change management

- Should apply to all proposed changes to the requirements
- Principal stages
 - Problem analysis. Discuss requirements problem and propose change
 - Change analysis and costing. Assess effects of change on other requirements
 - Change implementation. Modify requirements document and other documents to reflect change

Requirements change management



Key points

- The requirements engineering process includes a feasibility study, requirements elicitation and analysis, requirements specification and requirements management
- Requirements analysis is iterative involving domain understanding, requirements collection, classification, structuring, prioritisation and validation
- Systems have multiple stakeholders with different requirements

Key points

- Social and organisation factors influence system requirements
- Requirements validation is concerned with checks for validity, consistency, completeness, realism and verifiability
- Business changes inevitably lead to changing requirements
- Requirements management includes planning and change management