

Requirements Engineering (RE)

What is goal of RE process?

The main goal is to create and maintain system requirements document.

What activities (subprocess) are involved?

1. Feasibility study: Is this system useful to business?

2. Elicitation & Analysis: discovering the requirements from stakeholders

3. Req. specification: Converting requirements into standard form

4. Requirement Validation: Does these requirements meet customer needs?

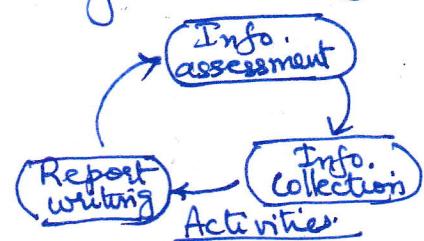
Each sequence of activities can develop requirements at different level of details. Thus could be modelled as evolution any process (Spiral) model

1. What is feasibility study?

This study should answer

- Does system meet the objectives of business organizations?
- Can the system be implemented with current technology?
- Given cost (budget) and within time?
- Can system work with existing already working systems in the organization?

What is output of feasibility study?
Feasibility report (document)



2. What is the purpose of Requirement Elicitation & Analysis?
Software engineers interact with stakeholders to know (discover) about domain, system services, constraints on services (performance reqs.), hardware constraints etc.

What is the output of Elicitation & Analysis?

Preliminary version of system requirements document.
(It is incomplete and still missing some sections.)
Writing requirements on cards may be more effective.

Spiral Model of Requirements Engineering Process

Requirement Specification (Analysis)

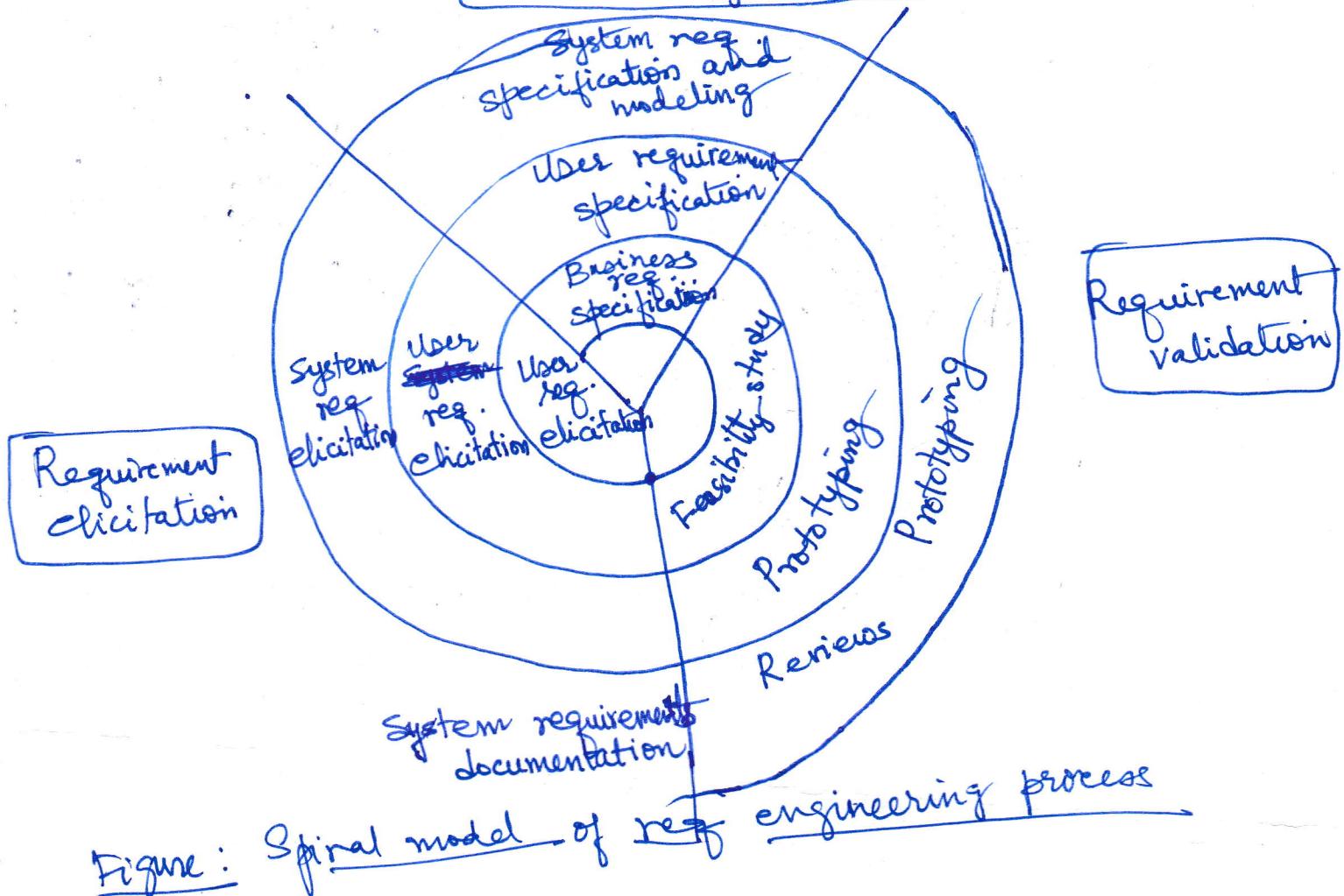


Figure : Spiral model of req engineering process

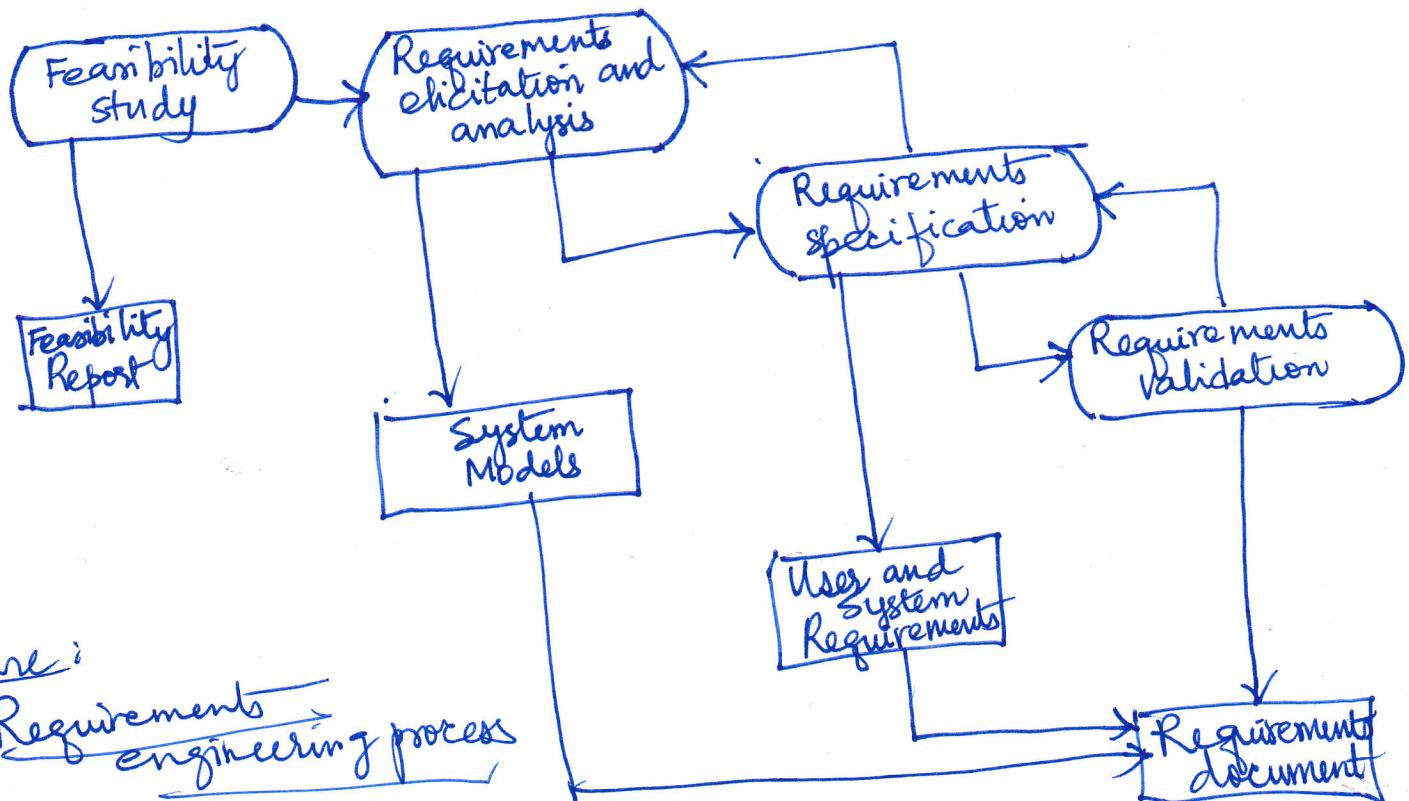
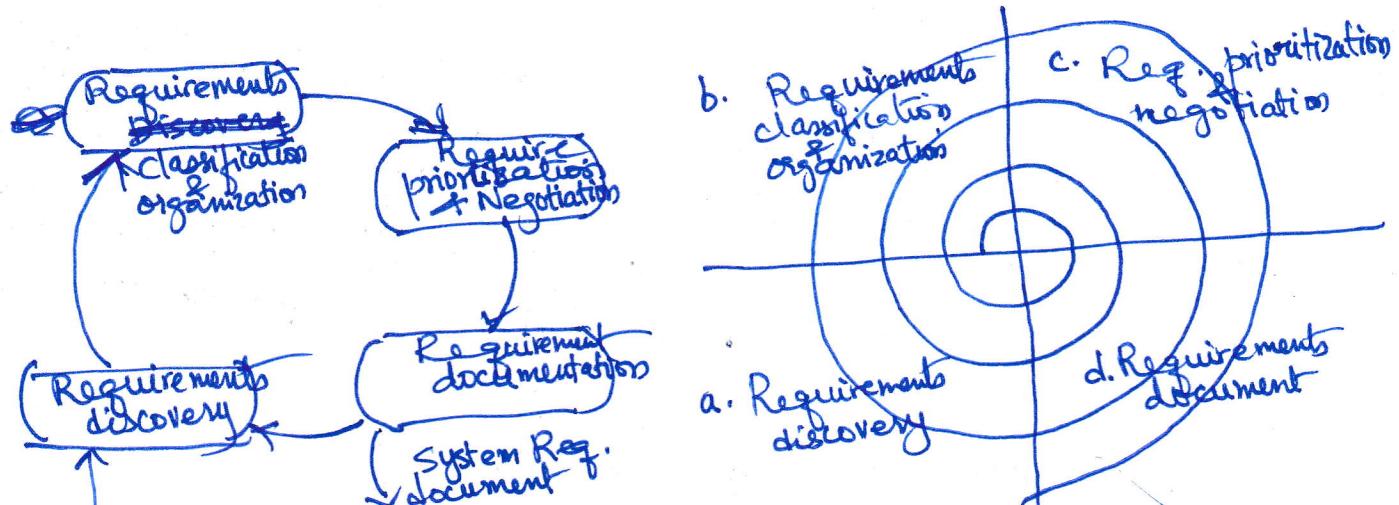


Figure : Requirements engineering process

Page 3 of —

(Reg.) Requirements Engineering



Process activities of Elicitation & Analysis

Why is Reg. elicitation & Analysis difficult?

- Stakeholders are not sure about requirements
- Stakeholders domain is different from Engineers
- Too many stakeholders and each having different requirements
- Political factors e.g. neglected stakeholders may sabotage the process
- Change in requirements due to dynamic nature of business environment
e.g. merger of two companies may change requirements drastically of the original system.

a. How (Techniques) to discover requirements from stakeholders?

- Viewpoints
- Interviewing
- Scenarios
- Use-cases

What is viewpoint approach to Reg. elicitation & Analysis?

• Viewpoints: Viewpoint approach organizes elicitation process and requirements as viewpoints. Each viewpoint is the a perspective of stakeholders.

* This approach is applicable when there are many different stakeholders (different viewpoints). Conflicts maybe discovered in perspectives as we consider multiple perspectives

4 of 8

How to classify stakeholders or sources of requirements as per this approach?

- **Interactor viewpoint**: People and systems that directly interact with system
- **Indirect viewpoints**: People or systems that do not directly interact but influence the requirements
- **Domain viewpoints**: domain characteristics and constraints that influence the requirements

Exercise on classifying Stakeholders Using viewpoints.

Eg: bank's account database

bank's customers, bank's security staff
bank managers, bank's interbank communication
Standards for

Classify the following as per viewpoints (generic) types

Indirect

Interactor

Domain

Library Manager

Finance

Article Provider

Student

External

System admin.

Staff

Cataloguer

VI
Standards

Classification
System

First identify viewpoints and then start discovering requirements based on most important view points.

Interviewing: Requirements engineers talk to system stakeholders to elicit requirements

Types:

1. Closed interviews: Ask set of predefined questions.

2. Open interviews: No specific agenda. This is more of exploration of issues and needs of stakeholders.

What cannot be elicited by interviews?
- Requirements due to application domain are hard to elicit because

- Application specialists use terminology automatically in ways the R.Engineers may easily misunderstand
- Stakeholders assume that domain knowledge is routine and understood by R.Engineers

- Interviews do not elicit organization requirements and constraints as power and politics are involved.

Scenarios: They are based on real-life interactions with the system. This is example-based rather than abstract way of describing requirements. Once requirement is identified then scenarios that requirement add a lot of detail to describe

What does a scenario include?

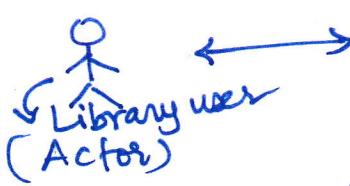
1. A description of what system and users expect when scenario starts
2. Normal flow of events
3. A description of what can go wrong
4. Other activities that are going at same time
5. System state once scenario is finished

Use-cases

A more ~~structured~~ structured approach to scenario-based elicitation is called use-case approach.

Use-cases and Scenarios are good at eliciting requirements from interactors point-of-view. They are not effective in eliciting constraints or high-level business and non-functional requirements (indirect viewpoints) or discovering domain requirements.

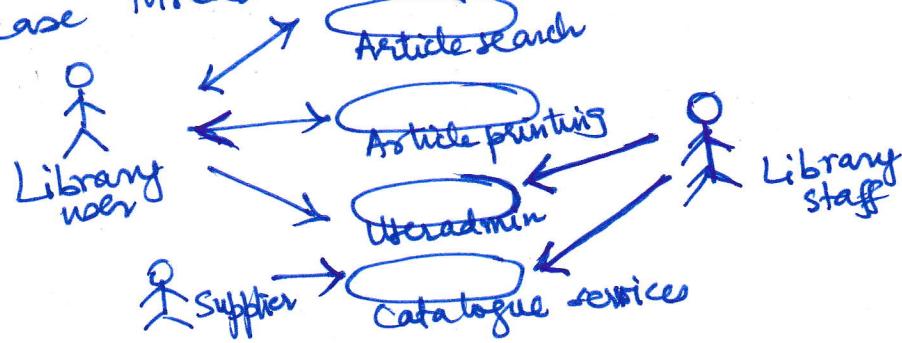
Use-cases was introduced in 1993 by Jacobson et.al. in Objectory method



use-cases identify type of interaction and actors involved. use-cases have been incorporated into UML models.

use-cases encapsulates multiple threads (scenarios).

Each use-case thread is a scenario.



Ethnography

Software systems exist in social and organizational context.

Ethnography is an observational technique that can be used to understand social and organizational requirements.

In this technique requirements due to social and organizational factors are elicited by observation by an analyst in the actual workplace.

The day-to-day tasks conducted by end-users(customers) is observed by an ethnographer(analyst). It can elicit requirements pertaining to way people actually work rather than process definitions. Requirements due to cooperation between people.

It is not good at eliciting organizational or domain requirements. It cannot elicit new features for a system. It should be used with other elicitation techniques like prototyping and use-case analysis.

Requirements Validation: what is it? elicited
Engineers should show that requirements they understood and specified are the requirements that the customer ~~wants~~ actually wants.

What checks need to be carried out?

1. Validity check: Check with customer if that requirement is what they want

2. Consistency check: Requirements cannot conflict with one another

3. Completeness check: All requirements should be included

4. Realism check: Engineers should ensure that technology exists to implement a requirements and can be completed within budget and time

5. Verifiability: This should ensure that requirement is correctly implemented. Test cases ensure that system delivers the function (requirement) correctly even before actual coding is performed.

what techniques are used for validation?

1. Requirement reviews
2. Prototyping
3. Test-case generation

Requirements management

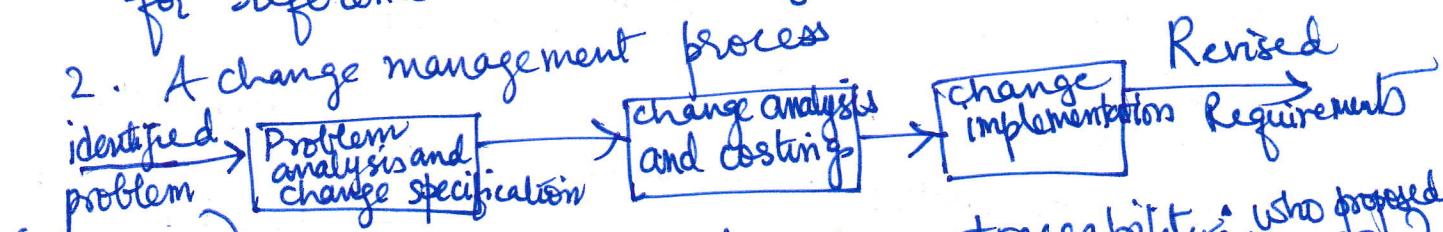
Problems: For large systems requirements are volatile. Stakeholders' understanding of problems change with time. Evolving reqs. can be classified into a enduring requirements. Solution: Req. management process is for understanding and controlling changing system requirements.

A formal process is established to track changing requirements and linking them to system requirements. The impact of requirement changes need to be assessed.

What does req. management involve?

1. Req. identification: Each req. is uniquely identified for reference & traceability

2. A change management process



Traceability involves:

- Source traceability: Who proposed this req?

- Reg. traceability: (which other reqs. are connected to this req.)

- Design traceability: Links requirements to design modules.

(what design changes occur due to req. change)

Defn: Traceability tracks links between requirements and links between requirements and system design.