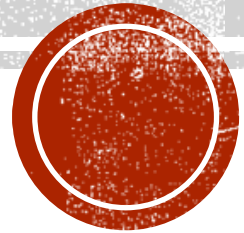
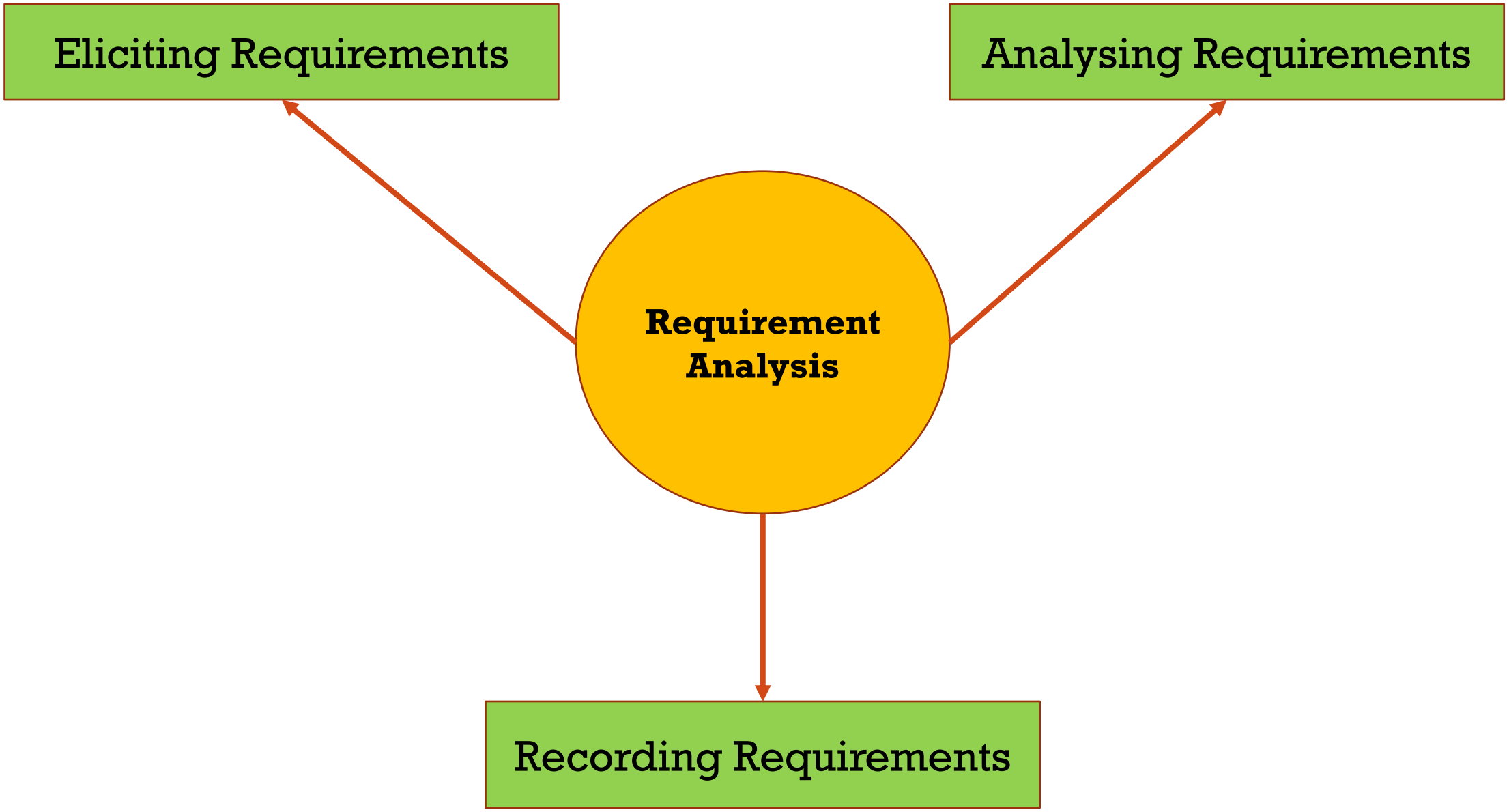
SOFTWARE REQUIREMENTS ANALYSIS AND SPECIFICATION

Yesoda Bhargava



INTRODUCTION

- One must understand the information domain before building a software product.
- Need to know the:
  - Required functions
  - Behaviour
  - Performance
  - Interfacing to be used
- The requirement for the system and software are documented and reviewed with the customer for further analysis.
- Software requirements specifications (SRS) are the foundations of the pillar of a software.
- Without requirements testers do not know what to test.


Eliciting Requirements
Analysing Requirements
Requirement Analysis
Recording Requirements

ELICITING REQUIREMENTS

- Communicating with customers and users to determine their requirements.
- Often called as the REQUIREMENTS GATHERING.

ANALYZING REQUIREMENTS

- Determine whether the stated requirements are unclear, incomplete, ambiguous, or contradictory, and then resolving the issue.

RECORDING REQUIREMENTS

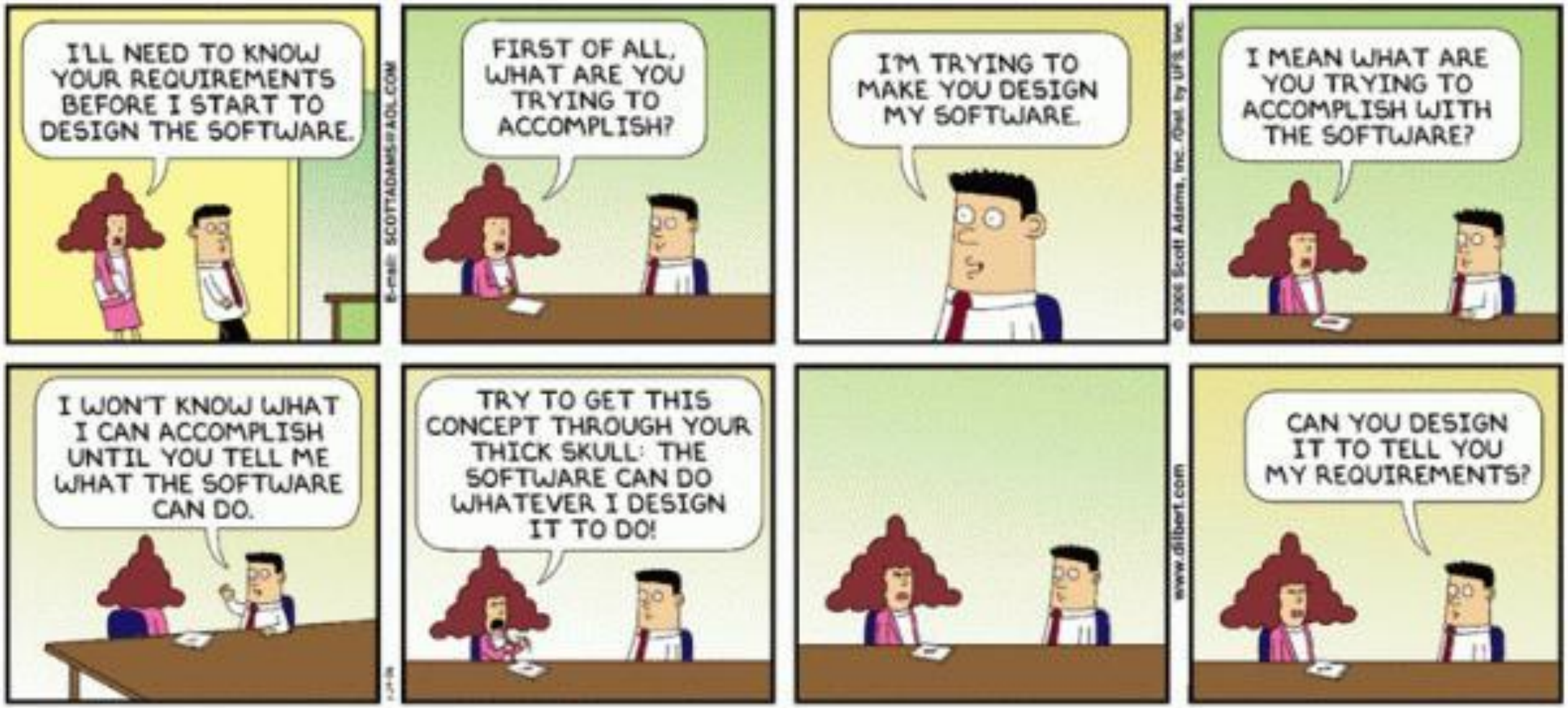
- Requirements might be documented in various forms, such as natural-language documents, use cases, suer stories, or process specifications.

TECHNIQUES TO ELICIT REQUIREMENTS

- Holding interviews (can be time-consuming and expensive)
- Holding focus group discussions
- Creating requirements lists
- Questionnaires
- Modern techniques include prototyping, and use cases.
- Combination of these methods.

FACTORS DIRECTLY RELATED TO THE SUCCESS OF A SOFTWARE DEVELOPMENT

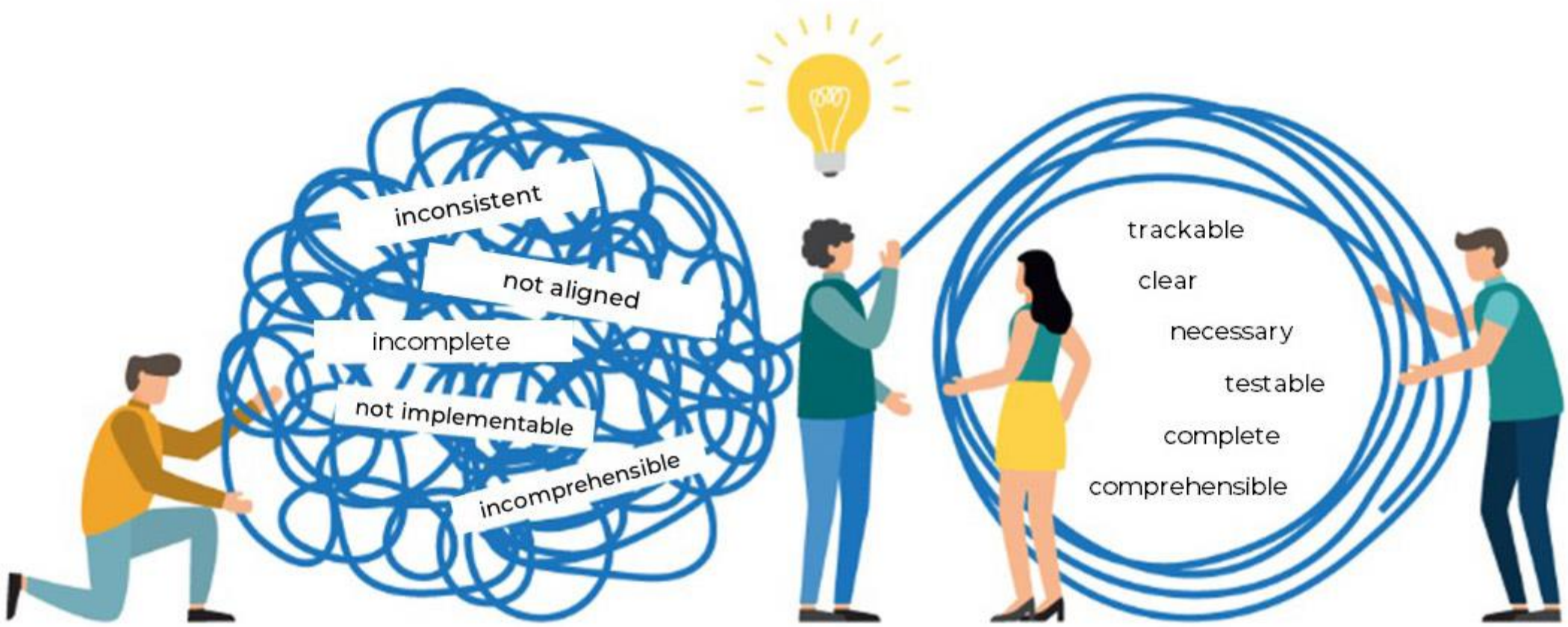
- Level of detail in planning.
- Quality of software specification
- Accuracy of software specification
- Quality and number of reviews and audits
- Effectiveness of the test and integration plan, specification, and test data.
- Level of preparation for system maintenance.



© Scott Adams, Inc./Dist. by UFS, Inc.

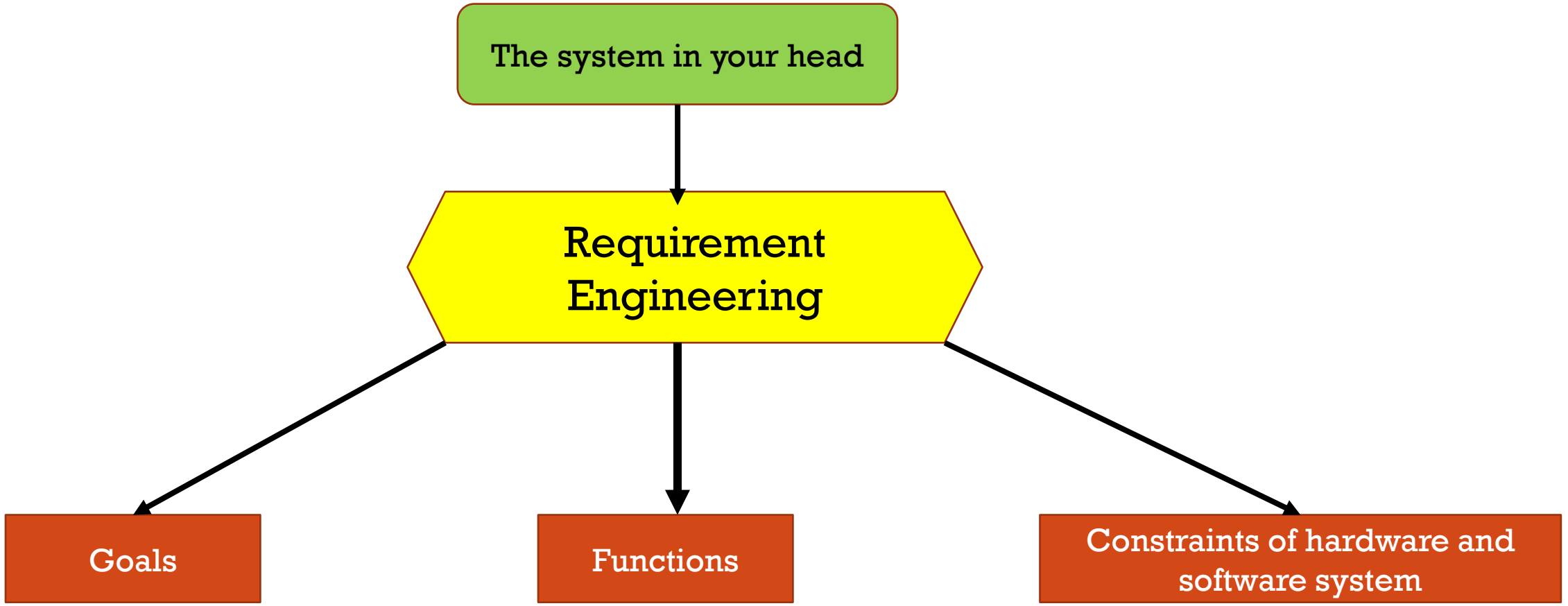
Bad Requirements

Good Requirements



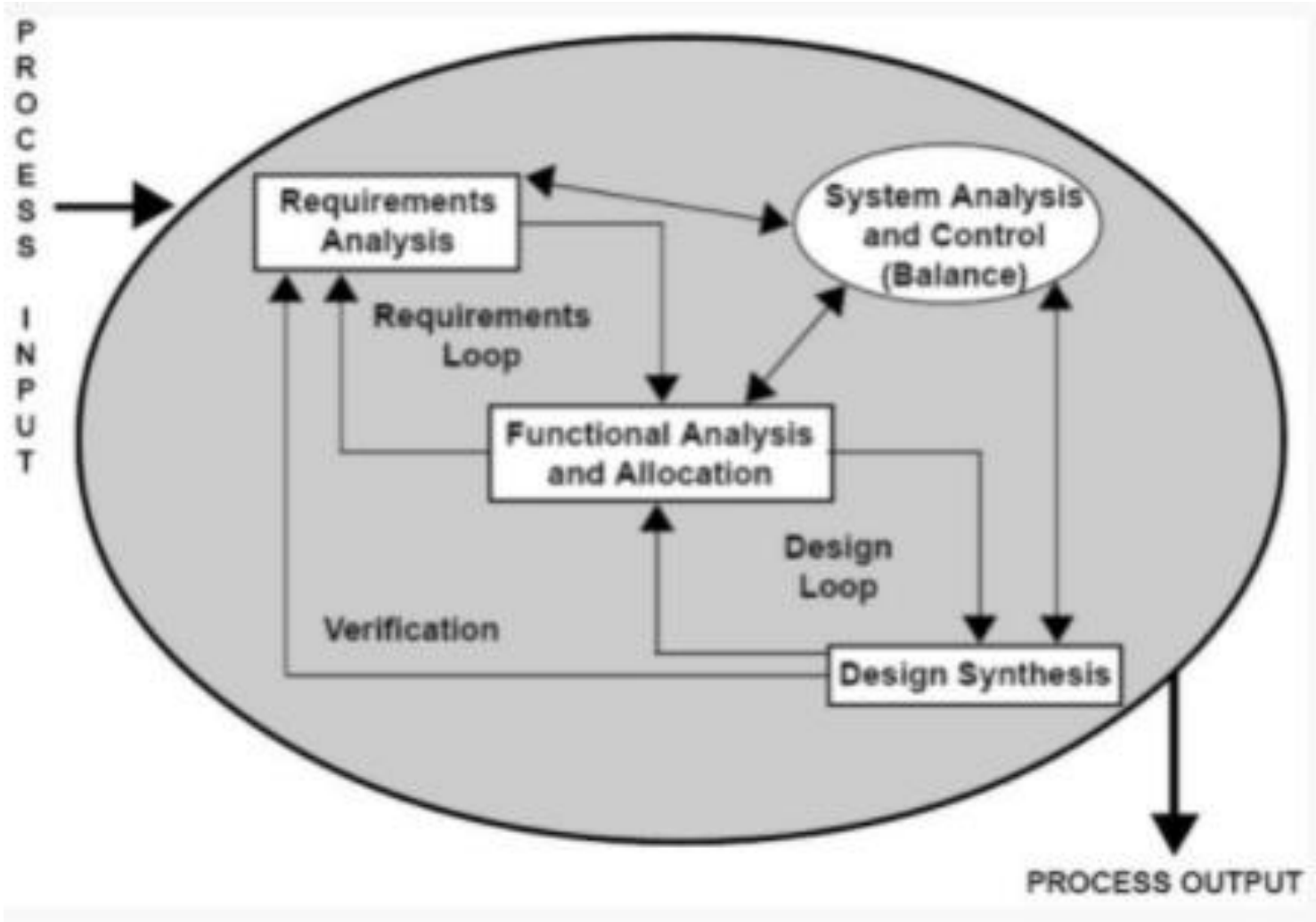
REQUIREMENT ENGINEERING

- Sub-discipline of systems engineering and software engineering.



Sometimes, requirement engineering process begins with a **feasibility study,** which leads to feasibility report. If the feasibility study suggests, the product should be developed, then the requirement analysis can begin.

SOFTWARE REQUIREMENT SPECIFICATION (SRS)



WHAT IS AN SRS?

- Complete description of the behaviour of the system to be developed.
- Includes set of use cases.
- Use cases are also known as FUNCTIONAL REQUIREMENTS (FR).
- Additionally, non-functional requirements (NFR) of supplementary requirements also form a part of SRS.
- NFR requirements impose constraints on the design or implementation such as:
  - Performance requirements
  - Quality standards
  - Design constraints

FORMAT OF A REQUIREMENT SPECIFICATION

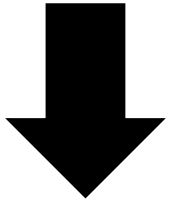
Section 1	Product overview and summary
Section 2	Development. Operating and maintenance environment
Section 3	External interfaces and data flow
Section 4	Functional requirements
Section 5	Performance requirements
Section 6	Exception handling
Section 7	Early subsets and implementation priorities
Section 8	Foreseeable modifications and enhancements
Section 9	Acceptable criteria
Section 10	Design hints and guidelines
Section 11	Cross-reference index
Section 12	Glossary of items

SOFTWARE REQUIREMENT DOCUMENTATION

- Requirement documentation is the description of what a software does or shall do.
- Used throughout development to communicate what the software does or shall do.
- Serves as a foundation for agreement.

Software Requirement Collection

Users



End users, Customers, product managers, project managers, sales, marketing, software architects, usability
Experts, interaction designers, developers, and testers.

Thus, requirements documentation has many different purposes.

- There is a relationship between the NEED for requirements documentation and the COMPLEXITY of the product, IMPACT of the product, and the LIFE EXPECTANCY of the software.
- If the software is complex or developed by many people (e.g., mobile phone software) requirements can help to better communicate what to achieve.
- **If the software is safety-critical and can harm human life (e.g., nuclear power systems, medical equipment), more formal requirements documentation is often required.**
- If the software life is relatively short, i.e., for 1-2 months, very little requirements documentation may be needed. E.g. Small mobile applications developed specifically for a certain campaign.
- If the software is a first release which is later worked upon, requirements documentation is very helpful when managing the change of software.

EXAMPLE OF SRS DOCUMENTATION

- Link 1
- Link 2
- Link 3
- Link 4
- A website link example for SRS.