AARHUS
UNIVERSITY
DEPARTMENT OF ENGINEERING

SOFTWARE ENGINEERING PRINCIPLES
REQUIREMENT ELICITATION

STEFAN HALLESTEDE
PETER GORM LARSEN
CARL SCHULTZ

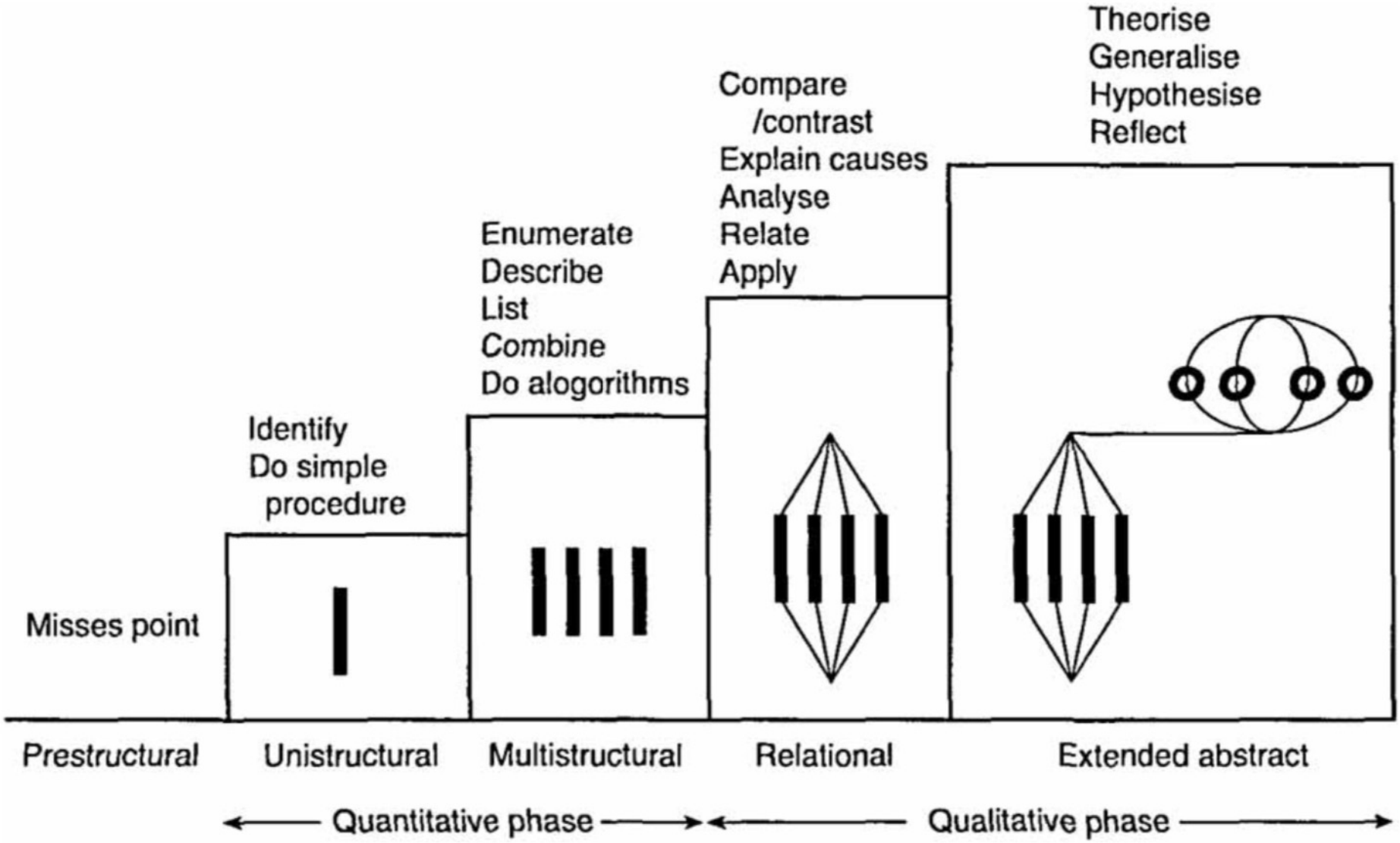
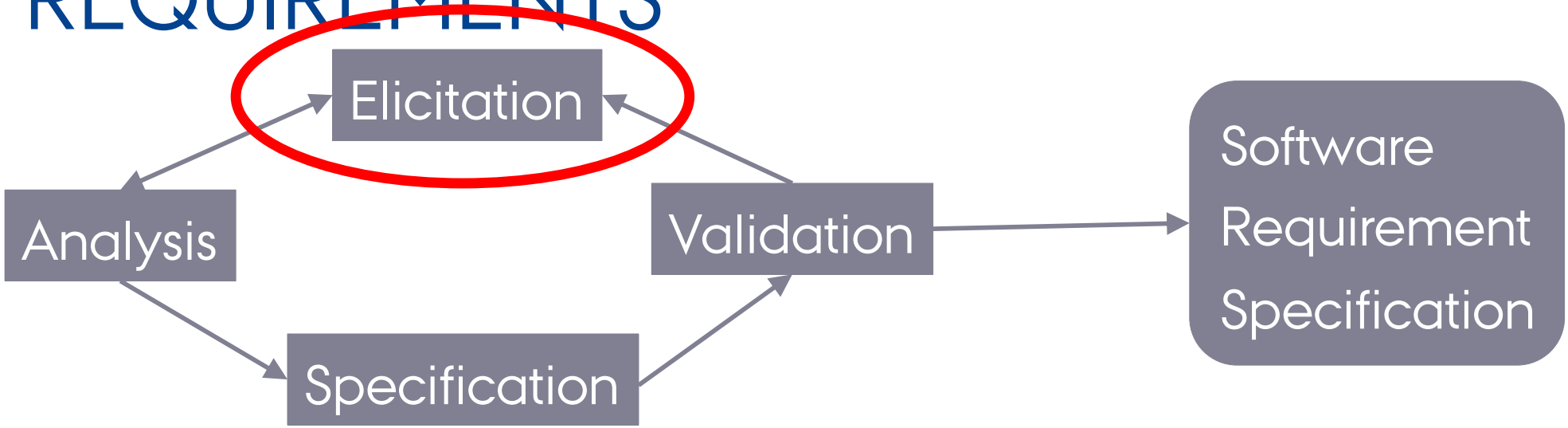


Figure 3. A hierarchy of verbs that may be used to form curriculum objectives.

SOLO Taxonomy - John Biggs and Kevin Collis

PROCESS FOR CAPTURING REQUIREMENTS



- **Elicitation**: Collecting the user's requirements
- **Analysis**: Understanding and modelling of desired behaviour
- **Specification**: Documenting the behaviour of the proposed software system
- **Validation**: Checking that the specification matches the user's requirements

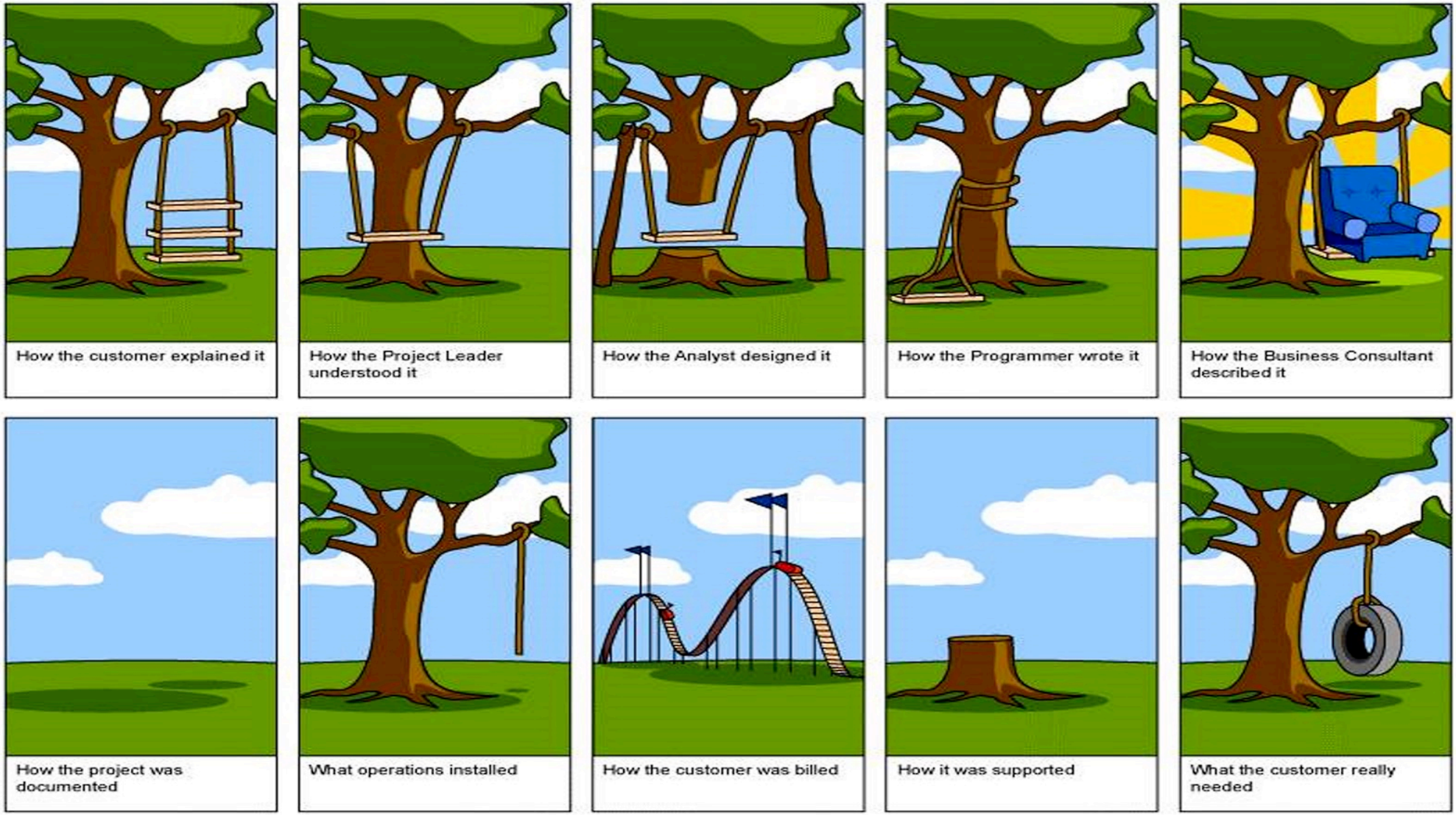
ELICITATION: FINDING AND FORMULATING REQUIREMENTS

- Includes a lot of steps:
- Formulating overall goal of the system (mission statement)
- Describing current work process and problems
- Detailed description of issues the system must solve
- Come up with possible solutions
- Turn issues and possibilities into requirements
- No sequential process – but iterative

ELICITATION – WHY IS IT HARD?

- Stakeholders cannot express what they want
- Hard for users to explain their daily tasks
  - And why they do these tasks
- Stakeholders come up with solutions – not demands
- Hard to imagine new ways of doing tasks
  - And consequences for this
- Different stakeholders have conflicting views
- General resistance to change
- Too many "nice to have" requirements are specified
- Changes spawn new requirements

ELICITATION – WHY IS IT HARD?



What are some examples of stakeholders?

REQUIREMENTS ELICITATION

- Who are the stakeholders?
  - A paying customer
  - Users of the current system
  - Domain experts
  - Market researchers
  - Lawyers or auditors, legislation
  - Software engineers
-

What activities do you do to get requirements?

REQUIREMENTS ELICITATION

- Sources for requirements:
  - Requirements gathered by stakeholders
  - Interview stakeholders
  - Agree with users on scenarios / use-cases
  - Review available documentation
  - Observe current system
  - Apprenticing with users
  - Brainstorming with current and potential users

ELICITATION – GOOD ADVICE

- Setting goals
  - Decide how to analyze data once collected
- Relationship with participants
  - Clear and professional
  - Informed consent when appropriate
- Triangulation
  - Use more than one approach
- Pilot studies
  - Small trial of main study

WORK PRODUCTS (CF. LAUESEN)

1. A description of the present work in the domain.
2. A list of the present problems in the domain.
3. A list of goals and critical issues (preliminary requirements).
4. Ideas for the large-scale structure of the future system.
5. Realistic possibilities.
6. Consequences and risks.
7. Commitment from stakeholders.
8. Conflict resolution between stakeholders.
9. Final requirements.
10. Priorities of requirements.
11. Checks to see that the requirements are complete, necessary, etc.

USER INVOLVEMENT

1. Members of design teams or workshops where the user interface is designed.
2. Knowledge sources of how tasks and business procedures are currently carried out.
3. Brainstorm participants who produce ideas and identify problems.
4. Test users who exercise the system at acceptance time to check that everything works.
5. Reviewers who assess the user interface.
6. Test users in usability tests, where they try to carry out tasks with the new user interface.
7. Members of the steering committee for the project.

ELICITATION TECHNIQUES (1)

- Stakeholder analysis
  - Who are the stakeholders?
  - What are their goals?
  - Which risks and costs do they see?
  - Made in large, small or 1-on-1 meetings

- User interviews
  - Current work process and problems are identified
  - Broad interview with many different users
  - Use open questions → more open discussion

ELICITATION TECHNIQUES (2)

- Observations
  - Hard to describe what is done – easier to observe work process

- Task demonstration
  - Task specific observation – be concrete!
  - Usability tools:
    - Think-out-loud
    - Measure time used
    - Measure errors made
    - Count number of keystrokes

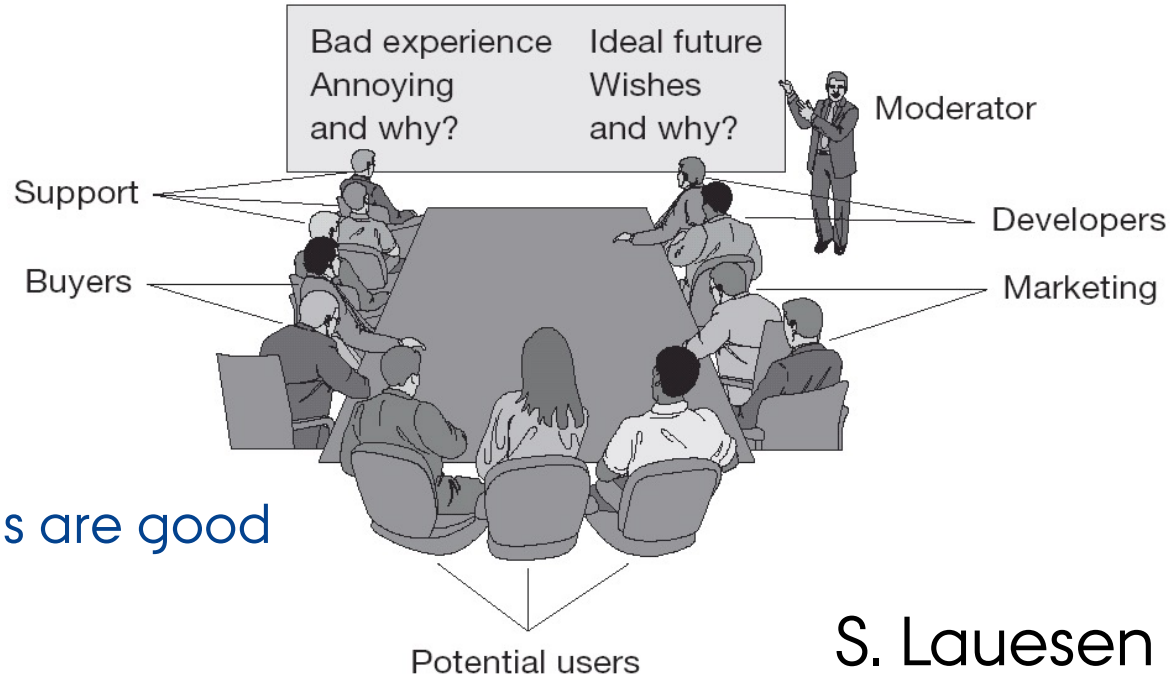
ELICITATION TECHNIQUES (3)

- Questionnaires
  - Larger group of users
  - Hard to analyse results – cannot ask users additional questions in order to understand their answers
  - Easy to misunderstand answers

- Brainstorm
  - Mixed group of stakeholders
  - Open to all suggestions
  - Suggestions often spawn new ideas

ELICITATION TECHNIQUES (4)

- Focus group
  - More structured than a brainstorm session
  - Focus on finding:
    - Problems and issues
    - Ideal solutions
    - Specify why the ideas are good



S. Lauesen

- Domain Workshop
  - Users and developers cooperate to analyse and design
  - Mixture of brainstorm and prototype sessions

WHEN TO DO WHAT?

- Does the order of tasks matter? E.g. (1) prototyping, (2) field studies

- Does the sequence that levels requirements are treated matter? E.g. (a) product- level, (b) design-level, (c) domain-level, (d) goal-level?

WHEN TO DO WHAT?

› Read article by Mjølner

MORE ELICITATION TECHNIQUES

See Elicitation Techniques - BA Framework - Dashboard.pdf

OVERVIEW OF TECHNIQUES

Elicitation Techniques - BA Framework - Dashboard.pdf

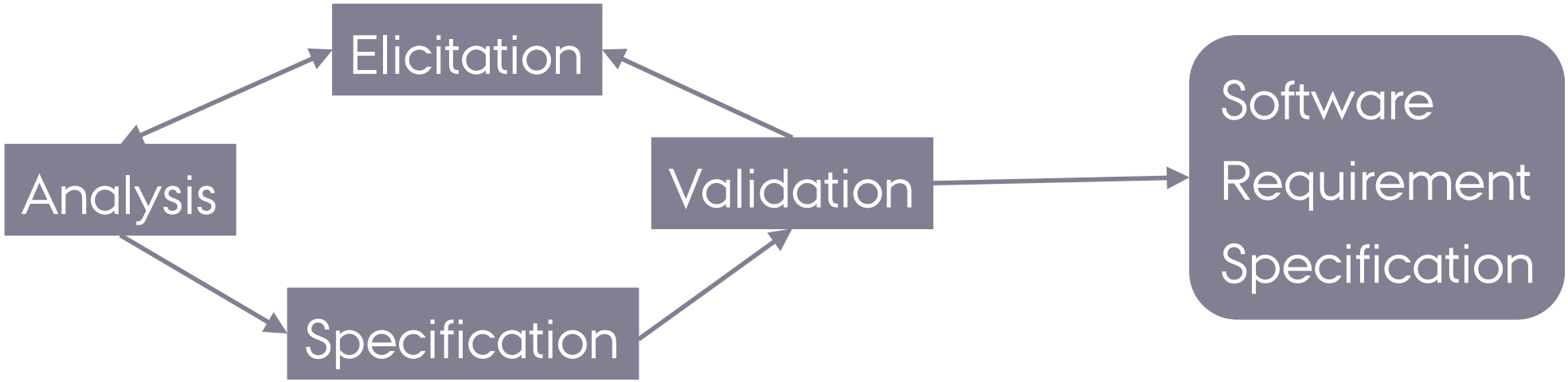
- Brainstorming
- Document Analysis
- Focus Group
- Interface Analysis
- Interview
- Observation
- Prototyping
- Requirements Workshop
- Reverse Engineering
- Survey/Questionnaire

- Purpose
- Description
- Process
- Intended
- Audience
- Strengths
- Weaknesses

ELICITATION IS EXPLORATORY

Elicitation involves the actions that are taken to **understand** the users and **discover** their needs. Elicitation includes the discovery and some invention, as well as recording those bits of requirements information that customer representatives and subject matter experts (users) offer to the analyst. Elicitation demands **iteration**. The participants in an elicitation discussion won't think of everything up front, and their thinking will **change** as the project continues. Requirements development is an **exploratory activity**.
Karl E. Wiegers (*More About Software Requirements*)

PROCESS FOR CAPTURING REQUIREMENTS



- **Elicitation**: Collecting the user's requirements
- **Analysis**: Understanding and modelling of desired behaviour
- **Specification**: Documenting the behaviour of the proposed software system
- **Validation**: Checking that the specification matches the user's requirements