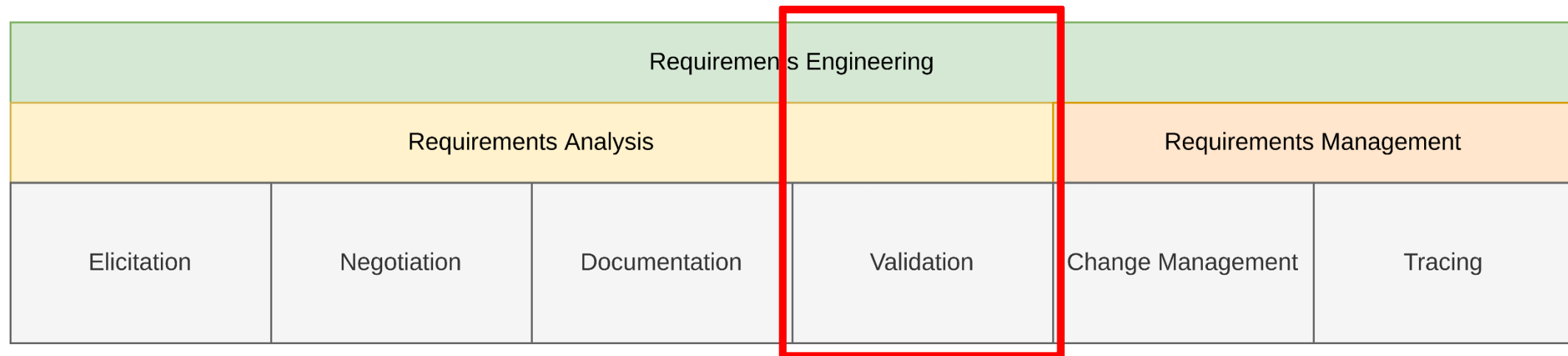


Requirement Engineering

Lecture 10: Requirements Validation

Prof. Dr. Benjamin Leiding
M.Sc. Anant Sujatanagarjuna

General Requirements Engineering Process Overview



Lecture 10: Requirements Validation

Content

1. Requirements Validation in General
2. Principles of Requirements Validation
3. Requirements Validation Techniques



REQUIREMENTS VALIDATION IN GENERAL

Requirements Validation in General

Why?

- Review of the requirements
 - Should uncover errors in requirements
- Present requirements to stakeholders
 - Identify deviations between requirements and actual wishes
- Quality of requirements is evaluated
 - Decision is made if quality is sufficient
 - Leads to approval of requirements
 - Should use predefined criteria

Requirements Validation in General

Quality Aspects of Requirements

Three major goals of validation:

- Content
 - Are all relevant requirements elicited and documented?
 - Is the level of detail appropriate?
- Documentation
 - Is the documentation according to predefined guidelines and specifications?
- Agreement
 - Do all stakeholders concur with the documented requirements

Requirements Validation in General

Quality Aspects - Content

- Refers to the content of the document
 - The requirements themselves
 - The completeness of the requirements

Requirements Validation in General

Quality Aspects - Documentation

- Validates the documentation of the requirements
 - Flaws in the documentation
 - Violation of documentation guidelines
 - Project specific
 - Organization specific
 - Understandability of document formats
 - Structure of the documents

Requirements Validation in General

Quality Aspects - Documentation (Guideline Violation Risks)

- Impairment of development activities
 - Development activities may require certain document formats
 - For example, consider the Rational Unified Process (RUP)
 - Later stages refine models created during the requirements engineering
 - If models have not been created, refinement is not possible
- Misunderstandings
 - The target audience may not understand or misinterpret requirements in certain document formats
 - Requirements may become unusable

Requirements Validation in General

Quality Aspects - Documentation (Guideline Violation Risks)

- Incompleteness
 - Relevant information may be missing from the document
 - Must not necessarily be the requirements themselves
 - Stakeholders involved
 - Persons responsible
- Overlooking requirements
 - Requirements documented at the wrong place → Not missing!
 - May be overlooked in later activities → Example: Developer wants to implement the persistence features of a system. Checks section “4.3 Persistence requirements” and implements all documented features. Additional persistence requirements hiding in other sections

Requirements Validation in General

Quality Aspects - Documentation (Test Criteria)

- Conformation to documentation format and rules
 - Are all documents in the predetermined documentation format?
 - Have modeling languages been used properly?
- Conformity to documentation structures
 - Has the structure of the document been maintained?
 - Is everything where it belongs?
- Understandability
 - Is the documentation understandable in the given context?
 - Is the glossary sufficient?

Requirements Validation in General

Quality Aspects - Agreement

- Stakeholders gain additional knowledge during the course of requirements engineering
 - May lead to changes between the begin of the elicitation and the end of the documentation
 - Validation offers a last opportunity for changes without impacting later phases
- Test criteria
 - Agreed → Is every requirement agreed upon with all relevant stakeholders?
 - Agreed after changes → Is every requirement agreed upon with all relevant stakeholders after it has been changed?
 - Conflicts resolved → Have all known conflicts with regard to the requirements been resolved?



PRINCIPLES OF REQUIREMENTS VALIDATION

Principles of Requirements Validation

Overview

- 6 principles of requirements validation

Principles of Requirements Validation

Overview

1. Involvement of the correct stakeholders

- Independence of the author
 - Author should not be the one validate a requirement
 - Has prior knowledge which influences the validation
- Internal auditors
 - Stakeholders that are members of the developing organization
 - Easy to organize
- External auditors
 - Higher effort
 - Have to become familiar with the document
 - Independent perspective

Principles of Requirements Validation

Overview

2. Separating the identification and the correction of errors

- Identify flaw
- Document flaw
- Fix flaw
- Separation allows concentration on identification → Better separation of resources

3. Validation from different views

- Different people → Different perspectives

Principles of Requirements Validation

Overview

4. Adequate change of documentation type

- Check if usage of different documentation types balances out weaknesses
- Natural language → Very expressive
- Conceptual models → Abstraction and complex behavior

5. Construction of development artifacts

- Creation of artifacts reveals ambiguities
- Example:
 - Creation of a test case to validate the requirement
 - Problems during the test creation

Principles of Requirements Validation

Overview

6. Repeated validation

- Positive validation at one point does not guarantee that a requirement is still valid at a later point in time
- Should be repeated in the following cases
 - Lots of innovative ideas and technologies
 - Significant gain of knowledge
 - Long-lasting projects
 - Very early previous validation
 - Unknown domain
 - Reuse of requirements

VALIDATION TECHNIQUES

Validation Techniques

Overview

- 6 validation techniques in this lecture

- Three types of reviews
 - Commenting
 - Inspections
 - Walk-throughs

- Three supporting techniques
 - Perspective-based reading
 - Prototypes
 - Checklists

Validation Techniques

Commenting

- Author gives requirements to another person
 - Usually a co-worker
- Receive expert opinion of the other person
 - Marks flaws in the document
 - Briefly discusses the flaws with the author
- Informal and cheap

Validation Techniques

Inspection

Systematic check development artifacts for errors

- Consists of 4 phases
 - Planning
 - Object and goals of the inspection are determined
 - Roles and participants are selected
 - Overview
 - Author explains the requirements to the inspection team → Common understanding
 - Error detection
 - Inspectors analyze the requirements for flaws
→ Individually or in teams
 - Error collection and consolidation
 - Results are collected and consolidated
 - Duplicate errors removed
 - Check if each finding is indeed an error

Validation Techniques

Inspection - Roles

- Organizer
 - Plans and supervises the inspection process
- Moderator
 - Leads collection and consolidation session
 - Neutral
- Author
 - Explains the requirements
 - Responsible for error correction
- Reader
 - Introduces the requirements to be inspected
 - Neutral (usually the moderator)

Validation Techniques

Inspection - Roles

- Inspectors / Reviewers
 - Responsible for the flaw detection
 - Communicate findings to members of the project team
- Minute-taker / Recorder
 - Takes minutes of the results during the collection and consolidation session
 - Writes protocol of the inspection results

Validation Techniques

Walk-Through

- “Lightweight” inspection
 - Less strict
 - Roles not strongly differentiated
- Step-by-step “walk-through” of the requirements during the review session
 - Author introduces the requirements
 - Can use the opportunity to give additional details about the requirements
 - For example, rationale for a decision

Validation Techniques

Perspective-based Reading

- Validate the requirements from a certain perspective
- Examples:
 - User/customer perspective → Do the requirements meet the expectations and desired functions of a user?
 - Software architect perspective → Do the requirements contain all information required for the architectural design?
 - Tester perspective → Do the requirements contain all information necessary to derive test cases?
 - Content / documentation / agreement perspective

Validation Techniques

Prototypes

- Implement something to try out
 - Not everything, but a subset of the requirements
- Throw-away prototypes
 - Use once, throw away
- Evolutionary prototypes
 - Develop further after prototypical use
 - Additional prototypes
 - Development artifact

Validation Techniques

Prototypes

- Validation with prototypes requires
 - Manuals / Instructions → Information necessary to apply the prototype
 - Validation scenarios
 - Scenarios to perform with the prototype
 - Encompasses data and user interactions
 - Checklist with validation criteria → Define how users of the prototype validate the requirements
- Results:
 - Protocol of the auditor → Experiences with the prototype of the auditor
 - Observation protocol (optional) → “Second person” writes protocol of how the auditor interacted with the prototype

Validation Techniques

Checklists

- Set of questions and/or statements about a certain circumstance
 - Make sure no aspect is omitted
 - Should not be longer than a single page
- Can be used in combination with all previously introduced techniques
 - Provide a guideline
 - Automatically structure review results

Validation Techniques

Checklists

- Sources for checklist entries
 - The three quality aspects of requirements
 - Principles of requirements validation
 - Criteria for good requirements (Lecture 04, Part 1, Section 2)
 - Characteristics for good requirements documents (Lecture 04, Part 1, Section 3)
 - Experience from prior projects
 - Error statistics

Validation Techniques

(Software-) Reviews - Its Complicated

- 90% of all companies choose testing as their only quality assurance measure
- Test requires executable code → Bugs are discovered late
- Failures are discovered not the bugs themselves
 - Isolation of the error requires additional effort
- Not all errors are found with testing
 - No proof of correctness
- Test effort ("debugging") often more than 50% of the development effort

Validation Techniques

(Software-) Reviews - Its Complicated

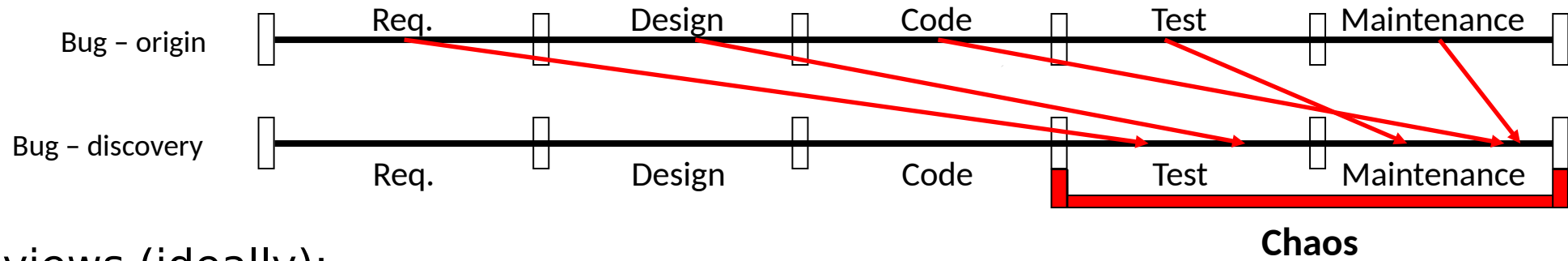
- 90% of all companies choose testing as their only quality assurance measure
- Test requires executable code → Bugs are discovered late
- Failures are discovered not the bugs themselves
 - Isolation of the error requires additional effort
- Not all errors are found with testing
 - No proof of correctness
- Test effort ("debugging") often more than 50% of the development effort

→ Do not use “reviews” as your only validation technique

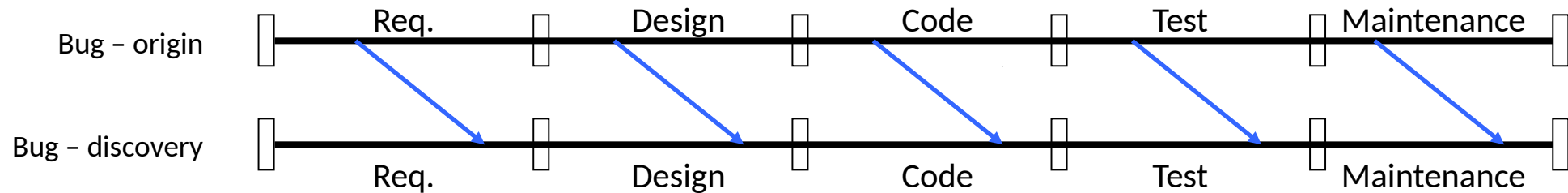
Validation Techniques

(Software-) Reviews - Overview

- Without reviews

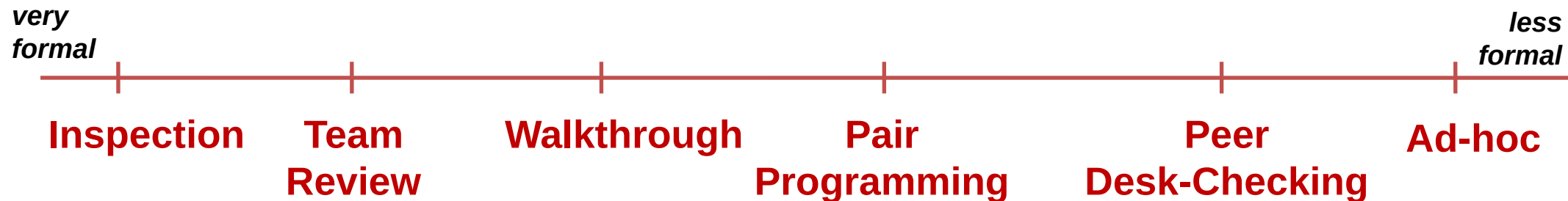


- With reviews (ideally):



Validation Techniques

Review Techniques - Overview



- Inspections as the most reliable review technique
 - Empirical studies
 - Inspections find 50% more defects than walkthroughs
 - Inspections find up to 6x more defects than ad-hoc reviews

Validation Techniques

Review Techniques

- Ad-hoc Review
 - If you cannot solve a problem, you spontaneously ask an employee for help
 - Result depends entirely on the experience of one employee

- Peer Desk-Checking
 - Similar to ad-hoc review
 - The employee "executes the product to be checked on paper" (mostly code)

Validation Techniques

Review Techniques

- Pair Programming
 - Two developers share a PC workstation. While one operates the keyboard, the other checks the input.
 - One of the practices of eXtreme Programming.

- Walkthrough
 - The author of a document presents it to collaborators to gain a general understanding and improve the quality of the document.
 - No predefined process and no guidance on how to find errors.
 - Risk: The author easily forgets to focus on the essential parts of the document during the presentation.

Validation Techniques

Review Techniques

- Team-Review
 - Similar to the inspection technique but less formal
 - Several employees inspect a product individually
 - The results are discussed in a meeting with the author

SUMMARY

Summary

- Validation ensures quality of the requirements
- Makes sure all stakeholders are on the same page
 - Also offers last chance for changes without negative impact on later phases
- Different aspects of quality
 - Content, documentation, agreement
- Should abide by the six principles of validation
 - Increases the quality of validation results
- Different techniques available for validation
 - Usually some flavor of a review
 - Checklists are helpful



Questions?

References

- Additional literature on reviews and software inspection
 - Karl E. Wiegiers (2002) – Peer Reviews in Software – A practical Guide.
 - T. Gilb, D. Graham (1993) – Software Inspection.