Faculty of ECMS / School of Computer Science

Software Engineering & Project Quality Management

Quality Management

Lecture 8

Chapter 27 (24 in Edition 9)
in the course text book

Outline

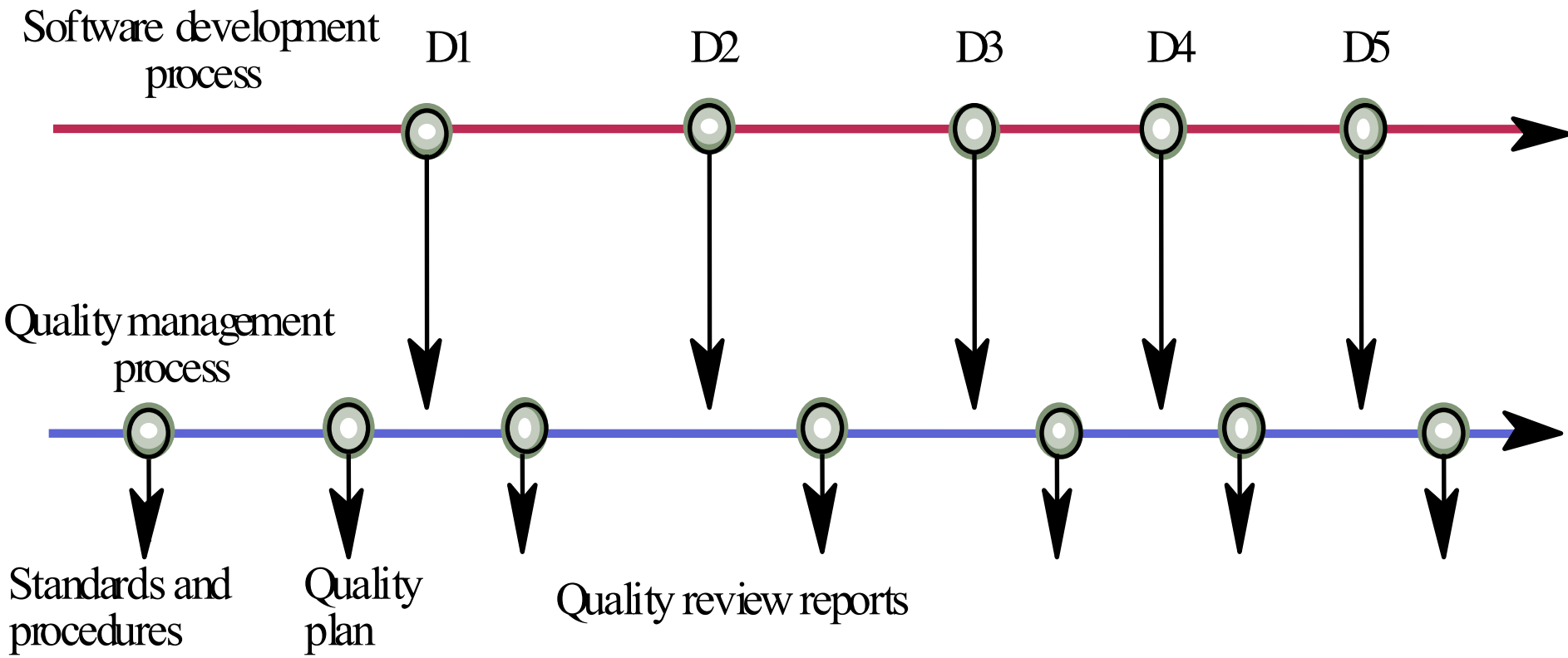
- Software quality: an overview
- Software quality management
- Quality assurance and standards
- Quality control techniques

Quality Management Activities

- Quality assurance
  - Establish organisational procedures and standards for quality
- Quality planning
  - Select applicable procedures and standards for a particular project and modify these as required
- Quality control
  - Ensure that procedures and standards are followed by the software development team

- Quality management should be adequately separated from project management to ensure independence

Quality Management and Software Development



Quality Planning

- A quality plan sets out the desired product qualities and how these are assessed and defines the most significant quality attributes
- The plan should define the quality assessment process
- The plan should set out which organisational standards should be applied and, where necessary, define new standards to be used

Quality Planning (Cont.)

- Quality plan structure
  - Product introduction
  - Product plans
  - Process descriptions
  - Quality goals
  - Risks and risk management

- Quality plans should be short, succinct documents.
  - If they are too long, no-one will read them

Scope of Quality Management

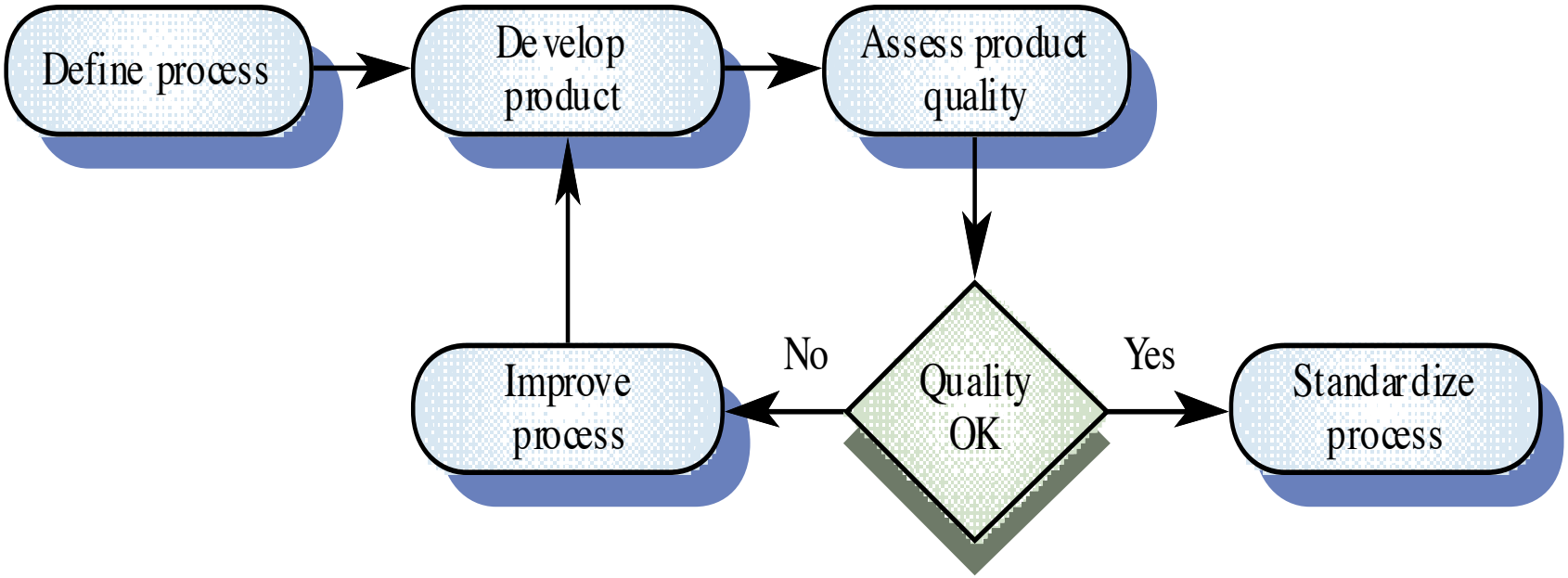
- For large, complex systems, particularly for safety-critical system
  - Quality management is particularly important
  - Rigorous approach needed to be taken
  - The **quality documentation** is required
    - quality documentation is a record of progress and supports continuity of development as the development team changes
- For smaller systems
  - Quality management needs less documentation
  - Focus on establishing a **quality culture**

Software Quality

- Quality, simplistically, means that a product should meet its specification
- This definition is problematical for software systems because
  - There is a tension between customer quality requirements (efficiency, reliability, etc.) and developer quality requirements (maintainability, reusability, etc.)
  - Some quality requirements (e.g., maintainability) are difficult to specify in an unambiguous way
  - Software specifications are usually incomplete and often evolving
- The focus may be "fitness for purpose" rather than specification conformance

Process and Product Quality

- The quality of a developed product is influenced by the quality of the production process



Software Fitness for Purpose

- Have programming and documentation standards been followed in the development process?
- Has the software properly tested?
- Is the software sufficiently dependable to be put into use?
- Is the performance of the software acceptable for normal use?
- Is the software usable?
- Is the software well-structured and understandable?

Software Quality Attributes

Safety	Understandability	Portability
Security	Testability	Usability
Reliability	Adaptability	Reusability
Resilience	Modularity	Efficiency
Robustness	Complexity	Learnability

- For each software project, the requirements of these attributes might be different!
  - There will always will be conflicts between attributes

Quality Assurance and Standards

- Standards are the key to effective quality management
  - **Classification 1:**
    - International
      - ISO9000, ISO 9001, IEEE, DoD….
    - National
    - Organizational
  - **Classification 2**
    - Product standards
      - Apply to the software produce being developed, e.g. coding style, documentation standard etc.
    - Process standards
      - define how the software process should be enacted

Product and Process Standards

Product standards	Process standards
Design review form	Design review conduct
Requirements document structure	Submission of documents to CM
Method header format	Version release process
Java programming style	Project plan approval process
Project plan format	Change control process
Change request form	Test recording process

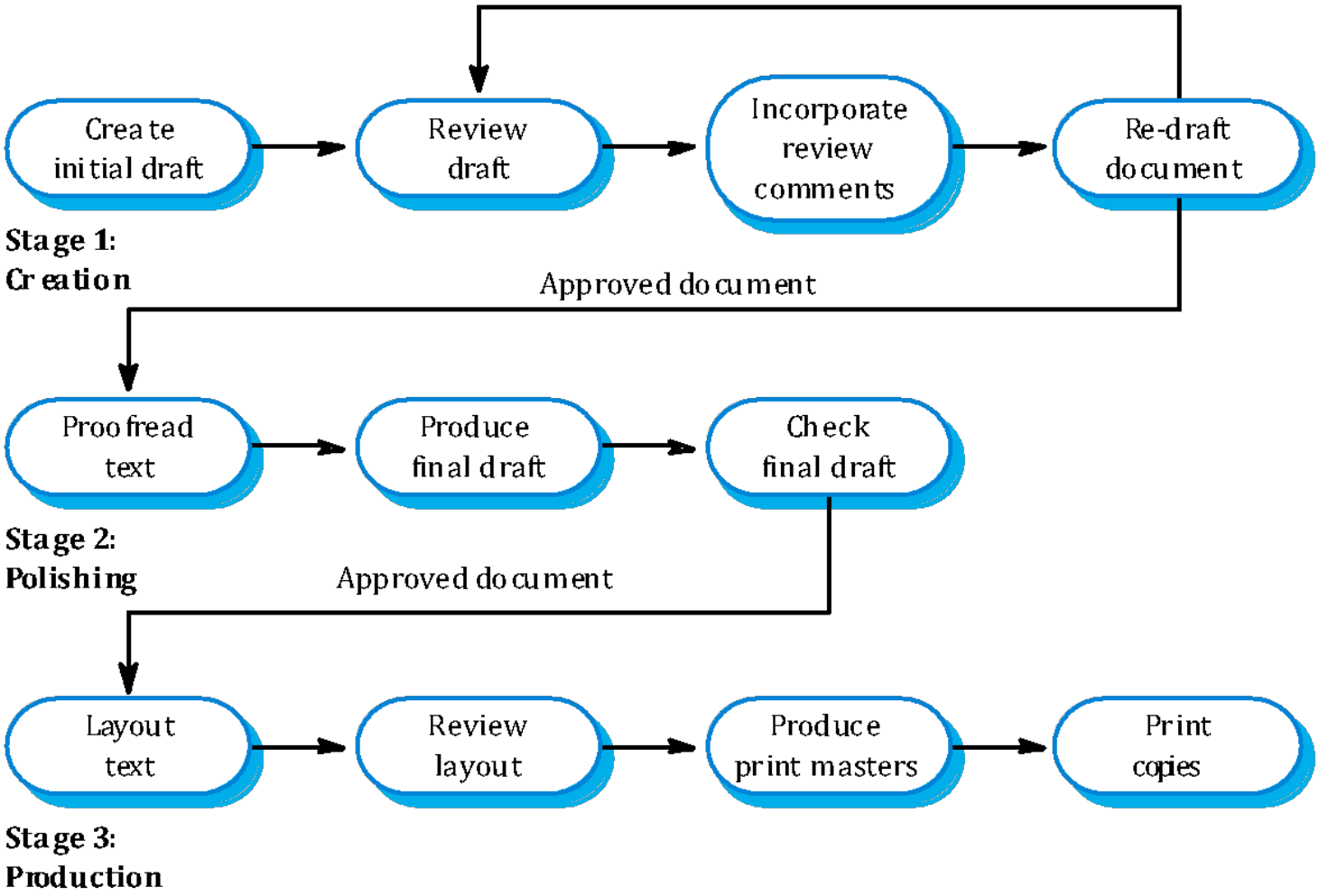
Importance of Standards

- Encapsulation of best practice
  - avoids repetition of past mistakes
- They are a framework for defining what quality means in a particular setting
- Standards provide continuity
  - new staff can understand the organisation by understanding the standards that are used

Documentation Standards

- Documents are the only tangible manifestation of the software
- Standards related to documentation includes:
  - Documentation process standards
    - Concerned with how documents should be developed, validated and maintained
  - Document standards
    - Document identification standards
    - Document structure standards
    - Document presentation standards
      - Define fonts and styles, use of logos, etc.
    - Document update standards
  - Document interchange standards
    - Concerned with the compatibility of electronic documents

Documentation Process



A More Typical "Process"
What not to do

- Writing tasks are assigned to individuals
- Individuals email their contributions to documentation manager
  - Often a short time before deliverable is due
  - Often as a Word document
- Documentation manager has a mad scramble to compile document
  - Will often have to write bits not completed by team mates
- Minimal reviewing of the document is done

Coding Conventions

- Coding conventions are important for a number of reasons
  - 80% of the lifetime cost of software goes to maintenance
  - Very few software is maintained by the original author
  - Code conventions improve readability
    - New code can be understood more quickly
  - Source code should be well packaged and clean when shipped.

  [Source: paraphrased from Sun coding convention]

Standards in Your Project

- We will not be expecting you to comply with any particular standard
  - We don't have a standards library
- However, we do have a number of document templates and guidelines (e.g. for SRS, SPMP, SDD)
  - These are informed by international (product) standards (IEEE)
  - You should use these as a guide
  - You may want to adapt them to your own needs

Quality Control

- This involves checking the software development process to ensure that procedures and standards are being followed

- A widely used approach to quality control is quality reviews

Quality Reviews

- A group of people carefully examine part or all of a software system and its associated documentation, to find potential problems
  - Code, designs, specifications, test plans, standards, etc. can all be reviewed
  - Software or documents may be 'signed off' at a review which signifies that progress to the next development stage has been approved by management

- It is the principal method of validating the quality of a process or of a product

Review Results

- Comments made during the review should be classified
  - **No action**. No change to the software or documentation is required
  - **Refer for repair**. Designer or programmer should correct an identified fault
  - **Reconsider overall design**. The problem identified in the review impacts other parts of the design. Some overall judgement must be made about the most cost-effective way of solving the problem
- Requirements and specification errors may have to be escalated to the client

Types of Reviews

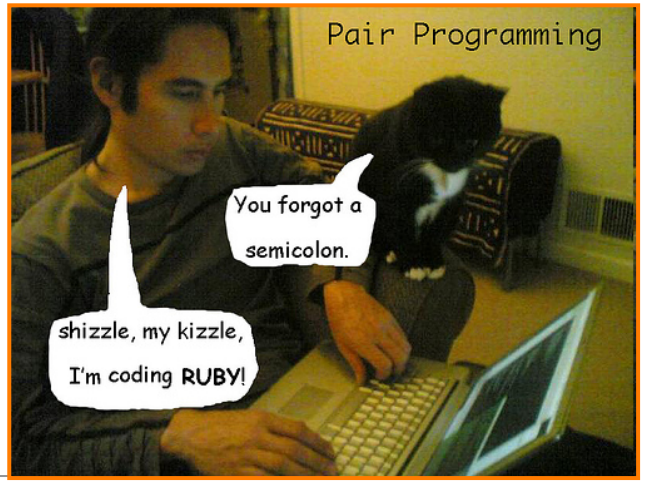
- Technical review
  - Review for conformance to standards or achievement of project milestones
  - Led by team leader with management participation
- Software (Fagan) inspection
  - Peer review with formal process
  - Led by independent moderator
  - Systematic data collection
  - Focus on fault detection and description
  - Process improvement goal

Types of Reviews (Cont.)

- Structured walkthrough
  - Less formal than inspection
  - Usually led by producer
  - No formal data collection
  - No participant preparation
- Audit
  - External review of work product
  - Independently managed
  - Usually late in the process
- Review team with size of around 4 people

Agile Approaches to Quality

- Pair programming (Develop software artefacts as a team)
  - Driver writes the software
  - Other person eyeballs the software looking for defects
  - Both contribute to the thought processes and problem solving
- Studies show a marked increase in quality
  - Especially in terms of defect rate
  - Strengthening the case for pair-programming (IEEE Software July/August 2000)
- Cost is no where near double that of single programmer



Agile Approaches to Quality

- Test-driven development
  - Test-first development
  - Incremental test development from scenarios (or use cases)
  - User involvement in the test development and validation
  - The use of automated test harnesses
  - More on this in testing lecture (Week 6)

Relevance to Your Project

- In your project:
  - What software quality activities you have planned and/or taken
  - What quality control techniques you have used in your project
    - Assign certain person as quality control manager
  - How do you use relevant techniques in your quality management process
  - International standards that you have followed, and/or the principles that you followed
- In report:
  - Tell the truth
  - Justification your claim

Key points revisited

- Software quality management is concerned with ensuring that software meets its required standards
- Quality assurance procedures should be documented in an organisational quality manual
- Software standards are an encapsulation of best practice
- Reviews are the most widely used approach for assessing software quality