Software Quality Management

Eduardo Figueiredo

http://www.dcc.ufmg.br/~figueiredo

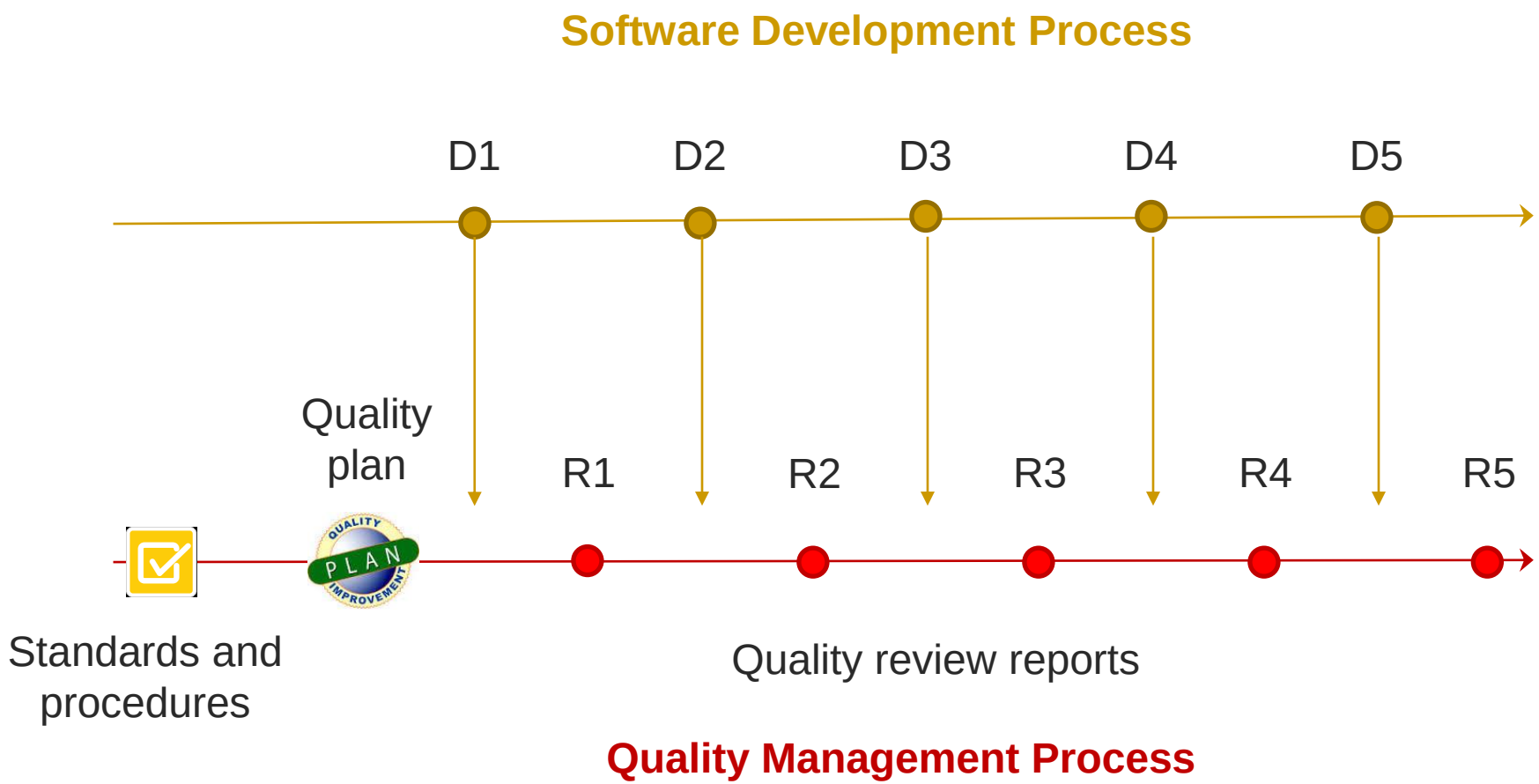
Software Quality

- Systems should meet the user needs, perform efficiently and reliably, and be delivered on time and within budget
- Quality management techniques have led to improvements in the past 20 years
  - Very important in large, long lasting systems

Organization vs. Project

- Quality management is an organization and project issue
  - **Organization**: establishes a framework of the organization processes and standards
  - **Project**: involves the application of specific quality processes to a project
- Project quality management includes defining a quality plan for a project

Development and Quality Processes



QA Team

- The QA team should work independently of the development team
  - They report quality issues to organization managers (above project managers)
  - They are not restricted by project budged and schedule
- In smaller companies, QA and development teams are closely related

Quality Planning

- A quality plan defines
  - The most significant quality attributes
  - How they are assessed
- Quality plan structure
  1. Product introduction
  2. Product plans
  3. Process descriptions
  4. Quality goals
  5. Risks and risk management

Outline for a Quality Plan

1. Product Introduction
   - Description of market, quality expectations, etc.
2. Product Plans
   - Release dates, distribution plan, responsibilities, etc.
3. Process Descriptions
   - Processes and standards used in the product
4. Quality Goals
   - Goals and plans for the product and its attributes
5. Risks and Risk Management
   - Description of key risks and actions to avoid them

Software Quality

- In general, quality means that a product should meet its specification
- This definition is not easy for software
  - Some quality requirements are difficult to specify in an unambiguous way
  - Software specifications are usually incomplete and often inconsistent
- The focus is on ‘fitness for purpose’ rather than specification conformance

Fitness for Purpose

- Has the software been properly tested?
- Is the performance of the software acceptable for normal use?
- Is the software usable?
- Is the software well-structured and understandable?
- Have programming and documentation standards been followed?

Software Quality Attributes

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

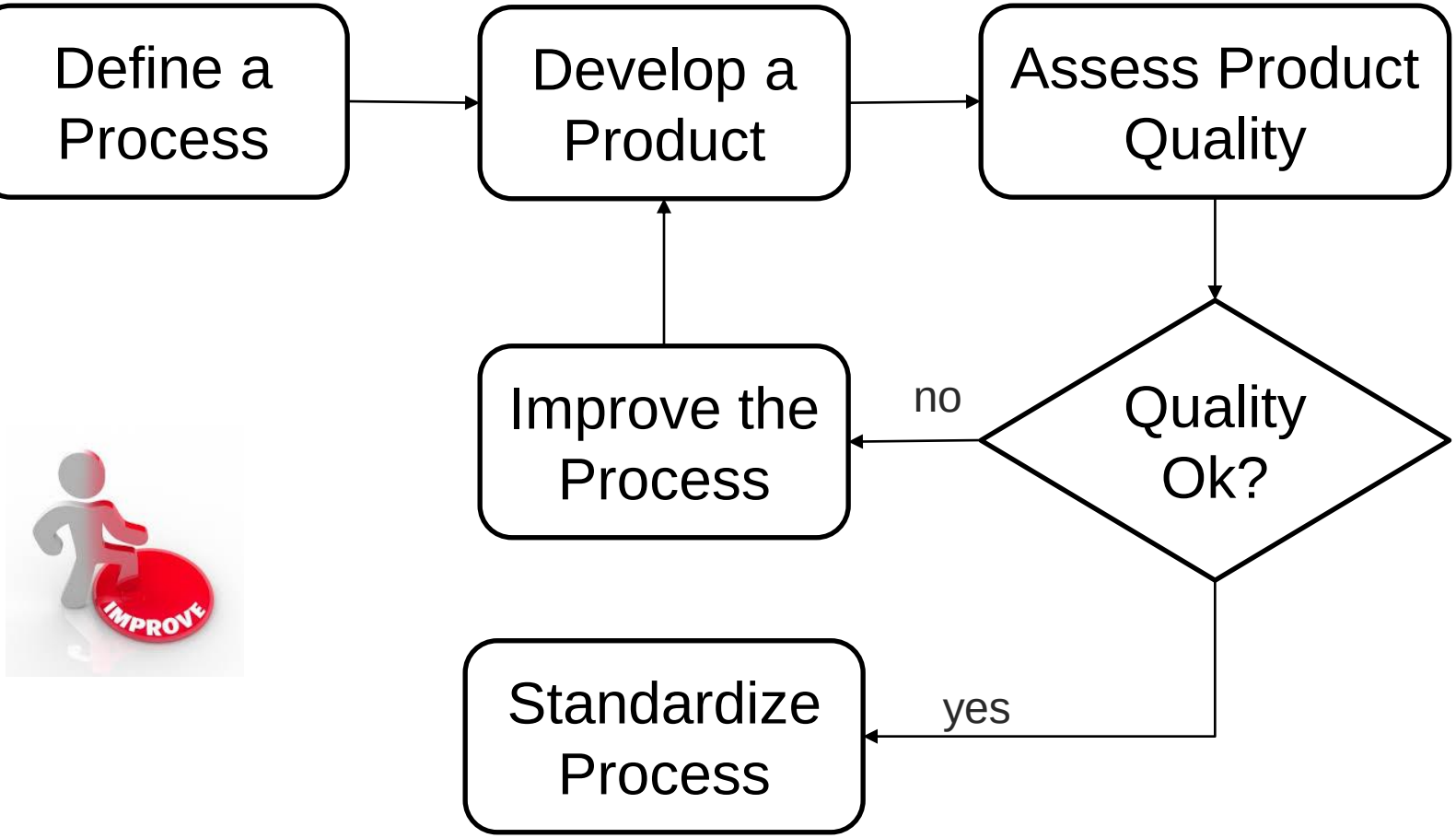
Conflicts of Quality Attributes

- It is not possible for any system to be optimized for all quality attributes
  - For example, improving robustness may lead to lower performance
- The quality plan has to define the most important quality attributes
  - The plan should also define how to assess each important quality attribute

Quality: Process and Product

- In general, the quality of a product is influenced by the quality of the process
- However, the relationship between software processes and product quality is complex
  - Individual skills and experience is important in software development
  - External factors, such as novelty of an application, may impact on product quality

Process-based Software Quality



Bibliography

- Ian Sommerville. **Software Engineering**, 10th Edition. Pearson Education, 2016.
  - Chapter 24 (up to Section 24.1)