Software Quality Management

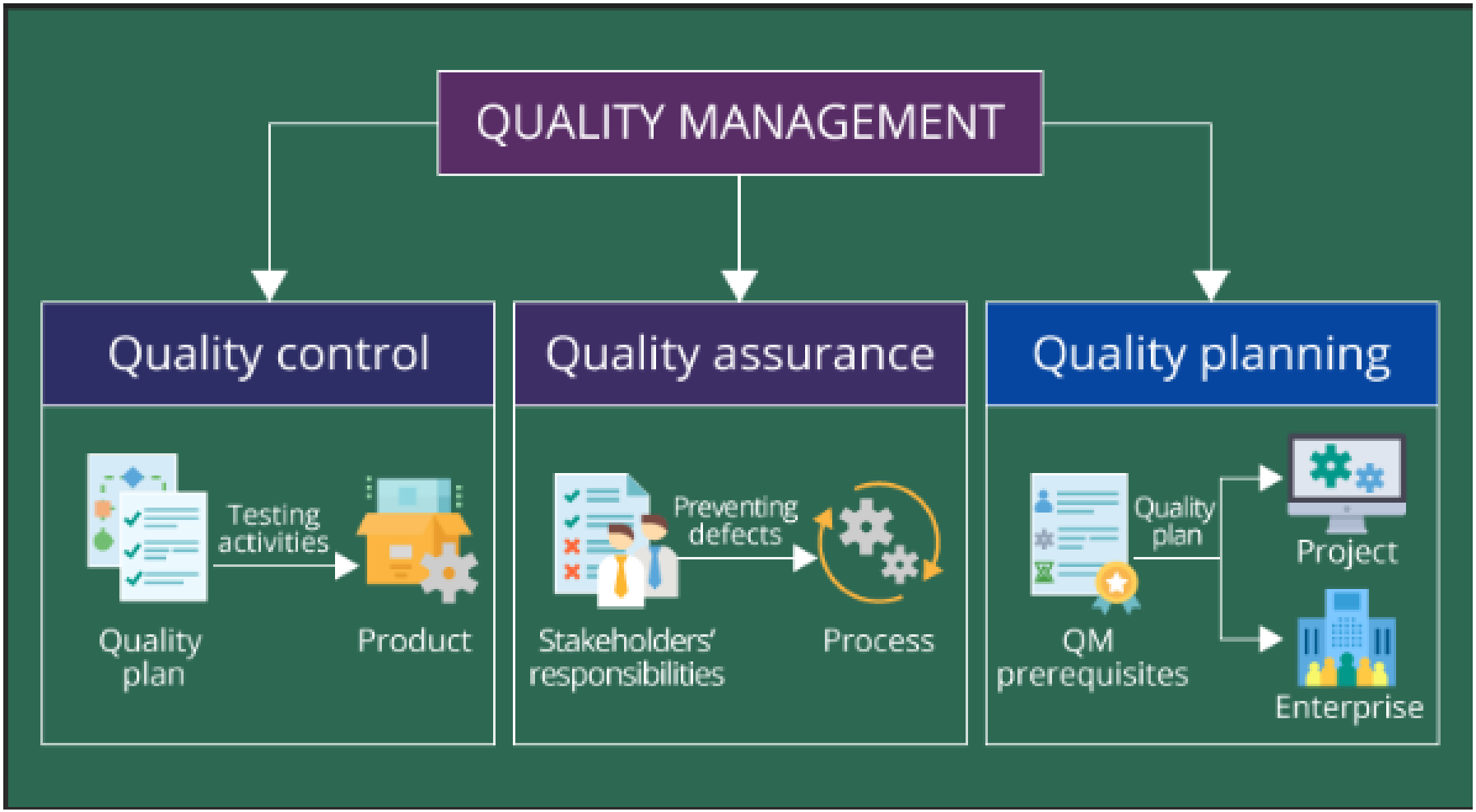
Topics covered

- Process and product quality
- Quality assurance and standards
- Quality planning
- Quality control

Software quality management (SQM)

- Concerned with ensuring that the required level of quality is achieved in a software product.
- Involves defining appropriate quality standards and procedures and ensuring that these are followed.
- Should aim to develop a ‘quality culture’ where quality is seen as everyone’s responsibility.

SQM Aspects


QUALITY MANAGEMENT
Quality control
Testing activities
Quality plan
Product
Quality assurance
Preventing defects
Stakeholders' responsibilities
Process
Quality planning
Quality plan
Project
QM prerequisites
Enterprise

What is quality?

- Quality, simplistically, means that a product should meet its specification.
- This is problematical for software systems
  - There is a tension between customer quality requirements (efficiency, reliability, etc.) and developer quality requirements (maintainability, reusability, etc.);
  - Some quality requirements are difficult to specify in an unambiguous way;
  - Software specifications are usually incomplete and often inconsistent.

The quality compromise

- We cannot wait for specifications to improve before paying attention to quality management.
- We must put quality management procedures into place to improve quality in spite of imperfect specification.
- Quality is subjective and thus it might be difficult to get universally acceptable quality specifications.
- The dynamic nature of user requirements may impact negatively on quality.

Scope of quality management

- Quality management is particularly important for large, complex systems.
- The 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 and should focus on establishing a quality culture.

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.

Cont.

- It is important to note that QA, planning, and QC are not mutually exclusive.
- In fact, they are all essential parts of a comprehensive software quality management (SQM) program.
- By working together, QA, planning, and QC can help to ensure that software is of the highest possible quality.

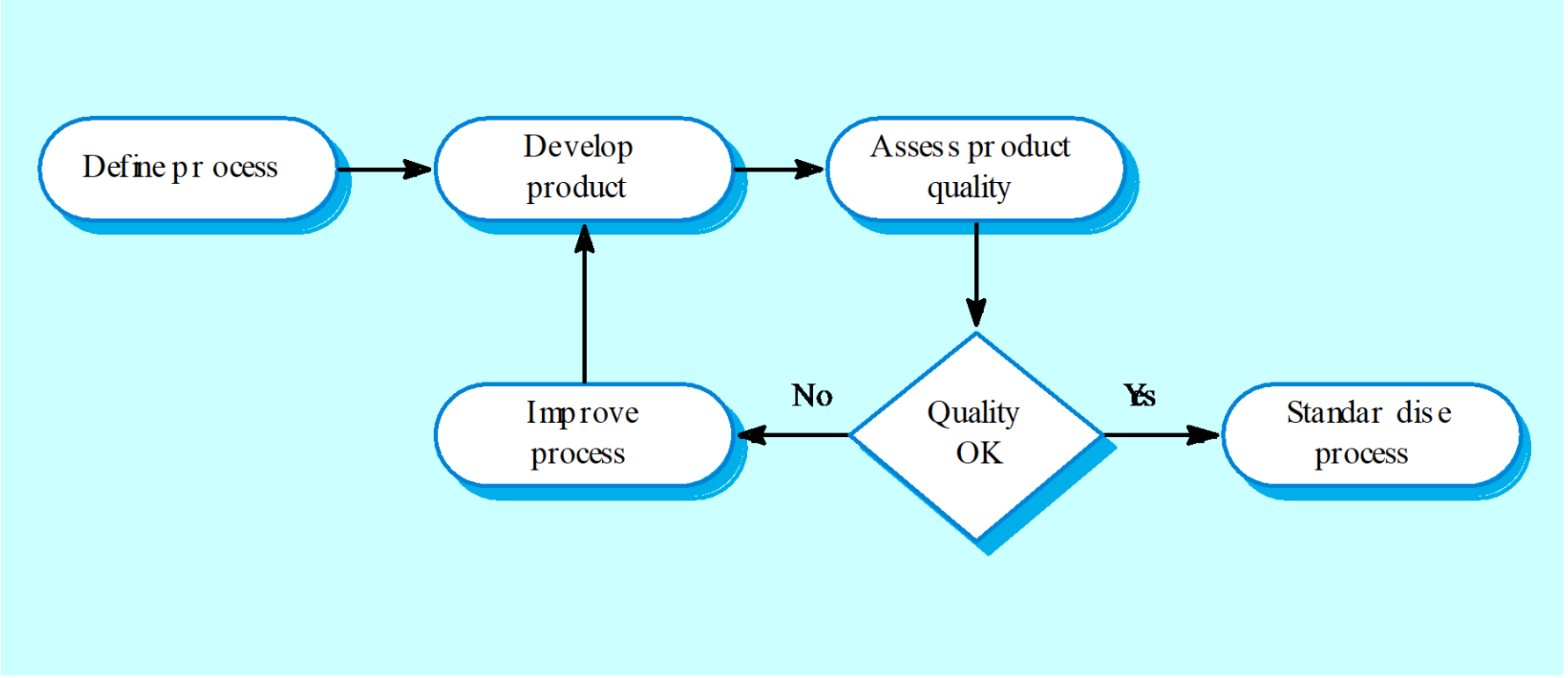
Process and product quality

- The quality of a developed product is influenced by the quality of the production process.
- This is important in software development as some product quality attributes are hard to assess.
- However, there is a very complex and poorly understood relationship between software processes and product quality.

Process-based quality

- There is a straightforward link between process and product in manufactured goods.
- More complex for software because:
  - The application of individual skills and experience is particularly important in software development;
  - External factors such as the novelty of an application or the need for an accelerated development schedule may impair product quality.
- Care must be taken not to impose inappropriate process standards - these could reduce rather than improve the product quality.

Process-based quality



Practical process quality

- Define process standards such as how reviews should be conducted, configuration management, etc.
- Monitor the development process to ensure that standards are being followed.
- Report on the process to project management and software procurer.
- Don’t use inappropriate practices simply because standards have been established.

Software Quality Assurance

- Software quality assurance (QA) is a proactive approach to ensuring the quality of software.
  - It involves establishing and maintaining processes and procedures that help to prevent defects from occurring in the first place.
  - QA also includes activities such as testing, verification, and validation.
- SQA works parallel to software development, an ongoing activity applied throughout the software development life cycle.
- Instead of making quality checks after completion, software quality assurance checks for quality issues in each development phase.

SQA Cont.

- These are the characteristics common to all software quality assurance processes:
  - A defined quality management approach
  - Holding formal technical reviews
  - Implementing a multi-testing strategy
  - Using effective software engineering technology
  - A measurement and reporting mechanism
- Software Standards must also be upheld

Quality assurance and standards

- Standards are the key to effective quality management.
- They may be international, national, organizational or project standards.
- Product standards define characteristics that all components should exhibit e.g. a common programming style.
- Process standards define how the software process should be enacted.

Importance of standards

- Encapsulation of best practice- avoids repetition of past mistakes.
- They are a framework for quality assurance processes - they involve checking compliance to standards.
- They provide continuity - new staff can understand the organisation by understanding the standards that are used.

Problems with standards

- They may not be seen as relevant and up-to-date by software engineers.
- They often involve too much bureaucratic form filling.
- If they are unsupported by software tools, tedious manual work is often involved to maintain the documentation associated with the standards.

Software Quality Planning

- Software quality planning is the process of defining the quality standards for a software project and developing a plan to achieve those standards.
- It is an important part of the software development process, as it helps to ensure that the final product meets the needs of the customer and stakeholders

Quality Planning Process

- The quality planning process typically includes the following steps:
  - ✓**Define the quality standards.** The first step is to define the quality standards for the software project. This includes identifying the features and functions that are required, as well as the non-functional requirements, such as performance, security, and usability.
  - ✓**Develop a plan to achieve the quality standards.** Once the quality standards have been defined, the next step is to develop a plan to achieve them. This plan should include the following:
    - ✓A process for testing the software
    - ✓A process for fixing defects
    - ✓A process for managing changes to the software

Cont.

- ✓**Implement the quality plan.** Once the quality plan has been developed, it is important to implement it. This means putting in place the processes and procedures that are necessary to ensure that the software meets the quality standards.
- ✓**Monitor and improve the quality plan.** The quality plan should not be a static document. It should be monitored and updated as needed to ensure that it is effective in achieving the quality standards.

Quality planning

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

Quality plans

- Quality plan structure
  - Product introduction;
  - Product plans;
  - Process descriptions;
  - Quality goals;
  - Risks and risk management.
- Quality plans should be short, precise documents
  - If they are too long, no-one will read them.

Software quality attributes

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

Quality control

- This involves checking the software development process to ensure that procedures and standards are being followed.
- Also involves checking the product to ensure that it is being built correctly.
- There are two approaches to quality control
  - Quality reviews;
  - Automated software assessment/measurement.

Quality reviews

- This is the principal method of validating the quality of a process or of a product.
- A group examines part or all of a process or system and its documentation to find potential problems.
- There are different types of reviews with different objectives
  - Inspections – for defect removal (product);
  - Reviews – for progress assessment (product and process);
  - Quality reviews – product and standards.

Quality reviews

- The objective is the discovery of system defects and inconsistencies.
- Any documents produced in the process may be reviewed.
- Review teams should be relatively small and reviews should be fairly short.
- Records should always be maintained of quality reviews.

Review functions

- **Quality function** - they are part of the general quality management process.
- **Project management function -** they provide information for project managers.
- **Training and communication function -** product knowledge is passed between development team members.

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 referred to the client.

Software measurement and metrics

- Software measurement is concerned with deriving a numeric value for an attribute of a software product or process.
- This allows for objective comparisons between techniques and processes.
- Although some companies have introduced measurement programmes, most organisations still don’t make systematic use of software measurement.
- There are few established standards in this area.

The End