

Software Quality and Standards



**Idaho State
University**

Computer
Science

Isaac Griffith

CS 3321
Department of Computer Science
Idaho State University

ROAR



Topics Covered

- Software Quality
- Software Standards



Software quality management

- Concerned with ensuring that the required level of quality is achieved in a software product.
- Three principal concerns:
 - At the organizational level, quality management is concerned with establishing a framework of organizational processes and standards that will lead to high-quality software.
 - At the project level, quality management involves the application of specific quality processes and checking that these planned processes have been followed.
 - At the project level, quality management is also concerned with establishing a quality plan for a project. The quality plan should set out the quality goals for the project and define what processes and standards are to be used.

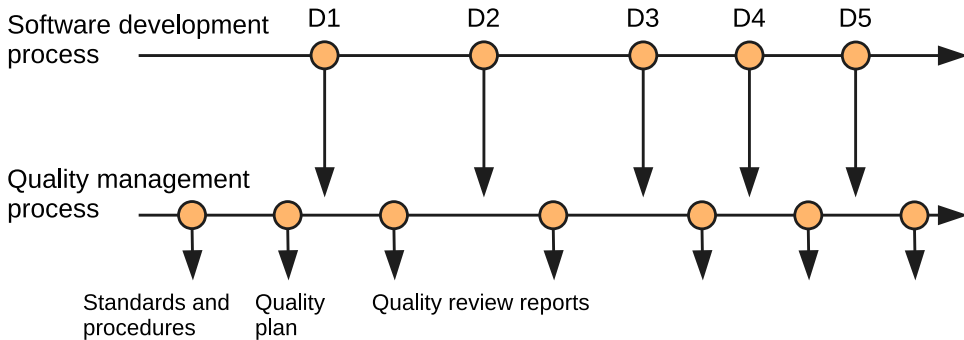


Quality management activities

- Quality management provides an independent check on the software development process.
- The quality management process checks the project deliverables to ensure that they are consistent with organizational standards and goals
- The quality team should be independent from the development team so that they can take an objective view of the software. This allows them to report on software quality without being influenced by software development issues.



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 quality plan should define the quality assessment process.
- It should set out which organizational 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, succinct documents
 - If they are too long, no-one will read them.

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.
- Techniques have to evolve when agile development is used.

Software quality



Software 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 focus may be 'fitness for purpose' rather than specification conformance.

Software fitness for purpose

- Has the software been 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?
- Have programming and documentation standards been followed in the development process?



Non-functional characteristics

- The subjective quality of a software system is largely based on its non-functional characteristics.
- This reflects practical user experience – if the software's functionality is not what is expected, then users will often just work around this and find other ways to do what they want to do.
- However, if the software is unreliable or too slow, then it is practically impossible for them to achieve their goals.



Software quality attributes

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



Quality conflicts

- It is not possible for any system to be optimized for all of these attributes – for example, improving robustness may lead to loss of performance.
- The quality plan should therefore define the most important quality attributes for the software that is being developed.
- The plan should also include a definition of the quality assessment process, an agreed way of assessing whether some quality, such as maintainability or robustness, is present in the product.

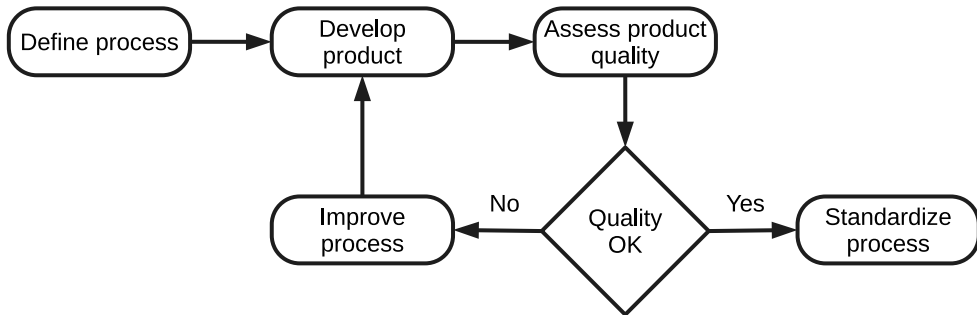


Process and product quality

- 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.
 - 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.



Process-based quality





Quality culture

- Quality managers should aim to develop a 'quality culture' where everyone responsible for software development is committed to achieving a high level of product quality.
- They should encourage teams to take responsibility for the quality of their work and to develop new approaches to quality improvement.
- They should support people who are interested in the intangible aspects of quality and encourage professional behavior in all team members.

Software standards



Software standards

- Standards define the required attributes of a product or process. They play an important role in quality management.
- Standards may be international, national, organizational or project standards.



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 i.e. that organization's view of quality.
- They provide continuity - new staff can understand the organization by understanding the standards that are used.



Product and process standards

- **Product standards**

- Apply to the software product being developed. They include document standards, such as the structure of requirements documents, documentation standards, such as a standard comment header for an object class definition, and coding standards, which define how a programming language should be used.

- **Process standards**

- These define the processes that should be followed during software development. Process standards may include definitions of specification, design and validation processes, process support tools and a description of the documents that should be written during these processes.

Product and process standards

Product standards

Design review form
Requirements document
structure
Method header format
Java programming style
Project plan format
Change request form

Process standards

Design review conduct
Submission of new code for system
building
Version release process
Project plan approval process
Change control process
Test recording process

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 form filling work is often involved to maintain the documentation associated with the standards.



Standards development

- Involve practitioners in development. Engineers should understand the rationale underlying a standard.
- Review standards and their usage regularly. Standards can quickly become outdated and this reduces their credibility amongst practitioners.
- Detailed standards should have specialized tool support. Excessive clerical work is the most significant complaint against standards.
 - Web-based forms are not good enough.

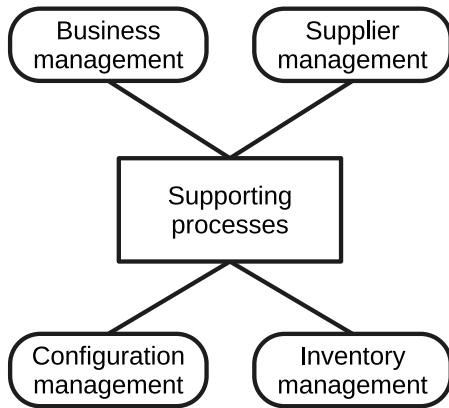
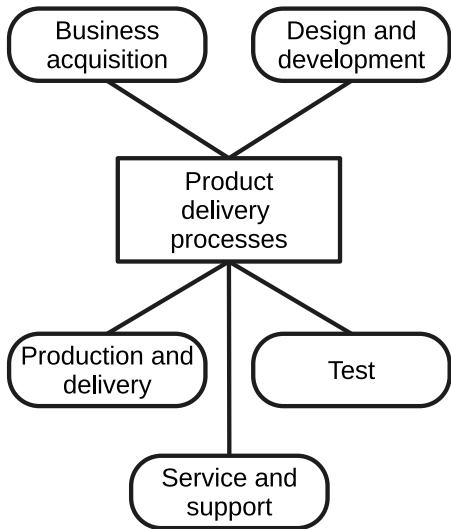


ISO 9001 standards framework

- An international set of standards that can be used as a basis for developing quality management systems.
- ISO 9001, the most general of these standards, applies to organizations that design, develop and maintain products, including software.
- The ISO 9001 standard is a framework for developing software standards.
 - It sets out general quality principles, describes quality processes in general and lays out the organizational standards and procedures that should be defined. These should be documented in an organizational quality manual.

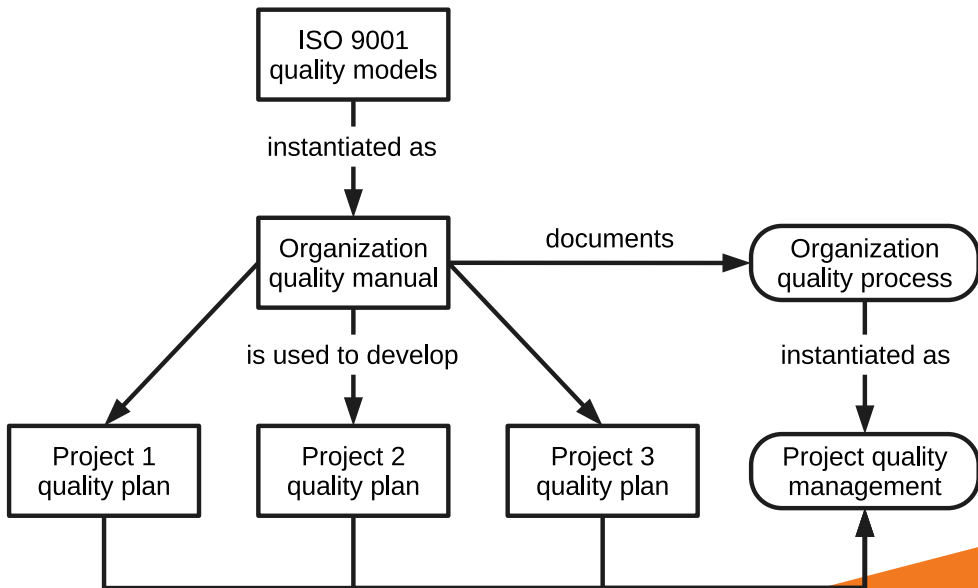


ISO 9001 core processes





ISO 9001 and quality management





ISO 9001 certification

- Quality standards and procedures should be documented in an organizational quality manual.
- An external body may certify that an organization's quality manual conforms to ISO 9000 standards.
- Some customers require suppliers to be ISO 9000 certified although the need for flexibility here is increasingly recognized.

Software quality and ISO9001

- The ISO 9001 certification is inadequate because it defines quality to be the conformance to standards.
- It takes no account of quality as experienced by users of the software. For example, a company could define test coverage standards specifying that all methods in objects must be called at least once.
- Unfortunately, this standard can be met by incomplete software testing that does not include tests with different method parameters. So long as the defined testing procedures are followed and test records maintained, the company could be ISO 9001 certified.



Key points

- Software quality management is concerned with ensuring that software has a low number of defects and that it reaches the required standards of maintainability, reliability, portability etc. Software standards are important for quality assurance as they represent an identification of 'best practice'. When developing software, standards provide a solid foundation for building good quality software.



Are there any questions?