



Lecture 1

Software Quality background and Software Quality Management



Outline

- The software quality challenge
- What is software quality?
 - Popular and professional views
 - Classification of the causes of software errors
 - Software quality definitions
- Software quality assurance
 - Definitions and objectives
 - Quality assurance and standards
- Total quality management



The software quality challenge

- The uniqueness of software quality assurance
 - Software product
 - Software development process
- Software quality assurance (SQA) environment

Factors affecting defect detection in software products vs. other industrial products

Characteristic	Software products	Other industrial products
Complexity	Usually, very complex product allowing for very large number of operational options	Degree of complexity much lower, allowing at most a few thousand operational options
Visibility of product	Invisible product, impossible to detect defects or omissions by sight (e.g. of a diskette or CD storing the software)	Visible product, allowing effective detection of defects by sight
Nature of development and production process	Opportunities to detect defects arise in only one phase, namely product development	Opportunities to detect defects arise in all phases of development and production: <ul style="list-style-type: none">-product development-Product production planning-manufacturing



The uniqueness of the software development process

- High complexity, as compared to other industrial products
- Invisibility of the product
- Opportunities to detect defects (“bugs”) are limited to the product development phase



The uniqueness of software quality assurance

- These differences between *software products* and *other products* create the need for an SQA methodology and tools for SQA that will meet the special and different challenges for the development and operation of quality assurance for software.



Software quality assurance (SQA) environment

- The main characteristics are:
 1. Being contracted
 2. Subjection to customer–supplier relationship
 3. Requirement for teamwork
 4. Need for cooperation and coordination with other development teams



Software quality assurance (SQA) environment (2)

5. Need for interfaces with other software systems
 - *Input interfaces*, where other software systems transmit data to your software system.
 - *Output interfaces*, where your software system transmits processed data to other software systems.
 - Input and output interfaces to the machine's control board, as in medical and laboratory control systems, metal processing equipment, etc.
6. Need to continue carrying out a project while the team changes
7. Need to continue maintaining the software system for years



Software quality assurance (SQA) environment (3)

- These characteristics demand that intensive and continuous managerial efforts be expended in parallel to the professional efforts that have to be invested in order to ensure the project's quality or, in other words, to assure the project's success.



Outline

- The software quality challenge
- **What is software quality?**
 - Popular and professional views
 - Software errors, faults and failures
 - Classification of the causes of software errors
 - Software quality definitions
- Software quality assurance
 - Definitions and objectives
 - Quality assurance and standards
- Total quality management



Quality: Popular views

- “I know it when I see it”
 - Intangible trait, which can be discussed, felt, and judged, but cannot be weighed or measured
- Quality connotes luxury, class and taste
 - Expensive products with sophisticated functionality



What is quality?

- The Oxford English Dictionary gives 16 primary definitions for “quality”.
- Definition 8.c “quality” is defined simply as “excellence, superiority”.



Excellence, superiority?

- The importance of context
 - Excellence, according to whom?
 - Superiority, compared with what?



Dimensions of ICT quality

- Algorithm quality
 - Design: elegance?
 - Implementation: performance, metrics, scalability
- Architecture quality
 - Hardware or software?
 - Elegance?



Dimensions of ICT quality - 2

- Software quality
 - Processes or products?
 - “the degree to which a set of inherent characteristics fulfills requirements” (ISO 9001)
 - Functionality, reliability, usability, efficiency, maintainability, portability (ISO 9126)
 - Three levels of quality (Denning 1992)
 - All basic promises were met.
 - No negative consequences were produced.
 - The customer was delighted.



Dimensions of ICT quality - 3

- User interface quality
 - (Raskin, *The Humane Interface*, 2000)
 - An interface should be
 - effective, habituating, reliable, efficient, and tested



How do we assess quality on any of these dimensions?

- Each dimension has its own approach to assessing quality
 - Quantitative approaches: metrics
 - Qualitative approaches: the “-ilities”



Quality: Professional Views

- Quality was defined as:
 - “conformance to requirements” [Crosby79]
 - “fitness for use” [Juran and Gryna 1970]



Software Quality

- What is software?
- Software errors, faults and failures
 - Software error – grammatical or logical made by a programmer
 - Software faults – improper functioning of the software
 - Software failures – disrupt our use of the software
- **Classification on the causes of software errors**
- Software quality definitions



Causes of Software Errors

- Faulty requirements definition
- Client-developer communication failures
- Deliberate deviations from software requirements
- Logical design errors
- Non-compliance with documentation and coding instructions
- Shortcomings of the testing process
- User interface procedure errors
- Documentation errors



Faulty definition of requirements

- Erroneous definition of requirements
- Absence of vital requirements
- Incomplete definition of requirements
- Inclusion of unnecessary requirements, functions that are not expected to be needed in the near future



Client-developer communication failures

- Misunderstanding of the client's instructions
- Misunderstanding of the client's requirements changes during the development period
 - Written
 - Oral
- Misunderstanding of the client's responses to the design problems presented by the developer
- Lack of attention to client messages



Deliberate deviations from software requirements

- The developer reuses software modules without sufficient analysis of the changes and adaptation needed
- Due to time or budget pressures, the developer decides to omit part of the required functions
- Developer-initiated unapproved improvements to the software



Logical design errors

- Process definitions that contain sequencing errors
- Erroneous definition of boundary conditions
- Omission of required software system states
- Omission of definitions concerning reactions to illegal operation of the software system



Coding errors

- Misunderstanding the design documentation
- Linguistic errors in the programming languages
- Errors in the application of CASE and other development tools



Shortcomings of the testing process

- Incomplete test plans leave untreated portions of software
- Failures to document and report detected errors and faults
- Failure to promptly correct detected software faults as a result of inappropriate indications of the reason for the fault
- Incomplete correction of detected errors due to negligence or time pressures



Documentation errors

- Errors in the design documents and in the coding documentation
- Errors in user manuals and in the “help” displays
 - Omission of software functions
 - Errors in the explanations and instructions given to users
 - Listing of non existing software functions



Outline

- The software quality challenge
- What is software quality?
 - Popular and professional views
 - Software errors, faults and failures
 - Classification of the causes of software errors
 - **Software quality definitions**
- Software quality assurance
 - Definitions and objectives
 - Quality assurance and standards
- Total quality management

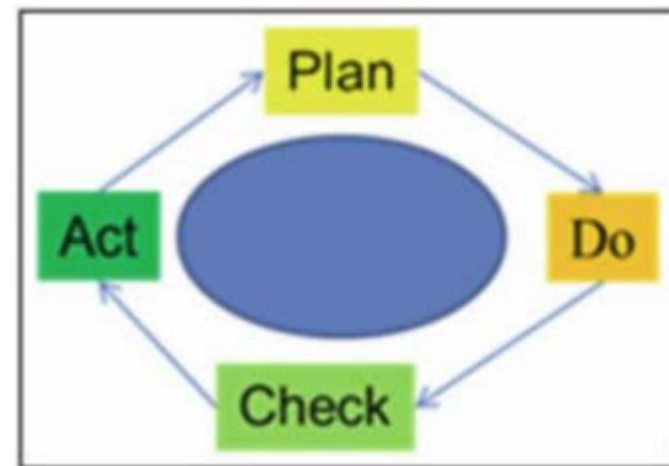
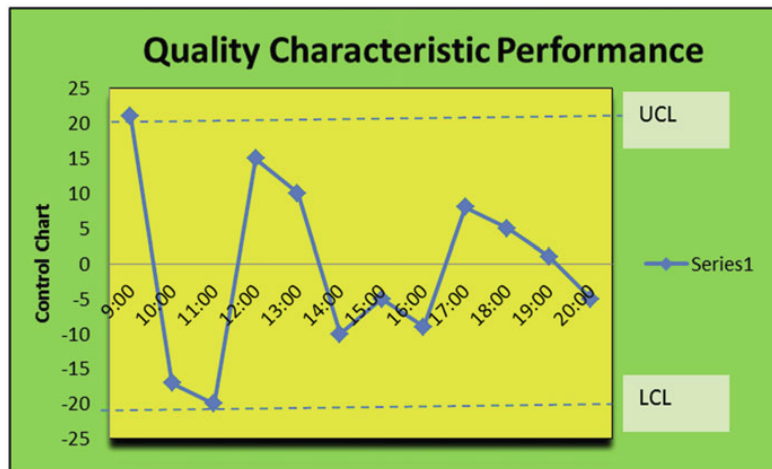


History of quality

- Influence the quality field of
 - Walter Shewhart
 - W. Edwards Deming
 - Joseph Juran
 - Philip Crosby
 - Watts Humphrey
- Other definitions

Walter Shewhart

- founder of statistical process control (SPC)
- The Shewhart model - PDCA Model





Shewhart cycle

Step	Description
Plan	<p>This step identifies an improvement opportunity and outlines the problem or process that will be addressed.</p> <ul style="list-style-type: none">• Select the problem to be addressed.• Describe current process.• Identify the possible causes of the problem.• Find the root cause of problems.• Develop an action plan to correct the root cause.
Do	<p>This step involves carrying out the improvements and it may involve a pilot of the proposed changes to the process</p>
Check	<p>This step involves checking the results obtained against the expected results to determine their effectiveness</p>
Act	<p>This step includes the analysis of the results to adjust process performance to achieve the desired results.</p>

Deming



- It is first necessary that people know what to do, and there must be a constancy of purpose from all individuals to ensure success
- proposed 14 principles to transform the western style of management of an organization to a quality and customer focused organization.



Deming-14 step programme

Step	Description
Constancy of purpose	Companies face short-term and long-term problems. The problems of tomorrow require long-term planning on new products, training, and innovation. This requires resources invested in research and development and continuous improvement of existing products and services.
Adopt new philosophy	Deming outlined the five deadly diseases which afflicted US companies. These included lack of purpose and an excessive interest in short-term profits.
Build quality in	Deming argued that performing mass inspections is equivalent to planning for defects as they are too late to improve quality. Consequently, it is necessary to improve the production process to build the quality into the product
Price and quality	Deming argued against the practice of awarding business on the basis of price alone, as the price of a product or service is meaningless unless there is an objective measure of the quality of the product or service being purchased.



Deming-14 step programme 2

Step	Description
Continuous improvement	There must be continuous improvement in all areas, including understanding customer requirements, design, manufacturing and test methods
Institute training	The organization must be a learning organization and this involves setting up a training program to educate management and staff about the company, customer needs, and pride of workmanship in the products. Supervisors and managers need training on the 14 point program to ensure they fully understand the enhanced contribution that their staff can make if barriers to good work are removed.
Institute leadership	Deming argues that management is about leadership and not supervision. Management should work to remove barriers, know the work domain in depth, and seek innovative solutions to resolve quality and other relevant issues.
Eliminate fear	The presence of fear is a barrier to an open discussion of problems and the identification of solutions or changes to prevent problems from arising.



Deming-14 step programme 3

Step	Description
Eliminate barriers	The objective here is to break down barriers between different departments and groups. It is not enough for each group to optimize its own area: instead, what is required is for the organization to be working as one team
Eliminate slogans	Deming argued that slogans do not help anyone to do a better job. Slogans may potentially alienate staff or encourage cynicism. Deming criticized slogans such as "Zero Defects" or "Do it right the first time" as inappropriate, as how can it be made right first time if the production machine is defective. Most problems are due to the system rather than the person. A slogan is absolutely inappropriate unless there is a clearly defined strategy to attain it, as otherwise the result is the opposite effect to that intended.
Eliminate numerical quotas	Deming argued that quotas act as an impediment to improvement in quality, as quotas are normally based on what may be achieved by the average worker. People below the average cannot make the rate and the result is dissatisfaction and turnover. Thus, there is a fundamental conflict between quotas and pride of workmanship.
Pride of	The intention here is to remove barriers that rob people of pride of



Deming-14 step programme 4

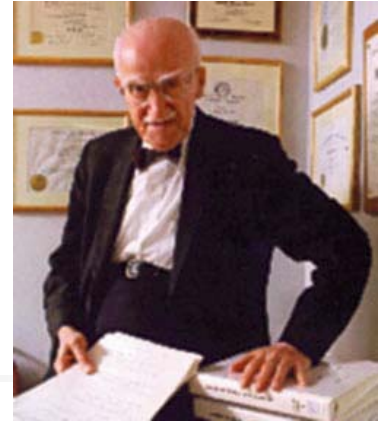
Step	Description
Self improvement	This involves encouraging education and self-improvement for everyone in the company
Take action	This requires that management agree on direction using the 14 principles, communicate the reasons for changes to the staff, and train the staff on the 14 principles.



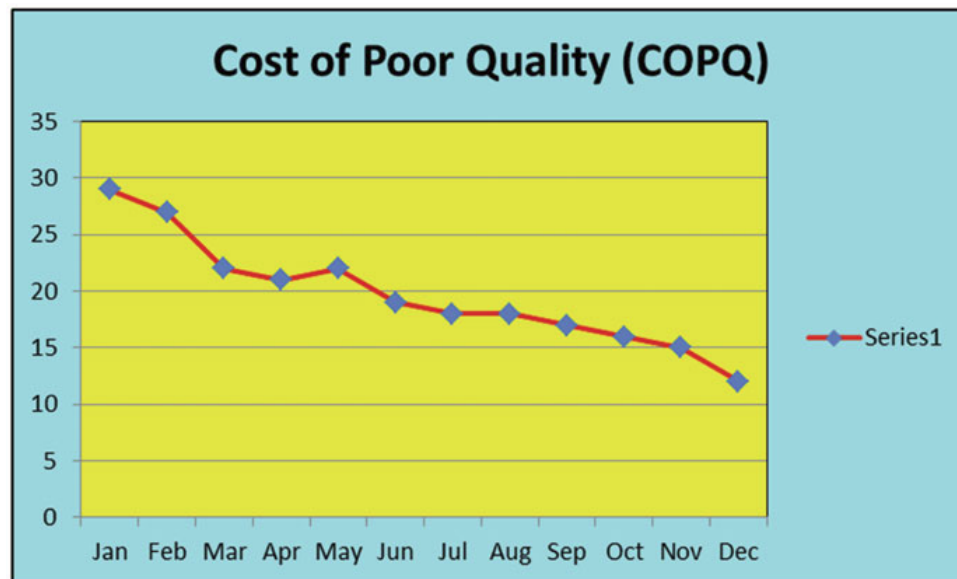
Deming – Five deadly diseases

Disease	Description
Lack of constancy of purpose	Management is too focused on short term thinking rather than long-term improvements
Emphasis on short term profit	A company should aim to become the world's most efficient provider of product/service. Profits will then follow.
Evaluation of performance	Deming is against annual performance appraisal and rating.
Mobility of management	Mobility of management frequently has a negative impact on quality.
Excessive measurement	Excessive management by measurement

Joseph Juran



- Quality - “fitness for use” - quality issues are the direct responsibility of management
- “Juran Trilogy” - quality planning, control, and improvement





Joseph Juran 2

- Quality planning consists of setting quality goals, developing plans, and identifying resources to meet the goals.
- Quality control consists of evaluating performance, setting new goals, and taking action.
- Quality improvement consists of improving delivery, eliminating wastage and improving customer satisfaction.



Juran' s10 step programme

Step	Description
Identify customers	This includes the internal and external customers of an organization, e.g., the testing group is an internal customer and the end user of the software is the external customer
Determine customer needs	Customer needs are generally expressed in the language of the customer' s organization. There is a need to elicit and determine the actual desired requirements with further communication with the customer understand the enhanced contribution that their staff can make if barriers to good work are removed.
Translate	This involves translating the customer needs into the language of the supplier.
Units of measurement	This involves defining the measurement units to be used
Measurement programme	This involves setting up a measurement program in the organization, and includes internal and external measurements of quality and process performance



Juran' s10 step programme 2

Step	Description
Develop product	This step determines the product features to meet the needs of the customer.
Optimize product design	The intention is to optimize the design of the product to meet the needs of the customer and supplier
Develop process	This involves developing processes which can produce the products to satisfy the customer' s needs
Optimize process capability	This involves optimizing the capability of the process to ensure that high quality products are produced
Transfer	This involves transferring the process to normal product development operations



Crosby

- his philosophy - doing things right the first time, i.e., the zero defects (ZD) program.
- Quality is defined as “conformance to the requirements” and he argues that people have been conditioned to believe that error is inevitable.
- Crosby notes that defects are due to two main reasons:
 - lack of knowledge or a
 - lack of attention of the individual



Crosby

- The net effect of a successful implementation of a zero defects program is higher productivity due to less reworking of defective products. Thus, quality, in effect, is free.
- Crosby's approach to achieve the desired quality level of zero defects was to put a quality improvement program in place. He outlined a 14 step quality improvement program



Crosby's 14 step programme

Step	Description
Management commitment	Management commitment and participation is essential for the success of the quality improvement program. The profile of quality is raised within the organization
Quality improvement team	This involves the formation of an organization-wide cross-functional team consisting of representatives from each of the departments
Quality measurement	The objective of quality measurements is to determine the status of quality in each area of the company and to identify areas where improvements are required
Cost of quality evaluation	The cost of quality is an indication of the financial cost of quality to the organization. The cost is initially high, but as the quality improvement becomes effective there is a reduction in the cost of quality



Crosby's 14 step programme 2

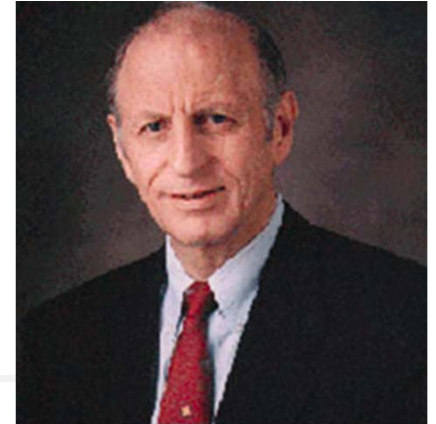
Step	Description
Quality awareness	This involves sharing the cost of poor quality with staff, and motivating staff to identify corrective actions to address quality issues
Corrective action	This involves resolving any problems which have been identified, and bringing any problems which cannot be resolved to the attention of management
Zero defect program	The next step is to communicate the meaning of zero defects to the employees The key point is that it is not a motivation program: instead, it means doing things right the first time, i.e., zero defects
Supervisor training	This requires that all supervisors and managers receive training on the 14 step quality improvement program
Zero defects day	This involves setting aside 1 day each year to high-light zero defects, and its importance to the company



Crosby's 14 step programme 3

Step	Description
Goal setting	This phase involves getting people to think in terms of goals and achieving the goals
Error cause removal	This phase identifies any roadblocks or problems which prevent employees from performing error-free work. The list is produced from the list of problems or road-blocks for each employee.
Recognition	This involves recognizing employees who make out-standing contributions to quality improvement
Quality councils	This involves bringing quality professionals together on a regular basis to share ideas on action
Do it over again	The principle of continuous improvement is a key part of the programme. Improvement is continuous

Watts Humphrey



- is considered the father of software quality
- established the software process programme at the SEI, and this led to the development of the software Capability Maturity Model (CMM) and its successors. Humphries asked questions such as:
 - How good is the current software process?
 - What must I do to improve it?
 - Where do I start?
- Humphries focused his later efforts to developing the Personal Software Process (PSP) and the Team Software Process (TSP)



Software quality definitions - 1

- Software quality is:
 - The degree to which a system, component, or process meets specified requirements
 - The degree to which a system, component, or process meets customer or user needs or expectations

IEEE definition



Software quality definitions - 2

- Software quality is defined as:
 - Conformance to explicitly stated functional and performance requirements, explicitly documented development standards, and implicit characteristics that are expected of all professionally developed software.

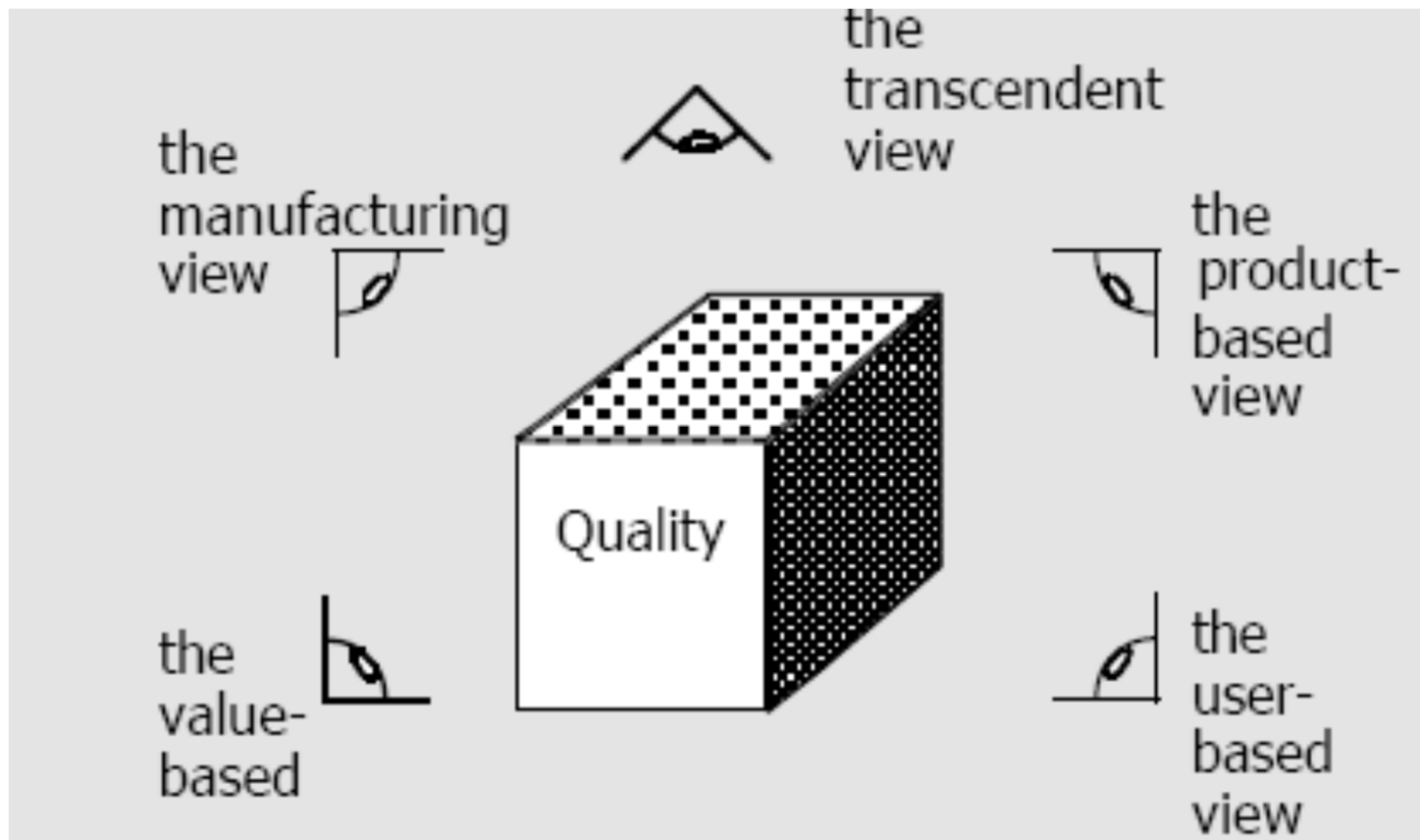
Pressman' s definition



Software quality definitions - 2

- Pressman's definition suggests 3 requirements for quality assurance:
 - Specific functional requirements, which refer mainly to the output of the software system
 - The software quality standards mentioned in the contract
 - Good Software Engineering Practices (GSEP), reflecting state-of-the-art professional practices

Software quality views - 1





Software quality views - 2

- Product view
 - Lack of “bugs” in the product
- Process view
 - Intermediate stages,
 - Intermediate users,
 - Intermediate deliverables with certain quality attributes



Software quality views - 3

- A definition in Steve McConnell's “Code Complete” divides software into two pieces: internal and external quality characteristics.
 - External quality characteristics are those parts of a product that face its users,
 - internal quality characteristics are those that do not face its users



Software quality views - 4

- Product view

- Defect rate

- Number of defects per million lines of source code, per function point, per other unit

- Reliability

- Number of failures per n hours of operation,
 - Mean time to failure,
 - Probability of failure-free operation in a specified time



Software quality views

- Customer satisfaction
 - Overall satisfaction with the software product
 - Satisfaction toward specific attributes
 - Capability
 - Usability
 - Performance
 - Reliability
 - Installability
 - Maintainability
 - Documentation
 - Availability



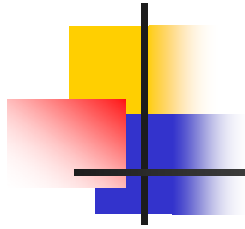
Software quality views

- Process view – need of
 - Measures of the characteristics and quality parameters of the development process and its stages
 - Metrics and models to ensure that the development process is under control and moving toward the product's quality objectives



Outline

- The software quality challenge
- What is software quality?
 - Popular and professional views
 - Software errors, faults and failures
 - Classification of the causes of software errors
 - Software quality definitions
- **Software quality assurance**
 - **Definitions and objectives**
 - **Quality assurance and standards**
- Total quality management



Software quality assurance

- Software quality assurance definitions
- Software quality assurance compared with software quality control
- The objectives of Software quality assurance



Software quality assurance - SWEBOOK

- SQA processes provide assurance that the software products and processes in the project life cycle conform to their specified requirements by planning, enacting, and performing a set of activities to provide adequate confidence that quality is being built into the software.



Software quality assurance definitions - 1

- Software quality assurance is:
 - A planned and systematic pattern of all actions necessary to provide adequate confidence that an item or product conforms to established technical requirements
 - A set of activities designed to evaluate the process by which the products are developed or manufactured.

IEEE definition



Software quality assurance definitions - 2

- Characteristics of the definition:
 - Plan and implement systematically
 - Refer to the software development process
 - Refer to the specifications of the technical requirements
- Directions to broaden the definition
 - Software quality assurance should not be limited to the development process
 - Software quality assurance actions should not be limited to the technical aspects of the functional requirements



Software quality assurance definitions - 3

- Software quality assurance is:
 - A systematic, planned set of actions necessary to provide adequate confidence that the software development process or the maintenance process of a software system product conforms to established functional technical requirements as well as with the managerial requirements of keeping the schedule and operating within the budgetary confines.

Expanded definition by D. Galin



Software quality assurance vs. software quality control

- **Quality control** is defined as a set of activities designed to evaluate the quality of a developed or manufactured product

IEEE

- The main objective of **quality assurance** is to minimize the cost of guaranteeing quality by a variety of activities performed throughout the development and manufacturing process/stages.



The objectives of software quality assurance activities - 1

- Software development (process-oriented)
 - Assuring an acceptable level of confidence that the software will conform to functional technical requirements
 - Assuring an acceptable level of confidence that the software will conform to managerial scheduling and budgetary requirements
 - Initiating and managing activities for the improvement and greater efficiency of software development and SQA activities



The objectives of software quality assurance activities - 2

- Software maintenance (product-oriented)
 - Assuring an acceptable level of confidence that the software maintenance activities will conform
 - to the functional technical requirements
 - to managerial scheduling and budgetary requirements
 - Initiating and managing activities to improve and increase the efficiency of software maintenance and SQA activities



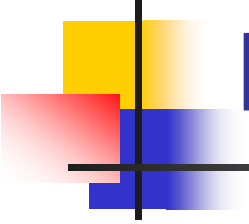
Outline

- The software quality challenge
- What is software quality?
 - Popular and professional views
 - Software errors, faults and failures
 - Classification of the causes of software errors
 - Software quality definitions
- Software quality assurance
 - Definitions and objectives
 - Quality assurance and standards
- **Total quality management**



Total Quality Management

- Style of management aimed at achieving long-term success by linking quality and customer satisfaction
- Creation of culture in which all members of the organisation participate in the improvement of processes, products and services.



The key elements of Total Quality Management system -1

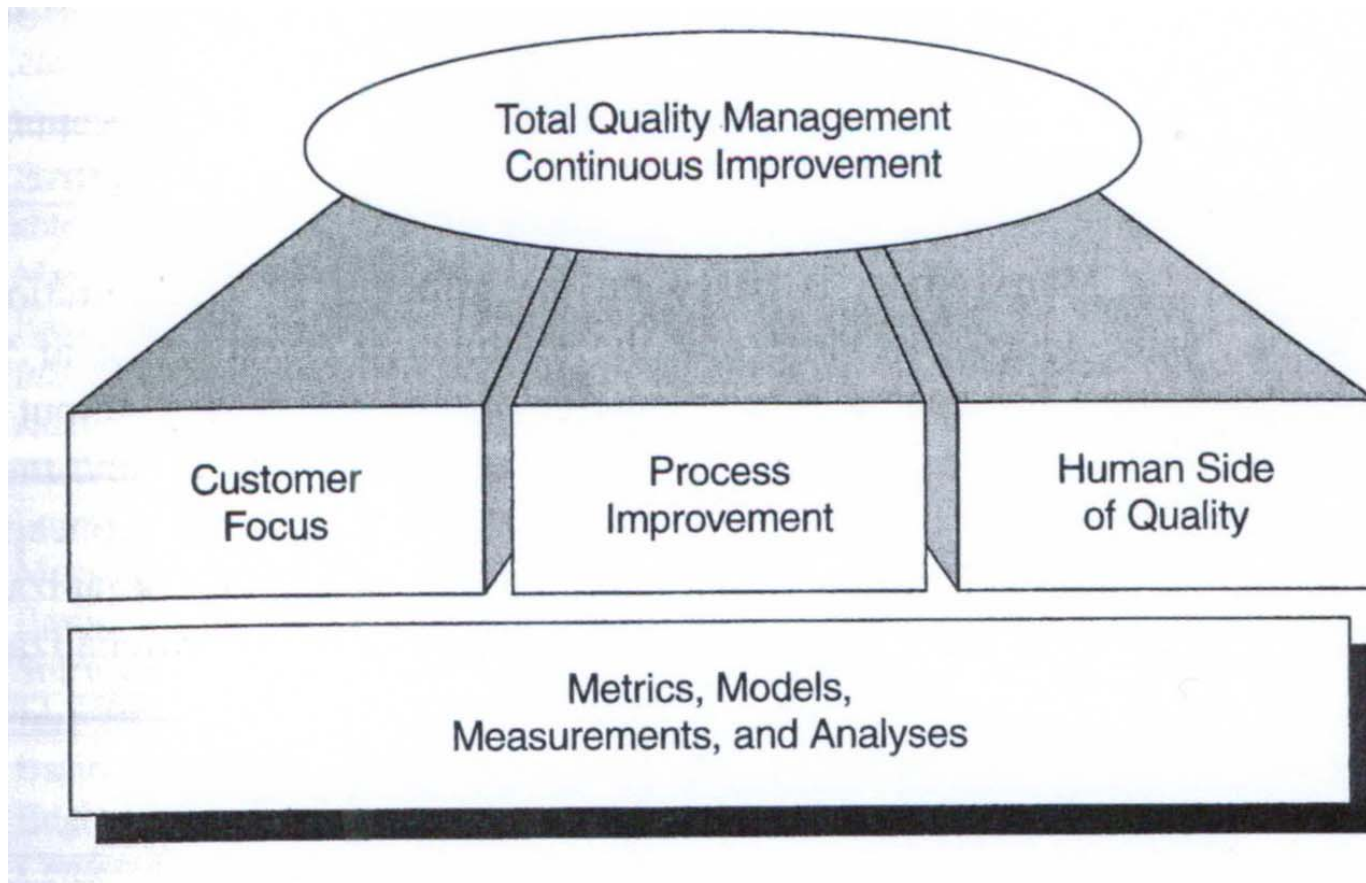
- Customer focus
 - Studying customers' wants and needs
 - Gathering customers' requirements
 - Measuring and managing customers' satisfaction
- Process
 - The objective is to reduce process variations and to achieve continuous process improvement



The key elements of Total Quality Management system -2

- Human side of quality
 - The objective is to create a companywide quality culture
- Measurement and analyses
 - The objective is to drive continuous improvement in all quality parameters by the goal-oriented measurement system

Total Quality Management - 2





Quality Management System

- A Quality Management System is
 - the organisational structure,
 - responsibility,
 - procedures,
 - activities,
 - capabilities and
 - resources

that together aim to ensure that software products will satisfy stated or implied needs.



Why Quality Management?

- Quality Management reduces the cost of failure:
 - costs of correcting defects, both before and after delivery;
 - overruns against time and budget;
 - unnecessary high maintenance costs;
 - indirect costs which users incur due to poor quality software

Survey data indicates that for a company with a turnover of £3m per year, failure costs are likely to be in the order of £600K or 20% of turnover, and saved costs due to implementing a quality management system are likely to be in the 25% – 50% of failure costs therefore saving £150K – £300K.



Perspective

- We will be looking at quality management primarily from a "supplier" perspective as distinct from a "purchaser" perspective
- however, software engineers may frequently find themselves on the purchaser side and will have to design their quality management system appropriately



Quality policy

- "The suppliers management should **define** and **document** its policy and objectives for, and commitment to, quality. The supplier should ensure that this policy is understood, implemented and maintained at *all levels* in the organisation."



Quality system

- Brings together the **functions, objectives** and **activities** that contribute to the product's or service's consistent quality...
- Should be documented, generally in the form of a **quality manual**, which must be...
 - appropriate,
 - concise,
 - practical,
 - up-to-date,
 - correspond to what really happens,
 - distributed to all relevant staff,
 - effectively implemented.



Organization

- a complete organisation structure is required showing the duties, responsibilities and authority of all staff who manage verify or perform work affecting quality;
- a management representative, with defined responsibilities and authority, needs to be nominated who will be responsible for all matters affecting the quality system;
- staff responsible for the verification of any or all aspects of the quality system must be properly skilled and trained.



TQM system

- The development of high quality software requires a good software development process to be in place, and this includes best practices in software engineering for:
 - Project management
 - Estimation methodology
 - Risk management process.
 - Requirements Development and Management



TQM system 2

- Design and Development
- Software development lifecycles
- Quality assurance/management
- Software inspections
- Software testing
- Supplier Selection and Management
- Configuration management
- Customer satisfaction process
- Continuous improvement



Summary

- Definition of software, software quality and software quality assurance
- Identification the various causes of errors and distinguish between software errors, faults and failures
- Explanation of the objectives of quality assurance activities and explain the difference with quality control
- Explanation of the Total Quality Management philosophy



Summary - 2

- Quality is the key to successful software development. To achieve quality in a software product or service requires planning, analysis and control of quality at every stage in development
- To do this quality management system must be developed and documented. This system and its development should itself be subject to analysis and control procedures.