CS577b Spring 2001
January 31, 2001

Software Quality Management – 2
A. Winsor Brown

Goals of Presentation

You should get an overview of Quality Management

- What it is & where it fits
- Principles behind it
- How it's distinct from QA and QI

We'll discuss quality property models

- What and when used
- Their application

Some Definitions – SEI's CMM

- quality - (1) The degree to which a system, component, or process meets specified requirements. (2) The degree to which a system, component, or process meets customer or user needs or expectations. [IEEE-STD-610]
- software plans - The collection of plans, both formal and informal, used to express how software development and/or maintenance activities will be performed. Examples of plans that could be included: software development plan, software quality assurance plan, software configuration management plan, software test plan, risk management plan, and process improvement plan.

Some Definitions – SEI's CMM (cont.)

- software quality assurance - (1) A planned and systematic pattern of all actions necessary to provide adequate confidence that a software work product conforms to established technical requirements. (2) A set of activities designed to evaluate the process by which software work products are developed and/or maintained.
- software quality goal - Quantitative quality objectives defined for a software work product.
- software quality management - The process of defining quality goals for a software product, establishing plans to achieve these goals, and monitoring and adjusting the software plans, software work products, activities, and quality goals to satisfy the needs and desires of the customer and end users.

Sommerville's Definitions(?)

Quality Management is a Process with [key process activities]

- Quality Assurance
  - The establishment of a framework of organizational procedures and standards which lead to high-quality software
- Quality Planning
  - The selection of appropriate procedures and standards from this framework and adaptation of these for a specific software project.
- Quality Control
  - The definition and enactment of processes which ensure that the project quality procedures and standards are followed by the software development team.

SEI Perspectives – CMM Levels

- Level 4: Managed - Detailed measures of the software process and product quality are collected. Both the software process and products are quantitatively understood and controlled.
- Level 5: Optimizing - Continuous process improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.
- Level 2: Repeatable
  - Software Quality Assurance KPA
  - The purpose of Software Quality Assurance is to provide management with appropriate visibility into the process being used by the software project and of the products being built.

SEI's Level 2 SQA KPA

- Software Quality Assurance involves reviewing and auditing the software products and activities to verify that they comply with the applicable procedures and standards and providing the software project and other appropriate managers with the results of these reviews and audits.

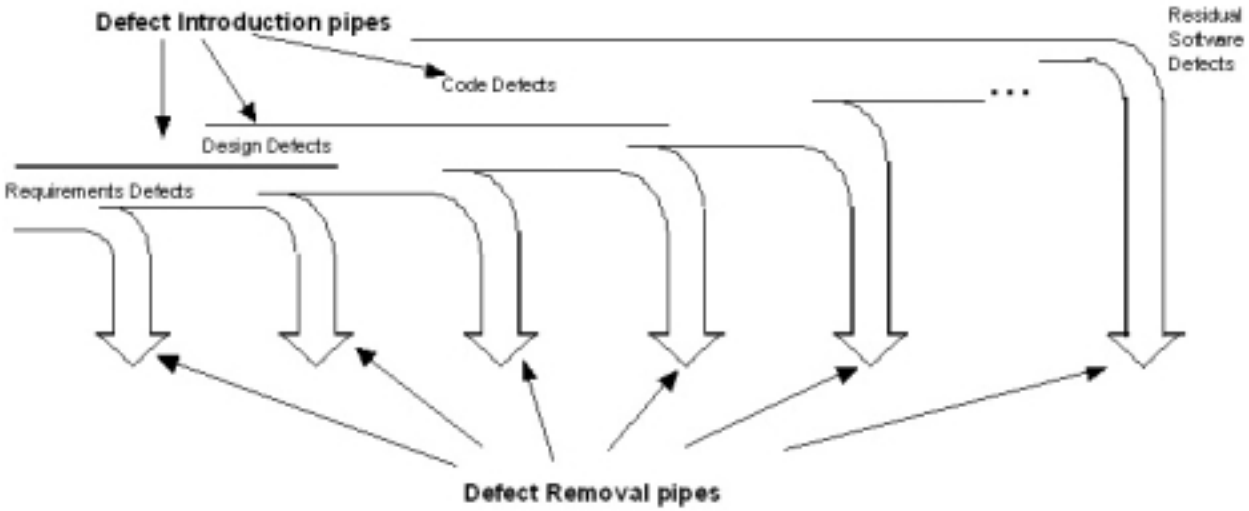
Software Quality Assurance (SQA) "Goals"

- SQA activities are planned.
- Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively.
- Affected groups and individuals are informed of software quality assurance activities and results.
- Noncompliance issues that cannot be resolved within the software project are addressed by senior management.

Quality Management Manifestations in MBASE

Quality Management
= Quality Assurance + Quality Improvement

Quality Assurance and Quality Improvement require Quality Assessment (against property models)



Defect Identification Techniques (AKA Assessment Methods)

Inspection

Testing: Formal and informal (debugging)

User Feedback: Formal and informal

Reviews

- External (technical?)
- Peer Reviews
  - Fagan's inspections
  - Other inspection methods
  - Buddy Check
  - Desk Check; Personal (PSP) Review
  - Walkthroughs
- Management Reviews
- Management & Technical Reviews

Removal Technique – Automated Analysis

Rating	Automated Analysis
Very Low	Simple compiler syntax checking.
Low	Basic compiler or additional tools capabilities for static module-level code analysis, and syntax- and type-checking.
Nominal	All of the above, plus: Some compiler extensions for static module and inter-module level code analysis, and syntax- and type-checking. Basic requirements and design consistency; and traceability checking.
High	All of the above, plus: Intermediate-level module and inter-module code syntax and semantic analysis. Simple requirements/design consistency checking across views.
Very High	All of the above, plus: More elaborate requirements/design view consistency checking. Basic distributed-processing and temporal analysis, model checking, symbolic execution.
Extra High	All of the above, plus: Formalized* specification and verification. Advanced distributed-processing and temporal analysis, model checking, symbolic execution. *Consistency-checkable pre-conditions and post-conditions, but not mathematical theorems.

Removal Technique – Reviews

Rating	Peer Reviews
Very Low	No people reviews.
Low	Ad-hoc informal walkthroughs Minimal preparation, no follow-up.
Nominal	Well-defined sequence of preparation, review, minimal follow-up. Informal review roles and procedures.
High	Formal review roles and procedures applied to all products using basic checklists*, follow up.
Very High	Formal review roles and procedures applied to all product artifacts & changes; formal change control boards. Basic review checklists, root cause analysis. Use of historical data on inspection rate, preparation rate, fault density.
Extra High	Formal review roles and procedures for fixes, change control. Extensive review checklists, root cause analysis. Continuous review process improvement. User/Customer involvement, Statistical Process Control.

* Checklists are lists of things to look for or check against (e.g. exit criteria)

Removal Technique – Test

Rating	Execution Testing and Tools
Very Low	No testing.
Low	Ad-hoc testing and debugging. Basic text-based debugger
Nominal	Basic unit test, integration test, system test process. Basic test data management, problem tracking support. Test criteria based on checklists.
High	Well-defined test sequence tailored to organization (acceptance, alpha, beta, flight, etc. test). Basic test coverage tools, test support system. Basic test process management.
Very High	More advanced test tools, test data preparation, basic test oracle support, distributed monitoring and analysis, active assertion checking. Metrics-based test process management.
Extra High	Highly advanced tools for test oracles, distributed monitoring and analysis, assertion checking. Integration of automated analysis and test tools. Model-based test process management.

577/MBASE Quality Data Collection Forms

Assessment Activity Based

- Review_Form_v3.xls
- 577-InspectionForms-v5.xls
- User_Feedback_Form_v1b.xls
- Testing_Form_v1.xls

Change Activity Based

- Change request (formal/informal) summary
  try using User_Feedback_Form_v1b.xls

Software Quality Plan

Identify what Assessment Activities will be used
[by reference is best, use inline definition ONLY if necessary]

Identify what "Quality Reports" will be generated

- For all assessment activities
- For all postfacto changes

Identify when [in a relative fashion is OK] the Activities takes place and/or Quality Reports are generated