Chapter 8: Software Quality Assurance

“What is not tracked is not done”

In software, so many things need to be done management cannot track all of them. So, some organization needs to do the tracking. That is the role of software quality assurance (SQA).

SQA is designed to insure that officially established processes are being implemented and followed. Specifically, SQO insures that:

- An appropriate development methodology is in place;
- The projects use standards and procedures in their work;
- Independent reviews and audits are conducted;
- Documentation is produced to support maintenance and enhancement;
- Documentation is produced during development and not after development;
- Mechanisms are in place and used to control changes;
- Testing emphasizes all the high-risk product areas;
- Each software task is satisfactorily completed before the next one begins;
- Deviations from standards and procedures are exposed ASAP;
The project is audible by external professionals;
The quality control work is itself performed against established standards;
The SQA plan and the software development plan are compatible.

The goals of SQA are:
• To improve software quality by approximately monitoring both the software and the development process that produces it;
• To insure full compliance with the established standards and procedures for the software and the software process;
• To insure that inadequacies in the product, the process, or the standards are brought to managements’ attention so these inadequacies can be fixed.
THE ROLE OF SQA
The people responsible for the software projects are the only ones who can be responsible for quality. The role of SQA is to monitor the way these groups perform their responsibilities. In doing this, there are several pitfalls:

- It is a mistake to assume that SQA staff can do anything about quality;
- The existence of SQA does not insure that standards and procedures will be followed;
- Unless management demonstrates its support for SQA by following their recommendations, SQA will be ineffective;
- Unless line management requires that SQA try to resolve their issues with project management before escalation, SQA and development will not work together effectively.
All SQA can do is alert management to deviations from established standards and practices. Management must then insist that the quality problems be fixed before the software is shipped; otherwise, SQA becomes an expensive bureaucratic exercise.
SQA RESPONSIBILITIES
SQA can be effective when it reports through an independent management chain, when it is properly staffed, and when it sees its role as supporting the development and maintenance personnel in improving product quality. Then, SQA should be given the following responsibilities:

• Review all development and quality plans for completeness;
• Participate as inspection moderators in design and code inspections;
• Review all test plans for adherence to standards;
• Review a significant sample of all test results to determine adherence to plans;
• Periodically audit SCM performance to determine adherence to standards;
• Participate in all project quarterly and phase reviews and register non-concurrence if appropriate standards and procedures have not be reasonably met.

SQA FUNCTIONS
Before establishing an SQA function, the basic organizational framework should include the following:
  • Quality assurance practices - Adequate development tools, techniques, methods, and standards are defined and available for Quality Assurance review;
  • Software project planning evaluation - If not defined at the outset, they will not be implemented;
  • Requirements evaluation - Initial requirements must be reviewed for conformance to quality standards;
  • Evaluation of the design process-
  • Evaluation of coding practices -
• Evaluation of the software integration and test process -
• In-process evaluation of the management and project control process -

SQA REPORTING
SQA reporting should not be under the software development manager. SQA should report to a high enough management level to have some chance of influencing priorities and obtaining enough resources to fix key problems. However, lower-level reporting normally results in better working relationships with developers, while the ability to influence priorities is reduced.
Some general guidelines are:

• SQA should not report to the project manager;
• SQA should report somewhere within the local company or plant organization;
• There should be no more than one management position between SQA and the senior location manager;
• SQA should always have a “dotted-line” relationship to a senior corporate executive;
• SQA should report to someone having a vested interest in software quality, like the staff head responsible for field services.
SQA CONSIDERATIONS

- SQA organizations are rarely staffed with sufficiently experienced or knowledgable people because such people usually prefer development/design work, and management often wants them in the latter, too;
- The SQA management team often is not capable of negotiating with development. This depends on the caliber of the SQA team;
- Senior management often backs development over SQA on a very large percentage of issues. Development then ignores the SQA issues, and SQA degenerates into a series of low-level, useless debates.
• Many SQA organizations operate without suitably documented and approved development standards and procedures; without such standards, they do not have a sound basis for judging developmental work, and every issue becomes a matter of opinion. Development also wins such generalized debates when schedules are tight.

• Software development groups rarely produce verifiable quality plans. SQA is then trapped into arguments over specific defects rather than overall quality indicators. SQA may win the battle but lose the war.
SQA PEOPLE
Getting good people into SQA can be a problem. Possible solutions include putting new hires there (but must also have experienced people there, too), rotating personnel through SQA (which may result in only poor developers being assigned there), and requiring that all new development managers be promoted from SQA after spending at least 6 months there (which can be very effective).

INDEPENDENT VERIFICATION AND VALIDATION
In DoD contracts, independent verification and validation (IV&V) is often specified. The IV&V organization provides an independent assessment of the quality of the software. However, do not confuse SQA and IV&V. SQA works for the developer; IV&V works for the customer.