Module 13

Software Reliability and Quality Management

Lesson 33

Statistical Testing and Software Quality Management

Specific Instructional Objectives

At the end of this lesson the student would be able to:

- Identify the primary objective of statistical testing.
- Define what is meant by the operation profile of a software product.
- Identify the steps in which statistical testing is performed on a software product.
- Identify the advantages and disadvantages of statistical testing.
- Identify the main quality factors of a software product.
- Explain what is meant by quality management system.
- Identify the phases over which quality management system has evolved in the last century.

Statistical testing

Statistical testing is a testing process whose objective is to determine the reliability of software products rather than discovering errors. Test cases are designed for statistical testing with an entirely different objective than those of conventional testing.

Operation profile

Different categories of users may use a software for different purposes. For example, a Librarian might use the library automation software to create member records, add books to the library, etc. whereas a library member might use to software to query about the availability of the book, or to issue and return books. Formally, the operation profile of a software can be defined as the probability distribution of the input of an average user. If the input to a number of classes {Ci} is divided, the probability value of a class represent the probability of an average user selecting his next input from this class. Thus, the operation profile assigns a probability value Pi to each input class Ci.

Steps in statistical testing

Statistical testing allows one to concentrate on testing those parts of the system that are most likely to be used. The first step of statistical testing is to determine the operation profile of the software. The next step is to generate a set of test data corresponding to the determined operation profile. The third step is to apply the test cases to the software and record the time between each failure. After a statistically significant number of failures have been observed, the reliability can be computed.

Advantages and disadvantages of statistical testing

Statistical testing allows one to concentrate on testing parts of the system that are most likely to be used. Therefore, it results in a system that the users to be more reliable (than actually it is!). Reliability estimation using statistical testing is more accurate compared to those of other methods such as ROCOF, POFOD etc. But it is not easy to perform statistical testing properly. There is no simple and repeatable way of defining operation profiles. Also it is very much cumbersome to generate test cases for statistical testing cause the number of test cases with which the system is to be tested should be statistically significant.

Software Quality

Traditionally, a quality product is defined in terms of its fitness of purpose. That is, a quality product does exactly what the users want it to do. For software products, fitness of purpose is usually interpreted in terms of satisfaction of the requirements laid down in the SRS document. Although “fitness of purpose” is a satisfactory definition of quality for many products such as a car, a table fan, a grinding machine, etc. – for software products, “fitness of purpose” is not a wholly satisfactory definition of quality. To give an example, consider a software product that is functionally correct. That is, it performs all functions as specified in the SRS document. But, has an almost unusable user interface. Even though it may be functionally correct, we cannot consider it to be a quality product. Another example may be that of a product which does everything that the users want but has an almost incomprehensible and unmaintainable code. Therefore, the traditional concept of quality as “fitness of purpose” for software products is not wholly satisfactory.

The modern view of a quality associates with a software product several quality factors such as the following:

- **Portability:** A software product is said to be portable, if it can be easily made to work in different operating system environments, in different machines, with other software products, etc.

- **Usability:** A software product has good usability, if different categories of users (i.e. both expert and novice users) can easily invoke the functions of the product.

- **Reusability:** A software product has good reusability, if different modules of the product can easily be reused to develop new products.

- **Correctness:** A software product is correct, if different requirements as specified in the SRS document have been correctly implemented.

- **Maintainability:** A software product is maintainable, if errors can be easily corrected as and when they show up, new functions can be easily added to the product, and the functionalities of the product can be easily modified, etc.

Software quality management system

A quality management system (often referred to as quality system) is the principal methodology used by organizations to ensure that the products they develop have the desired quality. A quality system consists of the following:

- **Managerial Structure and Individual Responsibilities.** A quality system is actually the responsibility of the organization as a whole. However, every organization has a separate quality department to perform several quality system activities. The quality system of an organization should have support of the top management. Without support for the quality system at a high level in a company, few members of staff will take the quality system seriously.

- **Quality System Activities.** The quality system activities encompass the following:
  - auditing of projects
  - review of the quality system
  - development of standards, procedures, and guidelines, etc.
  - production of reports for the top management summarizing the effectiveness of the quality system in the organization.

Evolution of quality management system

Quality systems have rapidly evolved over the last five decades. Prior to World War II, the usual method to produce quality products was to inspect the finished products to eliminate defective products. Since that time, quality systems of organizations have undergone through four stages of evolution as shown in the fig. 13.4. The initial product inspection method gave way to quality control (QC). Quality control focuses not only on detecting the defective products and eliminating them but also on determining the causes behind the defects. Thus, quality control aims at correcting the causes of errors and not just rejecting the products. The next breakthrough in quality systems was the development of quality assurance principles.

The basic premise of modern quality assurance is that if an organization's processes are good and are followed rigorously, then the products are bound to be of good quality. The modern quality paradigm includes guidance for recognizing, defining, analyzing, and improving the production process. Total quality management (TQM) advocates that the process followed by an organization must be continuously improved through process measurements.

TQM goes a step further than quality assurance and aims at continuous process improvement. TQM goes beyond documenting processes to optimizing them through redesign. A term related to TQM is Business Process Reengineering (BPR). BPR aims at reengineering the way business is carried out in an organization. From the above discussion it can be stated that over the years the quality paradigm has shifted from product assurance to process assurance (as shown in fig. 13.4).

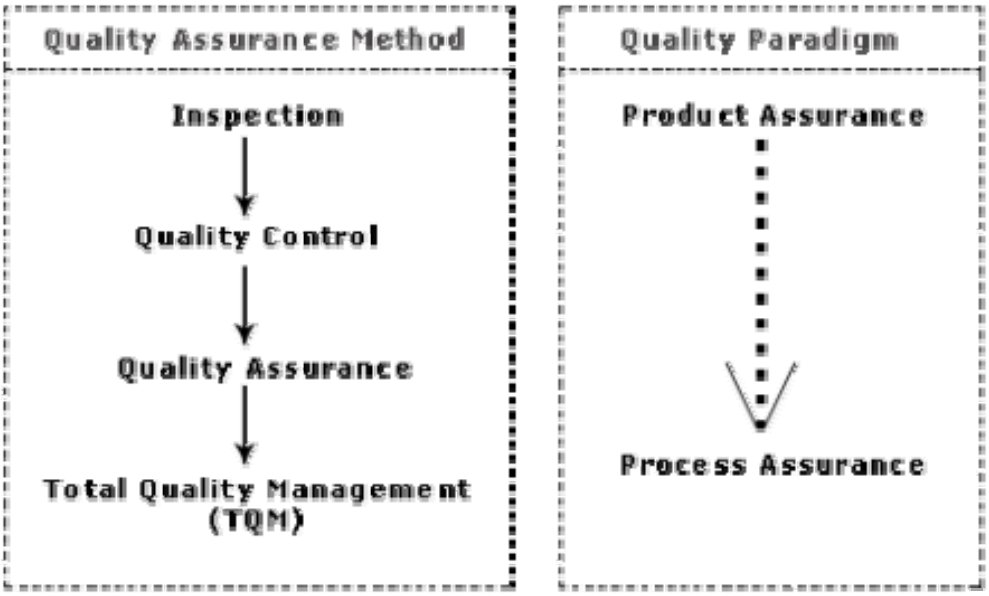

Fig. 13.4: Evolution of quality system and corresponding shift in the quality paradigm