A software quality plan for higher education—An abstract

by BARRY L. BATEMAN and CHADWICK H. NESTMAN
Southern Illinois University
Carbondale, Illinois

As the number of institutions of postsecondary education facing financial exigency increase, a method of tracking the progress of software projects in a systematic and unbiased manner is extremely critical. The software quality plan commonly found in business and industry can aid the college and university in developing guidelines and tests for the software development process, thus reducing costs.

Traditionally, testing and evaluation of software has been a minor or non-existent part of the programming effort within a university. If testing and evaluating does exist, it has normally been carried out by the programmer responsible for project development or by his close associate. This could mean that objectivity and selectivity in testing and evaluating can be lost because of a programmer's proximity to his work. This could also mean that more of the operating budget of the computing center has to be spent on testing and debugging after installation has been carried out. Testing and debugging in the user's environment can cause unnecessary problems in public relations and system acceptance.

Some industrial companies, such as TRW, Inc. and the NCR Corporation, have recognized this problem and have established formal Software Quality Assurance (SQA) departments or interest groups whose responsibilities include the task of assuring that software is free from as many errors as possible. In carrying out their functions, the SQA organization utilizes a formal plan or guideline known as the Software Quality Plan (SQP). The SQP is a document prepared by each of the participating internal departments within computer services (programming, operation, training, production, etc.) and the customer or client. The plan, once prepared, spans the entire development cycle and becomes a coordinating agent between all interested groups in the establishment of activities, schedules, commitments and budgets.

Within the institution of postsecondary education, responsibility for the development of the SQA should be under the direct supervision of the Director of Software Quality Assurance or his designate. In institutions where there is no formal quality assurance office, the plan could be produced by the internal auditor or at least by the chief officer in charge of computing services. Although the SQP is initiated early in the development cycle, it must be updated as the project moves from one development phase to another. When writing a SQP, the following criteria must be considered: financial and budgetary analysis, manpower requirements (by departments), program dependencies (hardware and software), milestone schedules, and commitments by the intended user in relation to their support and acceptance criteria. Test plans and responsibilities should clearly be outlined including the control procedure, methodology, limitations, and acceptance criteria for each test to be used. It is suggested that there be at least a review or test during the feasibility stage, in order to check architectural proposals; the coding stage, in order to assure proper design; pre-installation, in order to exercise programs with near real data; installation to insure users are adequately trained and that proper documentation is available. Resource requirements, in the sense of stating needed hardware, software, special test tools, additional manpower and required documentation should also be provided in the SQP.

In essence, software quality within the university framework is a necessity, especially in light of financial exigency. This quality can be attained by better control during the project development process and by assuring test objectivity by having the evaluation of software carried out by non-partisan people—people not involved with the actual writing and coding of the software. These non-partisan persons would be part of a Software Quality Department and would be responsible for the implementation of a Software Quality Plan, that is, a guideline for the monitoring of software quality throughout the development process. Above all else the plan would describe the various tests to be conducted, responsibilities in carrying out those tests, and the results expected. This plan is not intended as a panacea for all postsecondary quality ills, but is intended as a starting point for the assuring of quality software in the university environment by reducing costs normally incurred when errors are corrected on the "fly."