On the use of “software maintenance”

Software maintenance refers to the act of maintaining computer programs after they have been delivered to the customer or user (which may be a person or a system in which the programs are embedded).

In the January issue of Computer (“Replacing the term ‘software maintenance,’” p. 88), John A. Grosberg states, “the term software maintenance does not accurately denote the activities of that phase [of the software life cycle], and what’s worse, the use of the term fosters a mistaken concept of those activities. There really is no such activity as software maintenance and hence a different term should be used to identify that phase.” He goes on to show that there is no analogy between hardware maintenance, which “maintains” a design, and software maintenance, which “changes” the design. Finally, he proposes that we replace the term “software maintenance” with “continued development” and “debugging.”

According to Webster’s New Twentieth Century Dictionary, the word “maintenance” means “a maintaining.” Maintain is defined as “to keep up, to continue with, to carry on”; “to keep in existence or continuance”; and to keep in a certain condition or position, especially of efficiency.” Therefore, “maintenance” does not mean to simply keep something in its original condition, but to keep it in a condition that allows it to continue performing a useful and efficient function.

E. B. Swanson in his article, “The Dimensions of Maintenance,” (Proceedings of the Second International Conference on Software Engineering, IEEE, Oct. 1976, pp. 492-497) defined three types of software maintenance activities that may be required to keep the product in a useful state:

1. **Corrective maintenance** is concerned with the diagnosing and correcting of errors missed during design and development testing.
2. **Adaptive maintenance** performs the modifications necessary to allow the software to operate in a changed hardware environment.
3. **Perfective maintenance** includes adding new capabilities, user-requested functions, and other improvements that help keep the software useful over time.

Thus, we can see that the term “software maintenance” accurately describes the activities performed during this phase of the software life cycle. I propose that “software maintenance engineers” should become standard nomenclature for those who perform the vital function of correcting, modifying, and maintaining the thousands of existing computer programs.

Greg DeWild
665 Fairwood #276
Clearwater, FL 33759

**Author’s reply**

The difference I see between the position outlined in my article and that given by G. DeWild is that I say we are not “maintaining” software, while he says we are maintaining software’s usefulness. These statements are not incompatible.

To maintain a system (i.e., keep some aspect of it constant, such as speed or efficiency) when the environment/system relationship changes, we have to change a subsystem. Thus, to keep our car running properly, we replace parts. This is called “maintaining our car.” To keep making profits, the car manufacturer modified the car’s design to meet changing tastes. The manufacturer calls this “maintaining profits,” not “maintaining cars.”

True, we are maintaining the usefulness of software, but that is not the same as maintaining the software itself. Usefulness can be maintained only by changing the software. Thus, we may without self-contradiction talk about “software usefulness maintenance” or “software revision to maintain usefulness.” But we can’t without contradiction say “software maintenance to maintain usefulness” because we can’t maintain one without changing the other.

The closest analogy I can think of to the whole “software maintenance” issue is in the publishing industry. I’ll let the readers think about that, and maybe they will agree with me when I suggest that (1) the word “revision” be used instead of “maintenance” for software, and the categories of software revision activities be renamed “corrective revision,” “adaptive revision,” and “perfective revision”; and (2) the word “maintenance” be saved for applications that are not stretching its meaning to the breaking point.

John Grosberg
5842 N. 86th St.
Scottsdale, AZ 85253

**NJIT**

**Chairperson: Computer and Information Science**

The CIS chairperson leads a rapidly expanding department with the largest program in NJ. Additional faculty positions are anticipated within the next five years, several at tenure rank. Thus, the chairperson will have immediate effect on department development.

NJIT is the public technological university of New Jersey. There are 7000 students enrolled in baccalaureate through doctoral programs within three colleges: Newark College of Engineering, the School of Architecture, and Third College, the school of sciences, humanities and management.

AA/EO employer
Qualifications: Earned doctorate; documented teaching research, administrative and leadership experience.
Send resume or nominations by 12/10/84.

**Personnel Box CIS**

New Jersey Institute of Technology
Newark, NJ 07102

November 1984