



Instead of the Wrecking Ball

Rudy Alder
Publisher



Software development is in trouble. A 1990 study of software acquisition in one large U.S. federal government organization showed that only 1.5 percent of major software projects were used as delivered, 3 percent were used after modification, 19 percent were used but later abandoned within two years, 29 percent of the software was never delivered to contract, and 47.5 percent was delivered but never used.¹ In commercial industry, the average software project overshoots its schedule by 50 percent, 33 percent of the projects are canceled and of those that are not, 75 percent are declared operational failures.² By all rights, the entire industry should be condemned as unsafe, bulldozed over, and rebuilt from scratch.

Of course that would be impractical. Whatever reforms are made have to originate from within the current framework. Acquisition reform is one such effort to improve those abysmal statistics. Unfortunately, the way acquisition reforms are being implemented usually falls far short of the intended purpose.

The problem is twofold: First, recent work-force reductions leave many government programs few options but to outsource its acquisition functions to contractors. Although many contractors

possess superior acquisition skills, fewer government people for oversight functions means a greater probability of miscommunication and subsequent project failure.

Second, the government has decades of experience in acquiring hardware but only a few years in the acquisition of software. When acquiring a tank, the government merely draws up the specifications and waits for the finished product to roll off the assembly line. It is then relatively easy to test the tank's capabilities against the specifications: It either shoots straight or it does not.

But hardware acquisition cannot be used as a model for software acquisition because software is fundamentally different from hardware; software is distilled human intelligence, a collection of abstract ideas buried in the bowels of silicon memory. It makes no contact with human senses except through a hardware interface. This inherent intangibility means that greater care must be used to specify, regulate, and test the product. Unlike the hardware acquisition process, the user must be involved in every step of software's creation as it occurs.

"Black-box" testing at the end of the process will not suffice—a "white-box" review of the software during development also is essential to yield a better understanding of what is being created.

With fewer government employees to oversee the development of the desired product, prospects for meeting user requirements dim considerably.

Acquisition reform to this point has not been based on a recognition of these essential differences between software and hardware; until it is, most reforms will serve only to further perpetuate the poor performance of most software development projects.

Until that situation is remedied, those who wish to improve their acquisition capabilities will have to rely on other fixes. The Software Technology Support Center is equipped to assist projects in various aspects of acquisition management such as managing expectations, defining processes, or just-in-time skills. Our consultants can assess your current status and suggest improvements that could save your project from the wrecking ball. Contact us by phone, fax, E-mail, or surface mail; the addresses and numbers are on the inside back cover and in the center insert. ♦

Notes

1. *Systems Testing and Quality Assurance Techniques: Fundamentals*, Vol. 1, Ver. 4.1S, January 1997, p.VI-34.
2. *Guidelines for Successful Acquisition and Management of Software-Intensive Systems*, Software Technology Support Center, Hill Air Force Base, Utah, Ver. 2.0, June 1996, p.1-10.



Have Trainees Verify Uses for New Skills

I will disregard the common perception that management is merely giving lip service to training, particularly in these days of severe budget restraints, but I have some comments on Paula Shafer's article, "Planning an Effective Training Program," *March 1998*.

With respect to "management support" and motivation, I recall seeing a training film here on base (Hill Air Force Base) years ago in which a female clerk was sent to several training programs to upgrade her skills. After she completed

the programs, in the same breath that her boss congratulated her, he asked her to make another pot of coffee, i.e., business as usual. Then he was surprised when she wanted to move to another job. I am waiting to see people being moved to jobs where their acquired skills can be more fully used. Now, that would be motivation.

Shafer gave several valuable suggestions under "Theory vs. Practice" with respect to using the skills acquired. I suggest another: A patented training

program of the Boy Scouts of America, "Woodbadge," uses the unique (in my experience) device of having the trainees specify in writing how they will use the skills learned "on the job" over the next six-month to two-year period, and commit to doing it. Successful completion of the course is withheld until trainees submit a written report detailing their experiences in using those skills.

Tim Layton
*Operational Flight Program Support,
Avionics Software Test Section*