



Training and Education



One of my favorite resources is Webster's 1828 Dictionary. Unlike any of our modern dictionaries, the 1828 version provides a deeper understanding of words. I thought the 1828 definition of education was quite fitting for this month's theme. Webster's says that education comprehends all that series of instruction and discipline which is intended to enlighten understanding ... and fit them for usefulness in their future stations.

A primary question this issue attempts to highlight is *do our educational institutions in fact prepare software professionals to be successful in their future stations?* In October 2006, a study conducted by the Center for Strategic and International Studies revealed a significant shortfall in the number of software professionals that had been formally educated in software project management. The study indicated that our industry is lacking in program managers, software architects, systems engineers, and domain experts. Many industry experts agree that there is an adequate supply of programmers, but the pool of these critical senior managers and system experts is very limited.

As Department of Defense (DoD) systems become more and more software intensive, software developments get bigger, more complicated, and more dependent on senior software professionals to get the project on the right path and keep it there. Since studies seem to indicate that we are falling behind in our attempt to educate certain pockets of critical expertise and our software projects are in greater need of these same professionals, our problem is getting worse. I hope this month's articles will inspire you to think about how you will address this trend in your organization.

For a more thorough discussion of this trend from the perspective of aerospace engineering, I hope you will read *The Critical Need for Software Engineering Education* by Dr. Lyle N. Long. You will find more specific software educational issues addressed in the two articles that follow. In *Using Inspections to Teach Requirements Validation* by Lulu He, Dr. Jeffrey C. Carver, and Dr. Rayford B. Vaughn, the authors share an experiment that compares the use of both checklists and Perspective-Based Reading to aid in a requirements validation exercise. In *Integrating Software Assurance Knowledge Into Conventional Curricula* by Dr. Nancy R. Mead, Dr. Dan Shoemaker, and Jeffrey A. Ingalsbe, the authors compare the contents of the Common Body of Knowledge for Secure Software Assurance with the Computing Curricula 2005: The Overview Report. Their intent is to map our security needs with software-related curricula being recommended for our schools.

One of the options available to assist DoD readers with required software training is the software curriculums available through the Air Force Institute of Technology (AFIT); you can read more about AFIT in *Software Engineering Continuing Education at a Price You Can Afford* by Maj Christopher Bohn, Ph.D. You will find another thought-provoking article from Roger Stewart and Lew Priven with their discussion of successfully implementing software inspections in *How to Avoid Software Inspection Failure and Achieve Ongoing Quality, Cost, and Schedule Benefits*. We conclude this month's issue with an insightful article by Dr. Robert B.K. Dewar and Dr. Edmond Schonberg entitled *Computer Science Education: Where Are the Software Engineers of Tomorrow?*

It should be clear throughout the software community that education does not end once a graduate has a bachelor's degree. The technology we work with continues to grow at an astounding rate. Even if additional degrees are not sought, all professionals need additional education to keep current with the needs of the software community. As a co-sponsor for CROSSTALK, I am happy to provide one additional source for this education.

Kevin Stamey
Oklahoma City Air Logistics Center, Co-Sponsor