



## Requirements Management Is Required



In 1994, my organization was striving for Capability Maturity Model® (CMM®) Levels 2 and 3. Driving this activity, the Air Force required us to be Level 3 by 1998 or risk losing the ability to compete for workload. While working toward this goal, one of our major challenges was requirements management. Not yet in charge of the 309th, but a middle manager, I attended a group-level meeting where we discussed our struggles with requirements. At that meeting, one of our technical program managers stated that our organization needed to commit to follow a written policy of managing requirements. He discussed the reasons our projects experienced requirement changes, such as nebulous requests from the user, inadequate designs, and in-progress customer requests. The effects of these requirement changes were significant schedule overruns and dissatisfied customers. He mentioned the traps we got ourselves into like accepting work with inadequate guidance and being strong-armed by some customers to deliver very minor or unclear modifications, and taking on high-risk work without proper negotiations or replanning. When this happened, the end result was the customer and our organization both felt cheated because these requirement changes were not documented and plans were not adjusted appropriately.

One of the ironies of this situation is that we actually did have processes for proper requirements management in place, but we simply did not follow them! We had to make significant cultural changes to adequately implement those practices. Remember, the driving forces behind this issue were good ones; we wanted to support our customers and provide all that we could for the warfighter using our products. We had to convince ourselves that in attempting to give everything to everyone, we were actually hurting both the end-user and ourselves. In addition, because of our previous *do anything* approach, we had to re-educate our customers on the need to better manage requirements effectively; we had to be upfront with them on how requirements changes could effect the cost, schedule, and quality of our software products. Perhaps you can see some of your own challenges in our past struggles. The good news is that we did implement the recommendations of that technical program manager, long ago. Now, as a CMM Integration (CMMI®) Level 5 organization, we use our requirements management practices to ensure any requirements changes are approved by both our management and the customer and that our project plans reflect these changes. The end result is that our customers are intimately involved in our software projects. While requirements issues will always remain challenging, both we and our customers are now fully aware of these impacts; the result – we not only have very low schedule variances, but we have satisfied customers.

The articles in this issue of *CROSSTALK* aim to help the readers with their own requirements challenges. In our first article, *Twelve Requirements Basics for Project Success*, Dr. Ralph Young shares insights he has gathered from his own experience as well as reading about the experience of others. Deb Jacobs follows these basics with specific advice on understanding your requirements in *Interpreting Requirements in a He Said/She Said World*. Next, Dr. Nancy Mead gets even more specific as she discusses different requirements elicitation methods in *Experiences in Eliciting Security Requirements*. Our final theme article, *Requirements as Enablers for Software Assurance*, discusses work that Dr. Seok-Won Lee and Robin A. Gandhi have done to consolidate software security requirements from several guidance documents in the Department of Defense in order to identify the applicable set of security requirements necessary for certification and accreditation. Our supporting article by Dr. Jeffrey Carver, Dr. Forrest Shull, and Dr. Ioana Rus suggests performing requirements and design inspections from varying perspectives in *Finding and Fixing Problems Early: A Perspective-Based Approach to Requirements and Design Inspections*.

While all of the practices of the CMMI, AS9100, and ISO 9000 play a role in our current success with software delivery, requirements management was one of the earliest practices that showed visible results. Good requirements engineering requires more than managing existing requirements; it requires effectively eliciting the requirements, verifying requirements are accurate and useful, managing those requirements, and testing the end-product against the requirements. I anticipate the articles in this issue will provide useful insights for both the novice and expert alike.

*Randy B. Hill*

Randy B. Hill  
Co-Sponsor, Ogden Air Logistics Center