

CROSSTALK would like to thank **76 SMXG** for sponsoring this issue.



The Agile umbrella of software thought formally found its way into the discussions of the 76th Software Maintenance Group at Tinker Air Force Base nine years ago. Agile introduced a new level of consternation to our management team with its new terminology and a perception that the foundations of disciplined management practices were being threatened. Like many across the software development community, a second perception developed among the software engineers of our organization that the CMMI and Agile frameworks were at odds. It wasn't long until I begin to hear that "CMMI is too heavy a burden for this project, so we'll use Agile instead," or "Agile is too undisciplined, so we'll use CMMI instead." While both of these criticisms can be true about poor implementations of Agile or CMMI, neither characterizes the ways that CMMI or Agile were intended.

I am very pleased to say that after years of learning from ourselves and the larger software community, that Agile and CMMI are not mutually exclusive and that when used in a complimentary way can result in improved processes and products achieving world class results. Our organization's Agile path started as a grassroots effort when one of our projects discovered the Agile Scrum methodology, and approached management about the possibility of adopting it. Our organization's Engineering Process Group (EPG) invested heavily with our pilot teams and designed a new process set that embraced iterative software builds, increased customer collaboration, and responsiveness to requirements volatility. Organizational projects piloted these processes, while the EPG consulted with them to insure that all of the best practices identified in CMMI were still being performed. As the pilots wrapped up, the EPG took the work of the project and built a new lifecycle model based around Agile Scrum. That model is now available for any project in the organization to adopt and use.

One thing that I can say unequivocally is that CMMI and Agile can be used together successfully. CMMI provides the ideal framework for managing and continuously improving our organization's processes. Agile Scrum provides a new lifecycle development model that yields a better fit for many of our projects over the traditional waterfall model, driving greater customer interaction and employee investment. Both frameworks have provided value to the organization when we embraced them intelligently. When the best practices from both models are thoughtfully applied to the right project, it is possible to do CMMI the Agile Way.

I hope that this issue of **CROSSTALK** continues to inform and assist a software community that continues to adapt to address the ever growing complexity of systems. Our project teams are being asked every day to achieve a greater level of user experience with greater capability in a rapidly changing environment at a lower cost. These types of objectives will not be achieved without a continued directed focus on collaboration on improvement. I would like to encourage all of our community partners to redouble our focus on building relationships focused on learning, sharing, and improving.

Michael Jennings
Group Director, 76 Software Maintenance Group