Process Maturity Pays Off in Many Ways

Daniel J. Wynn
Chief, Software Engineering Division, Ogden Air Logistics Center

When I accepted division chief responsibilities for the Software Engineering Division (TIS) at the Ogden Air Logistics Center, Hill Air Force Base, Utah, the process improvement initiative had already begun. There was not a great deal of cost benefit data available at the time to document a business case for the Capability Maturity Model (CMM) style of process improvement, but there was considerable management commitment and momentum. I made a conscious decision to trust the judgment of the TIS management team and my predecessor by choosing to continue this effort. However, improvements to the way we were proceeding needed to be made.

The first improvement was to implement CMM process improvement as a planned project. Improvement is not without cost. It takes additional time and resources. It requires planning, execution, and tracking. The Software Engineering Process Group role was refined by giving them the task to create an implementation plan in cooperation with the management team. Management met on a regular schedule to review progress, refine the plan, and apply the necessary talent to move forward. As we defined our plans and measured our progress, there was an increase in our speed and ability to achieve the goal.

The second improvement was to the Quality Engineering Support Team (QuEST). They functioned in a staff role to TIS, independent of managers and projects; however, they were focused on verifying the quality of products. Because their role duplicated the existing testing functions, they were not achieving the desired results. The QuEST role was therefore redefined to verify the application of our defined processes. This not only improved the quality of our products, it enforced the applications of our processes. We were forced to make processes that worked because we knew we would be judged by them. In addition, it reinforced the organization's commitment to process improvement. In retrospect, it appears this is an essential ingredient to success because other government organizations that did not have this function and that were ahead of us in process improvement fell away from their initial commitment. Continuous self assessment is essential to process improvement.

Now that TIS has been assessed at Level 5, I have noted a change in morale. There is a greater level of confidence and employee satisfaction—a sense of accomplishment and an understanding that government employees can be and are some of the best software engineers available. Now the data has been collected to show a business case for CMM process improvement. Our customers enjoy a cost benefit with greater predictability and higher quality. I want to see continued senior leadership support for the kinds of improvements we have made. It was a good call on their part, and we have the data to validate their decision. ◆

About This Issue

This *CROSSTALK* special issue addresses two of our most highly requested article topics: the Capability Maturity Model (CMM) and process improvement lessons learned. I extend a special thank you to the Ogden Air Logistics Center, Software Engineering Division (TIS) for sharing its lessons learned and words of advice in this month's issue.

TIS is the first government organization to achieve a Level 5 CMM rating. With over 500 employees, 420 of whom are dedicated to software development and sustainment, TIS is the largest software organization to achieve the Level 5 rating. TIS develops flight programs and automatic test equipment for aircraft and weapons systems such as the F-16 Fighting Falcon, the B-52 Stratofortress, the B-1 Lancer, and the Minuteman missile. TIS is also the parent organization of the Software Technology Support Center, which publishes *CROSSTALK*.

Tracy Stauder
Managing Editor

DEPARTMENT OF THE AIR FORCE
HEADQUARTERS OGDEN AIR LOGISTICS CENTER (AFMC)
HILL AIR FORCE BASE, UTAH

10 December 1998

Major General Richard H. Roellig
Commander
7981 Georgia Street
Hill AFB UT 84056-5824

Lieutenant Colonel Richard P. Cashman
HQ ACC/DR-SMO-P
204 Dodd Blvd, Suite 226
Langley AFB VA 23665

Dear Colonel Cashman

I want to share some very good news with our customers. Our Software Engineering Division, OO-ALC/TIS, recently completed a two-week independent assessment of its software engineering processes. On a scale of 1 to 5, with 5 being the highest, the Software Engineering Division received a 5. The rating was determined in accordance with the Software Capability Maturity Model developed by the Software Engineering Institute (SEI) at Carnegie Mellon University. The five levels of maturity, defined in the Capability Maturity Model, are: (1) initial, (2) repeatable, (3) defined, (4) managed, and (5) optimizing.

The SEI is a Federally Funded Research and Development Center (FFRDC). It was organized in the mid 80s to help the Department of Defense address recurring weapon system software problems related to significant cost and schedule overruns and poor quality. The Software Capability Maturity Model and its associated assessment tool have been extensively used by software development organizations since 1987. Well over 700 organizations have been assessed to date.

The Level 5 assessment is a prestigious recognition. TIS is the first known government agency, federal or state, to achieve this level. Only four other software organizations are known to have achieved this rating, and TIS is by far the largest and most diverse of the four. Approximately 420 personnel within TIS are assigned duties of developing, maintaining, and managing software projects. TIS product lines include development and maintenance of aircraft operational flight programs, mission planning systems, and automatic test equipment.

What achieving Level 5 means to you, our customers, is better cost and schedule predictability and higher quality in the end product. It means OO-ALC/TIS is quite capable of meeting its customer commitments because it thoroughly understands what it is capable of doing. It means program managers have good visibility into the status of their programs.

As stated earlier, this achievement is exceptionally good, not only for OO-ALC, but the Air Force and the Department of Defense. We are proud our hard work has paid off. If you would like further information on the ramifications of this accomplishment, please contact Mr. Dan Wynn, TIS Division Chief, at (801)-777-2615.

Sincerely

Richard H. Roellig

RICHARD H. ROELLIG, Major General, USAF
Commander

The Capability Maturity Model: A Summary

Continuous process improvement is based on many small, evolutionary steps rather than revolutionary innovations. The Capability Maturity Model® (CMM®) provides a framework for organizing these evolutionary steps into five maturity levels that lay successive foundations for continuous process improvement. These five maturity levels define an ordinal scale for measuring the maturity of an organization's software process and for evaluating its software process capability. They also help an organization prioritize its improvement efforts.

A *maturity level* is a well-defined evolutionary plateau toward achieving a mature software process. Each maturity level comprises a set of process goals that, when satisfied, stabilize an important component of the process. Achieving each level of maturity framework establishes a different component in the software process, resulting in an increase in the process capability of the organization.

Organizing the CMM into the five levels shown in Figure 1 prioritizes improvement actions for increasing software process maturity. … The five levels can be briefly described as

Level	Focus	Key Process Areas
1 Initial	Heroics	
2 Repeatable	Project Management	Software Project Planning Software Project Planning and Oversight Software Subcontract Management Software Quality Assurance Software Configuration Management Requirements Management
3 Defined	Engineering Process	Organization Process Focus Organization Process Definition Peer Reviews Training Program Intergroup Coordination Software Product Engineering Integrated Software Management
4 Managed	Product and Process Quality	Software Quality Management Quantitative Process Management
5 Optimizing	Continuous Improvement	Process Change Management Technology Change Management Defect Prevention

Figure 1. *The five levels of the CMM and their key process areas.*

1. Initial

The software process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort and heroics.

2. Repeatable

Basic project management processes are established to track cost, schedule, and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.

3. Defined

The software process for both management and engineering activities is documented, standardized, and integrated into a standard software process for the organization. All projects use an approved, tailored version of the organization's standard software process for developing and maintaining software.

4. Managed

Detailed measures of the software process and product quality are collected. Both the software process and products are quantitatively understood and controlled.

5. Optimizing

Continuous process improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.

These five levels reflect the fact that the CMM is a model for improving the capability of software organizations. The priorities in the CMM, as expressed by these levels, are not directed at individual projects. A troubled project might well prioritize its problems differently from the taxonomy given by the CMM. Its solutions might be of limited value to the rest of the organization, because other projects might have different problems or be unable to take advantage of its solutions because they lack the necessary foundation to implement the solutions. The CMM focuses on processes that are of value across the organization.

Extracted from Paulk, Mark C., et al., The Capability Maturity Model: Guidelines for Improving the Software Process, *Version 1.1, Addison-Wesley, Reading, Mass., pp. 15-17.*

The Journey to CMM Level 5: A Time Line

Patrick W. Cosgriff
Ogden Air Logistics Center, Software Engineering Division

The answer to the question, "How long does it take?" a brief time line of our journey to Level 5, and a few tips that may help you get there sooner.

As I discuss process improvement with people, I am often asked how long it takes to achieve Capability Maturity Model (CMM) Level 5. My mind usually drifts back to when we were Level 1 or 2, when I asked someone a similar question and was given a philosophical answer like, "Well, process improvement isn't really a destination—it's a journey. You shouldn't look at it as trying to get a level. …"

Forget philosophical answers, because I have an answer: It takes approximately 7.559 years to go from CMM Level 1 to Level 5. I know this because we formally began process improvement in 1991, and we achieved Level 5 July 23, 1998. Of course, this assumes you have good senior management sponsorship, you have many process improvement champions that happen to be in the right place at the right time, and you have customers who are supportive of process improvement efforts. This also assumes you think you are different and that the CMM does not apply to you, you have an abundance of skeptics at all levels of your organization, you think you are much too busy to do process improvement, and you think the legacy systems you are forced to use supposedly do not support CMM-type measurement. If 7.559 years is too long, you always have the option of stopping production so that you can work on your processes full time.

Using Appraisal Feedback to Guide Early Process Improvement

In 1991, TIS began its CMM-based process improvement initiative (see Figure 1). Some projects had been doing some process improvement in an ad hoc way, but this was the beginning of our structured organization-wide process improvement efforts. We formed a Software Engineering Process Group (SEPG) and began process definition at the project level.

In May 1992, we were formally assessed using the Software Process Assessment (SPA) method. We were rated an emerging Level 2, which was a gentle way to say we were Level 1. The main Level 2 weaknesses identified were in project planning, project tracking and oversight, and software quality assurance. The assessment team noted that some areas had already institutionalized some of the key practices required for the organization to attain a Level 3. Encouraged by these results, we developed an action plan based on the findings and began to implement it. The focus of the action plan was to implement and institutionalize the Level 2 and Level 3 key process areas (KPAs).

In September 1994, we were assessed as a solid Level 2 organization. We were close to being a Level 3, with weaknesses concentrated in the training, peer review, and integrated software management KPAs.

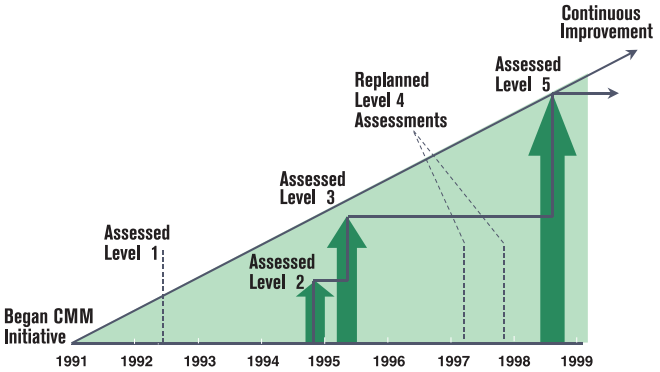


Figure 1. *TIS process improvement time line.*

Because there were only a few Level 3 weaknesses found in the 1994 assessment and because they were concentrated in just three KPAs, TIS sought the support of the Software Engineering Institute (SEI) to hold a Delta Appraisal and focus on the three KPAs found to be deficient. SEI agreed, but the KPAs would be rated in their entirety, and the assessment had to be held within six months of the previous assessment.

In March 1995, a Delta Appraisal was conducted. All of the weaknesses from the 1994 assessment were sufficiently addressed, and we were rated as CMM Level 3.

On Our Own for Implementing Levels 4 and 5 Processes

At this point, we thought we could again develop an action plan based on the assessment findings and go to work. However, there were not many findings on which to work. Although we had always owned our own process improvement planning process, we had always based our action plans on recommendations from an assessment team. Because Levels 4 and 5 were not in the scope of the assessments to this point, Levels 4 and 5 findings and recommendations had never been developed for us by the assessment teams.

Fortunately, our Level 3 processes had now put us in a much better position to assess ourselves and to map out appropriate strategies. The planning started with our senior management developing a new strategic plan outlining the goals for our organization for the next two to five years, including achieving Level 4 by 1997. From this and the assessment findings, the SEPG developed an action plan to further institutionalize Levels 2 and 3 practices and to implement Level 4 practices.

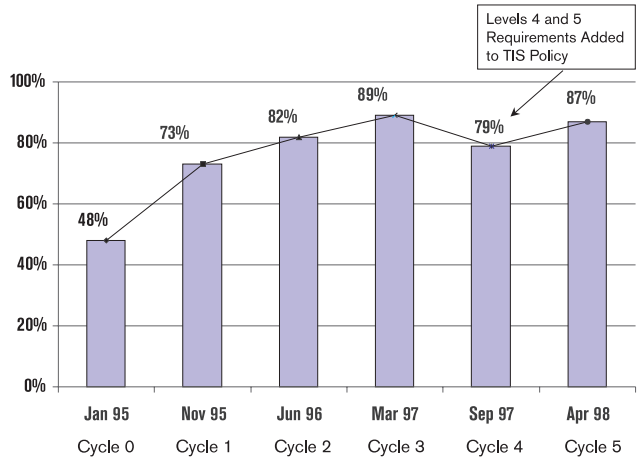


Figure 2. *TIS policy compliance.*

The Power of QuEST Feedback

We also were armed with a powerful new tool that proved invaluable for implementing consistent organizational process improvement: The Software Quality Assurance group, called the Quality Engineering and Support Team (QuEST). (For a detailed description of QuEST duties, see "Software Quality Assurance in a CMM Level 5 Organization," page 11 of this issue).

QuEST members report directly to the division chief—an independence from the projects that has proved invaluable. QuEST audited the projects in the organization against an extensive set of requirements taken directly from organizational policy (see Figure 2). The audits were objective and quantitative; they included detailed descriptions of the findings and recommendations for addressing non-compliance issues.

As an organization, we could analyze the QuEST data on a question-by-question basis to expose bad or unclear policy statements. The organization's willingness to update the organizational policy based on feedback fostered buy-in from all levels of the organization. People felt as though we were improving and listening to their concerns, which made them more willing to contribute.

Empowered with QuEST data, we were able to closely monitor our Level 4 implementation efforts and react quickly to unreasonable or poorly conceived plans—and we certainly had some. After all, we were in uncharted waters; there was not much practical, proven experience available on how to implement Level 4 practices.

We replanned our efforts twice, based on QuEST data and on our increasing understanding of the Levels 4 and 5 activities. We originally had planned to be reassessed in the spring of 1997. As we approached this milestone, our QuEST data indicated we would not achieve Level 4 by this date, so we replanned to assess in November 1997. As the SEPG analyzed the activities in Level 5, it was clear to us that if we could achieve Level 4 we could achieve Level 5. During the summer of 1997, the TIS management team discussed the advantages and disadvantages of delaying the assessment and going for Level 5. Our process action teams had already developed the processes for the Level 5 KPAs while working on the Level 4 KPAs, which left only the task of implementing and institutionalizing the processes.

In September 1997, at the end of our fifth QuEST cycle, our data indicated we would not successfully achieve CMM Level 5 by November 1997. We were struggling with measurement, data gathering, and data consistency issues—not with what to do with valid data. We replanned again, postponing the assessment until the summer of 1998. Instead of performing a formal assessment in November 1997, we contracted with two Software Technology Support Center assessors to perform a Snapshot assessment (a much less rigorous and less expensive assessment) of our organization to determine our weaknesses from the perspective of an outside assessor. We then used this input and our QuEST data to develop a new action plan for the final nine months. This plan addressed Levels 4 and 5 implementation issues as well as assessment preparation activities.

In July 1998, we were assessed again. We were rated a CMM Level 5 by a highly experienced assessment team.

Lessons Learned

To gauge how long it will take your organization to implement Level 5 practices, look at your unique circumstances. It will largely depend on the culture of your organization, the senior management sponsorship, your motivation, your expectations, and the resources available to apply to improvement. And the bottom line is, you need good people; processes do not improve processes—people improve processes.

Can the dramatic changes be done in significantly less time than we took? I cannot answer that, but I know from experience that it takes time to change the culture of an entire organization. However, I believe others can achieve Level 5 maturity with less pain than we experienced. There is much more training available now, more conference presentations on the higher maturity practices, and more off-the-shelf tools available. Most important, more and more organizations are now reaching the higher maturity levels. This provides an experience base from which to draw practical proven practices. We were helped tremendously by listening to the lessons learned at Boeing and IBM.

Following are a few suggestions that might speed up your journey.

- Understand the practices one level above the implementation level you are currently striving to obtain. Give some thought to how the practices interrelate and build off each other. This may save some rework in the long-run.
- Enforcement and implementation are basically the same, especially in large organizations. In other words, enforcement is the most effective implementation strategy. In my experience, objective audits done by capable, well-trained people and a clear set of audit requirements is the most effective enforcement and implementation strategy.

see TIME LINE, page 30

Benefits Realized from Climbing the CMM Ladder

Leon G. Oldham, David B. Putman, Mark Peterson, Bruce Rudd, Kevin Tjoland
Ogden Air Logistics Center, Software Engineering Division

This article discusses the benefits the Ogden Air Logistics Center (ALC) Software Engineering Division (TIS) reaped from its process improvement efforts. Those benefits are examined from two aspects: the qualitative or the effects on the developer's quality of life and the quantitative or the effects on the organization's development capability measured in cost, schedule, and product quality.

Why is the comic strip Dilbert© so popular? It makes fun of

- Managers committing to unrealistic costs and schedules.
- Projects being over cost and schedule.
- Poorly managed projects.
- Poorly defined requirements and requirements changing just before delivery.
- Poor quality.

Perhaps Dilbert's popularity lies in its proximity to the truth. Employees complain that management is clueless, whereas management complains that the employees do not give them enough information to manage the project better. But both sides are doing the best they can with the resources and information they have. The solution to end the feuding is simple: a common enemy they can fight together, side by side, and the enemy should be poor processes. Because each side wants to maintain their own ways and wants the other side to change, the proponents of process improvement will have to convince both sides that the process improvement effort should be seen as a friend and not as an enemy. This article will demonstrate the positive aspects of process improvement according to the Software Engineering Institute (SEI) Capability Maturity Model (CMM) for Software.

The original title of this article was "The Return on Investment from Climbing the CMM Ladder." The term *return on investment*, however, has a precise definition within the business community that requires specific knowledge of cause and effect regarding changes to processes or methods and the accompanying improvements in cost and productivity. Unfortunately, it is only when an organization reaches CMM Level 4 that the employees understand their processes in quantitative terms and can tie specific actions to process capability changes. Although it can be shown that tremendous improvements have been made in the TIS process capability in both the quality of software produced and the cost to produce that software, to correlate each change made over the years to specific quantitative improvements in process productivity or product quality is impossible. Instead, we can show general relationships and overall improvement across the years. A contributing factor to the improvement is the experience gained by the practitioners. This contribution was considered small because most of the core practitioners already had several years experience when significant process improvement began.

We will investigate the improvements in the Ogden ALC software development capability on two fronts. The first will be in qualitative terms, which means the quality of life of the practitioners, changes to their working environment, and general project expectations. The second approach will be in quantitative terms. Although these figures will be exact, we estimate their accuracy to be within 20 percent. Even with this uncertainty, we will show that the savings realized by the Air Force are worth the investment made.

This article is concerned with the overall investment in process improvement and the returns and benefits realized within the two software developmental product lines. In fact, our experience has been that quantitative gains within the automatic test product line have been difficult, if not impossible, to substantiate. The quantitative portion of this article, therefore, will reflect the savings gained in the Operational Flight Program (OFP) and mission planning product line. The quality of life and schedule issues, however, will reflect gains across the division as a whole.

Qualitative Benefits

Practitioner Working Environment

A brief questionnaire was sent to those employees who had been in the organization for the duration of the process improvement effort and who had a long-term perspective on the changes wrought by these efforts. Of 32 questionnaires sent, 18 were returned—a good number for voluntary participation. The questions and responses are summarized as follows:

- *Have you been more constrained or less constrained in performing your job?*

Ten of the respondents felt more constrained, four saw no difference, and four felt less constrained. Of those who felt more constrained, about half saw it as an inevitable side effect of providing beneficial structure to the development process. The constraint was not considered to be negative.

- *Is it easier to perform your duties with respect to tools, working environment, etc.?*

Thirteen of the respondents felt it was much easier, two felt it was a little easier, two about the same, and one said it was a little harder. The one who felt it was harder pointed to more complex and less

Figure 1. *Defect density by update.*

user-friendly tools. Of the positive responses, most attributed the improvements to tools and technology, e.g., paperless environment, and some also cited the benefits of better planning and coordination.

- *Are there more project surprises or fewer?*

Thirteen felt there were fewer surprises, four saw no difference, and one felt there were more.

- *Do you now feel that you have more input and control into project planning or less?*

Twelve felt they have more input into project planning, two felt they have a little more, two felt they have the same, and two felt they have less.

- *Do you feel that our CMM efforts have been a positive influence?*

The answer was a unanimous "yes."

- *Do you feel that you are producing better quality software?*

Sixteen felt that the quality of software produced had improved. Two felt that it was always good and had not changed.

Project Execution

The ability to control requirements changes, remove defects earlier, and consequently perform better planning and project control has significantly reduced the "fire drill" atmosphere typical of earlier projects. This is especially true of the end of the project cycle when last-minute changes without schedule relief and defects found in final testing wreaked havoc with delivery schedules. The resulting overtime and unhappy customers combined to make life more than a little unpleasant.

Overall Effect

The working environment and culture within the organization has changed significantly over the years. There are still last-minute glitches and surprises, but they are the exception, not the rule. The engineers do not see much difference in the way they do their work. The constraint on creativity many feared has not materialized. In fact, most still say they dislike process improvement and have not seen many changes. That is because the CMM is, for the most part, a management model. Most changes have been in the way we have managed our projects, not in how the engineers actually do design work. Changes have been slow in materializing, but the resultant change in culture is remarkable.

Quantitative Benefits

Quality Improvements

All errors are costly in one way or another, even though some might not believe that quality as measured in conformance to requirements specifications is important. Loss of market share due to customer dissatisfaction or just the increased cost of bringing the product to market have definite financial impacts on the software supplier. Although the latter may be more immediately visible, the former may be the long-term cause of organizational demise.

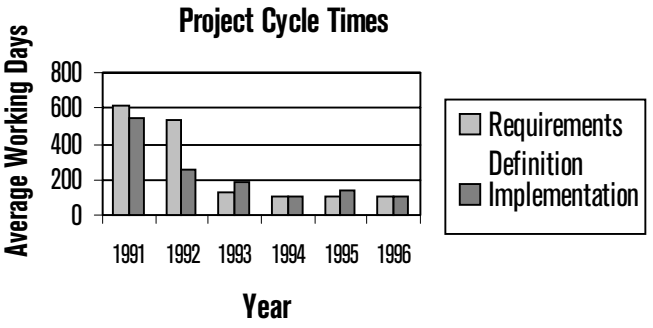
The quality of software delivered to our OFP customers over the years has improved dramatically. In showing this improvement, we chose the metric *defects per thousand source lines of code* (KSLOC) reported after delivery of the production tape. This measure was chosen to compare later projects with earlier projects. As the process improved, our metrics data changed over the years. This measure was available for previous updates. Our defect ratio (Quality Deficiency Reports [QDRs] generated against production deliveries divided by the size of the update in KSLOC) was not as useful as our current metrics. As part of our Level 5 improvement implementations, we now use measurements that show quality at each phase of the project to isolate and remove sources of errors. The defect ratio of production tapes over the years is shown in Figure 1. As can be seen, the quality of product at the point of measure has improved steadily over time. It is now a rarity to receive a QDR on a production tape. Projects D, E, G, and H have had no QDRs submitted. The two QDRs represented by the spike in data at Update F were found by our internal code inspections and testing being done for a later update.

Schedule and Cycle Time

In the early 1990s, the automatic test equipment (ATE) product line employees focused their efforts on reducing the cycle time. Our assumptions were that if we reduced our cycle time, we would reduce the costs of the projects. This assumption is not necessarily true in every case, but fortunately for us, the assumption appears to have been valid.

The average ATE project cycle times, shown in Figure 2, are the average number of days from the authorization to start work to the delivery of the product. We began our software process improvement (SPI) efforts in 1991 and achieved a

Figure 2. *Average ATE cycle time.*



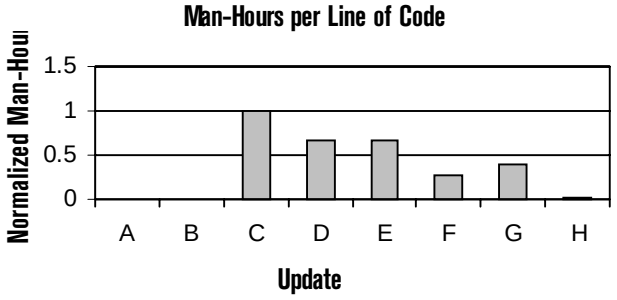


Figure 3. *Man-hours per line of code (normalized).*

CMM Level 3 in 1995. The purpose of Figure 2 is to point out that the improvements need not wait until you are a Level 5 organization. The known inefficiencies were corrected as quickly as possible. These cycle times were reduced by approximately 70 percent each while we worked on the CMM Level 2 and Level 3 issues. The projects are tracked separately as requirements definition (RD) projects, e.g., RD 1, RD 2, or projects to implement the approved enhancements. In this product line, an individual project can reference either new requirements definition or implementation to any one of more than 750 automatic test programs.

Although the cycle time displayed in Figure 2 was only a portion of the lengthy overall response time experienced by the end users of the products, an intangible benefit from the reduced cycle times is greater customer satisfaction. In 1995, the customer joined us in an enterprise-wide action team called the Falcon Software Express (FSE). The FSE was co-chaired by a member from TIS and the customer's lead program manager. The FSE team applied the same high-level process improvement concepts to the overall process, which crossed numerous organizational boundaries. The organizations and people affected by FSE included software engineering, program managers, equipment specialists, item managers, and funding managers. The FSE team achieved a similar reduction of approximately 70 percent for the overall cycle time experienced by the end user.

Schedule Variance

Mark Paulk, et al., stated, "An unpublished review of 17 major Department of Defense (DoD) software contracts found that the average 28-month schedule was missed by 20 months. One four-year project took seven years; no project was on time." [1] The average schedule variance for the 17 DoD contracts studied showed that, on the average, each project took 70 percent more time than scheduled. In comparison, our average schedule variance is less than 5 percent.

Productivity Improvements and Cost Reduction

Although product quality may be important and at least highly desirable, productivity and cost per unit of production are the immediate measures that management uses to determine the payback for investment in process improvement.

Savings in our OFP product line is shown in Figure 3, which shows the normalized cost per line of code based on lines of code produced and man-hours required for each update. (Note that these updates correspond to those shown in Figure 1.) Values for projects earlier than those shown were not available. Those projects with extremely low cost per KSLOC benefited from heavy reuse. Early in the program, our OFP system design engineers learned that they needed to work closely with the pilots to assure that conceptual ideas were understood and defined properly in the system's requirements document. Rapid prototyping and technical interface meetings were established to help assure that the products developed met both the system requirements and the needs of the end users.

Our ATE product line provides a level-of-effort type of support that makes the savings, on a per-project basis, more difficult to solidify. When loaded at the optimum level, the ATE product line now produces the software updates at savings of approximately 70 percent; however, when the workload is at a level less than the optimum level, the cost per project rises. In an effort to stay at the optimum workload level, the ATE product line works closely with the customer to forecast the predicted workload and manpower needs.

Return on Investment

In an attempt to put a value on the return to the Air Force from the investment TIS made in process improvement, a few basic tenets were established. First, since this and most software maintenance organizations—including those in the private sector—provide essentially a level-of-effort service to the customer, savings were computed based on cost per unit of deliverable product multiplied by the number of units delivered per year, i.e., cost per line of code or cost per test program set times the number delivered per year. Second, based on general business practices, an investment in process improvement for any given year will be assumed to be responsible, in part, for actual and projected savings garnered in the following five years. Third, as previously stated, we assume that most savings realized resulted from the process improvements institutionalized through this program. With these conditions in mind, the estimated return on investment for this division was a ratio of about 19-to-1. In other words, the Air Force received, in the form of additional software enhancements to the F-16 aircraft weapons systems and other weapons systems, nearly 20 dollars for every dollar invested. To date, that is well in excess of $100 million worth of weapons and test system enhancements and fixes.

We realize that these figures seem unrealistically high. But, as stated before, they are based on investment vs. payback over time, including projected payback over the next five years. This is consistent with management accounting practices used to determine the advisability of making capital investments in process improvements. Further, it is doubtful that doubling our investment would have significantly increased productivity. Likewise, if we had invested the money in process improvement without the management commitment to ensure implementation, our return would have been extremely low. In fact, the money would have been wasted.

We were fortunate to have struck the right balance of resources to move improvement along without waste and yet preserve enough momentum for the organizational culture to undergo the desired change.

Conclusion

Our end users are the ultimate beneficiaries of our SPI activities. The end users are receiving higher-quality products that perform as envisioned at a lower cost and with minimal project cost and schedule variances. At the same time, most practitioners believe their working conditions have improved or at least have not become worse, whereas management believes that they have better control of the situation.

Remember, the analysis and implementation of process improvement requires patience and time; it does not produce instant feedback. Both qualitative and quantitative metrics show a continual improvement over the years. Although it is difficult to show a one-to-one correspondence to each improvement with the benefits shown in the metrics, it is easy to show continual improvement.

Finally, to quote one of TIS's first-line managers who has several years experience in project management, "We have only been at a CMM Level 5 for a short time. Now that we have the tools in place to really understand our processes, real improvements can now begin." ◆

About the Authors

Leon G. Oldham, president of L.G. Oldham and Associates, is a consultant who supports the Software Technology Support Center (STSC). He has nearly 28 years experience in the software development field, five years experience in ATE, 20 years in OFP development, more than two years as a Software Engineering Process Group (SEPG) lead, and one year as a process improvement consultant. During his tenure in the OFP development field, he was a developer for eight years, technical program manager for three years, and an engineering supervisor for nine years. He has a bachelor's degree in electrical engineering from the University of Utah.

Voice: 801-777-4774 DSN 777-4774
E-mail: oldhaml@software.hill.af.mil

David B. Putman is the lead of the SEPG for TIS at Hill Air Force Base. He has more than 17 years experience in ATE, two years with Hughes Aircraft, and more than 15 years with the Air Force. During his involvement with ATE, he was the senior engineer within the Avionics Software Support Branch for nine years, and he supervised ATE engineering teams for two years. He also supervised the F-16 OFP system design and integration test teams for one year. He has a bachelor's degree in electrical engineering from the University of Utah and a master's degree in business administration from Utah State University.

Voice: 801-777-4726 DSN 777-4726
E-mail: putmand@software.hill.af.mil

Mark Peterson is a technical program manager for F-16 Block 30 OFP software development for TIS at Hill Air Force Base. He has 13 years experience in OFP development, including four years in radar software design and six years in software system-level design. For the past two years, he has been a member of the extended SEPG and was involved in the implementation of the CMM Level 4 and Level 5 key process areas for OFP development. He has a bachelor's degree in electrical engineering from South Dakota State University.

Voice: 801-777-5664 DSN 777-5664
E-mail: petersma@software.hill.af.mil

Bruce Rudd is chief of the F-16 OFP System Design and System Test section in TIS at Hill Air Force Base. He has more than 16 years experience in software development. He was technical program manager for F-16 software updates at Hill Air Force Base for seven years, lead engineer for one of the F-16 subsystem software development teams for five years, and developer for two years. He has a bachelor's degree in electrical engineering from Utah State University.

Voice: 801-775-2192 DSN 775-2192
E-mail: ruddb@software.hill.af.mil

Kevin Tjoland is a technical program manager for F-16 Block 30 OFP software development for TIS at Hill Air Force Base. He has 14 years experience in OFP development, including five years in avionics software design and six years in software system-level design. For the past two years, he has been a member of the extended SEPG and was involved in the implementation of the CMM Level 4 and Level 5 key process areas for OFP development. He has a bachelor's degree in electrical engineering from South Dakota State University.

Voice: 801-777-5701 DSN 777-5701
E-mail: tjolandk@software.hill.af.mil

Reference

1. Paulk, Mark, et al., *The Capability Maturity Model: Guidelines for Improving the Software Process*, Addison-Wesley, Reading, Mass., 1994.

Software Quality Assurance in a CMM Level 5 Organization

Rushby Craig
Ogden Air Logistics Center, Software Engineering Division

This article shares the evolution of the Ogden Air Logistics Center, Software Engineering Division's philosophy about quality assurance (QA), its perspective on applying the principles of QA from the CMM, and how the QA group has been used as a tool for process maturity measurement and to improve the organization.

Many people think that software quality assurance (SQA) is a mystifying or a dull subject. I have had many experiences that have brought this to my attention. The most frequently recurring one is when friends or acquaintances ask what I do for a living. Usually, I respond by saying that I am a software engineer. This response results in two types of reactions. The first type is, "Wow, that's interesting. So, how about this weather we've been having?" The second is more rare and a little more difficult to answer: "Does that mean you write software?" Surprised by their interest, I respond, "Well, actually I used to write software, but now I'm involved in doing quality assurance work." At this point, a bit of explanation seems appropriate, given the slightly puzzled look, so I add, "That means that I make sure that software projects are doing all of the right things in order to build quality products." The reaction to this also is predictable: "Cool. So, did you see the game on TV last night?" Here, in the Software Engineering Division (TIS), the attitude is a bit more favorable. QA is not perceived as interesting or fun by the typical engineer in TIS, but most have come to understand what QA is and also appreciate its contribution, within the framework of process improvement, to improve our quality, efficiency, and morale as an organization.

But process improvement initiatives were not always appreciated. This attitude metamorphosis has happened gradually over several years. Initially, the efforts in SQA were viewed as dimly by the software "cowboys" as the coming of a sheriff to the old Wild West. But after having a taste of civilization, these same cowboys found out that the law can help and protect you—that it brings about a common good. In fact, looking back, living in the Wild West had not been as great as the cowboys had once thought. The greatest part of the resistance to process improvement was eventually overcome, and a CMM Level 5 rating was recently achieved largely because of the unwavering commitment of division management to improve the quality of our products and services. SQA was an important tool that management used to continuously measure what we were doing right and wrong and where we stood with respect to our goals to improve as a function of time. It also helped to bring areas that were problems to the attention of all management levels so that attention was kept on them long enough to ensure their correction.

SQA Defined

According to the CMM, the purpose of SQA is "to provide management with appropriate visibility into the process being used by the software project and of the products being built." [1] Thus, the use of the word "visibility" implies that the SQA group is meant to be the eyes of management on what is going on in the organization. This visibility is not meant to punish perceived offenders but rather to give management data to help them to make bigger-picture decisions and corrections in the organization.

As you read articles on SQA, you will find that most people use the term differently from how the CMM defines it. The CMM defines SQA as "reviewing and auditing products and activities to verify that they comply with procedures and standards. …" [1] *Software products* are "the complete set, or any of the individual items of the set, of computer programs, procedures, and associated documentation and data designated for delivery to a customer or end user." [2] Meanwhile, *activities* are "any steps taken or functions performed, both mental and physical, toward achieving some objective. Activities include all of the work the managers and technical staff do to perform the tasks of the project and organization." [3] Activities are the things the process says we will do to complete a project and deliver a product to the customer. So, auditing activities is auditing the process and the adherence to the process by the project. The CMM's definition of SQA is a bit narrower than common usage, which includes testing and peer reviews. Because of this, there is almost always some confusion when using the term. The CMM also accounts for testing and peer reviews, but in different places. Peer reviews are given a separate key process area (KPA), and testing is addressed by the software product engineering KPA.

The reason the KPAs are organized this way is probably because of Total Quality Management (TQM) principles, which are the roots of the CMM. As you may recall (if you experienced TQM training in your organization), one of the principles of TQM is that everyone is responsible for quality. Quality is not something you put in after you build the product but is rather a result of the way it is produced. Consequently, it makes more sense in that context for testing and peer reviews to be considered part of the engineering function as opposed to the SQA function. This way of thinking

also fit the perspective of TIS because the Air Force had encouraged the adoption of TQM principles before the CMM emerged on the scene.

The key point here is that SQA is defined as auditing two things: products and process. This was important for us because it influenced the way TIS chose to implement SQA.

Product Quality

When the CMM was adopted as the model TIS would use for process improvement, we already had a group in place that was doing many of the things that fall under product quality auditing. The configuration management (CM) group in the division had for some time been performing tasks such as reviewing software products for compliance with format and ensuring that documentation was in order for the design, testing, reviews, acceptance, and configuration of software products. As time has passed, the role of the CM group has expanded in some cases (as the projects have requested it), giving them something like a watchdog role to ensure that activities are performed before a process block can be exited. As an example, in some projects, the CM person schedules and attends peer reviews, takes minutes, and ensures that all issues are resolved before the process proceeds from that point. The group also verifies that activities are properly completed before projects pass from one process block to another. In some software maintenance projects, the CM person takes a highly active role in attending and witnessing acceptance tests held with the customer. The CM person assures that the tests are held, that tests are performed within the bounds of the established ground rules, that the tests are documented, and that all of the paperwork is properly filled out, correct, and signed off by both parties. All these activities assure quality in the software products.

The CM group is independent from the projects in the management chain to avoid conflicts of interest. CM employees can raise issues of noncompliance as high in the management chain as is necessary to resolve issues that arise.

Process Quality

One of the premises upon which the CMM is based is that "the quality of a software system is highly influenced by the quality of the process used to develop and maintain it." [4] Ensuring the quality of the processes in TIS is the job of the Quality Engineering Support Team (QuEST).

QuEST works directly for the TIS division chief. This is important because it creates independence from the projects and their management. Independence is as important in process QA as it is in product QA to avoid conflict of interest. Second, it gives QuEST members the authority needed to perform audits with minimum difficulty. Although many practitioners are sold on following the process and require no prodding, some need to have a reminder to keep focused on doing things in the established way. Without the authority of upper management, it would be easy for project personnel (who are so inclined) to be uncooperative or to not take audits or the results seriously. Enough practitioners would probably be overcome by the irresistible temptation to head back to the Wild West that the process would be inconsistently followed, and the organization would lose its process improvement momentum or perhaps even turn backward.

The QuEST group has historically varied from two to four people. They are software engineers who are selected from projects in the division and are selected to balance the experience on the team between the product lines that are represented in the division. Service in QuEST is voluntary. Management only considers candidates for these positions who have demonstrated a high level of competence in projects where they have worked. Additionally, personnel are rotated so that the term of service is between 18 and 30 months. These are basically the same requirements and ground rules for individuals chosen to serve in the Software Engineering Process Group (SEPG), with whom QuEST works closely. In some cases, positions in the SEPG are filled with employees who have experience in QuEST.

There are some good reasons behind the selection and length-of-service guidelines for QuEST. First, balancing the experience helps the team to have a broader point of view when reviewing projects. Since a QuEST member performing an audit may not have worked in and be familiar with the process specifics of a particular product line, it is helpful to have someone with more experience in that area on the team. Second, for the team to have credibility with projects teams they review, it is important that they have the respect of the team members. Having employees on the QuEST team who have a good reputation as a practitioner also gives the audit process more credibility. Third, the rotation of employees through QuEST (and the SEPG) was calculated as a way to produce employees who are highly trained in process improvement and have an intimate knowledge of projects and their processes throughout the division. When they have finished their rotation, they return to a project where they can be a highly valuable resource, and it is hoped, a champion for process improvement. These individuals also can help projects to improve their processes by sharing information and experience gained from seeing how others in the division and others in industry do things.

Managing QA

One of the things we believe has contributed to the success of the QuEST group is the decision to manage the group and its activities as a project. Back in the early days, when the division was struggling to implement things such as documented processes, peer reviews, etc., QuEST was often asked if we were practicing what we preached. It turned out that we usually were not. Consequently, it was difficult for QuEST to explain how to do a required activity or to give suggestions when we had not done them ourselves. Changes were made in that area, and now QuEST operates using Level 5 principles tailored somewhat for what we are doing. Some examples follow:

- Requirements Management – Determining through discussions with

division and branch managers which software projects and support groups should be audited during the upcoming cycle.

- Documented Plans and Processes – Written plans that describe how QuEST will perform all required activities, and processes and work breakdown structures that describe the sequence of events as process blocks and the corresponding entry and exit criteria for each, along with the required tasks and methods used to internally verify that the tasks and exit criteria are satisfied.
- Schedules – A schedule is created for a QuEST audit cycle as well as for activities performed between them. Progress is tracked as time goes on, and metrics are kept on schedule variance.
- Training – Training plans for QuEST are created, tracked, and signed off as the elements are completed.
- Peer Reviews – Held to review audit report findings, recommendations, and statistics. Defects found are tabulated and tracked by process block.
- Intergroup Coordination – Done with the SEPG, CM, etc.
- Quality Assurance – QuEST is audited by the SEPG to verify compliance with policy, plans, and processes.
- Project Tracking and Oversight – Regular coordination meetings are held in the QuEST group with all members present. Also, management reviews are held every two weeks with the division chief, where the QuEST schedule, issues, metrics and their trends, action items, goals, etc., are reviewed and discussed.
- Quality Process Management and Product Quality Management – Quantitative baselines are established for effort, schedule, and quality. Current performance is calculated and tracked, with corrective actions taken as thresholds are exceeded.
- Defect Prevention and Process Change Management – At the end of each audit cycle, QuEST reviews lessons learned, audit question score averages, customer satisfaction survey statistics, and project metrics to see if our plans, procedures, or questions asked need to be revised. Common root causes are sought for problems that are identified. Changes are made as required.

Now, when somebody asks us, maybe a little sarcastically, if QuEST does all of these things that we look for in our audits, we can honestly say, "Yes." Then, they wish they had not asked as we explain (probably in more detail than they want) how the team operates. We take pride in the way we do business.

QuEST Audits

As the CMM requires, TIS has organizational-level documents that are the boundaries within which software projects must operate. Following is a brief synopsis of these documents.

- *TIS Strategic Plan* – Documents vision, mission, and values of the organization as well as goals and objectives and an action plan to achieve them.
- *TIS Policy for Engineering Development and Support Project Management* – The governing policy for project execution and the management of projects. Defines roles, responsibilities, and requirements in the division in order to meet the objectives of each KPA in the CMM. The purpose of the TIS policy is to help the division execute the Strategic Plan.
- *TIS Standard Engineering Process* – General process that the TIS policy requires to be used as the framework from which to build the process for a project.
- *TIS Metrics Implementation Guide* – Designated by the TIS policy as the document that contains the standard formulas to be used for metrics, which are kept and reported to upper management. This helps to maintain consistency of metrics across the organization and thus makes quantitative management easier.

QuEST audits software projects and support functions in the division to verify that they meet the requirements of the TIS policy. The process to execute audits has the following steps.

- In-Briefing (if required) – Explain to new projects the audit process, why we are doing it, and what to expect.
- Preparation – Review past audit reports.
- Project Managers Interview – Get responses to questions (approximately 100) on standardized questionnaire. Get visual proof of items where possible (see Figure 1).
- Initial Report Write-Up – Make rough draft of report.
- Verify Practitioner Interviews – Verify that practitioners' views agree with project managers' views. Answer their questions about why policy requires certain things. Ask in confidence about their concerns, problems, etc., they wish to share with the division.
- Draft Report Write-Up – Finish draft of report and give quantitative scores on questions. Roll these up to scores by KPA, by CMM level, and overall.
- Peer Review – QuEST and SEPG review findings, debate them, and make changes as necessary by consensus.
- Review with Project Manager – Draft report is sent to the project manager who has five days to bring up and resolve issues with QuEST.
- Out-Briefing – Results of the audit are briefed to project employees and management.
- Send Report – The final report is sent to the project manager and other management.
- Action Plan – An action plan is submitted to QuEST within 30 days after the out-briefing. QuEST reviews the action plan to verify it resolves issues brought up in the report. If it does not, issues are resolved by management at the lowest level possible but the highest necessary.

Results of QuEST Audits

TIS has found the results of QuEST audits to be extremely useful to measure where we stand as an organization, as well as where projects stand, in relation-

ship to the KPAs and CMM levels. It is true that some perspective is needed by the organization to avoid the trap of getting a CMM level rating. The goal should be to improve. But there are some good reasons to measure this way. One is that the CBA-IPI is set up as all or nothing with respect to its levels. This is, in a way, unfortunate because it does not tell you quantitatively how close or how far you are from compliance with the KPAs required for each level. Another problem is that such assessments are often done for the whole organization; therefore, the findings can be at too high a level to provide a basis for a detailed project action plan. So, issues that pertain to only one project may not make it to the assessment findings.

The other consideration is the substantial expense associated with CBA-IPIs. When planning for these formal assessments, it is a great bonus to know with some certainty where you stand. It would be disappointing to the organization to fail to achieve a certain level because of minor areas of noncompliance—especially when these issues could have been discovered and corrected before the money was spent.

For these reasons, QuEST uses a measurement method to quantify TIS policy compliance and hence CMM compliance. Questions on the questionnaire are scored on a 0-3 scale, where the scores represent different levels of compliance. The scoring correlates with possible answers as follows: No evidence of compliance or evidence of noncompliance = 0, some evidence of compliance or occasional compliance = 1, mostly satisfied (some evidence of noncompliance) = 2, and institutionalized compliance to policy = 3. A percentage of compliance is then calculated for each KPA, for each CMM level, and for overall compliance to TIS policy. Using these measurements, we can calculate where a project stands. A roll-up of the measurements from projects is used to measure where a product line and where the organization as a whole is, in terms of compliance. Because we have been doing this for several years, we can also show the trends for these measurements historically (see Figure 2).

These types of measurements have helped management in our organization manage process improvement efforts better because we know where to focus our SPI efforts.

In preparation for the July 1998 CBA-IPI, we had a great deal of data from prior QuEST reviews. We also participated in a mini- or Snapshot CBA-IPI with the SEPG and some employees from the Software Technology Support Center (STSC), where a report of strengths and weaknesses for each KPA was produced. When all was said and done, we found that the results of our QuEST audits, the miniassessment, and the CBA-IPI were consistent. In many ways, the QuEST data is better because these audits are more detailed than formal assessments, the measurements are taken more often, they cost less (because the organization must do SQA anyway), and they are much more quantitative than CBA-IPI results. This experience has given the organization an increased measure of confidence in the data produced by QuEST audits.

Another benefit that results from QuEST audits is the flow of information from practitioners up to the top of the management chain. Confidentiality is important for QuEST in order to create an atmosphere of nonretribution. But information about the concerns, gripes, praise, or suggestions of practitioners in general is often heard in the course of interviews. Practitioners' concerns are expressed to management in general terms to give them a feel for what is going on in the division.

Additionally, QuEST audit reports are not merely scores on questions. They detail why the project received the score, what verification was seen, and most important, what QuEST recommends they do to comply with the policy re-

Figure 1. *Examples of questions from the questionnaire.*

Project Planning		
Question	Answer	Rating
1. Are project commitments, i.e., plans, bids, quotes, and schedules approved by the appropriate branch chief? (A3.1.3) (LEVEL II)	Yes	3
2. Are workload estimate schedules derived from the WBS? (A3.1.4.) (LEVEL II)	Yes	3
3. Is a complete WBS prepared and consistent with the project's process? (A3.1.5.) (LEVEL II)	Yes, the WBS is documented on the SCF form.	3
4. Are cost-estimating procedures, assumptions, and parameters documented and stored for historical review? (A3.1.8.) (LEVEL II)	Yes, historical data is located in the SCF folders, which are online. The procedure is located in the MDPSEP beginning at Section 4.3.1.3, "Create an SCF Audit Form." Visually verified.	3
5. Have the required planning activities been performed, and have the plans been documented (Process, Schedule, Requirements, Acceptance Criteria, WBS, Quality, CM, Risk Analysis)? (A3.1.9.) (LEVEL II)	Yes, all information is located in the MDPSEP, MIP database, TIS data forms, MIP data spreadsheet, and SCF folders. Visually verified.	3

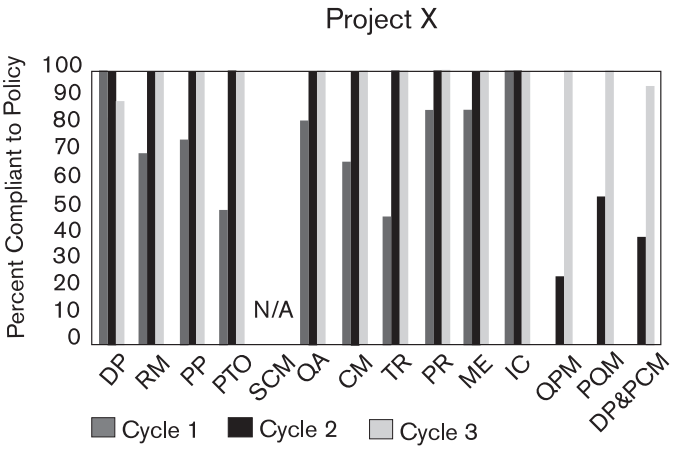


Figure 2. *Example of how to track project compliance to organizational policy over time. The KPAs listed on the X-axis are a slight modification of those in the CMM. QuEST combines some CMM KPAs into one. The percentages in the figure are examples. Subcontract management (SCM) does not apply to most projects in TIS.*

quirements. These recommendations often explain why the project needs to do certain things and why they are important. When projects are struggling in a specific area, suggestions are made to look at ways other particular groups are successfully implementing the activity in question. We believe it is important to be as helpful to the project as possible rather than merely be process police. When you are helpful to a project rather than a critic, it makes the job of SQA a lot easier. People are more receptive if they believe your recommendations are well thought out and are relative to their specific project.

Dealing with Resistance

As previously mentioned, the resistance to process improvement in our division was significant in the beginning of our CMM efforts. But the resistance has largely been overcome, because we now have a different culture—the way of doing business has changed, and part of that way of doing business is the acceptance of the idea that changing, to improve, is a virtue. Still, there are always some people who will only be comfortable with the status quo. Others accept the idea of change but still feel uncomfortable when they are going through it. Being Level 5 does not mean standing still, so there is always going to be some degree of resistance and discomfort. Change can be hard.

QuEST members earnestly try to adhere to the following strategies to minimize the effects of resistance.

- Do not take it personally. SQA people are magnets for jokes and high jinks, so have a sense of humor about it.
- Listen. Practitioners and project managers have lots of experience and knowledge.
- It is better to get a negative reaction than indifference. If someone has a negative attitude in an SQA audit, it may just be that they have not been sold on an idea yet. This is often an opportunity to show that person the bigger picture, which perhaps they have not seen or understood.
- Some people will never accept the idea of process improvement, so just deal with that reality.
- SQA's job is to measure, help, and encourage—not to expose and punish.

Is Level 5 Perfection?

So you might wonder if being in a Level 5 organization is the software development Utopia that some might imagine. I think it is safe to say that no one in TIS believes that our processes or our implementation of the CMM is perfect. There are many things that we believe should still be modified and improved. But the CMM does not say that a Level 5 organization is perfect. It merely means that you are doing the right things in the effort to be as good as possible at providing your customer with a quality product at the right time at the right price. SQA has proven to be a highly valuable tool in our organization for ensuring that we are striving to meet these goals. As for Utopia, I will know that we are there when the heating and air conditioning always work properly in my office. ◆

About the Author

Rushby Craig is a member of QuEST in TIS. Before joining QuEST, he worked with maintenance of ATE software and developed personal computer-based flight-line test systems. Prior to that, he was project manager for one of TIS's ATE software customers. He has a bachelor's degree from the University of Utah and a master's degree from the Air Force Institute of Technology, both in electrical engineering.

Quality Engineering Support Team
OO-ALC/TIS-3
7278 Fourth Street
Hill AFB, UT 84056-5205
Voice: 801-777-0541 DSN 777-0541
Fax: 801-775-4444
E-mail: craigr@software.hill.af.mil

References

1. Paulk, Mark C., et al., *The Capability Maturity Model: Guidelines for Improving the Software Process*, Addison-Wesley, Reading, Mass., 1995, Section 7.5, p. 171.
2. Paulk, Appendix C, p. 366.
3. Paulk, Appendix C, p. 353.
4. *Introduction to the Capability Maturity Model*, Course Charts, Software Engineering Institute, Carnegie Mellon University, 1996, p. 2-5.

The Right Things for the Right Reasons
Lessons Learned Achieving CMM Level 5

Patrick W. Cosgriff
Ogden Air Logistics Center, Software Engineering Division

This article discusses lessons we learned on our journey to implementing Capability Maturity Model (CMM) Level 5 processes at the Technology & Industrial Support Directorate (TIS), Software Engineering Division at Hill Air Force Base, Utah. Although ours is not the only way to implement the CMM, we have learned that the only way to use the CMM effectively is to do the right things for the right reasons, using the CMM as a guide. Do not limit solutions to the activities and subpractices listed in the CMM, which are general examples; specific situations require specific solutions. This article describes some of the unique processes we implemented as we strived to understand the underlying CMM principles and looked for solutions that make sense with respect to those principles.

As our organization prepared for our Level 5 assessment, there was no great fear of failure. We by no means thought we had it made, but we were confident we were doing the right things for the right reasons. We had identified areas of risk, most of which involved what the CMM calls "alternate practices." These are methods of satisfying the intent of the CMM that are different than the practices and subpractices listed in the CMM. We were unsure how these alternate practices would be received by an assessment team; however, we decided to do process improvement our way because it made sense and it worked. After all, our ultimate goal was and is to continually improve our performance, not to get a rating.

During the assessment, we were pleased that the highly experienced assessment team agreed that our alternate practices satisfied the intent of the CMM. This article discusses some important processes we have implemented at TIS. All are used because they work well and they make sense—the right things for the right reasons.

Management Sponsorship

You have probably heard again and again that senior management support is crucial to implementing the CMM—if something is not a priority to the people who control an organization's future, it will not be a priority to the organization's members. Management support alone is not enough, but without it, you cannot even get the troops to start.

We were fortunate to have solid management support at all levels, which was born from a desire to improve rather than a desire to win the "get a level" game. We would have done things differently if our only goal was to satisfy the CMM. Our strategic goals are to improve the quality of our products, improve our estimating accuracy, improve cost performance, improve schedule performance, and decrease cycle time. These goals are set quantitatively for the entire division by senior management. Product-line goals are kept consistent with division goals.

Packaging Project Plans

There are approximately 24 different plans referred or alluded to in the CMM. We did not believe this meant we should create 24 different planning documents. We found that when too many documents were required, people began to substitute documents for plans—they spent too much time developing documents that neither represented actual planning activities nor used to manage the projects. We learned to emphasize true planning activities more than documents. Our policy was changed to emphasize that projects must be planned, and the planning must be documented. It is a subtle difference, but by focusing on the activity more than the packaging, we put the emphasis on the important part.

Our current simplified view of a project plan answers five questions:

- What will we accomplish?
- How will we accomplish it?
- How much will it cost?
- How long will it take?
- How much deviation is acceptable?

Requirements must be documented and configured. The project must have processes, procedures, and the necessary support plans. The budget must be agreed to and configured. The project must have a schedule that is baselined and configured. Thresholds or control limits must be established based on the project's capability baseline.

These artifacts are in the form and location most convenient for the people who need to use them. A big binder full of material may be convenient for an assessment team, but we are in the business of developing and maintaining software—not in the business of being assessed. Usability is the key. A plan that is unusable becomes shelfware. Our organizational policy defines project-planning requirements, but the Extended SEPG (ESEPG) (see "Product Lines" section) individually decide the best way to package processes and plans.

Our product-line approach also simplifies project planning. By grouping similar projects together into product lines, many activities, such as configuration management, quality assurance, intergroup coordination, and

process documentation, were standardized. This greatly reduced the planning necessary for individual projects.

Operational Definitions

What is the difference between a process and a procedure? What is a defect? Having terms consistently defined across your organization is more important than the correctness of the definition. Operational definitions are critical, and the issue becomes more critical the larger the organization. Defining terminology is crucial to communicating ideas consistently throughout the organization. We needed to define terms such as process, procedure, software development plan, defect, requirements, capability baseline, threshold, and control limit. Before standardizing, even the term "project" had a different meaning from product line to product line.

Software Quality Assurance and Configuration Management

According to the software CMM, Version 1.1, "The purpose of Software Quality Assurance [SQA] is to provide management with appropriate visibility into the process being used by the software project and of the products being built." Intuitively, it would seem to make sense to lump both product and process quality assurance into one KPA; however, when we tried to implement this, it was clear that product quality assurance and process quality assurance were different activities. They require different skills, different training, and are done at different times. The assessment team considered our implementation of SQA to be an alternate practice.

The focus of our alternate practice was to break process quality assurance (QA) apart from product QA. We formed a team called the Quality Engineering Support Team (QuEST) (see "Software Quality Assurance in a CMM Level 5 Organization," page 11), whose primary responsibility was to perform process QA. The product QA duties were performed by our configuration management (CM) group. Both QuEST and CM are independent groups outside of the chain of command of the projects. This is essential to ensure cost or schedule problems do not cause quality to suffer. Their only responsibility is to assure quality products are being produced and quality processes are being followed. Project cost and schedule performance is not their concern.

Product Quality Assurance

To understand how the CM group performs the product QA role for us, you must understand how the CM group operates in our environment. Many people still think of CM as happening only at the beginning and at the end of a project. At OO-ALC/TIS, configuration specialists are involved throughout the process. They check in and check out baseline work products and developmental work products. They schedule and attend peer reviews, where they often act as recorder, taking minutes and recording anomalies. They ensure that the appropriate people have signed off on any anomalies that require further investigation or rework.

They are thought of as people like traffic officers who control the flow of work products between groups and phases in the project. In preparation for peer reviews, they review the work products for compliance to style guides, templates, checklists, etc. This leaves the technical people free to focus on the technical accuracy of the work product being reviewed instead of style-guide details.

Process Quality Assurance

Process QA is handled by our QuEST. QuEST is a three-person team that audits projects for compliance to division policy. They audit about 80 percent of the division every six to eight months. Originally, QuEST only reviewed software engineering projects. Now, they also audit non-software engineering projects and support groups. They also review the management chain of each branch, as well as the Executive Board, division chief, and division staff as a group, for compliance to division policy.

QuEST audits projects with a set of questions that are taken directly from our organizational policy. QuEST audits are performed by interviewing project personnel, including practitioners, lead engineers, and project managers. They produce a report that documents all noncompliance issues. The report is peer reviewed with the SEPG. The project manager gets a chance to review the draft report for inaccuracies. When the report is finished, the results are briefed to the project team and its management chain. The project's progress toward resolving the noncompliance issues is tracked in monthly project management reviews.

When the QuEST was first formed, members of projects looked upon it less than favorably. QuEST members were thought of as process police—people who were going to get them into trouble or cause them work. We believe this was limiting our process improvement progress. We needed to change people's attitudes toward the QuEST group and process improvement in general.

QuEST decided to emphasize service to the projects. As QuEST members were doing their reviews, they tried to view all potential noncompliance issues from the project's perspective. The QuEST worked as hard on understanding the problem and making a workable recommendation as they did on finding the problem. Since most of the early problems were common across the organization, consistent recommendations led to consistent implementation of policy across the entire organization.

Once the project members perceived the SQA group as doing something for them (instead of to them), their attitudes started to change, and so did the culture of the organization. Before our Level 5 assessment, we even had a group call the QuEST and complain that nobody from their functional area had been interviewed during the last project audit. This was a complete change from the earlier attitude.

Product Lines

Sometimes, the smallest adjustment can make the biggest difference; just changing the way you look at a problem can make the solution suddenly seem obvi-

ous. For years, we struggled while we tried to make everything from 100-person projects to one-person projects fit the CMM as it was written—verbatim. This was like creating one set of rules for both football and golf. Using one model for different types of projects was the wrong thing for the wrong reason.

When we began to implement the CMM, we wanted to compartmentalize our process definition and improvement by product line. However, because we knew little about the CMM and were concerned about levels and what assessment teams might think, we decided to try to make one size fit all. It was a safe move and probably the right one at the time. However, as we began to deal with higher-maturity level issues, we realized we had to align our process improvement efforts with the way work was being done. Our early Level 4 implementation efforts failed because we were trying to force different product lines to do things exactly the same way.

One of our first Level 4 breakthroughs occurred when a few members of the Software Engineering Process Group (SEPG) visited Boeing Space Trasportation Systems, which at the time had recently been assessed at CMM Level 5. Boeing shared the concept of the ESEPG. The concept is that those who use the process should be in charge of process improvement. The visit with Boeing, plus the fact that the latest draft of CMM, Version 2 addressed the product-line concept, gave us the confidence to reorganize our process improvement initiative along product lines.

Extended Software Engineering Process Group

The ESEPG is a group at the head of each product line that takes on some SEPG duties. The group consists of project managers, supervisors, and lead engineers who work part time on process improvement issues. They work closely with the organizational SEPG to ensure efforts of each product line are consistent with organizational policy. They also coordinate defect-prevention activities at the product-line level. They meet quarterly to review defect data and causal analysis action items.

Because they are connected to the work, the ESEPGs have unique insight into the better ways to implement organizational policy and prevent defects in their product line.

By allowing the ESEPGs to implement product-line-specific solutions consistent with organizational policy, we allowed product-line differences to strengthen the organization instead of weakening it. Quantitative Process Management (QPM), Software Quality Management (SQM), and Defect Prevention (DP) activities are performed on pure product-line data by the people who manage the process.

Intergroup Coordination

When implementing the CMM, people like a discrete set of activities. Just do these tasks, and requirements management is covered; do these other activities, and project planning is satisfied. This makes for an understandable model. However, in the application of this model to real situations, the discrete independent nature of the activities is not so clear or even desirable. Do you want to have a separate meeting for every activity in the CMM that implies people need to meet? This type of implementation is advocated by some people because it is simpler to explain. It makes it easier to measure and easier to verify. And if the process were the product, it would probably be the best implementation.

However, in the real-world application of the CMM, virtually everything is integrated. Every KPA contains activities that touch or affect virtually every other KPA. The same applies to intergroup coordination. This is one of the more important KPAs, but it is mostly comprised of activities that are not activities unto themselves but are modifiers of other activities. If a requirements review is the activity, having all affected groups represented at the review is not a separate activity—it describes the right way to hold the requirements review.

The assessment team was impressed with the intergroup coordination not being just a set of activities or meetings to be attended—it was a philosophy inherent in all our activities. All groups are involved in requirements definition, all groups have input to the project plans, all affected groups are represented in peer reviews, and requirements changes are reviewed and agreed to by all affected groups. Our intergroup coordination is built directly into and distributed across the entire process.

Measurement

It seems as though the CMM says to measure everything and gives countless examples of what to measure, most of which are the kinds of things on which you would never do statistical analysis. Therefore, it is important to know the difference between measuring to know the status of an activity and measuring for statistical analysis.

For instance, minutes of a requirements review or technical percent complete from a status review are examples of measurements. From these measurements, individual managers can determine the status of their projects. However, to do statistical analysis on this data, these attributes should be measured consistently across the organization. The measurement should be made by the people doing the work and should be based on agreed-to definitions. This is why the requirement to measure is integrated throughout the CMM, but the concept of using statistical techniques is not required until Levels 4 and 5, when your measurement capabilities have grown and matured to the point where you have data that is consistent and suitable for statistical analysis.

Everything you measure costs money and takes time. To measure it with the rigor required for statistical analysis costs even more. We focused on automating the measurements we needed to produce our core metrics for cost, schedule, and quality. We measure hours of effort by process block, technical percent complete, start and stop dates, and source lines of code. For defects, we record process block injected and block detected, type, severity, and a description. Our strategy is to

measure these attributes against blocks in our process.

Quantitative Project Management vs. Quantitative Process Management

Managing projects is not the same as managing processes. Project management is a real-time activity, like sailing a ship from London to New York. Once under way, your main concern is where we are now, and what is the most efficient way to get where we are going. Quantitative process management is about using data to plot a safer, faster course for the next voyage.

Unfortunately, many metrics programs are not made to do both. For instance, the Cost Schedule Control System is made to manage open, active projects—not to analyze process performance of closed projects. Many systems also are overly complicated, requiring a class in how to interpret them every time there is a management review.

Our simple metrics program treats cost, schedule, and quality as separately derived attributes of status. Cost variance is based on planned cost and incurred cost to date, which we can accurately measure. Schedule variance is based on planned schedule performance and actual schedule performance. Actual schedule performance is determined from a combination of earned value for completed milestones and technical percent complete estimates from the people working uncompleted milestones.

Cost variance is measured in dollars, and schedule variance is measured in days. Actual schedule performance is estimated as accurately as possible based on all available data—not merely completed milestones, which may be too hard to plan in small enough increments to be accurate. Our primary quality metrics are defect detection ratio (yield), defect density, and defect injection rate.

Our system works for both project management and process management. The main difference is that project management metrics are calculated on open, active projects. On the other hand, planned cost performance, planned schedule performance, budget at completion, and schedule at completion are taken from the latest negotiated agreement with the customer. Again, the emphasis is on how we get from where we are today to where we want to be.

Process management metrics are calculated from data on closed projects. The same attributes are used, but are taken from the original estimates, plus estimates from negotiated functional requirements changes. The emphasis is on making better estimates and producing more viable project plans. Averages and standard deviations for our core process management metrics and other product-line-specific metrics make up the capability baseline for each product line. These are reviewed quarterly by the ESEPG and updated with the data from projects closed within the last quarter. The capability baseline data is then used to estimate new projects.

Quantitative Process Management, Software Quality Management, and Defect Prevention

Many people think of project status in terms of cost and schedule. High-maturity organizations view status in terms of cost, schedule, *and* quality. Quantitative process management and software quality management are essentially the same activities: One is performed on cost and schedule data, the other on defect data. However, when the CMM was written, it was probably uncommon or even controversial to think of a defect detection ratio as a project status indicator.

In our implementation, quantitative process management, software quality management, process change management, and defect prevention are tightly integrated activities. One reason we jumped from a Level 3 to a Level 5 is once we had the data to do software quality management (defect types, block injected, block detected, severity, etc.), we had everything we needed to do defect prevention. Once we had the data necessary to do quantitative process management, we realized we had everything we needed to do process change management, so we pushed back our assessment approximately one year and pursued Level 5.

Our approach was to measure continuously and analyze periodically. Cost, schedule, and quality data are measured continuously by the people doing the work. We tried to automate this as much as possible, using single-point-of-entry systems. For instance, when we first started gathering defect data from peer reviews, an engineer would record anomalies on a form. Then, at the review, valid anomalies would be manually recorded in the peer review minutes. Then, the project leader would enter the defect data into a spreadsheet to make it easier to analyze. If we wanted to analyze the data from several projects together, we again somehow had to combine it. This was an overly redundant and time-consuming process.

One product line solved this problem by integrating a defect database with a CM database that was already being developed. Now, the reviewer enters anomaly data directly into a database. At the review, the anomalies are brought up online and marked as valid, invalid, or duplicate. Since it is integrated with the CM database, products cannot be configured unless all valid anomalies are resolved and checked off in the database. This database also has standard pull-down menus for defect type, process blocks, severity, and a text field to record a description. Most engineers do not mind recording data if it is quick and easy and does not get in the way of what they consider the real work. We also leveraged off our time-accounting system to measure effort by process block. Now, when people charge time to a project, they also have to attribute it to the process block.

If you minimize redundant handling or entering of the data and automate the analysis, people will participate.

The ESEPGs meet quarterly to review the data and update their capability baseline as necessary. Cost, schedule, productivity, and quality data are analyzed. Defect detection ratios and defect injection rates are calculated for each process block. Pareto diagrams are done for defect type, block injected,

block detected, severity, rework by process block, etc. As a result of this analysis, action items are assigned, and (if necessary) action teams are formed to further analyze process problems, causes of defects, or opportunities for improvement. The ESEPG also acts as the Configuration Control Board for product-line process changes.

Process Change Management and Technology Change Management

Because these two KPAs are similar, we combined the implementation into the same process. The Technology Change Management (TCM) process is an open process supported by an Intranet-based database accessible by everyone in the organization; anyone can submit proposals via the Intranet. The SEPG screens the proposals and facilitates as necessary. The key concept is that technology and process changes happen all the time. All we are trying to do is get people involved in using a consistent process to plan, evaluate, and implement changes. We do this by offering a way for them to formally propose ideas and resources to evaluate and test new technologies and a forum to publicize their results.

The TCM database is a repository for technology proposals and data. The SEPG only gets formally involved in the evaluation if they are requested to do so or if the proposal affects multiple product lines. Issues that affect only one product line are handled by the ESEPG. The SEPG wants to be viewed as supporting technology and process change initiatives at all levels of the division—not controlling them.

One way we evaluate major new technology or process changes is to pilot them on a small controlled scale. For example, one pilot project implemented the Team Software Process (TSP). Project members tailored a process from our existing development process, adding in the detail necessary for the TSP.

Conclusion

In the past, the term "software engineering" was usually an oxymoron. The enterprise of developing and maintaining software was not disciplined enough to be considered engineering, and the occurrence of post-release defects was considered an annoying but expected problem. Discipline is what the CMM and other such models offer. However, like any model, it needs to be applied with common sense. Treat the CMM as a guide, strive to understand the concepts on which it is based, then do the right things for the right reasons. ◆

About the Author

Patrick W. Cosgriff is an SEPG member in the Technology and Industrial Support Directorate, Software Engineering Division at the Ogden Air Logistics Center at Hill Air Force Base, Utah. Before entering the SEPG, he was the lead of the Quality Engineering Support Team. His operational background is in automatic test equipment doing test program set development and maintenance. He has a bachelor's degree in electrical engineering from South Dakota State University.

OO-ALC/TIS-3
7278 Fourth Street
Hill AFB, UT 84056-5205
Voice: 801-777-3799 DSN 777-3799
Fax: 801-775-4444
E-mail: cosgrifp@software.hill.af.mil

References

1. Paulk, Mark C., et al., *The Capability Maturity Model: Guidelines for Improving the Software Process*, Addison-Wesley, Reading, Mass., 1995.

Assessing a Level 5 Organization

Mark C. Paulk, *Software Engineering Institute*
David B. Putman, Ogden Air Logistics Center, *Software Engineering Division*

This article describes the assessment of the Ogden Air Logistics Center (ALC) Software Engineering Division (TIS) that resulted in a Capability Maturity Model (CMM) Level 5 rating. It also discusses the issues in preparing for the assessment and reviewing the processes of TIS from both an internal (Putman) and external (Paulk) assessor's viewpoint: concerns going into the assessment and how they were resolved, alternate implementations that were discussed by the assessment team and how the Software CMM practices were judged to be satisfied, and controversial issues that sparked discussion in the assessment team and how a consensus was reached on their resolution. Specific issues include separating process and product assurance responsibilities, stability of continually improving processes and the related data, satisfactory evidence of institutionalization, and adequate implementation of Quantitative Process Management. The article concludes with a description of the challenges that TIS overcame and some of its strengths that may be of use to maturing organizations.

TIS has been actively engaged in process improvement using the Capability Maturity Model® for software (software CMM®) since the early 1990s and was assessed at Level 3 in 1995.

An assessment was performed in 1998 using the CMM-Based Appraisal for Internal Process Improvement (CBA-IPI) method [1]. It resulted in a Level 5 rating for TIS, the highest that can be achieved for the software CMM. The assessment covered all the software development and maintenance activities within the division. The diversity of the software efforts within the division range from well over 100 employees who support a single operational flight program (OFP) update to one-person efforts to support automated test equipment (ATE) test programs. The goals for this assessment were to

- Measure software process improvement progress made since the 1995 assessment.
- Provide a maturity level rating for TIS.
- Provide findings to a level of detail sufficient to enable TIS to refocus the process improvement efforts division-wide and to identify improvement candidates unique to TIS product lines.

The purpose of this article is to discuss the issues involved in preparing for the assessment and coming to a team consensus on the TIS processes and how the issues address the key process areas (KPAs) in the software CMM.

Capability Maturity Model and CMM are registered in the U.S. Patent and Trademark Office.

Preparing for the Assessment

The assessment team consisted of nine people, five of whom were external to TIS (three from the Software Engineering Institute [SEI] and two from Warner Robins Air Logistics Center) and led by Mark Paulk and Brian Larman. Six of the team members were SEI-authorized lead assessors. Before making the assessment, the team went through a refresher on the CBA IPI method plus a tutorial on Level 4 and Level 5 to ensure a common understanding and perspective on the high-maturity KPAs and the assessment method.

The appraisal covered all KPAs of the software CMM with the exception of software subcontract management, which was defined as "not applicable" by TIS. Prior assessments had only covered the Level 2 and Level 3 KPAs and had taken two weeks to complete. In these prior assessments, the team members often began work at 8 a.m. and occasionally worked past midnight. The scope of this assessment added Level 4 and Level 5 KPAs, and the assessment period was shortened by one day because TIS has a 5-4-9 work schedule (work nine days in two weeks and have every other Friday off). To increase the assessment scope while shortening the on-site period presented a risk, but a high-maturity organization possesses extensive data and can be expected to demonstrate the capability and effectiveness of its processes fairly quickly, so the risk was judged manageable.

The sponsor requested that although TIS be designated as the organization for which the rating was to be awarded, unique findings were to be provided to each of two product lines. One of TIS's stated objectives for the assessment was that the assessment team identify areas in which TIS could improve. As a result, TIS wanted to be sure it could apply the recommendations from the assessment to the proper area. This meant that for each product line, separate observations were maintained for those areas that could potentially compromise the CBA-IPI attribution and confidentiality rules. Four focus projects, two from each product line, were selected as representative samples of the projects within the division. There was some concern that selecting only two focus projects from each product line would limit the assessment team's ability to get complete data and cause difficulty in obtaining adequate corroboration. To address these concerns,

- The team interviewed nine project leaders in group interviews—usually referred to as functional area representative (FAR) sessions—in addition to the individual project leader interviews for the focus projects.

- The team interviewed 25 practitioners in various practitioner interviews. The practitioners were selected to provide representation from all the projects within the organization.
- Assessment participants (interviewees) were informed of the potential for product-line-specific findings in the opening briefing.

Prior to the assessment, 51 CMM questionnaires were completed by employees across TIS and analyzed by the assessment team. For the assessors, the primary value of these questionnaires was in the comments from the respondents that identified specific processes and documents that the respondents interpreted as addressing the question areas.

Concerns About the Assessment – An External Perspective

From an external perspective, there were three primary concerns:

- Management sponsorship for true improvement, as opposed to obtaining a score.
- Potential conflict of interest for the TIS team members who were in the Software Engineering Process Group (SEPG).
- The definition of "organization."

One of the hard-learned lessons for lead assessors is that sponsors must be truly motivated to achieve process improvement and create a "quality culture," as opposed to achieving a high score regardless of true capability. One of the tools in the *Lead Assessor's Guide* [2] is an assessment readiness survey. It was clear prior to the assessment that there was an expectation or desire that TIS would "score well." In meetings prior to the assessment, the assessment team leaders probed this issue carefully with the sponsor. Our conclusion was that it did not appear to be a major problem, and as the assessment progressed, the evidence of a quality culture across TIS was convincing.

The assessment team included four people from within TIS: two from the SEPG, one from an extended SEPG, and one from the Software Technology Support Center. Since SEPG members are responsible for process improvement, there is the potential for a conflict of interest when adverse findings arise. This concern was carefully monitored by the team leaders and never became a significant issue during the assessment. During the SEPG interviews, these four team members switched roles and acted as interviewees rather than assessment team members.

The last concern was over the definition of "organization." Since TIS consisted of two product lines, with identifiably different organizational processes, it was possible that problems in one product line could limit the maturity level rating for the organization as a whole. Since findings were requested to be specific to product lines where appropriate, the consequences of a lower-than-desired level rating might be traced to a product line, which would lead to a potential confidentiality issue. One alternative was to rate the two product lines separately. This concern was discussed with the sponsor, and the decision was made to go with the division-level definition of organization, based on the argument that it was divisional process capability that was the concern of the sponsor.

Preparing for the Assessment – An Internal Perspective

One TIS goal in preparing for the assessment was to make the assessment as easy as possible, from both the assessor's and the interviewee's points of view. For the interviewee's point-of-view, TIS was able to draw on personal experience from previous assessments. In order to address the assessor's point of view, the TIS SEPG reviewed the documentation (reports, lessons learned, etc.) from previous assessments and requested input from prior SEPG members who were involved with the previous assessments. (SEPG members are assigned on a rotational basis. As a result, the SEPG members involved with the previous assessments had left the group.)

The TIS Quality Engineering Support Team (QuEST) provides process quality assurance with regard to the requirements identified in division-level policy. It was decided to perform a Snapshot assessment in addition to the normal QuEST activities because QuEST is not chartered to look for differences in interpretation between division-level policy and the CMM. This Snapshot was the first time that the organization's practices relating to the Level 4 and Level 5 CMM issues were assessed from the interpretation of the CMM. This Snapshot assessment helped assure TIS that it was on the right track.

Since the focus of QuEST reviews and the Snapshot assessment were different, TIS required that the action plans for the Snapshot findings be separate from the action plans generated from the QuEST reviews. However, in order to manage the two action plans, the tasks identified in the two action plans were combined into one process improvement Gantt chart.

The SEPG took the approach that the assessment was not only an assessment of the maturity of the organization but also an assessment of how well the SEPG helped the projects prepare for the assessment. Six months before the assessment, the SEPG prepared a high-level Gantt chart that planned the remaining activities necessary to prepare for the assessment. These activities helped to reduce the anxiety and stress experienced by the organization as it went through the formal assessment. The activities also proved to be extremely beneficial in helping the assessment team adequately assess the organization's maturity.

It has always been a requirement that TIS managers manage their process improvement efforts as a project. Following a QuEST audit (see "QuEST Audits," page 13), each project is responsible for defining its process improvement activities in an action plan. The general format of TIS action plans is to respond to the QuEST findings with a proposed corrective action and a proposed completion date for each finding. The action plans can be lengthy for fledgling projects. Even a two- or three-page action plan for a mature project can be difficult to manage if the tasks are not properly planned. As a result, the projects convert their action plans into condensed Gantt charts.

In preparing for the assessment, the SEPG took the approach that the results

Tag	CMM Requirement	Organizational-Level Documents				
		OI 63-1	SP	MIG	SEP	QP
Ability 1	For each project, responsibility is established for analyzing the system requirements and allocating them to hardware, software, and other system components.	Paragraphs A2.2, A2.2.1-3			Paragraph 4.2.1-2	
Ability 1	A group that is responsible for coordinating and implementing SQA for the project, i.e., the SQA group, exists.	Paragraphs A6.2.4, A6.2.4.1-2			Paragraph 4.1	Paragraph 2.1.3
Commitment 1	The organization follows a written policy to measure and quantitatively control the performance of the project's defined software process.	Attachment 3, Attachment 4, A10.1.1, A10.1.1.1-2, Attachment 11	Chapter 2, Section 1.3.6.2, Chapter 3, Section 6.1.1.1	The overall TIS-MIG	Paragraph 5.2	Paragraphs 4.14, 4.20

Table 1. *Mapping of TIS organizational-level processes to CMM practices. This same detail of mapping was performed using the product-line documents in place of the organizational-level documents.*

of the assessment should not come as a surprise. To make sure there were no surprises, the first step was to develop a spreadsheet that mapped every goal, commitment, activity, etc., for each KPA to the corresponding paragraph in each of the documents, at the organization level and product lines. An abbreviated example of this mapping is shown in Table 1.

This activity provided TIS with four major benefits:

- A verification that a critical item had not been overlooked. In a few cases, the comment was heard, "We are doing the activity, but the wording in our process is vague." This exercise enabled TIS to improve the areas that were vague.
- The organization was comforted to know they had covered all of the issues.
- The mapping gave the interviewees confidence in their ability to easily find any reference necessary.
- The mapping helped the assessment team easily follow "the thread" from the CMM reference down through all of the organizational-level and product-line documents.

A potential problem with the implementation of this mapping was discovered: If not maintained properly, the spreadsheets quickly become outdated as changes are made to the processes. For example, the insertion of a paragraph in the TIS policy could have a ripple effect on the paragraph numbers that followed in the policy. The benefits of mapping TIS documents to the CMM are considered worth the effort necessary to keep the mapping up-to-date.

TIS took the same approach to the assessment as a college student might take in preparing for an open-book oral examination in which the student is given the questions in advance. It was known that the assessment team would want to see examples of TIS activities. To ensure that the examples were readily available, the SEPG chose to have all the project leaders place their examples in three-ring binders with pre-labeled tab pages and check-sheets that corresponded to the KPAs in the CMM. The check-sheets were not exhaustive; they were meant to be used as an example of the type of documents that would demonstrate that the activity was being performed.

The project leaders were responsible for placing proper examples in the binders. In addition to the examples, the SEPG recommended that a reference be included on how the assessment team could get additional examples, such as indicating the point-of-contact and where the documents were located.

Many of the TIS processes have been automated, and all of the documentation is available on internal networks. As it turned out, the online systems are extremely beneficial for day-to-day TIS activities, but they quickly proved to be cumbersome for the assessment team. Tracing the process thread through the documents often required jumping back and forth between the documents. Using hard-copy printouts proved to be easier than tracing the thread online. As a result, the SEPG printed copies of all TIS processes, plans, procedures, etc., for each of the four miniteams.

The SEPG saw the TIS familiarity with the CMM as both an asset and a risk. Many of the day-to-day activities were no longer perceived as CMM issues. There also were concerns about the wording of the interview questions because TIS terminology is often different from terms used in the CMM. Table 2 contains a mapping of some of the TIS terms to the CMM-equivalent concept. Note that in some cases, multiple TIS terms map to a single CMM concept and vice versa. These relationships had to be understood by the assessment team and communicated to the assessment participants as needed.

During the Site Visit – Coming to Team Consensus

There is a difference between describing *what* must be done vs. describing *how* something must be done. The KPAs in the CMM were written to show what an organization should be doing; they often include examples that may help the

Acronym	TIS Term	CMM-Equivalent Concept
ATE	Automatic Test Equipment	Organization and Project
	Branch Chief	Senior Management
CBA-IPI	CMM-Based Appraisal for Internal Process Improvement	CMM-Based Appraisal for Internal Process Improvement
CM	Configuration Management	Software Configuration Management and Software Quality Assurance
CMM	Capability Maturity Model	Capability Maturity Model
	Division Chief	Senior Management
ECP	Engineering Change Proposal	
ESEPG	Extended Software Engineering Process Group	
FAR	Functional Area Representative	
GCAR	General Corrective Action Report	
KPA	Key Process Area	Key Process Area
MEP	Maintenance Engineering Process	Organization's Standard Software Process
MIG	Metrics Implementation Guide	
MIP	Material Improvement Project	
OCP	Organic Change Proposal	
ODG	OFP Development Guide	Organization's Standard Software Process
OJT	On-the-Job Training	
OFP	Operational Flight Program	Organization
OO-ALC	Ogden Air Logistics Center	Organization
PIPR	Process Improvement Peer Review	
PMR	Program Management Review	
PSP	Personal Software Process	
QuEST	Quality Engineering Support Team	Software Quality Assurance
ROM	Rough Order of Magnitude	Estimate
	Section Chief	Senior Management
SEP	Standard Engineering Process	Organization's Standard Software Process
SEPG	Software Engineering Process Group	
SPC	Statistical Process Control	
SQA	QuEST and CM	Software Quality Assurance
TIS	Technology and Industrial Support Directorate	Organization
TPM	Technical Program Manager	Project Manager and Project Software Manager

Table 2. *Mapping TIS terms to CMM concepts.*

organization meet the goals of a stated objective. The organization is responsible for determining how to design their processes to fulfill the stated objectives. Pat Cosgriff's article, "The Right Things for the Right Reasons" on page 16, describes how TIS learned to fully understand the underlying concepts and intent of each KPA. By understanding the concepts and intent, TIS was able to implement processes that met both the needs of the organization and the requirements of the CMM.

Four miniteams of two people each were assigned the primary responsibility for investigating each of the KPAs. Although each KPA had its miniteam, all team members participated in the interviews with the branch chiefs, section chiefs, technical program managers (TPMs), and FARs, and all assessment observations and conclusions were obtained by consensus of the entire team.

The appraisal team interviewed TPMs in individual sessions for each of the four focus projects as well as in a FAR session with TPMs from several other projects within TIS. In addition, a broad sampling of employees from across the organization at all levels—from managers through practitioners, plus members of the SEPG and QuEST—were also interviewed. In all, the assessment team interviewed 67 people during the two-week site visit.

From the interviews, document reviews, and questionnaire data, the team crafted 477 observations about the TIS processes. The following sections will present some specifics of the assessment and address some of the assessment teams findings and recommendations.

Software Project Planning

In TIS, project team members assist the TPMs in developing estimates for the project across both product lines. These estimates are documented in a variety of artifacts. As the tasking matures, the estimates are refined and documented in approved project directives, schedules, requirements documents, etc., in accordance with the project's defined software process. The project's defined process is documented in the OFP development guide (ODG) for the OFP product line

or in the maintenance engineering process (MEP) for the ATE product line.

Project plans were distributed across several documents; the ODG and MEP did not explicitly describe the components of the project plan. The assessment team, with some difficulty, located the areas in which the project plan criteria, as identified in CMM, Version 1.1, were satisfied. A new program manager could have difficulty following and understanding the project planning process. To address this potential deficiency, the assessment team agreed that although the plan criteria were in place, it would be desirable to add an introductory paragraph before the product-line processes that describes the various components of the plan and how project planning is accomplished.

The TPMs establish interdisciplinary project teams at the beginning of the project to plan intergroup activities. These project teams meet periodically to track and resolve intergroup issues.

The assessment team had no difficulty agreeing that the product-line processes in the ODG and MEP were consistently used by projects across TIS. It also was apparent that TPMs used extensive product-line data in planning and establishing thresholds to trigger corrective action.

Product and Process Software Quality Assurance

Software quality assurance (SQA) activities are divided between QuEST and configuration management (CM); QuEST ensures process quality, and CM ensures product quality. QuEST focuses on ensuring compliance to division policy and product-line processes and procedures. The CM staff ensures compliance of work products to style guides, templates, and standards. The assessment team identified this as an alternate practice to the traditional independent SQA group responsible for both process and product assurance but agreed that this satisfied the "objective verification" goals for SQA. As seems characteristic of most high-maturity organizations, a significant portion of the SQA function is embedded in the process [3].

The assessment team observed that project members value the discipline provided by the CM group and the recommendations and training provided by QuEST. The implementation of this CMM alternate practice has worked particularly well for TIS. The alternate practice of dividing SQA between CM and QuEST has allowed QuEST to focus on measuring and reporting status on software engineering practices as well as providing informal training to all levels of TIS.

Organizational Process Focus – Process Definition and Improvement

When TIS began its process improvement efforts, it developed many individual processes. As the organization matured, it slowly embraced the concept of (or recognized the existence of) two main software product lines within the division: OFP and ATE. (See "Product Lines," page 17.) Over time, the numerous individual processes were divided into one of the two main product-line processes, the ODG and MEP.

TIS has a high-level standard engineering process (SEP) from which the product-line processes are tailored. The SEP provides a framework to embrace the strengths and diversity of each product line, and the associated product-line process documents reflect the product-line view of the SEP. The assessment team was easily able to trace from TIS policy through the SEP to the product-line process documentation.

The assessment team quickly identified the product-line processes as one of the strengths of TIS. The environment and terminology of these two product lines differ greatly. TIS has captured a great deal of product and process expertise in the ODG and MEP. The benefits that TIS has experienced as a result of adopting the product-line processes are a reduction in the cost to maintain the processes, and CMM activities have become more common across the product lines.

The TIS Metrics Implementation Guide (MIG) defines cost, schedule, and quality measures to be collected across TIS. This is particularly important for defining operational definitions of TIS measures that are consistent and comparable across the division. Operational use of the MIG data, however, is within a product line, and some measures are defined differently for different product lines. For example, the term "size" has a different meaning in each product line. The OFP product line produces major capability upgrades, whereas the ATE product line produces numerous and frequent upgrades. To measure the number of source lines of code works well in the OFP product line. Throughput, or the number of products produced, provides a better "size" estimate in the ATE product line. Cost, schedule, and quality data are tracked to the process block. This level of granularity is crucial to analyzing process performance and determining process capability accurately.

The assessment team had no difficulty agreeing that people across TIS understand and are committed to the stable processes achieved through implementation and institutionalization of the CMM. The team observed the use of detailed procedures, templates, and checklists on all projects in a culture where process discipline and process improvement are considered integral to success.

Process improvement activities are managed like a project and briefed monthly at program management reviews. Extended SEPGs (ESEPGs), (see page 18), work closely with the SEPG to identify strengths and weaknesses in the processes. Anyone in the organization can submit process improvement recommendations via an automated submission process, which is also used to track technology change proposals.

Process improvements are planned following each formal assessment and QuEST audit. The assessment team observed a strong quality and process culture that has been instilled across TIS.

Training Issues

TIS has sophisticated and powerful processes, yet there is little "formal" training specific to the TIS ODG and MEP processes. TIS has established a

mentoring program, with well-defined criteria, as a form of training new employees. TIS training is planned and tracked through a training matrix, a project skills document, and individual development plans. Some formal training takes place, for example, process training by the SEPG, but most training is informal and on the job.

Training was extensively discussed by the assessment team. This implementation was a conscious decision by the TIS Training Board, which is responsible to oversee the training requirements for the division. It was clear that there had been few problems from reliance on mentoring and on-the-job training, and the stability of the TIS work force was a significant factor in this success. At the same time, however, TIS was actively working to expand its workload and customer base. A significant influx of new employees could put the stability of the TIS processes at risk. This risk was, however, one that TIS had considered and has deliberately chosen to accept. The assessment team concluded that the training program KPA had been adequately addressed.

Quantitative Process Management – Managing the Process and the Product

The toughest decision for the assessment team regarded the satisfaction of the quantitative process management KPA. Although the use of data and management by fact were thoroughly embedded in the TIS processes and culture, there also were incorrect applications of statistical techniques.

TIS frequently used "control charts," but the "control limits" were not always calculated according to control charting principles. In some cases, the control limits were thresholds set by management.

Some control charts used standard deviation as an estimator for sigma, which is a common mistake. Control limits for some processes were extremely wide—too wide to provide value.

The assessment team discussed these concerns in depth. The consensus was that although there were some mistakes in the analytical techniques used to control some processes, the general culture of measurement-driven decision making was good, and the analyses, both good and bad, were comparable to those of other Level 4 and Level 5 organizations. Few, if any, software organizations have truly mastered statistical process and quality control when initially assessed at Level 4 or Level 5. The journey of continual process improvement addresses these types of problems, and the issues were reported in the findings.

Another challenge for TIS and the assessment team is the validity of process data when the processes are being continually improved. Many process changes occurred during the months prior to the assessment, which affected the reliability of the process data to predict and control the process. Capability baselines for OFP and ATE were established using 12-month averages for closed projects, but four-month averages were also charted to allow for process changes (plus other analyses to confirm data validity). The assessment team agreed that TIS had recognized this risk and was managing it effectively.

Quality goals for projects are based on product-line history and revised quarterly, although projects keep their initial goals from project initiation. Software quality performance is reviewed by the project teams, ESEPGs, and management. The assessment team observed that the goal of releasing fewer defects to the customer was being achieved. The assessment team concluded that the project results reflected the high process capability observed.

Non-CMM Observations

The assessment team concluded that TIS is composed of professionals who are highly motivated, greatly experienced and knowledgeable, proud of the significance of the work they perform, cooperative in their team work, and focused on product quality and continual improvement. TIS's quality culture is thoroughly established.

There were a number of business issues concerning adding work and expanding the customer base, and TIS management was actively working on these issues. The team reported on these concerns, which had been repeatedly expressed by the assessment participants, but they were ultimately outside the scope of the software process improvement effort.

After the Assessment

The assessment team documented its findings and recommendations in a final findings report. The TIS staff and managers were delighted to hear that they had achieved Level 5, but the challenge of continual process improvement remains. Even a Level 5 organization must pro-actively deal with the stresses of a dynamic business environment.

Achieving Level 5 is not the end of the journey. Following the assessment, TIS immediately began to plan for the future, which included the following issues.

- Planning the next QuEST process quality review cycle.
- Updating the organization's strategic plan to address the recommendations from the assessment team.
- Planning for an off-site meeting of the Executive Board to thoroughly review the strategic plan and update the organization's long-term goals and objectives.
- Evaluating the Capability Maturity Model–Integrated Software/System Engineering (CMMI SW-SE) and addressing the possibility of piloting the CMMI SW-SE at an enterprise-wide level of implementation.

Conclusion

The preparations for the assessment were worth the effort. The SEPG believes that it achieved its goal of making the assessment as easy as possible, from both the assessor's and the interviewee's points of view. To assess a Level 5 organization is much easier than assessing a Level 1 organization, because the data to demonstrate a process and its effectiveness are readily available and easily understood.

The organization achieved its goals of achieving the Level 5 maturity, re-establishing a baseline of its capability, and of identifying areas on which TIS can focus while continuing its journey of continual process improvement.

Acknowledgments

We thank Dan Wynn, the sponsor of the assessment, for his support; the TIS support staff—Lynn Silver, Walt Donohoo, Rushby Craig, and Steve Tiede—and the other assessment team members: Mike Ballard, Bonnie Bollinger, Pat Cosgriff, Donna Dunaway, Dave Haakenson, Brian Larman, and Millee Sapp. ◆

About the Authors

Mark C. Paulk is a senior member of the technical staff at the SEI. He has been with the SEI since 1987, initially working with the Software Capability Evaluation project. He was the principal contributor and editor for Version 1.0 of the CMM for software. He also was the project leader during the development of CMM, Version 1.1. He is also actively involved with the International Organization for Standardization and the Institute of Electrical and Electronics Engineers (IEEE) software engineering standards. Prior to joining the SEI, he was a senior systems analyst for System Development Corporation (later Unisys Defense Systems) at the Ballistic Missile Defense Advanced Research Center in Huntsville, Ala. He has a master's degree in computer science from Vanderbilt University and a bachelor's degree in mathematics and computer science from the University of Alabama in Huntsville. He is a senior member of the IEEE and the American Society for Quality (ASQ) and an ASQ certified software quality engineer.

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213
Voice: 412-268-5794
Fax: 412-268-5758
E-mail: mcp@sei.cmu.edu or
Mark.Paulk@ieee.org

David B. Putman is the lead of the TIS SEPG at Hill Air Force Base. He has worked in software engineering for 20 years. He has more than 17 years experience in ATE, two years with Hughes Aircraft and more than 15 years with the Air Force. During his involvement with ATE, he was senior engineer in the Avionics Software Support Branch for nine years, and he supervised ATE engineering teams for two years. He also supervised the F-16 OFP System Design and Integration Test teams for one year. He has a bachelor's degree in electrical engineering from the University of Utah and a master's degree in business administration from Utah State University.

OO-ALC/TIS-3
7278 Fourth Street
Hill AFB, UT 84056-5205
Voice: 801-777-4726
Fax: 801-775-3023
E-mail: putmand@software.hill.af.mil or
david.putman@hill.af.mil (effective late summer 1999)

References

1. Dunaway, Donna K. and Steve M. Masters, "CMM-Based Appraisal for Internal Process Improvement (CBA-IPI): Method Description," Software Engineering Institute, Carnegie Mellon University, CMU/SEI-96-TR-007, DTIC No. ADA307934, 1996.
2. Dunaway, Donna K., "CMM-Based Appraisal for Internal Process Improvement (CBA-IPI) Lead Assessor's Guide," Software Engineering Institute, Carnegie Mellon University, CMU/SEI-96-HB-003, March 1996.
3. Paulk, Mark C., "Practices of High Maturity Organizations," The 11th Software Engineering Process Group Conference, Atlanta, Ga., March 8-11, 1999.

Additional Reading

1. Paulk, Mark C., et al., *The Capability Maturity Model: Guidelines for Improving the Software Process*, Addison-Wesley, Reading, Mass., 1995.
2. Wheeler, Donald J., *Understanding Variation: The Key to Managing Chaos*, SPC Press, Knoxville, Tenn., 1993.

USDA's National Finance Center Receives CMM Level 2 Rating

The National Finance Center (NFC), an agency of the U.S. Department of Agriculture, completed its external assessment Sept. 18, 1998 and received a Software Engineering Institute Capability Maturity Model (CMM) Level 2 rating. NFC's efforts, under the direction of John Ortego, place the center among the top 30 percent of all federal organizations assessed under the CMM since 1986. NFC also received the *Government Computer News Award* in March 1999 for its CMM efforts. Look for an article in the June 1999 issue of *CROSSTALK* that describes NFC's 10-month journey to Level 2, their challenges and successes, and the help they enlisted from the Software Technology Support Center.

A CBA-IPI Reveals More Than Findings

David P. Quinn
National Security Agency

Since the beginning of the Capability Maturity Model (CMM)-Based Appraisal for Internal Process Improvement (CBA-IPI) method, Software Engineering Institute and CBA-IPI assessors have stated that appraisals are meant to improve an organization, not merely determine its current maturity level according to the CMM. Yet, some people still view the CBA-IPI as a way to get a maturity rating. Some shun the CBA-IPI in favor of the less expensive, less intensive Software Capability Evaluation, but they need to know that the CBA-IPI discovers more than findings, which makes it worth the extra investment.

What Appraisal Teams Do

The CBA-IPI appraisal team gathers data about the organization to compare with the practices in the CMM. The data comes from documents, interviews, and questionnaire responses from a number of projects. The data are characterized as strengths or weaknesses as they are mapped to the CMM practices.

Throughout the appraisal, the data status goes from "needing more information" to "corroborated." Practices are judged as either common practices—*the rule*—or outliers of the organization—*the exception*—depending on the number of projects that use them. The appraisal team reports those practices deemed the rule during the Final Findings Brief. The exceptions go unreported but remain with the appraisal team members when the organization begins its improvement program.

The Importance of the Categories

When we view the strengths and weaknesses by the rule or the exception, the data take on additional meaning for the organization's improvement program. Table 1 shows a short definition of the meaning of each category. By applying the data as described, the CBA-IPI moves beyond a rating method to become an improvement tool.

Strength as the Rule

Every organization has some strengths in its practices. In more mature organizations, they are found in the Organization Standard Software Process (OSSP); less mature organizations use these practices as a matter of survival. Whether part of the OSSP or merely a means to survive, these practices represent the organization's current best practices. Projects share them, and practitioners use them on a regular basis.

The CBA-IPI reports these best practices for each key process area (KPA) rated. They receive some fanfare during the appraisal but little attention afterward. After the appraisal, the appraisal team can identify where these best practices reside, so the organization can leverage them as part of the process improvement program.

Weakness as the Rule

The real reason to do an appraisal is to find the *targets for improvement*—those practices that are considered general weaknesses with regard to the CMM. Weaknesses may not be glaring problems for the organization, but they are inhibitors to sound software engineering.

Whenever someone mentions appraisal findings, they tend to think in terms of the organization's general weaknesses. Findings are listed by KPA and act as the requirements for the improvement program. Action planning is done based on these findings. Progress is shown by removing these weaknesses from general practice.

Strength as the Exception

It is not unusual for the appraisal team to stumble on a unique practice that provides additional software engineering capability to a project, especially when appraising a Level 1 organization. The practice may match a CMM practice that does not exist in the rest of the organization, or it may be an alternative practice to the CMM that someone in the organization innovated. Either way, the practice is not in general use by the organization, but it is considered a strength.

Usually, exceptions are important to the organization even though they are not mentioned in the Final Findings Brief. They represent a potential solution to targets of improvement or provide an additional alternative to the organization's best practices. The appraisal team must identify exceptions during the action-planning phase of the improvement program to leverage the practice across the organization.

Weakness as the Exception

Because software engineering practices are performed by humans, it is unrealistic to expect everyone in the organiza-

Table 1. *Categories of data.*

	The Rule	The Exception
Strengths	Current Best Practices	Potential Solutions
Weaknesses	Targets for Improvement	Resistance or Irrelevance

tion to do everything the same way. Some variations are healthy and are part of tailoring the OSSP to a project's specific needs. Other variations represent exceptions that may inhibit the organization's ability to mature. These exceptions are weaknesses that may or may not be addressed by the improvement program, depending on their impact. However, the appraisal team must keep track of these exceptions to allow the organization to take corrective actions.

An organization does not address an exception for legacy systems with a life of less than a year. Because these exceptions have a limited life, they are not worth fixing. They represent limited risk to the organization because they are not likely to be continued in other projects.

An organization needs to address an exception that represents resistance to sound software engineering. Fortunately for the organization, the appraisal identifies a set of best practices to provide to these resisters. The appraisal team should make sure the improvement program is aware of the resisters and recommend the best practices to use.

Summary

When an organization undergoes a CBA-IPI, the Final Findings Brief presents the general strengths and weaknesses of the organization by KPA. Although these findings are officially reported as part of the appraisal's end product, they are not the only strengths and weaknesses found during the appraisal. The appraisal team discovers exceptions that the organization should consider as part of the improvement program.

Exceptions that are strengths represent potential solutions for targets of improvement. It is crucial that the organization have access to these exceptions to make the improvement program more effective. The appraisal team provides that access.

Exceptions that are weaknesses represent possible resistance. The organization needs to address this resistance to minimize its impact. The best way to address this resistance is to use the strengths reported by the appraisal. The appraisal team can recommend which current best practices address the resistance.

Without the extensive investigation of a CBA-IPI, the exceptions in the process will not be captured or retained. The exceptions play an important role in the improvement program. This makes the CBA-IPI a valuable tool that goes beyond a mere rating method. It is an improvement tool worth any added expense. ◆

About the Author

David P. Quinn is chief of the Network Management Software Branch at the National Security Agency. He has more than 15 years experience in software development and maintenance. He spent four and one-half years working on software process improvement as a process consultant. He is certified by the Software Engineering Institute as a lead assessor for CBA-IPIs. He was also a member of the CMM Advisory Board.

National Security Agency
9800 Savage Road, Suite 6675
Fort Meade, MD 20755-6675
Voice: 301-688-2299
Fax: 301-688-2259
E-mail: dpqsr@romulus.ncsc.mil

Coming Events

The 21st International Conference on Software Engineering: Preparing for the Software Century

Dates: May 16-22, 1999
Location: Los Angeles Airport Marriott Hotel, Los Angeles, Calif. Collocated with SSR '99 (Symposium on Software Reusability).
Host: The Los Angeles chapters of the Association for Computing Machinery Special Interest Groups on Software Engineering, Programming Languages, and Ada.
Theme: "Software Engineering Challenges for the Global Electronic Community"
Contact: Ashley Queen, registration manager
Voice: 919-419-8242 ext. 17
E-mail: ashleyqueen@mindspring.com
Internet: http://sunset.usc.edu/r8/icse99

SSR '99: Symposium on Software Reusability

Dates: May 21-23, 1999
Location: Los Angeles Airport Marriott Hotel, Los Angeles, Calif. Collocated with The 21st International Council on Software Engineering.
Internet: http://csalpha.unomaha.edu/~ssr99

12th International Quality Week '99 (QW '99)

Dates: May 24-28, 1999
Location: San Jose, Calif.
Theme: Facing the Future
Sponsors: Software Assurance Technology Center at NASA, Bay Area Quality Assurance Association, Center for National Software Studies, Santa Clara Software Quality Association, and Software Research, Inc.
Contact: Rita Bral, conference director
E-mail: bral@soft.com
Internet: http://www.soft.com/QualWeek/QW99

PSQT '99: The International Conference on Practical Software Quality Techniques

Dates and Locations: June 7-10, 1999, San Antonio, Texas; Oct. 4-7, 1999, St. Paul, Minn.
Sponsor: The San Antonio Software Process Improvement Network
Featuring: Watts Humphrey, James Bach, Karl Wiegers, and Bob Glass.
Conference Chairman: Magdy S. Hanna, 8476 Bechtel Avenue, Inver Grove Heights, MN 55076
Voice: 651-552-0716
Fax: 651-552-0791
E-mail: mhanna@softdim.com
Internet: http://www.softdim.com

from TIME LINE, page 6

- If your organization is large and diverse, you may want to coordinate the development of project processes with the development of your organizational processes from the start. Again, this may reduce rework associated with fundamental style and design differences between different product lines.
- Emphasize performance of project planning activities—not creation of documents that gather dust.
- Do not ignore intergroup coordination. It is hard to get your hands on it, but it is critical. Think of coordination as a characteristic you want integrated into all your activities, not a discrete set of activities unto themselves.
- Consider the data requirements of the software quality management KPA when implementing peer reviews. It is not difficult to gather the extra data needed to support software quality management and defect prevention. Having historical data on defects will give you a big jump on implementing these higher maturity practices.
- Levels 4 and 5 KPAs can be implemented together—in fact, defect prevention is the logical extension of software quality management, and process change management is the logical extension of quantitative process management.

I have finally realized that person was right all those years ago: Process improvement is a journey. The levels are good because, like any journey, you need recognizable milestones along the way to keep you from feeling lost or discouraged. However, the blind desire to achieve the next level will usually send you down the wrong road. The desire to improve will keep you on the right track. ◆

About the Author

Patrick W. Cosgriff is an SEPG member in the Technology and Industrial Support Directorate, Software Engineering Division at the Ogden Air Logistics Center at Hill Air Force Base, Utah. Before entering the SEPG, he was the lead of the Quality Engineering Support Team. His operational background is in automatic test equipment doing test program set development and maintenance. He has a bachelor's degree in electrical engineering from South Dakota State University.

OO-ALC/TIS-3
7278 Fourth Street
Hill AFB, UT 84056-5205
Voice: 801-777-3799 DSN 777-3799
Fax: 801-775-4444
E-mail: cosgrifp@software.hill.af.mil

CROSSTALK

Sponsor	***Lt. Col. Joe Jarzombek*** 801-777-2435 DSN 777-2435 jarzombj@software.hill.af.mil
Publisher	***Reuel S. Alder*** 801-777-2550 DSN 777-2550 publisher@stsc1.hill.af.mil
Editor in Chief	***Brent Baxter*** 801-775-5747 DSN 775-5747 edit_chief@stsc1.hill.af.mil
Managing Editor	***Tracy Stauder*** 801-775-5746 DSN 775-5746 managing_editor@stsc1.hill.af.mil
Senior Editor	***Sandi Gaskin*** 801-777-9722 DSN 777-9722 senior_editor@stsc1.hill.af.mil
Associate Editor	***Lorin J. May*** 801-775-5799 backtalk@stsc1.hill.af.mil
Editorial Assistant	***Bonnie May*** 801-777-8045 editorial_assistant@stsc1.hill.af.mil
Features Coordinator	***Denise Sagel*** 801-775-5555 features@stsc1.hill.af.mil
Customer Service	801-775-5555 custserv@software.hill.af.mil
Fax	801-777-8069 DSN 777-8069
STSC Online	http://www.stsc.hill.af.mil
CROSSTALK Online	http://www.stsc.hill.af.mil/Crosstalk/crostalk.html
ESIP Online	http://www.esip.hill.af.mil

Subscriptions: Send correspondence concerning subscriptions and changes of address to the following address:

Ogden ALC/TISE
7278 Fourth Street
Hill AFB, UT 84056-5205

E-mail: custserv@software.hill.af.mil
Voice: 801-775-5555
Fax: 801-777-8069 DSN 777-8069

Editorial Matters: Correspondence concerning *Letters to the Editor* or other editorial matters should be sent to the same address listed above to the attention of *CROSSTALK Editor* or send directly to the senior editor via the E-mail address also listed above.

Article Submissions: We welcome articles of interest to the defense software community. Articles must be approved by the *CROSSTALK* editorial board prior to publication. Please follow the *Guidelines for CROSSTALK Authors*, available upon request. We do not pay for submissions. Articles published in *CROSSTALK* remain the property of the authors and may be submitted to other publications.

Reprints and Permissions: Requests for reprints must be requested from the author or the copyright holder. Please coordinate your request with *CROSSTALK*.

Trademarks and Endorsements: This DoD journal is an authorized publication for members of the Department of Defense. Contents of *CROSSTALK* are not necessarily the official views of, or endorsed by, the U.S. Government, the Department of Defense, or the Software Technology Support Center. All product names referenced in this issue are trademarks of their companies.

Coming Events: We often list conferences, seminars, symposiums, etc., that are of interest to our readers. There is no fee for this service, but we must receive the information at least 90 days before registration. Send an announcement to the *CROSSTALK* Editorial Department.

STSC Online Services: STSC On-Line Services can be reached on the Internet. World Wide Web access is at http://www.stsc.hill.af.mil. Call 801-777-7026 or DSN 777-7026 for assistance, or E-mail to schreifr@software.hill.af.mil.

Back Issues Available: The STSC sometimes has extra paper copies of back issues of *CROSSTALK* available free of charge. If you would like a copy of the printed edition of this or another issue of *CROSSTALK*, or would like to subscribe, please contact the customer service address listed above.

The ***Software Technology Support Center*** was established at Ogden Air Logistics Center (AFMC) by Headquarters U.S. Air Force to help Air Force software organizations identify, evaluate, and adopt technologies that will improve the quality of their software products, their efficiency in producing them, and their ability to accurately predict the cost and schedule of their delivery. *CROSSTALK* is assembled, printed, and distributed by the Defense Automated Printing Service, Hill AFB, UT 84056. *CROSSTALK* is distributed without charge to individuals actively involved in the defense software development process.

The Key to Effective Writing and Coding: Quality Assurance

Today we are going to talk about the software development industry's number one problem: to many erorrs. Careless misteaks damage credibility more than you'd imagine. Because I"m constantly reveiwing article's written by software people, I have found that, unfrotunately, a certain segment of the developer species needs to focuss more on the quality ethic.

Don't get me wrong: Most of the article submmissions we recieve have well-develoepd, well-presented ideas. But even some of the better ones sometimes have easily aviodable quality prolbems. Being the perfectionist that I am , I'd like to share som common problems I encounter in some articles, plus corresponding areas that might need improvemen in software, such as.

- They got poor grammar and spellling. I'd die of embarrassmment if somone read my writng before they had been though a spellchecker and grammer checker. *(Dont trust a compiler or a testor to catch every coding and design probelm, have soemone do peer reviews .)*
- Sentences are padded. One of the things that is a good idea to loook at is the tendency for some poeple who are writnig articles to use more words than they wuold need to express a thuoght that only needs a few words to be properly expressed before its understood. *(Your code?)*
- Obfuscation through multifarious convolutions. Please, eschew any overtly labyrinthine scrutinizations and prolongations. *(Nobody doesnt care for that artsy-fartsy talk.)*
- There articles are filled with unnecessary and unwanted redundancy. Redundancy can be a huge, gigantic problem, so don't do it—avoid it wherver possible. Just state your point and move on. If you dont simply state you're point and move on, your guilty of being redundant. So I advice against redundency. *(How tight, condenesd, and unrepetitve is you're design and coding? Or is it redudant, overwrought, and not-to-the point?)*
- Sentence fragments. A big problem. No verb, no sentence. Clear? *(code comments, too.)*
- Incomplete or poorly defined ideas. A concept is intoduced or given as an example without clearly explaining *(e.g., bandwidth limitations)*. And even worse is whenn your in the middle of a sentence but it never
- Unacceptabel syntax *AND* word choices!! This *REALLY* bug's me! The thing that, really bug's me most about it, is shouldnt these poeple have taken english in HIGH SCHOOL???!!! *(This syntax stuff is NOT C++ complextiy, guys!!!)* :-)
- Acronym soup. An NISS (National Institue of Studying Stuff) study of NATKBBOC (Not All That Knowedgeable But Bordering On Competent) TWRPS (managers) showd SLUD-oriented confusion when TAD (total acronymn density) of a document I (is) higher than $15,977 MSRP (miles per gallon). So just ovoid it (IT). *(Bottom line: It I a good idea to KO unnessary TAD @ any cost ASAP, OK? Actual milage may very.)*
- Being married to you're words . I subscribe to the old sayng, "Theirs more than one way to skin a cat by the horns." Be humble enough to realize there might be a better way too present you're work, i.e., mine. Rest asurred, if you submit something to the journal, I won't change anythign unless theirs a really, *really* good reason, e.g., I'm bored. *(Hint: If you really cant stand the idea of having others "improve" you're work, Its not hard to weasel you're untouchd work past the "gatekeepers" until its to late to chage it back . BESIDES, who'se going to notice the differnce?!?!)*

– Lorin May

Got an idea for BACKTALK? Send an E-mail to backtalk@stsc1.hill.af.mil