Readers “Thinking Objectively”

The following letters are in response to Mauri Laitinen’s and Mohamed Fayad’s “Thinking Objectively” column (Nov. 1997, p. 125).

While Laitinen and Fayad have many valid points, the essence of their arguments revolves around the application of the Software Engineering Institute Capability Maturity Model (CMM) rather than the model itself and its attendant assessment process as originally conceived. We must remember, however, that there was a dual purpose: the U.S. Department of Defense wanted a way to evaluate the relative risk of choosing Contractor A vs. Contractor B. (Watts Humphrey came up with a way to do this and also provide guidance on process improvement.)

Humphrey’s objective was to provide guidance to the military services in selecting capable software contractors. The resulting method for evaluating their strengths and weaknesses has proved valuable for assessing other software organizations.

Humphrey’s book, Managing the Software Process, describes the technical and managerial topics these assessments have found most critical for improvement. The techniques outlined in these pages are grounded in the durable principles that have fueled several centuries of scientific and engineering advancement. They provide a powerful conceptual framework for learning about and improving the software engineering process.

Many organizations have focussed on “achieving level x” rather than on using the principles of the CMM to guide their process improvement activities and help decide where to best spend their next improvement dollar. That is a danger we must fight in the context of powerful customer pressures to treat the CMM as a goal. On the other hand, if we’re smart, we can use that pressure as an incentive to achieve valuable process improvements that are beneficial from a business perspective.

Dick Waina
Plano, TX

As a practicing software quality manager in the commercial world, I believe the reason—not excuse—why it is so difficult to introduce documented processes is simply that the people we want to follow them don’t enjoy working that way.

You can reason and rationalize all you want, but the fact is people resist what they don’t like, and many of the people we employ to develop software simply don’t like structure or organization or anything they see as a restriction.

They won’t tell you they come to work to do what they like. They won’t argue with your unassailable reasoning, because they can’t. They just don’t want to do what you ask. So they don’t. They’ll offer excuses. And if you overcome the excuses they just find more because you have not tackled the problem.

I don’t have any answers, except maybe hire only people who like working in a structured environment. But will they be the people who have the best skills and insights? Or a few could be fired to encourage the rest, probably forcing all the good ones to leave. Me, I’ll go on working with the willing, struggling to persuade the unwilling and not expecting great success.

Josh Jones
Magill, SA, Australia
Laitinen's and Fayad's points about the use of assessments in immature organizations are quite apt. We find that an audit-like CMM-Based Appraisal for Internet Process Improvement (CBA-IPI) checking each of the activities of level 2 doesn’t really serve the organization. Participants are only too likely to perceive the results as indicating that they do nothing right.

With any assessment, there are two data collection goals: check compliance with a given level and identify where to go from there. Immature organizations usually don’t comply with many activities, so the focus is better placed on what the most important problems are to be tackled. This was the focus of the original SEI assessment technology called Software Process Assessments (SPA), and we find that the newer CBA-IPI doesn’t serve level-1 organizations as well. We generally tailor CBA-IPIs for such clients to make them look more like SPAs.

The general point about the applicability of the CMM to small organizations keeps coming up. The CMM was meant to be tailored. The key misunderstanding is that the CMM is not a list of activities that must be in place. It is a set of goals that a process should achieve, by whatever mechanism works. The goals are intended to prevent certain sets of problems from occurring. If an organization has other ways of preventing such problems that differ from the CMM from occurring, an assessor still considers the goal met and the process requirements satisfied. These are referred to as “alternative practices” in the CBA-IPI method. The point here is that organizations can look at the problems they are facing and look to the CMM for ideas of practices that might prevent them from occurring. They don’t have to put elaborate systems in place just to meet a “CMM checklist.” This mental checklist is something we have to work hard to change. It is worth reiterating that everything in the CMM is designed to either prevent a problem or provide mechanisms for process improvement.

The order of adoption of key process areas (KPAs) and practices is a very contentious point and one of the differences between the CMM and other models. Certainly there are special organizational conditions that could require a higher-level practice or process area sooner. However, in general, the principle of focusing on level-X issues before level X+1 involves the following points: First, the CMM is addressing key process areas and key practices. This does not mean that other areas are not also being performed (software development itself does not appear until level 3). It simply means that focusing scarce resources and even scarcer organizational attention on higher-level practices won’t yield as much benefit as it would if lower-level practices were already in place.

For example, quantitative root-cause analysis (level 5) doesn’t yield big benefits unless done in a quantitatively managed environment (level 4). You can’t do quantitative management unless you have consistent data (level 3). Your data won’t be very meaningful if you don’t have the discipline to carry out a repeatable process (level 2). Again, this is not to say you aren’t doing some measurement, analysis, and defect prevention at lower levels. It just means this is not a big bang-for-buck area compared to putting in place basic project management discipline. You’ll get much more benefit out of tackling the level-2 issues before the level-3 issues, level-3 before level-4, and so on.

I don’t have systematic academic studies to back this up. However, many clients have come to us, and, through CMM training and coaching, realized they were trying to juggle too many processes at different levels, and that was why they couldn’t seem to make progress with any one process. Also, when we do assessments, we don’t find level-1 organizations mentioning problems that relate to level 3 or higher. Similarly, level-2 organizations don’t mention problems relating to level 4 or higher. Again, assessment (and improvement) is related to solving/preventing problems, and the order implied in the CMM is based on what was seen during assessments when the CMM was being developed.

I have several other comments I’ll condense down into the following points (the expanded version can be found at www.processinc.com/fayad_reply.html).

- Level 2 is not a “negative designation.” We still don’t see many level 2s around. It is a big cultural jump that takes time and effort and can’t reasonably be divided into smaller chunks.
- A key contribution of the CMM is to show that there are several levels. Although each process area in any given level is “simple,” it is not “easy” and
expectations should be set accordingly.

• Yes, assessment findings can be difficult to address, and yes, improvement can take time and money. Neither of these is an argument that they are not worth doing. Nor do they indicate a failure of either the model or the organization.

• The scarcity of data is not an indication that the existing data is false or misleading. In the absence of data contradicting the SEI, we use what we have so far and collect more.

Laitinen and Fayad point out that “smaller organizations cannot afford the two- or three-year duration it normally takes to reach CMM level 3.” It is not a matter of affording it. It is simply how long it takes to get there, if that is where you want to go.

The authors suggest organizations use assessment to evaluate how projects are done. Please keep in mind this is the data collection role of assessment and misses the critical consensus-building role. Without organizational consensus as to which problems to address first, improvement is much more difficult. I heartily agree with their final points:

• Process improvement must be tailored
• Models must be used intelligently, not slavishly
• Focus should be on solutions, not processes for their own sake

David Constant
Pittsburgh, PA

Laitinen and Fayad write, “Assessments are very expensive, and in an immature organization the results are likely to be meaningless.” This is not true. Assessments are not expensive either in terms of money or the amount of time required of the assessed organization. I don’t know what data was used as a basis for this, but the data from the 35 assessments I have led doesn’t jive with this statement.

Someone has seriously misled Laitinen and Fayad about the application of the CMM to different organizational situations. Expert assessors know how to tailor the model to the situation. The suggested practices are not a check list or an idealized set to measure against. The spirit of the CMM lives in the KPA purposes and goal statements. Alternate practices to those listed in the CMM are legitimate ways to accomplish the goals. The organization has broad freedom in deciding how to accomplish the goals.

The authors write, “In the decade since the CMM was first published, most development organizations are still at level 1… most of the world’s best known commercial software is produced by organizations at or below level 3.”

Although I agree that better differentiation of organizations at level 1 is needed (levels 0 and -1 have been described and proposed before), I am curious as to how the authors know that most development organizations are still at level 1 (perhaps those in the SEI’s database are, but is that representative of all organizations?). And if it is true that the best known software is produced by level-3 and below organizations, then maybe that explains why we have so much subpar software today. Most of the world’s commercial software organizations struggle to get their software products out on time and would love to reduce their cost of software quality. They and we would benefit from more mature practices.

The authors add, “What is the value of attaining CMM level 5? … getting there is so difficult and rare, does it have meaning for most development organizations?” How do the authors know how difficult it is to get to level 5? Have they worked in or helped grow a level-5 capability in an organization? Do they have the data to support this claim of difficulty? Also, the incentive for achieving higher levels of maturity is not getting contracts, although that may be a direct result of consistently producing higher-quality software on time and under budget. The evidence is that high-maturity organizations come closer to assuring that life-critical or business-critical software works under almost any condition.

As for the value of moving beyond level 3, it is true levels 4 and 5 are undergoing significant redefinition in CMM V2.0 based on what the SEI has learned since V1.1. Nonetheless, those organizations now at those levels have no doubt of the added value of getting there. Reported benefits of moving beyond level 3 include:

• Better predictability
• Improved product quality
• Much higher process quality
• Increased customer satisfaction
• Increased job satisfaction
• Reduced cycle time
• Increased levels of reuse
Have the authors looked at the measured results of these leaders lately? If not, they should check out my latest article, “Accumulating the Body of Evidence for the Payoff of SPI,” at www.utexas.edu/coe/sqi/archive.

HBRB KRASNER
Austin, TX

Response

We reiterate that software process improvement (SPI) is essential to any serious development organization. The CMM (and other improvement frameworks) are useful in providing guidelines for improvement, and assessment, done at the proper time, is a key element of improvement.

Our concerns revolve around the model’s use, the meaning of its levels, especially in industry, the value of assessment for low-maturity groups, the value of the CMM after a decade in action, and whether the CMM has universal applicability.

We believe that SPI cannot usefully be separated from organizational goals. Paulk [3] says, “It is important to remember that the improvement effort operates inside the larger context of a business.” SPI conducted without goals for improved performance yields limited, or even negative, results. In other words, SPI should not be an end in itself. Without appropriate business goals, a CMM level-4 company may produce a poorer product than a focused level-1 company. A number of letters indicate that it is the CMM’s use rather than the model that is problematic. Specifically, they state that organizations focus on achieving level “x” rather than focusing on the “spirit” of the CMM and that the CMM is wrongly used as a checklist.

When the U.S. Department of Defense decided to start mandating CMM level 3 as a minimum requirement, the goal shifted from SPI to achieving the necessary level, and the CMM’s elements became a checklist to meet. It doesn’t really matter that this differed from the CMM’s original intent, and it is unfair to blame companies that strive to attain the CMM level as their primary goal. Immediate economic priorities take precedence over longer-term initiatives with less well-defined economic benefit. In such an environment, it makes more sense to use the CMM as a checklist than as a guide and risk not meeting level 3. It also tended to change the idea of tailoring. Instead of an honest evaluation of strengths and weaknesses, the emphasis shifted to finding some way to meet the guidelines. We can lament this, but it is unlikely that the trend will reverse.

Another problem with the CMM’s use is the widespread misunderstanding of CMM levels. Paulk [3] states that one company’s documented estimation process might be “Go ask George,” while another uses a more analytical approach. Both processes address the KPA but work differently, have different bases for their validity, and scale up very differently. This means a company can get to at least CMM level 3 through defined, but theoretically shaky, processes. Moreover, attainment of a certain CMM level does not confer domain competence. A level-4 defense contractor is unlikely to produce a more suitable spreadsheet program than a level-1 commercial software company. Unfortunately, mandating level 3 causes some to read more into the designation than is actually there.

This may explain why the CMM has not been as widely adopted outside the defense industry and why so few companies have moved up the scale. Other economic priorities seem to give a better return on investment. We hope this question will be investigated further.

Specific Concerns

Dick Waina says that Humphrey’s book, Managing the Software Process, describes techniques that “are grounded in the durable principles that have fueled several centuries of scientific and engineering advancement.” We think this is overstated. The value of the book is undeniable, but as long as people are involved, management will remain more art than science, and key practices, unlike Newton’s laws, will fundamentally change as industry and the workforce evolve.

We agree with Josh Jones that people often resist the documentation efforts required in meaningful process improvement, but the resistance, we think, stems most often from the way processes are implemented. It is easy to blame the frequent failure of improvement programs on the people who deal with them daily, but change programs are too often applied like patent medicine—without clear goals, training, and fundamental work practice revision. People expected to take on new tasks justifiably expect to see an improvement accruing to them. Just adding key processes to an already stressed development group is
bound to generate resistance.

David Constant’s point that low-maturity organizations find their efforts best used in improvements appropriate to their level is well taken. We agree that CMM level 2 should not be taken as a negative. However, if level 3 becomes a minimum qualification for contract awards, then level 2 implies failure to meet an entry-level standard. While we agree with Constant that the scarcity of data with respect to CMM successes does not imply failure, neither does it imply success. The simple fact that so few companies have achieved even level 3 suggests the difficulty of moving up does not outweigh the perception of the model’s value. We hope more investigation will be done.

Herb Krasner disagrees with everyone, it seems—the SEI, other respondents, and us—about the time and effort required to assess and advance. It is unfortunate that he provides no data to support his disagreement. Perhaps Krasner has a different view of what is expensive. A typical five-day assessment might include six employees at various pay and responsibility levels, plus a qualified assessor. That works out to 240 employee hours, and the billing cost alone exceeds $30,000. This figure also ignores assessor’s fee, indirect costs, and opportunities lost. For a large company, the cost is repeated for each division or subdivision. Justifying the ROI does not eliminate this significant up-front expense.

There is no evidence, anecdotal or otherwise, to suggest, as Krasner does, that the SEI has significantly underestimated the maturity level of software development organizations. Attaining level 5 is still so rare as to make news. According to the Business Software Alliance [1], the software industry is comprised of 40,000 companies, and we assume this group’s members have the greatest interest in using CMM to improve their competitive position. This has not happened. It is also important to know that most, if not all, defense contractors are not categorized as part of the software industry. Unless we accept Krasner’s outlandish implication that the industry is filled with level-5 groups that have either not realized how good they are or are too shy to publicize their achievement, then we repeat: There are only a handful of CMM level-5 organizations after a decade of the CMM.

Krasner’s article [3] merely rehashes already published data about those few organizations that have achieved CMM level 5 and cites references describing a number of large defense contractors’ SPI experiences. We agree that SPI is important, but he does not discuss the payoff, if any, for small companies moving beyond level 3. Note also that there are many more small companies than large ones in software. Krasner’s article contains misleading findings based on only a few data points. For example, he indicates that ROI of SPI is on average 7 to 1 in two years. This is an unbelievable return. If the ROI of SPI is that good, why doesn’t everyone follow it. This sounds like a get-rich-quick infomercial.

(Our expanded response can be found at www.cs.unr.edu/~fayad/process.improve/index.html.)

MOHAMED FAYAD
Reno, NV
MAURI LAITINEN
Lake Tahoe, CA

REFERENCES

DO YOU HAVE A QUESTION ABOUT MEMBERSHIP?

Contact ACM:
ACM Headquarters
1515 Broadway
New York, NY 10036 USA
Voice: +1-212-626-0500
Fax: +1-212-944-1318
Email: orders@acm.org

ACM European Service Centre
108 Cowley Road
Oxford, OX4 1JF, UK
Voice: +44-1865-382-338
Fax: +44-1865-381-338
Email: acm_europe@acm.org

ACM URL: www.acm.org