Software Quality and the Capability Maturity Mo

bout the time Fred Brooks was warning us there was not likely to be a single, "silver bullet" solution to the essential difficulties of developing software [3], Watts Humphrey and others at the Software Engineering Institute (SEI) were busy putting together the set of ideas that was to become the Capability Maturity Model (CMM) for Software.[1] The CMM adopted the opposite of the quick-fix silver bullet philosophy. It was intended to be a coherent, ordered set of incremental improvements, all having experienced success in the field, packaged into a roadmap that showed how effective practices could be built on one another in a logical progression (see "The Capability Maturity Model for Software" sidebar). Far from a quick fix, it was

[1]CMM and Capability Maturity Model are service marks of Carnegie Mellon University.

CMM has had an enormous impact on the practice of software engineering. There is now substantial evidence of the business benefits of CMM-based software and a growing understanding of the factors that contribute to a successful improvement effort.

del

expected that the improvements would take considerable time and effort to put into place and would usually require a major shift in culture and attitudes.

Judging by its acceptance in the software industry, the CMM has already been a major success. It has spread far beyond its origins in military avionics applications, and is now used by major organizations in every sector of the economy around the globe (see box "Adoption of the CMM: A Growing Phenomenon" sidebar). While we have no accurate estimates of its penetration in the global industry, based on what we do know it surely includes thousands of organizations, and the resources expended on CMM-based software process improvement (SPI) are certainly in the billions of dollars.

However, the CMM is not without its critics [1]. It is sometimes claimed that adopting the CMM encourages too much bureaucracy,

James Herbsleb, David Zubrow, Dennis Goldenson, Will Hayes, and Mark Paulk

The Capability Maturity Model for Software (CMM or SW-CMM) is a reference model for appraising software process maturity and a normative model for helping software organizations progress along an evolutionary path from ad hoc, chaotic processes to mature, disciplined software processes. The CMM is organized into five maturity levels as shown in Box 1.

Except for Level 1, each maturity level is decomposed into several key process areas that indicate the areas an organization should focus on to improve its software process. These "vital few" areas are listed as shown in Box 2.

The rating components of the CMM, for the purpose of assessing an organization's process maturity, are its maturity levels, key process areas, and their goals. Each key process area is further described by informative components: key practices, subpractices, and examples. The key practices describe the infrastructure and activities that contribute most to the effective implementation and institutionalization of the key process area.

CMM Level	Major Characteristics
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. Projects use an approved, tailored version of the organization's standard software process(es) 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 facilitated by quantitative feedback from the process and from piloting inovative ideas and technologies.

Box 1.

CMM Level	Focus	Key Process Areas
1 Initial	Competent people and heroics	
2 Repeatable	Project management processes	Requirements management Software project planning Software project tracking and oversight Software subcontract management Software quality assurance Software configuration management
3 Defined	Engineering processes and organizational support	Organization process focus Organization process definition Training program Integrated software management Software product engineering Intergroup coordination Peer reviews
4 Managed	Product and process quality	Quantitative process management Software quality management
5 Optimizing	Continuous process improvement	Defect prevention Technology change management Process change management

Box 2.

or that the CMM is incomplete or flawed. This debate is partly concerned with scope, policy issues, and conceptual questions (such as whether the model harmonizes appropriately with international standards such as ISO-9000). But the debate also focuses on the supposed consequences of adopting the CMM as the basis for SPI efforts. Will the organization get bogged down in red tape or suffer other damage, or will it benefit and show improved performance?

Within the last several years, a significant evolution has taken place as the debate has evolved into a more scientific investigation. Many of the most important questions about the CMM can be

addressed by careful collection and analysis of data, rather than the exchange of rhetoric and undocumented anecdotes that has often characterized this sort of discussion in the past. In this article, we will present the results to date of the SEI's efforts to test critical claims and assertions about the CMM. The effort is still under way, but we believe the current findings are significant. While we will focus on our own efforts, we will also briefly discuss other relevant work.

Claims about the CMM

In order to organize our CMM studies, we worked extensively with users and potential users of the CMM to identify the questions of greatest practical concern. At a high level, the most pressing issues are:

- Process maturity: How long does it take, how much does it cost, and how will it benefit the business?
- What are the factors that influence the success and failure of CMM-based SPI?
- Is the CMM an appropriate framework for guiding improvements in a way that can be understood and applied to the full variety of software organizations?

We have completed three studies to date. Each employed a different approach and different data sources. Through the use of multiple studies and methods, we reduced our vulnerability to the inevitable weaknesses of any single effort.

Multiple-case study. When we began this effort, several case studies had already been published [such as 4, 5, 9, 12]. These studies showed dramatic improvements in such important organizational performance parameters as productivity, reduction of rework, and improvements in cycle time. In our initial effort [7], we tried to locate any additional existing data of this sort that organizations undergoing SPI might already have available. We eventually received usable data from 13 organizations. In order for us to consider the data usable, we had to understand the data collection and analysis process well enough to have a reasonable degree of confidence the data point was meaningful, and ensure the organization was engaged in a CMM-based SPI effort, which appeared to be causing these results.

We presented these results as changes over time[2] within the organization, in order to avoid comparing results from different organizations, which typically defined data in very different ways.

After-the-appraisal survey. The case study evidence (from both the previously published case studies and our own multiple-case study) has several inherent limitations:

- Are these cases typical, or are we only studying a select group of success stories?
- In most cases, only a few types of data are reported from each organization, leaving open the possibility that the organization traded off other performance dimensions (like quality for cycle time) to get these results.
- Did CMM-based SPI cause the improvements in performance, or merely coincide with them?

We undertook the survey described in this section specifically to address these shortcomings. In order to address the first two concerns, we needed to look more broadly across organizations using the CMM, and try to get some small but comprehensive set of performance indicators from them. A survey is an effective tool for this purpose. The third concern—establishing the causal connection between process improvement and performance—must be addressed by accumulating evidence from a number of different studies, using different methods and relying on different assumptions. The survey provides a cross-sectional view of organizations with a wide range of characteristics, and hence provides a good complement to the longitudinal case studies of a few successful organizations.

The goals of this survey [8] were to find out what typically happens to SPI efforts after assessments, to learn as much as possible about the reasons for success or failure, and to see if the performance reported by more mature organizations is, in fact, superior to the performance reported by less mature organizations.

We used our database, which contained over 450 assessments at that time, to select appraisals conducted no less that one year ago (so there was time for change to take place) and no more than three years ago (so we could find people able to give good

---

[2]Our basic analysis was percentage change per year, which is much like a compound interest rate. A threefold increase over three years, for example, works out to about a 44% increase per year for three years.

accounts of what happened after the appraisals). In order to get a broad and balanced perspective, we decided to try to contact a senior technical person and a project manager as well as a member of the software engineering process group (SEPG) for each appraisal.

All told, we were able to obtain contact information for about 167 individuals representing 61 assessments. Of the 167 questionnaires we sent out, we received completed and usable data from 138 of them, for a return rate of 83%. We also succeeded in obtaining responses from individuals in several roles. Of the 138 questionnaires returned, 47 were from senior members of the technical staff, 47 were from project managers, and 44 were from members of an SEPG. Interestingly, and perhaps surprisingly, we found no systematic differences among the responses of these three groups.

Appraisals from reassessed organizations. Since the SEI developed the Software Process Assessment (SPA) method, it has been collecting SPA results from organizations using the method. The results include the maturity level of the organization, the identified process strengths and weaknesses, the organizational scope of the assessment, and the date the SPA was conducted. In this study [6], we used the information in our SPA database to address these two questions:

• How long does it take for an organization to move up a maturity level?
• What are the process challenges that distinguish those who move from the initial level (Level 1) to the repeatable level (Level 2) and those who remain at the initial level?

To address these questions, we focused on organizations that have undergone multiple SPAs. This allowed us to investigate the experiences and changes in individual organizations. From the database housing the SPA results, we extracted the data for 48 organizations that had conducted two or more SPAs. As a group, these organizations have conducted 104 assessments. To address the first question, we looked at the elapsed time between assessments in those cases where organizations moved up in level on a subsequent assessment. To address the second question, we categorized the "weakness" findings according to which key process area (KPA) they served,[3] and compared the weaknesses in organizations that improved their maturity levels with those that did not (see CMM sidebar).

Process maturity: How long does it take, how much does it cost, and how will it benefit the business?

The CMM is best regarded as a tool to be used to pursue an organization's business goals. So it is extremely important to determine the time and effort that must be invested as well as the effects of SPI on organizational performance.

An examination of reassessments shows the median time between assessments (where organizations have moved up on a subsequent assessment) is about two years (see Figure 1). No doubt this interval is, in part, a reflection of the "common wisdom" about timing of assessments, a frequent recommendation being 1.5 to 2.5 years. Only about 25% move from Level 1 to Level 2 in 21 months or less, and about 25% take 37 months or more. Moving from Level 2 to Level 3 appears to be a little quicker, with 25% moving up in 17 months or less and 25% taking 31 months or more.
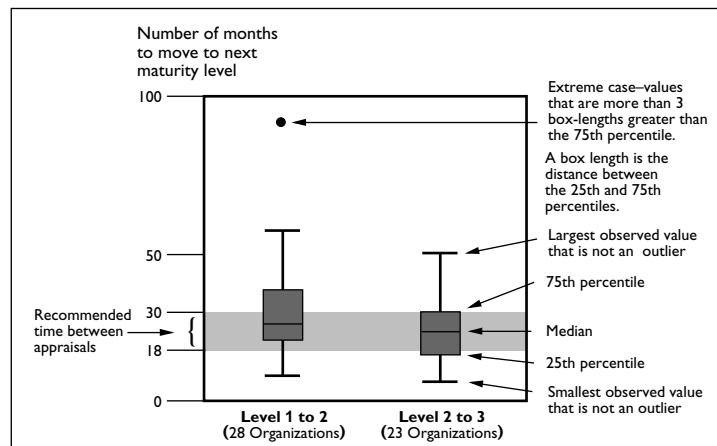

Figure 1. Time to move from Level 1 to Level 2, and from Level 2 to Level 3. The medians are 26.5 months for Level 1 to Level 2, and 24 months for Level 2 to Level 3. (This data is updated regularly; at press time, this summary is accurate.)

We have only a few data points about the actual cost of a SPI program, all of which came from our multiple-case study. We normalized the cost by the number of software engineers in the organization. The range was $490 to $2,004 per software engineer per year, with a median figure of $1,375. What was included in these costs varied somewhat depending on the accounting practices of each organization, but

---

[3]We established the reliability and validity of this categorization scheme which is described in [6].

in general it included the cost of any assessments, any CMM-related training, and the cost of staffing the SEPG.

The cost and time required for a SPI program apparently exceeded the expectations of many people. In our survey we found that slightly over three-fourths (77%) agreed or strongly agreed that SPI "took longer than expected," and slightly over two-thirds (68%) said it "cost more than expected."

Table 1. Summary of case study performance results [7].

Category	Range	Median	Data pts.
Productivity gain/year	9%–67%	35%	4
Time to market (reduction/year)	15%–23%	—	2
Post-release defects (reduction/year)	10%–94%	39%	5
Business value ratio	4.0–8.8:1	5.0:1	5

Almost half (49%) said there was "lots of disillusionment over lack of progress." CMM-based SPI is not a cheap nor a quick fix.

It is difficult to tell from these results alone whether the time and cost is exceeding expectations because the actual numbers are high relative to the benefit or because the organization had little information or experience with which to set realistic expectations.

The results from our multiple case study are consistent with those from previous case studies in organizations such as Hughes Aircraft [9], Raytheon [5], Schlumberger [12], and Tinker AFB [4]. For all the data points that satisfied our criteria, organizations engaged in CMM-based SPI tended to improve substantially in quality, cycle time, and productivity. The business value ratios (benefits divided by cost) were also substantially above 1. Table 1 shows a summary of the data we reported.

Our survey, which allowed us to look at a much broader sample of software organizations, gave us similarly encouraging results. We asked our respondents to rate their organization's performance on a number of dimensions, such as ability to meet schedules, ability to stay within budgets, product quality, and so on. For each of these dimensions, they rated their organization's performance as "excellent," "good," "fair," or "poor."[4] We combined the percentages of excellent and good responses then cross tabulated these responses with the organization's maturity level to produce the results shown in Figure 2.

With the exception of customer satisfaction, all these comparisons show improved performance with increased maturity level. These differences are statistically significant for all but "ability to meet budget." Ratings of customer satisfaction show a dip from Level 1 to Level 2, before reaching 100% good or excellent at Level 3. While this pattern is statistically significant, that is, different from a horizontal line, the difference between levels 1 and 2 is not sig-
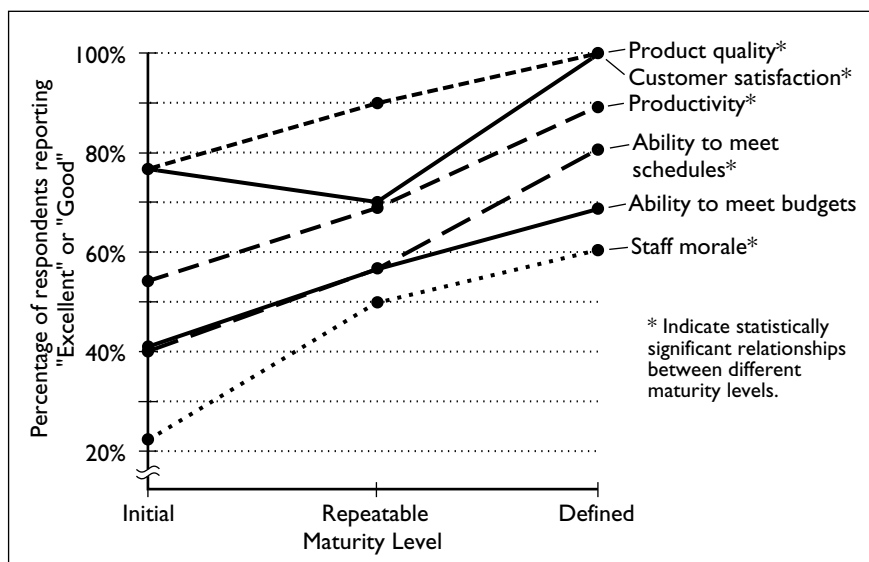

Figure 2. The percentage of respondents who reported that their organization had "excellent" or "good" performance in each area. The asterisks indicate statistically significant relationships between different maturity levels.

nificant. This means we are not able to tell whether this is a genuine difference or if it is just noise in the data. We hope future studies will sort this out. There have been several plausible suggestions about why this dip might occur, including the possibilities that

---

[4]We'd like to be as clear as we can about what these numbers do and do not mean. We do not know the actual bases for the respondents' answers. All we need to assume, however, is that if 138 people are asked "How good is your ability to meet schedule?" those answering it is "good" or "excellent" are, on average, better able to meet schedules than are those who answer their ability is "fair" or "poor."

some customers do not initially like the discipline that requirements management brings to customer interactions, and that customers suffer as attention is focused internally as SPI gets under way.

In addition to these studies by the SEI, an elegant study of software produced under contract for the U.S. Air Force was recently published [11]. Their results also indicate the ability to meet schedules and stay within budget was substantially better in higher maturity organizations. In another study, Krishnan [10] examined the relationship of CMM process maturity and software quality and cost. His sophisticated statistical analysis of data from a large software development laboratory in a Fortune 100 company showed that process maturity significantly increased quality, but did not show evidence of a direct effect on cost. Higher product quality, however, significantly reduced both development and support cost, so to the extent that process maturity increased quality, it

may also have indirectly decreased cost.

Note that virtually all the reports on the benefits of process maturity come from comparisons among organizations at the *initial, repeatable,* and *defined* levels, or from observations of organizations over time as they move through these three stages. Very little is known at this point about the benefits of the higher maturity levels, since there has been relatively little experience to draw on.

Criticisms of the CMM

As we mentioned, several criticisms have been made of CMM-based improvement. The ones that seem to have received the most attention are that CMM-based SPI will be counterproductive, will cause the organization to neglect important non-CMM issues, to its detriment, and will cause the organization to become rigid and bureaucratic, making it more difficult to find creative solutions to technical problems.

The CMM was originally developed to assist the U.S. Department of Defense (DoD) in software acquisition. The rationale was to include likely contractor performance as a factor in contract awards. The model for determining likely contract performance also became a guide or framework for software process improvement. DoD contractors quickly learned they needed to mold and guide their organizations to become more aligned with the CMM if they were to be successful in winning DoD contracts.

This focus is clearly seen in the data on the number and proportion of assessments conducted by DoD contractors in the period from 1987 through 1992. Most, if not all, of the major DoD contractors began CMM-based software process improvement initiatives as they vied for DoD contracts.

But the CMM has not remained a "stick" in the DoD contracting

community. Through the efforts of the SEI to obtain broad participation in the development and improvement of the CMM, the model gained visibility in the wider software engineering community. Gradually, commercial organizations began to adopt the CMM as a framework for their own internal improvement initiatives. In 1989, only a handful of commercial organizations conducted software process assessments, but each year since 1993, commercial organizations have performed more assessments than all DoD and other Federal contractors combined. Furthermore, since 1994, several case studies of the impact of using the CMM in commercial organizations have appeared, including Motorola, Schlumberger, Bull HN, and Siemens.

Many of these companies are multinational in scope. As they acknowledged the benefits of software process improvement in their U.S. sites, they sought to apply this improvement strategy to their sites around the world. Furthermore, many organizations outside of the U.S. compete in markets with U.S. organizations and were stimulated

to adopt a CMM approach to improvement through competitive pressure.

Today we see growing indications of the global adoption of the CMM. One indicator of the adoption is the number of CMM-based software process assessments conducted outside of the U.S. There has been a steady increase in this number over the past few years. According to the most recent software process maturity profile, 17% of the assessments on file at the SEI were conducted at sites located outside of the U.S.

Additionally, the number of Software Process Improvement Network (SPIN) groups outside of the U.S. is on the rise. SPINs are local organizations whose members have an interest in software process improvement. Twenty-six such groups have been established outside of the U.S. and can be found in Australia, Europe, Asia, and South and North America. They were established to facilitate communication of field experiences and lessons learned among champions and practitioners of software process improvement.

In our survey, we asked about whether any of these performance problems had actually occurred. In each case, the overwhelming majority of respondents (84% to 96%) disagreed or strongly disagreed that they had experienced the problem (see Table 2). These concerns appear to be misplaced for all but a relatively few organizations.

Another criticism occasionally made is that CMM-based SPI causes organizations to become

Table 2. Percentage of respondents that disagreed or strongly disagreed that CMM-based SPI caused these problems with organizational performance.

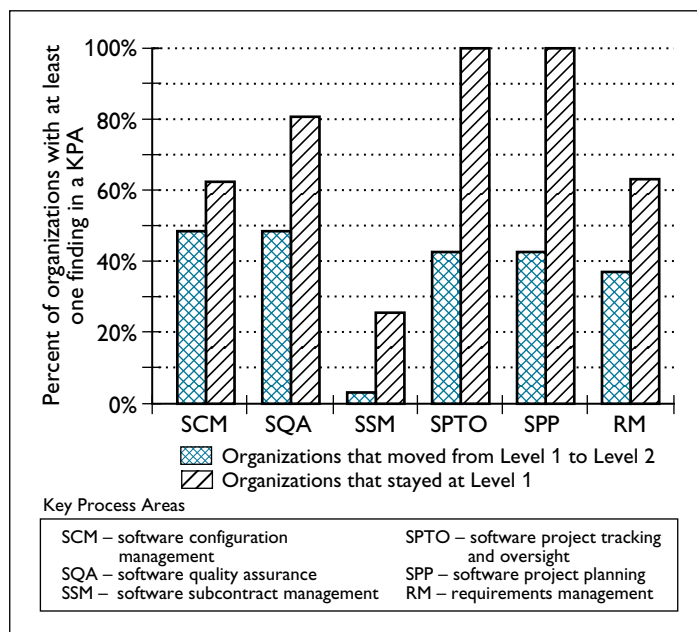| Question	Disagree or strongly disagree
SPI was counterproductive	96%
Neglect non-CMM issues	90%
Became more rigid and bureaucratic	84%

Figure 3. Percentage of organizations that had at least one finding in each of the Level 2 key process areas. (These are results from a second assessment.) All of the organizations were originally assessed at Level 1.

risk-averse. The argument apparently is that mature organizations will not pursue risky (but potentially high-payoff) projects for fear of "losing their maturity rating." On the other hand, one might argue that if process maturity lowers the level of risk on typical projects, the organization can more easily add high-risk projects to its portfolio. We have data from our survey that bear on this

issue. One question asked "How much risk is management generally willing to take?" In Level 1 organizations, only 42% responded "substantial" or "moderate" (the other choices were "some" and "little if any"). The figure for Level 2 organizations was 74%, and it rose to 79% in Level 3 organizations. This difference is statistically significant. This data indicates that people from higher maturity organizations report their managers are more willing, not less willing, to take risk.

The time and cost of a CMM-based SPI program often exceeds the expectations of those involved. However, substantial evidence has now accumulated that software process maturity, as defined by the CMM, has significant business benefits. A number of case studies, two correlational studies, and the survey we reported here all point toward this conclusion. There is also little evidence to suggest that using the CMM leads to the adverse effects predicted by its critics.

What are the factors that influence the success and failure of CMM-based SPI?

Clearly, not every organization that has attempted process improvement has succeeded. It is very important to learn more about what it takes to succeed so that more organizations can reap the benefits earlier.

In our analysis of reassessments, we examined the weaknesses that were most typical of organizations that were initially assessed at Level 1, then assessed again at Level 1 on a subsequent assessment. If we contrast these weaknesses with those found in organizations that succeeded in achieving Level 2, we can see the areas in which these organizations seemed to have the most difficulty.

As Figure 3 shows, the organizations not moving up were more likely to have a finding in each of the Level 2 KPAs. But the largest differences are in the areas of planning and tracking software projects. Every organization that failed to move up to Level 2 had a finding in both of these areas. This strongly suggests that these areas are either the most neglected or are the most difficult types of practices to put in place, or both.

Our survey revealed several problems which are encountered frequently in SPI efforts. Two of them are probably very general problems with organizational change efforts. Of the respondents, 42%

agreed or strongly agreed with statements that SPI had "been overcome by events and crises;" 72% agreed or strongly agreed that it has "often suffered due to time and resource limitations."

Another frequent problem stems from the characteristics of the CMM itself. Two thirds of the respondents agreed with the statement, "We understood what needed to be improved, but we needed more guidance about how to improve it." Similarly, over half agreed that they needed more mentoring and assistance. We had heard anecdotal evidence of these types of problems before, but the survey gave us a better sense of how widespread they really are. What is needed is clear, practical guidance on how to introduce the CMM into a software organization. This is currently being addressed in several ways at the SEI.[5]

In order to investigate the overall success rate of CMM-based SPI, we included a question on our survey which simply asked: "How successfully have the findings and recommendations of the assessment been addressed?" The distribution of responses is shown in Figure 4.

These results clearly indicate that success is not guaranteed, and that it is very important to learn about factors that distinguish the successes from the failures.

Lessons Learned

In our multiple case study, we identified a number of lessons learned by successful organizations. Many of these lessons are factors identified by those involved in the SPI effort as critical to the effort's success. The factors most often identified as important were the following:

- The SPI effort requires visible support and commitment from senior management.
- Middle management support is important and often hard to get because they have major project responsibility and often no additional resources for process improvement.
- Grassroots support and involvement is also extremely important.

- Obtaining observable results, backed up with data if possible, is important early on to keep the effort visible, and to motivate and sustain interest.
- The process improvement effort must be planned, managed, and given sufficient dedicated resources.
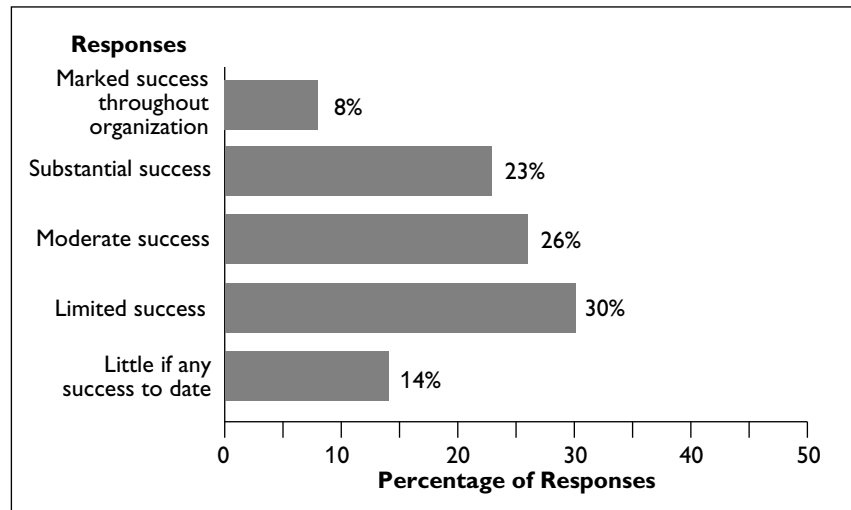- The SPI effort must serve business interests and

Figure 4. Distribution of responses to question about the degree of success in addressing the assessment findings and recommendations. (The numbers do not add up to 100 because of rounding error.)

must be coordinated with other parts of the business in order to have the necessary foundation for the cultural change required by successful SPI.

In our survey, we were able to examine success factors in a more systematic way. We asked a number of questions about characteristics of the organization and of the SPI effort, and identified a number of characteristics associated with successful and with unsuccessful efforts. There is considerable agreement with the more informally developed lessons learned. Highly successful efforts tended to have the following characteristics:[6]

- Senior management actively monitors SPI progress
- Clearly stated, well understood SPI goals
- Staff time/resources dedicated to process improvement
- Clear, compensated assignment of responsibility
- SEPG staffed by highly respected people
- Technical staff is involved in improvement

On the other hand, agreement with the following was associated with less successful SPI efforts:

---

[5]See the SEI web site for more information, http://www.sei.cmu.edu.
[6]Agreement with these statements was statistically associated (at the .05 level) with the higher categories of "success" as shown in Figure 4.

- High levels of "organizational politics"
- Turf guarding
- Cynicism from previous unsuccessful improvement experiences
- Belief that SPI "gets in the way of real work"
- Need more guidance on how to improve, not just what to improve

As these studies suggest, a number of factors appear to be associated with success or failure of a process improvement effort. In order to get off the ground, particular attention should be given to planning and tracking projects, an area that seems to be holding many Level 1 organizations back from achieving Level 2. There are several factors under management control that also appear to be critical to success, including active monitoring, giving the effort adequate resources, and staffing it with highly respected people. Participation and buy-in at all levels, including middle management and technical staff, is also very important. Showing concrete results quickly may help with this. Organizations that have dysfunctional attitudes such as turf guarding, internal political contention, and cynicism about the effort are going to have a more difficult time. There is also a tendency for SPI programs to be starved for resources and to be overcome by events. A common problem, which has not yet been adequately addressed, is the need for more guidance on how to go about making the improvements.

Is the CMM an appropriate framework for guiding improvements in a way that can be understood and applied to the full variety of software organizations?

It is sometimes suggested that some features of the CMM are inappropriate for organizations that differ substantially from the large-project defense avionics environment for which the CMM was originally developed. In particular, it is often suggested that small organizations [2] and commercial companies may find the CMM less useful or more difficult to apply.

In our survey, we were able to compare success rates of organizations of various sizes operating in different sectors in order to see if these factors played a major role in determining success. Most organizations in the survey were in the commercial (23), government contractor (19), or government (12) sectors, and our results show no systematic differences in the success rates among these sectors.

We also had a number of sizes of organizations represented. The smallest 25% had fewer than 54 software engineers, while the largest 25% had 300 or more. Again, there was no systematic difference in success rate due to organizational size. Interestingly, we found that small organizations had fewer of the problems such as organizational politics and turf guarding that appeared to inhibit success.

Despite these findings, there is some limited evidence which suggests it may be more difficult to apply the CMM, or at least parts of the CMM, in small organizations and in commercial organizations. The evidence is from an unpublished survey that we conducted of 84 people who took the SEI's "Introduction to the CMM" course during the period from late 1993 until early 1995. The survey was conducted by mail from late 1995 to early 1996 with a return rate of over 60%. Approximately one to two years passed between the survey and the time the students had completed the course—enough time for them to make informed judgments about its value added in practice.

Figure 5 shows the distribution of responses to the question: "How much of the subject matter that was covered in the course is applicable/relevant to your work?" In all of the organizational size categories, well over 60% answered that much or most of the material was applicable. However, all of the "little, if any" responses came from the two smallest categories of organizations. There is a similar pattern of results, although it did not achieve statistical significance, for how well they have been able to use the material in their organization.
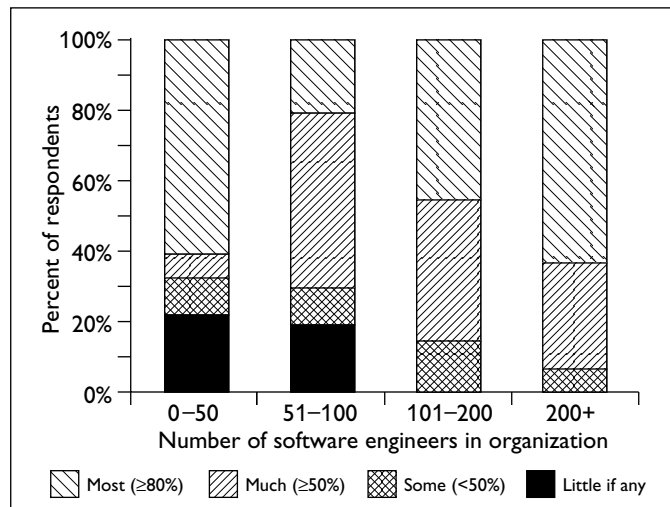

Figure 5. Distribution of responses to question about relevance of CMM course material. The overall pattern of results is statistically significant.

There are similar hints in the data broken down by type of company. The rates for relevance and ability to actually use the material were quite high for each type of organization, again with well over 60% saying "much" or "most" was applicable. As before, these rates are slightly lower (although the differences did not quite achieve statistical significance) for commercial companies than for defense contractors.

In summary, the data to date does not point to any actual differences in success in using the CMM for companies of various sizes and types. There are some hints, however, that small companies and commercial companies may find some of the CMM irrelevant or hard to apply. These differences do not appear to be large.

What do We Know Now About The CMM and Software Quality?

In the past several years the empirical studies of organizations using the CMM— both studies performed by the SEI and by others—have produced significant advances in our understanding of the costs, benefits, problems, and risks of CMM-based SPI. The most broadly supported claim is that CMM-based SPI has substantial business benefits for those moving from the initial through the defined levels. A number of individual case studies, a multiple-case study, a survey, and two correlational studies are quite consistent in showing important organizational performance improvements associated with process maturity. Future work should aim to identify the precise mechanisms that relate process and performance so the existence and nature of the causal relationship can be determined.

Many factors associated with success and failure are fairly well established, since they crop up in many case studies and were also good predictors of successful efforts in the surveyed organizations. The results about how widely the CMM applies to organizations of various sizes and types should still be regarded as tentative.

There are several important areas where there has been very little work to date. There have been no published studies we are aware of on the results of moving to the highest maturity levels, although there have been studies of some of the individual practices included in those levels. There is also a whole set of issues about change, resistance, and institutionalizing new ways of working in software organizations that we need to better understand in order to become more effective at putting innovations into practice. ◼

REFERENCES
1. Bach, J. Enough about process: what we need are heroes. *IEEE Soft.* 12, 2 (1994), 96–98.
2. Brodman, J. G. and Johnson, D. L. What small businesses and small organizations say about the CMM. In *Proceedings of ICSE '94* (Sorrento, Italy, May 16–21).
3. Brooks, F.P. No silver bullet: Essence and accidents of software engineering. *IEEE Comput.* 20, 4 (1987), 10–19.
4. Butler, K.L. The economic benefits of software process improvement. *CrossTalk* (July 1995), 14–17.
5. Dion, R. Process improvement and the corporate balance sheet. IEEE Soft. 10, 4 (July 1993), 28–35.
6. Hayes, W. and Zubrow, D. Moving on up: Data and experience doing CMM-based software process improvement. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA CMU/SEI-95-TR-08, Sept. 1995.
7. Herbsleb, J., Carleton, A., Rozum, J. Siegel, J, and Zubrow, D. Benefits of CMM-based software process improvement: Initial results. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA CMU/SEI-94-TR-13, Aug. 1994.
8. Herbsleb, J.D. and Goldenson, D.R. A systematic survey of CMM experience and results. In *Proceedings of ICSE '96* (Berlin, Mar. 25–30).
9. Humphrey, W.S., Snyder, T.R., and Willis, R.R. Software process improvement at Hughes Aircraft. *IEEE Soft.* 8,4 (July 1991) 11–23.
10. Krishnan, M.S. Cost and Quality Considerations in Software Product Management. Dissertation, Graduate School of Industrial Administration, Carnegie Mellon University, 1996.
11. Lawlis, P.K., Flowe, R.M., and Thordahl, J.B. A correlational study of the CMM and software development performance. *CrossTalk* (Sept. 1995), 21–25.
12. Wohlwend, H and Rosenbaum, S. Schlumberger's software improvement program. *IEEE Trans. Soft. Eng. 20*, 11 (Nov. 1994), 833–839.

JAMES HERBSLEB (herbsleb@research.bell-labs.com) is a member of the technical staff in the Software Production Research Group, Bell Laboratories, Lucent Technologies, Naperville, Ill.
DAVID ZUBROW (dz@sei.cmu.edu) is a member of the technical staff at the Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa.
DENNIS GOLDENSON (dg@sei.cmu.edu) is a member of the technical staff at the Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa.
WILL HAYES (wh@sei.cmu.edu) is a member of the technical staff at the Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa.
MARK PAULK (mcp@sei.cmu.edu) is a member of the technical staff at the Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa.