4   Using the CMM

The CMM establishes a set of public in available criteria describing the characteristics of mature software organizations. These criteria can be used by organizations to improve their processes for developing and maintaining software, or by government or commercial organizations to evaluate the risks of contracting a software project to a particular company.

This chapter focuses on two SEI-developed methods for appraising the maturity of an organization's execution of the software process: software process assessment and software capability evaluation.

> *Software process assessments* are used to determine the state of an organization's current software process, to determine the high-priority software process-related issues facing an organization, and to obtain the organizational support for software process improvement.
>
> *Software capability evaluations* are used to identify contractors who are qualified to perform the software work or to monitor the state of the software process used on an existing software effort.

This overview is not sufficient by itself for readers to conduct either an assessment or evaluation. Anyone wishing to apply the CMM through these methods should request further information on assessment and evaluation training.

The CMM is a common foundation for both software process assessments and software capability evaluations. The purpose of the methods are quite different, however, and there are significant differences in the specific methods used. Both are based on the model and the products derived from it. The CMM, in conjunction with the CMM-based products, enables the methods to be applied reliably and consistently.

4.1 Software Process Assessment and Software Capability Evaluation Methods

Software process assessments focus on identifying improvement priorities within an organization's own software process. Assessment teams use the CMM to guide them in identifying and prioritizing findings. These findings, along with guidance provided by the key practices in the CMM, are used (by a software engineering process group, for example) to plan an improvement strategy for the organization.

Software capability evaluations are focused on identifying the risks associated with a particular project or contract for building high-quality software on schedule and within budget. During the acquisition process, software capability evaluations may be performed on bidders. The findings of the evaluation, as structured by the CMM, may be used to identify the risks in selecting a particular contractor. Evaluations may also be performed on existing contracts to monitor their process performance, with the intent of identifying potential improvements in the software process of the contractor.

The CMM establishes a common frame of reference for performing software process assessments and software capability evaluations. Although the two methods differ in purpose, the methods use the CMM as a foundation for appraising software process maturity. Figure 4.1 provides a summary description of the common steps in assessments and evaluations.
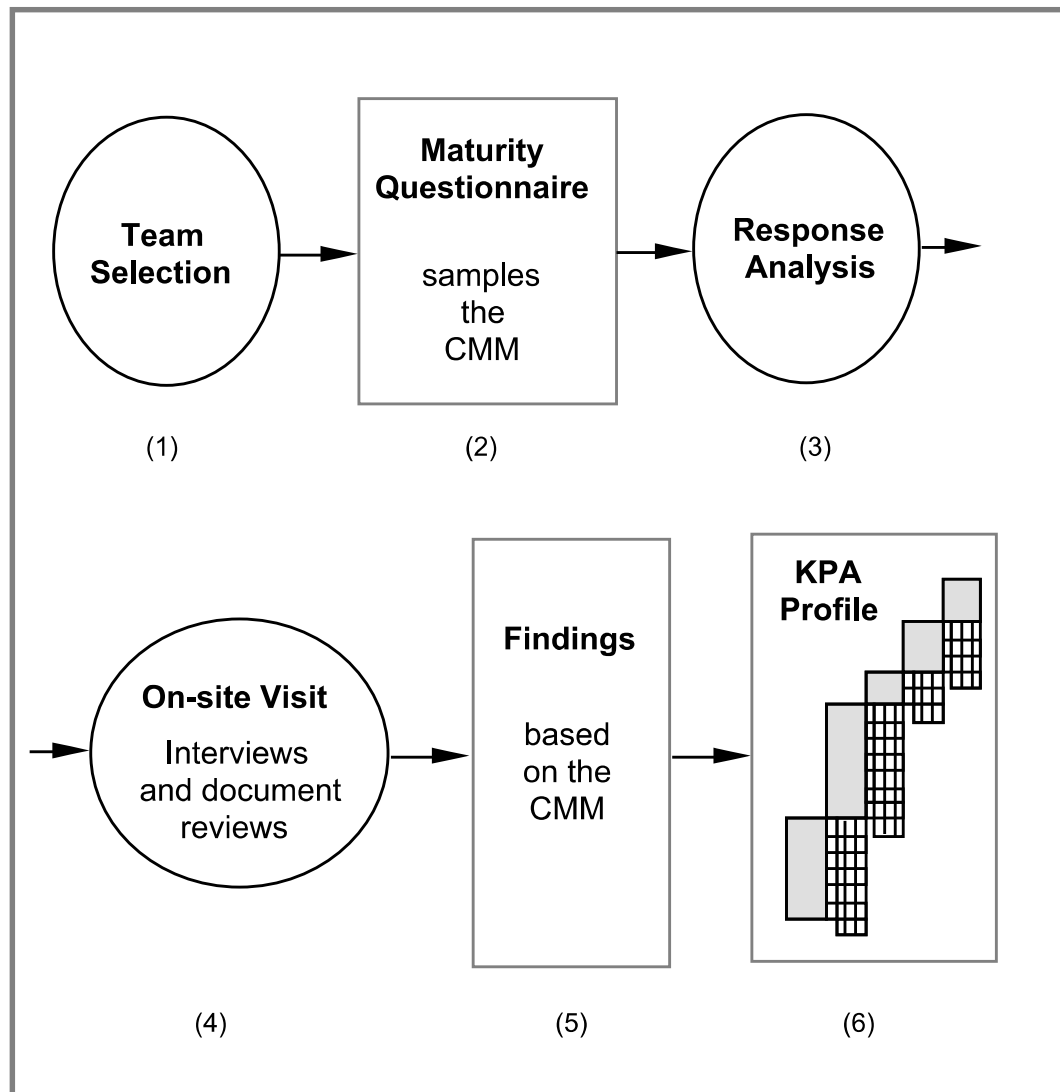
Figure 4.1 **Common Steps in Software Process Assessments and Software Capability Evaluations**

The first step in is to select a team. This team should be trained in the fundamental concepts of the CMM as well as the specifics of the assessment or evaluation method. The members of the team should be professionals knowledgeable in software engineering and management.

The second step is to have representatives from the site to be assessed or evaluated complete the maturity questionnaire and other diagnostic instruments. Once this activity is completed, the assessment or evaluation team performs a response analysis (step 3), which tallies the responses to the questions and identifies those areas where further exploration is warranted. The areas to be investigated correspond to the CMM key process areas.

The team is now ready to visit the site being assessed or evaluated (step 4). Beginning with the results of the response analysis, the team conducts interviews and reviews documentation to gain an understanding of the software process followed by the site. The key process areas and key practices in the CMM guide the team members in questioning, listening, reviewing, and synthesizing the information received from the interviews and documents. The team applies professional judgment in deciding whether the site's implementation of the key process areas satisfies the relevant key process area goals.[6] When there are clear differences between the key practices in the CMM and the site's practices, the team must document its rationale for judging that key process area.

At the end of the on-site period, the team produces a list of findings (step 5) that identifies the strengths and weaknesses of the organization's software process. In a software process assessment, the findings become the basis for recommendations for process improvement; in a software capability evaluation, the findings become part of the risk analysis performed by the acquisition agency.

Finally, the team prepares a key process area profile (step 6) that shows the areas where the organization has, and has not, satisfied the goals of the key process areas. A key process area can be satisfied and still have associated findings, provided the findings do not identify major problems that inhibit achieving any goals of the key process areas.

---

[6] These judgements may have to take place without complete information when company proprietary or security issues may be involved.

In summary, the software process assessment and software capability evaluation methods both:

use the maturity questionnaire as a springboard for the on-site visit,

use the CMM as a map that guides the on-site investigation,

develop findings that identify software process strengths and weaknesses in terms of the key process areas in the CMM,

derive a profile based on an analysis of the satisfaction of the goals within the key process area, and

present their results, to the appropriate audience, in terms of findings and a key process area profile.

4.2 Differences Between Software Process Assessments and Software Capability Evaluations

In spite of these similarities, the results of a software process assessment or software capability evaluation may differ, even on successive applications of the same method. One reason is that the scope of the assessment or evaluation may vary. First, the organization being investigated must be determined. For a large company, several different definitions for "organization" are possible. The scope may be based on common senior management, common geographical location, designation as a profit and loss center, common application domain, or other considerations. Second, even in what appears to be the same organization, the sample of projects selected may affect the scope. A division within a company may be assessed, with the team arriving at findings based on a division-wide scope. Later, a product line in that division may be evaluated, with that team arriving at its findings based on a much narrower scope. Comparison between the results without understanding the context is problematic.

Software process assessments and software capability evaluations differ in motivation, objective, outcome, and ownership of the results. These factors

lead to substantive differences in the dynamics of interviews, the scope of inquiry, the information gathered, and the formulation of the outcome. The assessment and evaluation methods are quite different when the detailed procedures employed are examined. Assessment training does not prepare a team to perform evaluations, or vice versa.

Software process assessments are performed in an open, collaborative environment. Their success depends on a commitment from both management and the professional staff to improve the organization. The objective is to surface problems and help managers and engineers improve their organization. While the questionnaire is valuable in focusing the assessment team on maturity level issues, the emphasis is on structured and unstructured interviews as tools for understanding the organization's software process. Aside from identifying the software process issues facing the organization, the buy-in to improvement, the organization-wide focus on process, and the motivation and enthusiasm in executing an action plan are the most valuable outcomes of an assessment.

Software capability evaluations, on the other hand, are performed in a more audit-oriented environment. The objective is tied to monetary considerations, since the team's recommendations will help select contractors or set award fees. The emphasis is on a documented audit trail that reveals the software process actually implemented by the organization.

This does not mean, however, that the results of software process assessments and software capability evaluations should not be comparable. Since both methods are CMM-based, the points of comparison and difference should be evident and explainable.

4.3   Other Uses of the CMM in Process Improvement

For software engineering process groups or others trying to improve their software process, the CMM has specific value in the areas of action planning, implementing action plans, and defining processes. During action planning, the members of the software engineering process group, equipped with knowledge of their software process issues and business environment, can compare their current practices against the goals of the key process areas in the CMM. The key practices should be examined in relation to corporate goals, management priorities, the level of performance of the practice, the value of implementing each practice to the organization, and the ability of the organization to implement a practice in light of its culture.

The software engineering process group must next determine which process improvements are needed, how to effect the change, and obtain the necessary buy-in. The CMM aids this activity by providing a starting point for discussion about process improvement and by helping to surface disparate assumptions about commonly accepted software engineering practices. In implementing the action plan, the CMM and the key practices can be used by the process groups to construct parts of the operational action plan and to define the software process.

5 Future Directions of the CMM

Achieving higher levels of software process maturity is incremental and requires a long-term commitment to continuous process improvement. Software organizations may take ten years or more to build the foundation for, and a culture oriented toward, continuous process improvement. Although a decade-long process improvement program is foreign to most U.S. companies, this level of effort is required to produce mature software organizations. This time frame is consistent with experience from other industries, such as the U.S. automotive industry, that have achieved significant gains in process maturity [Gabor90].

5.1 What the CMM Does Not Cover

The CMM is not a silver bullet [Brooks87] and does not address all of the issues that are important for successful projects. For example, the CMM does not currently address expertise in particular application domains, advocate specific software technologies, or suggest how to select, hire, motivate, and retain competent people. Although these issues are crucial to a project's success, some of these issues have been analyzed in other contexts [Curtis90]. They have not, however, been integrated into the CMM. The CMM was specifically developed to provide an orderly, disciplined framework within which to address software management and engineering process issues.

5.2 Near-Term Activities

Tutorials and courses on the CMM are being presented at major conferences and seminars throughout the United States to ensure that the software industry has adequate awareness of the CMM and its associated products. CMM-based tools (e.g., the maturity questionnaire), software process assessment training, and software capability evaluation training are being developed and/or revised to incorporate the CMM.

The near-term focus on CMM development activities will be oriented towards tailored versions of the CMM, such as a CMM for small projects and/or small organizations. CMM v1.1 is expressed in terms of the normative practices of large, government contracting organizations, and these practices must be tailored to the needs of organizations that differ from this template.

5.3 Long-Term Activities

During the next few years, the CMM will continue to undergo extensive testing through use in software process assessments and software capability evaluations. CMM-based products and training materials will be developed and revised as appropriate. The CMM is a living document that will be improved, but it is anticipated that CMM v1.1 will remain the baseline until at least 1996. This provides an appropriate and realistic balance between the needs for stability and for continued improvement.

For the next version of the CMM, Version 2, the SEI will turn its attention to improving the overall model. While all levels of the model may be revised, the emphasis will be on Levels 4 and 5. Currently the key process areas for Levels 2 and 3 have been the most completely defined. Since few organizations have been assessed to be at Levels 4 or 5 [Humphrey91a, Kitson92], less is known about the characteristics of such organizations. The practices for these two levels will be refined as the SEI works closely with organizations that are striving to understand and achieve Levels 4 and 5. The CMM may also become multi-dimensional to address technology and human resource issues.

The SEI is also working with the International Standards Organization (ISO) in its efforts to build international standards for software process assessment, improvement, and capability evaluation. The development of the ISO standards will influence CMM v2, even as the SEI's process work will influence the activities of the ISO.

5.4   Conclusion

Continuous improvement applies to the maturity model and practices, just as it does to the software process. The potential impact of changes to the CMM on the software community will be carefully considered, but the CMM, the maturity questionnaire, and the software process assessment and software capability evaluation methods will continue to evolve as experience is gained with improving the software process. The SEI intends to work closely with industry, government, and academia in continuing this evolution.

The CMM provides a conceptual structure for improving the management and development of software products in a disciplined and consistent way. It does not guarantee that software products will be successfully built or that all problems in software engineering will be adequately resolved. The CMM identifies practices for a mature software process and provides examples of the state-of-the-practice (and in some cases, the state-of-the-art), but it is not meant to be either exhaustive or dictatorial. The CMM identifies the characteristics of an effective software process, but the mature organization addresses all issues essential to a successful project, including people and technology, as well as process.