Abstract

The SEI’s Capability Maturity Model for Software has popularized the notion of measuring the software process maturity of organizations. ISO’s SPICE project is currently creating a set of international standards for software process management that attempts to harmonize existing approaches. One of the SPICE objectives is to create a way of measuring process capability, while not using a specific approach such as the SEI’s maturity levels. The approach selected is to measure the implementation and institutionalization of specific processes; a process measure rather than an organization measure. Maturity levels can be viewed as sets of process profiles using this approach. During the development of version 2 of the CMM, one of the technical issues to be decided is whether to re-architect the CMM by layering organizational maturity on top of the SPICE process capability framework. This paper discusses the pros and cons of these two approaches.

1 Introduction

The Capability Maturity Model for Software (CMM) [Paulk93a, Paulk93b] developed by the Software Engineering Institute (SEI) has popularized the notion of measuring the software process maturity of organizations. Based on the work of the SEI and other appraisal method developers (e.g., Trillium, Software Technology Diagnostic, Bootstrap, Healthcheck, etc.), the International Organization for Standardization (ISO) is currently creating a set of international standards for software process assessment that attempts to harmonize existing approaches [Paulk94a].

One of the objectives of the ISO effort is to create a way of measuring process capability, while avoiding a specific approach to improvement such as the SEI’s maturity levels. The approach selected is to measure the implementation and institutionalization of specific processes; a process measure rather than an organization measure. Maturity levels can be viewed as sets of process profiles using this approach [Paulk94b].
2 The Capability Maturity Model for Software

The CMM describes the principles and practices underlying software process maturity and is intended to help software organizations improve the maturity of their software processes in terms of an evolutionary path from ad hoc, chaotic processes to mature, disciplined software processes. The CMM is organized into five maturity levels:

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.

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. The key process areas are listed in Figure 1.
<table>
<thead>
<tr>
<th>Level</th>
<th>Focus</th>
<th>Key Process Areas</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 Optimizing</td>
<td>Continuous process improvement</td>
<td>Defect Prevention Technology Change Management Process Change Management</td>
</tr>
<tr>
<td>4 Managed</td>
<td>Product and process quality</td>
<td>Quantitative Process Management Software Quality Management</td>
</tr>
<tr>
<td>3 Defined</td>
<td>Engineering processes and organizational support</td>
<td>Organization Process Focus Organization Process Definition Training Program Integrated Software Management Software Product Engineering Intergroup Coordination Peer Reviews</td>
</tr>
<tr>
<td>2 Repeatable</td>
<td>Project management processes</td>
<td>Requirements Management Software Project Planning Software Project Tracking &amp; Oversight Software Subcontract Management Software Quality Assurance Software Configuration Management</td>
</tr>
<tr>
<td>1 Initial</td>
<td>Competent people and heroics</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1. The key process areas in the CMM.

Each key process area is described in terms of key practices, which describe the specific infrastructure and activities that contribute most to the effective implementation and institutionalization of the key process area. Key practices are organized by common features. The common features are attributes that indicate whether the implementation and institutionalization of a key process area is effective, repeatable, and lasting. The five common features are

- Commitment to Perform,
- Ability to Perform,
- Activities Performed,
- Measurement and Analysis, and
- Verifying Implementation.

Some practices apply generically to any process, e.g., establishing policies and providing training. General practices that apply to every key process area at every maturity level are categorized by the common features. Establishing policies is a common practice in Commitment to Perform; providing training is a common practice in Ability to Perform; auditing is a common practice in Verifying...
Implementation. Practices specific to a particular process are contained in Activities Performed. The common features can be viewed as forming a mini-PDCA (Plan-Do-Check-Act) cycle.

3 ISO’s SPICE Project

ISO is developing a suite of standards on software process assessment under the rubric of SPICE – Software Process Improvement and Capability Determination [Paulk94a]. Organizations will be able to use this standard in many ways:
• in capability determination mode, to help a purchasing organization determine the capability of a potential software supplier,
• in process improvement mode, to help a software organization improve its own software development and maintenance processes, and
• in self-assessment mode, to help an organization determine its ability to implement a new software project.

The core set of SPICE products comprising the proposed software process assessment standard are:
• Introductory Guide
• Baseline Practices Guide
• Assessment Instrument
• Process Assessment Guide
• Process Improvement Guide
• Process Capability Determination Guide
• Assessor Training and Qualification Guide

3.1 The Baseline Practices Guide

The Baseline Practices Guide¹ (BPG) [BPG94] defines, at a high level, the goals and fundamental activities that are essential to good software engineering. The BPG is the SPICE equivalent of the SEI’s CMM. The BPG describes what activities are required, not how they are to be implemented. The baseline practices may be extended through the generation of application/sector specific Practice Guides to take account of specific industry, sector, or other requirements. The CMM is a possible example of a sector-specific Practice Guide for large, software-intensive projects and organizations.

¹ The BPG has recently been renamed “Software Process Assessment Part 2: A model for process management” as part of consolidating the SPICE product suite and aligning their names with ISO conventions. The term BPG is likely to be used informally for some time, however, and is used in this paper.
The BPG categorizes processes into five process categories. The Customer-Supplier process category consists of processes that directly impact the customer, support development and transition of the software to the customer, and provide for its correct operation and use. The Engineering process category consists of processes that directly specify, implement, or maintain a system and software product and its user documentation. The Project process category consists of processes which establish the project, and coordinate and manage its resources to produce a product or provide services which satisfy the customer. The Support process category consists of processes which enable and support the performance of the other processes on a project. The Organization process category consists of processes which establish the business goals of the organization and develop process, product, and resource assets which will help the organization achieve its business goals.

Each process in the BPG can be described in terms of base practices, which are its unique software engineering or management activities (similar to the Activities Performed in the CMM). Process categories, processes, and base practices provide a grouping by type of activity. These processes and activities characterize performance of a process, even if that performance is not systematic. Performance of the base practices may be ad hoc, unpredictable, inconsistent, poorly planned, and/or result in poor quality products, but those work products are at least marginally usable in achieving the purpose of the process. Implementing only the base practices of a process may be of minimal value and represents only the first step in building process capability, but the base practices represent the unique, functional activities of the process when instantiated in a particular environment.

Evolving process capability is expressed in terms of capability levels, common features, and generic practices in the BPG. A capability level is a set of common features (sets of activities) that work together to provide a major enhancement in the capability to perform a process. Each level provides a major enhancement in capability to that provided by its predecessors in the performance of a process. They constitute a rational way of progressing through the practices, which can be used to harmonize most of the different approaches to rating software processes.

Capability levels provide two benefits: they acknowledge dependencies among the practices of a process, and they help an organization identify which improvements it might perform first, based on a plausible sequence of process implementation. There are six capability levels in the BPG:

- The Not-Performed level (0) has no common features. There is general failure to perform the base practices in the Process.
- At the Performed-Informally level (1), the base practices of the process are generally performed. The performance of these base practices may not be rigorously planned and tracked. Performance depends on individual
knowledge and effort, but there is general agreement that this action is performed as and when required.

- At the Planned-and-Tracked level (2), performance of the base practices in the process are planned, tracked, and verified.
- At the Well-Defined level (3), base practices are performed according to a well-defined process using approved, tailored versions of standard, documented processes.
- At the Quantitatively-Controlled level (4), detailed measures of performance are collected and analyzed, which leads to a quantitative understanding of process capability and an improved ability to predict performance and objectively manage the process.
- At the Continuously-Improving level (5), continuous process improvement against quantitative process effectiveness and efficiency goals is enabled by quantitative feedback from performing the defined processes and from piloting innovative ideas and technologies.

A common feature in the BPG is a set of practices that address the same aspect of process implementation or institutionalization. BPG common features are clustered according to capability levels. A gap analysis can be performed on missing or incomplete features during an assessment to identify the major inhibitors to effective use of the process. BPG common features should always be applicable to any process.

A generic practice is an implementation or institutionalization practice (activity) that enhances the capability to perform any process. The generic practices characterize good process management that results in an increasing process capability for any process. A planned, well-defined, measured, and continuously improving process is consistently performed as the generic practices are implemented for a process. This process capability is built on the foundation of the base practices that describe the unique, functional activities of the process.

A process can be characterized as achieved a capability level if it has implemented the generic practices for that capability level and the lower capability levels.

4 Pros and Cons of the Two Architectures

Both the CMM’s staged approach to describing software process maturity and the BPG’s continuous approach to describing process evolution have advantages and disadvantages.

**Process evolution.** The BPG capability levels and the CMM maturity levels are similar, yet distinctly different. Capability levels are applied on a per process basis; organizational maturity levels can be defined as a set of profiles for these processes.
It is also practical to add level 0: a specific process may not be performed at all. One of the advantages of the BPG architecture is that it measures the evolution of each process separately from other processes.

In contrast, the key process areas in the CMM are defined as residing at a single maturity level. As processes mature they are described in different key process areas at different levels. For example, Software Project Planning and Software Project Tracking and Oversight at level 2 evolve to Integrated Software Management at level 3, are further refined in Quantitative Process Management at level 4, and are an aspect of Process Change Management at level 5. Processes that are not described in the CMM at a particular maturity level also exist and evolve. For example, the engineering processes for requirements analysis, design, code, and test are not described until the Defined level, yet organizations at lower maturity levels do perform these activities. One of the advantages of the CMM architecture is that it focuses on the “vital few” areas that typically block process performance at a particular stage in the organization’s life.

A disadvantage of the BPG architecture is that less important process issues can drown out the “vital few” issues as there are clashes over improvement priorities. A disadvantage of the CMM architecture is that people can lose sight of the processes that are not a focus of a particular maturity level, but which must still be performed.

**Establishing improvement priorities.** The BPG architecture does not prescribe any particular organizational improvement path. Improvement priorities are completely left up to the software organization, as determined in the context of its business objectives. Individual processes, at either the organization or project level, can be easily measured using this “continuous improvement” architecture for rating processes. Capability levels measure a specific process.

The “staged” architecture of the CMM builds process capability by focusing on the “vital few” issues for the organization as a whole, with the purpose of improving software supplier capability. The maturity levels prioritize general software problems. Although organizational capability and process capability are inextricably intertwined, building a continuous model on top of an architecture for improving organizational capability (i.e., the CMM maturity levels) is very difficult. The CMM emphasizes project management at level 2. Organizational and engineering processes are deferred to level 3. While engineering processes are certainly performed at levels 1 and 2, it would be more difficult to describe organizational processes at level 2 without violating the emphasis on project issues. Maturity levels measure the organization.
Building a staged model on top of a continuous architecture is more straightforward. The maturity levels in the CMM can be layered on top of the BPG capability levels, as is illustrated at a high level of abstraction in Figure 2.

![Figure 2. Layering organizational maturity onto process capability.](image)

To provide a relevant picture of the relationship, the mapping should be at the BPG process level rather than the process category level. For example, the current version of the CMM does not cover the customer-supplier relationship in detail, therefore it is desirable to only partially shade in the Customer-Supplier process category to indicate that only some processes in that category are addressed in the CMM. A detailed mapping of the CMM to the BPG has been done by Paulk and Konrad [Paulk94b] based on an early draft of the BPG. This should suffice to convey the concepts until the BPG is officially published by ISO, probably as a technical report in 1995 prior to submission for balloting as an international standard in 1997.

A disadvantage of the BPG architecture is that it may be difficult to decide which issues should be attacked first. A disadvantage of the CMM architecture is that it prescribes attacking project management issues before engineering ones (although
this prioritization is based on experience in successful process improvement). One could say that the BPG describes the terrain; the CMM provides a roadmap.

**Assessment usability.** Many of the concepts expressed in the CMM’s higher-level key process areas, in particular at level 4, are represented as common features and generic practices in the BPG. A disadvantage of the BPG is the potentially large number of decisions in rating a process. There are 26 generic practices and 35 processes, which suggests potentially 910 rating decisions, even if one ignores the 5-6 base practices per process. The CMM contains 316 key practices. Our pilot testing suggests that even rating the 112 activities performed at levels 2 and 3 can be a daunting task. Developing an assessment method that can effectively use the BPG architecture in rating the complete set of BPG processes is one of the challenges still facing the SPICE project (one suggestion has been to rate common features rather than generic practices, but this is still under discussion).

5 Conclusion

Re-architecting the CMM according to the BPG approach would involve significant structural changes to the CMM, but one of the deferred change requests for version 1.0 of the CMM was to have key process areas span maturity levels. This architectural change would address that request and perhaps communicate process capability and organizational maturity issues more effectively.

The BPG is still a work in progress. The road to international standardization is an arduous one. Although the SEI is working with ISO’s SPICE project to build the best possible international standard, our participation does not imply a commitment to use the standards eventually approved. That decision will depend on the final standard, on prototypes of version 2 of the CMM, on field trials of the prototypes, and on feedback from the CMM user community.

6 References


