SANS SOFTWARE SECURITY with Frank Kim

Secure Coding.
Practical steps to defend your web apps.

Copyright SANS Institute

Author Retains Full Rights

This paper is from the SANS Software Security site. Reposting is not permited without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Defending Web Applications Security Essentials (DEV522)"
at http://software-security.sans.orghttp://software-security.sans.org/events/

The Capability Maturity Model and Its Applications

Jim Hurst

The Capability Maturity Model and Its Applications

Introduction

Developing good software is difficult. Much time and effort has been spent searching for ways to improve software development. Many solutions have been proposed to help development projects deliver a quality product on time and on budget. One of the most sophisticated of these approaches is the Capability Maturity Model, or CMM. The CMM is a process improvement approach to software engineering, originally developed by the Software Engineering Institute (SEI) at Carnegie-Mellon University. The term CMM is occasionally used to describe any process improvement approach to software development. In this discussion, it refers strictly to the SEI model.

A process improvement model defines a set of steps intended to define, evaluate, and optimize the business processes within an organization. The CMM outlines a series of steps software development organizations can take to improve results. Organizations are evaluated against a scale of five maturity levels, and the steps needed to achieve the next level are identified. The CMM is an attempt to formalize what is often a chaotic and disorganized process. It emphasizes well-defined processes, goals, and practices as a way to turn software production into a successful, repeatable endeavor.

In 2002, CMM was replaced by the more comprehensive Capability Maturity Model Integration (CMMI). In the following section, the history of CMM and CMMI and the maturity levels of the two models are explained. The discussion concludes with an overview of how CMM and CMMI can be applied in an organization.

History

The CMM was developed between 1987 and 1997 by SEI for the United States Air Force. The CMM found its greatest uptake in large organizations such as governments and government contractors. It was also well received in industries with large, mission-critical projects, such as avionics software. The CMM represented a change in the way organizations viewed software development, integrating the lessons learned from high-precision manufacturing. Prior to the CMM, organizations tended to emphasize the results of development, rather than focusing on improving the process. The CMM surprised the industry by maintaining that flawed processes produce flawed results, and the way to improve the results is to improve the process.

CMMI was introduced in 2002, with the goal of improving the use of maturity models in software engineering and other disciplines. It integrates capability models for software development, systems engineering, and integrated product and process development. It provides for three overlapping areas of application, emphasizing software development, delivery of services, and government purchasing. CMMI provides two types of materials. First, it helps an organization evaluate existing processes. Second, it provides them a systematic way of improving those processes.

CMM and CMMI

The CMM is used to evaluate the maturity level of an organization. Five levels (1 through 5) are defined. The maturity levels effectively classify development organizations by their capability to control critical processes. The SEI maintains that predictability, effectiveness, and control of the development process improve as an organization moves up through the levels.

The goal is to achieve a level of discipline that provides for continuous improvement in the overall development process. Key process areas are sets of related activities that are used as a group to achieve a goal or set of goals. Defining goals is critical. The extent to which goals are met determines the capability of the organization at a given level. Goals define the scope, intent, and boundaries of each key process area.

The two different implementations of CMMI are staged and continuous. The staged model is descended from the software development CMM. The latest version of CMMI, version 1.2, contains 22 process areas, divided into four categories, with maturity levels for each (the previous version, 1.1, contains 25 process areas). The four categories are process management, project management, engineering, and support. Each of the 22 process areas contains 1 to 4 goals, with each goal consisting of different practices. An additional set of goals and practices (the generic goals and practices) apply across all process areas.
The staged model uses appraisals to produce a CMMI level rating. CMMI appraisals use the Standard CMMI Appraisal Method for Process Improvement (SCAMPI) to provide benchmarks in the different process areas. The focus of the remainder of this article is the use of the CMM and CMMI for software development.

The Levels

The adoption of CMMI includes more definition of process areas. Rather than a single grade, an appraisal in CMMI evaluates performance in all the process areas. This finer-grained approach allows for more detailed assessment of the areas an organization needs to improve. The levels in CMMI and CMM are approximately the same, although some have been renamed. Meeting the goals at one level explicitly requires meeting all goals at lower levels. In both models, maturity levels provide a way to predict the future performance of organizations based on process discipline. Recent versions of CMMI include a level 0, which is defined as Incomplete. Some authors have posited that negative levels should be provided, to account for environments that are actively counterproductive.

The key features of the maturity levels are briefly discussed next.

Maturity Level 1: Initial

In the beginning, everything is chaotic. SEI frequently uses the phrase “ad hoc” to describe processes at this maturity level. The organization does not provide stability in its processes. Success depends upon individual competence, motivation, and effort. Level 1 organizations can produce good results, but they are frequently late and over budget.

Maturity Level 2: Repeatable (in CMMI, Managed)

At level 2, success in development projects is repeatable. Requirements are managed, and processes are planned, performed, measured, and controlled. Project management is used to track costs and schedules. The discipline is present to ensure that practices endure in times of stress. Project status and delivery is visible to management at agreed-upon points, for example, major milestones.

Maturity Level 3: Defined

Level 3 organizations have standardized their development processes so that they are well-documented and understood. These standard processes are used to provide consistency across the organization. Management sets project objectives based on standard processes, and ensures that these goals are addressed. At maturity level 3, the standards and procedures for an individual project are derived from organizational standards to suit that particular project. This is a key distinction between levels 2 and 3. Both levels require project standards, procedures, and process descriptions. In level 2, they might be unique to a project. In level 3, they are tailored from broader organizational standards. It is expected in level 3 that processes will be described in more detail and with more rigor. Management is expected to understand relationships between processes and to collect detailed metrics of performance.

Maturity Level 4: Managed (in CMMI, Quantitatively Managed)

Level 4 is about introducing precise measurements into the process. Maturity level 4 organizations use quantitative metrics, including statistics, to control key processes and subprocesses. Numeric goals are established for quality and performance, and are used in process management. Detailed measures of performance are collected, analyzed, and archived for future reference. The critical distinction between level 3 and level 4 organizations is predictable performance. Level 4 organizations control performance with feedback based on quantitative information. When processes vary outside the normative performance, the sources of performance degradation are identified and corrective measures are applied.

Maturity Level 5: Optimizing

Organizations at maturity level 5 have developed continually improving process performance, based on a detailed understanding of the relationships between the processes and quantitative monitoring of process performance. Process performance improvement might be based on incremental improvements in existing processes, or through the introduction of technological innovations. Quantitative goals for process improvement for the organization are established, revised as business requirements change, and used in the managing projects. Process improvements are identified, analyzed, and implemented to address the organization’s most common problems. To be successful at level 5, organizations must motivate and empower employees in alignment with the organization’s business objectives and values. A key distinction between level 4 and level 5 is the types of process variation that are addressed. At level 4, the concern is with individual projects experiencing delays and variations. Level 5 organizations develop processes to address the common causes of process delay and variation, and to change processes to improve performance.

Moving Through Maturity Levels

The expected course of development for an organization implementing CMMI is sequential progress through the levels. The firm needs to stabilize at a given maturity level before advancing. Because each level builds upon the foundation of the previous level, skipping levels can be counterproductive. Organizations with mismatched levels in different process areas in a single project can put improvements at risk. For example, a maturity level 3 engineering process is more likely to fail if maturity level 2 management practices make for poorly planned schedules or a lack of change control.

Applying CMMI

Moving an organization to a capability maturity approach is an ambitious undertaking. It involves redesigning the corporate culture around processes, and in particular, around understanding and improving processes. CMM and CMMI have been implemented at thousands of organizations around the world. More than 1,000 organizations have chosen to make their appraisals public, and the results are available online through the SEI website.

Introducing capability maturity models to a new organization requires a high level of commitment from the management team. Corporate team members need formal training in the methods of process improvement. SEI recommends preparing the organization for change by creating a compelling case, including the reasons for the change, and the expected costs and benefits. The next step is to create an engineering process group, which will exist for the duration of the process improvement activity. This group can be used to complete a gap analysis to compare the organization's current workflows with the CMMI model. This picture of the current status can then be used to set the goals for the organization and build the process improvement plan. The next steps are to begin implementing the plan and tracking the progress of that implementation.

Market Acceptance of CMM and CMMI

These models were developed to provide metrics for assessing the capability of software development organizations. Formalizing development methods have helped many firms to produce quality software on schedule and on budget. The maturity models emphasize many beneficial elements, such as formal software specifications of what is to be built, technical specifications of how software is to be used, peer review of code, and version control. At a more abstract level, the idea that software development is a scientific process that is carefully controlled can have a beneficial impact.

The capability maturity models have not taken over the world, however. These are heavyweight processes compared to popular software methodologies such as agile software development. CMM requires significant resources to deploy and might not meet the needs of all organizations. Critics have claimed that CMM is best-suited for large bureaucratic organizations.

Summary

The Capability Maturity Model and its successor the Capability Maturity Model Integration are process improvement models applied to improving software development. They are based on five maturity levels that predict an organization's capability to

successfully develop software based on process discipline. The maturity levels are a framework to encourage discipline in the development process.

Governments, government contractors, and the avionics industry have found particular success with CMM and CMMI. Adopting capability maturity methods means changing the corporate culture, putting more emphasis on quality processes to produce quality products.

References:

Capability Maturity Model Integration, Version 1.1. CMMI Product Team

http://www.sei.cmu.edu/pub/documents/02.reports/pdf/02tr012.pdf

Published Appraisal Results. Software Engineering Institute.

http://sas.sei.cmu.edu/pars/pars_detail.aspx?a=7672

What Is CMMI? Software Engineering Institute.

http://www.sei.cmu.edu/cmmi/general/general.html

Last Updated: May 1st, 2017

Upcoming SANS App Sec Training

SANS Security West 2017	**San Diego, CA**	**May 09, 2017 - May 18, 2017**	**Live Event**
SANS Stockholm 2017	**Stockholm, Sweden**	**May 29, 2017 - Jun 03, 2017**	**Live Event**
SANS Minneapolis 2017	**Minneapolis, MN**	**Jun 19, 2017 - Jun 24, 2017**	**Live Event**
SANS London July 2017	**London, United Kingdom**	**Jul 03, 2017 - Jul 08, 2017**	**Live Event**
SANSFIRE 2017	**Washington, DC**	**Jul 22, 2017 - Jul 29, 2017**	**Live Event**
SANS Network Security 2017	**Las Vegas, NV**	**Sep 10, 2017 - Sep 17, 2017**	**Live Event**
Secure DevOps Summit & Training	**Denver, CO**	**Oct 10, 2017 - Oct 17, 2017**	**Live Event**
SANS Seattle 2017	**Seattle, WA**	**Oct 30, 2017 - Nov 04, 2017**	**Live Event**
SANS OnDemand	**Online**	**Anytime**	**Self Paced**
SANS SelfStudy	**Books & MP3s Only**	**Anytime**	**Self Paced**