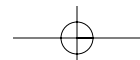
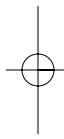
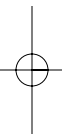


PART ONE

About CMMI



CHAPTER 1

INTRODUCTION

More now than ever, companies today want to deliver products better, faster, and cheaper. At the same time, in the high-technology environment of the twenty-first century, nearly all organizations have found themselves building more and more complex products. Today, a single company usually does not develop all the components that compose a product. More commonly, some components are built in-house and some are acquired; then all the components are integrated into the final product. Organizations must be able to manage and control this complex product development and maintenance.

Many organizations have also found themselves in the software business. Organizations that were not typically software companies—such as financial institutions, car manufacturers, airplane manufacturers, and insurance companies—find that much of their business relies on software. Software is often what differentiates them from their competitors. The problems these organizations address today involve both software and systems engineering. More and more, these disciplines are becoming a critical part of their business. In essence, these organizations are product developers that need a way to manage an integrated approach to their software and systems engineering as part of reaching their business objectives.

In the current marketplace, there are maturity models, standards, methodologies, and guidelines that can help an organization improve the way it does business. However, most available improvement approaches focus on a specific part of the business and do not take a systemic approach to the problems that most organizations are facing. For example, there are many maturity models available such as the Software Engineering Institute's (SEI's) Capability Maturity Model[®] for Software (SW-CMM[®]), which focuses on improving software, and the Electronic Industries Alliance's (EIA's) Systems Engineering Capability Model (SECM), which focuses on systems engineering. By focusing on improving one area of a business, these models have unfortunately perpetuated the stovepipes and barriers that exist in organizations.

4 PART ONE ABOUT CMMI

Capability Maturity Model® Integration (CMMI®) provides an opportunity to avoid or eliminate these stovepipes and barriers through integrated models that transcend disciplines. CMMI consists of best practices that address product development and maintenance. It addresses practices that cover the product's life cycle from conception through delivery and maintenance. There is an emphasis on both systems engineering and software engineering and the integration necessary to build and maintain the total product.

About Capability Maturity Models

The SEI has found several dimensions that an organization can focus on to improve its business. Figure 1.1 illustrates the three critical dimensions that organizations typically focus on: people, procedures and methods, and tools and equipment.

But what holds everything together? It is the processes used in your organization. Processes allow you to align the way you do business. They allow you to address scalability and provide a way to incorporate knowledge of how to do things better. Processes allow you to leverage your resources and to examine business trends.

This is not to say that people and technology are not important. We are living in a world where technology is changing by an order of magnitude every ten years. Similarly, people typically work for many companies throughout their careers. We live in a dynamic world. A focus on process provides the infrastruc-

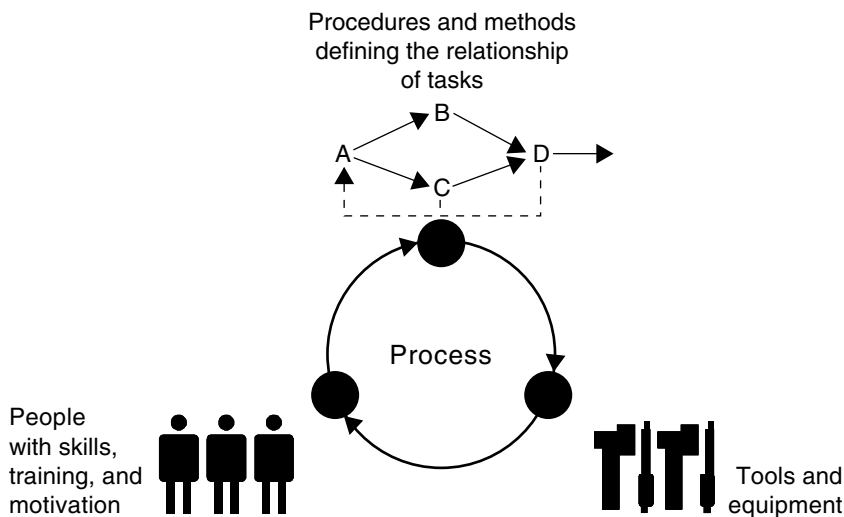


FIGURE 1.1
The Three Critical Dimensions

ture necessary to deal with an ever-changing world and to maximize personnel and technology to be more competitive.

Manufacturing has long recognized the importance of process effectiveness and efficiency. Today, many organizations in manufacturing and service industries recognize the importance of quality processes. Process helps an organization's workforce meet business objectives by helping them work smarter, not harder, and with improved consistency. Effective processes also provide a vehicle for introducing and using new technology in a way that best meets the business objectives of the organization.

In the 1930s, Walter Shewhart began work in process improvement with his principles of statistical quality control [Shewhart 31]. These principles were refined by W. Edwards Deming [Deming 86] and Joseph Juran [Juran 88]. Watts Humphrey, Ron Radice, and others extended these principles even further and began applying them to software in their work at IBM and the SEI. Humphrey's book *Managing the Software Process* [Humphrey 89] provides a description of the basic principles and concepts on which many of the capability maturity models are based.

The SEI has taken the process-management premise, "the quality of a system or product is highly influenced by the quality of the process used to develop and maintain it," and defined capability maturity models that embody this premise. The belief in this premise is worldwide in quality movements, as evidenced by the International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) body of standards.

Capability maturity models (CMMs) focus on improving processes in an organization. They contain the essential elements of effective processes for one or more disciplines and describe an evolutionary improvement path from ad hoc, immature processes to disciplined, mature processes with improved quality and effectiveness.

Mark Paulk and others at the Software Engineering Institute created the first capability maturity model designed for software organizations and published it in a book, *The Capability Maturity Model: Guidelines for Improving the Software Process* [SEI 95].

The SEI's book took the principles introduced almost a century ago and applied them to this never-ending cycle of process improvement. The value of this process improvement approach has been confirmed over time. Organizations have experienced increased productivity and quality, improved cycle time, and more accurate and predictable schedules and budgets [Herbsleb 97].

Evolution of CMMI

Since 1991, CMMs have been developed for a myriad of disciplines. Some of the most notable include models for systems engineering, software engineering,

6 PART ONE ABOUT CMMI

software acquisition, workforce management and development, and integrated product and process development.

Although these models have proved useful to many organizations, the use of multiple models has been problematic. Many organizations would like to focus their improvement efforts across the disciplines in their organizations. However, the differences among these discipline-specific models, including their architecture, content, and approach, have limited these organizations' ability to focus their improvements successfully. Further, applying multiple models that are not integrated within and across an organization is costly in terms of training, appraisals, and improvement activities.

The CMM IntegrationSM project was formed to sort out the problem of using multiple CMMs. The CMMI Product Team's mission was to combine three source models:

1. The Capability Maturity Model for Software (SW-CMM) v2.0 draft C
2. The Systems Engineering Capability Model¹ (SECM)
3. The Integrated Product Development Capability Maturity Model (IPD-CMM) v 0.98

The combination of these models into a single improvement framework was intended for use by organizations in their pursuit of enterprise-wide process improvement.

These three source models were selected because of their widespread adoption in the software and systems engineering communities and because of their different approaches to improving processes in an organization.

Using information from these popular and well-regarded models as source material, the CMMI Product Team created a cohesive set of integrated models that can be adopted by those currently using the source models, as well as by those new to the CMM concept. Hence, CMMI is a result of the evolution of the SW-CMM, the SECM, and the IPD-CMM.

Developing a set of integrated models involved more than simply adding existing model materials together. Using processes that promote consensus, the CMMI Product Team built a framework that accommodates multiple disciplines and is flexible enough to support the different approaches of the source models.

CMMI is the designated successor of the three source models. The SEI has released a policy to sunset the Software CMM; the revisions and improvements made during development of the Software CMM version 2.0 draft C are captured in CMMI with further improvements incorporated that were

1. The Systems Engineering Capability Model is also known as Electronic Industries Alliance 731 (EIA 731) [EIA 98].

discovered since 1997. The same can be said for the SECM and the IPD-CMM. These models are expected to be succeeded by CMMI.

The CMMI Framework was also designed to support the future integration of other disciplines. Furthermore, CMMI was developed to be consistent and compatible with the ISO/IEC 15504 Technical Report for Software Process Assessment [ISO 98].

CMMI has gone through an extensive review process. CMMI version 0.2 was publicly reviewed and used in pilot activities. Following release of that version, improvement was guided by change requests from public reviewers, piloting organizations, and focus groups.

The CMMI Product Team evaluated more than 3,000 change requests to create CMMI version 1.0. Shortly thereafter, version 1.02 was released, which incorporated several minor improvements. As with any release, opportunities for improvement remained.

Version 1.1 incorporated improvements guided by feedback from early use, more than 1,500 change requests submitted as part of the public review, and hundreds of comments as part of the change control process. No major changes to CMMI version 1.1 are expected before 2004.

Coverage of the Bodies of Knowledge

The intent of CMMI is to provide a CMM that covers product and service development and maintenance but also provides an extensible framework so that new bodies of knowledge can be added. Currently, four bodies of knowledge are available to you when planning process improvement using CMMI:

- Systems engineering
- Software engineering
- Integrated product and process development
- Supplier sourcing

This text refers to these bodies of knowledge as “disciplines.” In other words, when we refer to selecting a “discipline,” it can be one of the bodies of knowledge listed above. Other bodies of knowledge may be integrated into the CMMI Framework in the future; however, none are planned at this time.

Systems Engineering

Systems engineering covers the development of total systems, which may or may not include software. Systems engineers focus on transforming customers’ needs, expectations, and constraints into products and supporting these products throughout their life.

8 PART ONE ABOUT CMMI

Software Engineering

Software engineering covers the development of software systems. Software engineers focus on applying systematic, disciplined, and quantifiable approaches to the development, operation, and maintenance of software.

Integrated Product and Process Development

Integrated product and process development (IPPD) is a systematic approach that achieves a timely collaboration of relevant stakeholders throughout the life of the product to satisfy customers' needs, expectations, and requirements. The processes to support an IPPD approach are integrated with the other processes in the organization.

If a project or organization chooses IPPD, it performs the IPPD best practices concurrently with other best practices used to produce products (e.g., those related to systems engineering). That is, if an organization or project wishes to use IPPD, it must select one or more disciplines in addition to IPPD.

Supplier Sourcing

As work efforts become more complex, project managers may use suppliers to perform functions or add modifications to products that are specifically needed by the project. When those activities are critical, the project benefits from enhanced source analysis and from monitoring supplier activities before product delivery. Under these circumstances, the supplier sourcing discipline covers the acquisition of products from suppliers.

Similar to IPPD best practices, supplier sourcing best practices must be selected in conjunction with best practices used to produce products.

Selecting Disciplines

Disciplines are addressed in this book by the process areas associated with them and by model components called discipline amplifications.

A process area is a cluster of related best practices in an area that, when implemented collectively, satisfies a set of goals considered important for making significant improvement in that area.

A discipline amplification is a model component that contains information relevant to a particular discipline. In Part Two, you will find paragraphs labeled "For Software Engineering." These paragraphs are discipline amplifications for software engineers. This information applies only if you are improving your software engineering processes. The same is true for the other disciplines.

Because this book contains all of the disciplines currently available in CMMI, you must selectively apply the process areas found in this book to achieve your objectives.

Process Areas for Systems Engineering

If you are improving your systems engineering processes, you should select from the following process areas. The discipline amplifications for systems engineering receive special emphasis.

- Causal Analysis and Resolution
- Configuration Management
- Decision Analysis and Resolution
- Integrated Project Management (the first two specific goals)
- Measurement and Analysis
- Organizational Innovation and Deployment
- Organizational Process Definition
- Organizational Process Focus
- Organizational Process Performance
- Organizational Training
- Product Integration
- Project Monitoring and Control
- Project Planning
- Process and Product Quality Assurance
- Quantitative Project Management
- Requirements Development
- Requirements Management
- Risk Management
- Supplier Agreement Management
- Technical Solution
- Validation
- Verification

Process Areas for Software Engineering

If you are improving your software engineering processes, you will choose from the process areas that are the same as those listed for systems engineering. The only differences are that the discipline amplifications for software engineering receive special emphasis.

Process Areas for Integrated Product and Process Development

If you are improving your integrated product and process development processes, you will choose from the process areas that are the same as those listed for systems engineering with two additional process areas and additional best practices in the Integrated Project Management process area. The discipline amplifications for IPPD receive special emphasis.

The additional process areas are as follows:

- Integrated Teaming
- Organizational Environment for Integration

Process Areas for Supplier Sourcing

If you are improving your source selection processes, you will choose from the process areas that are the same as those listed for systems engineering with one additional process area. The discipline amplifications for supplier sourcing receive special emphasis.

The additional process area is as follows:

- Integrated Supplier Management

Multiple Disciplines

If you are improving multiple disciplines, choose from the process areas listed under all of the relevant disciplines and pay attention to all of the discipline amplifications for those disciplines.

A Conclusion

The only distinction between CMMI models for systems engineering and software engineering is the type of discipline amplifications included. This similarity of material was an intentional decision made during the development of CMMI. CMMI focuses on product development, improving both your systems engineering and software engineering functions with an integrated approach.

Resolving Different Approaches of CMMs

The definition of a capability maturity model allows the community to develop models having different approaches. As long as a model contains the essential elements of effective processes for one or more disciplines and describes an evolutionary improvement path from ad hoc, immature processes to disciplined, mature processes with improved quality and effectiveness, it is considered a CMM.

All of the source models for CMMI are considered capability maturity models; however, each has a different approach. Review and examination of each source model led to the discovery of two types of approaches to presenting capability maturity models. These types of approaches have been given the label “representations” in the process improvement community. A representation reflects the organization, use, and presentation of components in a model.

All capability maturity models have process areas that are defined by levels.² An example of a process area is Project Planning. There are two types of CMMI model representations: staged and continuous.

The staged representation is the approach used in the Software CMM. It is an approach that uses predefined sets of process areas to define an improvement path for an organization. This improvement path is described by a model component called a maturity level. A maturity level is a well-defined evolutionary plateau toward achieving improved organizational processes.

The continuous representation is the approach used in the SECM and the IPD-CMM. This approach allows an organization to select a specific process area and improve relative to it. The continuous representation uses capability levels to characterize improvement relative to an individual process area.

CMMI supports both representations because of the familiarity that people had with the source models and the concern that if one representation were selected over the other, part of the community would not adopt CMMI. Although this adds complexity to CMMI, it also provides an easier transition to CMMI for people familiar with one representation or the other.

Choosing a Representation

If you are new to process improvement and are not familiar with either the staged or continuous representation, you cannot go wrong if you choose one representation or the other. There are many valid reasons to select either representation.

If you have been using a CMM and you are familiar with a particular representation, we suggest that you continue to use that representation because it will make the transition to CMMI easier. Once you have become completely comfortable with CMMI, you might then decide to use the other representation.

2. Two of the source models use other terms for the concept of a process area. The Software CMM uses the term *key process areas*; the SECM uses the term *focus areas*.

12 PART ONE ABOUT CMMI

Because each representation has advantages over the other, some organizations use both representations to address particular needs at various times in their improvement programs. We provide the advantages and disadvantages of each representation to help you decide which representation is best for your organization.

Continuous Representation

The continuous representation offers a flexible approach to process improvement. An organization may choose to improve the performance of a single process-related trouble spot, or it can work on several areas that are closely aligned to the organization's business objectives. The continuous representation also allows an organization to improve different processes at different rates. There are some limitations on an organization's choices because of the dependencies among some process areas.

Capability levels are used to measure the improvement path through each process area from an unperformed process to an optimizing process. For example, an organization may wish to strive for reaching capability level 2 in one process area and capability level 4 in another. As the organization's process reaches a capability level, it sets its sights on the next capability level for that same process area or decides to widen its scope and create the same level of capability across a larger number of process areas.

If you know the processes that need improvement in your organization and you understand the dependencies among the process areas described in CMMI, the continuous representation would be a good choice for your organization.

Staged Representation

The staged representation offers a systematic, structured way to approach process improvement one step at a time. Achieving each stage ensures that an adequate improvement has been laid as a foundation for the next stage.

Process areas are organized by maturity levels that take much of the guesswork out of process improvement. The staged representation prescribes the order for implementing each process area according to maturity levels, which define the improvement path for an organization from the initial level to the optimizing level. Achieving each maturity level ensures that an adequate improvement foundation has been laid for the next maturity level and allows for lasting, incremental improvement.

If you do not know where to start and which processes to choose to improve, the staged representation is a good choice for you. It gives you a specific set of processes to improve that have been determined through more than a decade of research and experience in the software community.

Comparison of the Continuous and Staged Representations

Table 1.1 compares the advantages of each representation and may assist you with determining which representation is right for your organization.

Factors in Your Decision

Three categories of factors that may influence your decision when selecting a representation are business, culture, and legacy.

Business Factors

An organization with mature knowledge of its own business objectives is likely to have a strong mapping of its processes to its business objectives. Such an organization may find the continuous representation useful to appraise its

TABLE 1.1 Comparative Advantages of Continuous and Staged Representations

<i>Continuous Representation</i>	<i>Staged Representation</i>
Grants explicit freedom to select the order of improvement that best meets the organization's business objectives and mitigates the organization's areas of risk	Enables organizations to have a predefined and proved improvement path
Enables increased visibility of the capability achieved in each individual process area	Focuses on a set of processes that provide an organization with a specific capability that is characterized by each maturity level
Provides a capability-level rating that is used primarily for improvement in an organization and is rarely communicated externally	Provides a maturity-level rating that is often used in internal management communication, statements external to the organization, and during acquisitions as a means to qualify bidders
Allows improvements of different processes to be performed at different rates	Summarizes process-improvement results in a simple form—a single maturity-level number
Reflects a newer approach that does not yet have the data to demonstrate its ties to return on investment	Builds on a relatively long history of use that includes case studies and data that demonstrate proved return on investment
Provides an easy migration from the SECM to the CMMI	Provides an easy migration from the Software CMM to CMMI
Affords an easy comparison of process improvement to ISO/IEC 15504 because the organization of process areas is derived from 15504	Allows comparison to 15504, but the organization of process areas does not correspond to the organization used in ISO/IEC 15504

14 PART ONE ABOUT CMMI

processes and in determining how well the organization's processes support and meet its business objectives.

If an organization with a product lines focus decides to improve processes across the entire organization, it might be served best by the staged representation. The staged representation will help an organization select the critical processes to focus on for improvement.

The same organization may opt to improve processes by product line. In that case, it might select the continuous representation—and a different appraised rating of capability might be achieved for each product line. Both approaches are valid. The most important consideration is which business objectives you would like your process improvement program to support and how these business objectives align with the two representations.

Cultural Factors

Cultural factors to consider when selecting a representation have to do with an organization's ability to deploy a process improvement program. For instance, an organization might select the continuous representation if the corporate culture is process based and experienced in process improvement or has a specific process that needs to be improved quickly. An organization that has little experience in process improvement may choose the staged representation, which provides additional guidance on the order in which changes should occur.

Legacy

If an organization has experience with a staged representation, it may be wise to continue with the staged representation of CMMI, especially if it has invested resources and deployed processes across the organization that are associated with a staged representation. The same is true for the continuous representation.

Both staged and continuous representations were included in CMMI so that the communities that have used them successfully could continue in a manner that is comfortable and familiar as well as successful.

Why Not Both Representations?

Whether used for process improvement or appraisals, both representations are designed to offer essentially equivalent results. More than eighty percent of the CMMI model's content is common to both representations. Therefore, an organization need not select one representation over another.

In fact, an organization may find utility in both representations. It is rare that an organization will implement either representation exactly as prescribed. Organizations that are successful in process improvement often

define an improvement plan that focuses on the unique needs of that organization and therefore use the principles of both the staged and continuous representations.

For example, organizations that select the staged representation and are at maturity level 1 often implement the maturity level 2 process areas but also the Organizational Process Focus process area, which is included at maturity level 3. Another example is an organization that chooses the continuous representation for guiding its internal process improvement effort and then chooses the staged representation to conduct an appraisal.

Choosing Your Approach to Process Improvement

Now that you know the differences between the two representations, you should be able to decide on the approach that best fits your organization. To use CMMI as intended, you select two things: a set of disciplines and a representation. Unlike the CMMI models on the SEI Web site, this book contains both representations and all of the current disciplines that compose the CMMI Framework. This “complete picture” of the CMMI Framework in Part Two enables you to use exactly what you need as you learn about it. It also allows you to quickly use other information if you decide to change the representation or disciplines that you are using.

In Part Two, markings in the margins indicate when model components are “Staged Only” (apply only when using the staged representation) or “Continuous Only” (apply only when using the continuous representation).

To use one representation or the other in this book, locate the text in Part Two that is shaded and has margin notes. The model components that are unmarked apply when using either representation. (See pages 30 through 32 for a description of other typographical conventions used in this book.)

To demonstrate how to use this book, let’s look at two different scenarios. The first scenario is an organization that wants to improve its product development processes using a continuous approach. The second scenario is a software development company that uses IPPD, has been using the Software CMM, and now wants to use CMMI. This company has recently been rated at maturity level 3 according to the Software CMM version 1.1.

Scenario 1

In this scenario, you are using a continuous approach and therefore you select the processes that are important to your business objectives. Since there are twenty-five process areas to choose from, this is usually too many to focus on when starting out. You may need to narrow your focus. For

16 PART ONE ABOUT CMMI

example, you may find that your competitor is always getting its product released before yours. You may then choose to focus on improving your engineering and project management processes.

Building on this decision, you select these engineering process areas as a starting point: Product Integration, Requirements Development, Requirements Management, Technical Solution, Validation, and Verification. You also select Project Planning and Project Monitoring and Control.

You may at this point decide that eight process areas are still too many to focus on initially and you decide that the requirements process is really where the problems are. Consequently, you select the Requirements Development and Requirements Management process areas to begin your improvement efforts.

Next you decide how much improvement is needed in the requirements area. Do you have any processes in place already? If you don't, your process-improvement objectives may be to get to capability level 1.

Do you have your requirements development and management processes in place for each project but they are not repeatable processes? For example, policies, training, and tools are not implemented to support the processes. If your requirement processes are in place but there is no supporting infrastructure, then your process-improvement objectives may be to get to capability level 2.

Do you have all your requirements development and management processes in place but each project performs these processes differently? For example, your requirements elicitation process is not performed consistently across the organization. If this is the case, then your process-improvement objectives may be to get to capability level 3.

Do you consistently perform your requirements development and management processes but do not have an objective way to measure and improve these processes? If this is the case, then your process-improvement objectives may be to get to capability level 4.

Do you want to ensure that you are selecting the right processes to improve based on quantitative objectives to maximize your business? If yes, then your process-improvement objectives may be to get to capability level 5 for selected processes. In the description of each process area, remember to look for discipline amplications introduced by the phrases, "For Systems Engineering" and "For Software Engineering." Use all information that has no specific markings and the material that has the markings "Continuous Only" in the margins.

As you can see from this scenario, you need to understand what processes need improvement and also how much you want to improve each process. This is the fundamental principle behind the continuous representation.

Scenario 2

In the second scenario, you are a software development company using IPPD, using the Software CMM, and wanting to use CMMI. You select the process areas at maturity levels 2 and 3 for both the software and IPPD disciplines.

This selection includes the following seven process areas at maturity level 2: Requirements Management, Project Planning, Project Monitoring and Control, Supplier Agreement Management, Measurement and Analysis, Process and Product Quality Assurance, and Configuration Management. It also includes the following thirteen process areas at maturity level 3: Requirements Development, Technical Solution, Product Integration, Verification, Validation, Organizational Process Focus, Organizational Process Definition, Organizational Training, Integrated Project Management (all the specific goals), Risk Management, Integrated Teaming, Decision Analysis and Resolution, and Organizational Environment for Integration.

Since you have already been rated at maturity level 3 for the Software CMM, look at the CMMI process areas that were not in the Software CMM. These process areas include Measurement and Analysis, Requirements Development, Technical Solution, Product Integration, Verification, Validation, Risk Management, Integrated Teaming, Decision Analysis and Resolution, and Organizational Environment for Integration. Determine if you have these processes in your organization even though they were not described in the Software CMM. If there are processes in place that correspond to these process areas and for the other process areas that were in the Software CMM, perform a gap analysis against the goals and practices to make sure that you addressed the intent of each of the CMMI process areas.

Remember, in each process area you select, to look for the discipline amplications introduced by the phrases “For Software Engineering” and “For Integrated Product and Process Development.” Use all information that has no specific markings and the material that has the markings “Staged Only” in the margins.

This has been an initial look at the two representations. A more detailed description is provided in chapter 5.

The Advantages of CMMI

Since many organizations have been using the Software CMM or the SECM, it is important to see how CMMI is the next generation of process improvement—a clear step forward and upward. There are unmistakable benefits to making the transition to CMMI products or to beginning process improvement using CMMI products instead of others.

18 PART ONE ABOUT CMMI

CMMI provides more detailed coverage of the product life cycle than other process-improvement products used alone. For example, the engineering emphasis of CMMI has exceeded that found in the Software CMM. The process management emphasis of CMMI has exceeded that found in the SECM.

CMMI products incorporate many lessons that were learned during the development, maintenance, and use of the source models from which they were developed. Therefore, CMMI products have addressed some of the problems found in both the Software CMM and the SECM, for example.

Organizations that achieved maturity levels 4 or 5 using the Software CMM provided information to the SEI on their successes and difficulties. This information was used to develop more robust, high-level best practices in CMMI. Therefore, CMMI products better address the needs of organizations at higher maturity levels.

CMMI provides an opportunity to eliminate the stovepipes and barriers that typically exist in different parts of an organization and that typically are not addressed by other process-improvement models. The combination of useful information on engineering a product and proved practices for managing processes results in a set of well-integrated models that will facilitate project management and improve the development process—and the resulting products.

CMMI, which integrates software engineering and systems engineering into product engineering, is a valuable tool for many organizations. CMMI promotes collaboration between systems engineering and software engineering, thereby shifting the focus to the end product and its associated processes. Further, CMMI enables model and appraisal training to be simpler and more effective.

CMMI is valuable to organizations that produce software-only solutions. The systems engineering functions, not typically addressed in detail in other software-only models, are valuable to those producing software-only solutions. The handling of requirements, for example, is discussed in much more detail than in the Software CMM. Although not previously addressed in CMMs for software-only organizations, these practices use familiar terminology and model architecture and help to manage and prevent difficulties related to software requirements—a concept that is not new to many software organizations.

CMMI allows users to select the model representation (or both representations) that best suits their business objectives. The flexibility built into every CMMI model supports both staged and continuous approaches to process improvement with common terminology, architecture, and appraisal methods.

Although the initial focus of CMMI was on product and service engineering, CMMI was designed for other disciplines as well, thereby supporting enterprise-wide process improvement.

Like any other CMM, CMMI requires you to use professional judgment to interpret the information in Part Two. Although process areas describe behavior that should be exhibited in any organization, all practices must be interpreted using an in-depth knowledge of CMMI, the organization, the business environment, and the circumstances involved.

