

REQUIREMENTS MANAGEMENT UNDER CMMI



JAMES R. PERSSE

No. 4 in the Series

REQUIREMENTS MANAGEMENT UNDER CMMI[©]



JAMES R. PERSSE



Olive, LLC
Atlanta

*Excerpt from the book, Process Improvement Essentials,
published by O'Reilly Media*



Olive, LLC
Atlanta

Copyright © 2007 by James R. Persse

All Rights Reserved. This book, or parts thereof, may not be reproduced in any form without permission. For consideration, inquire with the copyright holder at 4580 Old Norcross Rd., Duluth, Georgia, 30096.

Olive is an authorized SEI Partner



For more information visit:

www.OliveI.net

or call

678-939-8710

Persse, James

Requirements Management under CMMI /

James R. Persse, PhD

p. cm.

1. United States – Computers – Reference – Technology

I. Title

CMMI and the CMMI swirl logo are copyrights of the Software Engineering Institute

Table of Contents

Introduction.....	1
Goals and Practices.....	10
Benefits of Requirements Management.....	32
Some Example Program Components.....	36





1

INTRODUCTION

Our systems are a reflection of the requirements we develop in tandem with our clients. Requirements encapsulate the business need, the functional expectations, and most importantly the value of the solution. Although the systems we design and build are by nature highly complex, we strive to maintain requirements sets that are straightforward, declarative, and simply organized. And well controlled. This promotes a direct and ongoing line of communications with the customer.

*Johann Strenberg, CEO, Tenzer Spring Dynamics
Digital data management for clinical assessments*

There are no doubt plenty of technology professionals who believe that requirements management is the single most important job for any project manager on any project. Most of these folks have reached this conclusion because at one time or another they have had to deal with the impact of poor requirements management.

Scope creep, misaligned expectations, backtracking, perplexed customers—these are all conditions that tend to pop up when a project’s requirements are left to fend for themselves. These kinds of conditions can cloud a project’s mission and objectives. Without active requirements management, a manager’s best efforts to control the traditional domains of cost, schedule, scope, and resources can become quickly compromised.

That’s why CMMI introduces this Process Area at Maturity Level 2. Requirements management under CMMI is seen as a basic and fundamental project management responsibility.

In this chapter, we'll look at how CMMI defines requirements management and explore how you might implement established requirements management practices on your projects.

The Purpose of Requirements Management

There are many potential ways to define a project's requirements. We'll take a specific look at some of those ways in the upcoming subsection Requirements—What Qualifies? But from a high-level, general perspective, it's reasonable to define requirements, broadly speaking, as those general expectations you share with your customer concerning what you're going to work on together, what you're going to build, what it should do after it's built, and what it's going to take to build it.

Because requirements are usually the main source of these expectations, it's helpful to think of them as a contract of mutual understanding. If you and your customer share this same understanding across the life of a project, chances are good that you'll both emerge in the end still pretty much in agreement.

But if your understandings begin to drift apart over time, your expectations may likewise begin to move farther and farther away from your customer's. What you think about what you and your teams ought to be building may not be what your customer thinks you ought to be building. And at some point, sooner or later, both of you are going to be standing together asking questions such as, "How did we end up here, and what went wrong?"

With a sound approach to requirements management, you and your customers can employ a mechanism to stay in sync, to move together, not drift apart.

No one building a new house would want the architect or the contractor to decide in private that an angled staircase is just as pleasing as a curved staircase and then act on that choice. And a waitperson who brings you beef when you ordered fish is not keeping up his or her end of the bargain. It's the same with a project's requirements. Requirements are a key ingredient to

project success. Very often they establish the initial customer relationship. And because they rest at the very core of project activity, requirements should be carefully and continuously managed.

A Core Project Management Responsibility

There are lots of good reasons to proactively manage requirements on your projects. Those managers I know who have learned this lesson the hard way can offer up plenty of reasons, with plenty of gory detail. But if you look at requirements from a fundamental business perspective, it becomes clear that it's important to manage them for five basic reasons.

1. They form the basis for agreement.
2. They define the scope and character of project work.
3. They may establish a legal responsibility.
4. They can serve as objective success criteria.
5. They are naturally subject to change.

Let's discuss these reasons further.

Basis for Agreement

By their very nature, requirements are the basis of the agreement and the understanding you share with your customer: "This is what we are required to do." However, requirements establish this understanding not just with your customers—they also forge a link between you and your management, and between you and your project teams. Requirements are the elements that keep you, your teams, and your customers in a harmony of understanding. The requirements are not the sole contributor here, of course, as there are other factors at work, but the part the requirements play is both crucial and significant.

Because they are typically introduced early in the lifecycle of the project (at least in baseline form), requirements are one of the

few touchstone artifacts that can reflect the evolution of the project in visible, tangible form, from beginning to end. And because requirements serve so well as a basis of agreement, it's important to ensure right from the start that they are recognized in that role and that mechanisms are set in place to ensure that the integrity of this basis remains intact.

Scope Definition

This is the classic reason why it's important to manage requirements. They define and bound the scope of all subsequent project work. If you look at the PMI's PMBOK, or at the ISO 9001:2000 standard, or at Six Sigma's emphasis on the Voice of the Customer, you'll see this point emphasized again and again. The requirements—in whatever form they come to you or in whatever form you generate them—determine or impact just about every aspect of project work: the amount of work to be done, the kind of work to be done, the types of resources you will need, the length of time, the amount of money, and so on.

Good project managers know as a matter of practice that it's smart to create plans based in large part on the requirements. (For more detail on this subject, see Chapter 3.) That being the case, if you can control the requirements, you ought to be able to keep a firm hand on your plans. And if your plans remain under control, all those issues of cost, time, and resources should be, by and large, pretty much under control, too.

Performance Obligation

Another reason to manage requirements can take on a special significance depending on the kinds of clients you deal with. With some clients—perhaps those issuing you large contracts or engaging you on projects of high sensitivity—the requirements can actually constitute a legally binding responsibility on your part. They may define a performance standard that you and your teams will be required to live up to. This is becoming more and more common today in organizations of all sizes, as IT projects expand in reach and business criticality.

Because fulfillment of the requirements can be a binding obligation, it becomes especially important to control them in an effective and conscientious manner. Failure to do so, or even a weak ability to do so, could have enormous ramifications not just for your client but for you, your management, and your organization as well.

Success Criteria

This fourth reason sheds light on the real and positive benefits you can reap from successful requirements management. You and your customers can use the requirements as a benchmark for project success. The full realization of the requirements can be equated, at the end of the lifecycle, with project success.

This is one tactic I like to see project teams and customers get together on. If both can agree to look at the requirements as a major part of the success criteria early in the project, chances are you'll pay attention to the requirements across the life of the project. You'll want to keep them current. You'll have a reason to work with your customers to make sure all expectations concerning them are staying in line.

Using the requirements as your success criteria is a great way to forge a common partnership with your customers, one in which obtaining the goal brings equal value and rewards to both parties.

Subject to Change

Now we come to a volatile reason why it's important to manage requirements.

By their very nature, requirements are subject to change.

It might be more accurate to say that they are destined to change. In the fields of systems engineering, software engineering, and technology development, we tend to employ a lot of fairly stabilized tools: programming languages, code compilers, modeling systems, diagrams, charts. But of all these tools, the one we most depend on—the English language—is the least stable, the one most open to interpretation. And because those of us in the

U.S. IT industry tend to express requirements in English, there is always going to be a degree of interpretation involved when working with them.

That trait carries with it the basic reality that, over time (a lot of time or a little time), the meaning of requirements may change, information may evolve, new elements might come to light. It's only natural. It's only human. It's difficult for any group of people to capture all at once what a complex system might have to do. It's even more difficult if those groups are new to technology projects or don't work in technology domains.

So under most circumstances, requirements are going to change. They will probably change on a somewhat regular basis. And with change comes another chief responsibility of project management: the introduction of change management. (See the upcoming subsection A Continuous Activity.)

Requirements—What Qualifies?

In a prototypical development organization, it should be easy to identify what the requirements are. They are the specs in the software requirements specification, or the contents in the requirements document, or maybe scenarios in a business needs report. Whatever form they take, the requirements stand out as the descriptions of what the technology teams have to build.

Things can get a little less clear-cut in shops that are not "typical," which is probably most shops. Teams dealing with integrated systems, combinations of hardware and software, and cross-performance interfaces have to analyze closely what their exact requirements are. Attention to domains and detail becomes crucial.

And then there are teams who say they probably have no requirements.

I worked with a major Medicare claims processing company that wanted to adopt CMMI across its IT groups. Most of these

groups, it turned out, didn't really build anything, not from scratch. They maintained existing systems.

Early in the initiative, the manager of one of these groups came to me and confessed that, as best he could figure, his requirements were nothing more than the service-level agreements contained in his department's annual contract. We looked at the contract, and he was right. On top of that, his people had only three service requirements: (1) In the event of an outage, restore system functionality within ten minutes; (2) formulate database queries for all registered customers on a first-in, first-out basis; and (3) prepare and issue a set of standard monthly and quarterly reports to Medicare services management.

The point here is that requirements can be very detailed or very high level. They can be extremely volatile or relatively stable. Look at your organization and at the way it establishes its customer agreements. You may base the relationship on a formal contract, an agreement, or a statement of work, or maybe on a traditional requirements document.

If you'd like a way to determine just what your requirements might be, make a list describing what you'll do for the client, and then compare those items with the five points we just discussed. Here they are again.

1. Requirements form the basis for agreement.
2. They define the scope and character of project work.
3. They may establish a legal responsibility.
4. They can serve as objective success criteria.
5. They are naturally subject to change.

If the descriptions tend to fall into or cover most of these points, chances are you're pretty safe considering those to be your requirements.

A Continuous Activity

The CMMI framework makes the point more than a few times that requirements management should be considered a continuous project activity. It is not just an early-lifecycle job. Nor is it a job that's put to sleep and then resurrected when test time rolls around. It's an activity that project management should account for across all phases of the project.

After all, new requirements might be introduced at any time. Existing requirements might have to be modified or extended. Other requirements might turn out to be obsolete or unnecessary. A development project, no matter what its discipline or business domain, is a dynamic entity, internally and externally. Its smooth evolution is one of the chief responsibilities of project management, and this includes the ongoing evolution of the requirements. As we'll see later in this chapter, the Specific Practices defined for the CMMI Requirements Management Process Area give us a set of recommendations we can use to establish this ongoing management in a smooth and integrated way.

No Requirements *Development* at CMMI Level 2?

This is a question people wonder about all the time. Why does CMMI include the Requirements *Management* Process Area at Level 2 but does not introduce the Requirements *Development* Process Area until Level 3? Requirements Development, of course, describes the set of practices used to elicit, define, and organize the requirements.

This order seems backwards to some people. But the Software Engineering Institute has a reason for it. The SEI's take is that a project team will not be able to do a good job of working with its requirements until it first has a way to manage them. So the approach is to first become practiced at managing the requirements and then move forward to develop them in an organized way with your customer.



2

GOALS AND PRACTICES

CMMI defines one Specific Goal and five Specific Practices for Requirements Management. Here is the official specification text for each of these elements.

SG 1 MANAGE REQUIREMENTS

Requirements are managed and inconsistencies with project plans and work products are identified.

SP 1.1 OBTAIN AN UNDERSTANDING OF THE REQUIREMENTS

Develop an understanding with the requirements providers on the meaning of the requirements.

SP 1.2 OBTAIN COMMITMENT TO THE REQUIREMENTS

Obtain commitment to the requirements from the project participants.

SP 1.3 MANAGE REQUIREMENTS CHANGES

Manage changes to the requirements as they evolve during the project.

SP 1.4 MAINTAIN BI-DIRECTIONAL TRACEABILITY OF REQUIREMENTS

Maintain bi-directional traceability among the requirements and the project plans and work products.

SP 1.5 IDENTIFY INCONSISTENCIES BETWEEN PROJECT WORK AND REQUIREMENTS

Identify inconsistencies between the project plans and work products and the requirements.¹

Note: Generic Goals and Practices

See Book 10 in this series for information about the important Generic Goals and Practices that support all the CMMI Process Areas.

SG 1: Manage Requirements

There is only one Specific Goal for Requirements Management: to manage the requirements. Simple enough. Obvious enough.

How you manage requirements in your organization is naturally going to depend on the makeup of your organization, the kind of work you do, the kinds of clients you have, and the policies of your management.

¹ Cited from "CMMI for Development," © 2007, Software Engineering Institute. Technical specification available for free download from www.sei.cmu.edu.

Let's look at the full description of the goal. "Requirements are managed and inconsistencies with project plans and work products are identified." In other words, you manage the requirements to ensure that your team is producing products that truly reflect what the requirements call for at any phase in the project lifecycle.

It is a simple goal, and it is an obvious one, but to get there you'll want to establish some project management procedures and approaches you can share across your teams so you can achieve this goal in a consistent and repeatable fashion. For example, the goal of managing requirements is accomplished in large part by giving project stakeholders the opportunity to review and understand the requirements, then provide input for clarification and completeness (Figure 5-1). Once all parties are in agreement, the baseline version of the requirements can be officially approved.

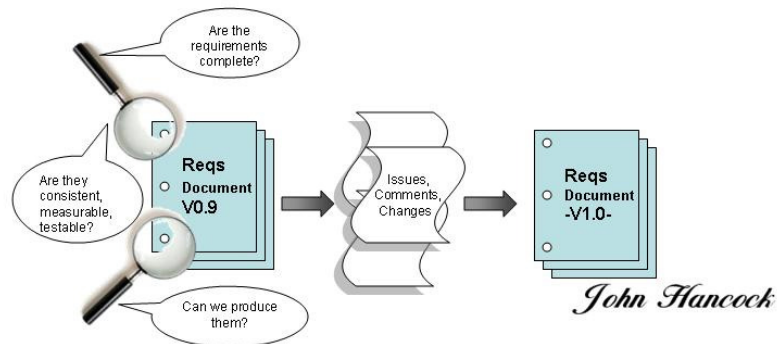


Figure 1

Thorough opportunity for review of the specifications is essential to commitment.

CMMI defines five Specific Practices that can be used together to help you effectively manage requirements. In the following subsections, we'll take a look at each practice and discuss ways that organizations might typically put these practices into place.

At this point, it's important to remember that we're presenting ideas and approaches we have seen work well in other shops.

These practices are not rules. They are not items you must implement in order to officially achieve Level 2 through a formal appraisal. Under CMMI, these are called Specific Practices, but they are meant to be flexible, as flexible as you need them to be.

A better way to look at the following material is to see it as a basis for planning how you might implement your own requirements management practices. To do this, you'll want to keep your culture in mind, think about what would work best for your folks, and try to capitalize on things you're already doing well that might fit in this domain.

Let's begin with the first practice.

SP 1.1: Obtain an Understanding of the Requirements

On most technology projects, there is always a temptation to hurry up and go. People might think that if they aren't hearing the tap-tap-tap of keyboards, progress is not being made. This first Specific Practice can be thought of as a reminder to resist jumping into the work until everyone has a comfortable understanding of what the requirements call for.

Enthusiasm is a good thing, but it is better when accompanied by focus. Understanding the requirements will help ensure that all stakeholders—team members, management, and customers—have the chance to establish a common understanding at the outset and to occupy common ground from that point on.

We have found that you can promote understanding of the requirements in a shop by using a set of five simple but effective activities (in any order or with a special focus on any steps you find especially helpful).

1. Document the requirements.
2. Identify the stakeholders.
3. Distribute the requirements for review.
4. Allow time for adequate review.
5. Encourage feedback.

Document the Requirements

If the requirements aren't documented, it's going to be hard to get a good understanding of them, or at least an understanding that can be shared. As strange and as risky as it seems, some technology shops resist putting requirements into writing. Such shops may start off with a documented core set, but then they seem to prefer to talk through the evolution, thinking that a casual approach will be more conducive to a free flow of ideas.

That may be true in some places, but it's not a sound project management practice. The strength of documented requirements is that everyone has access at any one time to the same sets of information. And that's the most effective way to manage how information is collated, disseminated, and refined.

The approach you take to documenting the requirements can take shape from the preferences of your shop. Some teams simply use a word processor and print out versions and updates as needed. Others use some of the more sophisticated automated tools available on the market today. Either approach is fine. The key is to ensure that the requirements exist in an externalized form, that they are not kept in someone's head—or lots of different heads—where they might soon take on a life of their own.

Identify the Stakeholders

The ultimate purpose of understanding the requirements is to get everyone comfortable enough so that they can commit to the requirements and establish a recognized baseline to work from. To do this effectively, you'll probably want to define who "everyone" really is. Chances are, it's not actually everyone. It's often not practical to include an all-encompassing group in the requirements review and approval process. That's why it's usually up to project management to identify the relevant stakeholders for this activity, those people who will be most impacted by the review and approval process.

Stakeholders are typically those people who will have some direct responsibility concerning the requirements: confirming

them, inspecting them, working from them. You can look two places when you want to identify the proper stakeholders for your project. First, identify external stakeholders, people outside your project team. These are usually customer contacts or liaisons who own the business side of the project. You work with these folks to confirm that you have the right sets of requirements and that they stay in line with customer needs across the life of the project. External stakeholders may also be the people who help confirm, during some stage of testing perhaps, that you have accounted for all of the requirements.

Next are internal stakeholders. These are the key members of your project team who will be charged with controlling how the requirements are worked through the various phases of development. Internal stakeholders also usually include select members of the management team.

Distribute the Requirements for Review

In the realm of CMMI and in the domain of sound project management, you are not required to force people to understand the requirements. Your job instead is to make sure that all the right people have an ample *opportunity* to understand them. To foster this opportunity, you should probably take on the job—or appoint someone to the job—of handling distribution of the requirements to the stakeholders when the need arises. You can handle this any number of ways: by sending an e-mail with attachments, by providing access into a set repository, by dropping off printed copies on desks. If you've accounted for the task of getting the requirements into the hands of the people who need to review and understand them, you can probably count on the stakeholders to take it from there.

Remember, you might have to redistribute updated versions of the requirements from time to time. It might be a good idea to establish a distribution procedure you can use throughout the project. (For more information on this, see the discussion of the Configuration Management Process Area in Chapter 6.)

Allow Time for Adequate Review

Your stakeholders will need ample time to review the requirements if they are going to acquire a good understanding of them. Some requirements sets are quite large. Others might be relatively small. The amount of time your stakeholders will need to get their arms around the requirements will naturally depend on how many requirements there are. It may also depend on how complex they are, how integrated they are, and any number of other factors.

Keep this in mind when you're sending the document out for review. Try to give your stakeholders a comfortable amount of time to roll this task into their current schedules. If your reviewers feel rushed or if their review time is squeezed, they may not be able to do a thorough job. That might lead to missed issues, gaps, and a level of uncertainty that could slow down getting the right level of commitment.

Encourage Feedback

Your stakeholders will probably do a good job of reviewing the requirements if they feel that their comments are going to be seriously considered. Sometimes in this kind of activity people have a tendency to hand off the requirements the way they might a live bomb—a little nervously, hoping it doesn't explode. But the great benefit of a review procedure is that you almost always end up with better requirements.

If you encourage people to give as much feedback as they'd like, you may end up getting a lot of it. But you can get work through and assess a pile of feedback; it's the lack of feedback that can introduce risks.

Encourage feedback, then, and give your stakeholders a way to submit it. You can use review and comment forms, e-mails, issue logs, or anything that will make it easy to document issues, concerns, and points for clarification.

SP 1.2: Obtain Commitment to the Requirements

The second practice under Requirements Management is to obtain commitment to the requirements. This is a natural extension of the first practice, understanding the requirements. In the same way that you don't want to begin project work until the requirements are understood, you also don't want to begin work until the requirements have been approved, or at least until a baseline set has been approved.

Commitment is important for a few reasons. First, it creates an atmosphere of acceptance—people have agreed that work is ready to proceed. Second, it implies input—if people are asked to approve, they probably have had the opportunity to not approve. And third, commitment is a visible way to establish consensus. Commitment demonstrates to your management and to customers that the project teams are indeed working together.

The concept of commitment can mean different things in different organizational cultures. In some shops, a verbal “yes” might be sufficient. In others, people might have to sign formal documents. However it's done, the purpose of commitment remains the same: to demonstrate common agreement. In light of that, then, here are five activities that have worked well at other shops for getting people to commit to the requirements.

1. Identify appropriate approver groups.
2. Incorporate feedback.
3. Set a time limit.
4. Ensure that commitment allows for future change.
5. Seek signatures.

Let's take a quick look at each one.

Identify Appropriate Approver Groups

As early as you can in the project lifecycle, try to identify who should probably approve the requirements. Most of the time, these folks will be readily apparent. But when they're not, project

management should walk the hallways to identify these people and then integrate them into the process.

You may have identified them already by identifying who the reviewers should be. Sometimes the reviewers and the approvers are the same group. Sometimes, though, you'll want to tap a broad set of reviewers for input and then use a select or different group to be responsible for approval.

It comes down to a question of authority and practicality. The people you want to commit to the requirements should be those stakeholders—internal and external—who have the ability to ensure that, once approved, the requirements will begin to be realized. Their blessing should be enough to assure that. At the same time, you want to be practical. Seeking out too many approvers, to get total consensus, might be a tall order in many organizations. Keeping the approver groups compact and appropriately authoritative will help you obtain commitment as smoothly and efficiently as possible.

Incorporate Feedback

In the section about SP 1.1, we looked at the importance of seeking feedback from your reviewers. Now the idea is to visibly incorporate that feedback into the requirements sets. There's a real value at play here, but it's one that has to be balanced with your judgment. The value is that the approvers will be much more comfortable committing to the requirements when they see that their questions, issues, and concerns have all been explicitly addressed. The balance comes from deciding which changes to incorporate.

If you've given your customers and teams ample time to review the requirements and if you've encouraged feedback, you may have to work through quite a few issues. You don't need to feel obligated to make everyone's recommended changes. A better course—and probably what the reviewers and approvers really want to see—is a serious consideration of all the questions, issues, and concerns. If people feel that the project stakeholders have had a chance to address these points and work through them, they will

probably be amenable to the changes you do incorporate and be ready to commit to those as being acceptable.

Set a Time Limit

Project managers have to be effective time managers. Tasks, deliverables, and goals almost always depend on some kind of time table. Keeping the project running according to schedule is a primary responsibility for project managers. That's why it's important to establish a time frame for obtaining commitment to the requirements.

If the window for review and discussion looks like it's always open, the stakeholders may want to analyze and reanalyze, never feeling quite ready to let the requirements go. Setting a deadline—a reasonable deadline—for this process will help your teams focus their efforts toward a degree of consensus. This sounds like a simple point, and it certainly is simple, but its importance should not be minimized. Many, many projects have gone off track early because their teams could not come together over the requirements. Management can help avoid this kind of disconnect by establishing the review and approval process up front, communicating this process to the people who will need to employ it, and then setting the time frame the process will need to operate within. Given this kind of pathway and a predetermined end point, your teams in all likelihood will be able to arrive at the state of consensus your project needs. And you can further support this by using the technique that follows: assuring your teams that commitment now still allows plenty of opportunity for future change.

Ensure That Commitment Allows for Future Change

If the people involved in your project are new to technology development or if you are dealing with a complex solution, you may find that people are hesitant to release or approve the requirements because they feel that once they commit, they won't have a chance to say anything else again. You can relieve them of that concern by assuring them that the requirements management process will be an ongoing and iterative activity across the project

and that there will be plenty of opportunity to tweak and adjust the requirements as needed.

The management objective at this juncture is to establish a baseline of requirements. The baseline will serve as the common go-point for the team and as the benchmark for all subsequent iterations. If your customers, technical team members, and other stakeholders feel comfortable that you'll accommodate ongoing input, they'll be much more willing to commit to the release of a baseline. And this will help you work through the next activity: getting approval signatures.

Seek Signatures

There are many ways to obtain commitment to the requirements. The idea to seek signatures is the most traditional way. In the world of business, ink on paper is a good thing. Signing is about as clear a way as any to show commitment. So, if you can get your stakeholders—those you've identified as approvers—to formally sign acceptance of the requirements, you've paved a clean path toward establishing a baseline.

But not all business cultures support such a traditional approach. Distance and location may preclude it. Organizational practice might promote other alternatives. Electronic signatures might be preferred. Your customers and management might be fine with e-mail approvals or even verbal acceptance.

The goal that project management wants to reach here is a recognizable milestone of agreement, some form of empirical evidence showing that commitment to the requirements has been achieved and that subsequent project work can now commence. The kind of "signature" that will work for your project will depend on the culture you operate in. Therefore it's a good idea to define this method of commitment early in the project lifecycle. In the domain of project management, the requirements are not the only data set that will need to be approved. You'll need similar approval methods of your plans, change requests, and project deliverables. Somehow, in some form, you'll need to seek signatures for all of these.

SP 1.3: Manage Requirements Changes

The third Specific Practice for this Process Area is to manage changes to the requirements. The first two practices—to obtain an understanding and to obtain commitment—are in place to help you establish working baselines for the requirements. You might well move through these practices at multiple times during a project. This third practice is defined to help manage the in-between evolution of the requirements.

Managing changes to the requirements implies a project management approach to change control and smooth change integration. We'll look at this topic in more depth when we look at the Configuration Management Process Area in Chapter 6, but for now let's look at five basic considerations you can employ to help manage requirements as they evolve.

1. Know that requirements will change.
2. Control with baselines.
3. Honor your customers' needs.
4. Assess proposed changes.
5. Incorporate changes in an orderly manner.

Let's take a quick look at each one.

Know That Requirements Will Change

This is really a reminder more than it is an activity. Because you may go through a good amount of work to get your teams to baseline a set of requirements, the thought that the set might change could make any project manager cringe. But change is a natural part of the business and development domains, so it's important to recognize that the requirements will change. Part of the focus of the Requirements Management Process Area is to track and monitor changes to requirements over time and to ensure that the work products that support the requirements remain in sync with the latest version of the requirements (Figure 2).

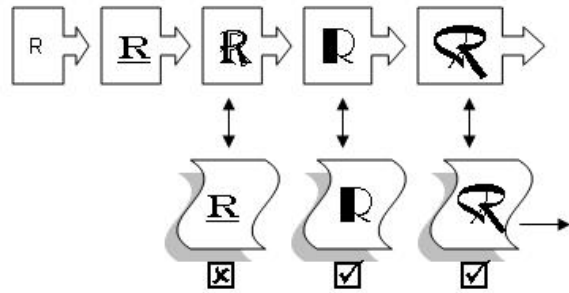


Figure 2
Managing requirements changes as a key to synchronicity

An important part of project manager's job is to know that requirements will change, to communicate that they may change, and then to support the inherent dynamic nature of requirements with a sound methodology to control and manage the changes. If you recognize change as an acceptable part of the project lifecycle, chances are your teams, management, and customers will work with you to make sure change is handled in as smooth a way as practical.

Control with Baselines

Data management is another important aspect of project management: knowing who needs access to which data, when, and to what extent. The requirements document (for the sake of convenience, we'll call it a document) is one work product that many people will need to access. It's also one of the few products that will move (or should move) through every cycle of a project. Because of its wide-reaching impact, requirements documents should be conscientiously controlled. One of the best ways to do this is through the use of version-controlled baselines.

Using such baselines implies a degree of ownership and centralized management. Project management assigns ownership of the requirements document to a responsible party, who then provides for the distribution of the document, ensures its integrity,

and provides for the smooth integration of approved changes into subsequent versions. (For more on the subject of creating and releasing baselines, see Chapter 6.)

Managing baselines this way will help make sure that your project teams are always working with the latest version of the specifications and that proposed changes can be assessed against the latest version as well.

Honor Your Customers' Needs

"All change is good" is an ancient Mandarin proverb. That might not always be true for technology projects, but some degree of change is inevitable. Most of the time you'll find that requests for change come from your customers. Something was forgotten or not accounted for in its full scope, or something changed. Of course, managing requirements in a professional manner means keeping a tight reign on them, making sure they don't run away from the project. But at the same time, this professionalism calls for you to keep your customer's mission in mind, to help the client realize what needs to be built for the business.

Project management should work to strike a balance between two important responsibilities. First is the responsibility to create the kind of product (or service) that will effectively serve the customer's business needs. Second is the responsibility to manage the amount of time, money, and resources the customer can provide to the project. Sometimes these two responsibilities butt up against each other. It's not always an easy job, and it may have more to do with the art of project management than with the technique of project management, but the project manager should always work to balance the customer's needs. This may mean, from time to time, introducing new or extended specs into the requirements mix. It might also mean working with the customer to establish a set of second-phase requirements, one that can be assessed once the current project goals have been met.

The objective here is to maintain an open ear to your client's needs. Understanding that the requirements will be subject to

change across the lifecycle of the project will help you and your team members keep that channel open and that balance point set.

Assess Proposed Changes

Open communication is one hallmark of a well-run project. The freedom to communicate fully and openly, combined with proper communication channels, permits project teams, customers, and management to exchange information, updates, and current status in a way that supports consensus and promotes openness. It's also a way to open the gate for a lot of change. Another prime responsibility of project management, then, is not only to control the amount of change introduced during a project but also to evaluate the value and appropriateness of each change request.

Customers might often feel pulled between including all the functionality they would like to have and accepting the amount of functionality they can currently afford. You can help balance this pull by establishing a formal way to assess requirements changes once they are submitted. If you can establish a protocol people can follow to review requirements changes and then determine the impacts on the project and the potential value the changes might bring, you'll have a way to objectify what might begin as a subjective preference. This kind of cost/benefit analysis can be useful for approving some items, tabling others for later consideration, and seeking further information for others.

It's also a good idea to assess these requests through the use of a change review committee, representatives from the various team and stakeholder groups who take on the job of reviewing and approving requirements changes as they arise. (In Chapter 6, we'll look at the potential use of a change control board and a change request procedure as a way to manage the intake of change requests and to assess their potential value to the project.)

Incorporate Changes in an Orderly Manner

The classic phrase used to describe the situation where requirements are poorly managed is *scope creep*. Scope creep can begin to take hold when project managers don't control

requirements baselines, when they don't work to assess change requests, or when they fail to integrate change in an orderly manner. Indeed, if the mission of this CMMI Process Area had to be distilled to its essence, it might well be to prevent scope creep. And remember, scope creep can move backward as well as forward. Forgetting to put in approved changes can be just as damaging as putting in any change willy-nilly.

So it's important to implement a way to incorporate change in an orderly manner. This may involve appointing someone on the team to own the requirements baseline and to be responsible for keeping it current. When a certain cycle of changes have been incorporated or when a new date-based version is ready, this owner will ensure that the contents are up to date. And just as importantly, this owner should ensure that the existence of the new version is communicated to the appropriate teams. Smooth change means not only incorporating change into the requirements documents but also making these changes available to the people who must incorporate them into their work.

That being the case, project management should consider ways to control the input of requirements changes into relevant documents (or repositories), establish a method to release new versions of the baselines, and then communicate the impact of the new materials to relevant team members, customers, and management.

SP 1.4: Maintain Bi-directional Traceability of Requirements

The fourth Specific Practice for Requirements Management is to manage the requirements, in part, by establishing bi-directional traceability. Traceability simply means establishing a mechanism to follow the life of each requirement as it moves from phase to phase in a project. Because (as we have seen) the requirements will change, because you may be dealing with a significant set of them, and because their integration into technical work products can be less than visible, traceability serves a valuable purpose. Basically,

traceability is a tool to make sure you don't lose any requirements along the way. And it's a great way to demonstrate at the end of the project that everything the customer asked for at the start has been successfully accounted for.

A lot of discussion always occurs when the concept of traceability comes up. The basic idea is pretty simple. CMMI sees traceability as a thread that weaves through the various phases of the project, connecting a requirement or specification to each distinct activity involved in product realization. The purpose is straightforward, too. It's a technique you can use to ensure that the requirements are being appropriately accounted for every step of the way. The detailed discussions around this practice usually deal with the extent of traceability and how sophisticated it needs to be.

Traceability can be very well managed by any of the requirements management tools available on today's market. But you don't need a special tool to take care of the job. Simple spreadsheets can often work just as well. The idea is to use traceability as a way to regularly monitor the requirements, to follow their integration into your solution. Whatever tool works for your team will meet the intention of CMMI.

Traceability can serve three project management points of focus.

1. Trace to plan.
2. Trace to anticipate.
3. Trace to know.

Trace to Plan

Setting up a structure to help you trace requirements can also help you plan downstream project activities. The structure of tracing will help you plan this work. If, for example, you are using a lifecycle that calls for functional analysis, high-level design, low-level design, coding, integration testing, and acceptance testing, you can use a traceability matrix to plan when logical groups of requirements might flow through each phase.

If you follow this method, you'll find the matrix can also help you organize and group requirements. It can help you prioritize requirements and then negotiate and allocate them across teams.

As your management picture of the requirements begins to take shape in the matrix, you will then realize an additional benefit: a framework for tracking and communicating requirements management and development progress.

Trace to Anticipate

Another strong benefit of using a traceability matrix is that it gives you a way to forecast upcoming activities and to anticipate any potential issues, risks, or bottlenecks. From a purely visual standpoint, the matrix can provide visible clues as to what appears to be on the horizon. Tracing will help you forecast events that may need mitigation and then understand what steps you might need to take to keep your other plans and activities on track. This way, traceability can become an additional management tool to help you allocate and balance resources and capacities in an effective manner.

Trace to Know

Perhaps the most important benefit traceability can deliver is the ability to know with a high degree of confidence where your requirements stand at any one time in the production process. Traceability will give you a foundation to always be able to communicate where you are in the realization process. It will help you know what you have accomplished. It will help you know what you need to get done. And it will help you share this information with others.

By employing traceability, you gain an increased degree of control over the flow of requirements as the product moves closer and closer to deployment.

SP 1.5: Identify Inconsistencies between Project Work and Requirements

The fifth Specific Practice for Requirements Management is to identify and resolve inconsistencies between plans, work products, and the requirements. This final practice can be seen as the culmination of the four previous practices. If you have worked to understand the requirements, obtained commitment to them, managed their changes, and traced them across project phases, you have the ability and the insight to keep the requirements continuously aligned with your plans and work products.

This practice is set into place to help you achieve two project benefits:

1. Harmony with plans
2. Harmony with work products

Harmony with Plans

Requirements are not the only things that will probably change during the course of a technology project. Plans are likely to change, too. In the realm of project management, planning is pretty much an ongoing activity. Schedule intervals may be refactored, resource levels rebalanced, costs re-estimated. Because both plans and requirements may coexist in a state of flux, it becomes especially important to keep the two aligned.

The ability to realize the requirements will be heavily influenced by the validity of your planning. And the integrity of the plans will be heavily influenced by how accurately they reflect the requirements. The relationship is one of mutual dependence.

Project management carries the responsibility for project planning, one of its most visible and noteworthy duties. But because of the tight link forged between the requirements and the plans, project management is also necessarily charged with making sure the plans accurately reflect what the requirements call for the teams to build at any point in time.

Harmony with Work Products

Harmony with work products is just as crucial a trait as harmony with project plans. In fact, this may be a more difficult activity to ensure, due to the level of detail that may be involved. That's one reason why proper change control and traceability are so helpful to requirements management. Project management should continually and regularly monitor the evolution of work products to make sure that what is being built—what's beginning to emerge off the assembly line—truly reflects the current state of requirements.

Many project managers address this job by holding regular status and review meetings. Milestone deliverables can then be meticulously compared to the requirements at predetermined points to ensure consistency. The types and number of work products you submit to this level of scrutiny will naturally depend on the type of project you are engaged in. The determining factor will be to identify those work products, in progress and completed, that have the largest potential to impact the validity of the final work. (For more on keeping the requirements aligned with project plans and work products, see Chapter 4.)

Some Other Ways to Achieve the Practices

The five Specific Practices just described can help you and your teams achieve the goal of managing project requirements in a complete and effective manner. We discussed a series of activities or considerations you might wish to use to help each practice work within your organization. It's important to remember that these are simply recommendations. There are many other ways you can set these practices into place. The ones you can call best practices are those that work for your organization and its culture. Here are some other ways we have seen teams set these practices into place.

SP 1.1: Obtain an Understanding of the Requirements

- Host group review and discussion sessions.
- Establish mini review teams.
- Conduct requirements training sessions.
- Sponsor customer-led orientation sessions.

SP 1.2: Obtain Commitment to the Requirements

- Take a vote.
- Employ the “Silence implies consent” rule.
- Designate a single authoritative approver.

SP 1.3: Manage Requirements Changes

- Set up a suggestion box.
- Establish an e-mail address for change requests.
- Recognize only certain stakeholders as change requestors.
- Set time limits on submitting domain changes.

SP 1.4: Maintain Bi-directional Traceability of Requirements

- Use a wall chart to map requirements and project phases.
- Invest in an automated requirements management tool.
- Use a word processor to create a table or spreadsheet.
- Use a relational database keyed to project phases.

SP 1.5: Identify Inconsistencies between Project Work and Requirements

- Hold regular technical status meetings.
- Conduct peer review inspections.
- Sponsor customer-led, in-progress inspections.



3

BENEFITS OF REQUIREMENTS MANAGEMENT

The five practices defined for this CMMI Process Area have been set into place to help project teams stay aligned with what their customers expect from them. Understanding the requirements, obtaining commitment to them, managing change as it occurs, tracing progress and integration, and maintaining harmony among the requirements, plans, and work products are recommendations that not only work well as stand-alone activities but also fit well together, complementing and augmenting one another.

From the perspective of effective and responsive project management, there are many benefits to managing requirements using these techniques. Control is enhanced. Accountability is made more visible. Risk is lowered. Predictability is raised. Each of these is the result of proactive and planned project management concerning the requirements. But let's take a look at four benefits

that are specifically related to the practice of sound requirements management:

1. Synchronicity
2. Enhanced control
3. Management visibility
4. A standard for fulfillment

Synchronicity

Synchronicity is the condition that exists when all parts of a system are properly aligned. Things turn smoothly. They work well together. For technology projects, sound requirements management is a way to introduce synchronicity across the various project teams. Because the requirements describe the system to be built and are documented, they can serve as the basis for establishing the scope and deliverables for a project and for establishing the basis for ongoing agreement concerning scope and deliverables.

With a common understanding of the requirements kept in place across the project and its phases, you and your stakeholders can expect to share the same general mission and goals for the project. This synchronicity extends from you to your customer, from you to your technical teams, and from you to your management.

Enhanced Control

A conscientious approach to requirements management adds an extra degree of control to technology projects. Schedules, budgets, and resources are all heavily influenced by the requirements. In the purest sense, these items should be direct reflections of the requirements. If management allows the requirements to drift from the plans set to contain them, a project can quickly lose its bearings. Scope creep, schedule delays, and cost overruns—all the classic project management issues—can easily materialize when a disconnect appears between what one party thinks you are working

on and what another thinks, or when the normal amount of change one might expect is handled in a way that appears disorganized.

By working diligently to place the requirements in the forefront of project activities and by providing procedures to address change as it occurs, you can help assure that such a disconnect does not occur. The control you'll exercise over the project will allow you to present a consistent and predictable management position to your customers and senior management.

Management Visibility

One of the strengths of well-tendered project management is that it brings visibility into the management process. Black-box management—hiding the process from sight—is almost always a sign of trouble. By extension, then, one of the chief strengths of sound requirements management is that it's a technique to add visibility, to shed light on the project so that everyone knows what is being worked on and what the priorities are. There are no hidden corners of functionality, no side-door negotiations.

Requirements management as described under CMMI helps you establish a single functional view of the project's mission and scope that can be readily shared with all your stakeholders. And because the view has been supported by protocols and procedures that address and manage change, it becomes open to everyone who may have a stake in the evolution of the requirements.

A Standard for Fulfillment

The fourth benefit of requirements management using CMMI recommendations is the realization of a standard for fulfillment. Most technology projects culminate in some form of verification and validation activities. Verification tests make sure the requirements can be fully traced to the resulting product. Validation tests make sure the product operates in its intended environment.

When you manage the requirements well over the life of a project, the verification and validation activities should emerge with clear and reliable results. The requirements here serve as a standard of fulfillment, as the benchmark you can use to compare the performance of what you have built. This standard gives you a reference point that is objective and can be relied on and referenced by all stakeholders.



4

SOME EXAMPLE PROGRAM COMPONENTS

In this chapter, we've looked at one Specific Goal and five Specific Practices recommended for the Requirements Management Process Area. As noted, the single intention that all this detail steers us toward is the consistent and informed management of the requirements as they move from phase to phase in a project. This results in a predictable path people can follow on a project and a path that the organization can use from project to project.

To support this degree of consistency and repeatability, you may want to create a set of assets and artifacts that your organization can use as part of its requirements management program. Here are some typical tools other organizations have used to help them achieve compliance with CMMI and manage their requirements in a common way.

Requirements Management Policy

A policy is an executive mandate that promotes the purpose and intention of your requirements management activities. A policy of this kind is usually a short document (a page or two) that summarizes the general approach the organization will take toward managing requirements. Setting a requirements management policy in place is a good way to establish an organizational standard and introduce your teams to the accepted methods of requirements management.

Requirements Document Review Procedure

In one form or another, you'll want to establish a process your teams can follow to review the requirements. This process will help teams review the requirements for two purposes: to understand them and to commit to them. This does not have to be a complex procedure; it can be a few simple steps. The reason to document the review procedure is to give your people a ready method to follow each time they have to undertake a review activity. If you have a process ready for them to follow, your teams can focus on the review activities, not on figuring out how to manage the review activities.

Requirements Review Checklist

This checklist provides your teams with a way to evaluate the requirements in a consistent way, using common criteria. The checklist might ask the teams to mark whether the requirements appear to be complete, whether they are clear, whether any appear to conflict with others, and so on. The checklist can be used as a review support tool to remind teams of attributes to look for to make sure the requirements are in a useful condition for the project teams.

Requirements Document Stakeholder ID Form

Identifying relevant stakeholders is an important part of requirements management. These stakeholders are those people—internal and external to the project teams—who are both qualified to judge the quality of the requirements and chosen as the logical parties to approve and commit to the requirements. An ID form is a helpful document to use to initially identify and then track the recognized stakeholders for requirements activities. This form can serve as a ready reference for people who may need to contact the stakeholders or work with them.

Requirements Review and Comments Form

A great way to understand the requirements and then obtain commitment to them is to provide your people with an opportunity to review the requirements and submit questions and comments on them. A requirements review and comments form can be a handy tool for supporting these activities. This need not be fancy. A basic form can capture information such as the name of the document being reviewed, the name of the reviewer, and a listing of questions or issues the reviewer noted about the document.

The form is useful in two ways. First, it gives the reviewers a tangible way to document their questions and concerns. Second, it provides a good mechanism for submitting questions and concerns for discussion and consideration.

Requirements Change Request Procedure

We'll take a deeper look at this procedure in Chapter 6. For now, the idea is to provide a procedure that your teams can follow to create a requirements change request, submit it to a recognized review body, and then track the progress and status of the request. Having such a change request procedure is an immensely helpful asset for any project and any organization. Since the management of change in general is such a major responsibility and the

management of requirements changes in particular is such a crucial one, having a formalized change request procedure provides a mechanism for the smooth management of all aspects of change.

Requirements Baseline Sign-Off Form

The goal of requirements management is to ultimately arrive at an approved requirements document: a baseline version for distribution. Once the stakeholders have indicated commitment to the requirements, you may want to provide a sign-off form to use for a couple of purposes. First, it provides the paper trail to show commitment. Your stakeholders can use this form to sign on the dotted line (or indicate approval in whatever way works best for your organization). Second, this form can also serve as the official check-in form your configuration manager or document owner can use to begin the work of managing a new version of the requirements.

Look to the Web Site for . . .

- Requirements management policy
- Requirements document review procedure
- Requirements review checklist
- Requirements document stakeholder ID form
- Requirements review form
- Requirements review invitation notice
- Requirements change request procedure

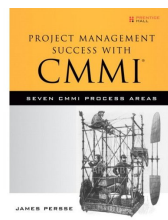
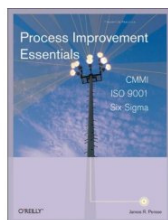
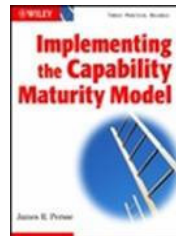
About the Author

James Persse has 18 years of experience providing process improvement design and consulting services to both large and small technology organizations. His specialties include CMMI, ISO 9001:2000, Six Sigma, and ITIL. He holds a doctoral degree in Information Technology Management with an emphasis in process improvement. He has worked with a diverse client base that includes T-Mobile USA, Athena Technologies, the U.S. Department of Defense, Pitney Bowes International, MCI, BellSouth Science and Technologies, and Palmetto GBA.

He is the author of the following:

- *Process Improvement Essentials: CMMI, ISO 9001, Six Sigma* (O'Reilly Media, 2006)
- *Project Management Success with CMMI* (Prentice Hall, 2007)
- *Implementing the Capability Maturity Model* (John Wiley and Sons, 2001)
- *Bit x Bit: Topics in Technology Management* (Little Hill, 2000)

Dr. Persse can be reached at jpersse@AltairSol.com or jrp@persse.com. Or call 678-939-8710.



[Books available at Amazon and Barnes & Noble.]



REQUIREMENTS MANAGEMENT UNDER CMMI

This clear and concise book includes practical information on...

- THE PURPOSE OF REQUIREMENTS MANAGEMENT
- GOALS AND PRACTICES
- BENEFITS OF SOUND REQUIREMENTS MANAGEMENT
- SOME EXAMPLE PROGRAM COMPONENTS

James Persse has 18 years of experience providing process improvement design and consulting services to both large and small technology organizations. His specialties include CMMI, ISO 9001:2000, Six Sigma, and ITIL. He holds a doctoral degree in Information Technology Management with an emphasis in process improvement. He has worked with a diverse client base that includes T-Mobile USA, Athena Technologies, the U.S. Department of Defense, Pitney Bowes International, MCI, BellSouth Science and Technologies, and Palmetto GBA.

