



SEI SERIES IN SOFTWARE ENGINEERING



Integrating CMMI[®] and Agile Development



Case Studies
and Proven
Techniques
for Faster
Performance
Improvement

Paul E. McMahon



CarnegieMellon Software Engineering Institute

The SEI Series in Software Engineering

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.

CMM, CMMI, Capability Maturity Model, Capability Maturity Modeling, Carnegie Mellon, CERT, and CERT Coordination Center are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

ATAM; Architecture Tradeoff Analysis Method; CMM Integration; COTS Usage-Risk Evaluation; CURE; EPIC; Evolutionary Process for Integrating COTS Based Systems; Framework for Software Product Line Practice; IDEAL; Interim Profile; OAR; OCTAVE; Operationally Critical Threat, Asset, and Vulnerability Evaluation; Options Analysis for Reengineering; Personal Software Process; PLTP; Product Line Technical Probe; PSP; SCAMPI; SCAMPI Lead Appraiser; SCAMPI Lead Assessor; SCE; SEI; SEPG; Team Software Process; and TSP are service marks of Carnegie Mellon University.

The author and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

The publisher offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales, which may include electronic versions and/or custom covers and content particular to your business, training goals, marketing focus, and branding interests. For more information, please contact: U. S. Corporate and Government Sales, (800) 382-3419, corpsales@pearsontechgroup.com.

For sales outside the United States, please contact: International Sales, international@pearsoned.com.

Visit us on the Web: informit.com/aw

Library of Congress Cataloging-in-Publication Data

McMahon, Paul E.

Integrating CMMI and agile development: case studies and proven techniques for faster performance improvement / Paul E. McMahon.

p. cm.

Includes index.

ISBN 978-0-321-71410-7 (pbk. : alk. paper)

1. Capability maturity model (Computer software) 2. Capability maturity model (Computer software)—Case studies. 3. Agile software development. I. Title.

QA76.758.M35 2010

005.1—dc22

2010018025

Copyright © 2011 Pearson Education, Inc.

All rights reserved. Printed in the United States of America. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, write to: Pearson Education, Inc., Rights and Contracts Department, 501 Boylston Street, Suite 900, Boston, MA 02116, fax: (617) 671-3447

ISBN-13: 978-0-321-71410-7

ISBN-10: 0-321-71410-5

Text printed in the United States on recycled paper at RR Donnelley in Crawfordsville, Indiana.
First printing, August 2010

Contents

Foreword by Mike Phillips	xxi
Foreword by Hillel Glazer	xxiii
Preface	xxv
Acknowledgments	xxx
Part I. Introduction	1
Chapter 1. Introduction and CMMI/Agile Primers	5
1.1 Introduction and CMMI Primer	5
1.2 Agile Primer	10
1.3 General Information about the Case Studies	12
1.4 General Information about Terminology Used in the Book	13
Part II. Helping Mature Organizations Increase Agility	15
Chapter 2. Techniques to Increase Agility in CMMI Mature Organizations	17
2.1 What You Will Learn in This Chapter	18
2.2 LACM Case Study Background	18
2.3 Where to Start When Using the CMMI Model to Increase Agility	18
2.4 Where Many Organizations Wrongly Start When Using the CMMI Model	20

2.5	How the CMMI Model Is Often Used, and Options Not Well Understood	20
2.6	Aligning Your Process Initiatives with Your Real Business Objectives	21
2.7	Aligning Process Descriptions and Training with the Real Process	22
2.8	Two Specific Examples to Increase Agility: Pruning and Leaning	23
2.9	Why More Organizations Don't Prune and Lean Their Processes Today	25
2.10	Understanding the CMMI Model Intent to Help Your Organization Succeed	25
2.11	Options You Have in Using the CMMI Model for Appraisals	26
2.12	An Alternative Approach to Agility	27
2.13	Summary: How CMMI Helps Agile	29
2.14	Summary: How Agile Helps CMMI	29
Chapter 3.	Agility and the Higher CMMI Level Practices	31
3.1	What You Will Learn in This Chapter	31
3.2	Background on the Higher CMMI Level Practices	32
3.3	Case Study Background	33
3.4	Measurement Fundamentals	33
3.5	Measurement in the Case Study	34
3.6	Stepping Back	34
3.7	Digging Deeper for Candidate Root Causes	35
3.8	Specific Context Relevant Measures	36
3.9	Deriving the Right Data and Caring about the Data	37
3.10	What Does This Have to Do with CMMI High-Level Practices?	38
3.11	The Right Time to Implement CMMI Level 4/5 Practices	38
3.12	Relationships among CMMI, Agile, and Lean	38
3.13	Back to the Case Study: How CMMI, Agile, and Lean Can Help Together	39

3.14	What Happened in the Case Study and Process Improvement Insights	40
3.15	Back to the Case Study Again: What Really Happened	40
3.16	Insight	41
3.17	More about the Real Intent of CMMI Level 4 and 5 Practices	42
3.18	Continuous Process Improvement at LACM	45
3.19	Why the Unprecedented Success at LACM?	48
3.20	Diddling in DOORS: A Story about Real Work Management and Measurement	48
3.21	Finance Perspective on Work Management and Measurement	51
3.22	Is the CMMI Measurement and Analysis Process Area Inconsistent with the Agile Principle of Simplicity?	52
3.23	How LACM Handled Measurement and Analysis from the CMMI Perspective	53
3.24	Summary	53
3.25	Summary: How CMMI Helps Agile	54
3.26	Summary: How Agile Helps CMMI	54
Part III. Helping Agile Organizations Increase Maturity		55
Chapter 4. Bringing Process Maturity to Agile Organizations—Part I		57
4.1	What You Will Learn in This Chapter	57
4.2	BOND Case Study Background	58
4.3	What Is a Gap Analysis and Why Is It Crucial for Agile Organizations?	59
4.4	Keys to Conducting a Gap Analysis for an Agile Organization	60
4.5	Example of “Potential Weakness” Against CMMI in an Agile Organization	62
4.6	Running Process Improvement like a Project	64
4.7	TWG Approach for Agile Organizations	64

4.8	Revisiting the Goal and Challenges on the Process Improvement Project	66
4.9	Alternative Practices and Tailored Agile TWG	67
4.10	Returning to the Peer Review Example	69
4.11	Tailored TWG Techniques and Lessons at BOND	70
4.12	Preparation Work for Running Agile TWGs	71
4.13	Packaging of Processes	71
4.14	An Agile Organizational Process Asset Structure	73
4.15	Process Asset Guidelines Used at BOND	77
4.16	Different Organizations with Different Process Asset Structures	77
4.17	Agile TWG Roles and Responsibilities	78
4.18	Effective Techniques to Run an Agile TWG	79
4.19	Separating the TWG Work from the Lead Offline Work	79
4.20	What Do You Do When You Find a Gap?	80
4.21	Answers to Common Questions When Running an Agile TWG	81
4.22	Do I Need a DAR Process?	82
4.23	Do I Need to Verify Everything I Develop?	82
4.24	Do I Need to Make Sure the Steps in My Processes Are in the Right Order?	83
4.25	Do I Need to Make Sure Process Descriptions Are Not Redundant?	84
4.26	Can Requirements Be Captured in an Email or PowerPoint Slides?	85
4.27	Do Requirements Need to Be Captured in Single “Shall Statements”?	86
4.28	Formalizing Informality	86
4.29	Summary	88
4.30	Summary: How Agile Helps CMMI	88
Chapter 5. Bringing Process Maturity to Agile Organizations—Part II		91
5.1	What You Will Learn in This Chapter	91
5.2	BOND Case Study Background	92

5.3	Project Management at BOND	95
5.4	Starting with Roles and Responsibilities at BOND	96
5.5	Growing Project Leaders from the Inside	98
5.6	Example Stretch Point: Adding a Project Management Plan per Agreed Template	99
5.7	“The What”—Scoping the Effort	101
5.8	“The Who”—Managing Your Resource and Skill Needs	102
5.9	Common “Undocumented-Super-Spreadsheet” Resource Management Process	104
5.10	“The When”	104
5.11	Life Cycle—It’s Your Choice	106
5.12	“The How”—Team Meetings, Task Monitoring, and Course Correction	108
5.13	Senior Management Briefings: An Area in Which the CMMI Can Help Agile	108
5.14	Example of Senior Brief Evolution: Backup Slides for Efficient Use of Time	109
5.15	“The How Much”—Don’t Force the Team to Perform “Unnatural Acts”	110
5.16	Lessons from Formalizing Planning at BOND	111
5.17	The Plan as a Living Document at BOND	113
5.18	The Power of Templates	113
5.19	Do I Need to Write Down Meeting Minutes and Action Items?	116
5.20	Involving Relevant Stakeholders	118
5.21	Involving Relevant Stakeholders —Additional Help Sometimes Needed	119
5.22	Sharing Across the Organization	120
5.23	A Measurement and Analysis Process That Fits an Agile Organization	123
5.24	Training All Project Personnel in the Organization	126
5.25	Technical Solution in an Agile Organization	127
5.26	Product and Process Quality Assurance	128
5.27	Mitigating the Risk of Your CMMI Appraisal in an Agile Organization	129

- 5.28 Lost Momentum Risk After Reaching Your CMMI Goal 130
- 5.29 Party Time! We’re Level 3! The Meeting a Year Later
with Ethan 131
- 5.30 Summary 132
- 5.31 Summary: How CMMI Helps Agile 133

- Part IV. CMMI Helping Address Agile Misapplications 135**

- Chapter 6. Common Misunderstandings of Defined Processes
and Agility 137**
- 6.1 What You Will Learn in This Chapter 138
- 6.2 NANO Case Study Background and Problem Faced 139
- 6.3 How NANO Achieved Success and Then Got in Trouble 139
- 6.4 The Positive Side of NANO’s Agility 140
- 6.5 Where NANO’s Agile Approach Broke Down 140
- 6.6 Complicating Factors at NANO 141
- 6.7 Preparing for the Gap Analysis at NANO 141
- 6.8 Gap Analysis Findings at NANO 142
- 6.9 Example of a Generic Practice 142
- 6.10 How Some View Process in Agile Organizations 143
- 6.11 An Example of Process Misunderstanding 144
- 6.12 Another Example of Process Misunderstanding 145
- 6.13 The Good and Not So Good Sides of Distributed
Process Ownership 146
- 6.14 Priority Recommendations at NANO 146
- 6.15 Develop an OPF and OPD Process at NANO 147
- 6.16 Using the CMMI Framework as a Process Roadmap
at NANO 148
- 6.17 Example of Using CMMI Framework as a Roadmap 149
- 6.18 Addressing the Stakeholder Weakness at NANO 149
- 6.19 Maintaining a Successful Agile Culture as You Grow
Requires Training 150
- 6.20 You Can’t Just Use Another Organization’s Processes
and Get the Intended Value 152

6.21	Another Example of Formalizing Informality	152
6.22	Addressing Risk in the Process Improvement Plan at NANO	154
6.23	The NANO Process Improvement Plan	156
6.24	Priority-Based Incremental Deployment Supported by Scenario Training	156
6.25	More on GP 2.7 and Clarifying Roles and Responsibilities at NANO	157
6.26	The NANO Roles and Responsibilities Off-Site Meeting	158
6.27	“White Space” Tasks	159
6.28	An Alternative Approach to Defining Roles and Responsibilities	161
6.29	An Alternative Approach to Tailoring at NANO	162
6.30	Planning with Uncertainty Using an Agile and CMMI-Compliant Approach	163
6.31	CMMI Project Planning Consistent with Agile Planning	166
6.32	Summary: How CMMI Helps Agile	167
6.33	Summary: How Agile Helps CMMI	168
Chapter 7.	Bringing Process Maturity to an R&D Culture	169
7.1	What You Will Learn in This Chapter	169
7.2	GEAR Case Study Background	170
7.3	Common Patterns at GEAR	171
7.4	The Common Pattern of Unclear Process Asset Requirements	171
7.5	Criteria and Product Content Templates	172
7.6	Writing Processes for People in “My Department”	173
7.7	Stakeholder Matrix and Product Template Recommendations	174
7.8	OPF and OPD for Agile Organizations	174
7.9	At GEAR, “No One Has a Hammer”	175
7.10	Another Advantage to Keeping the “How-to” Guidance Separate	175
7.11	Aligning Engineering and Project Management at GEAR	176

- 7.12 At GEAR, “It Depends on Who Shows Up” 177
- 7.13 Does the Written and Trained Process Match the Real Process? 178
- 7.14 Requirements Change Approval Alignment with Real Work 179
- 7.15 Asking the Intent Question Leads to Behavior Change 180
- 7.16 Process Development and Deployment Optimizations at GEAR 181
- 7.17 Advantages and Disadvantages to the “Thread” Approach 186
- 7.18 Process Tailoring 188
- 7.19 Strengths and Weaknesses of Tailoring at GEAR 188
- 7.20 Tailoring Recommendations at GEAR 188
- 7.21 Agile Process Tailoring Guidance: Always Tailor Up 189
- 7.22 Tailoring Down—The Wrong Approach but Used in Many Organizations 190
- 7.23 Why Tailoring Up Makes Sense 190
- 7.24 Will Tailoring Up Solve All Your Tailoring Issues? 190
- 7.25 The Purpose of Criteria and How They Can Help Tailoring 191
- 7.26 Process Compliance Issues at GEAR—The Problem 192
- 7.27 Process Compliance from a CMMI Model Perspective 193
- 7.28 Product and Process Quality Assurance (PPQA) 193
- 7.29 GP 2.8 Monitor and Control the Process 194
- 7.30 Options to Achieve GP 2.8 194
- 7.31 Keeping an Organization “Balanced” Versus Shifting a Culture 194
- 7.32 An Option to Help Achieve GP 2.8 Through Gates 195
- 7.33 “How to” Options to Implement PPQA 195
- 7.34 Recommendations at GEAR: First Step Is, Define the Rules 197
- 7.35 Recommendations at GEAR: Second Step Is, Compliance Checks 197
- 7.36 The Power of Criteria to Aid Agility 198

7.37	A True Story about the Abuse of Criteria	200
7.38	Summary: How CMMI Helps Agile	202
7.39	Summary: How Agile Helps CMMI	203
Chapter 8.	People Challenges Implementing a “Hybrid” Agile Approach in a CMMI Process Mature Organization	205
8.1	What You Will Learn in This Chapter	206
8.2	Introduction	206
8.3	DART Case Study Background	207
8.4	DART Post-Mortem Project Assessment	208
8.5	More Case Study Background	208
8.6	The Way an Agile Approach Should Work with Respect to Task Management	209
8.7	Mistakes Made on DART	210
8.8	Why Didn’t We Prepare AI for His Collaboration Challenges?	211
8.9	More on the DART Case Study	212
8.10	Technique 1: 10 Percent Rule	213
8.11	Technique 2: Scope and Collaboration Management	214
8.12	More on the DART Case Study	215
8.13	How Did I Make the Decision Each Day on What Was Most Important?	216
8.14	More about “Less Visible” Tasks That Require More Time on Agile Projects	217
8.15	More about the Importance of Using a Scope Document	218
8.16	Technique 3: Push-Pull	219
8.17	How Can the CMMI Help Us Implement an Effective Hybrid Agile Approach?	221
8.18	Examples of CMMI Helping Agile Teams Self-Manage	221
8.19	How Is Management Affected by an Agile Approach?	227
8.20	The Importance of Personal Safety to Establishing a Culture of Trust	231
8.21	Summary: How CMMI Can Help “Hybrid” Agile	234
8.22	Summary: How “Hybrid” Agile Can Help CMMI	235

Part V. How Real Performance Improvement Is Achieved 237

Chapter 9. Your Repeating Specific Weaknesses: Finding Them, Why They Are Bad, Eliminating Them, and Keeping Them from Coming Back 239

 9.1 What You Will Learn in This Chapter 240

 9.2 Motivation and Objective 240

 9.3 Using the Same Approach I Use to Help Clients 241

 9.4 Determining the “As-Is” State of My Golf Game 241

 9.5 The Stages of Mastering a New Skill 242

 9.6 A Few Simple, but Critical Steps 245

 9.7 My Golf Swing Repeating Specific Weaknesses 246

 9.8 Repeating Specific Weakness Lessons 247

 9.9 Golf Weaknesses and Analogies to Business 249

 9.10 Agile Approach 250

 9.11 Selecting Specific Checkpoints 250

 9.12 Measurement Objectives and Aligned Measures 251

 9.13 Another Checkpoint on the Golf Improvement Project 251

 9.14 A Critical Distinction: Traditional CMMI and Agile Approach 251

 9.15 Were the Checkpoints for the Three Repeating Weaknesses Sufficient? 255

 9.16 Analysis 256

 9.17 How Did I Address the Problem of My Golf Swing Getting Shorter? 257

 9.18 Rhythm in Golf and High-Tech Organizations 257

 9.19 What Business People Can Learn from Golf Professionals ... 259

 9.20 How the Checkpoints Helped to Achieve the Golf Project Goal and More 260

 9.21 Revisiting CMMI Level 4/5 Practices and Their Relationship to Agility 262

 9.22 Summary: How Agile Can Help CMMI 263

Chapter 10. Summary and Conclusion	265
10.1 What You Will Learn in This Chapter	265
10.2 What Can We Learn from the Case Studies in This Book?	265
10.3 What Have We Learned from NANO and GEAR?	268
10.4 What Have We Learned about Measurement?	269
10.5 What Have We Learned by Thinking Out of the Box (Golf Project)?	270
10.6 The Value of Small Changes to Aid Real and Consistent Performance	271
10.7 Supporting Small Changes in Business: The Two Sides of Tailoring and Criteria	272
10.8 Conclusion	274
Epilogue: What Does Passion Have to Do with Performance?	279
Appendix A. Twelve Principles Behind the Agile Manifesto	285
Appendix B. Example Agile Project Management Plan (PMP) Template	287
Appendix C. Example Agile Organizational Process Asset Guidelines	293
Appendix D. Example Agile Process Asset Approval and Release Process	297
Appendix E. Example Agile Organizational Process Focus Process	299
Appendix F. Example Agile Organizational Process Definition Process	303
Appendix G. Terminology Used in This Book	307
References	309
About the Author	313
Index	315

This page intentionally left blank

Foreword by Mike Phillips

As I write this, the CMMI Product Team is crafting our next release of CMMI models—CMMI V1.3. A critical element of the V1.3 release is to improve the models' coverage of the unique elements that Agile methods have provided to accelerate software development in innovative ways. We on the product team believe that the synergies that Agile methods and CMMI models have when used together demands this expansion of model coverage. In our update of CMMI models for V1.3, we were directed by criteria that required that we minimize the models' growth. (As of this writing, we have over 110,000 people trained in CMMI models, and over 4,000 organizations that have demonstrated their adoption of the practices using CMMI benchmark appraisals. Therefore, a large amount of change would require them to be retrained and reduce the overall benefit to users.) The product team has chosen to add supporting material to process areas that have the strongest correlation with Agile methods, and where Agile methods might be perceived as significantly different in approach from CMMI practices.

Release of this book precedes the release of Version 1.3 models. The book provides some of the key insights from Paul's work with a number of organizations to show ways that CMMI and Agile methods can effectively be teamed for success. The collection of examples that Paul uses to illustrate effective process improvement confirms our conclusion that these two approaches are complementary and are not, in fact, competitors. Each can complement the other and add value to any organization's development efforts.

Paul's book allows you to gain many more hints, tips, and insights than we can ever include in a CMMI model. I particularly like the mixture of "myths," "lessons," "insights," and straightforward "questions and answers" that he has sprinkled throughout the book. The lessons from real organizations that he has renamed to be RAVE, BOND, LACM, NANO, and GEAR, are part of Paul's delightful way of sharing his consulting experiences with you. Each

of these lessons provides a potential “takeaway” for your process improvement journey—ideas that will make your application of Agile methods and CMMI better for having read them.

With this book, Paul is providing invaluable leadership that will fuel the move forward with various mixes of Agile methods and CMMI models. Your use of the tools and techniques captured within this work will enable you to join us in the effort to grow these ideas and improve your organization’s performance.

—*Mike Phillips*
CMMI Program Manager

Foreword by Hillel Glazer

Looking back at writing and discussions that brought CMMI and Agile concepts together, arguably, the conversation, at best, entered the mainstream between the years 2004 and 2006. As of the publication of this work, many people are still skeptical about whether CMMI and Agile can truly co-exist. (I suppose, when there are more than 20 years of history associated with a brand (e.g., SEI's CMM), it might take a few more years for the broader market to catch on. Thanks to social media and the Internet, it will hopefully take less than two decades to drill the message into people's minds.)

There are several things I like about Paul's book, which make it worthwhile reading for people interested in this topic. His case studies are typical across many types of companies and many situations. As I reviewed an early version of the work, I found myself believing he was working with many of my own clients and former employers. Prior to reviewing the manuscript, I had never met Paul or collaborated with him. For us to have such similar experiences merely provides further evidence that Pareto was right: 80% of the problems *can* be explained by 20% of the issues. The cases Paul describes can be easily related and extrapolated to many organizations. Even if/when his case studies don't match a reader's experience precisely, that doesn't mean they are not relevant or there aren't lessons to be learned and applied.

Another attribute I like is that Paul seems to spend most of his efforts in these client cases on the basics that are common to CMMI and to Agile. In particular, he dutifully applies Lean principles and practices to empower Agile practices and facilitate CMMI practices. If I had to point to one "take away" from this book, it would be this.

I should point out that, like many of us, Paul's been doing this for a while. His experience pre-dates the named "Agile" movement just as "Lean" is a progenitor of the Agile movement. An important meta-observation about Paul's work, in general, is that it often takes an expert like Paul to effectively

(and objectively) bring “Lean” principles into a software development organization. There’s something about the manner in which software and “processes” have been brought together over the years that have established many challenges in this space. Paul demonstrates several techniques to create conditions that allow for both flexibility and disciplined improvement that are worth emulating—both as a consultant, in general, but with respect to Lean principles, in particular. Readers without a firm grounding in “Lean” principles and practices would be well advised to have a guide or coach to try them out the first few times.

The same things I like also carry precautions to the broader reader audience. One thing is clear about both the companies Paul included in his cases, and about Paul’s approach: improvement as a business driver is a key to success. Implementing CMMI for the ratings or “Agile” for bragging rights won’t work. It must be human nature that causes people to continue to seek “silver bullet” solutions to their business challenges. Were there such solutions, there would be no challenges. Paul’s techniques and approach were adapted from his experience addressing his clients’ needs. They were not prefabricated in Paul’s office and then installed in his clients’ conference rooms. Paul generated appropriate solutions in the context of his clients’ needs. The caution is this: experiment, inspect, and adapt. For either CMMI or Agile to benefit an organization or from each other, and, for either to truly take advantage of experimenting, inspecting, and adapting, there are several attributes an organization must embody: self-awareness, learning, brutal honesty, trust, and refusal to settle for mediocrity as a goal. Organizations who merely try to copy Paul’s work clearly don’t “get it.”

Enjoy the work and I hope you all achieve your state of “excellence.”

—Hillel Glazer, Principal and CEO, Entinex, Inc.
CMMI High Maturity Lead Appraiser

Preface

Why You Should Read This Book

This book explains why combining an Agile approach with the CMMI¹ process improvement framework [1] is the best route to quickly achieve your business objectives² and it gives you practical and proven techniques to do it. But the book's greatest value might lie in its insights into how real performance improvement is achieved by focusing on "repeating specific weaknesses" that tend to be unique and closely related to culture in each organization. The book also provides

- Proven alternatives to traditional approaches to implement CMMI practices that can increase your agility
- Proven criteria to help make timely and effective decisions
- Proven techniques to extend Agile methods to Systems Engineering and Project Management
- Big picture insights, lessons, and cautions
- Specific "how-to" examples to quick-start a successful Agile and CMMI integration
- Common mistakes to avoid when implementing an Agile approach

First, to understand why more companies are not jumping at this great opportunity, you need to understand the problem.

1. The Capability Maturity Model Integration (CMMI) is a process improvement maturity model for the development of products and services developed by the Software Engineering Institute (SEI).

2. An organization's "business objectives" might not include "process improvement." Why it is important to start with business objectives is discussed in Chapter 2. Examples of business objectives are provided in Chapter 3.

The Problem

The mistaken belief persists that the Capability Maturity Model Integration (CMMI) and Agile approaches are at odds. In a Technical Note³ appearing on the Software Engineering Institute (SEI) Web site in November 2008 [2], a call to action is issued to both Agile and CMMI camps. CMMI experts are encouraged to engage the Agile community by including examples from multiple types of organizations. Agile experts are encouraged to learn about the CMMI and how its practices can complement Agile practices. The authors of the Technical Note universally agree that Agile methods and the CMMI “can not only coexist, but successfully integrate to bring substantial benefits to both Agile and traditional software development organizations.”

Why Conflicts Continue to Arise

One reason for many of the conflicts that arise when using the CMMI together with an Agile approach traces back to the origins of the CMMI found in the development of its precursor CMM model. As stated in the referenced technical note:

If we look at the genesis of the CMM, it predates the internet and nearly everything associated with internet technology. For that matter, CMM predates many software development, deployment, and infrastructure technologies, languages, and methods...

...In today's frequent discussions of increasing globalization and the important role played by trust in making effective collaboration happen across stakeholders, one might describe such a development context as exhibiting low trust. Users were typically not direct contributors to the evolution of the end product prior to field-testing. They instead had to depend on the contracting relationship, requirements, and standards to deliver the product they needed. These comments may be an over-generalization, but they are intended to summarize the DoD software acquisition environment that existed at the time. Further, these comments explain why the practices in the CMMI sometimes exhibit some of these same high ceremony and low trust characteristics found in the high-risk, government-contractor environment in which software failure could equal lives lost.

3. Technical Note CMU/SEI-2008-TN-003, November 2008, “CMMI or Agile: Why Not Embrace Both!”

Another reason for many of the conflicts is the differing views on just what “Agile” is. Some view “Agile” simply as quick when making a decision or light when it comes to writing things down, but these popular misunderstandings of agility have led many organizations down unsuccessful paths.

Why I Wrote This Book

I wrote this book to help bridge the chasm described previously. Through this book, I explain where the heart of the conflict exists, and what you can do about it. A fundamental claim made through the case studies is that:

Most of the conflicts that arise between the CMMI and Agile are based in either a historical view of what a “good practice” should look like when implemented—which may no longer be accurate given the world we live in today—or a misunderstanding of what “Agile practices” really are and how they should be executed.

It is my hope that CMMI experts, including lead appraisers, will consider this material and potentially re-think messages that might be being inadvertently shared related to what a “good CMMI-compliant” practice should look like when implemented. It is also my hope that organizations currently *misapplying Agile concepts* will begin to understand where their practices are deficient and see how the CMMI could help them locate their right level of agility given their business situation.

Throughout this book, I share numerous examples of how the CMMI can help Agile, and how Agile can help the CMMI.

How CMMI Can Help Agile

One goal of the book is to expose characteristics of Agile misapplications common in growing “Agile-like”⁴ organizations and share how the CMMI can help these organizations by providing “reminders” of critical practices that frequently lose visibility as organizations grow and project pressures rise. I also share how the CMMI can help even successful growing organizations that are applying fundamental Agile practices as intended.

4. When I use the phrase “Agile-like” or “wannabe Agile” in this book, I am referring to organizations that are trying to use an Agile approach but are missing key ingredients of true agility.

How Agile Can Help CMMI

In this book, I also provide numerous options to traditional “how-to” approaches to implement CMMI practices. Some of these options are not well known, and in one personal case study, I present some “out of the box” thinking with respect to the use of the CMMI to help an organization move beyond consistency to the kind of performance required to effectively and continually rise above the competition.

What This Book Is Not

This is not a book about the fundamentals of the CMMI, nor is it a book about Agile methods such as Scrum, Extreme Programming, and the Crystal Methodologies—although you will read about lessons learned from applying these software methods as well as many proven systems engineering and project management techniques that evolved consistent with these methods.

What This Book Assumes about the Reader

Some of the chapters in this book assume the reader is familiar with either traditional CMMI-based development and management approaches or Agile development approaches, and is interested in learning how the other could be used effectively to help an organization achieve its business objectives fast.

The book is intended for managers at all levels, systems engineers, software engineers, and process professionals in large and small organizations that currently employ traditional CMMI-based processes, Agile methods, or a mix of both. The book is also intended equally for both CMMI and Agile experts, as well as less experienced personnel, and those just starting out with new process improvement initiatives looking for the most effective implementation approach for their organization.

How This Book Is Structured

In this book, I share six major case studies, each with related lessons, insights, myths, and cautions. *Lessons* contain key fundamental information. *Insights*

contain key information that might require deeper reflection by the reader. *Myths* contain a belief about the CMMI model or an Agile approach that most people know is not true, but that organizations often treat as though it were. *Cautions* raise awareness of commonly observed pitfalls.

In Chapters 2 through 10, you will find 16 insights, 15 myths, 16 cautions, and 62 lessons. Lessons are numbered sequentially within each chapter. The book is structured into five major parts. Part I provides an introduction and a CMMI and Agile primer. Part II focuses on techniques to help CMMI process mature organizations increase their agility. Part III demonstrates how a successful Agile organization can increase its CMMI process maturity without compromising the agility that has brought it success. Part IV provides multiple examples demonstrating how the CMMI can help organizations that are trying to be agile but are missing key ingredients of true agility.

Part V focuses on the role of *repeating specific weaknesses* in achieving real performance improvements. Chapter 9 is intended to help you think a little “outside the box” by demonstrating the use of an Agile approach together with key CMMI practices to help solve a non-work-related challenge. Through this personal challenge, I draw some nontraditional conclusions—but conclusions backed up by case study data. This case study takes us beyond the fundamentals, examining how real “consistent high performance” is best achieved. This story brings us closer to the personal side of process improvement, and looks at how great organizations continually outperform the competition. In the concluding chapter, we step back, summarize what we have learned from these case studies, and provide an insight into real and consistent performance.

How Different Audiences Can Use This Book

This book can be used by different audiences in multiple ways. First, executives and senior managers looking for the big picture are encouraged to read the introductory material at the start of each of the five major parts of the book. Then scan the book, focusing on the *Scenarios* at the start of each chapter, the “What You Will Learn in This Chapter” paragraphs following the *Scenarios*, the highlighted *Insights*, *Lessons*, *Cautions*, and *Pause, Reflect, and Glance Forward* features throughout the chapters, and the summarizing tables at the end of each chapter entitled “How Agile Helps CMMI” and “How CMMI Helps Agile.” You can then go back and read more specific case

study information related to topics of greatest interest. You can also use the Roadmap in the Part I Introduction to help locate specific key information.

Second, technical leaders and developers looking for a deeper understanding can read the full case studies, which provide the rationale for approaches taken and the thought process we went through in applying the CMMI model to varying situations. This level of detailed information is necessary to understand why the options were chosen within each of the specific organizations. This information can in turn help you make better decisions given your own situation.

Third, process professionals and those looking for more detailed “how-to” information should first take the time to digest the case study information, understanding both what was done and why. This will lead to “how-to” questions. To help with the “how-to,” specific examples are provided in the appendices. These “how-to” annotated examples are referenced from footnotes within the case study chapters and can help your process improvement effort get started on the right track toward a successful Agile and CMMI integration.

Fourth, the novice (i.e., software engineer fresh out of college or college student) or those looking for the fundamentals are encouraged to first read Chapter 1, the introduction, and CMMI/Agile primers. Then, after reading the Summary tables at the end of each chapter, read Chapters 4, 5, and 8 and Part V. Chapters 4 and 5 provide a good foundation in fundamentals, while Chapter 8 demonstrates some of the most common challenges observed in traditional organizations when initially attempting an Agile approach, along with practical and proven solutions.

Bringing Process Maturity to Agile Organizations—Part I

Scenario: You are a small Agile organization that is successful and growing, but to date you have few documented processes and no formal training program for your people. To maintain your success as you grow you are going to need more process discipline. You would like to start a CMMI process improvement effort. However, you fear losing the Agile culture that has led to your current success. So what should you do? What options do you have?

4.1 What You Will Learn in This Chapter

- Five popular myths about processes in Agile organizations
- Common challenges faced initiating a CMMI process improvement effort within an Agile organization
- Successful techniques to guide a small growing Agile organization to CMMI level 3 while maintaining an Agile culture

- Answers to common questions related to developing Agile processes
- Practical techniques to structure an organizational repository supporting agility and CMMI compliancy

Section I

Key Case Study Points¹

4.2 BOND Case Study Background

In July 2007, I participated in a formal CMMI appraisal with the goal of achieving a full-staged (18 process areas) CMMI level 3 for a client I will refer to as BOND. I began helping this client years earlier when they had virtually no written processes, or training, and only 25 people. The company, which was started by two retired military men, had been rapidly growing at a rate of over 30 percent a year reaching over 150 people by the time of the 2007 appraisal.

The key challenge I was presented with at the onset was to help the organization add the needed process discipline the CMMI could bring to help them continue to manage their projects effectively as the organization grew. The owners also stressed the importance they placed on maintaining the successful Agile culture that they felt was an important component of their business success.

After I initially executed a gap analysis (I will explain what a gap analysis is shortly) against the CMM model for this organization in 2001, they attempted for a few years to move forward with their process initiative on their own, but were unsuccessful.

In 2003, I executed a second gap analysis (this time using the CMMI model). Subsequent to the presentation of my gap analysis findings to Senior Management, I was asked to become more involved in assisting the organization's process improvement effort.

They asked—as many clients do—if I had CMMI-compliant processes that could expedite their CMMI goals. I replied that I could help them develop

1. While the approaches described in this case study work for an Agile organization, they are actually intelligent ways to work in any organization.

their own processes addressing the areas the CMMI expected, and that I could share what I referred to as “starting point CMMI-based process templates.” I also emphasized that we wouldn’t achieve the goal they were searching for if we tried to use these process templates without taking the next important step. Now, let me explain what the next important step is and how we executed it to help BOND achieve their CMMI level 3 goal.

4.3 What Is a Gap Analysis and Why Is It Crucial for Agile Organizations?

Whenever I am asked to help a small Agile organization improve its process maturity, I always recommend we start with a gap analysis against the CMMI model.² The purpose of a gap analysis is to assess where an organization currently is from a process perspective and identify gaps based on the CMMI model. The result is a strengths and weaknesses report and an initial set of recommendations to help the organization achieve its current process goals.

When I present weaknesses I have observed based on the CMMI model practices, I always stress that these might or might not be actual weaknesses in the organization that require actions. Part of the follow-on plan always includes more analysis of these “potential weaknesses” to determine the proper course of action given the organization’s business situation and process needs.

Executing a gap analysis is important for any organization initiating a process improvement effort because it facilitates the most effective plan based on the correct priorities for that particular organization. I now want to share the key points on how I conduct a gap analysis for an Agile organization, and why the approach you use when doing a gap analysis is crucial when it comes to agility. This will lead to a discussion of additional techniques I use to help Agile organizations move forward with a successful CMMI process maturity effort.

2. The discussion to follow on a gap analysis and running a process improvement effort like a project relates to the expected practices in the Organizational Process Focus (OPF) process area in the CMMI model. Examples of OPF and Organizational Process Definition (OPD) processes are provided in the appendices to this book.

4.4 Keys to Conducting a Gap Analysis for an Agile Organization

There are multiple approaches to conducting a gap analysis. You can focus on documentation including the products an organization produces, and documented processes employed in developing those products. You can also spend time interviewing people in the organization who use those processes. I have seen a gap analysis conducted using exclusively the documentation route, and at times, this can make sense. Most often, a traditional gap analysis focuses on the documentation, supplemented with a few interviews.

When I do a gap analysis for an Agile organization, I switch this traditional emphasis from the documentation to the discussions with the people. The way I conduct these interviews is crucial to the success of the approach.

I conduct my interviews individually, not in groups as is often done with more formal CMMI appraisals. I am particularly careful how I phrase my questions during these interviews. I keep the interviews informal with an emphasis on letting the people being interviewed just talk about how they do their job. I have found that by phrasing questions as simply as possible, most people tend to talk openly and with ease about their job. An interview question I often start with is:

Can you tell me how you do your job?

I spend most of my time taking notes, letting the employee speak. My follow-on questions flow naturally from responses that lead me to dig deeper. I don't use any of the words from the CMMI model in asking the questions, but I do keep the model practices in mind. I am using those practices to trigger more detailed questions based on what I hear.

Late in the interview after I have learned how they view their responsibilities and carry out their activities to achieve those responsibilities, I ask:

Do you follow a process when you do your job?

Almost everyone in Agile organizations that have just begun a process improvement effort answers that question with:

No.

By the time I ask that question, I already know the answer, and most of the people have answered it incorrectly.

By this time, I have in my notes a great deal of the information that describes the process they actually do follow when doing their job. They, of course, when asked that question assume I mean a documented process.

I assess what they tell me they do against the CMMI model, and against whatever written processes exist. I look at examples of the products they produce to corroborate what they are telling me and what their documented processes say.

When I out-brief a client with strengths and weaknesses against each process area of the CMMI model, each point I make is backed up with objective evidence from what I heard in an interview and/or saw in documentation. What I hear in interviews and see through documentation—along with my own experience based on patterns I have seen in similar organizations—is shared in my report and serves as the objective data that leads to my recommendations. I always stress in my report that any weaknesses identified against the CMMI model are “potential weaknesses” to the business.

My reports go much deeper with detailed examples than most traditional gap analysis reports. This approach is counter to what is usually done partly because of nonattribution concerns. It is important that I don’t attribute specific findings to individuals in order to maintain an atmosphere in which people are willing to talk openly about their jobs.

However, too often valuable findings are raised up to an abstract set of statements leading to ultimate findings that become almost useless in helping the organization focus on the specific priority improvements needed.

Furthermore, it has been my experience that when a gap analysis does not provide specific examples with details backing up conclusions, Senior Managers do not place much value in the report resulting in minimum value to follow-on improvement efforts. See Table 4-1 for pros and cons of different gap analysis approaches.

Table 4-1 *Pros and Cons of Different Gap Analysis Approaches*

Gap Analysis			
Approach	Advantage	Disadvantage	Comment
Traditional Documentation Focus	Learn “gaps” if you followed documented processes	Don’t gain insight into real processes followed by people	Behavior change is the most difficult process improvement

Continues

Table 4-1 *Pros and Cons of Different Gap Analysis Approaches (Continued)*

Gap Analysis Approach	Advantage	Disadvantage	Comment
Agile Interview Focus	Learn the real process the people are following	Takes more effort requiring more analysis and digging	Leads to uncovering where the most valuable process improvements lie

Let me now give you a simple example of why I stress weaknesses identified in a gap analysis are “potential weaknesses” to the business and how we determine if these “potential weaknesses” require actions to resolve in the plan to move forward.

4.5 Example of “Potential Weakness” Against CMMI in an Agile Organization

Somewhere during every interview as we are talking about how the individual executes his or her job, we get to the products they produce as part of executing that job. Eventually, I ask:

Who else looks at these products you are producing?

This discussion leads to the question about whether they conduct peer reviews on their products. Often the answer I get in Agile organizations is:

We don’t do formal peer reviews on our products.

On the surface, this triggers a “potential weakness” against the CMMI model because peer reviews are a specific practice in the Verification Process Area of the CMMI model. We don’t have enough time to dig into each area I identify as a potential weakness during the one-hour interview. In most areas where I find potential weaknesses, I just make a note that those areas require more investigation and probably further discussion.

As an alternative, I could just list as part of my report all the areas my client must fix to “comply” with the CMMI model. I could tell them I heard you don’t do peer reviews and you need to do peer reviews because it is an expected practice within the CMMI. This is actually how I have observed the CMMI model used in many organizations. It is an example of using the model

in a prescriptive way. This is not the way the model was intended to be used by its authors, nor would this approach help achieve the goal my client is looking for.

If I were to use the prescriptive approach each time I found a potential weakness against the model, I would “impose” something extra for the organization to do, and therefore add work on top of what they already do without fully understanding the value of that added work.

This approach, in my view, would be a huge mistake particularly in a successful Agile organization that is relying on their existing proven “Agile culture” to continue to bring them the success they have achieved in the past.

This approach may appear to be the most direct way to prepare the organization for a formal appraisal. It would also be the easiest thing to do as a consultant because it requires the least amount of effort.

However, from experience I know it is also the fastest way to raise the risk of driving this organization away from its Agile culture, leading it to a less efficient process than it currently has. Each time I take this approach to a potential weakness, I raise the risk of making this organization less competitive in the future.

I have observed that many process improvement professionals take this approach, and I understand why. It is natural to assume that people who developed the CMMI model are probably smarter than most process people are and the likelihood is that most organizations should be complying with whatever expected practices exist within the model.

What is frequently missed in this line of reasoning is the following *implied myth*:

MYTH The CMMI developers understood when they came up with the model all the business situations where the model might be applied.

This myth rests at the core of why we so often hear that Agile approaches conflict with the CMMI. When the model is used this way we are inappropriately utilizing the model to dictate implementation, or “how to” issues the model was never meant to address.

I will explain further how to handle these apparent conflicts as they arise, and why the vast majority turns out to be no conflict at all. First, we need to discuss the recommended plan to move forward subsequent to the gap analysis.

4.6 Running Process Improvement like a Project

At BOND, part of the plan forward was to run the process improvement effort just like any other project in the company. I worked closely with the assigned Process Improvement Lead inside the company building a project plan with a schedule, tasks, and assigned resources.³ We used the Continuous Representation of the CMMI model and decided to prioritize process areas and attack them incrementally.

The Project Management process areas were identified as the highest priority and attacked first during the initial increment of work. To address each process area we used a tailored version of the Technical Working Groups (TWG) approach recommended by the SEI [21]. While the fundamental TWG approach is sound, there are lessons I have learned applying this approach to develop CMMI “compliant” processes that fit within an Agile culture.

4.7 TWG Approach for Agile Organizations

The purpose of a TWG is to use key subject matter experts (SMEs) in the organization to help develop, document, and deploy processes and related process support assets across an organization. In observing TWGs in the past

Pause, Reflect, and Glance Forward

If you are experienced with Agile approaches but are new to CMMI, you might be asking at this point: “If this organization is successful using an Agile approach, why go through all this effort?”

We will begin to answer this question in the next chapter where we discuss the added value the CMMI can bring to a successful Agile organization.

This subject will also be addressed further in Part IV where we investigate common misapplications of agility.

in multiple organizations, I have found common patterns I like to avoid when implementing this approach in an Agile organization. Those patterns have led to a tailoring of the TWG approach for Agile organizations, which are described in the following paragraphs.

One of the responsibilities of a TWG is addressing any potential weaknesses against the CMMI model that might have been identified. Another is to ensure the people in the organization who must use the process and supporting process assets are trained in those processes.

3. Refer to appendices for an example of a template for a Project Management Plan.

The primary goal is to help the organization become more successful, or maintain its current success. However, a secondary goal is to ensure that when the formal CMMI appraisal happens, the organization is prepared to demonstrate through both objective documented evidence and interviews that they have achieved the intent of the practices in the process area.

Training and process deployment are included under the responsibilities of a TWG because often in the past, these critical efforts have fallen through the cracks in many organizational process improvement efforts.

When a new process is first developed, those who were closest to its development are best equipped to provide the rationale for key decisions and share how the processes are intended to be used.

LESSON 1

Hold those responsible for developing processes also responsible for training those processes at least during the pilot project and initial organizational rollout.

Some organizations operate as if the following myth is true:

MYTH If an organization is Agile, it requires less process training.

You need to communicate the rationale for your processes. There is no one better equipped to explain why things were placed in a process than those who developed them. Too often, this critical knowledge is lost after a process development working group is disbanded. It is the rationale that leads to the needed *buy-in*, which is critical to ensure the organization achieves the intended value and the people are not just “going through the motions” to comply.

When you bring CMMI process maturity to an Agile organization by maintaining the Agile culture within their documented processes, you also need more—not less—training. The reason for this is that the Agile documented processes we develop will not address every possible scenario that is likely to arise in the use of the process. These processes must be supported by *mentoring* and *on-the-job* assistance especially during the period of initial deployment.

4.8 Revisiting the Goal and Challenges on the Process Improvement Project

The goal at BOND on the process improvement project was multifold. First, it was to help the project leaders manage their projects effectively as the organization grew. Second, it was to move the organization forward toward the achievement of a formal CMMI level 3 as rapidly as possible, but without adding significant risk to their ongoing business. This meant the TWGs had to keep an eye on the CMMI model practices addressing potential weaknesses. We also had to be sensitive to the use of key people in the organization who were actively engaged, often working closely with customer counterparts on critical projects.

Third, we had been given the added challenge by Senior Management to maintain the Agile culture the owners felt was critical to the organization's success to date. To accomplish this, I added a requirement for the TWGs. If we were to add activities to the existing processes in the organization, the TWG would have to provide the rationale during the training as to why this activity added value to the organization.

This led to some interesting discussions among TWG members. Some argued that we should be able to just tell those being trained that the CMMI required it and that was sufficient rationale. I objected to this line of reasoning.

I explained to each of the TWG members that the CMMI requires you to make conscious decisions related to certain practices based on your business needs. Any decision we made based on a CMMI practice should be explained during the training from a BOND business need perspective. While this approach led to more time being required by TWG members to discuss current processes and potential weaknesses it helped the organization *reason about its own processes* and determine what the right processes were given their current business and the anticipated potential growth.

Fundamental Rule: Always Ask the Intent Question, and Then Keep Digging

The first Fundamental Rule of our Agile TWG at BOND was based on something a lead CMMI appraiser once told me:

Always ask the intent question.

What she meant was, when assessing an organization against a practice in the CMMI model, ask yourself:

*What is the intent of this practice?*⁴

Another phrase the lead appraiser often used was:

You don't want to create unnatural behavior in the organization.

This approach leads to another question:

Is the organization achieving the intent?

If the answer is yes, but they don't appear to be following the expected practice, the next question is:

How are they achieving the intent?

and:

What activities are they following to achieve the intent?

The approach of asking these questions fits with our goal to maintain the "Agile culture." The Agile culture is a natural culture where people follow practices that have been proven to work in getting their job done successfully. BOND had a history of success, so whatever practices they were following were, for the most part, working. This was our starting point to extract and document the right processes for this organization.

4.9 Alternative Practices and Tailored Agile TWG

The approach described may lead to an *alternative practice*. An alternative practice is defined by the CMMI guidelines as, "A practice that is a substitute for one or more generic or specific practices contained in CMMI models that achieves an equivalent effect toward satisfying the generic or specific goal associated with model practices. Alternative practices are not necessarily one-for-one replacements for the generic or specific practices." However, my experience when digging "looking-for-intent" or "equivalent effect" has been that most often, you don't arrive at an alternative practice, but rather a different implementation of an expected practice.

4. The informative material within the CMMI model is the best source to help in determining the intent. Caution should be used when supplying one's own intention.

LESSON 2

Always keep in mind that the CMMI is primarily about “what you are expected to do,” not “how you do it.”

The “how you do it” should always be open for discussion. By keeping Lesson 2 in mind as the TWGs dig deeper in discussion, they are opening options they might not have previously understood existed in terms of “how” a given expected practice in the CMMI model can be legitimately achieved.

Another good question to ask yourself as you are digging is:

Is there a problem in the organization because this practice as we are reading it in the CMMI model does not appear to be followed?

INSIGHT If there isn’t a problem in the organization related to a given expected practice, it is likely the intent of the practice is being achieved. Keep digging and you will uncover what that technique is and probably find something worth sharing with others in your organization.

One valuable side effect of “digging deeper” is that often these TWG discussions lead subject matter experts to uncover what I refer to as a “local” practice. A “local” practice is one that works very well to achieve a given CMMI expected practice, but the practice just grew up as part of the organization’s culture and wasn’t even viewed by most as part of any “process.”⁵

These “local” practices are often found in organizations where culture is taken for granted. I have in fact discovered many such practices during a gap analysis and then reiterated them with TWGs afterward, reminding them of what they had told me during the interviews. This kind of memory jogger has been one of the main reasons I like to sit in on client TWGs at times to help facilitate the process and remind them of their own processes.

Other common examples of powerful processes in Agile organizations often taken for granted include brainstorming sessions on white boards, maintenance of informal team task lists, and early product demonstrations with customers. These are all examples of real processes that work, can be documented, and can be shared across the organization.

5. Examples of “local” practices discussed later in the book include the “Undocumented super-spreadsheet” resource management process, and “Doorway” risk management process.

Questioning and digging is the major difference in how the Agile TWG operates over traditional TWGs.⁶ The focus of the Agile TWG is digging to uncover the real activities that are being followed and used successfully in the organization—not to create new ones. Now let us return to the Peer Review example to learn more about how this TWG process works.

4.10 Returning to the Peer Review Example

What is the intent of the specific practices in the Verification Process Area related to performing peer reviews? The tips in the CMMI guidelines book give us good hints that can help us understand intent. In the Peer Review case, they tell us “peer reviews provide opportunities to learn and share information across the team,” and “many different types of reviews might be considered.” The text also tells us that the purpose of peer reviews is to:

Identify defects for removal and recommend other changes that are needed.

This information leads us to ask some different questions, which we did at BOND. When I asked:

How do you identify defects for removal and get recommendations for other changes that are needed?

I heard:

We demonstrate our products early and often to our customers.

and:

We meet daily with our teammates and discuss openly the work we are doing. Our products are checked into a library every day where others can see them and are encouraged to provide feedback. And they do.

As I listened to the answers, I realized that when they said they didn’t do “formal peer reviews” they meant they didn’t have a single defined time when people went into a conference room to provide feedback on a product. However, they did achieve the intent of “peer reviews” by doing continual “less formal” peer reviews throughout the development. This is a common practice in many Agile organizations.

6. See “Effective Techniques to Run an Agile TWG” later in this chapter, and the “Thread” Approach to Process Development and Deployment in Chapter 7, GEAR case study, for more information on running Agile TWGs.

This is an example of digging for the real process that is followed to achieve the intent of a given practice. At BOND after this discussion by the Verification Process Area TWG, it was decided that the process did need to be documented, but that it wasn't an alternate practice at all like first thought.

They were just using different “how to” techniques to “share information across the team” and “peer review” products. While this had been a concern early in preparing for the formal appraisal, it turned out there were no issues raised during the formal appraisal about peer reviews at BOND.

“Convenient, but False Arguments”

While BOND was successful, no company is perfect. Therefore, as you ask the intent question and conduct related discussions, I recommend that multiple people participate, including Agile knowledgeable and CMMI knowledgeable people, and others that might be independent of the organization to ensure the group is not creating “convenient, but false arguments.” An example of a “convenient, but false argument” would be an organization that claims it does continuous team reviews on its products, and/or frequent and early product demonstrations with the customer, but doesn't follow through in a disciplined way when conducting these activities.

CAUTION

Beware of “Convenient, but False Arguments”

This situation can usually be uncovered by asking questions to determine if there is a related problem in the organization.⁷

4.11 Tailored TWG Techniques and Lessons at BOND

Let us now discuss a few more key techniques used at BOND in conducting the TWGs to document and deploy Agile processes along with a few lessons we learned to help the TWGs run more effectively. Among these techniques and lessons you will see more examples of asking questions and digging leading to more typical Agile “how-tos” that often just needed to be documented.

7. Another example of a “convenient, but false argument” is provided in the NANO case study in Chapter 6 related to the need for training (GP 2.5).

Some are examples where documentation and minor additional behavior changes were required.⁸

4.12 Preparation Work for Running Agile TWGs

When you are first preparing your organization to conduct Agile TWGs, you don't need to involve all the subject matter experts who will eventually be needed to help define your processes. The first few tasks to complete before the TWGs get going revolve around establishing the structure of the organizational repository and the process assets. These are discussed in the following paragraphs.

An Agile Organizational Repository Structure

The CMMI does not prescribe a structure for the organizational repository.⁹ The Process Improvement Lead at BOND with my assistance established the organizational repository structure. Unless there is a good reason for a different structure (e.g., non-CMMI process requirements), I recommend establishing a repository structure that aligns with the process area categories in the CMMI model. For example, the structure could be partitioned by Engineering, Management, and Support. Process Management could have its own partition or be included under Management. This decision is ultimately up to each organization and should be made based on legacy process structure, ease of use, and organizational culture. It is recommended that the repository structure not be structured to align with a specific organizational structure since organizational structures tend to change.

4.13 Packaging of Processes

Processes do not need to align one for one with CMMI PAs. Many organizations do this, but it is not necessary. This decision is best made based on how

8. More significant behavior issues that needed to be addressed to achieve the full CMMI level 3 are addressed in the next chapter.

9. The discussion to follow in this book on the organizational repository structure and packaging of process assets relates to the expected practices within the Organizational Process Definition (OPD) process area of the CMMI model.

you do real work in your organization. You don't need to make the final decision for process packaging at the start of your process improvement effort. In fact, the brainstorming within TWGs may lead to the identification of processes that should be broken out separately, and processes that should be consolidated.

At BOND, the Technical Solution (TS) TWG broke out two distinct processes referred to as Design and Implementation. Verification and Validation were consolidated into one process, which is common in Agile organizations because the practices Agile organizations use for Verification and Validation tend to have significant overlap. This is because a common Agile technique is to develop complete slices of functionality in short increments, often leading to product demonstrations to the customer. As a result, Verification and Validation techniques tend to blend in such environments.

There was significant discussion over Project Planning (PP) and Project Monitor and Control (PMC) at BOND. The TWG ended up keeping these processes separate, although in other Agile organizations I have seen these consolidated. The factors to consider when making the decision to keep PP and PMC separate versus consolidating include the maturity of your organization's planning and project management activities.

In organizations where the project planning, monitoring, and control activities are sound and institutionalized, it can be more efficient to consolidate and train these processes together. This is because the expected practices under PMC align closely with those under PP and therefore can naturally be packaged and trained together. PMC expected practices revolve around monitoring and taking appropriate action associated with each of the items in your project plan. However, if your organization is just learning how to develop a project plan, it might be more effective to maintain distinct processes so each gets its proper focus.

Risk Management (RSKM) is usually broken out into its own process area, although in implementation in most Agile organizations it is frequently integrated with project planning, monitoring, and control. For example, most Agile organizations do not have distinct risk management review boards. The risk management reporting is usually integrated with project monitor, control, and reporting to Senior Management. Refer to Table 4-2 for an example of an Agile organization's eleven process descriptions and how they could provide coverage for all eighteen CMMI level 2 and 3 process areas.

Table 4-2 *Example Agile Organization Processes and CMMI Process Area Coverage*

Sample Agile Organizational Processes	CMMI Level 2 and 3 Process Area Coverage
Organizational Process Focus	OPF
Organizational Process Definition	OPD
Organizational Training	OT
Consolidated Management Process	PP, PMC, RSKM, IPM, DAR, MA
Supplier Agreement Process	SAM
Consolidated Requirements Management/Development Process	REQM, RD
Design Process	TS, DAR
Implementation Process	TS
Integration, Test, and Validation Process	VER, VAL, PI
Configuration Management Process	CM
Quality Assurance Process	QA

4.14 An Agile Organizational Process Asset Structure

The subject of organizational process asset structure has received a great deal of attention. I have heard the following myth expressed by Agile proponents:

MYTH The CMMI requires a process “superstructure.”

“Superstructure” means multiple types and tiers of process assets. This myth continues to exist not because of anything the CMMI requires, but because of the way in which many large organizations have chosen to implement their process assets in the past.

As an example, it is not uncommon in many large high-tech companies to see four levels (or tiers) of process assets such as policies, processes/practices,

work instructions/procedures, and enablers/templates. Policies identify the organization's expectations for establishing and maintaining the process. Processes or practices are often high-level process descriptions whereas work instructions/procedures provide more detailed steps related to the process. Enablers and templates can be any kind of process aid that helps carry out the process and can include tool guides, or templates to help build related documentation.

While the choice for a process asset structure is up to each organization, most Agile organizations I have helped have found that two tiers is sufficient. This is accomplished by consolidating a policy statement with the associated process description that encapsulates “what must be done” in carrying out the process. The second tier contains “how to” guidelines in carrying out the process and tailoring it. This level can be viewed as aids for tailoring the process, and usually includes supporting templates. I have found that in most Agile organizations, step-by-step procedures are replaced by tool guides and training/mentoring. It is worth noting here that a template, such as a Project Management Plan template, can serve as a process with the required process activities implied within the template.¹⁰ This is a common technique I have observed for developing effective Agile process descriptions. See Figure 4-1 for a comparison of a traditional¹¹ and Agile organizational process asset structure.

Key Recommendation for Agile Organizations in Support of Tailoring

While decisions on process asset structure are up to each organization, there is one key related recommendation I make to Agile organizations. This recommendation was used successfully at BOND. I will state it in the form of a lesson:

LESSON 3

Keep your process “must dos” packaged separate from your process “guidelines.”

The reason for this recommendation relates to a major concern that management and independent appraisers often hold—the fear that an Agile approach will lead to loss of project control. This ties to a popular myth:

10. Refer to the appendices for an example of a Project Management Plan (PMP) Template.

11. By “traditional,” I mean what I have commonly observed in many large high-tech organizations.

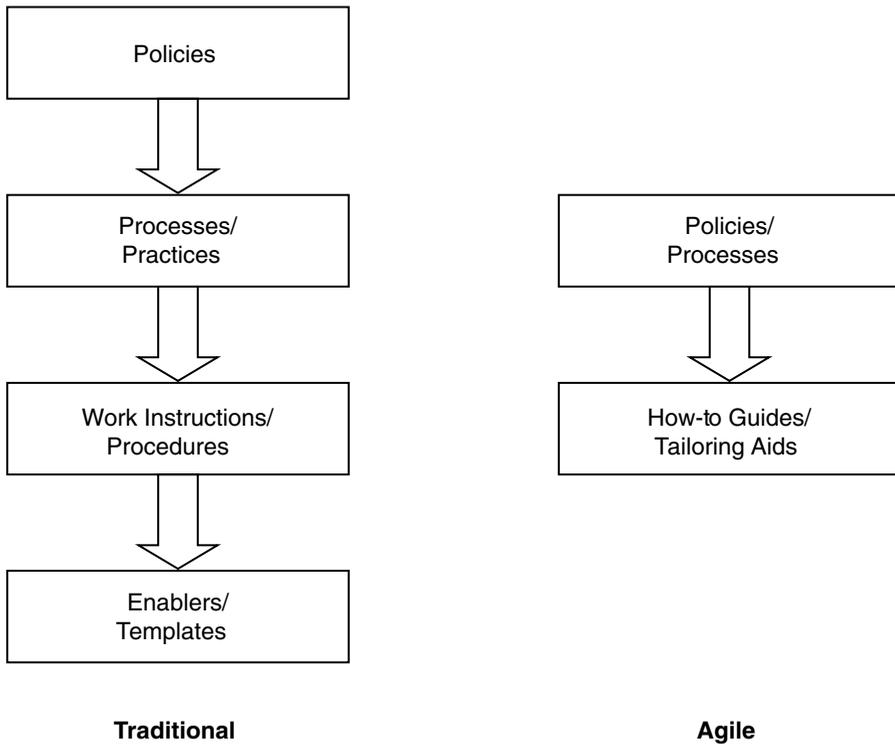


Figure 4-1 *Traditional (Sometimes) and Agile Organizational Process Asset Structure*

MYTH Agile organizations are less disciplined than traditional organizations and do not really follow any processes.

It has been my experience that organizations that understand and implement Agile practices appropriately tend to be more disciplined¹² in their development and management practices than many traditional development organizations. This is because they believe in their practices and therefore gravitate to them in times of crisis rather than abandoning them, as many more traditional organizations who don't really embrace their practices tend to do. The evidence of this often surfaces with the fervor that can

12. Refer to <http://www.ddj.com/architect/201804241> for a supporting article by Scott Ambler titled "The Discipline of Agile."

be sensed during the interview process when conducting a gap analysis or a more formal appraisal inside an Agile organization. In organizations in which compliance is achieved more through a “policing” approach, I have often found this same fervor and belief in the process missing.

Regardless of this observation, Agile organizations must still deal with the common perception that they don’t follow sound practices, and to be honest, many organizations that claim to be Agile are in fact using the term as a smoke screen to not comply and thus add to this perception.¹³

Following the recommendation in Lesson 3 prepares the organization to deal objectively with this perception by simplifying the tailoring process and making the “must dos” clear and visible to all. A fundamental implication of Lesson 3 is that no one tailors the “must do” practices. Everyone follows them. Hopefully, the reader is starting to appreciate the importance of establishing such rules early before the TWGs develop the processes. If you follow this recommended lesson, the TWGs must carefully consider what they agree to place in the process “must do” packages because this must make sense for all projects regardless of size or scale. Refer to Figure 4-2.

When you take this approach, which works well for organizations with Agile cultures, tailoring the process is integrated with project planning. Tailoring guidelines are used during project planning to make “how to” project specific decisions, such as decisions related to the use of certain tools. Since these guidelines are packaged separately from the process “must dos,” the process becomes very clear on what you are allowed to tailor and what must never be tailored.

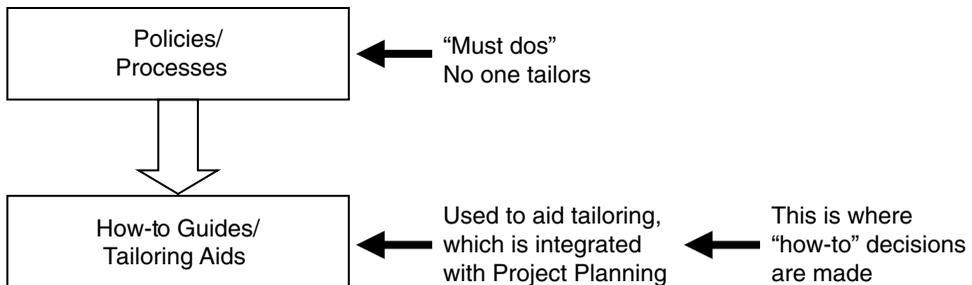


Figure 4-2 Tailoring and Process Asset Structure

13. Refer to the NANO and GEAR case studies later in the book.

By following this guidance, the visibility of compliance to the process becomes more evident in an Agile organization, not less. Fuzzy tailoring guidelines are now removed. It is for this reason I often make the claim that if you follow my guidance in the tailoring area when moving an organization with an Agile culture forward toward increased CMMI process maturity, you will find you have an increase in control rather than the loss of control that many falsely believe occurs in Agile organizations.

INSIGHT Many managers fear Agile will mean a loss of project control, but if you package your process assets and set up your tailoring guidelines in accordance with the guidance in this chapter, you will increase control, not lose control.

4.15 Process Asset Guidelines Used at BOND

Following are key guidelines we provided to the TWGs at BOND.¹⁴

- Process “must dos” are packaged separately from guidelines (hard rule).¹⁵
- No process is more than two pages (goal, soft rule).
- Processes do not contain “how-to” information or tool information unless you have decided to mandate this across all projects regardless of size or scale.
- Separate guidelines contain tailoring/planning options, and “how-to” information.
- Processes don’t stand alone; they require mentoring and training.

4.16 Different Organizations with Different Process Asset Structures

LACM and BOND are different types of organizations in many ways. LACM is large and product-centric; BOND is small and service-centric. LACM has

14. Refer to the appendices for example organizational process asset guidelines.

15. Refer to the example in the appendix of the Organizational Process Definition (OPD) Process description.

decided to mandate a number of tools and standards across their organization to support more effective product-centric development and reuse. BOND has decided it makes sense to mandate few standards and few tools because their business is software service oriented, and they need to be flexible in supporting whatever tools and standards are required based on the constraints of each project.

The resulting organizational repository structures in these two organizations are very different in size and structure based on their different business strategies, but both are “CMMI compliant”¹⁶ because they have been developed based on each organization’s business needs.

While their organizational repository structures are different, both organizations have achieved formal advanced CMMI levels using these different structures. Contrary to popular myths, the CMMI does not mandate an organizational repository “superstructure” as I have often heard Agile proponents claim.

The CMMI does require each organization to document its processes and maintain those processes at the organizational level where they can be shared and tailored to meet the needs of each project. How you execute your tailoring is up to each organization based on its business needs. The choice is yours as to the size, structure, and agility of your process assets. Nothing in the CMMI OPD expected practices is counter to an Agile approach.

4.17 Agile TWG Roles and Responsibilities

TWGs are composed of assigned personnel who take on two distinct roles: TWG lead and TWG members. The TWG lead is the “doer,” which means the lead is responsible for documenting the draft process assets according to the agreed-to process asset structure. This means the lead must clearly understand the process asset structure and guidelines. By minimizing the number of people who actually “write” the processes, we reduce the risk of extensive review cycles due to inconsistent process assets that don’t follow the agreed-to rules.

The TWG members are SMEs. Members are often some of the best people in the company and their time is valuable. This approach supports the most effective use of the members’ time by not requiring that they become experts

16. In this book, “CMMI compliant” means meeting the intent of the CMMI practices.

in the organizational process asset structure and the techniques of writing good processes.

4.18 Effective Techniques to Run an Agile TWG

One of the most effective ways to run an Agile TWG is a variant on how I conduct a gap analysis interview. You can think of an Agile TWG as the next step in “extracting” the real “as-is” process from the organization that started during the gap analysis.

To help extract the “as-is” process from TWG members I like to stand at a whiteboard and ask the TWG members to throw out words that are either activities they do as part of this process or products they produce. I tell them not to worry about creating full sentences. When you ask people to describe the process they follow, often they get wrapped up in talking about all kinds of extraneous detail. I find that it is best to let them talk this way during a gap analysis interview because it puts them at ease, allowing them to communicate more effectively. I have also found that TWGs can easily become bogged down with a great deal of nonessential discussion. This simplified guideline I have found helps to keep the working group focused on the task at hand. This is an area where the TWG lead needs to sense the group dynamics. For a small working group that has trouble getting started, it might work best to just let them talk about how they do their jobs for a period of time. However, if the leader senses the group is getting too far off task he or she might move to my simplified recommendation.

4.19 Separating the TWG Work from the Lead Offline Work

The techniques of running an Agile TWG described in the last section are intended to help keep the group at the desired level of discussion. If the discussion stays too high, the lead should ask more direct questions such as:

What aids do you use to get your job done such as guides, tools, templates?

The working group members usually do not need to discuss the packaging of the process assets into “must dos” and “guides.” This is often more efficiently handled by the TWG lead after the group adjourns. It is important for the lead to take all notes such as drawings or words that were jotted down on

a whiteboard. It is also important to capture the terminology used by the group members.

I have seen TWG leads who decided on their own to “translate” the terminology the group members were using in a working session into “CMMI terminology,” thinking this was part of their responsibility. This is definitely a mistake and should be guarded against.

LESSON 4

Keep your processes in your organization's terminology. Don't try to translate into “CMMI terminology.”

The reason for this lesson is really a variant of Lesson 2 in Chapter 2. The CMMI is not a set of dictated practices, and is not intended to dictate terminology. When we say it is a tool to help you reason about your processes, this means to reason about your terminology as well. It is therefore fine to discuss and raise potential issues about the right terminology in your organization. If a term is being used by some inappropriately, this should be discussed. Keep in mind our primary purpose is to “extract” the real process that is used first, and this includes extracting the real terminology used.

In the case when I observed a TWG lead “translating” the terms the group used, it caused a significant buy-in problem during the deployment stage of the project. This occurred because the TWG members felt the lead hadn't listened to them, and members said they didn't even recognize the process that resulted from the TWG effort as being the process they actually used and discussed in the working group. Don't let this happen to your process improvement efforts. TWG leader's responsibilities are primarily facilitation, listening, and documenting.¹⁷

4.20 What Do You Do When You Find a Gap?

A second gap analysis against the CMMI model is conducted offline by the TWG lead after the initial sorting out of the notes from the TWG session and creation of the initial draft Process and Guidelines documents.

When a gap is found, it usually becomes a topic for a follow-on TWG session where the group is also reviewing and commenting on the draft process and

17. The subject of terminology is also discussed in the NANO case study in Chapter 6.

guideline artifacts. This is where the facilitator should be in the “discovery” and “digging” mode as discussed earlier. Questions to be asking during this session include:

Is there a problem in the organization because this practice is not happening?

Usually, through this digging process if there isn’t a problem in the organization, the group should be able to uncover what is being done to accomplish the intent of this practice. Once this is discovered it should be added to the process documentation so it can be shared with others in the organization during training as discussed earlier.

If the answer is “yes,” the next question should be:

Do we all agree the organization should be “stretching” at this time to change its behavior to accomplish this practice?

If the group agrees the answer is “yes,” they might decide to add the must-do to the process. However, each decision should be carefully considered because we are now creating some of the most difficult potential process improvement work—that is, behavior change in the organization. This will require documentation, and training with rationale as to why this new practice is needed to help the organization achieve its business goals.¹⁸

INSIGHT The most difficult and costly process improvements are those that involve behavior change. Ensure all initiated changes are essential to achieving business objectives.

Section II

Answers to Common Questions

4.21 Answers to Common Questions When Running an Agile TWG

Following are answers to common questions that often arise when running an Agile Technical Working Group.

18. In the next chapter, we talk more about the most significant gaps found at BOND and what we did to address them.

4.22 Do I Need a DAR Process?

At BOND, it was decided that a distinct Decision Analysis and Resolution (DAR) process and guidelines were not required by the organization. Following is the logic that was used to arrive at this decision, which caused no difficulty during the formal CMMI evaluation.

In the DAR TWG at BOND, the group first found itself asking the question:

What are the relevant formal decisions that arise in our organization, and how do we handle them today?

This discussion led to the recognition that formal decisions at BOND were made in two areas: Risks and Designs. The group also discussed what “formal” meant in their organization. The CMMI doesn’t tell you what “formal” means, so each organization can make this decision for itself based on its own business needs. Formality at BOND (which did most things informally) was taken to mean the need to involve someone in the decision at a higher level of management. From a risk perspective, formal decisions involved the need to raise a risk to higher-level management. From a design perspective, formal decisions involved evaluating alternative design decisions that affected other groups.

In the case of a risk, the criteria to consider when deciding to raise the risk to Senior Management were included in the Risk Management guidelines that were developed as part of the Risk Management TWG. In the case of design alternatives, the criteria to use in making decisions were included in the design guidelines that were developed as part of the Technical Solution TWG. Therefore, DAR was handled through existing processes and no additional process assets were required.

4.23 Do I Need to Verify Everything I Develop?

The CMMI model does not dictate the work products that must be verified. SP 1.1 of the Verification Process Area expects each project to select the work products to be verified. Once again, as in so many areas of the CMMI model, this decision is up to you.

Often this practice is overlooked especially in organizations that have been building products for many years. A number of my client organizations are product-centric. Specific Practice 1.1 of Verification is a very good example of

how the CMMI can help us reason about our processes. It helps us ask questions that can in turn help us manage our work more effectively. Often the questions that result from using the CMMI are ones we might not think to ask otherwise.

As an example, organizations that rely heavily on product reuse should also be relying heavily on reusing the end product such as the software code, and reusing requirements, test cases, and test results. In other words, if I am reusing a product that has already been verified, I should be able to reuse that verification to gain the benefits of that effort. I will still need certain levels of verification and validation in the new environment where I am reusing the product, but the potential exists to “skip” certain lower levels of verification. To help us reason about where in our processes it makes sense to allow one to “skip” certain verification steps, SP 1.3 reminds us that we should have verification criteria. This leads to the question:

What are the criteria we use to determine when a verification level can be skipped?

It should be apparent that the creation of criteria can be a powerful aid to help an organization and its processes become more agile in making dynamic work-related decisions. However, criteria can only help if they have been created and personnel are trained in their use.¹⁹

4.24 Do I Need to Make Sure the Steps in My Processes Are in the Right Order?

MYTH CMMI-compliant processes require a sequence of steps.

I have observed numerous Technical Working Groups wasting valuable time arguing about the steps in a process and the order in which those steps occur. First, the CMMI defines processes as “activities that can be recognized as implementations of practices in a CMMI model.” It doesn’t say the order in which those activities occur must be specified.

It has been my experience that when first developing process documentation, any order dependencies should be one of the last items we worry about. I have found that TWGs can spend incredible amounts of time discussing sequence

19. The power of criteria in helping an organization make more rapid real-time decisions is discussed at greater length in the GEAR case study in Chapter 7.

topics that turn out to be noncritical. I am not saying order is unimportant, just that often areas where we think we have order dependencies turn out to be “soft” order dependencies at best. Any “hard” order dependencies can always be added later.

A good example of order dependencies I refer to as “soft” is project planning. When I teach planning, I talk about the “what,” “who,” “when,” “how,” and “how much.” There are certainly order dependencies here. I can’t fully define the “who” (resources I need on the project) until I know the skills I need, which depends on the “what” I need to produce. I can’t complete the “how much” it will cost until I have figured out all the other pieces to my plan because they all imply some level of cost. Nevertheless, I can provide a project plan template to be used as a great aid to help people plan without telling them which sections they must fill in before others. Such dependencies are best communicated through training, rather than captured through formal documented process descriptions.

4.25 Do I Need to Make Sure Process Descriptions Are Not Redundant?

Often I hear people in TWGs arguing over whether a certain activity should be included in some process document. For example, in the technical solution TWG there was considerable discussion related to whether the design process should refer to requirements development at the start of the process. I tell working groups that it is okay to include words about an activity that might be in another process if it adds to the understanding of this process. Many processes are closely connected, such as requirements and design. Because of the way most Agile teams work—iterating closely between requirement, design, implementation, and test—it makes sense to describe this process as it is executed in your organization. This is another reason why it is best not to get too hung up on order. The traditional order of requirements followed by design followed by implementation followed by test isn’t the way Agile teams work. While at a high level this view still might make sense, the activities Agile teams follow during a given day might appear to jumble this order.

The bottom line is that we want to capture the activities and products produced that relate to our processes. If it helps to describe closely related activities that are also included in another process document, it doesn’t hurt to say it again.

4.26 Can Requirements Be Captured in an Email or PowerPoint Slides?

This might sound like a strange question, but it is not uncommon to hear it in Agile organizations that are just starting out with a CMMI process effort. First, the CMMI does not dictate the format requirements must be captured in, so on the surface, nothing directly prohibits email or Microsoft PowerPoint slides from being used to document requirements. However, when you look more closely at related expected practices and start asking a few more questions the CMMI expected practices will raise, a different picture often results.

For example, Requirements Management PA, SP 1.3 states:

Manage changes to the requirements as they evolve...

and SP 1.4 states:

Maintain ... traceability among the requirements and work products.

These expected practices lead to the following questions:

How do you manage changes to requirements as they evolve if your requirements are captured only in email or PowerPoint slides?

Are you going to update the PowerPoint presentation or email whenever changes are agreed to so the current set of accepted requirements is clear?

One of the reasons traceability is an expected practice is to ensure our testing addresses all requirements including any changes. For this reason I have always suggested to clients that, while you might not need a formal requirements management tool, you do need to have your requirements organized and managed in a way that supports the assignment of requirements identifiers to each requirement so that those identifiers can be used in a test document to ensure your testing is complete.

As you start to ask these questions that arise from using the CMMI to reason about your processes, most organizations, including those with an Agile culture, decide that email and presentation tools cannot adequately do this job. Some very small organizations, and organizations with products that have very stable requirements, might be able to survive with requirements communicated through these means, but most organizations quickly recognize the limitations of these mechanisms.

4.27 Do Requirements Need to Be Captured in Single “Shall Statements”?

This question often arises in Agile organizations that do requirements using user stories or use cases. First, there is no expected practice in the CMMI with respect to “shall statements.” The same questions concerning the management of requirements through the life cycle, and traceability, need to be asked. In many Agile organizations, user stories or use cases are often found to help the developers initially understand the requirements and to develop the test cases. Once these test cases are established, the cases themselves often become the agreed-to requirements with the customers. If your customer agrees to this approach, this may suffice to achieve the intent of the requirements management specific practices related to requirements change management and traceability. This is an example where an organization needs to ask a number of “what if” questions related to future potential changes and possible consequences before making such decisions. Other good questions to ask at this time related to the way your organization currently operates include:

Is there a problem in the organization with respect to Requirements Management?

Do customers ever come back and challenge an earlier decision with respect to a requirement change?

Nothing in the CMMI says that a managed test document cannot meet the intent of managing requirements. Asking these types of questions that naturally result when using the CMMI model will often lead to very good discussions in your TWGs that help an organization understand its own processes better and where some process modifications could be of benefit.

4.28 Formalizing Informality

LESSON 5

You can “formalize” informality.

One of the greatest achievements with BOND was our close attention to their culture and maintaining it as they grew. As we added the necessary process formality to prepare them for both the organization’s continued growth and

their upcoming CMMI assessment, we monitored any changes closely to ensure we weren't damaging the Agile culture that had gotten them their rapid growth and success so far.

Key to our success at BOND was a strategy I have referred to as "formalizing informality."

If something is working well, you don't have to change it for CMMI. However, you do have to document it so it can be taught and shared with others.

It might sound odd to say this, but you can formalize informality, and we did it at BOND successfully. What I mean is if you have a process that works such as a risk management process, but it is "informal" in certain ways, you can teach what you do just like you do it, and document it just like you do it. I have found there almost always seems to be a strong tendency by process professionals to assume when working a process improvement effort, what people currently are doing must be wrong if they have no formal documented processes. This view rests at the heart of why we often find in large supposedly process mature organizations a large disconnect from what the people actually do, and what their processes say they do.

An Example of Formalizing Informality: "Doorway" Risk Management

Let me give you an example of formalizing informality. At BOND, one of the reasons the company was so successful was because risk management was an ingrained way of working. People lived risk management daily. When they had a risk they were often in the doorway of a Senior Manager's office strategizing the risk mitigation. They were doing it immediately, not waiting until a formal risk management meeting. Because of this informality, they were able to initiate risk mitigation almost instantly, thereby keeping potential risks from becoming real problems. Effective risk mitigation stood at the heart of why this organization was successful.

Rather than try to add unnecessary paperwork to this process that was already working effectively, we just described in the newly documented Risk Management Process exactly what the expectations were of how risks were identified, assessed, and categorized in the organization. We did add a small degree of documentation that wasn't going on before by adding a risk slide to the periodic senior management briefs, but we emphasized in the Risk Management training the existing culture that was expected to continue to effectively manage risks. We actually taught this informal "doorway risk management" approach.

4.29 Summary

We have shared many examples in this chapter to help illustrate what is mandated with the CMMI and what is not. “How-to” approaches are not mandated. You do need “how-to” approaches and the CMMI expects that you have them—but it doesn’t mandate what they need to look like. They can look traditional or Agile. CMMI doesn’t give you the answers, but it does tell you what questions you need to ask and answer for your organization and your project teams.

The focus of this chapter has been on extracting the real “as-is” process and packaging the results. However, even in very successful Agile organizations, there are practices within the CMMI where the intent is not being achieved. In these cases, often adding activities might be needed. Understanding the rationale for these added practices and how they were handled at BOND is the focus of the next chapter.

4.30 Summary: How Agile Helps CMMI

The following table provides a summary of how Agile approaches discussed in this chapter help the CMMI.

Table 4-3 *How Agile Helps CMMI*

“How-to” Approach in Agile Environment	How It Helps CMMI
“Doorway” Risk Management	Helps us implement Risk Management effectively achieving its real intent and timely risk mitigation
Customer demos early, continuous informal reviews	Helps us implement the intent of the Verification PA by identifying defects early and opportunities for improvement
Tailored Agile TWG/Gap Analysis approach	Helps us develop processes that reflect practical and proven techniques that work, and what people really do

Table 4-3 *How Agile Helps CMMI (Continued)*

“How-To” Approach in Agile Environment	How It Helps CMMI
Agile process packaging separating “must dos” from “guidelines” Agile tailoring process integrated with project planning	Supports agility and control
Agile “digging” approach when finding a gap	Helps us locate the most valuable process improvement candidates

Index

A

- action items, 93, 116–118
- Agile approach
 - aids to project management practices, 112–113
 - apparent conflicts with CMMI model, 63
 - applying to golf project, 250–251
 - customer collaboration in DART case study, 208–209, 211, 212–213
 - defined, 12, 307
 - focus on projects, 120–121
 - list of key practices, 11–12
 - partitioning work into increments, 44
 - relationship to CMMI, 266–268
 - role of Senior Management, 227–229, 230, 231
 - using Scrum, 209–210, 211
 - using together with (EVMS) Earned Value Management System, 219–221
- Agile Developer’s Guide, in RAVE case study, 27–28
- Agile Manifesto
 - defined, 10
 - examples, 165, 197, 277
 - identified values, 11
 - principles behind, 10, 285–286
- Agile methods, defined, 10. *See also* Scrum, as Agile method
- Agile Modeling, as Agile method, 10
- Agile organizations
 - defined, 12, 307
 - keys to gap analysis in, 60–62
 - maintaining project control, 74–77
 - “must do” practices compared with “how to” guidelines, 74–77
 - process packaging compared with CMMI
 - process areas, 71–73
 - repository structure, 71, 125–126
 - TWG approach, 64, 65, 78–79
- Agile principles. *See* Agile Manifesto
- Agile-like
 - defined, 12, 307
 - organizations as, 138, 139, 143, 177, 178
- agility, stealth, 27, 154, 308
- Aikido, 242–243, 250
- aligning
 - defined, 24, 307
 - GEAR case study example, 176–177
 - LACM case study example, 21–23
- appraisals
 - in BOND case study, 58, 70, 114, 120, 125, 129–130, 131
 - compared with gap analysis, 60, 63
 - compared with TWG approach, 65
 - in LACM case study, 48
 - in NANO case study, 148
 - options for using CMMI model, 26–27
- “as-is” process view
 - applying to golf project, 241–242
 - in BOND case study, 111, 114
 - as initial step, 35
 - in LACM case study, 26
 - in NANO case study, 142
 - role of TWGs, 79

B

- backlogs, Scrum task management items, 209–210, 211, 212, 214, 218–219, 233
- Bamberger, Judy, 216, 222

Boehm, Barry, 206

BOND case study

“as-is” process view, 111, 114

background, 58–59, 92

CMMI areas requiring greater attention, 92

CMMI level 3 goal, 66

compared with GEAR case study, 172

compared with LACM case study, 77–78

corporate decline, 131–132

formalizing informality, 86–87

implementing quality program, 196, 197–198

improving stakeholder involvement, 120

master schedule, 105–106

measurement program, 123–126

“must do” practices, 74–77, 124, 127–128, 172

peer review process, 62, 69–70

process asset structure and guidelines, 77–78

process writing working group, 182–183

project management, 95–111, 196

role of TWGs, 66–67, 69, 70, 72

running process improvement like project, 64

senior management briefings, 108–110

super-spreadsheet, 104

tailored TWG techniques, 70–71, 72, 76, 77, 182

training, 126–127

TWG approach, 64, 65

weighing need for DAR process, 82

weighing need for VER process, 82–83

brainstorming, 145. *See also* process writing working groups

briefings, senior management, 108–110

C

Capability Maturity Model Integration. *See* CMMI (Capability Maturity Model Integration)

CAR (Causal Analysis Resolution) Process Area

defined, 7

and golf project, 252

SP (Specific Practice) 1.1, 8

SP (Specific Practice) 2.1, 8, 39

uses for, 42–43

Causal Analysis Resolution. *See* CAR (Causal Analysis Resolution) Process Area

causes, assignable (people) *vs.* common (process), 40, 41

checkpoints, effective, 250–251, 252, 253–256, 259, 260–261. *See also* quality control checks

Cink, Stewart, 255–256

CM. *See* configuration management

CMM, defined, 17

CMMI (Capability Maturity Model Integration)

addressing potential weaknesses against, 61, 62–63, 64, 65

apparent conflicts with Agile approach, 63

applicability in DART case study, 221–227, 234, 235

applying to golf project, 241, 250, 251, 252, 253, 255, 258–259

background, 5–10

Continuous Representation of model, 9–10, 39

defined, 5

employing hybrid Scrum/traditional approach, 206

focus on both projects and organizations, 120–122

formats for documenting requirements, 85–86

higher-level practices, 31–55

“imposition” method of using model, 5

mitigating appraisal risk, 129–130

“nonimposition” method of using model, 5

order of process activities, 83–84

as process roadmap in NANO case study, 148–151

as reference model, 165–166

relationship of EVMS to, 225–226

relationship to Agile, 266–268

role in gap analysis, 60–62

role of TWGs, 64–65

Staged Representation of model, 9

where to start, 18–20

CMMI compliance

defined, 10, 17, 307

GEAR case study issues, 192–198

CMMI for Development software, 37

Cockburn, Alistair, 40, 231, 242, 268

collaboration management

in BOND case study, 110–111

- in DART case study, 208–209, 211, 212–213
- role of “push-pull” technique, 219–221
- role of Scope documents, 214–215
- compliance. *See* CMMI compliance
- configuration management, 10, 73, 103, 116, 291
- continuous process improvement, in LACM case study, 45–53
- Continuous Representation, CMMI model, 9–10, 39
- criteria
 - as agility-enhancing mechanism, 191, 198–200
 - as aid in decision making, 160–161, 200–201
 - in GEAR case study, 191, 198–200
 - relationship to tailoring up, 191
 - role in project planning, 191
- Crystal, as Agile method, 10
- Crystal Clear* (Cockburn), 231
- customer collaboration
 - in BOND case study, 110–111
 - in DART case study, 208–209, 211, 212–213
 - role of “push-pull” technique, 219–221
 - role of Scope documents, 214–215

D

- DAR (Decision Analysis & Resolution)
 - Process Area
 - compared with Agile organizational process, 73
 - defined, 7
 - SP (Specific Practice) 1.2, 143, 159
 - weighing need for, 82
- DART case study
 - background, 207–209
 - role of Scope document, 215, 218–219, 220, 224–225
 - role of Scrum, 207, 209, 210, 211–212, 213
 - what went wrong, 208, 210–211, 229
- Decision Analysis & Resolution. *See* DAR (Decision Analysis & Resolution)
 - Process Area
- decision making
 - criteria as aid, 160–161, 200–201
 - establishing time priorities, 215, 216–218
 - and golf project, 253
 - in LACM case study, 40

- Demarco, Tom, 229
- developer’s guide, as approach to agility, 27–28
- disconnects, defined, 34, 307
- distributed process ownership
 - in GEAR case study, 171–174
 - in NANO case study, 146, 149
- documentation
 - in BOND case study, 112–118, 182–183
 - formats for capturing requirements, 85–86
 - in GEAR case study, 184
 - matching written processes to real processes, 178–180
 - process writing working groups, 182–183, 184
 - processes compared with policies, 74, 152
 - Scope documents, 214–215, 218–219, 220, 224–225
- DOORS tool, 48, 49–51
- doorway risk management, 151

E

- Earned Value Management System (EVMS)
 - relationship to CMMI, 225–226
 - using Agile approaches with, 219–221
- enablers, 74, 293, 294, 295. *See also* templates
- EVMS (Earned Value Management System)
 - relationship to CMMI, 225–226
 - using Agile approaches with, 219–221
- Extreme Programming, as Agile Method, 10

F

- formalizing informality, 86–87, 152–154
- foundation measures, 33–34, 36

G

- gap analysis
 - in Agile organizations, 60–62
 - applying to golf project, 241–242
 - conducting interviews, 60–62, 79, 80–81
 - in GEAR case study, 170, 171, 174, 175, 177–178, 180, 187, 192
 - in NANO case study, 139, 141–142, 143, 156
 - ways to conduct, 60–62

- GEAR case study
 achieving intent, 180
 aligning engineering and project management, 176–177
 background, 170–171
 common patterns, 171–173
 compared with BOND case study, 172
 compared with NANO case study, 171, 172
 criteria as agility-enhancing mechanism, 191, 198–200
 distributed process ownership, 171–174
 and lessons from golf project, 244, 249, 253
 lessons learned, 268–269
 matching written processes to real processes, 178–180
 “must do” practices, 172–173, 175, 176
 process compliance issues, 192–198
 process optimizations, 181–187
 process writing working group, 184
 quality checks, 175, 192, 197–198
 R&D environment, 169, 170, 175, 192
 repeating specific weaknesses, 244, 249
 role of CMMI Process Areas, 184–186
 role of tailoring, 188–191
 stakeholder issues, 173–174, 225
 “thread” approach to process development, 184–187
 unclear process asset requirements, 171–172
 value of meetings, 177–178
- Generic Practices (GPs), CMMI. *See also specific GPs by number*
 description, 6, 7–8
 list, 8
 NANO case study view, 142–143
 purpose, 9
 redundancy with Process Areas, 10, 194
- golf project
 applying CMMI and Agile approach, 241, 250–251, 252, 253, 255, 258–259
 checkpoints, 250–251, 252, 253–256, 259, 260–261
 conducting gap analysis, 241–242
 lessons learned, 270–271
 measurable objectives, 249–250, 251
 passion as factor, 279–283
 relationship to business, 249–250, 259–260
 repeating specific weaknesses, 243–244, 245, 246–249
 and tailoring, 277
 visualization and integrated practice, 257, 259
- GP (Generic Practice) 2.1
 defined, 8
 in NANO case study, 149
- GP (Generic Practice) 2.2, 8
- GP (Generic Practice) 2.3
 in BOND case study, 96–98
 in DART case study, 222
 defined, 8
 in project management planning, 102
 template example, 289
- GP (Generic Practice) 2.4
 in BOND case study, 96–98
 in DART case study, 222
 defined, 8
 in project management planning, 102
 template example, 289
- GP (Generic Practice) 2.5
 in BOND case study, 127
 in DART case study, 224, 226–227
 defined, 8
 and personal safety, 231–232
 in project management planning, 102
 redundancy with OT Process Area, 10, 127
 template example, 289
- GP (Generic Practice) 2.6, 8, 10, 291
- GP (Generic Practice) 2.7
 in DART case study, 224–225
 defined, 8
 defining relevant stakeholders, 118–119, 215
 in NANO case study, 142–143, 157
- GP (Generic Practice) 2.8
 compared with Product and Process Quality Assurance (PPQA) Process Area, 194
 defined, 8
 in GEAR case study, 194
 in NANO case study, 194
 options for achieving, 194–195
- GP (Generic Practice) 2.9
 defined, 8

redundancy with PPQA Process Area, 10
 template example, 291
 GP (Generic Practice) 2.10
 in BOND case study, 109, 110
 defined, 8
 GP (Generic Practice) 3.1, 8, 272–273
 GP (Generic Practice) 3.2, 8, 121

H

Ha, stage of Aikido, 242, 250
 high maturity, defined, 13, 21, 307. *See also*
 CMMI (Capability Maturity Model
 Integration), higher-level practices
 “how” in project management plans, 84, 101,
 108–110, 115, 116, 119, 126–127
 “how much” in project management plans,
 84, 101, 110–111, 115
 Humphrey, Watts, 22, 33, 37
 hybrid Agile, defined, 12, 206, 307

I

informality, formalizing, 86–87, 152–154
 Integrated Project Management. *See* IPM
 (Integrated Project Management)
 Process Area
 intent, achieving, 66–67, 68, 69–70,
 110–111, 180
 interviews, role in gap analysis, 60–62, 79,
 80–81
 IPM (Integrated Project Management)
 Process Area
 compared with Agile organizational
 process, 73
 defined, 7
 and personal safety, 231
 iterations, work, 44

J

just-in-time training, 156, 224

K

Kern, Paul, 240, 242, 243

L

LACM case study
 aligning process improvement effort with
 business objectives, 21–23
 background, 18
 compared with BOND case study, 77–78
 current approach to process improvement,
 45–53
 higher-level CMMI practices,
 42–43, 46–47, 48
 implementing measurement process, 53
 and lessons from golf project, 245, 253
 mix of process and people problems, 41
 “must do” practices, 53, 124
 peer review process, 24–25
 process asset structure, 77–78
 procurement issues, 35, 40, 41–42, 47, 51,
 258, 262, 269
 repeating specific weaknesses, 245
 starts CMMI-based process improvement
 effort, 18–20
 training, 226
 using CMMI model for informal appraisal,
 26–27
 leaders
 growing from inside, 98–99
 project, 98–99, 127, 161–162
 technical, 126–127
 TWG, compared with members, 78–80
 leaning (streamlining), 24–25
 “less visible” tasks, 216, 217–218
 level 3, CMMI Maturity. *See also* CMMI
 (Capability Maturity Model
 Integration), higher-level practices; high
 maturity, defined
 defined, 13, 20, 307
 life cycle, project, 106–107
 Lister, Tim, 229
 “local” practices, 68

M

MA (Measurement & Analysis) Process Area.
See also measurement
 compared with Agile organizational
 process, 73

MA (Measurement & Analysis) Process Area
(Continued)
 defined, 7
 lessons learned, 269
 SP (Specific Practice) 1.1, 8, 52, 123
 SP (Specific Practice) 1.2, 8
 SP (Specific Practice) 2.4, 8
 template example, 292

Management Steering Group (MSG)
 in BOND case study, 110
 in GEAR case study, 171, 172, 173
 in sample OPF template, 300, 301

Manifesto. *See* Agile Manifesto

master schedule, in BOND case study, 105–106

measurement. *See also* MA (Measurement & Analysis) Process Area
 in BOND case study, 123–126
 foundation measures, 33–34, 36
 in golf project, 249–250, 251
 in LACM case study, 34, 53
 lessons learned, 269–270
 need for specific context-relevant measures, 36–37
 problem of insufficient metrics, 36
 process measures, 33
 product measures, 33, 50, 51
 purpose, 33, 37, 123
 repository for, 71, 125–126
 resource measures, 33
 traditional compared with Agile, 153–154
 types of measures, 33–34

Measurement & Analysis. *See* MA (Measurement & Analysis) Process Area

measurement flow down, 153, 154

meetings
 minutes from, 116–118
 in NANO case study, 158–162
 off-site, 158–162
 standup, 105, 108, 117
 value in GEAR case study, 177–178

mentoring. *See also* training
 in BOND case study, 127, 196, 197
 in DART case study, 216
 as less visible task, 218

minimums, 28, 114, 127, 147, 172, 174, 175, 189, 190, 192, 197, 253

momentum, losing, 130–131

MSG (Management Steering Group)
 in BOND case study, 110
 in GEAR case study, 171, 172, 173
 in sample OPF template, 300, 301

“must do” practices
 in BOND case study, 74–77, 124, 127–128, 172
 compared with “how to” guidelines, 53, 74–77, 172–173, 175, 176
 in GEAR case study, 172–173, 175, 176
 in LACM case study, 53, 124
 minimum, 114, 127, 260
 in OPD documentation process, 305
 in process writing groups, 184
 tailoring recommendations, 189, 190

myths, defined, 307

N

NANO case study
 background, 139–141
 CMMI as process roadmap, 148–151
 compared with GEAR case study, 171, 172
 conducting installs, 144–145
 gap analysis, 139, 141–142, 143, 156
 and GP 2.8, 194
 implementation of CMMI, 137–138
 lessons learned, 268–269
 off-site meeting, 158–162
 process improvement plan, 154–156
 repeating specific weaknesses, 244, 249
 review of written processes, 143–144
 role of Director, 139–140, 141, 142, 145, 146, 155, 157, 158, 159, 161–162, 163
 and tailoring, 275–276, 277
 training, 150–151

O

off-site meetings, 158–162

OPD (Organizational Process Definition) Process Area
 compared with Agile organizational process, 73
 defined, 7
 establishing rules for process assets, 171–172

- in GEAR case study, 174
- in NANO case study, 147
- process asset guidelines, 293, 294
- sample asset approval and release process, 297–298
- sample template, 303–306
- SP (Specific Practice) 1.1, 8, 293
- SP (Specific Practice) 1.3, 9, 273, 294
- SP (Specific Practice) 1.5, 294, 297–298
- OPF (Organizational Process Focus) Process Area
 - compared with Agile organizational process, 73
 - defined, 7
 - in GEAR case study, 174
 - in NANO case study, 147, 162–163
 - sample template, 299–302
 - SP (Specific Practice) 1.1, 9, 22, 26, 29
 - SP (Specific Practice) 1.3, 162
- Organizational Process Definition. *See* OPF (Organizational Process Focus) Process Area
- Organizational Process Focus. *See* OPF (Organizational Process Focus) Process Area
- Organizational Training. *See* OT (Organizational Training) Process Area
- OT (Organizational Training) Process Area
 - in BOND case study, 127
 - compared with Agile organizational process, 73
 - compared with GP 2.5, 10
 - defined, 7
 - purpose, 150

P

- PAs. *See* Process Areas (PAs), CMMI
- passion, as factor, 279–283
- peer reviews
 - in BOND case study, 62, 69–70
 - compared with verification, 69–70, 199, 200–201
 - identifying intent, 69–70
 - in LACM case study, 24–25
- personal safety, 231–232
- Personal Software Process (PSP), 37

- PMC (Project Monitor and Control) Process Area
 - in BOND case study, 72
 - compared with Agile organizational process, 73
 - in DART case study, 225–226
 - defined, 6
 - relationship to project planning, 164
 - SP (Specific Practice) 1.1, 9, 225–226
 - SP (Specific Practice) 1.6, 290
 - SP (Specific Practice) 1.7, 290
 - SP (Specific Practice) 2.1, 116–117, 225–226, 290
 - SP (Specific Practice) 2.2, 117, 290
 - SP (Specific Practice) 2.3, 9, 117, 290
 - template example, 290
- PMPs (project management plans)
 - creating template for planning document
 - in BOND case study, 99–111
 - “how,” 84, 101, 108–110, 115, 116, 119, 126–127
 - “how much,” 84, 101, 110–111, 115
 - reconciling, 113
 - requiring documentation in BOND case study, 112–118
 - role of templates, 114, 115–116
 - sample template, 287–292
 - “what,” 38, 84, 101–102, 115, 126
 - “when,” 84, 101, 104–107, 115
 - whether to require, 96
 - “who,” 84, 101, 102–104, 115
- PP (Project Planning) Process Area
 - in BOND case study, 72
 - compared with Agile organizational process, 73
 - in DART case study, 222, 223
 - defined, 6
 - in NANO case study, 163–166
 - relationship of project monitoring and control to, 164
 - role of scheduling in project management, 104–106
 - SP (Specific Practice) 1.1, 289
 - SP (Specific Practice) 1.2, 9, 110, 222, 223, 272
 - SP (Specific Practice) 1.3, 9, 106–107, 176, 289

- PP (Project Planning) Process Area (*Continued*)
- SP (Specific Practice) 1.4, 166
 - SP (Specific Practice) 2.1, 104–105, 289, 291
 - SP (Specific Practice) 2.4, 289
 - SP (Specific Practice) 2.7, 9
 - SP (Specific Practice) 3.2, 9, 113
 - template example, 289, 291
- PPQA (Product & Process Quality Assurance) Process Area
- in BOND case study, 128–129, 196
 - compared with Agile organizational process, 73
 - compared with GP 2.6, 7
 - compared with GP 2.8, 194
 - compared with Verification (VER) Process area, 193
 - defined, 7
 - “how to” options for implementing, 195–196
 - and process compliance, 193
- practices. *See* Generic Practices (GPs), CMMI; Specific Practices (SPs), CMMI
- Process Areas (PAs), CMMI. *See also names of specific PAs*
- compared with Agile organizational processes, 71–73
 - description, 6–7
 - in GEAR case study, 184–186
 - institutionalization, 9
 - list of key PAs, 6–7
 - list of key SPs for each PA, 8–9
 - order of activities, 83–84
 - redundancy with GPs, 10, 194
 - representations of model, 9–10
 - role of GPs, 9
- process assets
- adding new activities, 93–95
 - Agile organizational structure, 73–77
 - BOND case study guidelines, 77
 - defined, 23, 308
 - enablers, 74
 - establishing rules for, 171–172
 - four-tier, 73–74
 - keeping terminology “real,” 80
 - LACM compared with BOND case studies, 77–78
 - “must do” practices compared with “how to” guidelines, 53, 172
 - in Organizational Process Asset Library, 147, 294, 297–298
 - policies, 73, 74
 - practices, 73, 74
 - procedures, 74
 - processes, 73, 74
 - sample agile organizational guidelines, 293–295
 - sample approval and release process, 297–298
 - superstructure, 73–74
 - templates, 74
 - traditional compared with Agile structures, 74, 75
 - two-tier, 74
 - unclear requirements in GEAR case study, 171–172
 - work instructions, 74
- process, defined, 74, 116
- Process Management Steering Group. *See* MSG (Management Steering Group)
- process tailoring. *See* tailoring
- process writing working groups
- in BOND case study, 182–183
 - in GEAR case study, 184
- procurement. *See* LACM case study
- Product & Process Quality Assurance. *See* PPQA (Product & Process Quality Assurance) Process Area
- Product Backlog task management item, Scrum, 209–210, 211, 212, 214, 219, 233
- product measures, 33, 50, 51
- Product Owner, Scrum, defined, 233
- Project Management Plan (PMP). *See* PMPs (project management plans)
- Project Monitor and Control. *See* PMC (Project Monitor and Control) Process Area
- Project Planning. *See* PP (Project Planning) Process Area
- pruning of processes, 23–24, 25
- “push-pull” task flexibility, 219–221
- ## Q
- QA (quality assurance). *See* PPQA (Product & Process Quality Assurance) Process Area

QPM (Quantitative Project Management) Process Area
 defined, 6
 empowering small teams to implement, 38
 and golf project, 252
 lessons learned, 269
 SP (Specific Practice) 1.3, 9
 SP (Specific Practice) 2.1, 39
 quality control checks, 175, 192, 194–195, 196, 197–198. *See also* PPQA (Product & Process Quality Assurance) Process Area
 Quantitative Project Management. *See* QPM (Quantitative Project Management) Process Area

R

RAVE case study, alternative approach to agility, 27–28
 RD (Requirements Development) Process Area
 compared with Agile organizational process, 73
 in DART case study, 224
 defined, 6
 SP (Specific Practice) 1.1, 215, 224
 reconciling project plans, 113
 representations, CMMI model, 9–10
 REQM (Requirements Management) Process Area
 compared with Agile organizational process, 73
 defined, 6
 in NANO case study, 156
 SP (Specific Practice) 1.3, 9, 230
 Requirements Development. *See* RD (Requirements Development) Process Area
 Requirements Management. *See* REQM (Requirements Management) Process Area
 resource measures, 33
 rhythm, in golf, 252, 255, 257–258, 260
 Ri, stage of Aikido, 242
 Risk Management. *See* RSKM (Risk Management) Process Area
 role-based scenario training, 156

RSKM (Risk Management) Process Area
 in BOND case study, 72
 compared with Agile organizational process, 73
 in DART case study, 223
 defined, 6
 SP (Specific Practice) 3.1, 9
 template example, 291

S

safety, personal, 231–232
 SCAMPI (Standard CMMI Appraisal Method for Process Improvement), 26, 27, 129
 scenario training, 156
 scheduling guidelines, 104–106
 Schwaber, Ken, 258
 scope. *See* “what” in project management plans
 Scope documents
 in DART case study, 215, 218–219, 220, 224–225
 flowing requirements downward, 225
 role in collaboration management, 214–215
 Scrum, as Agile method
 in DART case study, 207, 209, 210, 211–212, 213
 defined, 10, 206
 list of key practices/terminology, 233–234
 Product Backlog task management item, 209–210, 211, 212, 214, 219, 233
 Sprint Backlog task management item, 209–210, 211, 213, 218–219, 233
 10-percent rule, 213–214
 ScrumMaster, defined, 233
 “shall” statements, 86
 Shu, stage of Aikido, 242, 243, 250
 size measures, 124
 slices, work, 44
 small changes, 271, 272–274
 SMEs (subject matter experts)
 in BOND case study, 182
 in GEAR case study, 172
 in NANO case study, 181
 and “thread” approach, 185, 186–187
 and TWGs, 64–65, 68, 71, 78–79

Specific Practices (SPs), CMMI, overview, 6, 7, 8–9

Sprint Backlog task management item, Scrum, 209–210, 211, 213, 218–219, 233

Sprint meetings, Scrum, 233–234

Sprint, Scrum, defined, 233

sprints, work, 44

SSTC (Systems and Software Technology Conference), 32, 216

Staged Representation, CMMI model, 9

stakeholder matrix, 119, 149, 150, 152, 156, 174, 225

stakeholders

- involving, 118–120
- in NANO case study, 149–150

standup meetings, 105, 108, 117

statistical control, selecting subprocesses for, 44, 46, 47, 51

stealth Agile, 27, 154, 308

streamlining. *See* leaning (streamlining)

super-spreadsheets, 104

Sutherland, Jeff, 213, 223

T

tailoring

- in BOND case study, 74–77, 182
- common mistake, 28
- in GEAR case study, 174, 188–191
- key recommendations, 74–77
- in NANO case study, 162–163
- in RAVE case study, 27–28
- and small changes, 272–274
- strengths and weaknesses, 188

tailoring down, 189, 190

tailoring up, 189, 190–191

task lists, 105, 106, 107, 113, 116–118

teams. *See also* stakeholders, involving;

- TWGs (Technical Working Groups)
- example of coordinating systems
 - engineering and software engineering work, 50–51
 - role in deriving data measures, 37
 - using Scrum in DART case study, 210–211

technical leaders, 126–127

Technical Solution. *See* TS (Technical Solution) Process Area

templates

- in BOND case study, 114, 115–116
- developing, 114, 115–116
- in GEAR case study, 173
- relationship to process, 116
- role as process aids, 113–116
- sample for Organizational Process Definition (OPD), 303–306
- sample for Organizational Process Focus (OPF), 299–302
- sample for Project Management Plan (PMP), 287–292

10-percent rule, Scrum, 213–214

“thread” approach, 184–187

time-boxes, work, 44

“to-be” process view, 35, 241. *See also* “as-is” process view

training. *See also* mentoring

- in Agile organizations, 227
- in BOND case study, 126–127
- CMMI overview, 226–227
- as informal, 224
- just-in-time, 156, 224
- in LACM case study, 226
- in NANO case study, 150–151
- using role-based scenarios, 156

trust. *See* personal safety

TS (Technical Solution) Process Area

- in BOND case study, 127–128
- compared with Agile organizational process, 73
- defined, 6
- in GEAR case study, 173
- in NANO case study, 156
- SP (Specific Practice) 1.1, 128, 143

Turner, Rich, 206

TWGs (Technical Working Groups)

- in BOND case study, 66–67, 69, 70, 72
- conducting gap analysis interviews, 79, 80
- leaders compared with members, 78–80
- members as SMEs, 78–79
- purpose, 64
- role in Agile organizations, 64, 65, 78–79
- running, 79–81
- tailored, 64, 67–69, 70, 76–77, 182
- traditional compared with Agile, 69

U

uncertainty, 153, 163–165, 223, 227–228

V

VAL (Validation) Process Area

in BOND case study, 72

compared with Agile organizational process, 73

consolidating with VER (Verification), 72

Validation. *See* VAL (Validation) Process Area

value, defined, 17, 308

VER (Verification) Process Area

in BOND case study, 72

compared with Agile organizational process, 73

compared with PPQA (Product and Process Quality Assurance), 193

consolidating with VAL (Validation), 72

defined, 6

in NANO case study, 156

and peer review, 69–70, 199, 200–201, 290

role of criteria, 199, 200–201

in sample PMP template, 290

SP (Specific Practice) 1.2, 9

SP (Specific Practice) 1.3, 9, 143, 199

SP (Specific Practice) 2.1, 290

SP (Specific Practice) 2.2, 290

SP (Specific Practice) 2.3, 290

weighing need for, 82–83

verification, compared with peer reviews, 69–70, 199, 200–201. *See also* VER (Verification) Process Area

W

Waltzing with Bears (Demarco and Lister), 229

wannabe Agile, defined, 12, 308. *See also* Agile-like

weaknesses, potential

and BOND case study goals, 64, 66

defined, 59

example in Agile organization, 62–63

identifying against CMMI model, 61, 62–63, 64, 65

TWG responsibilities, 64, 66

“what” in project management plans, 38, 84, 101–102, 115, 126

“when” in project management plans, 84, 101, 104–107, 115

white papers, 145

white space tasks, 158, 159–160

“who” in project management plans, 84, 101, 102–104, 115