

SECOND EDITION

THE
PROJECT MANAGER'S
GUIDE TO MASTERING
AGILE

PRINCIPLES AND
PRACTICES FOR
AN ADAPTIVE
APPROACH

CHARLES G. COBB



WILEY

**›The Project
MANAGER'S GUIDE
TO MASTERING
AGILE**

›The Project MANAGER'S GUIDE TO MASTERING AGILE

Principles and Practices
for an Adaptive Approach

Second Edition

Charles G. Cobb

Agile Project Management Academy
<https://agileprojectmanagementacademy.com/>

WILEY

Copyright © 2023 by John Wiley & Sons Inc. All rights reserved.

Published by John Wiley & Sons, Inc., Hoboken, New Jersey.
Published simultaneously in Canada.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning, or otherwise, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 750-4470, or on the web at www.copyright.com. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permission>.

Trademarks: Wiley and the Wiley logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates in the United States and other countries and may not be used without written permission. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

Limit of Liability/Disclaimer of Warranty: While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the publisher nor author shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages. Further, readers should be aware that websites listed in this work may have changed or disappeared between when this work was written and when it is read. Neither the publisher nor authors shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For general information on our other products and services or for technical support, please contact our Customer Care Department within the United States at (800) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic formats. For more information about Wiley products, visit our web site at www.wiley.com.

Library of Congress Cataloging-in-Publication Data

Names: Cobb, Charles G., 1945- author.

Title: The project manager's guide to mastering agile : principles and practices for an adaptive approach / Charles Grinnell Cobb.

Description: Second edition. | Hoboken, New Jersey : Wiley, [2023] |

Includes bibliographical references and index.

Identifiers: LCCN 2022047014 (print) | LCCN 2022047015 (ebook) | ISBN 9781119931355 (paperback) | ISBN 9781119931362 (adobe pdf) | ISBN 9781119931379 (epub)

Subjects: LCSH: Agile software development. | Computer software—Development—Management.

Classification: LCC QA76.76.D47 C63 2023 (print) | LCC QA76.76.D47 (ebook) | DDC 005.1—dc23/eng/20220930

LC record available at <https://lccn.loc.gov/2022047014>

LC ebook record available at <https://lccn.loc.gov/2022047015>

Cover Design: Wiley

Cover Images: © Den Rise/Shutterstock

CONTENTS

PREFACE	xvii	Summary of Key Points	19
ACKNOWLEDGMENTS	xxiii	Discussion Topics	20
		Notes	21
① Introduction to Agile Project Management	1		
<hr/>			
The Chasm in Project Management Philosophies	2		
What's Driving These Changes?	3		
The Impact on the Project Management Profession	4		
The Evolution of Agile and Waterfall	6		
Definition of Waterfall	7		
Definition of Agile	7		
Comparison of Predictive (Plan-Driven) and Adaptive (Value-Driven) Approaches	8		
Which Approach Is Better?	10		
The Evolution of the Project Management Profession	11		
The Early History of Project Management	12		
Transformation of the Project Management Profession	13		
What's Driving This Change, and Why Now?	14		
Agile Project Management Benefits	17		
		Part 1 Fundamentals of Agile	
		② Agile History and the Agile Manifesto	25
		<hr/>	
		Agile Early History	25
		Dr. Winston Royce and the Waterfall Model (1970)	26
		Early Iterative and Incremental Development Methods (Early 1970s)	28
		Further Evolution of Iterative and Incremental Development (Mid-to-Late 1970s)	28
		Early Agile Development Methods (1980s and 1990s)	29
		Agile Manifesto (2001)	30
		Agile Manifesto Values	30
		Agile Manifesto Principles	33
		Summary of Key Points	39
		Discussion Topics	40
		Notes	41

③ Scrum Overview	43	Spikes	73
Scrum Framework	44	Progressive Elaboration	74
Sprints	45	Value-Based Functional Decomposition	74
Product Backlog	45	Agile Requirements Practices	75
Scrum Meetings	47	The Role of a Business Analyst in an Agile Project	75
Scrum Roles	50	“Just Barely Good Enough”	77
Product Owner Role	50	Differentiating Wants from Needs and the “Five Whys”	77
Scrum Master Role	51	MoSCoW Technique	78
Team Role	53	User Personas and User Stories	79
Scrum Values	54	User Personas	79
Commitment and Focus	55	User Stories	80
Openness	56	Epics	82
Respect	57	Product Backlog	83
Courage	58	What Is a Product Backlog?	83
General Scrum/Agile Principles	58	Product Backlog Grooming (Refinement)	84
Variability and Uncertainty	59	Summary of Key Points	86
Prediction and Adaptation	60	Discussion Topics	88
Validated Learning	61	Notes	89
Work in Progress	62		
Progress	63		
Performance	64		
Summary of Key Points	66		
Discussion Topics	66		
Notes	67		
④ Agile Planning, Requirements, and Product Backlog	69		
Agile Planning Practices	69		
Planning Strategies	70		
Capacity-Based Planning	72		
		Part 2 Agile Project Management Overview	
		⑤ Agile Development, Quality, and Testing Practices	95
		Agile Software Development Practices	96
		Code Refactoring	96
		Continuous Integration	97

Pair Programming	98	What Is a Kanban Process?	115
Test-Driven Development	99	Differences Between Scrum and Kanban	116
Extreme Programming (XP)	100	Work-In-Process (WIP) Limits in Kanban	117
Agile Quality Management Practices	100	Kanban Boards	118
Key Differences in Agile Quality Management Practices	100	Theory of Constraints	119
Definition of “Done”	101	Summary of Key Points	122
The Role of Quality Assurance (QA) Testing in an Agile Project	102	Discussion Topics	123
Agile Testing Practices	103	Notes	123
Concurrent Testing	103		
Acceptance Test-Driven Development	103	7 Agile Estimation	125
Repeatable Tests and Automated Regression Testing	104	Agile Estimation Overview	125
Value-Driven and Risk-Based Testing	104	What’s Different about Agile Estimation?	125
Summary of Key Points	104	Developing an Estimation Strategy	127
Discussion Topics	107	Management of Uncertainty	127
		Agile Estimation Practices	129
		Levels of Estimation	129
		Story Points	130
		Other Relative Sizing Techniques	133
		What Is Planning Poker?	134
		More Sophisticated Agile Estimation Techniques	134
		Velocity and Burn-Down/Burn-Up Charts	135
		Velocity	135
		Burn-Down Charts	135
		Burn-Up Charts	137
		Summary of Key Points	138
		Discussion Topics	139
		Notes	141
6 Time-Boxing, Kanban, and Theory of Constraints	109		
The Importance of Flow	111		
Small Batch Sizes	111		
Just-In-Time Production	111		
Concurrent Processing	111		
Time-Boxing	112		
Time-Boxing Advantages	112		
Additional Time-Boxing Productivity Advantages	113		
The Kanban Process	113		
Push and Pull Processes	114		

8 Agile Project Management Role 143

Agile Project Management Shifts in Thinking	145
Emphasis on Maximizing Value Versus Control	145
Emphasis on Empowerment and Self-Organization	147
Limited Emphasis on Documentation	148
Managing Flow Instead of Structure	149
Potential Agile Project Management Roles	149
Making Agile Work at a Team Level	150
Hybrid Agile Project Role	151
Enterprise-Level Implementation	152
Using Agile Concepts in Non-Agile Projects	155
AGILE, PMI®, AND PMBOK®	156
Prior PMBOK® Versions	156
What's Different about PMBOK® Version 7?	157
The Difference Between Explicit and Tacit Knowledge	159
Summary of Key Points	160
Discussion Topics	161
Notes	161

9 Agile Communications and Tools 163

Agile Communications Practices	163
Information Radiators	163
Face-to-Face Communications	165

Daily Scrum Meetings	166
Distributed Teams	166
Agile Project Management Tools	167
Benefits of Agile Project Management Tools	168
Characteristics of Enterprise-Level Agile Project Management Tools	169
Summary of Key Points	172
Discussion Topics	173
Notes	173

10 Learning to See the Big Picture 175

Systems Thinking	175
What Is Systems Thinking?	175
How Is Systems Thinking Used in Organizations?	176
Complex Adaptive Systems	177
What Are Complex Adaptive Systems?	177
Characteristics of Complex Adaptive Systems	179
Summary of Key Points	182
Discussion Topics	183
Notes	184

11 The Roots of Agile 185

Influence of Total Quality Management (TQM)	185
Cease Dependence on Inspection	186
Emphasis on the Human Aspect of Quality	188
The Need for Cross-functional Collaboration and Transformation	189

Importance of Leadership	190	Fit for Purpose	218
Ongoing Continuous Improvement	191	As a Transition to a Full Agile Approach	218
Influence of Lean Manufacturing	192	What Are the Benefits of a Hybrid Agile Approach?	219
Customer Value	195	General Benefits of a Hybrid Agile Approach	219
Map the Value Stream	196	Other Benefits of a Hybrid Agile Approach	219
Pull	196	What's Different About a Hybrid Agile Approach?	220
Flow	200	Key Differences from a Plan-driven (Waterfall) Approach	221
Respect for People	203	Key Differences from an Agile Approach	222
Perfection	204	Choosing the Right Approach	223
Principles of Product Development Flow	205	Most Important Factors to Consider	223
1. Economics	205	Other Factors to Consider	224
2. Queues: Actively Manage Queues	205	Summary of Key Points	224
3. Variability: Understand and Exploit Variability	206	Discussion Topics	225
4. Batch Size: Reduce Batch Size	206	Notes	225
5. WIP Constraints: Apply WIP Constraints	206		
6. Control Flow Under Uncertainty: Cadence and Synchronization	207		
7. Fast Feedback: Get Feedback as Fast as Possible	207		
8. Decentralize Control	207		
Summary of Key Points	208		
Discussion Topics	209		
Notes	210		

Part 3 Agile Project Management Planning and Management

12 Hybrid Agile Models 217

Why Would You Use a Hybrid Agile Approach?	218
--	-----

13 Value-Driven Delivery 227

Value-Driven Delivery Overview	227
What's Different about Value-Driven Delivery?	228
What Are the Advantages of Value-Driven Delivery?	229
Principles of Value-Driven Delivery	230
Focus on Customer Needs Rather Than Solutions	231
The Pareto Rule	232
Customer-Value Prioritization Overview	233

Levels of Prioritization	233	What Are the Benefits of a Product Roadmap?	258
Factors to Consider in Prioritization	234	Tips for Creating a Product Roadmap	258
MoSCoW Prioritization	234	Exploratory 360 Assessment	259
Value-Driven Delivery Tools	235	Agile Functional Decomposition	261
Minimum Viable Product	235	Relationship of Functional Decomposition to Agile	261
Minimum Marketable Feature	235	Functional Decomposition Examples	262
Summary of Key Points	236	Project Charter	264
Discussion Topics	238	Summary of Key Points	265
Notes	239	Discussion Topics	268
		Notes	269
14 Adaptive Planning	241		
Rolling-Wave Planning	242	16 Agile Stakeholder Management and Agile Contracts	271
Overview of Rolling-Wave Planning	242	What Is a Stakeholder?	272
Comparison of Planning Approaches	244	Internal Stakeholders	272
Progressive Elaboration and Multilevel Planning	247	External Stakeholders	272
Progressive Elaboration	247	Why Is Stakeholder Management Important?	273
Multilevel Planning	248	Stakeholder Management Can Be Difficult	273
Summary of Key Points	251	What Can Go Wrong?	273
Discussion Topics	253	Common Stakeholder Management Mistakes	274
Notes	253	Stakeholder Management Process	275
		Identify and Analyze Stakeholders	275
		Prioritize Stakeholders	276
		What's Different About Agile Stakeholder Management?	277
15 Agile Planning Practices and Tools	255		
Product/Project Vision	255		
What Is a Product/Project Vision?	255		
Product/Project Vision Examples	256		
Tips for Creating a Compelling Vision	257		
Product Roadmaps	258		

Advantages of an Agile Stakeholder Management Approach	277
Agile Stakeholders Have Rights and Responsibilities	278
Responsibility for Stakeholder Management in an Agile Environment	278
Eight Tips for Agile Stakeholder Management	278
Agile Contracts	280
How Would an Agile Contract Work?	280
Types of Agile Contracts	280
An Agile Contracting Example	282
Summary of Key Points	283
Discussion Topics	284
Notes	285

17 Distributed Project Management in Agile **287**

What Is Distributed Project Management?	287
Distributed Project Management Roles	290
Developer Project Management Responsibilities	291
Product Owner Project Management Responsibilities	292
Scrum Master Project Management Responsibilities	295
Summary of Key Points	295
Discussion Topics	297
Note	298

Part 4 Making Agile Work for a Business

18 Scaling Agile to an Enterprise Level **301**

Enterprise-Level Agile Challenges	302
Differences in Enterprise-Level Agile Practices	302
Reinterpreting Agile Manifesto Values and Principles	303
Enterprise-Level Obstacles to Overcome	307
Collaborative and Cross-Functional Approach	307
Organizational Commitment	308
Risk and Regulatory Constraints	309
Enterprise-Level Implementation Considerations	310
Architectural Planning and Direction	310
Enterprise-Level Requirements Definition and Management	311
Development Team Integration	313
Release to Production	314
Enterprise-Level Management Practices	315
Project/Program Management Approach	316
The Role of a Project Management Office (PMO)	317
Project/Product Portfolio Management	319
Summary of Key Points	321
Discussion Topics	323
Notes	323

19 Scaling Agile for Multiple-Team Projects 325

Scrum-of-Scrums Approach	325
Large-Scale Scrum (LeSS)	327
Nexus	328
Scrum at Scale	329
Summary of Key Points	330
Discussion Topics	331
Notes	331

20 Adapting an Agile Approach to Fit a Business 333

The Impact of Different Business Environments on Agile	334
Product-Oriented Companies	334
Technology-Enabled Businesses	335
Project-Oriented Businesses	336
Hybrid Business Model	337
Adapting an Agile Approach to a Business	337
Typical Levels of Management	338
Overall Business Management Level	338
Enterprise Product/Project Portfolio Management Level	342
Product Management Level	344
Project Management Level	344
Corporate Culture and Values	345
The Importance of Corporate Culture and Values	345
Value Disciplines	347
Summary of Key Points	352
Discussion Topics	353
Notes	353

21 Enterprise-Level Agile Transformations 355

Planning an Agile Transformation	355
Define the Goals You Want to Achieve	355
Becoming Agile Is a Journey, Not a Destination	356
Develop a Culture That Is Conducive to Agile	357
Manage Change	359
Don't Throw the Baby Out with the Bathwater	361
Tools Can Be Very Important	362
Adaptive Project Governance Model	364
Executive Steering Group	365
Project Governance Group	366
Working Group Forums	366
Project Teams	366
Summary of Key Points	366
Discussion Topics	368
Notes	369

Part 5 Enterprise-Level Agile Frameworks

22 Scaled Agile Framework® 373

SAFe® Competency Areas	373
SAFe® Core Values	377
Lean Agile Mindset in SAFe®	378
SAFe® Lean Agile Principles	379
SAFe® Artifacts and Supporting Capabilities	380
Summary of Key Points	380

Discussion Topics	382	Project Organization and Work Streams	403
Notes	383	High-Level Process Overview	403
23 Disciplined Agile Delivery (DAD®)	385	Requirements Management Approach	408
DA® Life Cycles	386	Project Scheduling Approach	411
Life Cycle Summary	387	Project Management Approach	411
DA® Roles	387	Communications Approach	412
Primary DA® Roles	387	Roles and Responsibilities	414
Supporting DA® Roles	390	Summary of Key Points	418
DA® Mindset	391	Discussion Topics	422
DA® Principles	391	25 Summary of Enterprise-Level Frameworks	423
DA® Promises	392	High-Level Comparison	423
DA® Guidelines	392	How These Frameworks Have Evolved	424
DA® Tool Kit	392	Discussion Topics	424
Summary of Key Points	393	Part 6 Case Studies	
Discussion Topics	395	26 “Not-So-Successful” Case Studies	427
Notes	395	Company A	428
24 Managed Agile Development Framework	397	Background	428
Managed Agile Development Overview	398	The Approach	428
The Macro-Level	399	What Went Wrong	428
The Micro-Level	399	Overall Conclusions	428
Objectives of Managed Agile Development	399	Company B	430
Plan-Driven Benefits	399	Background	430
Agile Benefits	400	The Approach	431
Key Differences from a Typical Waterfall Approach	400	What Went Wrong	431
Framework Description	403		

Overall Conclusions	432	Management of IT Resources	464
Company C	436	Time-to-Market	464
Background	436	Alignment and Collaboration	465
The Approach	436	Employee Productivity and Morale	465
What Went Wrong	436	Delivering More Frequent Value to Customers	465
Overall Conclusions	441	Openness and Transparency	465
Discussion Topics	441	Responsiveness and Adaptivity	465
Notes	441	Software Quality	465
27 Case Study: Valpak	443	Lessons Learned	466
Background	443	Forming Projects Around Teams	466
Valpak Stakeholders	443	Planning Team Capacity and Developing a Sustainable Pace	466
Valpak Franchisees	444	Using Sprint Reviews and “Science Fairs”	467
Consumers	444	Discussion Topics	467
Merchants	444	Notes	467
Corporate	444		
The Role of Technology at Valpak	445	28 Case Study: Harvard Pilgrim Health Care	469
Overview	445	Background	469
Scaled Agile Framework Implementation	445	Overview	470
Project Management Approach	451	Impact of Outsourcing and Vendor Partnering	472
Tools, Communication, and Reporting	452	Role of the PMO	473
Challenges	453	Project Governance	474
Cultural and Organizational Challenges	453	Role of Tools	476
Technical Challenges	457	Project Methodology Mix	476
Other Challenges	459	Project Portfolio Management	477
Overall Summary	461	Project Management Approach	478
Key Success Factors	461	Project Methodology	478
Results and Conclusions	463	Implementation Package Development	480
More Strategic Management Focus	463		

Implementation Package Refinement	480
Project Reporting	481
Contractual Relationship with Dell Services	482
Challenges	483
Cultural and Organizational Challenges	483
Contractual Challenges	486
Technical Challenges	489
Other Challenges	491
Key Success Factors	493
Conclusions	494
Lessons Learned	494
Discussion Topics	497
Notes	497

DSDM Overview	504
DSDM Principles	505
Challenges	507
Cultural and Organizational Challenges	507
Contractual Challenges	507
Technical Challenges	508
Overall Summary	509
Key Success Factors	509
Conclusions	510
Lessons Learned	512
Tailor the Agile Delivery Technique as Part of Early Project Planning	512
Agile Techniques Can Be Applied to New Project Environments	512
Discussion Topics	512
Notes	512

29 Case Study: General Dynamics, UK 499

Background	499
Overview	500
Requirements Prioritization and Management Approach	500
Contract Negotiation and Payment Terms	501
Planning Approach	501
Personnel Management	502
Communication	502
Management and Leadership Approach	503
Project Management Approach	503

30 Agile Hardware Development 513

Agile Hardware Development Overview	514
Hardware Development Challenges	514
The Speed of Change Is What Is Important	515
How to Put This Into Practice	516
How It's Done at Tesla	518
The Tesla Approach	519
Overall Summary	522
The Trade-Off Associated with Creativity and Innovation	522

Does the Tesla Agile Hardware Development Model Work for All Companies?	522	What to Do Differently	538
Discussion Topics	523	General Recommendations	540
Notes	523		
31 Non-Software Case Studies	525		
Agile Home Remodeling	525		
Background	525		
Why Was This Project So Difficult?	526		
Project Planning and Inception	526		
Project Scope	526		
Contractor Selection	527		
How Did the Project Work Out?	529		
What Were the Results?	529		
Overall Conclusions and Lessons Learned	529		
Agile Book Publishing	530		
How Was the Agile Approach Different?	530		
Lessons Learned	531		
Why Do People Have Trouble with This?	532		
Discussion Topics	533		
32 Overall Summary	535		
Evolution of the Project Management Profession	535		
The Future of Project Management	535		
What Does It Take to Become a Good Agile Project Manager in This New Environment?	537		
		Appendices	
		Appendix A Additional Reading and Resources	545
		Appendix B Glossary of Terms	547
		Appendix C Example Project/Program Charter Template	557
		Appendix D Suggested Course Outline	563
		INDEX	571

PREFACE

BACKGROUND

THE PROJECT MANAGEMENT PROFESSION is going through rapid and profound changes due to the widespread influence of Agile:

It is becoming very apparent that the classical plan-driven approach to project management that has been the predominantly accepted way of doing project management for a long time is no longer the only way to do project management:

- Rather than force-fitting all projects to a classical plan-driven project management approach, it is much better to fit the approach to the nature of the project.
- It's particularly important to develop an adaptive approach for projects that have a high level of uncertainty and/or where creativity and innovation are more important than planning and control to achieve predictability.

Those changes are likely to dramatically change the role of project managers in many environments as we have known them, raise the bar for the entire project management profession, and perhaps even eliminate the role of some Project Managers as we have known them.

From an Agile perspective, there have also been some equally significant changes:

- Agile and Scrum have grown over the years from a focus on small, single-team projects to much larger and more complex enterprise-level projects requiring multiple teams.
- That has made it evident that scaling Agile for that kind of project requires some kind of overall management framework which might include some kind of project/program management.

In both of these environments, there is a recognition that well-defined and prescriptive “cook-book” approaches are no longer effective for dealing with the complexity of these challenges. Instead, there is a need to focus on defining principles that need to be interpreted in the context of a given situation:

- In an Agile environment, both the Scaled Agile Framework and Disciplined Agile Delivery have moved away from relatively well-defined frameworks to a more flexible, principles-based approach.

- In a classical plan-driven project management environment, PMBOK® version 7 has moved away from previous versions of PMBOK® that attempted to define a checklist of things to do in almost every conceivable project management situation to a less well-defined principles-based approach.

The movement to a principles-based approach in both of these environments will require a lot more judgment and skill for determining and implementing the right approach for a particular project.

It is critical for Project Managers and the Project Management Profession, as a whole, to be proactive, anticipate the most likely impact of these challenges, and adapt accordingly.

It is also important for the Agile community to recognize the need to scale an Agile approach for managing large, complex enterprise-level projects.

This raises a number of questions including:

- What is the role of project management in an Agile project?
- Are classical project management principles and practices in conflict with Agile principles and practices?
- What needs to be done to extend Agile principles and practices to larger and more complex enterprise-level projects requiring multiple teams?
- How does a typical Project Manager shape his or her career to move in a more Agile direction?

Those are the needs and challenges that this book is intended to address. This book should be of value to both project managers and Agile professionals to develop a more integrated approach.

LEARNING OBJECTIVES

The following is a summary of what I believe are the most important steps in the journey toward becoming an Agile Project Manager (not necessarily in this order):

1. Develop new ways of thinking and begin to see Agile principles and practices in a new light as complementary rather than competitive with classical project management practices.
2. Gain an understanding of the fundamentals of Agile practices and learn the principles behind the Agile practices at a deeper level in order to understand why they make sense and how they can be adapted as necessary to fit a given situation.
3. Learn how to go beyond the classical notion of plan-driven Project Management and develop an adaptive approach to Project Management that blends both Agile and classical plan-driven Project Management principles and practices in the right proportions to fit a given project and business environment.

4. Understand the potential roles that an Agile Project Manager can play and begin to reshape Project Management skills around those roles.
5. Learn some of the challenges of scaling Agile to an enterprise level and develop experience in applying these concepts in large, complex, enterprise-level environments.

Relationship to My Online Agile Project Management Training Courses

I have successfully developed an online training curriculum in Agile Project Management that is currently offered on three different platforms with over 175,000 students. Anyone who has taken any of those courses should see a lot of similarity between the material in this book and the material in my online Agile Project Management training courses.

Summary of Changes in the Second Edition

Many of the current trends that are going on in the project management community now have validated the original direction of the book when it was originally published in 2015. As a result, the changes required in the second edition are not radical. Here's a summary of the most important areas of change:

1. More detail on Agile Project Management Planning and Management: One of the professors currently teaching a course based on the book wanted to see more detail on Agile Project Management Planning and Management; so I have added six new chapters on that in Chapters 12 through 17.
2. Less detail on Agile Project Management Tools: The original edition of the book included some detailed material on Agile Project Management tools. Since that time, there have been many changes in that area, and it is apparent that the area of Agile Project Management tools will continue to evolve significantly. For that reason, I have limited the material in this edition of the book to a general, high-level discussion of the capabilities of Agile Project Management tools without going into specifics on any particular tool.
3. Revisions to enterprise-level frameworks: There have been a number of significant changes in the two enterprise-level Agile frameworks that are covered in the book (Scaled Agile Framework and Disciplined Agile framework) and both of those chapters needed considerable changes.
4. Additional case studies: This edition of the book includes two new chapters on case studies. One is on "Agile Hardware Development" and includes material on the Agile implementation at Tesla and the other is on "Non-software Case Studies" to show how to use Agile outside of a software development environment for common projects.

HOW THIS BOOK IS ORGANIZED

Agile Project Management is an art that will take time for anyone to develop and master. There's a concept from martial arts called *shu-ha-ri* that is very appropriate here. It outlines the stages of proficiency someone goes through to develop mastery of martial arts techniques. The same concept can be applied to Agile Project Management:

- **“Shu”**: In the “shu” stage, the student learns to do things more-or-less mechanically, “by the book,” without significantly deviating from the accepted rules and practices and without improving any new techniques. This stage is equivalent to a new inexperienced project manager following PMBOK or other accepted practices like you would follow recipes in a cookbook without necessarily adapting those practices to fit the situation.
- **“Ha”**: In the “ha” stage, the student begins to understand the principles at a deeper level and learns how to improvise and break free from rigidly accepted practices, but it's important to go through the “shu” stage and gain mastery of the foundational principles before you start improvising—improvisation without knowledge is just amateurish experimentation.
- **“Ri”**: Finally, in the “ri” stage, the student gets to the highest level of mastery and is able to develop his/her own principles and practices as necessary.

The way the book is organized follows the *shu-ha-ri* approach to learning. The initial chapters of the book start out with a very basic understanding of the “mechanics” of Agile and learning how to do it “by the book.” That is equivalent to the “shu” level of training.

The book will go deeper into the principles behind Agile and why they make sense. It is essential to understand the principles at a deeper level before moving on to the “ha” level and know how to customize an approach to fit a given situation.

The final goal is to move to the master level or “ri” level where you will learn to go beyond current ways of implementing both Agile and plan-driven project management approaches and learn how to blend them together as needed to fit a given project and business environment. That goal will come from actual practice in implementing these ideas in real world situations; however, it is hoped that the information in this book and the case studies that are included will help Project Managers move rapidly in that direction.

The book is organized into Parts as follows.

Part 1 – Fundamentals of Agile

The first step in learning to become an Agile Project Manager is to learn the fundamentals of Agile, which includes not only the mechanics of how an Agile project based on Scrum works, but also understanding the principles behind it at a deeper level so that you can go beyond just implementing it “by the book.”

Part 2 – Agile Project Management Overview

Agile is causing us to broaden our vision of what a *Project Manager* is and that will have a dramatic impact on the potential roles that a Project Manager can play in an Agile environment. In fact, the role of a Project Manager at a team level in a typical Agile/Scrum project is undefined. That will cause us to rethink many of the things we have taken for granted about Project Management for a long time to develop a broader vision of what an *Agile Project Manager* is.

Part 3 – Agile Project Management Planning and Management

Part 2 provided an overview of Agile Project Management. In this Part, we will go into much more detail on Agile Project Management planning and management practices including:

- Hybrid Agile Models
- Value-Driven Delivery
- Adaptive Planning
- Agile Planning Practices and Tools
- Agile Stakeholder Management and Agile Contracts
- Distributed Project Management in Agile

Part 4 – Making Agile Work for a Business

There are many precedents for successful implementation of Agile principles and practices at a project team level; however, extending the Agile principles and practices to large-scale enterprise implementations and integrating with a business environment can be very difficult and introduces a number of new challenges, which include:

- Large, complex projects that are commonly found at an enterprise level may require some reinterpretation and adaptation of Agile principles and practices as well as blending those principles and practices with classical, plan-driven project management principles and practices in the right proportions.
- Integrating Agile principles and practices with higher levels of management typically found at an enterprise level, such as project portfolio management and overall business management can be difficult. However, if an Agile implementation is limited to a development process only and does not address integration with these higher-level processes, it is not likely to be effective and may result in failure.

This Part of the book is intended to address these topics and provide an understanding of the key considerations that need to be addressed for:

- scaling an Agile approach for multiple teams and for larger, more complex enterprise-level projects
- integrating an Agile development approach with a business environment
- planning and implementing an enterprise-level Agile transformation.

Part 5 – Enterprise-Level Agile Frameworks

Putting together a complete, top-to-bottom, enterprise-level Agile solution can be a very challenging task, especially when some of the pieces are not designed to fit together.

To simplify the design of an enterprise-level Agile implementation, it is useful to have some predefined frameworks that can be modified to fit a given business environment, rather than having to start from scratch to design an overall management approach.

Three frameworks are discussed in this Part:

- Scaled Agile Framework (SAFe[®]) (Dean Leffingwell)
- Managed Agile Development framework (Chuck Cobb)
- Disciplined Agile Delivery framework (Scott Ambler).

Part 6 – Case Studies

In any book of this nature, it's always useful to go beyond theory and concepts and show how companies have actually put these ideas into practice in the real-world. Of course, there is no canned approach that works for all companies—each of these case studies is different and shows how a different approach may be needed in different situations. It also includes a chapter on “Not-So-Successful” case studies, which shows some of the problems that can develop in an Agile implementation.

Part 7 – Appendices

The appendices to the book include additional supplementary information:

- Additional Reading List
- Glossary of Terms
- Example Project/Program Charter
- Suggested Course Outline for a graduate-level course to accompany this book

ACKNOWLEDGMENTS

I USED A VERY AGILE approach for writing this book as well as my previous books. It was a team effort of a number of people who worked with me collaboratively as the book was being written to provide feedback and inputs. I particularly want to thank the following people for their contributions to the original edition of this book:

FIRST EDITION

- Erik Gottesman, Director General Management at Sapient—Erik is a significant thought leader in this area. He played a huge role in helping me develop my two previous books on Agile Project Management and provided some good advice and input on this book as well.
- Dr. Michael Hurst, PMO Director at Harvard Pilgrim Health Care—Michael has played a significant role in providing input and advice for both this book and my last book and he also played a key role in providing a case study on Harvard Pilgrim Health Care that is included in this book.
- Andrew Bone, IT Program/PMO director—Andrew did a thorough review of the entire book, provided a number of good comments and inputs, and also sponsored a presentation on the book with the Long Island, New York, PMI Chapter.
- Liza Wood, Senior Production Manager at Warner Bros. Games—Liza also did a thorough review of the entire book on behalf of the PMI Agile Community of Practice and provided a very large number of excellent comments.
- Several companies generously shared case studies with the results of successful Agile implementations:
 - Michael Hurst, Director PMO, Harvard Pilgrim Health Care—Michael and Harvard Pilgrim shared the results of a very large and successful enterprise-level Agile transformation effort of more than 200 projects.
 - Stephanie Stewart(now Davis), Director of Agile Leadership at Valpak—Stephanie and Valpak shared the results of an enterprise-level implementation of the Scaled Agile Framework at Valpak.
 - Nigel Edwards, Program Manager at General Dynamics, UK—Nigel shared the results of a very large and complex, Agile fixed-price government contracting effort.

SECOND EDITION

I would particularly like to thank the following individuals who made significant contributions towards completing the second edition of this book:

- Stephan Wohlfahrt, Director, Corporate Project Management at Bosch—Stephan has played a significant thought leadership role at Bosch in championing a modern Agile Project Management approach throughout the whole company for all Bosch Project Managers worldwide. He did a very detailed review of the entire second edition of the book and provided numerous helpful comments and suggestions.
- Dr. Winston Gonzalez, D.M., SPC is an Instructor and Subject Matter Expert for the School of Continuing Studies at Georgetown University and is also on the Corporate Faculty at Harrisburg University. He teaches a course based on this book and made numerous valuable contributions and suggestions for its content.
- Joe Justice, Chairman of the Board, Agile Business Institute graciously allowed me to summarize some of his very excellent materials on hardware development practices; and, in particular, on his work with Tesla to implement those ideas. Joe is a real thought leader in this area and has made numerous contributions to implementing Agile in a hardware development environment.

I would like to also thank the following individuals who took the time to review an early draft of the second edition of this book and provided very helpful feedback, comments, and suggestions: Kiron D. Bondale, PMP, PMI-ACP, PSM II, ICP-ACC, PMI-RMP, DASSM, DAC, DAVSC, Senior Consultant, World Class Productivity Inc., and Dr. Monica Kay, PMP, Adjunct Professor, Morgan State University.

1

Introduction to Agile Project Management

OVER THE PAST 20 TO 25 YEARS, there has been a rapid and dramatic adoption of Agile methodologies and this trend has significantly accelerated in the last few years:

The “15th State of Agile Report,” published by Digital.ai, comments:

- This year’s findings indicate significant growth in Agile adoption within software development teams, increasing from 37% in 2020 to 86% in 2021.¹

Business Wire (a Berkshire Hathaway company) comments:

- Driven by the global pandemic, Agile adoption rates double in non-IT lines of business with continued, strong adoption in software development.
- More than 90 percent of respondents say their company practices Agile, with most saying either the majority of or even all company teams have adopted Agile practices.
- Rapid Agile adoption fuels an increase in adoption of other trends, including DevOps transformation and value stream management (VSM) initiatives for more than two-thirds of organizations.
- Post-pandemic, a vast majority of IT respondents expect to permanently work remotely, making Agile adoption critical to driving collaboration and success across a globally distributed workforce.²

These statistics indicate that Agile is not a fad, it is having a significant impact on the way projects are managed, it’s definitely here to stay, and it is significantly accelerating in recent years. This trend has a significant impact on the career direction of project managers who have come from a classical plan-driven project management background since there is no formal role for a project manager at the team level in an Agile project.

Note: Throughout this book, we will focus heavily on Agile as a project management approach because that is where it has the most important impact for project managers; however, it's important to realize that Agile is really a much broader way of thinking and is not limited to projects. It is also not specifically limited to software projects although that is where it is most heavily used.

THE CHASM IN PROJECT MANAGEMENT PHILOSOPHIES

Despite this rapid and sustained proliferation of Agile, there is still somewhat of a chasm between the Agile and classical plan-driven project management communities. When I published the first edition of this book in 2015, that chasm was large:

- There had been only a limited amount of progress at that time on developing a more integrated project management approach that embraces both Agile and classical plan-driven project management principles and practices.
- Many project managers had been heavily indoctrinated into a classical plan-driven project management approach and seemed to see Agile and classical plan-driven project management principles and practices as competitive approaches that conflict with each other, and they essentially treated them as two separate and independent domains of knowledge.
- Considerable polarization between these two communities was based on some part on myths, stereotypes, and misconceptions about what *Agile* and *project management* are that existed at that time.

Since that time, that chasm has narrowed considerably but there is still a gap that needs to be closed. Many people still seem to think that there is a binary and mutually exclusive choice between “Agile” and “Waterfall.” The ideal goal would be to have a seamless integration of project management approaches from heavily plan-driven (Waterfall) at one extreme to heavily adaptive (Agile) at the other extreme with lots of alternatives between those two extremes as shown in Figure 1.1.

That leaves many project managers in a conundrum to try and figure out how these two very different approaches to project management can be integrated together. A major goal of this book is:

- to help project managers understand the impact of Agile on the project management profession
- to broaden and expand their project management skills as needed to develop a more integrated approach to adapt to this new environment.

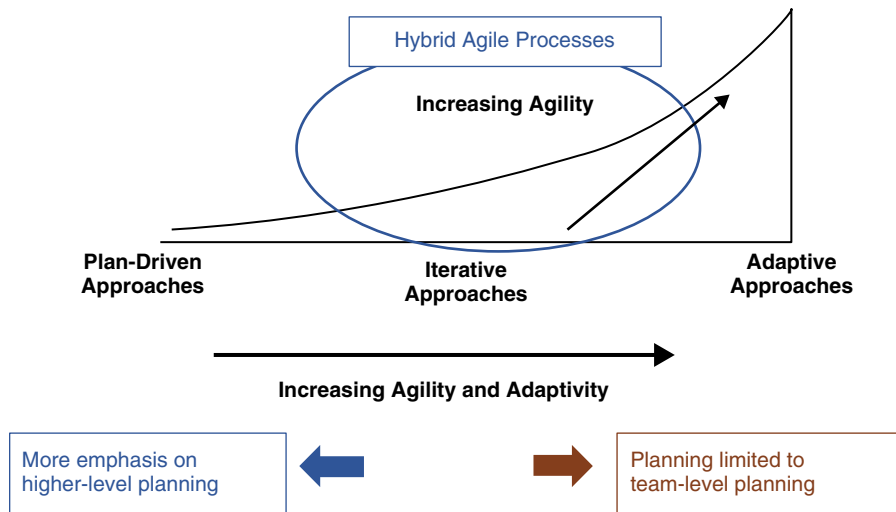


FIGURE 1.1 Spectrum of plan-driven and adaptive approaches

What's Driving These Changes?

In a classical plan-driven project management environment, a project was deemed to be successful if it delivered well-defined requirements on-time and within the approved budget. In today's world:

- There is a much higher level of uncertainty, which, in many cases, makes it very difficult, if not impossible, to document firm and well-defined requirements prior to the start of the project. In that environment, a much more flexible and adaptive approach is needed to further define and elaborate the detailed project requirements as the project is in progress.
- We also live in a very competitive environment where there is a much greater need for creativity and innovation to maximize the business value of the solution. An overemphasis on planning and control can stifle creativity and innovation. In fact, there have been many projects that have delivered well-defined requirements and met their cost and schedule goals but failed to deliver an acceptable level of business value.

This does not mean that a classical plan-driven project management is obsolete and no longer useful, but we need to recognize that it does have limitations and fit the project management approach to the nature of the project rather than force-fitting all projects to a classical plan-driven approach.

The important thing to consider is “value.” What is the “value” that the project is intended to produce? Meeting a cost and schedule goal for delivering well-defined requirements certainly has some value in many situations, but it is not necessarily the most important (or only) value in

a given project. The choice of an appropriate methodology for a project will depend on a number of factors:

1. **The level of uncertainty in the project:** A project that has a higher level of uncertainty in the requirements would naturally lean more towards a more flexible and adaptive approach.
2. **The level of training and sophistication of the project team:** It takes a considerable amount of skill and judgment to use an Agile approach successfully and it should not be attempted unless the team has been properly trained in Agile, the primary roles of Scrum Master and Product Owner are in place, and the necessary tools to support the project are also in place.
3. **The relationship with the customer of the project:** A classical plan-driven project is typically based on somewhat of a contractual relationship with the customer:

- The customer expects the project to be delivered as defined in the requirements within the approved cost and schedule goals.
- The customer does not need to be heavily involved in the implementation of the project until it is time to approve the final deliverables.

An Agile approach requires a much more collaborative approach with the customer. The customer needs to share responsibility for the successful completion of the project team by:

- taking an active role in the project to provide feedback and inputs on incremental results
- further defining and elaborating requirements as the project is in progress.

This means that there is no longer only one way to do project management and it takes a considerable amount of skill and organizational maturity to fit the most appropriate project management approach to the nature of the project:

- Not only is it important that the individuals responsible for product development are trained and skilled in an Agile approach.
- In addition, the businesspeople who are required to take an active role in the process need to understand how the process works.
- The overall organization needs to be committed to whatever level of organizational change that may be needed to make it successful.

The Impact on the Project Management Profession

This isn't just a matter of getting another certification—it can require a major shift in thinking for many traditional project managers that will take time and experience to develop. The Project Management Institute (PMI) has created the PMI-ACP® (Agile Certified Practitioner) certification, which has been very successful and is a great step in the right direction—but it doesn't go far enough, in my opinion.

- It doesn't test whether a project manager knows how to blend Agile and classical project management principles and practices in the right proportions to fit a given situation, and that is the real challenge that many project managers face.
- PMI-ACP is also not designed around a specific Agile role as many other Agile certifications are and the role that an Agile Project Manager might play is still somewhat undefined.

A lot of the polarization that has existed between the Agile and classical plan-driven project management communities has been rooted in some well-established stereotypes of what a *project manager* is that are based on how typical projects have been managed in the past. The role of a project manager has been so strongly associated with someone who plans and manages projects using classical plan-driven project management approaches that many people cannot conceive of any other image of a project manager. It's time to develop a new vision of what an *Agile Project Manager* is that goes beyond all those traditional stereotypes and fully integrates *Agile* within the overall portfolio of project management principles and practices.

It feels very similar to an evolution that took place when I worked in the quality management profession in the early 1990s. Up until that time, the primary emphasis in quality management had been on *quality control*, and inspection, and the image of a *quality manager* was heavily based on that role:

- The predominant quality management approach was based on final inspection of products prior to shipping them to the customer and rejecting any that didn't meet quality standards. It's easy to see how that approach was inefficient, because it resulted in a lot of unnecessary rework to correct problems after the fact, and it also wasn't that effective because any inspection approach is based on sampling, and it is impractical to do a 100% sample. For that reason, it can result in mediocre quality.
- A far better approach was to go upstream in the process and eliminate defects at the source by designing the process to be inherently more reliable and freer of defects, and build quality into the inherent design of the products. That didn't mean that the prior emphasis on quality control and inspection was obsolete and eliminated; it was just not the *only* way to manage quality and wasn't the most effective approach in all situations.

That was a gut-wrenching change for many in the quality management profession—instead of being in control of quality and being the gatekeeper with the inspection process, a good quality manager needed to become more of a coach and a consultant to influence others to build quality into the way they did their work. This changed the nature of the work dramatically for many in the quality management profession and eliminated a number of traditional quality management roles that were based on the old quality control and inspection approach. The similarity to the changes going on in the project management profession should be apparent:

- To be successful in more uncertain environments, project managers need to be able to take an adaptive approach that is appropriate to the level of uncertainty in the project and integrate

quality into the process rather than relying on final acceptance testing at the end of the project to validate the product that is being produced.

- They also need to give up some of the control that has become associated with the project management profession—in some cases, they may need to become more of a coach and a consultant to influence others rather than being in absolute control of a project.

This can dramatically change the role of a project manager. In some situations, the role of a project manager as we've known it may no longer exist. For example, at a team level in an Agile project, you probably won't find anyone with a title of *Project Manager* because the project management functions have been absorbed into other roles and are done very differently. That doesn't mean that *project management* is no longer important, but it may cause us to dramatically rethink what project management is in a much broader context than the way we might have thought about it in the past.

THE EVOLUTION OF AGILE AND WATERFALL

You will often hear people make a comparison between Agile and Waterfall. Many of those discussions are polarized and position them as competitive approaches. Here's an example:³

According to the 2012 CHAOS report, Agile succeeds three times more often than Waterfall. Because the use of Agile methodologies helps companies work more efficiently and deliver winning results, Agile adoption is constantly increasing.

While that statement is generally true, it's an oversimplification. There are at least two problems with that kind of statement:

1. It makes it sound like there are only two binary, mutually exclusive choices: Agile and Waterfall.
2. The meaning of the words *Agile* and *Waterfall* are typically not well-defined and are used very loosely.

For those reasons, I prefer to avoid comparing Agile to Waterfall because it tends to be a very polarized discussion—I prefer to take a more objective approach that is based on a comparison between a plan-driven and an adaptive (value-driven) approach to project management. So, let's first define both *Agile* and *Waterfall*, and then compare the two approaches.

Definition of Waterfall

The word *Waterfall* actually has a very specific meaning, but that's often not how the word is really used:

The Waterfall model is a popular version of the systems development life cycle model for software engineering. Often considered the classic approach to the systems development life cycle, the Waterfall model describes a development method that is linear and sequential. Waterfall development has distinct goals for each phase of development. Imagine a Waterfall on the cliff of a steep mountain. Once the water has flowed over the edge of the cliff and has begun its journey down the side of the mountain, it cannot turn back. It is the same with Waterfall development. Once a phase of development is completed, the development proceeds to the next phase and there is no turning back.⁴

Another aspect to the Waterfall model is that it is plan-driven; it attempts to define and document detailed requirements and a plan for the entire project prior to starting the project.

- When someone makes a statement comparing Waterfall to Agile, the word *Waterfall* is often used very loosely to refer to any kind of plan-driven methodology, and that's not really a very accurate and meaningful comparison.
- In some other comparisons like this, the word *Waterfall* refers to a general style of project management that obsessively emphasizes predictability and control over agility, and that's just bad project management. The Waterfall model will be discussed in more detail in Chapter 2.

Definition of Agile

Officially, Agile is defined by the principles and values of the Agile Manifesto of 2001 which will be discussed in Chapter 2. Agile is also an umbrella term used by many Agile practitioners to refer to different methods and frameworks that are based on adaptive, experimental, and extreme programming practices that have emerged since the mid-to-late 1990s.⁵

From a general perspective, Agile is a flexible and adaptive approach for developing and optimizing solutions in an uncertain environment. It is both incremental and iterative:

- “Incremental” means that the solution is broken up into “chunks” that are developed and tested individually and might also be released individually rather than waiting for the entire solution to be developed, tested, and released as a whole.
- “Iterative” means that the solution is progressively optimized and refined based on user feedback and inputs to maximize the value of the solution to the users.

It is particularly well suited for an environment with a high level of uncertainty because the process can start with only a high-level view of the project goals and requirements and those goals and requirements can be further elaborated and refined as the project is in progress. Of course, that does not mean that all Agile projects start with only a high-level view of the project goals and requirements. That will vary from one project to the next depending on the level of uncertainty in the project.

In actual practice, the meaning of the word *Agile* in this kind of comparison is also somewhat elusive because it has taken on some very strong connotations in actual usage. At a project level, the word *Agile* has frequently taken on a specific connotation associated with using the Scrum methodology on software development projects.

Scrum is an Agile software development framework based on multiple small teams working in an intensive and interdependent manner. The term is named for the scrum (or scrummage) formation in rugby, which is used to restart the game after an event that causes play to stop, such as an infringement. Scrum employs real-time decision-making processes based on actual events and information.

That definition has evolved over the years as Scrum has become somewhat of a de-facto standard for Agile projects; however, the original definition of *Agile* conceived in the *Manifesto for Agile Software Development*,⁶ published in 2001, was much broader than that. Better known as the Agile Manifesto, it laid out some simple and general principles and values that can apply to any kind of project (not just software development) (see Chapter 2).

Comparison of Predictive (Plan-Driven) and Adaptive (Value-Driven) Approaches

Traditional, classical plan-driven project management is a style of project management that is applied to projects where the requirements and plan for completing the project can be defined to a large extent prior to implementing the project. The emphasis in this style of project management is on predictability, and for that reason, the PMI calls it “predictive.” However, *plan-driven* is a relative term, and you won’t find many projects that start out with an absolutely rigid plan that is not expected to change at all. This style of project management is often loosely called “Waterfall.”

In contrast, an adaptive (or value-driven) style of project management starts the implementation of a project with a less well-defined plan of how the project will be implemented and recognizes that the requirements and plan for the project are expected to evolve as the project progresses. *Adaptive* is also a relative term; you won’t find many projects that have no plan whatsoever of how the project will be done.

The important point is that the terms *predictive (plan-driven)* and *adaptive (value-driven)* are relative—they are not discrete, binary, mutually exclusive alternatives. They should imply a continuous range of approaches with different levels of upfront planning. Table 1.1 shows a comparison of the two approaches.

TABLE 1.1 Comparison of approaches

	Classical project management approach	Hybrid	Agile approach
Project management approach	Plan-driven, predictive: The emphasis is on planning and predicting costs and schedules for projects with well-defined requirements	Blend of both	Value-driven, adaptive - The emphasis is on maximizing the value of the solution in an uncertain environment with uncertain requirements
Project management responsibility	Typically, a project manager is responsible for managing the overall project to meet approved cost and schedule goals		There may be no single “Project Manager” at the team level. Project management responsibility is typically distributed (see Chapter 17)
Project environment	Best suited for projects with a lower level of uncertainty where some level of predictability of costs and schedules is important		Best suited for projects with a higher level of uncertainty where some level of flexibility and adaptivity is needed to define the solution
Requirements management	Detailed requirements are defined prior to the beginning of the project		Detailed requirements are further defined and elaborated as the project is in progress
Change management	Changes in scope are controlled in order to maintain control over cost and schedule estimates		Changes are encouraged in order to support flexibility and adaptivity
Customer relationship	Contractual based on well-defined requirements		Collaborative based on a spirit of trust and partnership
Development management	Typically, Waterfall or an equivalent SDLC with controlled sequential phases		Typically, Scrum or an equivalent incremental and iterative approach
Solution delivery	The entire solution is tested and delivered all-at-once at the end of the project		The solution is tested and delivered incrementally at the end of each sprint and release
Testing	Testing is typically done sequentially by a separate and independent QA organization		Testing is integrated into the development effort and is done concurrently with development