Darrell Rigby • Sarah Elk • Steve Berez
BAIN & COMPANY, INC

DOING AGILE RIGHT

Transformation Without Chaos

HARVARD BUSINESS REVIEW PRESS

Contents

1. Praise Page
2. Title Page
3. Copyright
4. Dedication
5. Contents
6. Introduction: The Unbalanced Company
7. 1. How Agile Really Works
8. 2. Scaling Agile
9. 3. How Agile Do You Want to Be?
10. 4. Agile Leadership
11. 5. Agile Planning, Budgeting, and Reviewing
12. 6. Agile Organization, Structures, and People Management
13. 7. Agile Processes and Technology
14. 8. Doing Agile Right
15. Appendix A: A Leadership Team's Agile Manifesto
16. Appendix B: Definitions of Operating Model Components
17. Appendix C: Research Notes Summary
18. Notes
19. Index
20. Acknowledgments

21. About the Authors

Guide

1. Cover
2. Table of Contents
3. Start of Content
4. Index
5. Acknowledgments

Pagebreaks of the print version

1. Cover Page
2. a
3. iii
4. iv
5. v
6. vii
7. 1
8. 2
9. 3
10. 4
11. 5
12. 6
13. 7

14. 8
15. 9
16. 10
17. 11
18. 12
19. 13
20. 14
21. 15
22. 16
23. 17
24. 18
25. 19
26. 20
27. 21
28. 23
29. 24
30. 25
31. 26
32. 27
33. 28
34. 29
35. 30
36. 31
37. 32
38. 33
39. 34

40. 35
41. 36
42. 37
43. 38
44. 39
45. 40
46. 41
47. 42
48. 43
49. 44
50. 45
51. 46
52. 47
53. 48
54. 49
55. 50
56. 51
57. 52
58. 53
59. 54
60. 55
61. 56
62. 57
63. 58
64. 59
65. 60

66. 61
67. 62
68. 63
69. 65
70. 64
71. 66
72. 67
73. 68
74. 69
75. 70
76. 71
77. 72
78. 73
79. 74
80. 75
81. 76
82. 77
83. 78
84. 79
85. 81
86. 80
87. 82
88. 83
89. 84
90. 85
91. 86

92. 87
93. 88
94. 89
95. 90
96. 91
97. 92
98. 93
99. 94
100. 95
101. 96
102. 97
103. 98
104. 99
105. 100
106. 101
107. 102
108. 103
109. 104
110. 105
111. 106
112. 107
113. 108
114. 109
115. 110
116. 111
117. 112

118. 113
119. 114
120. 115
121. 116
122. 117
123. 118
124. 119
125. 120
126. 121
127. 122
128. 123
129. 124
130. 125
131. 126
132. 127
133. 128
134. 129
135. 130
136. 131
137. 132
138. 133
139. 134
140. 135
141. 136
142. 137
143. 138

144. 139
145. 140
146. 141
147. 142
148. 143
149. 144
150. 145
151. 146
152. 147
153. 148
154. 149
155. 150
156. 151
157. 152
158. 153
159. 154
160. 155
161. 156
162. 157
163. 158
164. 159
165. 160
166. 161
167. 162
168. 163
169. 164

170. 165
171. 166
172. 167
173. 168
174. 169
175. 170
176. 171
177. 172
178. 173
179. 174
180. 175
181. 176
182. 177
183. 178
184. 179
185. 180
186. 181
187. 182
188. 183
189. 184
190. 185
191. 186
192. 187
193. 188
194. 189
195. 190

196. 191
197. 192
198. 193
199. 194
200. 195
201. 196
202. 197
203. 198
204. 199
205. 200
206. 201
207. 202
208. 203
209. 204
210. 205
211. 206
212. 207
213. 208
214. 209
215. 210
216. 211
217. 212
218. 213
219. 214
220. 215
221. 216

222. 217
223. 218
224. 219
225. 220
226. 221
227. 223
228. 224
229. 225
230. 226
231. 227
232. 228
233. 229
234. 230
235. 231
236. 232
237. 233
238. 234
239. 235
240. 236
241. 237

"Highly recommended. A sparkling book that is not only easy to read and digest in one day but also packed with useful tips. Applicable to everyone, with clear do's and don'ts for all leadership styles. This book is a must-read for skeptics as well as believers that agile can lead to sustainable success."

—**CARLO VIVALDI,** Co-Chief Operating Officer, UniCredit

"No matter where your company is on its journey to agile, *Doing Agile Right* serves as a useful guidebook for senior management teams. The authors bring decades of experience and perspective to demystifying an often misunderstood topic."

—**RITCH ALLISON,** CEO, Domino's Pizza

"Business leaders responsible for leading transformative change will benefit greatly from *Doing Agile Right*. The specific examples and empirical evidence show how agile helps a company innovate and evolve."

—**PAUL SANFORD,** Senior Vice President, Solutions Delivery, Cigna

"*Doing Agile Right* lays out the steps organization-wide that allow traditional companies to perform software development like digital natives. CIOs and their business partners in the digital transformation journey will find this book an invaluable road map."

—**MICHELLE A. ROUTH,** Chief Information Officer, CARE USA

"Having experienced as a practitioner many of the challenges of large-scale agile transformation, I value this unfiltered view of the journey to agility. Every step has practical examples highlighting do's and don'ts."

—**PRAT VEMANA,** Senior Vice President and Chief Digital Officer, Kaiser Permanente

Darrell Rigby • Sarah Elk • Steve Berez
BAIN & COMPANY, INC.

DOING


AG
ILE

RIGHT

Transformation Without Chaos

HARVARD BUSINESS REVIEW PRESS • BOSTON, MASSACHUSETTS

Library of Congress Cataloging-in-Publication Data

Names: Rigby, Darrell, author. | Berez, Steve, author. | Elk, Sarah, author.
Title: Doing agile right : transformation without chaos / by Darrell Rigby, Steve Berez, and Sarah Elk (Bain & Company, Inc.).
Description: Boston, MA : Harvard Business Review Press, [2020] | Includes index.
Identifiers: LCCN 2019054607 | ISBN 9781633698703 (hardcover) | ISBN 9781633698710 (ebook)
Subjects: LCSH: Agile project management. | Success in business.
Classification: LCC HD69.P75 R33 2020 | DDC 658.4/06—dc23
LC record available at https://lccn.loc.gov/2019054607

ISBN: 978-1-63369-870-3
eISBN: 978-1-63369-871-0

To those who share our belief that great businesses should produce better people—and to the colleagues and clients who make our own work and growth so fulfilling.

Contents

1. Introduction: THE UNBALANCED COMPANY

2. HOW AGILE REALLY WORKS
3. SCALING AGILE
4. HOW AGILE DO YOU WANT TO BE?
5. AGILE LEADERSHIP
6. AGILE PLANNING, BUDGETING, AND REVIEWING
7. AGILE ORGANIZATION, STRUCTURES, AND PEOPLE MANAGEMENT
8. AGILE PROCESSES AND TECHNOLOGY
9. DOING AGILE RIGHT

10. Appendix A: A Leadership Team’s Agile Manifesto
11. Appendix B: Definitions of Operating Model Components
12. Appendix C: Research Notes Summary
13. Notes
14. Index
15. Acknowledgments
16. About the Authors

Introduction

THE UNBALANCED COMPANY

Agile—the business philosophy that relies on fast-moving, self-managing teams for innovation—has officially entered the mainstream of corporate management. Tour almost any large company these days and you will find scores of agile teams working to improve customer experiences and business processes. John Deere has used agile methods to develop new machines, USAA to transform its customer service, and 3M to run a major merger integration. Bosch—a global supplier of technology and services with more than 400,000 associates—has adopted agile principles to guide a step-by-step reshaping of the company. Digital natives such as Amazon, Netflix, and Spotify have incorporated agile methods into a wide range of innovation activities. Meanwhile, agile has virtually taken over IT departments, themselves the source of countless innovations. At last count, 85 percent of software developers use agile techniques in their work.[1]

The reasons for agile's rapid spread are neither obscure nor surprising. Most big companies find it difficult to innovate. They are weighted down by the structures and procedures of bureaucracy. Agile liberates the innovative spirit that so many organizations stifle. It helps companies reshape both what they offer their customers and how they operate internally. It transforms the work environment, making people's jobs more rewarding.

These are grand claims, but the data support them. Study after study find conclusively that agile teams are far more successful at innovation than teams that work in traditional fashion. Improvements come rapidly and at less expense. Satisfaction and engagement among employees rise. Moreover, companies can implement agile without the need to spin off separate business units or hide skunkworks from hierarchies. They can deploy agile teams in any business or function that might benefit from them, including corporate headquarters. Once they have learned the basics, they can scale agile, establishing hundreds of individual teams or teams of teams to tackle large projects. Saab's aeronautics business created more than one hundred agile teams operating across software, hardware, and fuselage for its Gripen fighter jet—a $43 million item that is certainly one of the most complex products in the world. Military analyst Jane's has deemed the Gripen the world's most cost-effective military aircraft.

So agile is spreading, and agile teams are mostly achieving their objectives. It looks like encouraging progress toward an appealing vision. Is there anything wrong with this picture?

Certainly there's nothing wrong with the basic idea. We are business consultants, and we have seen the power and potential of agile in hundreds of companies around the globe. We have helped many of these companies implement agile. We count ourselves among its biggest fans.

But as with so many good ideas, the practice sometimes belies the promise. Agile has spread so rapidly that it threatens to spin out of control. Along with the companies that use it effectively are those that misunderstand or misuse the ideas. They may be egged on by some zealot who promises the world. They may sign on to an agile transformation before they know anything about what such an effort might entail. They may use agile terminology to camouflage distinctly nonagile objectives.

The outcome of these misuses in many companies is chaos rather than constructive change. But the damage is greater than any one company's experience. When agile is done wrong, it almost always leads to lousy results. Lousy results lead to nervous customers, dissatisfied employees, activist investors, and a push to replace the management team. Replacement managers are understandably skeptical of any strategies that got the prior regime fired. They are likely to clean house, disband agile teams, and (probably) launch a round of layoffs. It's a version of Gresham's law: bad agile drives out good. If that happens too often, agile will be discredited—and the business world will be back where it started, with top-heavy bureaucratic corporations struggling hopelessly to keep up with brash upstarts and rapidly changing markets.

So in this book we want to bring agile down to earth, to separate Agile Done Right from Agile Done Wrong. Here, in the introduction, we'll focus on the wrongheadedness, the potholes and pitfalls, the ways in which companies have already misunderstood or misused agile. We hope that the lessons and cautionary tales will inoculate you against the idea that agile is some kind of magic quick fix. But we'll also introduce some of the ideas that inform the chapters that follow—chapters that will tell you how to do agile right. We'll provide a road map to these chapters, and we'll summarize the research that underlies the book. Doing agile right may take more time and experimentation than doing it wrong—but it's the only way to get the results that the philosophy promises.

Doing Agile Wrong

In the movie *The Princess Bride*, the swordsman Inigo Montoya famously chides the cunning Vizzini, "You keep using that word. I do not think it means what you think it means.” So it is with agile. Executives often fail to understand how agile operates and where and why it has succeeded. That doesn’t stop them from throwing around the terminology or from making assumptions about agile that simply aren’t true.

Some of these misunderstandings reflect the fact that agile methods—especially those relating to expanding the scope and scale of agile innovation teams—are still relatively new, and many business leaders haven’t yet learned much about them. It’s common to hear, for example, that agile is great but only for technology-based innovations and the IT departments that generate those innovations. This will come as news to National Public Radio, which used agile methods to create new programs; to the people developing the Gripen fighter jet or Haier’s home appliances; and to the many companies using agile to reshape their supply chains. Historically, to be sure, agile has spread most rapidly in IT. But it is widely and successfully used in many other contexts, some of which have only minor technology-based components.

Other failings, though we hate to say it, reflect a measure of cynicism on the part of corporate leaders. Consider the clever press release that Edward S. Lampert, CEO of Sears, issued in 2017: “In addition to the cost reduction target announced today, we continue to assess our overall operating model and capital structure to become a more agile … and innovative retailer focused on member experience.”[2] *Agile* in this context is a euphemism for layoffs. And Lampert isn’t alone. Every month we receive more requests for proposals that begin something like this: “The project objective (should you choose to accept it) is to reduce operating expenses by 30 percent this year while transforming the organization to agile ways of working and digital technology.”

The issuers of these requests don’t understand that there are basic inconsistencies between large, chaotic layoffs and agile. For one thing, large layoffs tend to happen in batches driven by precipitous restructurings or annual budgeting cycles. This is diametrically opposed to the continuous learning and adaptation processes prescribed by agile. For another, senior executives typically go behind closed doors to plan the layoffs, emerging with new structures and fixed targets. This is inconsistent with the agile principle of empowering people closest to the work to identify improvement opportunities. Worst of all, leaders who try to marry agile with layoffs inadvertently role-model antiagile behaviors. They create predictive, command-and-control events rather than agile, test-and-learn cultures. Moreover, research shows that large layoffs increase risk aversion and slow innovation. People scramble to learn new jobs. They battle for control of key operations, no matter what the organization chart says. They do everything possible to make sure they will have a chair next year, when the music is likely to stop once again. Mostly, they try to do the same things they know how to do, but with fewer people. This is not an environment in which agile can flourish.

But then there is a different kind of wrongheadedness, not generated by complete ignorance or cynicism. These misuses are propagated by well-meaning agile partisans. They are sold to leadership teams who badly want their companies to become nimbler and more innovative but who don’t really understand how agile operates. In our work with hundreds of companies launching thousands of agile initiatives, we commonly find three toxic mistakes.

AGILE, AGILE, EVERYWHERE

Some agile gurus pitch the approach as a panacea that must replace bureaucracy everywhere—in every company, in every business unit, in every function.

Consider a company that we’ll call MagicAgile. (This is a real company, but we don’t want to identify it because our conversations with its leaders were confidential.) The people at this company wanted to be like digital disruptors such as Spotify, the music streaming company that is well known for its agile innovation teams. So MagicAgile launched agile teams throughout its organization. It redesigned office spaces to create more open work areas. It innovated around customer and employee experiences. Through the eyes of agile evangelists, MagicAgile looked like a poster child—or at least it did if you ignore the fact that it lost about half of its market value in 2018 and early 2019 (or if you can claim with a straight face that things would have been even worse without the transformation, or that shareholder value doesn't matter anyway). But in candid meetings with the leadership team of MagicAgile, we heard nearly every executive voice frustrations with their less visible results. They said things like this: "Agile has created a leadership problem. We have no discipline or alignment. This is chaos." "We went too far. Everybody is talking about servant leadership and psychological safety. Nobody is allowed to use the word *manager*, and all managers have gone into hiding." "Responsibilities for P&Ls are getting confused." "Our leaders are criticized and ignored when we try to provide strategic guidance to our business units." "Agile became the objective. We are praying at the altar of a false church."

What the agile-everywhere zealots fail to understand is the time-tested virtues of bureaucracy—in some places and contexts. The same bureaucracy now so widely reviled as antithetical to change and innovation was itself one of the greatest innovations in the history of business. Hierarchical authority, specialized division of labor, and standard operating procedures—the hallmarks of the bureaucratic method—enabled companies to grow far larger than they had ever been. The principles of bureaucracy were taught as good management practices in business schools and corporate training programs. Companies learned the virtues of predictability and planning. Strong bureaucrats rose to the tops of their organizations.

Today we understand bureaucracy's limitations. The great German sociologist Max Weber, who was the first to offer a systematic description of bureaucracy and who well understood its efficiencies, famously warned that it could create a soulless "iron cage" that trapped people in dehumanizing organizations and limited their potential.[3] He was right: most people today work in bureaucracies, and most feel disengaged from their work. The widespread appeal to young people of start-ups and small businesses, as compared with careers spent climbing corporate ladders, reflects this flaw. Then, too, bureaucracies are terrible at innovating. A bureaucracy works when its organizational tasks—what to deliver and how to deliver it—are clear, stable, and predictable. Innovation, by definition, meets none of these criteria. These limitations have contributed to bureaucracy's bad reputation and to the growth of antibureaucratic approaches such as agile.

And yet. Consider the adverse consequences of encouraging wide variation, on-the-spot experimentation, and decentralized decision making—all hallmarks of agile—in areas such as food or drug safety, antidiscrimination and harassment policies, accounting standards, aircraft safety, quality controls, and manufacturing standards. Every company has to run its business, getting standardized products out the door and delivering predictable services to customers. Every company needs bureaucratic structures and procedures, including hierarchical approvals, specialized division of labor, and standard operating procedures, to do exactly that.

The challenge, in short, is not to replace bureaucracy with agile everywhere but to find the right balance between the two. Every company must run its businesses. It must be good at *operations*. Every company must also change the business, continuously introducing not just new products and services but new operating methods and procedures. It must be good at *innovating*. Although these tasks require different skills, they are not enemies. They are complementary, interdependent, mutually beneficial capabilities that need each other to survive. Insufficient focus on innovation leads to a static enterprise that will fail to adapt to changing conditions. Insufficient emphasis on operations creates chaos—poor quality, high costs, and dangerous risks to customers and to the business.

Most large companies today have tilted too far toward bureaucracy, starving innovation. They have created static organizations committed to delivering predictable results. This is why agile is so popular. But the solution isn't to tip the scale all the way in the other direction. Rather, it is to stick with bureaucratic rules and hierarchies where they are appropriate, humanizing them as much as possible, and at the same time to instill a healthy admixture of agile wherever it is appropriate. That may sound simple, but it isn't. Agile and bureaucracy are like oil and vinegar: they are good together, but they don't mix easily. (Sometimes the two act more like nitric acid and glycerol: they lead to explosions.) Agile teams thrive on doing things quickly. The teams try out new ideas, often before the ideas are fully formed, and test them with prospective customers. They don't respect red tape, and they don't follow detailed plans. If such teams are to thrive in an organization, they need a lot of freedom and a lot of support. Bureaucracies, of course, are just the opposite: they thrive on—indeed, require—tight control. They want to know exactly what a team has done so far, what it plans to do for the next twelve months, and how much it is all going to cost. To a traditional bureaucracy, agile teams can feel like foreign bodies infecting the organism. Like T cells in an immune system, bureaucrats often see their job as eliminating the infection or at least limiting its damage.

In a truly agile enterprise, bureaucracy and innovation become partners. They create a system where both elements improve and where people in each camp collaborate to generate superior results. We'll show how to harmonize the two in this book.

LET'S HAVE YOU FOLKS DO AGILE

Frederick Winslow Taylor claimed to take bureaucratic management from an art to a science. His stopwatch studies are classics in the annals of business. His 1911 book, *The Principles of Scientific Management*, outlined four fundamental tenets: (1) managers plan work, and workers do it; (2) managers scientifically analyze the most efficient ways for workers to work; (3) managers scientifically select and train the right workers for the right jobs; and (4) managers rigorously supervise workers as they perform tasks.[4] At the time, Taylor's methods drew harsh criticism for treating humans like machines. But his approach caught on, and indeed it has long outlived him—even today, many companies have plenty of managers and executives who are Taylorists at heart. And when Taylorists try to implement agile, bad things happen.

Here's how it often works: Top leaders plan the agile transformation for their subordinates, not for themselves. They create a high-powered program management office to drive the change. This office generates detailed budgets, milestones, and execution road maps, complete with Gantt charts and stoplight reporting systems, to ensure conformance to plans. It creates a slew of agile teams, typically led by Taylorists who are fresh from two days of training. When one of the teams registers a success, however tenuous, the program office broadcasts it, in hopes of convincing both internal and external audiences that the initiative is working precisely as planned. Meanwhile, the leadership team continues operating much as before, supervising and (often) micromanaging their subordinates, a group that now includes members of agile teams. These leaders frequently tell the teams not only what to do but how to do it. After all, isn't that the job of an executive?

But agile dies on the vine when it is micromanaged from above. All the agile language about self-management, testing and learning, and so forth begins to feel like a sham. Anyway, the tools of top-down management don't work in an agile environment. Benchmarks turn out to be worthless outside their unique contexts. Predictive plans are usually wrong, because they fail to recognize or adapt to unpredictable system dynamics. We use a survey called the Bain Agility Quotient to diagnose the health and maturity of agile initiatives throughout an organization. Wherever the Taylorist approach thrives, the perception gaps between senior executives and team members are large. Senior executives describe the company's agile initiatives as successful and satisfying. Team members, who are closer to the action, describe them as disappointing and frustrating, not much different from traditional task forces. At first we thought the executives must be lying, but we soon discovered they are merely out of touch. They are so distant from the agile work that they only know what subordinates tell them, and subordinates tell them only what they want to hear.

To be sure, some agile teams have succeeded even in Taylorist enterprises. They stay under executives' radar, and they thrive in spite of senior management rather than because of it. But a true agile transformation requires the active involvement and support of the company's leaders. Senior executives who really want to scale agile will do better by showing others how to do it than by sending subordinates off to training seminars. They themselves must understand agile, love it, and use its methods in teams of their own. Gandhi famously said, "If we could change ourselves, the tendencies in the world would also change." So it is with agile.

AGILE AS A QUICK FIX

A few companies, facing urgent strategic threats and in need of radical change, have pursued big-bang, everything-at-once agile transformations in some units. For example, in 2015 ING Netherlands anticipated rising customer demand for digital solutions and increasing incursions by new digital competitors known as fintechs. The management team decided to move aggressively. It dissolved the organizational structures of its most innovative functions, including IT development, product management, channel management, and marketing—essentially abolishing everyone's job. Then it created small agile "squads" and required nearly 3,500 employees to reapply for 2,500 redesigned positions on those squads. About 40 percent of the people filling the positions had to learn new jobs, and all had to profoundly change their mindsets.[5]

Experience has revealed countless problems with this approach. It confuses and traumatizes the organization. People aren't sure where to go or what to do. It assumes that thousands of individuals, most of whom have no experience or knowledge of agile, will suddenly understand and work according to its principles. Although radicalized converts have publicly touted their success, overall results frequently failed to meet unrealistic promises; stock prices (including ING's) have often declined, sometimes by 30 percent or more. Behind closed doors, these executives and their subordinates are more balanced, typically offering assessments that sound something like this: "Our leaders and culture were not ready for such radical change. The more we recited conventional clichés about 'ripping off the Band-Aid' and 'burning the boats' of retreat, the more we believed them. But nobody in our senior team had ever worked in an agile environment. We did not foresee or plan for the unintended consequences. Worse yet, we lost some great people who were branded as obstructionists for trying to point out those consequences. Our approach to agility was not very agile."

More common than big bangs in the quick-fix department is the bureaucrat's favorite innovation tool: copying others. Of course, executives use nicer names for it—benchmarking, competitive intelligence, or becoming a so-called fast follower—but it all boils down to copying. A favorite template is Spotify, well known for its original agile lexicon of squads, tribes, guilds, and the like. Some companies even find themselves copying some other company that originally copied Spotify.

The logic of copying is seductive. Agile pioneers like Spotify have spent years learning and applying agile principles. Why not replicate that success in six months? Particularly enticing is the idea that all you have to do is copy the pioneer's organization structure and office design. If you change the boxes and the layouts, you will surely force changes in how people approach their work. Changing the ways of working, in turn, will change outputs and outcomes. What could possibly go wrong?

Well, let us count the ways. For one, human organizations (like human bodies) are complex systems, which means that variables interact differently in different environments. Medications that work for some patients may be harmful to people with different genetics, genders, ages, or foods in their system. Managers who attempt to paste the structures of innovation departments at one company onto the entire enterprise of another are bound to produce unintended consequences. Spotify itself is sophisticated enough to understand this. It designed its engineering model to match its unique culture, capitalizing on the trust and collaboration inherent in the department's values. Spotify's engineering teams have fewer interdependencies than at most organizations because of its modular products and technology architecture. So would-be copiers with product lines requiring close coordination of interdependencies often end up with tribe structures that create chaos. Spotify people adamantly warn that its engineering model is constantly evolving and should not be copied by other companies or even by other areas within Spotify. Still, the copying continues.

A second problem with copying org charts: Too often, companies unintentionally destroy accountabilities in their business units. They create shiny new silos of agile teams that are every bit as challenging to integrate as functional silos ever were. General managers who once felt like CEOs of their units suddenly find themselves without the authority to make tough trade-offs. The financial performance of one company's credit card business, for example, deteriorated significantly when key revenue and cost levers were distributed among many different tribes beyond the influence of the business unit leader. Agile teams have to support properly defined business units—units that are accountable for meaningful P&Ls. They can't bypass or compromise those units without jeopardizing accountabilities.

Third, matrix management brings unexpected complexities. Agile teams are cross-functional teams. Cross-functional teams, by definition, require matrix organizations. Matrices may look easy on paper, but we often find ourselves cleaning up companies that launched hundreds of agile teams and never anticipated the inevitable turf battles. Who owns the teams? Who can launch additional teams? Should there be separate organization units for technology-based agile teams (sometimes called product teams) and all other innovation teams? Who funds the teams, how will decision rights work, how will teams be measured and rewarded—on and on. These details are invisible on organization charts. They are easy to overlook and impossible to copy from others.

But the worst problem is that copycats have not learned the keystone of agile success: the ability to continuously learn, evolve, improve, and grow. In trying to shortcut the process, they fail to develop skills for adapting, customizing, and harmonizing all the elements of an operating system. Agile transitions are never-ending journeys, not copy-and-paste projects. People need time to create—and get accustomed to—a new operating model. Predicting exactly how any given change will affect the organization is hard, so testing, learning, and step-by-step scaling are essential.

Agile methods, like all other management tools, have strengths and weaknesses. They do not eliminate problems. When used properly, in appropriate situations, they trade potentially disastrous problems for preferable problems. Small, autonomous agile teams are happier, faster, and more successful, but they also require more coordination and more frequent planning and funding cycles. Agile teams eliminate layers of hierarchy, but fewer layers mean fewer title changes and less frequent promotions. Failure to anticipate and address such challenges will confuse and disappoint team members. The best approach is not to choose agile over all other management approaches but to learn when, where, and how to use it in combination with other tools. This is consistent with what Aristotle, more than 2,300 years ago, called "finding the golden mean." It is also a practical path to achieving what others have called ambidextrous organizations using contingency theory and Theory Y philosophies.

Doing Agile Right: A Road Map of This Book

Back in 2001, after software developers had practiced so-called lightweight development methods for about a decade, a group of seventeen practitioners gathered to share what they had learned about better methods. They renamed the lightweight approaches *agile* and created a simple set of principles to define the process. Their Manifesto for Agile Software Development helped hundreds of thousands of individual software teams adopt and apply agile practices. Today, as companies have wrestled with agile at scale for about a decade, we're in a similar situation: we now have enough experience to analyze new patterns of success and failure. So we need to weed out the noxious misunderstandings and misuses of scaling agile before bad agile drives out good—before this powerful philosophy joins business process reengineering and quality circles on the scrap heap of management manias. It’s time to bring more sanity, practicality, and balance to the agile movement. That’s the purpose of this book. We want agile to become a valuable and practical tool rather than one more frustrating fad. We believe that agile mindsets and methods can make people in an organization far happier and more successful. We want readers to look back on their agile transitions in five to ten years with a sense of pride and fulfillment rather than disappointment and remorse.

Who will benefit from the book? We have several kinds of readers in mind. We want to help senior executives of large companies—especially large companies mired in bureaucracy—close the chasm between their bureaucratic malaise and their agile aspirations. We want to help those who are just beginning their agile journey to avoid the mistakes we have just described, and we want to help them develop agile attitudes and habitual behaviors that will create sustainable results rather than chaos. If a company has already begun its agile journey on the wrong foot, we hope to help it recognize and escape the pitfalls before it’s too late. Of course, we suspect that agile team members—and other employees who collaborate with agile teams—will also use this book to improve their performance (and maybe share it with some antiagile bosses). And we expect that start-ups already steeped in agile practices will use it to build balanced agile enterprises as they scale their success. In all these cases, our purpose is to help people build agile habits that will improve their results and increase their happiness.

For all these readers, we have tried to write a compact guidebook that time-starved businesspeople can actually read. We designed each chapter as a logical and sustainable step in the journey to an agile enterprise.

Chapter 1: How Agile Really Works. Few executives have ever