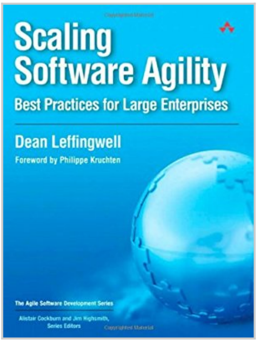

# Scaling Software Agility: Best Practices for Large Enterprises

*By Dean Leffingwell*

**Scaling Software Agility: Best Practices for Large Enterprises** By Dean Leffingwell

> “Companies have been implementing large agile projects for a number of years, but the ‘stigma’ of ‘agile only works for small projects’ continues to be a frequent barrier for newcomers and a rallying cry for agile critics. What has been missing from the agile literature is a solid, practical book on the specifics of developing large projects in an agile way. Dean Leffingwell’s book ***Scaling Software Agility*** fills this gap admirably. It offers a practical guide to large project issues such as architecture, requirements development, multi-level release planning, and team organization. Leffingwell’s book is a necessary guide for large projects and large organizations making the transition to agile development.”
> *–Jim Highsmith, director, Agile Practice, Cutter Consortium, author of Agile Project Management*
>
> “There’s tension between building software fast and delivering software that lasts, between being ultra-responsive to changes in the market and maintaining a degree of stability. In his latest work, ***Scaling Software Agility,*** Dean Leffingwell shows how to achieve a pragmatic balance among these forces. Leffingwell’s observations of the problem, his advice on the solution, and his description of the resulting best practices come from experience: he’s been there, done that, and has seen what’s worked.”
> *–Grady Booch, IBM Fellow*

Agile development practices, while still controversial in some circles, offer undeniable benefits: faster time to market, better responsiveness to changing customer requirements, and higher quality. However, agile practices have been defined and recommended primarily to small teams. In ***Scaling Software Agility,*** Dean Leffingwell describes how agile methods can be applied to enterprise-class development.

- Part I provides an overview of the most common and effective agile methods.
- Part II describes seven best practices of agility that natively scale to the

enterprise level.

- Part III describes an additional set of seven organizational capabilities that companies can master to achieve the full benefits of software agility on an enterprise scale.

This book is invaluable to software developers, testers and QA personnel, managers and team leads, as well as to executives of software organizations whose objective is to increase the quality and productivity of the software development process but who are faced with all the challenges of developing software on an enterprise scale.

# Scaling Software Agility: Best Practices for Large Enterprises

*By Dean Leffingwell*

**Scaling Software Agility: Best Practices for Large Enterprises** By Dean Leffingwell

> “Companies have been implementing large agile projects for a number of years, but the ‘stigma’ of ‘agile only works for small projects’ continues to be a frequent barrier for newcomers and a rallying cry for agile critics. What has been missing from the agile literature is a solid, practical book on the specifics of developing large projects in an agile way. Dean Leffingwell’s book ***Scaling Software Agility*** fills this gap admirably. It offers a practical guide to large project issues such as architecture, requirements development, multi-level release planning, and team organization. Leffingwell’s book is a necessary guide for large projects and large organizations making the transition to agile development.”
> *–Jim Highsmith, director, Agile Practice, Cutter Consortium, author of Agile Project Management*
>
> “There’s tension between building software fast and delivering software that lasts, between being ultra-responsive to changes in the market and maintaining a degree of stability. In his latest work, ***Scaling Software Agility,*** Dean Leffingwell shows how to achieve a pragmatic balance among these forces. Leffingwell’s observations of the problem, his advice on the solution, and his description of the resulting best practices come from experience: he’s been there, done that, and has seen what’s worked.”
> *–Grady Booch, IBM Fellow*

Agile development practices, while still controversial in some circles, offer undeniable benefits: faster time to market, better responsiveness to changing customer requirements, and higher quality. However, agile practices have been defined and recommended primarily to small teams. In ***Scaling Software Agility,*** Dean Leffingwell describes how agile methods can be applied to enterprise-class development.

- Part I provides an overview of the most common and effective agile methods.
- Part II describes seven best practices of agility that natively scale to the enterprise level.
- Part III describes an additional set of seven organizational capabilities that companies can master to achieve the full benefits of software agility on an enterprise scale.

This book is invaluable to software developers, testers and QA personnel, managers and team leads, as well as to executives of software organizations whose objective is to increase the quality and productivity of the software development process but who are faced with all the challenges of developing software on an enterprise scale.

**Scaling Software Agility: Best Practices for Large Enterprises By Dean Leffingwell Bibliography**

- Sales Rank: #748705 in Books
- Published on: 2007-03-08
- Original language: English
- Number of items: 1
- Dimensions: 9.10" h x 1.00" w x 7.30" l, 1.43 pounds
- Binding: Paperback
- 384 pages

## Editorial Review

From the Back Cover

"Companies have been implementing large agile projects for a number of years, but the 'stigma' of 'agile only works for small projects' continues to be a frequent barrier for newcomers and a rallying cry for agile critics. What has been missing from the agile literature is a solid, practical book on the specifics of developing large projects in an agile way. Dean Leffingwell's book "Scaling Software Agility" fills this gap admirably. It offers a practical guide to large project issues such as architecture, requirements development, multi-level release planning, and team organization. Leffingwell's book is a necessary guide for large projects and large organizations making the transition to agile development."

"-Jim Highsmith, director, Agile Practice, Cutter Consortium, author of Agile Project Management""There's tension between building software fast and delivering software that lasts, between being ultra-responsive to changes in the market and maintaining a degree of stability. In his latest work, "Scaling Software Agility, " Dean Leffingwell shows how to achieve a pragmatic balance among these forces. Leffingwell's observations of the problem, his advice on the solution, and his description of the resulting best practices come from experience: he's been there, done that, and has seen what's worked."

"-Grady Booch, IBM Fellow"Agile development practices, while still controversial in some circles, offer undeniable benefits: faster time to market, better responsiveness to changing customer requirements, and higher quality. However, agile practices have been defined and recommended primarily to small teams. In "Scaling Software Agility, " Dean Leffingwell describes how agile methods can be applied to enterprise-class development.Part I provides an overview of the most common and effective agile methods.Part II describes seven best practices of agility that natively scale to the enterprise level.Part III describes an additional set of seven organizational capabilities that companies can master to achieve the full benefits of software agility on an enterprise scale.This book is invaluable to software developers, testers and QA personnel, managers and team leads, as well as to executives of software organizations whose objective is to increase the quality and productivity of the software development process but who are faced with all the challenges of developing software on an enterprise scale.

About the Author

**Dean Leffingwell** is a renowned software development methodologist, author, and software team coach who has spent his career helping software teams meet their goals. He is the former founder and CEO of Requisite, Inc., makers of RequisitePro, and a former vice president at Rational Software, where he was responsible for the commercialization of RUP. During the last five years, in his role as both an independent consultant and as advisor/methodologist to Rally Software, Mr. Leffingwell has applied his experience to the organizational challenge of implementing agile methods at scale with entrepreneurial teams as well as distributed, multinational corporations. These experiences form much of the basis for this book. Mr. Leffingwell is also the lead author of *Managing Software Requirements, Second Edition: A Use Case Approach* (Addison-Wesley, 2003).

## Software Methodology Career Phase I: Experiences at RELA and Requisite, Inc.

I have been committed to improving software engineering and software development management practices throughout my career. While that goal has been a constant, I now recognize three distinct stages of my understanding of software development practice and maturity. In Phase I, I was the CEO of RELA, Inc., where we developed software for others on a contract basis. RELA developed a wide variety of software applications ranging from stomach-churning adventure park rides to life-sustaining medical devices. Because the software we wrote was always for others, we were constantly aware of the need to “build the right thing.” Our individual livelihoods, our company, and its stakeholders all depended on our ability to understand what problems our solutions needed to address and how to apply effective best practices in achieving that solution.

In order to do so, we depended on many of the rigorous practices based on the waterfall method in use at that time. Indeed, some of our customers, and other key regulatory agencies such as the FDA, mandated its use, so we followed it prescriptively even as we tried to improve it. While it is entertaining now for many of us to criticize and make fun of the methods we employed, *the fact is that the waterfall method was a substantial improvement over the cut-and-try methods of the past, and more importantly, it delivered results.* Much of my focus at that time was on the requirements process, because that is where the critical discovery happened, solution behavior was defined, and our contracts were based.

That experience led me to my next career as founder and CEO of Requisite, Inc., makers of RequisitePro, a product solution for requirements management. At Requisite, we advanced and developed requirements practices and products, so in a sense we became experts in the *front end* of the lifecycle. We sold Requisite to Rational in 1997, and I embarked on Phase II of my software development process career.

## Software Methodology Career Phase II: Experiences at Rational Software

In this phase, I was a senior executive at Rational Software and was involved in the promulgation of the Unified Modeling Language (UML) and the Rational Unified Process (RUP). At Rational, I had the good fortune of working directly with thought leaders such as Grady Booch, Ivar Jacobson, James Rumbaugh, Walker Royce, and Philippe Kruchten. During this time, my coauthor, Don Widrig, and I also published the first edition of the Addison-Wesley text *Managing Software Requirements* (2000).

Then our thinking was based on object orientation, and that technique provided additional flexibility in our development methods and additional resiliency in the software we wrote. It also led to a software process that was fundamentally different from the waterfall method and one that is characterized as being iterative and incremental. In this method, each iteration was a working piece of code that could be objectively assessed and evaluated. This method was far more agile than my prior experiences: we were no longer forced to rely solely on intermediate work products—documents, design reviews, and the like—but could see and measure tangible progress.

Rational codified that process in a written process description, the Rational Unified Process, and marketed and applied that process with good success across the industry. In addition, we applied the process in the development and release of Rational Suites, which required the coordination of as many as 800 team members in four countries. We released Rational Suites twice per year, each with an integrated set of products and a common install. Rational was eventually purchased by IBM, and today RUP is marketed under the auspices of IBM’s Rational Software Division and is in use by hundreds of thousands of practitioners.

## Software Methodology Career Phase III: Experiences with Agile and Rally

Upon leaving Rational, I became an independent consultant and adviser to development-stage software businesses, where I coached business strategy and software development practices to a half dozen new ventures. I used the opportunity to leverage some of the more innovative, lightweight methods, including XP and Scrum, and witnessed firsthand the improvements in productivity and quality that these methods delivered to these smaller teams.

After a short time, I was so won over by these methods that I soon refused to engage with any business or any team that did not have a strong sense of agility. The risk to the business was too great otherwise! At the same time, I began to see the limitation of those methods. As the teams and applications grew, the team’s ability to refactor code became less practical, and we also noticed the need for more assured communication of requirements as well as for more “architectural runway.” During this time, I was also consulting methodologist to Rally Software and helped to develop its hosted solution for distributed agile development. At Rally, I was heavily influenced by our interaction with agile thought leaders such as Ryan Martens, Ken Schwaber, Jim Highsmith, Mike Cohn, Tom and Mary Poppendieck, and Jeff Sutherland.

## Experiences with Agile at Enterprise Scale

Concurrently, I was personally challenged by a number of larger organizations to bring the level of agility and responsiveness I had witnessed to their enterprise. It was with some trepidation that I accepted this assignment and spent the next few years applying the core principles of agility in larger organizations, including applying my experience to large-scale development at BMC Software, Inc., where we worked with hundreds of highly distributed developers to deliver new applications of substantial scope and scale.

In so doing, I was pleased to discover that many of the best practices as taught by the agile methods delivered immediate out-of-the-box value to the enterprise. I also discovered that, by themselves, these best practices did not fully address the challenge at enterprise scale. Therefore, we gradually evolved an extended set of practices that were necessary to achieve enhanced agile results at scale. Finding little published in the marketplace to counsel larger companies, I resolved to write this book. *I do so in the hope that your enterprise can leverage what we've learned and apply it to deliver higher productivity and higher quality to your customers.* In a world dominated by software, it is hard to imagine a higher leverage point for our industry and, indeed, our economy as a whole.

## How to Read This Book

### Part I: Overview of Software Agility

This book is divided into three parts. Part I provides a short history of the agile movement with a discussion of some of the primary agile methods that are in use today, including XP and Scrum, as well as a discussion of RUP, which is an iterative and incremental method that can be applied in an agile fashion. In addition, we take a brief look at a number of other methods that helped sponsor the agile movement, including Lean Software Development, Dynamic System Development Method (DSDM), and Feature-Driven Development (FDD). We look at these methods not to teach the methods themselves but to provide a basis for understanding Parts II and III. As we will discover, each method has brought substantially new thinking to software development practices, and each has contributed substantially to the state-of-the-art. In addition, we'll start to see a set of agile best practices emerge, many of which have already been applied at significant scale, and we will use these as the basic building blocks of enterprise agility.

### Part II: Seven Agile Team Practices That Scale

Part II describes these building blocks, the *seven agile team practices that scale,* one per chapter. In a sense, these practices may be considered the essence of agile in that *all* the agile methods apply these practices either explicitly or implicitly. For those new to agile or for those large organizations contemplating implementation of these practices, Part II of the book should provide comfort, because by simply adopting any of the agile methods described—or better, mixing and matching as necessary for the company's current context—many of these best practices will naturally emerge and provide an immediate benefit to applications of virtually any scope. While they are not trivial to address and master, they have been proven in a wide variety of project contexts, and they will benefit any team that adopts them.

Together, Parts I and II of the book provide an overview of software agility and describe seven best practices that can be applied at virtually any scale. Each of these practices can directly and immediately improve the productivity and quality outcomes for teams who choose to adopt them.

### Part III: Creating the Agile Enterprise

To achieve true enterprise agility, however, more work remains, and that is the topic of Part III. We describe an additional set of capabilities, guidelines, principles, practices, and insights that will allow the organization to apply agility at virtually any level of application or system scale. These practices have been derived from experiences in the field in applying agile in larger circumstances. They include "green field" projects of smaller teams of 40 to 50 developers distributed across multiple countries, including extensive outsourcing, as well as larger organizations of up to a thousand developers working toward a common purpose on systems that required intense coordination among those teams. Some of the principles in Part III may seem obvious at first. Others are more subtle and have been discovered by experience in applying agile at scale. Many of these principles came about as teams reflected on their prior release efforts and came to modify their behavior over time so as to continually improve results.

Taken together, our hope this is that this book will help larger organizations achieve productivity and quality gains of as much as 200 percent, as such has been achieved by the smaller teams that have applied them. In turn, these results will provide benefits of faster time to market, higher return on development investment, and increased customer satisfaction for the enterprise. And lest we forget, organizations that head down this path have an additional intangible benefit: the teams themselves love agile methods, and empowering them to experiment and advance their methods is a key to engaging them in a virtuous cycle of empowerment, continuous process improvement, improved project outcomes, personal and professional growth, and higher job satisfaction. *In an industry that faces the challenge of encoding much of the world's intellectual property, what could be more virtuous than that!*

## Users Review

**From reader reviews:**

**Vance Malik:**

As people who live in the actual modest era should be change about what going on or info even knowledge to make these individuals keep up with the era and that is always change and move forward. Some of you maybe can update themselves by reading through books. It is a good choice for you but the problems coming to you actually is you don't know what type you should start with. This Scaling Software Agility: Best Practices for Large Enterprises is our recommendation to make you keep up with the world. Why, since this book serves what you want and wish in this era.

**Martha McKee:**

Hey guys, do you wants to finds a new book to study? May be the book with the concept Scaling Software Agility: Best Practices for Large Enterprises suitable to you? The particular book was written by well-known writer in this era. Typically the book untitled Scaling Software Agility: Best Practices for Large Enterprisesis one of several books in which everyone read now. This particular book was inspired a lot of people in the world. When you read this guide you will enter the new dimensions that you ever know prior to. The author explained their plan in the simple way, therefore all of people can easily to comprehend the core of this publication. This book will give you a large amount of information about this world now. So that you can see the represented of the world on this book.

**Loyd Tyler:**

Typically the book Scaling Software Agility: Best Practices for Large Enterprises will bring one to the new experience of reading some sort of book. The author style to describe the idea is very unique. Should you try to find new book to see, this book very ideal to you. The book Scaling Software Agility: Best Practices for Large Enterprises is much recommended to you to read. You can also get the e-book from the official web site, so you can more readily to read the book.

**Richard Eby:**

Do you have something that you like such as book? The book lovers usually prefer to opt for book like comic, short story and the biggest one is novel. Now, why not seeking Scaling Software Agility: Best Practices for Large Enterprises that give your pleasure preference will be satisfied by reading this book. Reading routine all over the world can be said as the means for people to know world better then how they react to the world. It can't be stated constantly that reading habit only for the geeky man but for all of you who wants to always be success person. So , for all you who want to start looking at as your good habit, you are able to pick Scaling Software Agility: Best Practices for Large Enterprises become your starter.

## Download and Read Online Scaling Software Agility: Best Practices for Large Enterprises By Dean Leffingwell #IRHUAXV3LOG

# Read Scaling Software Agility: Best Practices for Large Enterprises By Dean Leffingwell for online ebook

Scaling Software Agility: Best Practices for Large Enterprises By Dean Leffingwell Free PDF d0wnl0ad, audio books, books to read, good books to read, cheap books, good books, online books, books online, book reviews epub, read books online, books to read online, online library, greatbooks to read, PDF best books to read, top books to read Scaling Software Agility: Best Practices for Large Enterprises By Dean Leffingwell books to read online.

## Online Scaling Software Agility: Best Practices for Large Enterprises By Dean Leffingwell ebook PDF download

### Scaling Software Agility: Best Practices for Large Enterprises By Dean Leffingwell Doc

#### Scaling Software Agility: Best Practices for Large Enterprises By Dean Leffingwell Mobipocket

#### Scaling Software Agility: Best Practices for Large Enterprises By Dean Leffingwell EPub

#### IRHUAXV3LOG: Scaling Software Agility: Best Practices for Large Enterprises By Dean Leffingwell