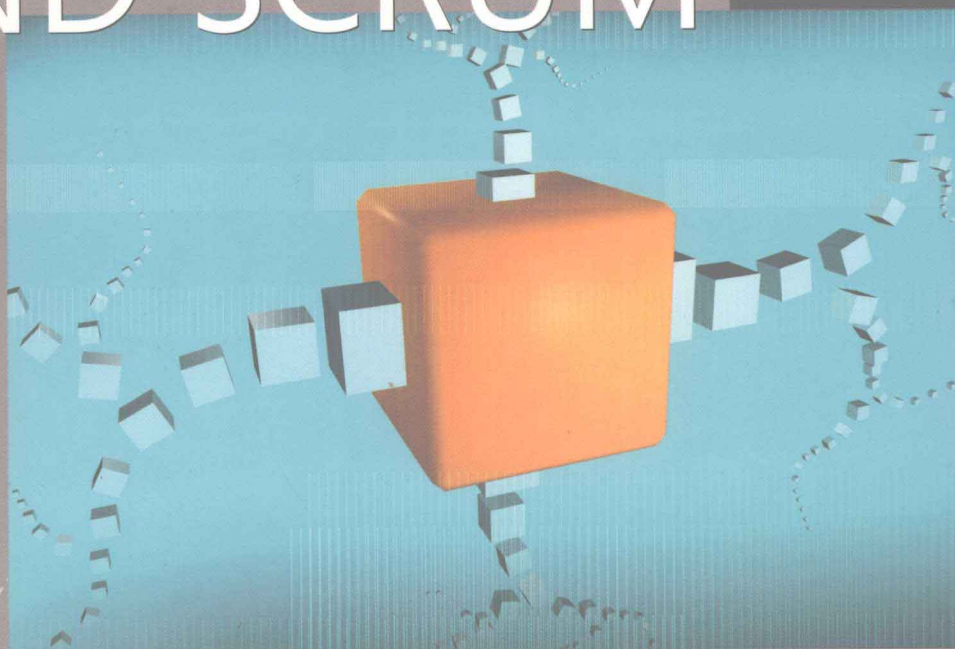


Microsoft

BEST PRACTICES

THE ENTERPRISE AND SCRUM



SCRUM与企业管理

China Edition
Published by Shanghai World Publishing Corporation

Ken Schwaber

Author of *Agile Project Management with Scrum*

世界图书出版公司

The Enterprise and Scrum

Ken Schwaber

图书在版编目(CIP)数据

Scrum 与企业管理:英文/(美)史威伯著. - 上海:上海世界图书出版公司,2008. 10

ISBN 978 - 7 - 5062 - 9291 - 7

I. S… II. 史… III. 企业管理:项目管理 - 英文 IV. F270

中国版本图书馆 CIP 数据核字(2008)第 159891 号

© 2007 by Microsoft Corporation. All rights reserved.

Original English language edition © 2007 by Ken Schwaber. All rights reserved. Published by arrangement with the original publisher, Microsoft Corporation, Redmond, Washington, U. S. A.

Scrum 与企业管理

[美]肯·史威伯 著

上海世界图书出版公司 出版发行

上海市尚文路 185 号 B 楼

邮政编码 200010

(公司电话:021 - 63783016 转发行部)

上海竟成印务有限公司印刷

如发现印装质量问题,请与印刷厂联系

(质检科电话:021 - 56422678)

各地新华书店经销

开本:787 × 960 1/16 印张:11 字数:350 000

2008 年 10 月第 1 版 2008 年 10 月第 1 次印刷

ISBN 978 - 7 - 5062 - 9291 - 7/T · 178

图字:09 - 2008 - 627 号

定价:188.00 元

<http://www.wpcsh.com.cn>

<http://www.mspress.com.cn>

Introduction

This book is for those who want to use Scrum throughout their enterprise for product development. Right now, you might have pockets within your enterprise that use Scrum, and they are more effective than elsewhere. You are at least partially convinced that using Scrum throughout the enterprise might be a way to make the whole enterprise more effective, but you could use some help in figuring out how to do so. This book is for you.

There are many reasons why your enterprise can't develop and deploy products and systems as rapidly, inexpensively, and with the quality that you would like. You and your staff probably can already list many of them. Scrum won't solve them. Scrum is simply a tool that will relentlessly and ruthlessly expose them. As you try to build product within the Scrum framework, every time one of these impediments is reached, it will be exposed in a somewhat painful way. You can then prioritize it and systematically eliminate it. When the impediments are mostly gone, Scrum is a framework that will enable the product development you desire. And it will continue to be your watchdog against any new impediment or old impediments returning home for a visit.

I've gathered quite a few experiences and stories as I've worked with enterprises adopting Scrum. In this book, I've organized them into guidance in the areas that are most problematic. Sometimes this is descriptive; other times I relate the guidance through stories. It is OK that there is no guidance in the other areas. The enterprise should figure out what is likely to work best for itself and try to use it. To the extent that an approach doesn't work, change it and change it again so that it works better and continues to work better.

Scrum does not prescribe. Scrum includes general guidelines about how to do development and principles to be applied when these recommendations are insufficient. What does this mean? This means that people have to learn to think differently. We want rules to follow, but life and product development are too complex for a single set of rules to suffice in all circumstances. You have to rely on decentralized decision-making, because there probably isn't one answer for every team any more than there is for every enterprise.

The first three chapters lay out the plan for adopting Scrum. The next two chapters provide insights into some habits that impede adoption and how some enterprises have coped with them. The remaining chapters provide techniques for solving some of the knottier issues. These will help you, but your enterprise's adoption will be different from anyone else's adoption. The only common ingredient is people, for better and worse. When people rise to the occasion and work heroically in teams, nothing is better. When they prefer to lay back, play politics, and undercut each other, nothing is worse. You'll get to see both, because Scrum will relentlessly expose everything as you proceed.

Not every enterprise that tries to adopt Scrum will succeed. At times, you and your people will hate Scrum. However, don't shoot it. It is only the messenger. To the extent that you and your enterprise succeed, though, you will always know where you stand. You will know what you can do and can't do. Sometimes such transparency let's us see things that aren't what we wish to see. However, I find knowledge preferable to uncertainty and ignorance. The goal is for you and everyone in your enterprise to wake up looking forward to coming to work, and for your competitors to wish they had never woken up.

Contents at a Glance

	Introduction	xi
Part I	Adopting Scrum	
1	What Do We Have to Do to Adopt Scrum?	3
2	Scrum qua Scrum	9
3	The First Year	13
4	Against Muscle Memory—The Friction of Change	21
5	Enterprises in Transition	29
Part II	Start Using Scrum for Enterprise Work	
6	Organizational Practices	45
7	Engineering Practices	59
8	People Practices	69
9	The Relationship Between Product Management/Customer and the Development Team	85
Part III	Appendices	
A	Scrum 1, 2, 3	101
B	More About Scrum	113
C	Example Scrum Kickoff Meeting Agenda	119
D	Initial Enterprise Transition Product Backlog	123
E	Scrum Musings	127

Table of Contents

Introduction xi

Part I **Adopting Scrum**

1 **What Do We Have to Do to Adopt Scrum?** 3

 Scrum Requires a New Enterprise Culture 4

 Prove to Yourself That It Is Worth the Effort 5

 Assess the Type of Change That Will Occur 5

 Caveats 7

2 **Scrum qua Scrum** 9

 Scrum Kickoff Meeting 11

3 **The First Year** 13

 The First Month 13

 The Second Month 15

 Sources of Transition Backlog Impediments 16

 What If? 17

 The Third Month and Beyond 18

4 **Against Muscle Memory—The Friction of Change** 21

 Waterfall Thinking 21

 Command and Control 23

 Commitment to Defying the Laws of Nature 24

 Hiding Reality 26

 Summary 27

 **What do you think of this book? We want to hear from you!**

Microsoft is interested in hearing your feedback so we can continually improve our books and learning resources for you. To participate in a brief online survey, please visit:

www.microsoft.com/learning/booksurvey/

5	Enterprises in Transition	29
	Contoso	29
	Situation	30
	Application of Scrum	30
	Outcome	31
	Additional Comments	31
	Humongous	32
	Situation	32
	Application of Scrum, Phase 1	33
	Outcome, Phase 1	33
	Situation, Phase 2	34
	Application of Scrum, Phase 2	34
	Outcome, Phase 2	34
	Additional Comments	35
	Woodgrove Bank	35
	Application of Scrum	36
	Litware	37
	Situation	37
	Application of Scrum	37
	Outcome	38
	Additional Comments	40

Part II **Start Using Scrum for Enterprise Work**

6	Organizational Practices	45
	#1: Organizing Enterprise Work	46
	#2: Organizing Enterprise Work for a High-Technology Product Company	46
	#3: Organizing Enterprise Work in Other Enterprises	51
	#4: Organizing Enterprise Work for New Systems that Automate an Enterprise Operation	52
	#5: Organizing the Complexity of Multiple Views	54
	#6: Organizing Work to Optimize Software Product Family Architectures	55
7	Engineering Practices	59
	#1: Multilayer System Work Organized by Functionality	60
	#2: Integration of Multiple-Layer Systems	63
	#3: Integrating the Work of Scrum Teams and Teams Not Using Scrum	66
	Summary	68

8	People Practices	69
	#1: Organizing People to Do Enterprise Work	70
	#2: Team Creation	73
	#3: Team Work	75
	#4: How People Are Managed	76
	#5: Functional Expertise	80
	#6: Compensation	81
	#7: Extra Managers	81
	#8: Teams with Distributed Members	82
	#9: Scarce Skills Needed by Many Teams	83
9	The Relationship Between Product Management/Customer and the Development Team	85
	#1: Shortening the Time to Release Through Managing Value	86
	Relative Valuation with Scrum	87
	#2: Just Do It	90
	#3: The Infrastructure, or Core	90
	#4: Accelerators to Recovery	92
	#5: The Mother of All Problems	93
Part III	Appendices	
A	Scrum 1, 2, 3	101
	The Science	101
	Empirical Process Control	102
	Complex Software Development	103
	Scrum: Skeleton and Heart	105
	Scrum: Roles	106
	Scrum: Flow	106
	Scrum: Artifacts	109
	Product Backlog	109
	Sprint Backlog	111
	Increment of Potentially Shippable Product Functionality	112
B	More About Scrum	113
	Scrum Terminology	113
	Scrum and Agile Books	117
	Scrum Books	117
	Books on Techniques Used in Scrum for Managing Product Development	117

	Books on Managing in an Agile Enterprise	117
	Books on Related Theory	118
	Books that Provide Insights into Agile	118
	Books on Agile Software Engineering Techniques	118
	Scrum and Agile Web Sites	118
C	Example Scrum Kickoff Meeting Agenda	119
	Conduct Kickoff Meeting	119
D	Initial Enterprise Transition Product Backlog	123
	Establish Preconditions a Project Must Meet to Use Scrum	123
	Establish New Metrics	124
	Suboptimal Metrics	124
	Change Project Reporting	124
	Establish a Scrum Center	125
E	Scrum Musings	127
	Value-Driven Development	127
	Realizing Project Benefits Early	129
	Eat Only When Hungry	130
	For Customers Only	131
	Bidding Work	133
	Managing Work	134
	A Cost-Effective Alternative to Offshore Development	136
	How to Use Scrum and Offshore Development	138
	Too Large Teams	139
	Virtual Teams Instead of Offshore Development	140
	Forming Cross-Functional Teams	142
	Cross-Functional Teams and Waterfall	143
	Index	147



What do you think of this book? We want to hear from you!

Microsoft is interested in hearing your feedback so we can continually improve our books and learning resources for you. To participate in a brief online survey, please visit:

www.microsoft.com/learning/booksurvey/

Part I

Adopting Scrum

This first section describes how an enterprise can adopt Scrum. Learning to use Scrum would be pretty simple and straightforward if we didn't have habits to do things differently. Fitting it into our enterprises, also, would be pretty straightforward if we already weren't organized and acculturated to do things differently.

Changing enterprise habits and culture is required to get the benefits of Scrum. In this section, we assess whether those benefits are of enough value for you to go through the effort. Then we look at how to initiate an enterprise transition project. This project uses Scrum to optimize your enterprise's ability to build and deploy products. We then look at some of the changes that an enterprise encounters to get the benefits.

The chapters in this section are briefly described in the following list:

- Chapter 1, "What Do We Have to Do to Adopt Scrum?" describes how to assess whether Scrum has enough value to your enterprise for you to proceed.
- Chapter 2, "Scrum qua Scrum," describes steps to initiate Scrum within your enterprise.
- Chapter 3, "The First Year," describes the first year of adopting Scrum.
- Chapter 4, "Against Muscle Memory—The Friction of Change," describes some of the most entrenched habits that impede productivity.
- Chapter 5, "Enterprises in Transition," describes some adoption projects at several enterprises. Read these in anticipation of and preparation for your enterprise's transition, for which guidance is provided in Section 2.

What Do We Have to Do to Adopt Scrum?

In this chapter:	
Scrum Requires a New Enterprise Culture	4
Prove to Yourself That It Is Worth the Effort.	5
Assess the Type of Change That Will Occur.	5
Caveats	7

Consider Scrum as part of the game of product and software development. Scrum lays out the playing field and rules for the game. Your enterprise has the players for the game. They go on the field and start playing against the competition. If they are skilled, it shows. If they don't yet work as a team, don't understand the rules, or have any other flaw in their capabilities, it is painfully obvious. Everyone on the team knows what improvements are needed—more coaching, more training, better teamwork.

When Scrum is used throughout an enterprise, we have an enterprise-wide game of product development. Coordination is more important than it would be if just a single team was playing, and it's harder to achieve. (Keep in mind that a single department could have 100 teams.) Again, though, Scrum helps everyone understand what needs to be improved. Every time product development occurs, Scrum rewards excellence and exposes inadequacies.

Scrum adoption has two aspects. First, Scrum is rolled out. You teach everyone how to play the game of product development using Scrum. You teach them how to work together in small teams. This stage takes six to twelve months. The second aspect is everyone in the enterprise improving their game so that they are the best possible enterprise of teams working together. During this time, we improve skills, teamwork, and everything needed for excellence. Every time we play Scrum, we can clearly see how good we've become and what we need to do to get better. To get really, really good requires three to five years of continued improvement through using Scrum in an enterprise. Staying really good and perfecting skills is an ongoing endeavor.

Your use of Scrum will expose every reason why your enterprise has trouble building products. Scrum will keep exposing the problems until they are fixed. Scrum does this within the simple framework of building increments of software, iteration by iteration, or Sprint by Sprint. The rules, roles, and time-boxes of Scrum are few and simple. Whenever they cause a

conflict with existing practices, an impediment has been encountered and made visible. The enterprise has to choose whether to change to remove the impediment or to give up on some of the benefits.

Scrum Requires a New Enterprise Culture

The Scrum paradigm embraces change, unpredictability, and complexity as inescapable constants in all product development. This complexity and unpredictability renders detailed long-term predictive plans meaningless and a waste of money. With Scrum, a vision of a project's value is projected in a baseline plan. The project moves forward, Sprint by Sprint, toward the vision. Increments are inspected every Sprint. Adaptations are then made to the project to optimize the likelihood of realizing the value.

Adventure Works, a game producer in San Diego, was the first in its industry to benefit from Scrum. Joris Kalz, Adventure Works' CTO, attended one of the very first Scrum certification sessions in 2003. Enthusiastically, he went back to Adventure Works and adopted the Scrum paradigm. His story is one of insight, persistence, and hard work. The Adventure Works story is one of culture shock and then redemption.

The product that was developed using Scrum was Vosod. It began to emerge in high-quality, regular increments. Joris adopted a sustainable pace of work, one of Scrum's practices. Everyone worked eight-hour days. Some people might look at that practice and think, "Oh, that means developers get out of working hard for the company!" Quite the contrary—a sustainable pace yields higher productivity and quality products.

Adventure Works was owned by a Japanese company. The Scrum practice of eight-hour workdays was unacceptable to the senior members of the Japanese management. They demanded longer hours, and the 12-hour work days that were normal prior to Scrum were restored. Defects rose 60 percent over the next several Sprints, more than offsetting the delivery of increased functionality. Joris restored Scrum's eight-hour workdays. When the Japanese managers in San Diego drove by the offices night after night, they again saw empty parking lots and darkened offices. This was intolerable to them. They reported to headquarters that employees at Adventure Works were indifferent and lazy. They recommended selling the company. The delivery of increments of high-quality software was good, but that was insignificant compared to the perceived sloth and cultural conflict.

The Japanese parent company sold Adventure Works to its American management in a management buyout. The parent company was glad to get rid of it. Two months later, Vosod was completed and ready to ship. Adventure Works sold Vosod to a game publisher for twice the price of the buyout. Did it make sense for the Japanese owners to sell the company when they did? Of course not, but the twisting paths of change often don't make sense. People and culture are involved—people who have feelings, beliefs, perceptions, and vested interests that cloud their perceptions.

Prove to Yourself That It Is Worth the Effort

The effort required to adopt Scrum is huge, and only enterprises with compelling reasons will make the effort. Your reason for adopting it might be unacceptable costs, missing functionality, inability to deliver software, customers going to other providers, developers leaving, lengthening release cycles, or your enterprise's increasing inability to compete. Another compelling reason is Scrum offers a significantly better way of building products.

Before you attempt an enterprise-wide adoption, you must believe that your enterprise has serious problems to fix and that Scrum is the tool to help you. The first step in gaining this belief is to use Scrum on several projects. Scrum is simple enough to understand from books (some of which are listed in Appendix B), but some initial ScrumMaster and Scrum training might be helpful. (Scrum terminology is fully defined in Appendix B.) Such training is available through www.scrumalliance.org. Select some high-value, high-risk initial work. Conduct a combined iteration planning meeting (called a *Sprint Planning Meeting*) and training session. Then start Sprinting. Conduct at least three Sprints. You will see value. You will clearly know the progress of a project and be able to easily accommodate changes. In addition, you will see increased productivity.

You have now seen Scrum's value on some simple projects. Now go for the jugular. Select another project—one that is difficult or one that the enterprise is having problems with. Prove to yourself that Scrum solves some of your most knotty problems. Identify several pieces of important functionality, which is enough to get going. This is the basis of the Product Backlog. Form a Scrum team and have them Sprint several times. When they've done that, the functionality should have the desired security characteristics, performance capabilities, and user experience as the finished product. Extrapolate the cost of the functionality in the third Sprint to get an estimate for the entire project. You have to wait until the third Sprint for people on the team to know each other and the system they are developing well enough to get a meaningful extrapolation.

If you are concerned whether a commercially available package works as claimed, subject it to the same process. Have Scrum teams build several pieces of high-value, tricky functionality in the package. Get early information on whether the package works as you need it to work.

Formally train people in Scrum for these projects. Courses are offered by the Scrum Alliance (www.scrumalliance.org) that will help them gain the needed skills. Just like in baseball, a little coaching helps a novice rapidly gain skills and technique.

Assess the Type of Change That Will Occur

You should now be convinced that Scrum can help your enterprise reach its goals. Before you proceed with adopting Scrum, however, you should consider the types of changes that other enterprises have gone through. These changes have repeatedly been more extensive than

other enterprises anticipated because everyday practices are exposed as impediments. You can expect the following changes and challenges:

Staff turnover will occur. Twenty-percent turnover is common. Some people say, “I don’t like this. I just want to come to work, be told what to do, and go home at the end of the day not worrying about it.” We’ve changed the ground rules with Scrum. People are asked to commit to solving problems in teams. Some people might not want this type of work.

The third through ninth months of the change will be particularly difficult. Problems and dysfunctions that have always existed in your enterprise will be highlighted at this stage. They haven’t been fixed yet because they are particularly entrenched or difficult. Solutions have been hard to devise or achieve. When Scrum again highlights them, others on the project might wonder why they ever embarked on the Scrum process. At this point, look back and observe the progress that has been made. Projects are moving forward, software is being delivered, risks are being identified and removed, and people are working together. You will have the courage to continue moving forward only by looking back at the progress made.

Conflict will occur. Expect conflict. Conflict is a sign of change. People have different opinions about how things should be done. A new way of operating must be conceived. Because many enterprises discourage conflict, people might not be skilled at resolving conflict. People need to be trained to resolve conflicts.

Product management’s job will change and will be harder. Product managers and customers are now Product Owners. They are responsible for managing the projects, Sprint by Sprint, to maximize value and control risk. They are accountable to senior management for the success or failure of the project. They are the single, wringable neck. If members of senior management want to find out how a project is doing, they will call the Product Owner. They will no longer call engineering or a project manager.

Engineering is accountable for quality. The engineering organization is responsible for figuring out how to build and deploy a quality increment every Sprint. The quality will be the same as that needed in the final product. The ScrumMaster will not allow them to lower quality to increase productivity.

Compensation policies need to change. Scrum is about team heroics, not individual heroics. The majority of the enterprise’s bonus and incentive funds need to be allocated based on the team’s performance rather than the individual’s performance. If a team does really well, reward everyone on the team.

Jobs will change. Some existing jobs will disappear, and people will fulfill new roles. For instance, a project manager might become a ScrumMaster. A functional manager will no longer have a function to manage and might become a ScrumMaster or Product Owner. Career paths become far less important than contribution to the team and the enterprise.

Management's primary responsibility will shift from command to servant leadership.¹

Managers are responsible for the performance of their area of the enterprise. Their usual tactics are to direct and command. They figure out what needs to be done and tell people who work for them to do it. This hierarchically decomposes until the bottom person is actually doing the work. With Scrum, management's responsibilities remain the same, but the philosophy and techniques change. Managers will lead and serve their staffs to achieve their goals. They will remove impediments. They will guide, train, coach, mentor, and get their people to do the best they can. Their role is very much like a parent: to grow their people so that they are mature and self-managing. These attributes are best learned through study and experience, not by being told what to do.

Management turnover will occur. Management is going to be asked to go through significant changes. (See the change details in the preceding paragraph.) They will do extremely difficult work over the next several years. Some managers won't want to. Up to 20 percent of them might leave as they find that they don't like the new way of working and managing.

More people might not be the answer. When we want more work done, we often hire more people. This is well documented as an ineffective approach.² Adding people to productive teams or diluting the ranks of existing skilled people by spreading them among new teams reduces both measured productivity and quality. In my experience, Scrum's self-managing teams generate at least 50-percent productivity improvement in the first year of use, and more thereafter. Focus on implementing Scrum, not adding more people.

Caveats

You probably have tried to implement new processes before. Please remember that Scrum is less a process than a tool for you to build processes appropriate to your enterprise. Like any tool, there are right ways and wrong ways to use it. Two caveats that you should keep in mind when using Scrum are as follows:

Do not change Scrum. Scrum isn't a process that you modify to fit your enterprise. Instead, it exposes every dysfunction in your enterprise while you build products. It is your canary in a coal mine.³ Whenever people change Scrum, it's because they have run into a problem, dysfunction, or conflict that they do not want to face and fix. Instead, they change Scrum so that the problem remains invisible and remains deadly to your enterprise. If you allow this to happen, you will have just lost Scrum's primary benefit.

1 James Autry, *The Servant Leader* (Three Rivers Press, 2004)

2 Frederick Brooks, *The Mythical Man Month* (Addison Wesley, 1995)

3 Coal miners placed canaries in the mines they worked in because canaries are more sensitive to carbon monoxide than people. When a canary died, it was time to get out of the mine.