

Microsoft

BEST PRACTICES

MORE ABOUT SOFTWARE REQUIREMENTS

Thorny Issues and Practical Advice



更多
软件需求

实际问题解决方案

Karl E. Wiegiers

Two-time winner of the Software Development Productivity Award

China Edition

Published by Shanghai World Publishing Corporation

世界图书出版公司

WHAT WE'RE ABOUT SOFTWARE REQUIREMENTS

Software Requirements Engineering

软件需求工程

软件需求工程

第1章 绪论

**More About Software
Requirements:
Thorny Issues
and Practical Advice**

Karl E. Wiegers

图书在版编目(CIP)数据

更多软件需求:实际问题解决方案:英文/(美)维格尔著.
—上海:上海世界图书出版公司,2007.6
ISBN 978-7-5062-8323-6

I. 更… II. 维… III. 软件开发-英文 IV. TP311.52

中国版本图书馆 CIP 数据核字(2007)第 082300 号

© 2007 by Microsoft Corporation. All rights reserved.

Original English language edition © 2006 by Karl Wiegers. All rights reserved.
Published by arrangement with the original publisher, Microsoft Corporation, Redmond,
Washington, U. S. A.

更多软件需求:实际问题解决方案

[美]卡尔·维格尔 著

上海世界图书出版公司 重印发行

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)

上海市印刷七厂印刷

如发现印装质量问题,请与印刷厂联系

(质检科电话:021-63113060)

各地新华书店经销

开本:787×960 1/16 印张:14 字数:420 000

2007 年 6 月第 1 版 2007 年 6 月第 1 次印刷

ISBN 978-7-5062-8323-6/T·148

图字:09-2007-443 号

定价:98.00 元

<http://www.wpcsh.com>

Praise for More About Software Requirements: Thorny Issues and Practical Advice

“A must-have—Weigers goes well beyond aphorisms with practical insights for everyone involved in the requirements process. This book is an experience-based, insightful discussion of what the software requirements expert ought to know to get better at his or her job. It’s the book you should read after whatever book you read to get an introduction to the topic of software requirements.

There’s a whole lot of reality in this relatively small volume. There is a section on ‘Managing Scope Creep,’ and another on ‘The Fuzzy Line Between Requirements and Design,’ that are worth the price of the book just because of the light they shine on some particularly difficult issues. I like this book. Its author speaks with experience and authority, in a readable way, with lots of nice illuminating anecdotes.”

—Robert L. Glass, Publisher/Editor, *The Software Practitioner*; Visiting Professor, Griffith University, Australia

“In an easy-to-read format, *More About Software Requirements* offers practical answers to the most vexing requirements problems faced by analysts and project managers. An essential companion to Wiegers’s *Software Requirements*, it cuts to the chase with real-world wisdom for practitioners. Use this book to find the bottom-line advice you need to succeed with your requirements efforts.”

—Ellen Gottesdiener, EBG Consulting; Author, *Requirements by Collaboration and Software Requirements Memory Jogger*

“Karl has done it again, he has captured the difficult requirement questions, provided sensible solutions, and shown us the related pitfalls.

If you do anything with software requirements you will find answers to your difficult questions clearly addressed and solution pros and cons plainly delineated.”

—Ivy Hooks, Compliance Automation, Inc.

“Sage advice from a requirements master. And, fortunately, Wiegers is a great writer, too. Professionals who take requirements seriously—and we all should take requirements seriously—must read this book.”

—Norm Kerth, Principal, Elite Systems

"If you have requirements problems (and who doesn't), this book will help. *More About Software Requirements* is filled with pragmatic down-to-earth advice. Wiegers helps the entire project team—project managers, developers, testers, and writers, not just the requirements analysts—understand what's happening with their requirements and how to fix those problems. Make this book required reading for your whole project team."

—Johanna Rothman, President of Rothman Consulting Group, Inc.; Coauthor, *Behind Closed Doors: Secrets of Great Management*; Author, *Hiring the Best Knowledge Workers*, *Techies & Nerds: The Secrets and Science of Hiring Technical People*

"Karl Wiegers has added to the treasure trove of advice in *Software Requirements, Second Edition*, by addressing some of the trickiest and most controversial issues in requirements engineering. I get asked questions in the areas covered in *More About Software Requirements* all the time.

The advice is sound; the writing is clear. This is a very useful, practical book."

—Erik Simmons, Requirements Engineering Program Lead, Corporate Quality Network, Intel Corporation

Preface

Requirements engineering continues to be a hot topic in the software industry. More and more development organizations realize that they cannot succeed unless they get the software requirements right. Too often, the people responsible for leading the requirements process are ill-equipped for this challenging role. They do the best they can, but it's an uphill climb without adequate training, coaching, resources, and experience.

As a consultant, trainer, and author, I receive many questions from practitioners about how to handle difficult requirements issues. Certain questions come up over and over again. Alas, there aren't simple answers for many of these. Many books on requirements engineering have been published during the last several years, including my own, *Software Requirements, Second Edition* (Wiegers 2003a). These books provide solid guidance on the challenges of requirements elicitation, analysis, specification, validation, and management. However, additional requirements topics are not covered well by the existing books. Also, some books contain guidance that I believe is ill-founded.

This book addresses some of these recurrent questions that puzzle and frustrate requirements analysts, such as the following:

- “How do I keep too much design from being embedded in the requirements?” (I heard this question again the day before I wrote these words.)
- “When should I baseline my requirements?”
- “How can I convince my managers that we need to do a better job on our project requirements?”
- “What are some good questions to ask in requirements interviews?”
- “Are use cases all I need for documenting the requirements?”
- “We can't get our customers to review the requirements specification. What should I do?”
- “What are some good metrics our organization should collect about our requirements?”
- “We're collecting requirements for multiple releases concurrently. How should I store those?”
- “How can I use requirements to estimate how long it will take to finish the project?”
- “How can I write better requirements?”

I've addressed some other topics in this book simply because little has been written about them. For instance, everyone talks about project scope, but the current books on requirements engineering say little about how to actually define scope. See Chapter 17, “Defining Project Scope,” for some recommendations. Still other topics are included because I don't see

practitioners using some of the established techniques that can help them do a better job. As an example, nearly all requirements specifications I see consist entirely of written text—there's not a picture to be found. However, the skilled analyst should have a rich tool kit of techniques available for representing requirements information. Text is fine in many cases, but other sorts of requirements "views" sometimes are more valuable. Chapter 19, "The Six Blind Men and the Requirements," addresses this topic.



The suggestions I propose in this book augment the "good practices" approach I took in my earlier book. Many cross-references are provided to chapters in *Software Requirements, Second Edition*, marked with the icon shown to the left of this paragraph. As with all such advice, you need to think about how best to apply these suggestions to your specific situation. Organizations are different, projects are different, and cultures are different, so techniques that work in one situation might not be just right for another. To illustrate the application of these practices, I've included many examples of actual project experiences, marked with the "true stories" icon shown to the left here.

Anyone involved with defining or understanding the requirements for a new or enhanced software product will find this book useful. The primary audience consists of those team members who perform the role of the requirements analyst on a software development project, be this their full-time job or just something they do once in awhile. Part II of this book, "On the Management View of Requirements," is focused on aspects of requirements engineering that are of particular interest to project managers and senior managers. Customer representatives who work with the software team will also find certain chapters valuable, particularly those in Part III, "On Customer Interactions," and Part IV, "On Use Cases."

I should point out that all the practices I recommend assume that you're dealing with reasonable people. Sometimes an unreasonable customer will insist on a specific solution that isn't a good fit for the problem. Unreasonable funding sponsors might impose their own inappropriate preferences, overriding the thoughtful decisions made by actual user representatives. Senior managers or influential customers sometimes demand impossible delivery dates for an overly constrained project. If you face such a situation, try educating the difficult people to help them understand the risks posed by the approaches they are demanding and the value of using a better approach. People who appear unreasonable often are just uninformed. Sometimes, though, they truly are unreasonable. I can't help you much with that.

You may download the templates and other process assets described in this book from the Process Impact Web site, <http://www.processimpact.com>. Feel free to share your experiences with me at kwiegers@acm.org.

I hope you'll find this book a valuable supplement to your other resources for software requirements engineering. But don't just read the chapters and say, "That's interesting." Set yourself a personal goal of finding at least three new practices that you want to try the next few times you wear your analyst hat.

Acknowledgments

First, I thank the many people in my seminars who have asked some of these challenging questions over the past several years, as well as the readers who have sent me e-mails with their own thorny requirements issues. I'm grateful for the review input provided by Wayne Allen, Michael Beshears, Steven Davis, Chris Fahlbusch, Lynda Fleming, Betty Luedke, Jeannine McConnell, Terry Nooyen-Coyner, Debbie Shyne, David Standerford, Donna Swaim, and Robin Tucker. The many comments I received from reviewers Ellen Gottesdiener, Andre Gous, and Shannon Jackson were especially valuable. A special thanks goes to Erik Simmons, who provided incisive suggestions on every chapter and greatly helped me sharpen the message. Thanks also to the Microsoft Learning editorial and production teams, including acquisitions editor Ben Ryan, copy editor Michelle Goodman, proofreaders Becka McKay and Sandi Resnick, and artist Joel Panchot. I especially enjoyed the opportunity to work with project editor Devon Musgrave again.

And finally, a big thank-you once again to my ever-cheerful wife, Chris Zambito. Fooled you this time, hon!

Contents at a Glance

Part I On Essential Requirements Concepts

- 1 Requirements Engineering Overview 3**
- 2 Cosmic Truths About Software Requirements 11**

Part II On the Management View of Requirements

- 3 The Business Value of Better Requirements 21**
- 4 How Long Do Requirements Take? 29**
- 5 Estimating Based on Requirements 33**

Part III On Customer Interactions

- 6 The Myth of the On-Site Customer 51**
- 7 An Inquiry, Not an Inquisition 57**
- 8 Two Eyes Aren't Enough 69**

Part IV On Use Cases

- 9 Use Cases and Scenarios and Stories, Oh My! 77**
- 10 Actors and Users 85**
- 11 When Use Cases Aren't Enough 89**

Part V On Writing Requirements

- 12 Bridging Documents 103**
- 13 How Much Detail Do You Need? 105**
- 14 To Duplicate or Not to Duplicate 113**
- 15 Elements of Requirements Style 117**
- 16 The Fuzzy Line Between Requirements and Design 129**

Part VI On the Requirements Process

- 17 Defining Project Scope 137**
- 18 The Line in the Sand 147**
- 19 The Six Blind Men and the Requirements 153**

Part VII On Managing Requirements

| | | |
|----|---|-----|
| 20 | Handling Requirements for Multiple Releases | 165 |
| 21 | Business Requirements and Business Rules | 171 |
| 22 | Measuring Requirements | 177 |
| 23 | Exploiting Requirements Management Tools | 183 |

Table of Contents

| | |
|---------------|------|
| Preface | xiii |
|---------------|------|

Part I On Essential Requirements Concepts

| | | |
|---|---|----|
| 1 | Requirements Engineering Overview | 3 |
| | "Requirement" Defined | 3 |
| | Different Types of Requirements | 4 |
| | Business Requirements | 5 |
| | User Requirements | 5 |
| | Functional Requirements | 6 |
| | System Requirements | 6 |
| | Business Rules | 7 |
| | Quality Attributes | 7 |
| | External Interfaces | 7 |
| | Constraints | 7 |
| | Requirements Engineering Activities | 7 |
| | Looking Ahead | 9 |
| 2 | Cosmic Truths About Software Requirements | 11 |
| | Requirements Realities | 11 |
| | Requirements Stakeholders | 14 |
| | Requirements Specifications | 16 |

Part II On the Management View of Requirements

| | | |
|---|---|----|
| 3 | The Business Value of Better Requirements | 21 |
| | Tell Me Where It Hurts | 21 |
| | What Can Better Requirements Do for You? | 23 |
| | The Investment | 25 |
| | The Return | 26 |
| | An Economic Argument | 28 |

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

Microsoft is interested in hearing your feedback about this publication 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/

| | | |
|----------|--|-----------|
| 4 | How Long Do Requirements Take? | 29 |
| | Industry Benchmarks | 29 |
| | Your Own Experience | 30 |
| | Incremental Approaches | 31 |
| | Planning Elicitation | 32 |
| 5 | Estimating Based on Requirements. | 33 |
| | Some Estimation Fundamentals | 33 |
| | Estimation Approaches | 35 |
| | Goals Aren't Estimates | 37 |
| | Estimating from Requirements | 37 |
| | Measuring Software Size | 38 |
| | Story Points | 40 |
| | Use Case Points | 41 |
| | Testable Requirements | 45 |
| | The Reality of Estimation | 47 |

Part III On Customer Interactions

| | | |
|----------|---|-----------|
| 6 | The Myth of the On-Site Customer. | 51 |
| | User Classes and Product Champions | 51 |
| | Surrogate Users | 53 |
| | Now Hear This | 54 |
| 7 | An Inquiry, Not an Inquisition | 57 |
| | But First, Some Questions to Avoid | 57 |
| | Questions for Eliciting Business Requirements | 59 |
| | User Requirements and Use Cases | 60 |
| | Questions for Eliciting User Requirements | 62 |
| | Open-Ended Questions | 65 |
| | Why Ask Why? | 66 |
| 8 | Two Eyes Aren't Enough | 69 |
| | Improving Your Requirements Reviews | 69 |

Part IV On Use Cases

| | | |
|-----------|--|-----------|
| 9 | Use Cases and Scenarios and Stories, Oh My! | 77 |
| | Use Cases | 77 |
| | Scenarios | 79 |
| | User Stories | 82 |
| 10 | Actors and Users. | 85 |
| 11 | When Use Cases Aren't Enough..... | 89 |
| | The Power of Use Cases..... | 89 |
| | Project Type Limitations | 90 |
| | Event-Response Tables..... | 91 |
| | Use Cases Don't Replace Functional Requirements..... | 93 |
| | Use Cases Reveal Functional Requirements | 96 |

Part V On Writing Requirements

| | | |
|-----------|---|------------|
| 12 | Bridging Documents | 103 |
| 13 | How Much Detail Do You Need?..... | 105 |
| | Who Makes the Call? | 105 |
| | When More Detail Is Needed | 107 |
| | When Less Detail Is Appropriate | 108 |
| | Implied Requirements | 110 |
| | Sample Levels of Requirements Detail | 110 |
| 14 | To Duplicate or Not to Duplicate | 113 |
| | Cross-Referencing..... | 114 |
| | Hyperlinks | 114 |
| | Traceability Links..... | 115 |
| | Recommendation | 116 |
| 15 | Elements of Requirements Style | 117 |
| | I Shall Call This a Requirement..... | 117 |
| | System Perspective or User Perspective? | 119 |
| | Parent and Child Requirements | 120 |
| | What Was That Again? | 121 |
| | Complex Logic | 121 |
| | Negative Requirements..... | 122 |

| | | |
|-----------------|---|------------|
| | Omissions | 124 |
| | Boundaries | 125 |
| | Avoiding Ambiguous Wording | 126 |
| 16 | The Fuzzy Line Between Requirements and Design | 129 |
| | Solution Ideas and Design Constraints | 130 |
| | Solution Clues | 131 |
| Part VI | On the Requirements Process | |
| 17 | Defining Project Scope | 137 |
| | Vision and Scope | 137 |
| | Context Diagram | 138 |
| | Use Case Diagram | 140 |
| | Feature Levels | 141 |
| | Managing Scope Creep | 143 |
| 18 | The Line in the Sand | 147 |
| | The Requirements Baseline | 148 |
| | When to Baseline | 149 |
| 19 | The Six Blind Men and the Requirements | 153 |
| | Limitations of Natural Language | 154 |
| | Some Alternative Requirements Views | 154 |
| | Why Create Multiple Views? | 157 |
| | Selecting Appropriate Views | 159 |
| | Reconciling Multiple Views | 161 |
| Part VII | On Managing Requirements | |
| 20 | Handling Requirements for Multiple Releases | 165 |
| | Single Requirements Specification | 166 |
| | Multiple Requirements Specifications | 167 |
| | Requirements Management Tools | 168 |
| 21 | Business Requirements and Business Rules | 171 |
| | Business Requirements | 171 |
| | Business Rules | 172 |
| | Business Rules and Software Requirements | 173 |

| | | |
|-----------|--|------------|
| 22 | Measuring Requirements | 177 |
| | Product Size..... | 177 |
| | Requirements Quality | 178 |
| | Requirements Status..... | 179 |
| | Requests for Changes..... | 180 |
| | Effort..... | 181 |
| 23 | Exploiting Requirements Management Tools | 183 |
| | Write Good Requirements First | 183 |
| | Expect a Culture Change..... | 184 |
| | Choose a Database-Centric or Document-Centric Tool | 184 |
| | Don't Create Too Many Requirement Types or Attributes..... | 185 |
| | Train the Tool Users | 186 |
| | Assign Responsibilities..... | 186 |
| | Take Advantage of Tool Features | 186 |
| | References | 189 |
| | Index | 195 |

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

Microsoft is interested in hearing your feedback about this publication 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

On Essential Requirements Concepts

This book begins with an overview of some fundamental concepts and principles about software requirements engineering. One of the big obstacles that requirements analysts and other practitioners run into is a lack of common understanding of what “requirements” are and what the requirements engineering process entails. Chapter 1, “Requirements Engineering Overview,” provides readers with a set of requirements-related terms that I’ll use throughout the rest of the book. This should provide a sufficient foundation for understanding the subsequent chapters. For more details on any topic, please refer to *Software Requirements, Second Edition*.



Chapter 2, “Cosmic Truths About Software Requirements,” presents several important insights about software requirements that every manager and practitioner should know. These truths apply to virtually every software project and software development organization, regardless of the nature of the product or the development methodologies being used. No matter what your role is on a project, keep these cosmic truths in mind anytime you’re involved with requirements activities.