

软件工程实践丛书

统一过程精解(影印版)

(美) Kendall Scott 著



THE UNIFIED PROCESS *Explained*



KENDALL SCOTT



清华大学出版社

软件工程实践丛书

统一过程精解

(影印版)

(美) Kendall Scott 著

清华大学出版社

北 京

The Unified Process Explained

Kendall Scott

♣ Addison-Wesley

Boston • San Francisco • New York • Toronto • Montreal
London • Munich • Paris • Madrid
Capetown • Sydney • Tokyo • Singapore • Mexico City

English reprint edition copyright ©2004 by **PEARSON EDUCATION ASIA LIMITED and TSINGHUA UNIVERSITY PRESS.**

Original English language title from Proprietor's edition of the Work.

Original English language title: **The Unified Process Explained** by Kendall Scott, Copyright © 2002.

All Rights Reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Pearson Education, Inc.

This edition is authorized for sale and distribution only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong, Macao SAR and Taiwan).

本书影印版由 Pearson Education 授权给清华大学出版社出版发行。

For sale and distribution in the People's Republic of China exclusively (except Taiwan, Hong Kong SAR and Macao SAR).
仅限于中华人民共和国境内(不包括中国香港、澳门特别行政区和中国台湾地区)销售发行。

北京市版权局著作权合同登记号 图字: 01-2004-0494

本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签, 无标签者不得销售。

图书在版编目(CIP)数据

统一过程精解=The Unified Process Explained/(美)斯科特(Scott, K.)著. —影印版. —北京: 清华大学出版社, 2004.4

(软件工程实践丛书)

ISBN 7-302-08267-7

I. 统… II. ①斯… III. 软件开发—英文 IV. TP311.52

中国版本图书馆 CIP 数据核字(2004)第 018435 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社 总 机: 010-62770175 客户服务: 010-62776969

文稿编辑: 文开棋

封面设计: 陈刘源

印 装 者: 清华大学印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 148 × 210 印张: 6.5

版 次: 2004 年 4 月第 1 版 2004 年 4 月第 1 次印刷

书 号: ISBN 7-302-08267-7/TP · 5963

印 数: 1 ~ 3000

定 价: 19.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770175-3103 或(010)62795704

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and Addison-Wesley was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.

The author and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

The publisher offers discounts on this book when ordered in quantity for special sales. For more information, please contact:

Pearson Education Corporate Sales Division
201 W. 103rd Street
Indianapolis, IN 46290
(800) 428-5331
corpsales@pearsoned.com

Visit us on the Web at www.aw.com/cseng/

Library of Congress Cataloging-in-Publication Data

Scott, Kendall

The unified process explained / Kendall Scott.

p. cm.

Includes bibliographical references and index.

ISBN 0-201-74204-7 (alk. paper)

1. Computer software—Development. 2. Software engineering. I. Title.

QA76.76.D47 S35 2002

005.1—dc21

2001053366

Copyright © 2002 Pearson Education, Inc.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher. Printed in the United States of America. Published simultaneously in Canada.

ISBN 0-201-74204-7

Text printed on recycled paper

1 2 3 4 5 6 7 8 9 10—CRS—0504030201

First printing, November 2001

Preface

Why This Book?

From the moment the Unified Process made its appearance, I heard many people describing it as really big and complicated, at conferences like UML World and in various public forums, such as Rational's Object Technology User Group (OTUG) mailing list. I agreed that in comparison to other well-known processes, it was rather large, but I didn't think that it was all that complicated, all things considered.

In Chapter 2 of *UML Explained*, I managed to describe the fundamental concepts that underlie the Unified Process in about ten pages. While I was still writing that book, it occurred to me that I could probably describe the most important details of the Unified Process in a book not much bigger than that one (that is, 200 pages rather than my usual 150 or so). So, I set about writing this book partially to debunk the notion that the process contained just too much for the average person to get his or her arms around, and also to establish that the process doesn't specify tasks that people on a project don't do anyway, in one way or another.

The result is a book that I've specifically conceived as a companion piece to *UML Explained*. Rather than try to teach you about the UML, which the Unified Process makes fairly heavy use of, I've included references to chapters and sections in that book that offer details about the various UML diagrams and techniques that come into play within the process. I've also brought a number of the diagrams over from that book into this one, to help the continuity and flow across both books. As Picasso said, "Good artists borrow; great artists steal."

Here are some other key features of this book:

- I've made domain modeling and business modeling, which tend to get short shrift in other books about this process, full players, with the associated artifacts and activities part of the Requirements workflow where they belong.
- I've minimized the amount of project management material. Walker Royce's *Software Project Management: A Unified Framework* (Addison-Wesley, 1998) is the definitive work on how to do project management in conjunction with the Unified Process, and I see no need to try to add value to what he's written.
- Chapters 7 through 9 include the story of how *The Internet Bookstore*, a sample project, was designed and built. Whereas Chapters 2 through 6 include diagrams from *UML Explained* relevant to that example, the later chapters explain how the project team broke the system down into chunks in the course of doing iterative and incremental development. *Applying Use Case Driven Object Modeling with UML* (Rosenberg and Scott, Addison-Wesley, 2001) contains other views of the same bookstore project. (My attitude is, stick with what you know!)

My goal was to write a book that would demystify what people like to call A Real Big Process or some variation of that. I hope you think I've succeeded.

Organization of This Book

The body of this book can usefully be divided into four parts.

The first part comprises Chapter 1. This chapter provides an overview of the Unified Process, in the form of a "nutshell" description, some history, exploration of the major themes (use case driven, architecture-centric, and iterative and incremental), and definitions of the major terminology: workflows, phases, iterations and increments, and artifacts, workers, and activities.

The second part comprises Chapters 2 through 6. These chapters provide the details about the five workflows (Requirements, Analysis, Design, Implementation, and Test) that the process defines. Each chapter includes the following:

- An introduction that offers a brief overview of what's included in the workflow and its primary goals
- Descriptions of the various artifacts that get produced during the workflow

- Descriptions of the various roles that people play during the workflow, expressed in terms of “workers”
- Descriptions of the various activities that workers perform during the workflow in order to produce the artifacts

Each of the Activities sections has a diagram that shows the nonlinear nature of the given workflow. Solid lines on this diagram show logical sequences in which to perform the activities; in some cases, one activity is basically a prerequisite to another activity, whereas in other cases, the work that the team performs for one activity will cause a cycling back to one or more activities that it previously performed. Dashed lines are for data flow: The contents of the artifact that results when one activity is finished feed into the next activity or a previous activity.

The third part comprises Chapters 7 through 9. These chapters provide the details about three of the four phases (Inception, Elaboration, and Construction) that the process defines. Each chapter includes the following:

- An introduction that offers a brief overview of what the project team does during the phase, including its primary goals and a high-level look at how each of the five workflows “cuts across” the phase. (You might think of the workflows and the phases as forming a matrix, with the workflows running down the left-hand side and the phases across the top.)
- A description of the tasks that the project manager should perform before the development team begins the activities defined by the phase.
- Descriptions of the activities, defined by one or more of the workflows, that the project team performs during the phase. These descriptions are expressed in terms of what specifically needs to happen during the phase (the team performs an activity to a greater or lesser extent depending on the context) and also in terms of The Internet Bookstore. The bookstore team did one iteration of Inception, three of Elaboration, and two of Construction; each chapter provides part of their story with text that describes what they did for each activity during each iteration and excerpts from the various models that they produced along the way.

Each of these chapters also calls out the deliverables of the given phase at appropriate places.

The fourth part comprises Chapter 10. This chapter describes the Transition phase, which is the phase during which the project team rolls out the system to its customers. The format for this chapter is the same as that for Chapters 7

through 9. This chapter is in a separate part because workflow activities don't cut across Transition the way they do the other three phases, and because the chapter doesn't discuss the bookstore project.

The book also includes the following end matter:

- Appendix A, which describes what the Rational Unified Process (RUP) adds to the core Unified Process
- Appendix B, which compares and contrasts the RUP with the fundamental aspects of eXtreme Programming (XP)
- Appendix C, which describes the ICONIX process, whose roots are the same as those for the Unified Process
- A bibliography, which lists all the books I mention and a few more for the sake of reference
- A glossary, which contains definitions for all the terms I introduce
- A complete index

Acknowledgments

I'd like to thank another set of Lucky 13 people, while acknowledging that some of them were part of *UML Explained*'s Lucky 13: Ross Venables, for leading the charge to get me the deal for this book; Paul Becker, for always being a supportive and patient editor; Doug Rosenberg, because he'd stop giving me work if I didn't thank him yet again; my reviewers (Jim Conallen, Joel Erickson, Jeffrey Hammond, and Jeff Kantor), for their insightful comments; Daphne Head, just because; Ivar Jacobson, for his encouraging words; Russ Coleman, for not minding that I'm implying that he should be writing his own books; Kim Arney Mulcahy, for helping my books look great; and Samson and Smokey, the dogs who keep me company here in the backwoods.

Kendall Scott
 Harrison, Tennessee
kendall@usecasedriven.com
<http://usecasedriven.com>

Contents

List of Figures	xv
Preface.....	xvii
Why This Book?	xvii
Organization of This Book.....	xviii
Acknowledgments.....	xx
Chapter 1: Overview	1
Introduction.....	1
History.....	2
Use Case Driven	3
Architecture-Centric	4
Understanding the Big Picture	5
Organizing the Development Effort.....	6
Facilitating the Possibilities for Reuse.....	6
Evolving the System	6
Guiding the Use Cases	7
Iterative and Incremental	7
Logical Progress Toward a Robust Architecture	7
Dealing With Ongoing Changes in Requirements.....	8
Greater Flexibility to Change the Plan.....	8
Continuous Integration.....	9
Early Understanding.....	9
Ongoing Focus on Risk	9
The Four Phases.....	10
Inception.....	11
Elaboration	12





Construction	13
Transition.....	13
The Five Workflows.....	13
Requirements	14
Analysis	15
Design.....	15
Implementation.....	15
Test	15
Iterations and Increments	16
Artifacts, Workers, and Activities	17
Artifacts	18
Workers	18
Activities.....	18
Chapter 2: The Requirements Workflow	19
Introduction	19
Reach Agreement on the System Context	20
List Candidate Requirements	20
Identify and Negotiate Functional Requirements.....	21
Specify Nonfunctional Requirements.....	21
Artifacts	21
Domain Model.....	22
Business Model	23
Glossary	23
Actor	24
Use Case	24
User-Interface Prototype	25
Use Case Model.....	25
Architecture Description (View of the Use Case Model)	26
Supplementary Requirements.....	26
Workers	27
System Analyst.....	27
Use Case Specifier.....	27
User-Interface Designer.....	27
Architect	28

Activities	28
Build the Domain Model.....	29
Build the Business Model	30
Find Actors and Use Cases	30
Prototype the User Interface.....	32
Prioritize the Use Cases	33
Detail a Use Case	33
Structure the Use Case Model.....	35
Chapter 3: The Analysis Workflow	37
Introduction.....	37
Artifacts.....	38
Analysis Class	38
Use Case Realization–Analysis.....	39
Analysis Package	41
Analysis Model	42
Architecture Description (View of the Analysis Model)	42
Workers.....	43
Architect.....	43
Use Case Engineer	43
Component Engineer.....	43
Activities	44
Perform Architectural Analysis.....	44
Analyze a Use Case.....	45
Analyze a Class	45
Analyze a Package	46
Chapter 4: The Design Workflow.....	49
Introduction.....	49
Artifacts.....	50
Design Class.....	50
Use Case Realization–Design	53
Interface.....	55
Design Subsystem	56
Design Model	56
Architecture Description (View of the Design Model).....	57

Deployment Model	58
Architecture Description (View of the Deployment Model)...	59
Workers	59
Architect	60
Use Case Engineer	61
Component Engineer	62
Activities	62
Perform Architectural Design.....	62
Design a Use Case	64
Design a Class	65
Design a Subsystem.....	66
Chapter 5: The Implementation Workflow	69
Introduction	69
Artifacts	70
Component	70
Interface	71
Implementation Subsystem	72
Implementation Model	72
Architecture Description (View of the Implementation Model).....	73
Integration Build Plan.....	73
Workers	74
Architect	74
Component Engineer	74
System Integrator.....	75
Activities	75
Perform Architectural Implementation.....	75
Implement a Class	75
Perform Unit Test	76
Implement a Subsystem.....	78
Integrate the System	78
Chapter 6: The Test Workflow	79
Introduction	79

Artifacts.....	79
Test Case	79
Test Procedure.....	80
Test Component	80
Test Model	80
Test Plan.....	80
Defect	80
Test Evaluation.....	81
Workers.....	81
Test Engineer	81
Component Engineer.....	81
Integration Tester	81
System Tester.....	82
Activities	82
Plan Test.....	82
Design Test.....	83
Implement Test.....	84
Perform Integration Test	84
Perform System Test.....	84
Evaluate Test.....	84
Chapter 7: The Inception Phase	87
Introduction.....	87
Getting Started	89
Plan the Inception Phase	89
Expand the System Vision	89
Establish the Evaluation Criteria.....	90
Requirements Activities.....	91
Build the Domain Model.....	91
Build the Business Model	92
Find Actors and Use Cases	92
Prioritize the Use Cases	93
Detail a Use Case	94
Analysis Activities	95
Perform Architectural Analysis.....	95
Analyze a Use Case.....	95



Design Activities	95
Perform Architectural Design	96
Taking Stock	97
Assess Each Iteration	98
Assess the Phase as a Whole	99
Looking Ahead	99
Make the Initial Business Case	99
Do Initial Planning for the Elaboration Phase	100
Chapter 8: The Elaboration Phase	101
Introduction	101
Getting Started	103
Plan the Elaboration Phase	103
Establish the Evaluation Criteria	103
Requirements Activities	104
Build the Domain Model	104
Build the Business Model	105
Find Actors and Use Cases	106
Prototype the User Interface	107
Prioritize the Use Cases	108
Detail a Use Case	108
Structure the Use Case Model	109
Analysis Activities	109
Perform Architectural Analysis	109
Analyze a Use Case	110
Analyze a Class	112
Analyze a Package	113
Design Activities	113
Perform Architectural Design	113
Design a Use Case	114
Design a Class	115
Design a Subsystem	116
Implementation Activities	118
Perform Architectural Implementation	118
Implement a Class	120
Perform Unit Test	121



Implement a Subsystem	122
Integrate the System	122
Test Activities	122
Plan Test	122
Design Test	123
Implement Test	123
Perform Integration Test	124
Perform System Test	124
Evaluate Test	124
Taking Stock	124
Assess Each Iteration	125
Assess the Phase as a Whole	125
Looking Ahead	125
Make the Full Business Case	125
Do Initial Planning for the Construction Phase	126
Chapter 9: The Construction Phase	127
Introduction	127
Getting Started	128
Plan the Construction Phase	128
Establish the Evaluation Criteria	129
Requirements Activities	129
Find Actors and Use Cases	129
Prototype the User Interface	130
Prioritize the Use Cases	131
Detail a Use Case	131
Structure the Use Case Model	131
Analysis Activities	131
Perform Architectural Analysis	132
Analyze a Use Case	132
Analyze a Class	132
Analyze a Package	133
Design Activities	133
Perform Architectural Design	133
Design a Use Case	133



Design a Class	134
Design a Subsystem.....	134
Implementation Activities	135
Implement a Class	135
Perform Unit Test	135
Implement a Subsystem.....	136
Integrate the System	136
Test Activities	136
Plan Test	136
Design Test.....	137
Implement Test.....	137
Perform Integration Test.....	138
Perform System Test	138
Evaluate Test	139
Taking Stock	139
Assess Each Iteration.....	139
Assess the Phase as a Whole	139
Looking Ahead.....	140
Do Initial Planning for the Transition Phase	140
Chapter 10: The Transition Phase.....	141
Introduction	141
Getting Started.....	142
Plan the Transition Phase	142
Establish the Evaluation Criteria.....	142
Activities	142
Get the Beta Release Out.....	142
Install the Beta Release	143
Respond to Test Results	143
Adapt the Product to Varied User Environments.....	144
Complete the Artifacts.....	144
Taking Stock	145
Assess Each Iteration.....	145
Assess the Phase as a Whole	145