

《统一软件开发过程》

图书基本信息

书名：《统一软件开发过程》

13位ISBN编号：9787302099178

10位ISBN编号：7302099170

出版时间：2005-1-1

出版社：清华大学出版社

作者：Ivar Jacobson, Grady Booch, James Rumbaugh

页数：496

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《统一软件开发过程》

内容概要

本书由UML创始人精心编写，介绍了创建流行软件的新标准，对软件开发人员或管理人员而言，具有重要的参考价值。

书中不仅指出了贯穿软件开发过程UML文档类型（比如用例、类图和状态转换图）与各种模型之间的关系，演示了如何用UML模型来说明过程，还清清楚楚地描述了模型中不同高级结构的语义和符号表示。通过本书的阅读，读者将了解到当前软件领域内的最佳做法。

《统一软件开发过程》

作者简介

Ivar Jacobson, Grady Booch, James Rumbaugh是UML的创始人,同时也是享誉全球的面向对象技术专家,他们为对象技术的开发做出了卓越的贡献。

《统一软件开发过程》

书籍目录

Part I: The Unified Software Development Process Chapter 1: The Unified Process: Use-Case Driven, Architecture-Centric, Iterative, and Incremental 1.1 The Unified Process in a Nutshell 1.2 The Unified Process Is Use-Case Driven 1.3 The Unified Process Is Architecture-Centric 1.4 The Unified Process Is Iterative and Incremental 1.5 The Life of the Unified Process 1.6 An Integrated Process Chapter 2: The Four Ps: People, Project, Product, and Process in Software Development 2.1 People Are Crucial 2.2 Projects Make the Product 2.3 Product Is More Than Code 2.4 Process Directs Projects 2.5 Tools Are Integral to Process 2.6 References Chapter 3: A Use-Case-Driven Process 3.1 Use-Case-Driven Development in Brief 3.2 Why Use Cases? 3.3 Capturing the Use Cases 3.4 Analysis, Design, and Implementation to Realize the Use Cases 3.5 Testing the Use Cases 3.6 Smming Up 3.7 References Chapter 4: An Architecture-Centric Process 4.1 Architecture in Brief 4.2 Why We Need Architecture 4.3 Use Cases and Architecture 4.4 The Steps to an Architecture 4.5 Finally, an Architecture Description! 4.6 Three Interesting Concepts 4.7 References Chapter 5: An Iterative and Incremental Process 5.1 Iterative and Incremental in Brief 5.2 Why Iterative and Incremental Development? 5.3 The Iterative Approach is Risk-Driven 5.4 The Generic Iteration 5.5 The Result of an Iteration Is an Increment 5.6 Iterations over the Life Cycle 5.7 Models Evolve from Iterations 5.8 Iterations Challenge the Organization 5.9 ReferencesPart II: The Core Workflows Chapter 6: Requirements Capture: From Vision to Requirements 6.1 Why Requirements Capture Is Difficult 6.2 The Purpose of the Requirements Workflow 6.3 Overview of Requirements Capture 6.4 The Role of Requirements in the Software Life Cycle 6.5 Understanding the System Context Using a Domain Model 6.6 Understanding the System Context Using a Business Model 6.7 Supplementary Requirements 6.8 Summary 6.9 References Chapter 7: Capturing the Requirements as Use Cases 7.1 Introduction 7.2 Artifacts 7.3 Workers 7.4 Workflow 7.5 Summary of the Requirements Workflow 7.6 References Chapter 8: Analysis 8.1 Introduction 8.2 Analysis in Brief 8.3 The Role of Analysis in the Software Life Cycle 8.4 Artifacts 8.5 Workers 8.6 Wordflow 8.7 Summary of Analysis 8.8 References Chapter 9: Design 9.1 Introduction 9.2 The Role of Design in the Software Life Cycle 9.3 Artifacts 9.4 Workers 9.5 Wordflow 9.6 Summary of Design 9.7 References Chapter 10: Implementation 10.1 Introduction 10.2 The Role of Implementation in the Software Life Cycle 10.3 Artifacts 10.4 Workers 10.5 Workflow 10.6 Summary of Implementation 10.7 References Chapter 11: Test 11.1 Introduction 11.2 The Role of Testing in the Software Life Cycle 11.3 Artifacts 11.4 Workers 11.5 Workflow 11.6 Summary of Testing 11.7 ReferencesPart III: Iterative and Incremental Development Chapter 12: The Need for Balance 12.1 The Need for Balance 12.2 The Phases Are the First Division of Work 12.3 The Generic Iteration Revisited 12.4 Planning Precedes Doing 12.5 Risks Affect project Planning 12.6 Use-Case Prioritization 12.7 Resources Needed 12.8 Assess the Iterations and Phases Chapter 13: Inception Launches the Project 13.1 The Inception Phase in Brief 13.2 Early in the Inception Phase 13.3 The Archetypal Inception Iteration Workflow 13.4 Execute the Core Workflows, Requirements to Test 13.5 Make the Initial Business Case 13.6 Assess the Iteration(s)in the Inception Phase 13.7 Planning the Elaboration Phase 13.8 The Deliverables for the Inception Phase Chapter 14: the Elaboration Phase Makes the Architectural Baseline 14.1 The Elaboration Phase in Brief 14.2 Early in the Elaboration Phase 14.3 The Archetypal Elaboration Iteration Workflow 14.4 Execute the Core Workflows-Requirements to Test 14.5 Make the Business Case 14.6 Assess the Iterations in the Elaboration Phase 14.7 Planning the Construction Phase 14.8 The Key Deliverables Chapter 15: Construction Leads to Initial Operational Capability 15.1 The Construction Phase in Brief 15.2 Early in the Construction Phase 15.3 The Archetypal Construction Iteration Workflow 15.4 Execute the Core Workflows-Requirements to Testing 15.5 Controlling the Business Case 15.6 Assess the Iterations and the Construction Phase 15.7 Planning the Transition Phase 15.8 The Key Deliverables Chapter 16: Transition Completes Product Release 16.1 The Transition Phase in Brief 16.2 Early in the Transition Phase 16.3 The Core Workflows Play a Small Role in this Phase 16.4 What We Do in the Transition Phase 16.5 Completing the Business Case 16.6 Assess the Transition Phase 16.7 Planning the Next Release or Generation 16.8 The Key Deliverables Chapter 17: Making the Unified Process Work 17.1 The Unified Process Helps You Deal With Complexity 17.2 The Major Themes 17.3 Management Leads Conversion to Unified Process 17.4 Specializing the Unified Process 17.5 Relate to the

《统一软件开发过程》

Broader Community 17.6 Get the Benefits of the Unified Process 17.7 References Appendix A: Overview of the UML A.1 Introduction A.2 Graphical Notation A.3 Glossary of Terms A.4 References Appendix B: The Unified Process-Specific Extensions of the UML B.1 Introduction B.2 Stereotypes B.3 Tagged Values B.4 Graphical Notation B.5 References Appendix C: General Glossary C.1 Introduction C.2 Terms Index

《统一软件开发过程》

精彩短评

- 1、本书是软件工程方面的比较权威的书。注重将以前分散、零星的各类软件开发过程，统一起来。就像交响乐队的指挥一样。为我们开启软件开发过程的一种新的组织形式。。。
- 2、对于一些大型软件项目，特别是有充足的资金和人员的情况下，可以借鉴这种软件开发思路。个人认为，它的需求，分析与设计过程非常值得借鉴，然后结合敏捷软件开发在“Iterator阶段”的方式，才更能整合出适合我们平时碰到的大多数项目的流程。当然，还是那句话，流程不是万能的，但没有流程却是万万不能的，可以好好借鉴一下这本书的思路，把经典软件工程抽象出来，结合现有项目进行思考。软件管理人员，项目经理和TTL应该读这本书，开发人员也应该了解它的思想。

《统一软件开发过程》

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu000.com