

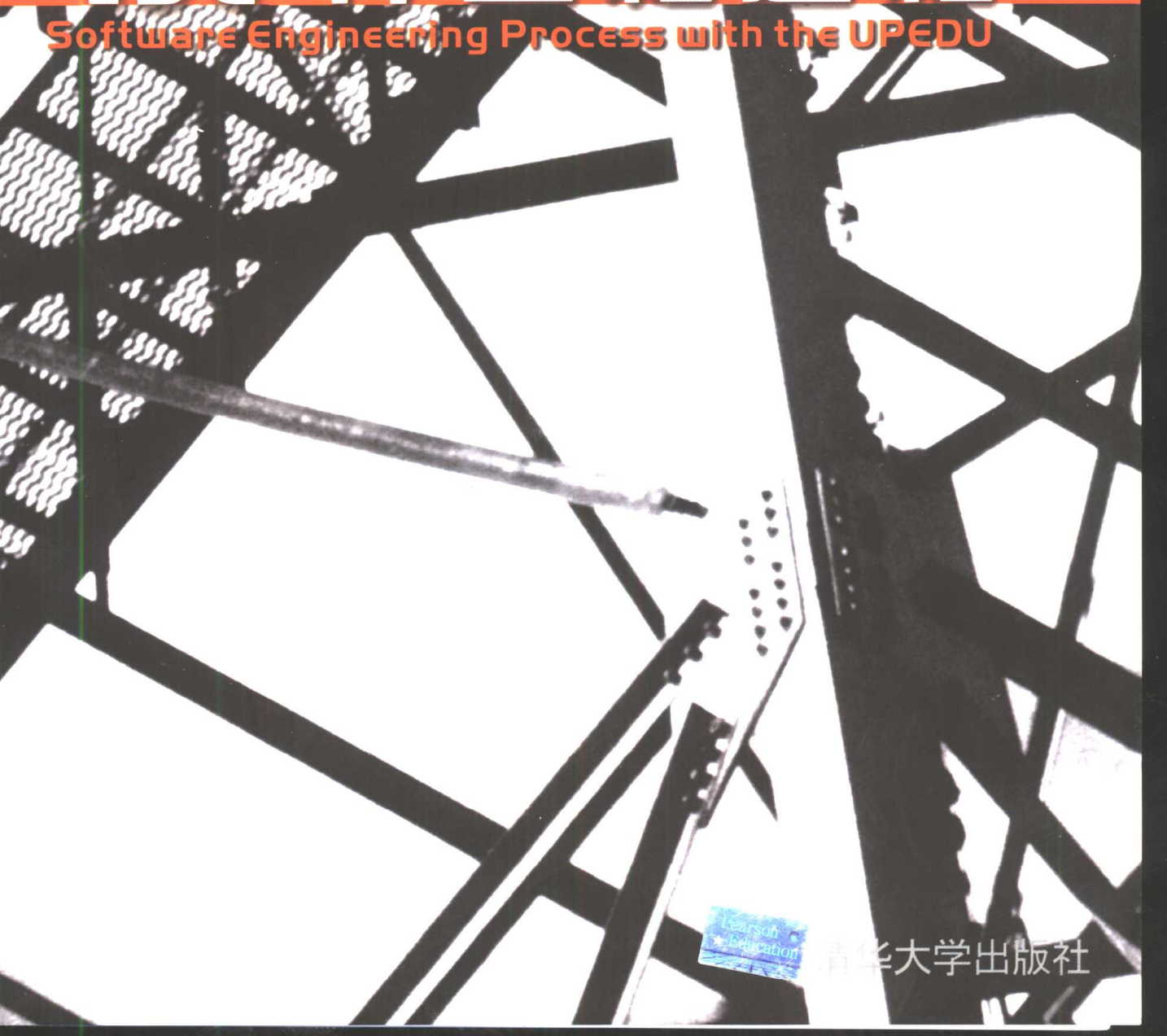
软件工程实践丛书



Pierre N. Robillard
[加] Philippe Kruchten 著 施平安 译
Patrick d'Astous

软件过程

Software Engineering Process with the UPEDU



清华大学出版社

软件工程实践丛书

软 件 工 程 过 程

[加] Pierre N. Robillard
Philippe Kruchten 著
Patrick d'Astous

施平安 译

清华大学出版社

北 京

内 容 简 介

本书以 UPEDU 软件工程过程作为具体的过程实例, 全面介绍了软件工程过程的基本知识, 阐述了一系列有助于在更短的时间内开发出更好的软件的活动。全书分 4 个部分: 第 I 部分介绍了软件过程的基本问题, 即软件生命期的方法、工具和概念; 第 II 部分和第 III 部分主要介绍软件工程规范和管理规范; 第 IV 部分介绍软件工程过程的质量和建模问题, 最后一章介绍了软件过程元模型, 它是所有软件过程的理论基础。

本书主要作为软件工程过程的教材, 要求读者具备一定的面向对象方法和 UML 知识。

Simplified Chinese edition copyright © 2003 by PEARSON EDUCATION ASIA LIMITED and TSINGHUA UNIVERSITY PRESS.

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

Original English language title: Software Engineering Process with the UPEDU, 1st Edition by

Pierre N. Robillard, Philippe Kruchten, Patrick d'Astous Copyright © 2003

EISBN: 0-201-75454-1

All Rights Reserved.

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

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

本书中文简体翻译版由 Pearson Education 授权给清华大学出版社在中国境内(不包括中国香港、澳门特别行政区)出版发行。

北京市版权局著作权合同登记号 图字: 01-2003-0853

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

软件工程过程/[加]罗毕拉德, [加]克鲁奇特, [加]阿斯顿著; 施平安译.—北京: 清华大学出版社, 2003

(软件工程实践丛书)

书名原文: Software Engineering Process with the UPEDU

ISBN 7-302-06734-1

I. 软… II. ①罗…②克…③阿…④施… III. 软件工程 IV. TP311.5

中国版本图书馆 CIP 数据核字 (2003) 第 045808 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: 赵春玉

封面设计: 立日新设计公司

印 刷 者: 北京牛山世兴印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 170×230 印 张: 22 字 数: 425 千字

版 次: 2003 年 8 月第 1 版 2003 年 8 月第 1 次印刷

书 号: ISBN 7-302-06734-1/TP·5020

印 数: 1~4000

定 价: 45.00 元

前 言

概 述

软件开发过程是任何软件开发项目不可分割的一个组成部分。一方面，过程指为了实现目标而执行的一系列操作；另一方面，工程指复杂产品的设计和制造。因此，软件工程过程指为了设计和制造复杂的软件而执行的一系列操作。本书全面介绍了软件工程过程的概念。我们作为实例介绍的具体过程取自 Rational 统一过程（Rational Unified Process，简称 RUP），但是书中所描述的操作是大多数软件过程描述广泛接受的。因此，本书既向读者介绍了一般的过程操作，又可以让他们熟悉 RUP。

本书组织形式

本书分为 4 个部分。

第 I 部分：首先通过一个简短的故事说明软件开发过程的基本问题；然后介绍软件开发生命期的概念，包括方法和工具。

第 II 部分和第 III 部分：主要介绍软件工程规范和管理规范。

第 IV 部分：关于软件工程过程的质量和建模问题，介绍过程评估和过程度量。最后一章介绍所有软件过程的理论基础，它基于软件过程元模型。

本书采用一种实用的软件过程方法，旨在介绍现实的过程经验。因此，每章都遵循相同的总体结构：

- 首先介绍该主题的基本概念——该主题的本质。
- 然后基于 UPEDU 软件工程过程，在活动和制品中应用这些概念。
- 为了说明这些基本概念，在第 1 章介绍了一些案例研究；然后在后面各章中，根据该案例研究导出一些小例子，并对它们进行研究。

- 此外，读者可以借助于本书的配套网站 www.YOOPEEDOO.org，详细地研究从该案例研究中得到的制品，加深对该主题的理解。
- 最后，每章都以一个标题为“业内人士视点”的部分结束。这些简短的文章由特约作者撰写，他们都是软件开发领域公认的专家。

教师资料

本书内容主要是作为教师资料进行安排的，因此介绍了许多有助于教师准备和讲授“软件开发过程”课程的特征。

案例研究

第 1 章介绍了一个案例研究，它贯穿本书始终。第 1 章通过描述创建时间监控工具（Time Monitoring Tool，简称 TMT）的环境和目的，介绍了软件过程的概念。TMT 是一个跟踪软件开发项目所用时间的工具。第 1 章形式化地定义了 TMT 案例研究的规格说明。在本书的其余章节中，TMT 作为例子使用。本书的配套网站 www.YOOPEEDOO.org 对该案例进行了扩展。

该案例研究是一个由 5 位学生组成的团队完成的完整的项目。可以获得所有的制品和操作软件，以便学生可以知道一个制品的全部细节，以及各制品在一个实际项目中是如何组装在一起的。

一个工业强度的过程

本书独有的另一个特征是 UPEDU 软件工程过程。UPEDU（读作 you'-pe-doo）指 Unified Process for Education（教学版统一过程）。UPEDU 是 Rational 统一过程的简化版本，适用于教育环境。同时，UPEDU 是一个完整的软件工程过程，可以在一个教育环境中用来支持课程计划。UPEDU 非常全面，足以支持所有必要的活动和制品；然而，它又足够小，学生在一个学期的课程时间内就可以理解和使用它。UPEDU 用实际的软件工程过程为学生提供动手经验，为他们从事任何成熟的工业化软件工程项目作准备。本书的每一章都强调 UPEDU 中详细讲述的概念和原则。

每讨论一个与 UPEDU 有关的概念或者指南时，就在页边的空白处标一个图标。本书各章中讲述的所有概念都可以用 UPEDU 软件过程进行确认。可以从配

套网站中得到 UPEDL。该网站上关于 TMT 案例研究开发过程的制品，都是遵循 UPEDU 完成的。

业内人士视点

本书的一个特点是每章结尾都有一个标题为“业内人士视点”的部分。每篇精选的短文都由一位特约作者撰写，介绍了与每章主题相关的观点。这些观点易于理解，它们提出了关于该主题的个人见解。这些短文由不同的业内人士撰写，旨在扩展学生们的感性认识，培养他们有关软件工程过程的关键意识。

问题和解答

为了帮助学生验证他们对各章主题的理解程度，每章后面包含了 3 套习题。第 1 套为复习题，只要仔细地阅读各章内容就可以解答这些问题。第 2 套为 UPEDU 问题，通过导航配套网站 www.YOOPEEDO.org 上的 UPEDU 可以回答这些问题。用于找到答案的路径提供了关于各章主题的有用信息。第 3 套习题称为“WWW 问题”，要求学生浏览网页才能找到答案。

本周项目

周项目是一个小项目，可以作为补充作业，由 2~4 个学生组成的团队完成。每个项目的平均工作量在 10 小时以内。

补充阅读材料

最后，每章后面列出了 3 篇经过仔细挑选的补充阅读材料（可以从配套网站上得到这些补充阅读材料的内容）。第 1 篇适于普通读者，通常是第 1 次接触软件工程的读者。第 2 篇适于高级读者；他们可能是上第 2 门软件工程课的学生，也可能是在该领域有些经验的业内人士。第 3 篇适于高年级大学生、研究生，以及希望熟悉软件工程过程各主题涉及的典型研究工作的研究人员。

Web 的利用

Web 提供了一种新的教育模式，是所有希望了解最新消息的软件工程师的重要资源。现在，只要点击鼠标就可以得到大量信息。但是处理所有的这些信息不是一件容易的事情，因为并非从 Web 搜集到的所有信息都是可靠的或者可证实的，

而且存在大量的重复信息。新手往往要花大量时间才能分辨出相关的信息。为了帮助读者熟悉 Web，每章都包含一些经过仔细挑选的与本章内容相关的可靠的 Web URL。这些 Web URL 包含 3 类地址。第 1 类地址是可能会充实本章内容的站点。第 2 类 URL 通常包括参考书目或者参考资料。第 3 类列出了论坛地址，教师、学生或者业内人士可以在这里交流与本章主题相关的信息。

使本课程适于教学

阅读本书的前提条件是较好地理解了面向对象方法和 UML。让学生们先学一门软件工程的入门课，这可以为他们学习软件工程过程课打下一个良好的基础。上完本课程后，学生们应该清楚地了解完整的软件工程过程。这样，他们在开始一个项目之前，会根据软件环境和团队结构，定制适于项目的软件工程过程，这是他们要完成的第一项任务。

本书还可以作为研究生教材。在这种情况下，应当强调补充阅读材料和个人工作。

教学大纲建议

尽管本书的结构非常适于教学，但这里给出关于第 1 章、第 3 章和第 13 章的作用和使用的额外说明。

第 1 章包括一个介绍性的故事，教师可以用任何切身体验对它进行补充。随后的内容是关于案例研究的规格说明，根据 IEEE 标准 830 进行编制。这时，教师可以强调标准在软件工程中的重要性，并确保学生明白需要标准的原因。

第 3 章首先简单地介绍了统一建模语言（Unified Modeling Language，简称 UML），熟悉 UML 的学生可以通过这些内容进行简要回顾。在讲授本章时，教师还可以介绍本课程的软件项目要使用的任何特定的软件开发工具。

第 13 章介绍软件过程的元模型，它形成了描述软件工程过程的理论基础。教师根据学生的基本情况和课程计划决定是否讲述该章内容。在讲述理论性很强的计算机科学课程时，最好提前讲述该章内容，例如，在讲述第 3 章内容之前介绍该章内容。对于更加注重实践的软件工程课程，例如工程系，可以最后介绍第 13 章。

软件开发业内人士也可以通过本书初步地理解软件工程过程，并能大致了解统一过程（Unified Process）。许多章节直接满足了业内人士的迫切需要。第 2 章

介绍软件过程和软件开发生命期；第 4~7 章介绍软件工程过程的工程规范；第 8~9 章介绍软件工程过程的管理规范。

致 谢

感谢 Addison Wesley Longman 的计算机科学组的编辑小组和出版小组，感谢他们的支持和鼓励。采集编辑 Maite Suarez-Rivas 立刻认可了本书的价值，认为本书非常值得出版。项目编辑 Katherine Harutunian 的经验和杰出的专业水平使我们受益匪浅。特别感谢 Maggie Carr，感谢他在手稿的规划和审稿中所做的大量工作。Pat Mahtani 和 Diane Freed 为出版过程付出了辛勤的劳动。

感谢罗彻斯特理工学院的 Henry A. Etlinger 教授、圣安东尼奥学院的 Thomas Friday 教授、Kennesaw 州立大学的 Dick Gayler 教授、印地安纳大学信息学院的 Dennis Groth 教授、得克萨斯州农业及工业大学的 William Lively 教授、北卡罗来纳州州立大学的 Laurie Williams 教授以及中部佛罗里达州大学的 David A. Workman 教授，他们对本书的有关章节进行了审阅。特别感谢 S2I-Solutions 公司的 John Slavich，感谢他从一个业内人士的视点全面审阅了本书。

特别感谢以下学生：Guillaume Boudreau, Jacqueline-Margot Escalante-Palacios, Christian Paré, Christian Robidoux 和 Martin Trudeau。他们在 Ecole Polytechnique de Montreal 完成工程学位期间建立了本书的案例研究。感谢 Bombardier Transport North America 的 Mikel Doucet，感谢他在定义案例研究的规格说明方面的合作。

还要感谢 Simon Hardy 及其团队、Nataly Désir 和 Maria Mbaye，感谢他们仔细地审阅和确认正文、幻灯片、UPEDU 以及案例研究制品间的链接。

UPEDU 的设计是作者与 Patrick d'Astous 密切合作的结果。Christian Robidoux 完成了最终实现。最后，配套网站在 Mario de Repentigny 和 Nicolas Kruchten 的努力下首次开通。UPEDU 的最后版本是 Christian Robidoux 辛勤工作的成果。

感谢 Rational 软件公司在 UPEDU 实现上的技术支持和经济支持，他们不仅提供了 RUP，而且还提供了用于设计本书中许多 PowerPoint 幻灯片和图形的原始材料；尤其要感谢 Dean Leffingwell 院长。感谢 Ecole Polytechnique de Montréal 提供了合适的环境，以及实现该项目和支持 YOOPEEDOO.org 网站所需的一些资源。

感谢撰写“业内人士视点”部分的朋友和同事，他们是：Rational 软件公司的 Joe Marasco, Geoff Clemm 和 Jim Rumbaugh；Mitre 公司的 James Moore；夫琅

和费软件工程实验学院(Fraunhofer Institute for Experimental Software Engineering)的前软件质量工程系主任,现为卡尔阿尔贝特州立学院教授的 Lionel Briand 以及 Alcyonix 的 Richard Basque。

我(皮埃尔)要感谢菲利普,几年前在范库弗峰初次见面后,他立即对 UPEDU 项目充满了信心,并组织团队进行了大量有益的工作。感谢我的妻子 Madeleine,感谢她再次对我的写作表现出的无限耐心,她善意的幽默给了我极大的帮助。感谢我的孩子 Catherine, Daniel 和 Julie,感谢他们能够理解父亲深夜工作。感谢我的儿子 Martin,他在不列颠哥伦比亚大学(University of British Columbia,简称 UBC)完成软件工程博士学位时,从学生的角度仔细地审阅了本书,并对规格说明进行了形式化。

我(菲利普)要感谢皮埃尔及其团队,感谢他如此成功地将其教学经验应用于我们的产品 RUP 中,并建立了出色的伴随过程:UPEDU。感谢我的妻子 Sylvie,感谢她表现出的耐心;感谢我的女儿 Alice 和 Zoé,感谢她们理解父亲再度出行 Montréal;感谢我的儿子 Nicolas,感谢他开始了 *YOOPEEDOO.org* 的开发工作。

配套网站: www.YOOPEEDOO.org

本书有一个配套网站,它提供了本书的支持和补充材料。该网站由 3 部分组成。

第 1 部分读者可以不加限制地进行访问。它包含大家普遍感兴趣的资料,包括 UPEDU 的更加详细的描述、关于软件开发过程的信息/新闻来源的链接,以及关于标准、工具等信息的链接。

第 2 部分限于本书的主人使用。他们有权访问 UPEDU 软件过程(它有 1 500 多个链接在一起的信息页面),以及案例研究的全部制品所对应的模板。他们还可以访问所选的阅读材料,这些阅读材料是各种杂志上发表的论文。最后,本书的主人还可以访问本书中所有 PDF 格式的图形。为了便于访问,所有这些材料都按章进行组织。

第 3 部分只有有资格的教师才能访问。教师不仅可以访问本书主人可以访问的所有资源,而且在第 3 部分中,他们还可以发现其他有助于讲课的内容。例如,他们可以找到 Microsoft PowerPoint 格式的本书插图,以便他们可以根据需要进行修改。他们还可以访问大多数习题的答案,可以使用专题论坛和邮件列表,他们可以在这里与其他教师交流思想和考试问题。

译者序

软件开发是一个集科学、艺术和工程于一身的过程，涉及需求、分析与设计、实现、测试以及管理等活动，是任何软件开发项目都不可缺少的。一方面，过程指为了实现目标而执行的一系列操作；另一方面，工程指复杂产品的设计和制造。因此，软件工程过程是指为了设计和制造复杂的软件产品而执行的一系列操作。本书以 UPEDU 软件工程过程作为具体的过程实例，全面介绍了软件工程过程的基本知识，阐述了一系列有助于在更短的时间内开发出更好的软件的活动。

本书作者皮埃尔（Pierre）是专门从事软件过程教育的专家，有着丰富的教学经验。作者菲利普（Philippe）是 Rational 统一过程的主要设计师之一。UPEDU 是皮埃尔关于软件过程的教学经验和 RUP 相结合的产物，是 Rational 统一过程的简化版本，专门用于教学。UPEDU 完全支持一个软件开发过程的所有必要的活动和制品，学生在一个学期的课程时间内就可以理解和使用它。UPEDU 为学生提供了动手经验，为他们以后从事任何成熟的工业软件工程项目做好了准备。

本书的内容安排非常适于教学。第 I 部分通过一个简短的故事说明了软件开发过程的基本问题，然后介绍软件过程涉及的方法、工具以及软件开发生命期的概念；第 II 部分和第 III 部分主要介绍软件工程规范和管理规范；第 IV 部分介绍了软件工程过程的质量和建模问题，包括过程评估和过程度量。最后一章介绍软件过程元模型，它是所有软件过程的理论基础。本书在内容安排上的一个鲜明特点是，每章结尾都有一篇精选的由特约作者撰写的短文，介绍了关于每章主题的个人观点，旨在扩展学生们的感性认识，培养他们有关软件工程过程的关键意识。

本书主要作为软件工程专业的教材。既适用于初次接触软件工程的读者，也适于高年级大学生、研究生，以及希望熟悉软件工程过程各主题涉及的典型研究工作的研究人员。此外，软件开发业内人士也可以通过本书初步了解软件工程过程。

全书由我一人翻译，施惠琼负责全书的录入和审校工作，在此对她表示感谢。还要感谢我的家人在我翻译本书的过程中表现出来的无限耐心，感谢他们为我所提供的帮助和支持。尤其是我的女儿，她那天真而幼稚的话语“爸爸，您最辛苦了！”使我感到无限的安慰。

在翻译过程中，我对本书中出现的所有术语和难词难句都进行了仔细的推敲和研究，然而有些方面译者本人也是初次接触，也是在学习和理解的过程中进行翻译，所以疏漏和争议之处在所难免，望广大读者提出宝贵的意见。

施平安

2003/2/21 于海军广州舰艇学院

目 录

第 1 章	对软件过程与案例研究的需求	1
1.1	对一个单周项目进行规定	1
1.2	开发一个单周项目	8
1.3	学习基本的软件工程过程	12
1.4	对案例研究进行规定	15
1.5	小结	23
第 2 章	软件过程和软件生命期	29
2.1	理解软件开发与软件演变	29
2.2	软件生命期的定义	31
2.3	软件过程的定义	33
2.4	软件生命期和软件过程的集成	38
2.5	软件过程定制	42
2.6	小结	44
第 3 章	模型和工具	53
3.1	为什么需要模型和工具	53
3.2	建模概念的定义	56
3.3	获取建模图	62
3.4	寻找和使用正确的 CASE 工具	71
3.5	小结	73
第 4 章	需求规范	81
4.1	需求范围	81
4.2	需求制品的定义	84
4.3	引出需求	89
4.4	需求的演变	96
4.5	小结	97

第 5 章 分析与设计规范	105
5.1 结晶过程类比	105
5.2 理解分析与设计规范	107
5.3 分析与设计活动的定义	109
5.4 分析与设计规范的记录	114
5.5 模型视图	117
5.6 小结	121
第 6 章 实现规范	131
6.1 理解实现规范	131
6.2 实现规范的执行	134
6.3 实现规范的详细介绍	142
6.4 构造阶段迭代介绍	147
6.5 小结	149
第 7 章 测试规范	157
7.1 软件测试与软件质量的关系	157
7.2 对测试规范活动进行定义	160
7.3 测试规范的详细描述	168
7.4 测试活动的评估	175
7.5 小结	178
第 8 章 软件配置与变更管理规范	185
8.1 软件配置与变更管理规范的定义	185
8.2 配置与变更管理规范的操作特征	191
8.3 软件配置与变更管理规范的管理	196
8.4 实施软件配置与变更管理面临的挑战	202
8.5 小结	206
第 9 章 软件项目管理规范	213
9.1 定义项目经理的角色	213
9.2 项目规划	218
9.3 迭代的管理	222
9.4 风险管理	225

9.5 小结	229
第 10 章 团队	243
10.1 了解团队	243
10.2 安排团队环境	246
10.3 团队的定义	248
10.4 建立一个团队	252
10.5 支持团队活动	256
10.6 小结	258
第 11 章 过程评估与改进	265
11.1 过程的定义和参考过程模型	265
11.2 注册和认证	269
11.3 UPEDU 评估	277
11.4 小结	285
第 12 章 软件过程度量	293
12.1 过程活动的度量	293
12.2 度量范式的定义	296
12.3 工作量度量	301
12.4 认知活动的度量	306
12.5 小结	313
第 13 章 过程工程元模型	321
13.1 元模型	321
13.2 结构化元模型	324
13.3 统一软件过程元模型概述	330

第 1 章 对软件过程与案例研究的需求

短语“软件工程过程”可能会使人想起大量枯燥的结合件，充满策略、指令和有大量管理行话的表单。有人可能会这样认为，使用一个由如此多规则规定的过程，也许只对非常大的公司才有用。这些公司有大量的程序员，他们聚集在一个巨大的房间里，在一个“精明的经理”的领导下，在许多程序员的努力下，非常缓慢地开发软件。这些软件交给政府机构或者财富排行在 500 强内的公司使用。

软件工程过程没必要如此巨大。可以根据手头的工作和开发组织的规模决定软件工程过程的大小。软件工程过程经过剪裁，可以变成适用于 200 个开发人员的软件工程过程，也可以变成适用于 1 个开发人员的软件工程过程。

软件工程过程的目的是不是使开发者的日子更难过，也不是仅仅在文书工作上做文章。它惟一的真正目的是，确保软件开发组织能够有预见性地设计软件，并在预算内按时交付满足用户所有需要、需求和约束的高质量软件。

本章介绍一个单周、单人软件开发过程，并介绍了在全书中用来说明 UPEDU 软件工程过程的各种制品的案例研究。

在下文中，一位经验丰富的软件工程师尼克（Nick），讲述了他最近为朋友马田（Martin）完成的一个单周项目的故事。

1.1 对一个单周项目进行规定

尼克是一位经验丰富的软件工程师。他已有 12 年左右的软件开发经历，虽然现在喜欢单独工作，但他慎重而自觉地遵守一个经过良好定义的开发过程。在下文中，尼克讲述了他最近为老朋友马田完成的一个单周项目的故事。

星期六：晚上

我和朋友马田在我们最喜欢去的公众聚谈处相遇。他是一家小公司的软件开发经理。公司最近接受了个人软件过程（Personal Software Process）培

训^①，以提高公司的过程效率和可预测性。

但是马田遇到了一点麻烦：该方法的一个关键因素是各开发者要记下他们在每类活动上所花的时间。每个开发者都用一种不同的方法来记录其在一个项目上所花的时间，到一周结束时，收集并整理所有这些数据真是一件可怕的事情。助手不得到处搜集信息，将随机的报事贴、电子邮件和语音邮件收集在一起，来估计开发者如何支配时间。由于没有时间跟踪系统，公司失去了监督开发者的生产率、识别过程改进的潜力以及规划未来项目的机会。

我建议马田或多或少地实现跟踪开发者工作的乏味过程的自动化。方法是让开发者在屏幕上使用一个小计时器。他们可以激活该计时器，使它与一个活动名称关联；在他们吃午饭或者出现其他中断时挂起，恢复，然后在任务完成时关闭计时器。跟踪数据可以存储在一个安全的地方，然后在一周结束时检索这些数据并在一个电子表格中整理它们。

“好主意！”马田说。“尼克，你能帮我开发它吗？它将为我减少很多麻烦，并能节约大量费用。我会满足你的任何要求。开发这个项目要多少钱？”

我告诉马田还要考虑一下。我下周没有事情，也许能在几个小时内制成某个东西。不行，要用几天吧。我让他下星期一早上 11 点左右到我的办公室来，到时将为他提出某种方案。

星期天：上午-下午

1.1.1 业务案例

在周末的其余时间，我不断地考虑马田的工具，终于在脑海中形成了某种概念，以及一两种可能的实现思路。但这是一个严格的业务方案，不允许有任何不明确的地方。为了使这成为一项有趣的工作，我们需要一个严格的业务案例（Business Case）。还需要考虑能够向马田要多少钱、建立什么，以及需要向它投入多少钱和哪几种资源——大部分是我的时间，也许还要购买一些软件。

在星期一吃过午饭以后，我需要与马田达成一致意见，以继续下面的工作，并使他答应至少要先付给我一定的经费。为了实现此里程碑，必须使他同意我的构想、计划和预算。对于我自己，必须考虑在星期一与马田见面时明确地提出开

^① Watts S.Humphrey, A Discipline for Software Engineering. SEI Series in Software Engineering. Reading: Addison-Wesley, 1995.

应用程序必须保证凭据信息的安全。

安全通信提供了以下两种特性：

- **保密性(privacy)**：保密性用于确保数据的机密性，不能被那些可能安装有网络监视软件的窃听器看到。通常通过加密来提供保密性。
- **完整性(integrity)**：安全通信信道必须确保数据在传输过程中不会有意或无意地被修改。通常通过消息验证码(MAC, Message Authentication Code)来提供完整性。

由于在企业网内部会发生多种形式的信息泄露和违反安全规定的行为，因此将安全通信技术同时应用于防火墙的内部和外部是很重要的。

1.3 各种开发技术

可使用各种不同技术和产品来开发 ASP.NET 应用程序。在应用程序的各层中，需要使用不同的身份验证、授权和安全通信方式来确保实现一种深入防范的安全策略。

图 1.1 总结了各种安全技术以及每种技术所提供的主要身份验证和授权方式。

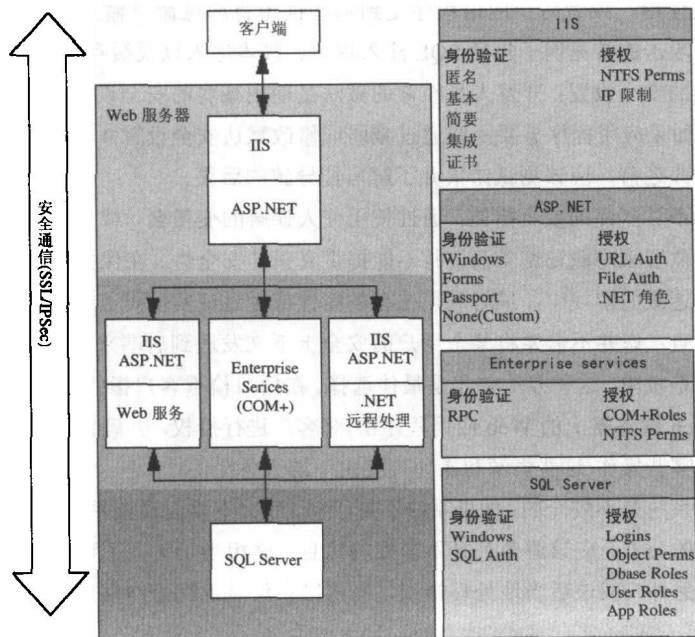


图 1.1 .NET Web 应用程序安全