前　言

1970 年以来，软件生产率一直稳步提高，但是还不足以弥补人们对软件行业的要求和目前状况之间的差距。今天，由于软件成本在计算机系统成本中的比例持续升高，又由于软件缺陷仍然是很多常见故障的原因，因此能够促进软件质量和开发生产率有一个数量级改进的任何措施，都会把软件行业从长期的危机状态中解救出来。基于重用的软件工程自从 20 世纪 60 年代末以来，一直被炒作为“近期”可以提供这种改进的惟一实用和现实的途径。本书将介绍基于重用的软件工程各个方面的最新状况和实践，并努力解释这种所谓“近期”为什么迟迟不能到来的原因，以及如何解决有关的问题。

尽管人们在软件工程和人工智能方面已经开展了几十年的大量研究，但是软件生成仍然是难以达到的目标。当然，进展还是有的，但是目标不断变化，速度甚至越来越快，超过了生成手段可以提供的软件种类和范围。因此，对源于模糊和矛盾的用户需求的软件，只要没有充分地理解建立该软件的过程，就可能遇到成本、可靠性、上市时间、可移植性等诸多因素的冲突。我们应该设法重用已成功的产品或开发过程。基本的假设是：(1) 我们今天所开发的计算机系统具有很多共同点，(2) 通过重用以前开发团队的过程和产品解决新问题，提高生产率并改进最终系统的质量。

重用不仅是自然的，而且可能是进展的关键。心理学家和认知科学家一直认为我们人类很少通过头一个法则就能够解决问题。当面临要解决的问题时，我们首先要进行“机械式的回忆”，看看这个问题是否碰巧以前已经解决过。当不是这种情况时，我们会进行“近似回忆”，希望从已经解决了的问题中，找出非常接近的例子，其解决方法可以局部修改，以便解决新问题。只有不能找到这种解决方案时我们才会回过头来分析问题，至少努力将原始问题分解为一组更具可控性的子问题。有人可能认为，进展的关键是重用（从过去的经验中学习）和沟通（能够

将知识传递给新一代)。

在软件发展的初期，由于要构建的系统规模很小，程序设计语言的表达层次很低，计算机系统硬件和执行时间的成本很高，因此人们感到可以主要从可执行程序的重用中获得效益。软件工程，特别是软件重用的很多研究工作，都集中在软件打包问题上，更具体地说，集中在支持模块化和抽象的语言特性上。在这个研究方向上已经取得了很大进展，但是目标在不断变化。当典型软件的规模和复杂性不断增加时，软件产品和构建产品的团队组织，开始在复杂性、成本和总的影响上支配软件开发过程。这意味着软件重用研究需要关注早于编码的开发阶段，力度超过过程或例程的软件产品的编码、打包和组织的方式。同时，随着有关人员数量的增加，经济问题日益重要，因此需要关注重用的组织和经济问题。

20 世纪 70 年代以来，一些机构认识到软件重用在改进生产率、提高所构建产品质量方面上的潜力。20 世纪 80 年代初，美国联邦政府，特别是国防部，在其软件提供商圈内，进而在整个软件界，发起了有助于理解、组织和推动软件重用的一些活动。20 世纪 70 年代以来，就有一些机构宣称开始进行软件重用，但结果有很大差异，对于那些重用成功的案例，所使用的方法有时是不能重复、不可剪裁的，或者既不能重复也不能剪裁的，重用的效益是不能度量的。今天，人们普遍认识到为了达到企业范围内的有价值、可预测和可重复使用水平，必须在三个层次上解决软件重用问题：

- 开发并打包高质量软件人工制品的技术，这些人工制品是得到广泛应用且经济合算的。
- 针对生产、使用和管理重用产品共享库，组建有要求、有纪律、有技能、有资源的机构。
- 用于规划、控制和评估机构到达其目标程度的成套的控制和管理工具。

本书将讨论所有这三个方面的问题。在所有这三个问题上的全面、深入的研究成果，使我们感到值得撰写本书，也希望本书值得购买，值得阅读。

目前在所有这三个领域中已经有了很多研究，并开始逐渐形成固定的实践知识。但是，这三个方面还没有达到相同的成熟度水平，本书将反映这种现实。最成熟的领域可能要算技术领域。有关重用软件打包技术的研究自从 20 世纪 60 年代末以来，一直都在进行，实践滞后理论的时间在不断缩短，有时甚至反了过来，产生手段和工具的发展，甚至快过研究人员能够将其理论化的速度，但有时也落后于技术发展。我们所面临的挑战就是维持这种态势。在这个领域中，我们通过学习、尝试、咨询、指导和教授各种使重用成为可能的手段，逐渐积累经验。为了使这类知识的寿命比最新时尚的语言、工具、手段或方法长一些，我们必须寻找并努力固定一些用于重用的模型和程序设计基础。通过使用某些技能培养“ 反省专家 ”以便使用下一代技术，如果不开发这些技术的话。我们希望这些材料有助于完善本书所提供的更实用的技能。

软件重用的机构问题自从 20 世纪 80 年代末以来已经引起了越来越多的关注。因为这是一种不同的知识体，并且与更一般的问题有关，例如过程成熟度框架和与过程再工程和变更管理问题。机构模型的来源包括：（1）对业界已知成功重用机构案例的综合，（2）从过程再工程、风险管理和变更管理中提取一般知识，解析地构建这种模型。输出是由公众（例如政府）、超公众（例如“ 软件工程研究所 ”，SEI）和业界财团（例如“ 软件生产率集团 ”，SPC）描绘的一组比较精细的机构模型。这些相对、相似并且有常识基础的模型可以用一本书的篇幅描述。我们决定综合这些知识，重点关注其共性，并同时提供信息源和这些机构模型或模型变种成功使用的已知案例。

软件重用的经济学建模领域可能是最不成熟的领域。经济学建模的核心是重用度量。长期以来，软件度量只限于软件工程的研究人员和专业人员的小圈子，没有成为更主流的研究主题。在实践中，这也是为什么大多数软件机构不是能力成熟度（CMM）一级以上机构的主要原因之一。很少有软件机构度量软件及生产软件的过程，知道如何处理度量数据的机构就更少了。软件度量领域正在升温，主要是在软件质量度量和再工程与维护结合方面。但是，这种升温并不能在近期使软件重用的经济学建模受益，因为在经济学建模中，我们感兴趣的是功能规模度量，

在这个方面软件重用又增加了一层复杂性。与软件重用经济学建模有关的材料是一些教材和理论研究，我们介绍的模型可以作为供重用战略规划人员和经理受益的很有价值的模拟工具。

本书面向的是技术类读者。本书提供的材料与大多数经理没有多少直接的关系。对于急需在第二天就要拿到评审会议上讨论的计划中增加撒手锏的经理，还有其他一些内容和风格都很适合的专著。

本书通过既提供实践和指导材料又提供基础材料，希望对教学人员和实际工作者都有用。本书还提供复习题和练习，主要对学校教育及职业培训提供教学支持。

实际工作者 本书提供的材料对于开发人员及技术经理和领导都很有用。开发人员可以直接从第 3 部分开始，阅读第 7 章和第 8 章，然后进入第 4 部分，其引言对第 10 到 14 章做了详细说明。第 5 部分的第 15 章和第 17 章也是必读的，可以略过第 16 章（组件存储与检索）。感兴趣的读者可以阅读第 7 部分中与自己的机构或项目有关的一章（重用的具体形式）。技术经理应该阅读第 1 到第 3 部分（第 9 章除外）、第 4 部分的最后两章（第 13 章应用框架；第 14 章体系结构框架）以及第 7 部分的所有内容。技术领导（例如体系结构设计师）应该阅读第 1 部分和第 3 部分到第 5 部分（可以略过第 4 部分中的一些基础材料，参见该部分的引言），以及第 7 部分中与自己开发实践关系最密切的章节。

教学工作者 本书适合用做高级软件设计课程的第 6 或第 7 学期教材。我们建议使用第 1 章、第 3 部分的第 6 章和第 7 章、第 4 和第 5 部分的全部，根据学生所具备的一般软件工程背景知识的具体情况，还可以使用第 21 章（基于组件的软件工程）。本书还可以作为计算机科学或管理信息系统（MIS）专业的研究生课程。对于计算机科学专业的学生，可以完全略过第 6 部分，有些对面向对象（OO）技术的介绍材料也可以略过。对于 MIS 专业的学生，可以阅读第 1 和第 2 部分，第 3 部分中除第 9 章以外的所有内容，第 4 部分和第 5 部分的引言（第 10 章和第 15 章），以及第 6 部分和第 7 部分。有些章提供了复习题和练习。练习中的大写字母表示问题的困难程度或复杂程度：（A）最简单，（B）

中等难度，（C）难度最大，（R）研究性问题。

我们希望读者能够通过使用本书提供的材料得到很大乐趣，就像我们在编写本书时得到很大乐趣一样。

Hafedh Mili，Ali Mili，Sherif Yacoub，And Edward Addy

软件工程丛书

基于重用的软件工程

——技术、组织和控制

[美] 摇摇著

韩摇柯摇等译

北京·月摇摇摇

内 容 简 介

摇摇软件重用的理论研究和实践，在软件发达国家已经有了一定的历史。实践证明，软件重用是从根本上提高软件生产率、降低成本、提高质量的有效途径；同时也说明，软件重用需要在关键技术、管理、指标等很多方面引入注重实效的变革。本书讨论了基于重用的软件工程实践模型和其他有关的基础问题；全面介绍了基于重用的软件工程的当前状态和未来；比较全面地讨论了重用的关键技术、管理和组织问题；详细论述了这些理论知识在基于组件的软件开发生命周期和产品线工程中的运用问题。
摇摇本书非常适合计划推进本单位软件重用实践的管理和技术人员阅读。相信本书对高校计算机应用专业研究生和高年级本科生也有很好的参考价值。

悦霸懂驿骤戒© 圆囤圆迅遭憎优霖壮宰篮瓣驭 杂操捏援

粤酋酝酶酥剃酶酶酶酶援粤悦操酶酶酶壮酶酶酶壮酶酶 贼赋霖秋与酶酶操遭与惹酶操酶酶酶壮责遭酶酶操遭迅憎优霖壮宰篮瓣驭 杂操择 陨赋援

本书中文简体专有翻译出版权由美国 优霖壮宰篮瓣驭 杂操捏公司授予电子工业出版社。该专有出版权受法律保护。

版权贸易合同登记号摇图字：园酶圆囤圆囤瓣员豫

图书在版编目 (悦阳) 数据

基于重用的软件工程 技术、组织和控制 辕美) 迈利 (酝篮鲁 勻援 等著 ;韩柯等译 郾–北京 :电子工业出版社,圆囤囤员
(软件工程丛书)
书名原文 :阿奥斯特维兹杂赋赋酶酶赋秋与酶酶操壮与惹酶操酶酶酶壮酶酶 韦赋酶操酶酶壮 酶阅酶韦酶酶酶操壮, 蔡凿悦操阿酶酶择 陨阅星 死酶酶酶酶酶酶酶戒
Ⅰ郾基…摇Ⅱ郾①迈…②韩…摇Ⅲ郾软件工程摇Ⅳ郾味酶酶员豫

中国版本图书馆 悦阳数据核字 (圆囤囤豫 第 园员员豫豫号

责任编辑：赵摇平摇齐摇莉
印　　刷：
出版发行：电子工业出版社
　　北京市海淀区万寿路 员豫信箱摇邮编 员囤囤豫园
经摇销：各地新华书店
开摇本：员远/员远 印摇张：圆园援缘摇字数：远豫豫千字
印摇次：圆囤囤园年 员月第 员次印刷
印摇数：缘园园册摇定价：远豫援园元

摇摇凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系。联系电话：(园员园) 远豫豫员园园园转 质量投诉请发邮件至 扎遭泽岳责藻赠藻择泽遭援泽藻 盗版侵权举报请发邮件至 凿遭择岳责藻赠藻择泽遭援泽藻

译 者 序

软件重用是近年来软件工程一个很活跃的研究领域。实践证明，软件重用是从根本上提高软件生产率、降低成本、提高质量的有效途径；同时也说明，软件重用需要在关键技术、管理、指标等很多方面引入注重实效的变革。虽然有很多机构都宣称正在推行软件重用，但是效果却不是很大，其中有技术方面的原因，但更主要的还是机构和组织管理问题。很明显，要想真正发挥软件重用的效能，取得预期的成功，不认真研究软件重用的理论和总结经验教训是不行的。

本书作者根据大量研究和实践，提出为了达到企业范围内的有价值、可预测和可重复使用的水平，必须在三个层次上解决软件重用问题，即高质量软件产品的成套开发和打包技术，以重用产品共享库为核心的组织机构，提高管理水平的成套控制和管理工具。本书围绕这三个方面的问题，全面、深入地进行了论述，观点明确、层次分明、实用性强，具有很高的参考价值。

在翻译本书过程中，除了对原文个别明显文字错误进行了相应更正外，我们力求忠实原文。但由于我们的知识水平和实际工作经验有限，不当之处在所难免，恳请读者批评指正。参加本书翻译、审校和其他辅助工作的还有：杜旭涛、李宗泽、姜圣国、魏军、吴铃、张红旭、原小铃、李津津、王威、屈健、黄惠菊、韩文臣、朱军、杜蔚轩、解冀海、付程、孟海军、耿民等。

译 者

2003 年 11 月

目　　录

第 1 部分　引　　论

第 1 章　软件重用与软件工程 …… (3)
1.1　概念与术语 …… (4)
1.1.1　软件重用的定义 …… (4)
1.1.2　软件重用：潜能与陷阱 …… (6)
1.1.3　练习 …… (7)
1.2　软件重用产品 …… (8)
1.2.1　可重用资产 …… (8)
1.2.2　重用库：纵向集与横向集 …… (9)
1.2.3　练习 …… (11)
1.3　软件重用过程 …… (12)
1.3.1　组织结构 …… (12)
1.3.2　领域工程 …… (13)
1.3.3　应用工程 …… (14)
1.3.4　企业监督 …… (15)
1.3.5　练习 …… (15)
1.4　软件重用范例 …… (17)
1.4.1　软件检索范例 …… (17)
1.4.2　软件适配范例 …… (18)
1.4.3　软件合成范例 …… (19)
1.4.4　练习 …… (20)
1.5　补充读物 …… (21)
第 2 章　最新理论研究与实践 …… (23)
2.1　软件重用管理 …… (23)
2.1.1　最新研究状况 …… (23)
2.1.2　实践情况 …… (25)

2.1.3 前景 …… (27)
2.1.4 练习 …… (27)
2.2 软件重用技术 …… (28)
2.2.1 最新研究状况 …… (28)
2.2.2 实践情况 …… (31)
2.2.3 前景 …… (33)
2.2.4 练习 …… (34)
2.3 软件重用活动 …… (34)
2.3.1 软件重用库 …… (34)
2.3.2 软件重用方法 …… (37)
2.3.3 软件重用标准 …… (42)
2.3.4 练习 …… (43)
2.4 补充读物 …… (43)
第 3 章 软件重用问题 …… (47)
3.1 组织问题 …… (47)
3.1.1 管理基础设施 …… (47)
3.1.2 技术基础设施 …… (47)
3.1.3 重用的引入 …… (48)
3.1.4 练习 …… (48)
3.2 技术问题 …… (49)
3.2.1 领域工程问题 …… (49)
3.2.2 组件工程问题 …… (50)
3.2.3 应用工程问题 …… (50)
3.2.4 练习 …… (51)
3.3 经济问题 …… (51)
3.3.1 软件重用指标 …… (51)
3.3.2 软件重用成本估计 …… (51)
3.3.3 软件重用投资回报 …… (52)
3.4 补充读物 …… (52)

第 2 部分 组 织 问 题

第 4 章 软件重用组织 …… (55)

4.1 软件重用团队结构 …… (55)
4.1.1 特征特性 …… (55)
4.1.2 软件重用团队结构 …… (57)
4.1.3 确定因素 …… (63)
4.1.4 练习 …… (64)
4.2 重用技能 …… (65)
4.2.1 重用库管理员 …… (65)
4.2.2 重用经理 …… (66)
4.2.3 领域工程师 …… (66)
4.2.4 应用工程师 …… (67)
4.2.5 组件工程师 …… (68)
4.2.6 练习 …… (68)
4.3 补充读物 …… (68)
第 5 章 支持服务 …… (71)
5.1 配置管理 …… (71)
5.2 质量保证 …… (73)
5.3 测试 …… (73)
5.4 检验与确认 …… (74)
5.4.1 领域级任务 …… (77)
5.4.2 符合性任务 …… (78)
5.4.3 沟通结果 …… (79)
5.5 风险管理 …… (79)
5.6 确认 …… (80)
5.7 练习 …… (81)
5.8 补充读物 …… (81)
第 6 章 重用制度化 …… (83)
6.1 机构就绪状态 …… (83)
6.2 重用障碍 …… (88)
6.2.1 文化障碍 …… (89)
6.2.2 管理障碍 …… (90)
6.2.3 技术障碍 …… (90)
6.2.4 基础设施障碍 …… (90)

6.3 克服重用障碍 …… (91)
6.3.1 行政层的支持 …… (91)
6.3.2 培训 …… (91)
6.3.3 激励 …… (91)
6.3.4 渐进方法 …… (92)
6.4 练习 …… (94)
6.5 补充读物 …… (95)

第 3 部分 领域工程：针对重用构建

第 7 章 构建可重用资产：概述 …… (99)
7.1 可重用性 …… (99)
7.1.1 可使用性 …… (100)
7.1.2 有用性 …… (106)
7.2 获取可重用资产 …… (108)
7.2.1 构建与购买 …… (109)
7.2.2 内部构建可重用资产 …… (110)
7.2.3 构建应用系统生成器 …… (112)
7.3 领域工程生命周期 …… (115)
7.3.1 问题 …… (116)
7.3.2 领域工程生命周期的一个样本 …… (119)
7.3.3 小结 …… (122)
7.4 结论 …… (124)
第 8 章 领域分析 …… (127)
8.1 基本概念 …… (128)
8.1.1 领域 …… (128)
8.1.2 领域分析 …… (129)
8.1.3 领域模型 …… (130)
8.1.4 练习 …… (131)
8.2 领域范围 …… (131)
8.2.1 范围评判准则 …… (131)
8.2.2 范围过大与过小 …… (132)
8.2.3 练习 …… (133)

8.3 领域需求与应用需求 …… (133)
8.4 领域组件的剖析 …… (135)
8.4.1 组件族模型 …… (136)
8.4.2 组件族的设计考虑 …… (138)
8.4.3 练习 …… (139)
8.5 抽象与领域分析 …… (139)
8.5.1 抽象与共性分析 …… (139)
8.5.2 抽象维 …… (144)
8.5.3 练习 …… (148)
8.6 领域分析方法 …… (148)
8.6.1 面向功能的领域分析（FODA） …… (148)
8.6.2 机构领域建模（ODM） …… (150)
8.6.3 联合面向对象领域分析（JODA） …… (150)
8.6.4 重用库过程模型（RLPM） …… (151)
8.6.5 领域分析与设计过程（DADP） …… (152)
8.6.6 与具体领域有关的软件体系结构（DSSA） …… (153)
8.6.7 SYNTHESIS 领域分析方法 …… (153)
8.6.8 重用业务方法论 …… (154)
8.6.9 比较 …… (155)
8.6.10 练习 …… (161)
8.7 领域分析工具 …… (161)
8.7.1 KAPTUR，一种基于知识的工具 …… (162)
8.7.2 GENESIS，一种数据库管理系统构建工具 …… (162)
8.7.3 练习 …… (163)
8.8 补充读物 …… (163)
第 9 章 程序设计范例与可重用性 …… (165)
9.1 可使用性属性 …… (165)
9.1.1 可重用性——抽象边界 …… (165)
9.1.2 抽象与可合成性 …… (168)
9.2 表达与建模范例 …… (173)
9.2.1 描述性与过程性表达 …… (173)
9.2.2 面向对象的建模 …… (178)

9.3 开发范例中的抽象与合成 …… (181)
9.3.1 描述性表达：逻辑程序设计案例 …… (181)
9.3.2 过程性（函数）程序设计 …… (185)
9.3.3 面向对象的开发 …… (190)
9.4 向多范例开发发展 …… (198)

第 4 部分 面向对象的领域工程

第 10 章 面向对象实用介绍 …… (203)
10.1 引言 …… (204)
10.1.1 概述 …… (204)
10.1.2 财务领域 …… (205)
10.2 面向对象程序设计的准则 …… (206)
10.2.1 一段样本程序 …… (206)
10.2.2 封装与信息隐藏 …… (208)
10.2.3 重载与一般化 …… (213)
10.2.4 子类型与类继承 …… (220)
10.2.5 方法解析、晚捆绑与多态性 …… (232)
10.3 语言大战 …… (238)
10.4 讨论 …… (244)
第 11 章 面向对象中的抽象与参数化技术 …… (245)
11.1 面向对象建模中的抽象手段 …… (247)
11.1.1 对象的本体 …… (247)
11.1.2 知识表示中的继承 …… (251)
11.1.3 面向对象分析中的继承 …… (253)
11.1.4 元建模 …… (255)
11.1.5 理解元建模 …… (262)
11.1.6 实践做法——小结 …… (266)
11.2 面向对象程序设计语言中的抽象手段 …… (272)
11.2.1 抽象类 …… (272)
11.2.2 一般类 …… (277)
11.3 元程序设计 …… (280)
11.3.1 构建一种解释器 …… (281)

11.3.2 计算反射与元类 …… (284)
11.3.3 实现强类型 …… (290)
11.4 设计模式 …… (292)
11.4.1 结构抽象模式 …… (293)
11.4.2 行为抽象模式 …… (297)
11.5 小结 …… (300)
第 12 章 面向对象中的合成技术 …… (303)
12.1 问题 …… (304)
12.1.1 可合成性需求 …… (304)
12.1.2 结构与行为合成 …… (305)
12.1.3 抽象与粒度 …… (307)
12.1.4 捆绑时间 …… (308)
12.2 语言方法 …… (309)
12.2.1 约束与基于逻辑的程序设计 …… (309)
12.2.2 函数合成 …… (312)
12.2.3 面向对象与描述混合方法 …… (316)
12.3 模块化与打包方法 …… (321)
12.3.1 面向特性的程序设计 …… (321)
12.3.2 面向主题的程序设计 …… (324)
12.3.3 面向视图的程序设计 …… (328)
12.3.4 其他方法 …… (332)
12.4 基于设计的方法 …… (335)
12.4.1 基于事件的合成 …… (335)
12.4.2 假反射 …… (340)
12.4.3 合成设计模式 …… (347)
12.5 小结与讨论 …… (352)
第 13 章 应用框架 …… (355)
13.1 框架中有什么内容 …… (356)
13.1.1 一种初步定义 …… (356)
13.1.2 框架分析 …… (357)
13.1.3 框架重用生命周期 …… (362)
13.2 完成框架合同 …… (364)

13.2.1 组件可替换性 …… (365)
13.2.2 合成问题 …… (372)
13.3 构建框架 …… (381)
13.3.1 框架作为领域工程产品 …… (381)
13.3.2 框架作为应用系统开发的有计划的副产品 …… (383)
13.4 SWING 框架 …… (385)
13.4.1 综述 …… (385)
13.4.2 事件处理框架 …… (387)
13.4.3 可接插外观感觉框架 …… (391)
13.5 小结 …… (393)
第 14 章 体系结构框架 …… (395)
14.1 什么是体系结构 …… (396)
14.1.1 定义 …… (396)
14.1.2 体系结构的质量属性 …… (399)
14.1.3 体系结构风格与连接器 …… (401)
14.2 体系结构与重用 …… (407)
14.2.1 软件体系结构的开发生命周期 …… (408)
14.2.2 可重用性的方向维 …… (411)
14.2.3 体系结构框架中的问题 …… (413)
14.3 CORBA …… (414)
14.3.1 问题 …… (414)
14.3.2 核心体系结构 …… (417)
14.3.3 处理方法调用 …… (421)
14.3.4 实现应用系统对象 …… (424)
14.4 基于 Java 的技术 …… (427)
14.4.1 Java RMI …… (428)
14.4.2 EJB 体系结构 …… (432)
14.5 COM 族 …… (439)
14.6 小结与讨论 …… (446)
14.7 补充读物 …… (447)

第 5 部分　应 用 工 程

第 15 章　应用工程 …… (451)
15.1　应用工程范例 …… (451)
15.2　应用工程生命周期 …… (453)
15.3　应用工程开发任务与探索 …… (456)
第 16 章　组件存储与检索 …… (461)
16.1　软件库介绍 …… (461)
16.1.1　存储与检索术语 …… (461)
16.1.2　评估准则 …… (464)
16.1.3　存储和检索方法的特征描述 …… (467)
16.1.4　练习 …… (470)
16.2　针对存储和检索分类软件资产 …… (471)
16.2.1　软件资产分类的障碍 …… (471)
16.2.2　软件存储和检索问题 …… (473)
16.2.3　分类软件库 …… (474)
16.2.4　练习 …… (475)
16.3　补充读物 …… (476)
第 17 章　可重用资产集成 …… (481)
17.1　资产实例化范例 …… (482)
17.1.1　组件选择 …… (483)
17.1.2　组件生成 …… (483)
17.1.3　组件规格说明 …… (483)
17.2　资产合成范例 …… (484)
17.2.1　可合成性情景 …… (485)
17.2.2　可合成性媒介 …… (488)
17.3　集成组件问题 …… (489)
17.3.1　组件问题 …… (489)
17.3.2　过程问题 …… (490)
17.3.3　质量问题 …… (492)