

Microsoft Press

软件需求

2

(第2版)

Software Requirements



(美) Karl E. Wiegars 著

刘伟琴 刘洪涛 译



清华大学出版社

软件需求(第2版)

(美) Karl E. Wieggers 著

刘伟琴 刘洪涛 译

清华大学出版社

北 京

内 容 简 介

本书是有关软件需求的经典教材,本书全面而深入地讲述了软件开发中一个至关重要的问题——软件需求问题。软件开发人员及用户往往容易忽略沟通的重要性,导致软件开发出来后,不能很好地满足用户的需要。返工不仅在技术上给开发人员带来巨大的麻烦,并且会造成人力、物力和资源的浪费,还使软件性能深受影响,所以在开发早期提高项目需求分析的质量,减少重复劳动,通过控制项目范围的扩大及需求变更来达到按计划完成预定目标,是当前软件业急需解决的问题,也是本书讨论的主要内容。

本书对第1版的内容进行了扩展,不仅对原有的知识点进行了补充,还引入了一些新知识,以求与时代发展同步。

本书可以作为计算机专业及软件工程专业学生的教材使用,也非常适合作为项目经理、软件开发人员的指导性参考书。

软件需求(第2版)

Software Requirements(second edition)(ISBN 0-7356-1879-8)

Karl E. Wiegerts

Copyright © 2003 by Microsoft Corporation.

Original English Language Edition Copyright © 2003 by Microsoft Corporation.

Published by arrangement with the original publisher, Microsoft Press,
a division of Microsoft Corporation, Redmond, Washington, U.S.A.

本书中文版由 Microsoft Press 授权清华大学出版社出版。

北京市版权局著作权合同登记号 图字:01-2003-3097

版权所有,翻印必究。举报电话:010-62782989 13901104297 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用清华大学核研院专有核径迹膜防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

软件需求(第2版)/(美)威格斯(Wiegerts,K.E.)著;刘伟琴,刘洪涛译. —北京:清华大学出版社,2004.11
书名原文:Software Requirements (second edition)

ISBN 7-302-09834-4

I. 软… II. ①威…②刘…③刘… III. 软件开发 IV. TP311.52

中国版本图书馆CIP数据核字(2004)第111917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:李朋朋 张 莉

封面设计:陈刘源

印 刷 者:北京牛山世兴印刷厂

装 订 者:北京市密云县京文制本装订厂

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:23.25 字数:555千字

版 次:2004年11月第1版 2004年11月第1次印刷

书 号:ISBN 7-302-09834-4/TP·6782

印 数:1~4500

定 价:39.00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

译者序

随着计算机软件项目的规模越来越大,竞争日趋激烈,软件开发组织越来越认识到软件质量的重要性,在这种情况下软件工程的观念已渐渐深入人心,人们已经从中受益。

软件需求作为软件工程的一个阶段,在软件项目开发中起着至关重要的作用。软件项目要取得成功,最重要的莫过于了解所要开发的软件需要解决哪些问题,这就是软件需求所要解决的问题,因此,软件需求为软件项目的成功奠定了基础。如果软件开发人员与客户不进行充分的交流与沟通,没有就产品的功能性需求和非功能性需求达成共识,就匆匆忙忙开始着手编写代码,其后果可想而知,很可能不能满足用户的需要,从而不得不对项目进行返工,这就造成了人力和物力的巨大浪费。如果我们在软件项目开发之前,充分地完成软件需求的相关活动,就可以避免这种情况的发生。

本书是一本非常实用的需求工程参考书,书中按照需求工程的各个阶段,即需求获取阶段、需求分析阶段、编写需求规格说明阶段、需求确认阶段和需求管理阶段组织起来,并提供了许多有效技术,这些技术为用户、开发人员和管理人员之间进行交流提供了方便。本书作者卡尔·E·威格(Karl E. Wiegers)是需求工程领域的权威人士,他曾担任过软件开发人员、软件经理以及软件过程和质量改进负责人,在长期的工作中积累了丰富的经验。本书第1版曾荣获“软件开发”效率大奖,目前已成为参与软件开发过程的所有人员必不可少的参考书。本书第2版对第1版中所提出的最佳实践进行了许多扩充,这一版不仅在每一章中都列举了大量的实例并提供了新的案例,而且,作者还根据自己的亲身经历,为完成不同的任务提供了颇具特色的检查列表、范例文档和模板。另外,作者还从自己丰富的职业生涯中精选出了一些趣闻轶事,增加了技术书籍的趣味性。相信阅读本书之后,读者对于需求工程一定会有一个全面而透彻的理解。

参加本书翻译工作的人员还有苏正泉、米强、张颖、夏红、谷昀、江峰、徐利生、李宏为、赵琪、姬凌岩。

由于时间仓促以及水平有限,错误之处在所难免,敬请读者批评指正。

译者

2004年4月

前言

尽管软件业已经经历了 50 年的发展历程，但是许多软件开发组织仍然在努力收集它们的产品需求，将这些需求编写成文档并进行管理。许多信息技术项目之所以不能在预定的进度和预算范围内向用户交付预期的所有功能，其主要原因不外乎就是缺少来自用户的信息，需求不够完整及需求发生了变化。许多软件开发人员都感到收集客户需求并不是一件轻而易举的事情，或者说他们并不精通于此。客户经常没有耐心参与需求开发，或者是指派不合适的人选来提供需求信息。项目参与者甚至经常不能就“到底是什么是需求”达成一致意见。正如一位作者所观察到的：“工程师们宁肯解读 Kingsmen 乐队在 1963 年演唱的经典歌曲 ‘Louie Louie’ 中的歌词，也不愿意解读客户的需求”。

软件开发所涉及的人员沟通至少与计算一样多，但是，我们却经常只强调计算而忽略了沟通。本书提供的许多工具为沟通提供了方便，并可以帮助软件从业人员、管理人员、市场人员和客户应用有效的需求工程方法。第 2 版新增加的章节内容包括需求分析员的角色、业务规则的重要性，以及将需求工程应用到维护性项目、软件包解决方案、外包项目和渐进式项目中所采用的方法。书中大量的特殊段落中所提供的材料都是真实的，旨在阐明与需求相关的典型经历，这些段落用图标  标示。

这里所介绍的这些技术是需求工程中主流的“良好实践”，而不是陌生的新技术，也不是声称可以解决所有需求问题的详尽方法学。我于 1999 年编写了本书的第 1 版，此后在 100 多个研讨会中为各类公司和政府组织中的人员讲授过软件需求。我已经了解到这些做法实际应用于各种项目，包括那些后续追加服务的方法，无论是小型项目还是大规模的项目，开发新项目还是维护已有项目均可适用。另外，这些技术也并不局限于软件项目，它们也同样适用于硬件和系统工程。与任何一种其他的软件实践一样，我们需要根据常识和经验来搞清楚如何使这些方法更好地为我们服务。

从本书中获得的收益

在您所能采用的所有软件过程改进中，改进的需求工程实践为我们所带来的好处很可能是最大的。我重点描述了实用的并已得到认可的技术，这些技术可以在下列方面对我们有所帮助：

- 在开发周期早期改进项目需求的质量，可以减少返工和提高生产率。
- 通过控制范围扩大和需求变更来满足项目的进度目标。
- 达到更高的客户满意度。
- 降低维护成本和技术支持的成本。

我的目的是帮助大家改进收集并分析需求、编写并确认需求规格说明、在整个产品开发周期中管理需求等几个方面所采用的过程。我希望大家真正地将改进的实践用于具

体的项目中，而不要只是读一读而已。了解新的实践并不难，然而，事实上要改变人们的工作方法却不是一件容易的事。

本书读者对象

需要定义或理解软件产品需求的所有人员都会从本书中获得对自己有用的信息。第1类对象是在项目开发中承担需求分析员角色的人，他们可能是专职的需求分析专家，也可能只是临时承担需求分析员的角色。第2类对象包括设计人员、程序员、测试人员以及其他必须理解并满足用户要求的团队成员。负责指定使产品在商业上获得成功的特性和属性的市场人员和产品经理也会发现这些实践十分具有价值。必须按时交付产品的项目经理也可以通过本书了解到如何管理项目需求活动和处理需求变更。第3类对象是客户——他们希望自己定义的产品能够满足功能和质量的需要。本书将帮助客户理解需求过程的重要性以及他们在这一过程中所扮演的角色。

全书展望

本书共分4个部分。第I部分“什么是软件需求？为什么要实现软件需求？哪些人应参与软件需求”，包括第1章至第4章，这一部分一开始提出了一些定义，并描述了优秀需求具备的若干特性。如果你们是负责技术的一方，那么我希望你们能与重要客户共同阅读第2章中有关客户和开发人员合作伙伴关系这一部分。第3章介绍了业界需求开发和管理的几十个“良好实践”，以及需求开发的总体过程。第4章的主题是需求分析员的角色。

第II部分“软件需求开发”包括第5章至第17章。这一部分首先介绍了定义项目的业务需求所采用的方法。其他章节描述了如何找到合适的客户代表，获取他们的需求，以及将用例、业务规则、功能性需求和质量属性编写成文档。第11章描述了几种分析模型，这些分析模型可以从不同的角度来表示需求。第13章描述了如何使用软件原型来减小风险。其余的章节提出了划分需求优先级和确认需求的各种方法。在这一部分的最后，描述了在某些特殊的项目情况下需求开发所面临的特殊难题，并研究了需求如何影响项目工作的其他方面。

第III部分“软件需求管理”包括18章至21章，这一部分的主题是需求管理的理论和实践，重点强调处理需求变更所用的方法。第20章描述了如何通过需求可跟踪性将单个需求与它们的起源、下游开发的可交付成果联系起来。这一部分的最后介绍了几种商业工具，这些工具能够进一步改进管理项目需求使用的方法。

本书的最后一部分“实现需求工程”包括第22章至附录D，帮助我们将理论概念运用到具体实践中。第22章将新的需求工程技术加入到项目组的开发过程中。第23章描述了与需求相关的一些常见的项目风险。附录A中的当前需求实践自我评估能够帮助我们选择最合适的需求基数和方法。其他附录介绍了需求和过程改进模型、需求错误

诊断指南和几个需求文档范例。

案 例 研 究

为了演示本书所描述的方法，我们从基于实际项目的若干案例研究中选择了一些实例，特别提供了一个称为“化学制品跟踪系统”的中等规模的信息系统(不要着急——理解这一项目并不需要我们了解任何化学知识)。遍布于整本书中的项目参与者之间的样例对话均来自于这些案例研究项目。无论你所在的项目组构建何种软件，我认为你都可以找到相关的对话。

从理论到实践

扫清将新的知识付诸于行动所遇到的障碍需要相当的精力，要做到这一点绝非易事。停留在已经熟知的实践领域会比较轻松，人们都不喜欢变化。为了在改进需求的过程中助你一臂之力，每一章的最后都提供了“下一步”，你可以遵循这些步骤着手应用相应章节的内容。书中提供了带有注释的需求文档模板、审查列表、需求优先级电子数据表、变更控制过程和许多其他的过程资产，可以通过 <http://www.processimpact.com> 下载。通过这些材料，我们就可以开始着手应用这些技术。最初可以先实施一些小的需求改进，但不要拖延，要马上行动。

有些项目参与者不愿意尝试新的需求技术。有些人则完全不切实际，如果你正在与这样的人打交道，那么这些技术都不能发挥作用。使用本书中所提供的材料来教育你的同行、客户和经理，提醒他们在以前的项目中所遭遇到的与需求相关的一些问题，并讨论如果尝试某些新方法会收到哪些潜在的收益。

我们不必在一个全新的开发项目中开始应用改进的需求工程实践。第 16 章讨论了几种方法，可以将本书中所介绍的许多技术应用到维护性项目中。增量地实现需求实践也是一种低风险的过程改进方法，这可以为在下一个较大的项目中采用一整套新技术作好准备。

需求工程的目的是要开发优秀的需求，使得我们能够在在一个可以接受的风险级别上继续进行设计和构造工作。在需求工程上需要我们花费足够的时间，以尽量减少返工、产品不被接受和拖延进度等风险。本书还提供了一些工具，便于相关人员协同工作，为合适的产品开发合适的需求。

目 录

第 I 部分	
什么是软件需求?	
为什么要实现软件需求?	
哪些人应参与软件需求	
第 1 章 软件需求基础知识	3
1.1 软件需求的定义	5
1.1.1 对需求的不同解释	5
1.1.2 需求的层次	6
1.1.3 不属于需求的内容	8
1.2 需求的开发与管理	9
1.2.1 需求开发	9
1.2.2 需求管理	10
1.3 所有项目都有需求	10
1.4 优秀的团队遇到糟糕的需求	11
1.4.1 用户参与不足	12
1.4.2 用户需求扩展	12
1.4.3 有歧义的需求	12
1.4.4 镀金问题	13
1.4.5 过于抽象的需求	13
1.4.6 忽略了某类用户	13
1.4.7 不准确的计划	13
1.5 优质需求过程的好处	14
1.6 优秀需求的特点	15
1.6.1 需求陈述的特点	15
1.6.2 需求规格说明的特点	16
第 2 章 客户眼中的需求	18
2.1 客户	19
2.2 客户与开发人员的合作伙伴关系	20
2.2.1 软件客户的权利法案	22
2.2.2 软件客户的义务法案	23
2.3 关于“签字”	25
第 3 章 需求工程的推荐方法	28
3.1 知识技能	29
3.2 需求获取	30
3.3 需求分析	32
3.4 规格说明	33
3.5 需求验证	34
3.6 需求管理	35
3.7 项目管理	36
3.8 开始新实践	37
3.9 需求开发过程	38
第 4 章 需求分析员	41
4.1 需求分析员的职责	41
4.1.1 需求分析员的工作	42
4.1.2 需求分析员必备的技能	44
4.1.3 需求分析员必备的知识	45
4.2 如何培养需求分析员	46
4.2.1 从用户转为分析员	46
4.2.2 从开发人员转为分析员	47
4.2.3 主题专家	47
4.3 营造合作的氛围	47
第 II 部分	
软件需求开发	
第 5 章 确定产品前景与项目范围	51
5.1 通过业务需求定义前景	51
5.1.1 相互矛盾的业务需求	52
5.1.2 业务需求与用例	53
5.2 前景与范围文档	53
5.3 关联图	59
5.4 保持范围的适度	60
第 6 章 获取客户的需求	62
6.1 需求的来源	62

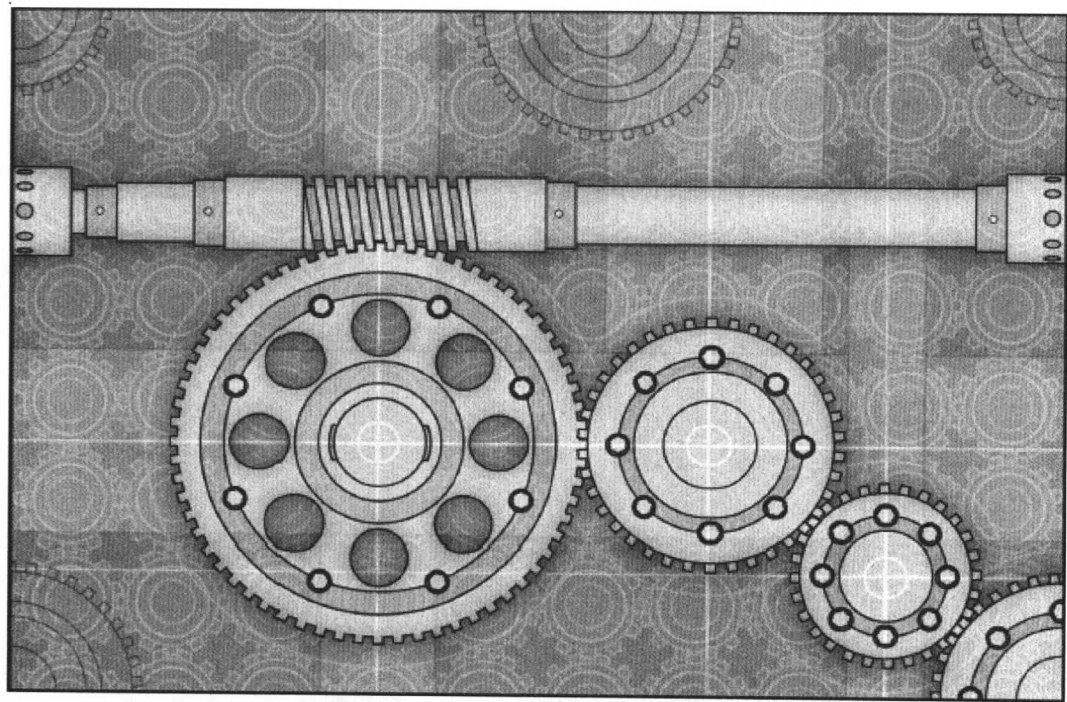
6.2 用户类	63	第 10 章 编写需求文档	112
6.3 寻找用户代表	66	10.1 软件需求规格说明	112
6.4 用户代言人	67	10.1.1 需求的标识	113
6.4.1 外部的用户代言人	68	10.1.2 处理不完整性	115
6.4.2 对用户代言人的要求	69	10.1.3 用户界面和软件 需求规格说明	115
6.4.3 设置多位用户代言人	70	10.2 软件需求规格说明模板	116
6.4.4 如何让人接受用户 代言人的概念	71	10.3 编写需求文档的原则	123
6.4.5 用户代言人应避免 的陷阱	72	10.4 改进前后的需求示例	126
6.5 谁来做出决策	72	10.5 数据字典	129
第 7 章 聆听客户的需求	75	第 11 章 一图胜千言	132
7.1 需求获取	76	11.1 需求建模	132
7.2 需求获取讨论会	77	11.2 从客户需求到分析模型	133
7.3 将客户的意见归类	79	11.3 数据流图	134
7.4 需求获取中的注意事项	83	11.4 实体—关系图	137
7.5 寻找遗漏的需求	83	11.5 状态转换图	139
7.6 如何判断需求获取是否已完成	85	11.6 对话图	142
第 8 章 理解用户需求	87	11.7 类图	145
8.1 用例法	88	11.8 判定表和判定树	146
8.1.1 用例与使用场景	88	11.9 最后的提醒	148
8.1.2 确定用例	92	第 12 章 软件质量属性	149
8.1.3 编写用例	92	12.1 质量属性	150
8.1.4 用例与功能性需求	98	12.2 定义质量属性	151
8.1.5 用例的好处	99	12.2.1 对用户重要的属性	152
8.1.6 使用用例时应避免的问题	100	12.2.2 对开发人员重要的属性	155
8.2 事件—响应表	101	12.3 性能需求	157
第 9 章 遵守规则	104	12.4 用 Planguage 定义非功能性需求	158
9.1 业务的规则	105	12.5 属性的折中方案	159
9.1.1 事实	105	12.6 实现非功能性需求	160
9.1.2 约束	106	第 13 章 通过制作原型减少 项目风险	162
9.1.3 动作触发规则	106	13.1 什么是原型和为什么要 建立原型	162
9.1.4 推论	107	13.2 水平原型	163
9.1.5 计算	107	13.3 垂直原型	164
9.2 在文档中记录业务规则	108	13.4 废弃型原型	164
9.3 业务规则和需求	109		

13.5 演化型原型	165	第 17 章 超越需求开发	209
13.6 书面原型和电子原型	167	17.1 从需求到项目规划	210
13.7 原型评估	168	17.1.1 需求和预估	211
13.8 创建原型所带来的风险	169	17.1.2 需求和进度安排	212
13.9 原型法成功的因素	170	17.2 从需求到设计和编码	213
第 14 章 设定需求优先级	172	17.3 从需求到测试	216
14.1 为什么要设定需求优先级	172	17.4 从需求到成功	217
14.2 优先级规则	173	第 III 部分	
14.3 优先级的等级	174	软件需求管理	
14.4 根据价值、成本和风险来 设定优先级	175	第 18 章 需求管理的原则和实践	221
第 15 章 需求确认	181	18.1 需求基线	222
15.1 需求评审	183	18.2 需求管理过程	222
15.1.1 审查过程	184	18.3 需求版本控制	223
15.1.2 需求评审面临的 困难	190	18.4 需求属性	225
15.2 测试需求	192	18.5 跟踪需求状态	226
15.3 制定验收标准	196	18.6 评估需求管理的工作量	228
第 16 章 需求开发面临的特殊难题	198	第 19 章 变更管理	230
16.1 维护项目的需求	198	19.1 管理范围蔓延	231
16.1.1 开始捕获信息	198	19.2 变更控制过程	232
16.1.2 亲身实践一下新的 需求技术	200	19.2.1 变更控制策略	233
16.1.3 遵循跟踪链	201	19.2.2 变更控制过程描述	233
16.2 软件包解决方案的需求	202	19.3 变更控制委员会	237
16.2.1 开发用例	202	19.3.1 CCB 的组成	238
16.2.2 考虑业务规则	203	19.3.2 CCB 规章	239
16.2.3 定义质量需求	203	19.4 变更控制工具	240
16.3 外包项目的需求	204	19.5 测量变更活动	240
16.4 突发型项目的需求	205	19.6 变更需要付出代价：影响分析	242
16.4.1 非正式用户需求 规格说明	206	19.6.1 影响分析的过程	242
16.4.2 现场客户	206	19.6.2 影响分析报告模板	245
16.4.3 尽早地而且要经常地 设定优先级	207	第 20 章 需求链中的联系链	247
16.4.4 简单的变更管理	207	20.1 需求跟踪	247
		20.2 需求跟踪动机	249
		20.3 需求跟踪矩阵	251
		20.4 需求跟踪工具	254
		20.5 需求跟踪过程	255
		20.6 需求跟踪可行吗？必要吗？	255

第 21 章 需求管理工具	257
21.1 使用需求管理工具的益处.....	259
21.2 需求管理工具的功能.....	260
21.3 实现需求管理自动化.....	262
21.3.1 选择适当的工具.....	262
21.3.2 改变文化.....	263
21.3.3 使需求管理工具 服务于自己.....	264
第 IV 部分	
实现需求工程	
第 22 章 改进需求过程	269
22.1 需求与其他项目过程的联系.....	269
22.2 需求和各涉众组.....	271
22.3 软件过程改进的基本原则.....	272
22.4 过程改进周期.....	274
22.4.1 评估当前采用的方法.....	274
22.4.2 规划改进活动.....	275
22.4.3 建立、实验并实现 新过程.....	276
22.4.4 评估结果.....	277
22.5 需求工程过程资产.....	278
22.5.1 需求开发过程资产.....	279
22.5.2 需求管理过程资产.....	280
22.6 需求过程改进路线图.....	281
第 23 章 软件需求与风险管理	283
23.1 软件风险管理基本原理.....	284
23.1.1 风险管理的要素.....	284
23.1.2 编写项目风险文档.....	285
23.1.3 制定风险管理计划.....	287
23.2 与需求相关的风险.....	288
23.2.1 需求获取.....	288
23.2.2 需求分析.....	289
23.2.3 编写需求规格说明.....	290
23.2.4 需求确认.....	290
23.2.5 需求管理.....	291
23.3 风险管理是我们的好帮手.....	291
附录 A 当前需求实践的自我评估	293
附录 B 需求和过程改进模型	298
B.1 软件能力成熟度模型.....	298
B.2 CMMI-SE/SW.....	300
B.2.1 需求管理过程域.....	302
B.2.2 需求开发过程域.....	302
附录 C 需求错误诊断指南	304
C.1 根本原因分析.....	304
C.2 需求问题的常见现象.....	305
C.3 实现解决方案常常会遇到的障碍.....	306
附录 D 需求文档范例	323
D.1 前景和范围文档.....	323
D.1.1 业务需求.....	323
D.1.2 解决方案的前景.....	325
D.1.3 范围和局限性.....	325
D.1.4 业务上下文.....	326
D.2 用例.....	328
D.3 软件需求规格说明.....	336
D.3.1 介绍.....	336
D.3.2 总体描述.....	337
D.3.3 系统特性.....	339
D.3.4 外部接口需求.....	342
D.3.5 其他非功能性需求.....	343
D.3.6 附录 A 数据字典和 数据模型.....	343
D.3.7 附录 B 分析模型.....	346
D.4 业务规则.....	347
术语表	349
结语	358

第 I 部分

什么是软件需求？为什么要实现软件需求？
哪些人应参与软件需求



第 1 章 软件需求基础知识

“您好。是 Phil 么？我是人力资源部的 Maria。我们使用您做的人事管理系统时遇到点问题。有位女职员想把名字改成 Sparkle Starlight，可我们在系统里怎么都改不过来。能帮帮忙吗？”

“她嫁了一个姓 Starlight 的人么？”

“没有，她没结婚，只是改了名字。” Maria 答道，“所以才有这样的麻烦。好像只有在婚姻状况改变时才能改名字。”

“是的。我从来没想到谁会无缘无故地改名字。我们讨论系统的时候您可没跟我提过这种可能。所以只能从修改婚姻状况的对话框进入修改名字的对话框。”

“谁都可以改名字。只要他愿意，随时都行，这是合法的。我以为您知道呢。” Maria 说，“星期五之前必须搞定，不然 Sparkle 就兑换不了支票。您能在那之前把这个错误改过来么？”

“这根本就不是错误！” Phil 反驳道，“我从来不知道您需要这个功能。我正忙着做一个新的性能评估系统。而且我还要对人事管理系统进行一些修改，”（话筒里传来翻纸的声音），“对，这就有一个。月底没准能改好，这周肯定不行，抱歉。下回早点儿告诉我，麻烦把问题写下来。”

“我怎么跟 Sparkle 说？” Maria 问，“兑不了支票她就得赔账。”

“搞清楚，Maria，这可不是我的错。” Phil 抗议了，“如果您当时告诉我要能随时修改姓名，就不会有今天的事。您不能怪我没猜到您的想法。”

Maria 很生气却无可奈何，只好气冲冲地说：“好了。就是这种事让我恨透了计算机。改好了马上通知我，这总可以吧。”

如果您曾经有过这种客户经历，您肯定明白这种连最基本的操作都完不成的软件多么让人烦恼。即便开发人员最终可能会帮您改好，您通常也不愿总求助于他。然而，站在开发人员的立场，如果系统完成后才从用户那里得知需要什么功能，也的确很难接受。已经完全按最初的要求实现了系统，却不得不停下手头的项目去修改系统以便满足用户的新需求，这也是件很讨厌的事。

许多软件问题都源于收集、记录、协商和修改产品需求过程中的方式不当。前面 Phil 和 Maria 的例子中就有这些方面的问题，包括信息收集方式不正规，没有明确提出想要的功能，假设是未经沟通的错误假设，需求的定义不够充分，以及未经仔细考虑进行需求变更等。

很少有人会甩给建筑商 30 万美金而不详细说明自己对房子的想法和要求。相反，他们会不厌其烦地提出各种细节要求。要对房子进行改造就得掏钱，购房者尽管不情愿，

却都能理解。然而,在软件开发中遇到同样问题时,人们却常常轻率地将其忽略。软件项目中40%~60%的缺陷都是由需求分析阶段的过失所致(Davis 1993; Leffingwell 1997)。对欧洲软件行业所做的大规模调查显示:确定和管理用户需求是问题最多的两个环节(ESPITI 1995)。尽管如此,许多组织仍然没有采取有效手段来实施这两个必要的项目活动。由此导致的结果常常是用户和开发者之间产生需求的鸿沟——二者对产品需求的理解相去甚远。

软件或系统项目涉众(stake holder, 产品或项目相关人员)的利益之间的相互作用在需求过程中表现得最为强烈。项目涉众包括:

- 客户:为达到其公司的业务目标而投资项目或购买产品。
- 用户:直接或间接与产品打交道,是客户的一部分。
- 需求分析员:负责编写需求并传达给开发团队。
- 开发人员:设计、实现和维护产品。
- 测试人员:确定产品的行为是否与预计的相一致。
- 文档编制人员:负责编写用户手册、培训资料和系统帮助。
- 项目经理:制定项目计划并带领开发人员获得成功。
- 法律人员:确保产品符合所有相关法规。
- 生产人员:制造包含该软件的产品。
- 市场营销、技术支持及其他与产品和客户打交道的人员。

如果处理得当,各方利益的相互作用将能够使产品获得成功,同时使客户感到满意,并使开发人员充满成就感;否则,就会导致误解、挫折和矛盾,从而降低产品的质量和商业价值。由于需求是软件开发和项目管理活动的基础,所以涉众必须承诺遵循有效的需求过程。

但是开发和管理需求绝非易事,没有任何捷径与魔法。由于很多组织被一些同样的问题所困扰,所以我们可以寻找共同的解决方法,以用于多种不同的情况。本书介绍了很多这类方法。虽然都是以新系统的开发为背景引入这些方法,但其中大部分其实也适用于项目维护和选用现成的商业软件(参见第16章)。这些需求开发及管理方法并非只适用于采用顺序式瀑布型开发生命周期的项目,即使采用增量开发模式的项目开发小组也需要知道每次该增加什么功能。

本章将帮助您:

- 理解软件需求工程的一些重要术语。
- 区分需求开发与需求管理。
- 保持对潜在的与需求相关的问题的警觉性。
- 了解完善的需求应该具备的特征。

为自己把脉

只要对照下面各项来检查最新的项目,就能对组织的项目需求状况作出快速评估。如果项目中有3项与以下情况符合,那就读读这本书吧。

- 项目的前景(vision)和范围(scope)未曾明确定义。
- 客户太忙, 没时间与需求分析员和开发人员一起讨论需求。
- 用户代理, 如生产经理、开发经理、用户负责人或营销人员, 自诩可以代表用户, 其实他们不能准确说出用户的需要。
- 需求只存在于组织中那些所谓专家的脑子里, 没有被记录下来。
- 客户坚持所有需求都很重要, 不愿排出它们的优先次序。
- 开发人员在编码过程中发现需求有歧义, 缺少足够的信息, 只能去猜测。
- 开发人员与客户沟通时只关心用户界面, 忽略了用户需要用软件去做什么。
- 客户签字确认了需求却又一直提出修改要求。
- 项目范围因接受需求变更而扩大, 却没有相应地增加投入或减裁功能, 进度因此被延误。
- 需求变更的请求被弄丢, 开发人员和客户都不了解所有变更请求的状态。
- 开发人员按客户要求实现的功能无人问津。
- 需求规格说明(SRS)中的要求都实现了, 客户却不满意。

1.1 软件需求的定义

软件行业存在这样一个问题, 用于描述需求工作的术语没有统一的定义。对同一项需求, 不同的人会有不同的描述, 称其为用户需求、软件需求、功能需求、系统需求、技术需求、业务需求或产品需求。客户对需求的定义, 在开发人员看来可能只是高级别的产品概念; 而开发人员的需求概念对用户来说也许就是详细的用户界面设计。定义的多样性导致了令人迷惑和沮丧的沟通问题。

需求必须被记录成文档, 这一点很重要。我曾在一个项目中经历过开发人员的角色轮换。每次轮换后, 新任需求分析员都跑来对客户说: “我们来谈谈需求吧。” 客户不胜其烦, 回答也变得很不客气: “我早就把需求告诉您那些前任了, 赶紧把系统给我做出来!” 实际情况是没有谁把需求写下来, 所以每位新任分析员都只能从头开始。仅凭一堆电子邮件、语音邮件、便条、会议记录, 以及对走廊中几次交谈的模糊印象就自称掌握了需求, 那纯属自欺欺人。

1.1.1 对需求的不同解释

咨询专家 Brian Lawrence 提出, 需求是“任何促成设计决策的因素”。很多信息都属于这一范围。

IEEE 的软件工程标准术语表(1990)则将需求定义为:

- 用户为解决某个问题或达到某个目标而需具备的条件或能力。
- 系统或系统组件为符合合同、标准、规范或其他正式文档而必须满足的条件或必须具备的能力。
- 上述第一项或第二项中定义的条件和能力的文档表达。

这一定义既体现了用户对需求的看法(系统的外部行为)，也代表了开发人员的观点(一些深层的特性)。术语用户隶属于涉众，因为并非所有涉众都是用户。我对需求的理解是：产品为向涉众提供价值而必须具备的特性。下面这条定义则确认了需求类型的多样性(Sommerville 和 Sawyer 1997)：

需求是……对应该实现什么功能的说明——可以是对系统运行方式或系统特征与属性的描述；还可能是对系统开发过程的约束。

很显然，对于需求是什么没有一个统一的定义。为便于交流，我们需要协商决定一组限定词用来修饰“需求”这个内涵丰富的术语，并认识到用可通用的形式记录需求的重要性。

注意 不要一厢情愿地认为项目涉众对需求的理解是一致的。应该事先给出定义，才能保证大家谈论的是同一个问题。

1.1.2 需求的层次

本节将给出我对需求工程领域一些常用术语的定义。软件需求包括3个不同的层次——业务需求、用户需求和功能需求。除此之外，每个系统还有各种非功能需求。图1.1中的模型给出了各种需求关系的示意图。和其他所有模型一样，这个模型也不能面面俱到，但它确实有助于理解需求的整体概念。图中的椭圆代表各类需求信息，矩形则是存储这些信息的载体(文档、图形或数据库)。

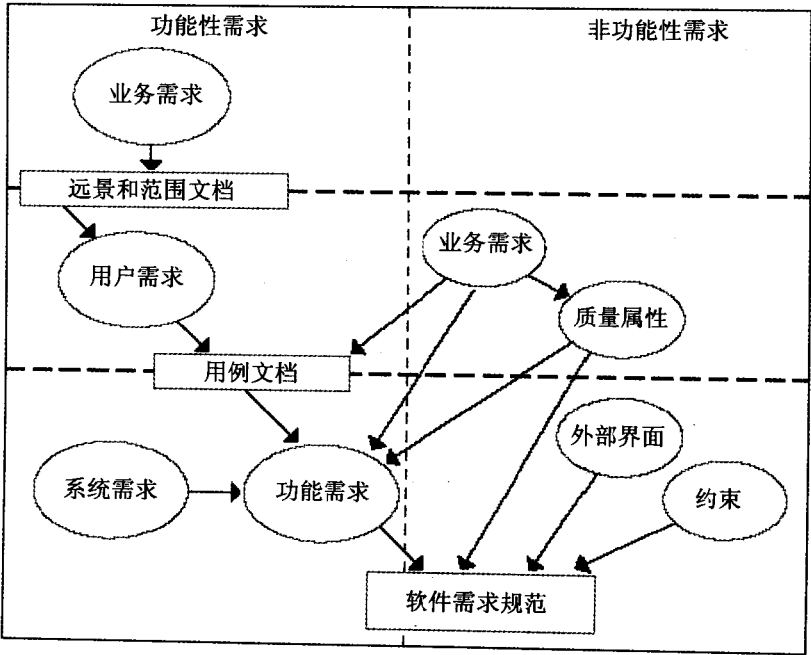


图 1.1 各种需求的关系

注：第7章中介绍了各种需求的示例。