



管理、决策与信息系统丛书

软件需求工程： 原理和方法

金芝 刘璘 金英/编著



科学出版社
www.sciencep.com

TP311.52

172

管理、决策与信息系统丛书

软件需求工程： 原理与方法

金 芝 刘 璘 金 英 编著

科学出版社

北 京

内 容 简 介

本书面向从事计算机软件科学研究和软件工程实践的读者,为读者介绍软件需求工程相关的理论和方法学知识。本书首先系统论述了软件需求工程的相关背景和基本知识,分析了软件需求工程的本质特征及其重要性。其次,选择目前比较有代表性的面向目标的需求方法、面向主体和意图的方法、基于情景的方法、问题框架方法、基于知识的方法、文档驱动的方法和面向方面的方法等进行了详细分析和阐述。不仅介绍了各种主流软件需求方法的核心概念,而且分析和对比了这些方法之间的异同和各自的适用范围。本书不仅包含国内外有代表性的软件需求建模与分析方法,还融合了作者在该领域研究多年的工作成果和心得。另外,全书在介绍软件需求工程领域研究的经典理论、最新进展和未来发展方向的同时,也为软件工业界的从业者了解本领域先进的方法学提供系统深入的指导。

图书在版编目(CIP)数据

软件需求工程:原理与方法/金芝,刘璘,金英编著. —北京:科学出版社, 2008

(管理、决策与信息系统丛书)

ISBN 978-7-03-021464-5

I. 软… II. ①金… ②刘… ③金… III. 软件开发-研究 IV. TP31.52

中国版本图书馆 CIP 数据核字(2008)第 039951 号

责任编辑:马 跃 赵静荣 / 责任校对:陈玉凤
责任印制:张克忠 / 封面设计:耕者设计工作室

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

骏 主 印 刷 厂 印 刷

科学出版社发行 各地新华书店经销

*

2008 年 7 月第 一 版 开本: B5(720×1000)

2008 年 7 月第一次印刷 印张: 21 3/4

印数: 1—3 000 字数: 407 000

定价: 48.00 元

(如有印装质量问题,我社负责调换〈环伟〉)

《管理、决策与信息系统丛书》
编辑委员会

主 编 汪寿阳

副主编 陆汝铃 章祥荪 杨晓光

委 员 (按姓氏笔画排列)

于 刚	邓小铁	石 勇	杨晓光
邹恒甫	汪寿阳	张汉勤	陆汝铃
岳五一	金 芝	赵修利	黄海军
章祥荪	程 兵		

丛书序

管理理论、决策科学与信息系统技术在 20 世纪获得了巨大的发展。在 20 世纪 80 年代,为了推动这三大领域在中国的发展以及推动这些领域之间的学科交叉研究,中国科学院管理、决策与信息系统重点实验室在科学出版社的支持下编辑出版了这套“管理、决策与信息系统丛书”。这套丛书不求全而求新,以反映最新的研究成果为主。经过编委会的各位专家,特别是前任主编许国志院士的努力和作者们的辛勤劳动,这套丛书在社会上尤其是在科学界得到了广泛的关注和好评。

回顾管理理论的发展历史,我们不难发现一个趋势:系统的概念和方法越来越多地应用到管理的各个方面,并成为管理理论发展的第三阶段的重要特征。管理理论的第一阶段形成于 20 世纪初,以 F. W. Taylor 为代表,倡导科学的管理,为提高工厂劳动生产率而提出了标准化原理。管理理论的第二阶段,从 20 世纪 20~30 年代开始,以行为科学为特点,主要代表有 A. H. Maslow, K. Lewin, R. Jannen 唱 baum 和 D. McGregor 等人。他们研究人的需要、动机、激励和定向发展;研究正式和非正式团体的形成、发展和成熟;研究个人在团体中的地位、作用、领导方式和领导行为等。管理理论的第三阶段出现在第二次世界大战后,这一阶段有各种学派,例如社会系统学派、决策理论学派、系统管理学派、管理科学学派和经验主义学派等。他们从不同角度强调系统的概念、理论和方法。这三个发展阶段并非截然分开,而是相互交叉的。

不论管理理论有多少学派,人们大致可以将它们分成三种模式:机械模式、生物模式和社会模式。生物模式认为:组织像一个生物,有头脑机构,有职能部门和分支机构。一个企业的目标可以分解,各部门完成其中的一部分。在这种模式下,目标管理得以发展。社会模式认为:各级组织都是一个交互的系统,它们有共同的目标、交互作用和信息联系,管理者是交互作用的中心。其特点是强调交互式管理(Interactive Management)和强调以系统方法来管理。这正是它不同于传统管理的地方。而传统管理大致可分为三类:回顾式(Reactive)管理、

被动式 (Inactive) 管理、预测式 (Preactive) 管理。回顾式管理是在自下而上地总结过去经验的基础上, 去发现组织的弱点, 找出克服其弱点的措施, 并在条件允许下去逐个地解决问题。被动式管理的特点是危机管理, 是“救火队”, 领导疲于处理当前各种各样的问题。而预测式管理的决策基于对今后的经济、技术、顾客行为和环境等的预测。这三类管理可以混合成各种样式的管理方式, 正像红、黄、蓝可以组成各种颜色一样。交互式管理强调系统的方法, 认为某个企业出现的市场问题绝不仅仅是一个市场问题, 而与 R&D、生产、原材料供给和人事等有关, 是一个系统的问题。回顾式管理的弱点是缺乏系统的观点。交互式管理强调要设计可见的未来, 创造一条尽可能实现它的道路, 这是“救火队”所不能做到的, 但它又不把一切都寄托于预测。交互式管理还强调“全员参与”和“不断改进”。

决策理论学派以 E. W. Simon 等人为代表, 是从社会系统学派中发展起来的。它认为决策贯穿于管理的全过程, 管理就是决策。决策的优劣在很大程度上依赖于决策者的智慧、素养和经验。计算机技术的发展不仅使人们能够快速地解决决策中的复杂计算问题, 而且可以有效地进行决策过程中的信息处理、分析等工作, 从而达到提高决策质量的效果。今天正处在新的发展阶段的决策支持系统 (DSS) 和管理信息系统 (MIS) 正是集管理理论、系统理论和信息技术三大领域的交叉学科方向, 它们为解决许多复杂决策问题提供了有力的工具。粗略地说, 决策问题大致可分为三个层次: 战略决策、结构决策和运行决策。战略决策是指与确定组织发展方向和远景有关的重大问题的决策。结构决策是指组织决策, 运行决策是指日常管理决策。

从信息论的观点看, 整个管理过程就是一个信息的接收、传输、处理、增功与利用的过程。计算机信息处理技术应用于管理走过了三个阶段: 数据处理 (EDP)、管理信息系统和决策支持系统。作为管理信息系统和决策支持系统的支持环境, 相对独立于计算机软件的开发, 需要研究和建立各类管理信息系统独特的支持软件系统和开发环境, 例如分布式数据库管理系统和分布式知识库管理系统, 面向用户、通用性较强和面向特殊用户的模型库、方法库管理系统, 以及一些专门的用户接口语言。

展望未来, 管理、决策与信息系统这个交叉学科的研究领域的发展有以下几个趋势:

1. 更加重视人的行为的研究, 企业的管理将不仅强调竞争, 而且应在竞争的前提下注重合作与协调;
2. 非线性建模与分析, 将取得大的突破;
3. 互联网的飞跃发展, 将为管理与决策分析提供新的研究问题以及支持平台。

这些趋势有两个重要特点：(1) 利用信息技术与数学中的最新成就去研究管理与决策问题；(2) 通过观察管理决策与信息系统发现其规律，形成数学与信息科学中具有挑战性的研究课题。

在这套丛书的编辑出版中，我们将不仅注重每本书的学术水平，而且也关注丛书的实用价值。因此，这套丛书有相当的适用面。丛书的作者们将竭尽全力把自己在有关领域中的最新研究成果和国际研究动态写得尽可能地通俗易懂，以便使更多的读者能运用有关的理论和方法去解决他们工作中遇到的实际问题。

本丛书可供从事管理与决策工作的领导干部和管理人员、大专院校师生以及工程技术人员学习或参考。

汪寿阳

前言

在构造任何工程制品之前，首先要弄清楚人们为什么需要它，要弄清楚它将在什么地方发挥何种作用，作用的程度如何等等，关于这些问题的答案是构造这个工程制品的意图和出发点，也就是该工程制品的需求。软件作为一种工程制品，需求的重要性就更为突出。这是因为软件与一般的工程制品相比，具有很强的特殊性。首先，它不能独立存在，总是用于支持一个包含人或其他机器在内的更大的系统。而需求恰恰定义了软件在这个大系统中所处的位置，以及与这个大系统其他部分之间的关系。其次，软件具有很强的目的性，其目的通常都是要通过计算机与人类共同完成现实世界的任务，以达到人类期望现实世界所要满足的状态。此外，它还不像任何其他工程制品一样有可描述的产品形态，是抽象无形的。软件唯一的物理形态是代码，而这个形态与它所参与完成的任务和实现的目的之间没有直接的对应关系。目的的抽象性和产品形态的不可预知性，使得准确理解软件行为存在很大困难。因此，在软件产品的开发中，需要进行系统化的需求工程活动。

一般认为软件需求工程的基本活动包括：需求获取、需求建模、需求分析、需求沟通、需求磋商与认可以及需求演化。与软件需求工程相关的这些活动在软件系统的开发中到底有多重要呢？本书第1章论述了软件需求工程在软件开发过程中的重要地位以及开展软件需求工程活动的困难所在。如果软件工程是关于如何解决软件设计问题的，则软件需求工程就是要定义待解决的问题，而通常所面对的问题几乎总是定义不足的。因此，软件需求工程可以看做是把一个还没有定义的问题或者定义不足的问题转换为一个定义良好的问题，只有先把问题定义明确了，才有可能找到问题的解决方案。

与软件工程相比，软件需求工程是一个新兴的研究方向，只有十几年的历史。从总体上说，还没有出现系统性的解决方案。软件需求工程方法论的研究在国际上也还仍然处在百花齐放、百家争鸣的局面之中。而国内关于软件需求工程的研究则只能说是处于起步阶段。目前，国际上出版过一些与软件需求工程相关

的图书和教材，但据作者所知，在这些图书中，有相当大的一部分完全侧重在软件需求阶段的工程化实施和过程管理方面，另外一些相关书籍则仅仅专门论述某种特定的软件需求工程方法的研究进展。

本书作者多年来一直从事与软件需求工程相关的研究工作，这使我们得以系统地了解和掌握当前国内外软件需求工程研究的主要方向和重要研究进展。同时，多年的研究工作和教学实践，也使我们深感需要有一本系统的综合性图书介绍软件需求工程的基本原理以及主要方法和技术，作为进行软件需求工程方法和技术研究以及软件需求工程实践的参照和起点。

本书的内容分为两大部分。其中，第一部分包括第1~3章，这一部分系统地论述了软件需求工程的相关背景和基本知识，分析了软件需求工程产生的契机和必然性，以及如何逐步得到学术界和工业界的重视，阐述了软件需求工程的本质特征以及所面临的困难和挑战。然后，分别就软件需求工程的过程、过程的活动和过程的管理，以及软件需求建模的基本手段等进行了详细的论述。仅希望了解软件需求工程一般性原理的读者可以以这部分为主进行学习。第二部分包括第4~10章的内容，主要选择目前国际上比较有代表性的软件需求工程方法进行详细分析和阐述，它们包括面向目标的方法、面向主体和意图的方法、基于情景的方法、问题框架方法、基于领域建模的预需求分析、文档驱动的方法和面向方面的方法等。需要进行软件需求工程研究，或实施软件需求工程实践的读者，可以从中找到自己感兴趣的内容。从这部分内容中，读者可以看到，不同的软件需求工程方法蕴含了不同的系统建模的理念和软件问题建模的切入点。例如，面向目标的方法和面向主体和意图的方法明显地是以社会系统建模为基础，其建模原语和分析技术都是从社会系统建模的角度出发而设计。问题框架方法则更关注目标软件系统将处于的物理/社会环境以及可能与环境发生的交互，认为软件系统的需求将蕴含在软件系统将处于的环境及其与环境的交互之中。文档驱动的方法也是从与环境的交互中建模软件系统的需求，但它更侧重在对抽取出的需求的形式化表达和分析处理技术上。基于知识的方法从原理上可以说是一种普适性的方法，也就是说，如果特定的建模和分析方法和/或特定领域的知识能够表达成一种可重用的形式，并且能提供相应的知识重用机制的话，都可以采用基于知识的方法来辅助整个软件需求工程活动的进行。面向方面的方法是近年来才提出的，它从面向方面的程序设计语言、程序设计方法以及软件开发方法的提出而产生，与目前软件工程中关注点分离的思想一致。

作者十分感谢中国科学院数学与系统科学研究院管理、决策与信息重点实验室“管理、决策与信息系统丛书”编委会的全体成员，特别是汪寿阳研究员和杨晓光研究员，感谢他们对本书的撰写和出版给予的热情支持和鼓励。我们还特别感谢科学出版社经管法出版分社马跃编辑的耐心和宽宏，本书书稿的完成时间大

大超过了原合同的规定，于今仍能顺利出版，这样的支持是难能可贵的。

本书的研究与撰写得到国家自然科学基金杰出青年基金项目“基于环境本体的需求工程理论和方法研究”（编号 60625204）、国家自然科学基金青年基金项目“基于意图的软件需求形式化建模方法研究”（编号 60503030）、国家重点基础研究发展计划（973 计划）课题“基于知识的网构软件需求建模理论和方法研究”（编号 2002CB312004）、国家高技术发展计划（863 计划）课题“面向方面的文档驱动需求工程技术研究”（编号 2007AA01Z123），和国家高技术发展计划课题“知识制导的需求工程技术研究”（编号 2006AA01Z155）的支持。本书第一作者曾于 2003 年在中国科学院计算技术研究所开设“软件需求工程论文导读”课程，并于 2004 年向中国科学院研究生院提出开设“软件需求工程”课程的申请并得到批准，因而得以开始承担中国科学院研究生院“软件需求工程”课程的讲授工作。本书第二作者也于 2005 年秋季起在清华大学软件学院开设软件工程硕士课程“软件需求工程”。本书内容和结构的成型得益于作者这些年来的课程讲授和科研实践。在写作过程中，我们还曾就书中的许多学术思想与目前活跃在软件需求工程及其相关领域的大师们，如软件工程 JSD 方法创建者及问题框架方法的发明人 Michael Jackson 先生，文档驱动方法的创始人 David Parnas 教授以及面向目标、面向主体和意图方法的提出者和研究者 John Mylopoulos 教授、Eric Yu 教授，基于知识的方法的倡导者陆汝钤院士等，进行过深入的讨论，从中受益良多。在此，我们谨向上述单位和个人致以衷心的感谢！

最后，值得说明的是，软件需求工程不仅涉及面广，学科交叉性强，而且目前仍然是一门发展中的学科，有很多问题还有待深入研究。另外，软件需求工程方法和技术的成型同时对其理论性和实践性方面均有较高的要求。尽管书中的内容和写作形式是作者经过再三斟酌确定的，但其中的错误、疏漏和不妥之处在所难免，我们恳切希望各位读者给予批评指正。

金 芝 刘 璘 金 英

2008 年 3 月 8 日

目 录

丛书序

前言

第1章 软件需求工程原理	1
1.1 失败的教训	2
1.2 软件加强型系统	7
1.3 软件需求工程的目的与任务	13
1.4 软件需求工程的作用和困难	24
1.5 本书结构	30
参考文献	31
第2章 软件需求工程过程	33
2.1 软件生命周期与软件需求工程过程	33
2.2 软件需求工程过程模型	36
2.3 需求抽取和发现	43
2.4 需求建模和文档化	50
2.5 需求分析和协商	52
2.6 需求审查和验证	56
2.7 需求管理	59
2.8 小结	64
参考文献	65
第3章 软件需求建模基础	67
3.1 概念建模	67
3.2 功能需求建模	71

3.3 非功能需求建模	81
3.4 形式化建模	88
3.5 小结	89
参考文献	90
第4章 面向目标的方法	92
4.1 目标概述	92
4.2 建模元素和表示	95
4.3 目标驱动的需求获取方法	103
4.4 小结	120
参考文献	120
第5章 面向主体和意图的需求方法	122
5.1 面向主体方法概述	122
5.2 基于策略主体的建模元素与表示	128
5.3 基于策略主体的早期需求建模	136
5.4 小结	142
参考文献	144
第6章 基于情景的需求方法	145
6.1 基于情景的需求方法概述	146
6.2 基于情景的方法系列——CREWS	149
6.3 基于用例的建模——UML 和 UCM	155
6.4 形式化的情景分析方法	163
6.5 形式规约获取	170
6.6 小结	188
参考文献	190
第7章 问题框架方法	192
7.1 问题框架方法概述	192
7.2 问题框架描述原语	199
7.3 基于问题框架的需求分析	211
7.4 几个需要关注的问题	227
7.5 小结	230
参考文献	231

第 8 章 基于领域建模的预需求分析	232
8.1 概述	232
8.2 基于本体的领域建模	237
8.3 基于本体的业务情况抽取	248
8.4 基于本体的业务情况建模	253
8.5 从业务情况描述到规格说明	261
8.6 小结	270
参考文献	271
第 9 章 文档驱动的方法	272
9.1 需求文档	273
9.2 文档驱动的需求方法	274
9.3 Tabular 表达式	282
9.4 SCR 需求方法	300
9.5 小结	314
参考文献	315
第 10 章 面向方面需求工程	317
10.1 背景	317
10.2 面向方面需求工程概述	318
10.3 几种主要的面向方面需求工程方法	322
10.4 小结	329
参考文献	331

第1章 软件需求工程原理

讨论软件需求工程，有必要从软件工程说起。关于软件工程，通常可以从软件工程教科书中看到这样的陈述：软件工程的基本目标是，采用系统化的构建方法和工程化的管理手段去高效地开发出满足用户需求的软件系统。这句话表明，软件工程的目标要依赖一个前提，就是需要有能清楚表达的正确的需求。因为系统开发者要根据已经清楚表达出来的用户需求去开发软件系统，并根据这个用户需求来判断开发出来的软件系统是否满足要求。

显然，在构造任何系统之前都应该先决定想要构造什么。软件系统的开发更是如此，因为软件系统的构造要求有很高的描述精确度，还要求开发小组之间的协调。如果没有对将要构造什么给出一个准确并被一致同意的描述，则可能会开发出很多不能满足要求的软件系统，产生很多不满意的客户。定义用户需求并给出准确的描述，就是建立对要构建的软件系统的一个共同理解。

那么，用户需求到底是什么？应该如何构建这样的共同理解？构建出来的是关于什么的共同理解？如果构建用户需求不是一项简单的任务，是否也需要一些手段、策略和方法去获得用户需求，并通过系统化的过程将其一步步构建并描述出来呢？是否也需要尽可能高效地获得高质量的用户需求呢？等等。这些问题的提出，导致了软件需求工程的产生。因此，对应于软件工程，软件需求工程的目标则应该是，采用系统化的方法和工程化的管理手段，高效地开发出能准确表达用户需求的软件规格说明。也就是说，软件需求工程的目的是为软件系统提供一个合理正确的开发文档。从这个意义上说，软件需求工程无疑是软件工程之前的一个重要阶段，它的成功实施是软件的后续开发过程得以成功的先决条件。

近年来，软件需求工程已经得到长足的发展。但从其发展进程和未来的发展趋势来看，软件需求工程不能仅仅局限在软件工程为需求阶段设定的目标上。也就是说，除了软件工程为它设定的上述目标外，它更强调要用一种自然的方式，去系统地寻找和发现正确的和/或（特别是）潜在的用户需求，为这些需求建立适当的模型，并在对模型进行正确性分析和有效性验证的基础上，为软件开发的后续阶段提供合理和正确的软件规格说明。

软件需求工程是一个充满生机的新兴领域。随着信息技术的不断发展，软件系统的规模不断扩大，软件所能实现的功能日益复杂，软件及其运行环境和与外界的交互方式和交互场景变得越来越不可预测。在这种情形下，软件需求工程的出现和发展可以说已经成为一种必然。本章将沿着这个思路，去回答读者可能会

提出的涉及软件需求工程这个研究方向的一些基本问题。例如，为什么要将软件需求工程从软件工程中独立出来进行专门的研究和实践？软件需求工程将对解决什么样的问题有用？等等。在此基础上，本章将向读者展示的内容还包括：软件需求工程的基本原理、进行软件需求工程要面对的困难以及软件需求工程应该发挥的作用等。

1.1 失败的教训

自 20 世纪 60 年代末起，软件的数量急剧膨胀，软件需求日趋复杂，采用工作坊式的软件开发方法难以应付软件的开发任务，软件开发的成本越来越高，软件维护的难度也越来越大。这些问题迫使软件开发者们开始对软件及其特性进行深层次的研究。1968 年，在北大西洋公约组织召开的一次国际会议上，研究者们第一次提出要解决“软件危机”问题。也就是在这个会议上，软件工程作为一门独立的学科正式诞生了，其目的是要为软件开发过程建立具有良好定义的工程化原理，以便能够经济地获得可以在实际机器上有效运行的可依赖的软件。

20 世纪 90 年代中期至末期，在欧美等发达国家，IT 技术及其应用取得了飞速的发展。政府或企业对 IT 应用项目的投资总量急剧增长。根据美国 Standish Group 在 1995 年的统计结果，在 90 年代中期，美国国内平均每年要投入 2 500 多亿美元，用于支持约 17.5 万个 IT 应用项目的开发。其中，大型公司得到的平均项目投资是 232.2 万美元，中型公司得到的平均项目投资是 133.1 万美元，小型公司得到的平均项目投资是 43.4 万美元。软件系统的开发进入高峰期，所需要的软件系统的规模越来越大，软件系统本身及其应用环境也越来越复杂。

尽管当时软件工程已经诞生了近 20 年，软件工程研究者和实践者也已经提出了许多软件开发方法和技术，并建立了一些软件项目的管理手段。但是，当时的软件开发方法和技术以及软件项目的管理手段，对胜任功能日益复杂和规模不断扩大的软件的开发，仍然存在这样或者那样的问题，使得很多开发项目最后还是以失败告终。下面罗列的是几个经常被软件需求工程教科书或者研究论文所引用的、由于需求阶段的问题导致的严重的软件开发失败案例。

- 1990 年：英国 Wessex 地区信息系统项目（Regional Information Systems Plan, RISP），在花费 4 300 万英镑后被放弃。

- 1992 年：英国演出权益协会（Performing Rights Society）的信息系统项目（PROMS），在花费 1 100 万英镑之后被放弃。

- 1992 年：伦敦救护车服务派遣系统（London Ambulance Service Despatch

System, LAS), 在花费 150 万英镑开发出来后, 于 1992 年 10 月 26 日正式投入使用, 全面替代原先的手工派遣系统。但在运行两天后被迫停止使用, 因为它被指控延迟了救护车到达急救地点而引起了病人的死亡。

- 1993 年: 伦敦股票交易所 (London Stock Exchange) 的 TAURUS 项目, 在花费了 7 500 万英镑后被取消。

- 1998 年: Swanick 空中交通控制系统 (Swanick Air Traffic Control), 计划在 1998 年完成, 但 2001 年还未完成 (额外开支 1.8 亿英镑)。

这些都是投资非常大的项目, 其软件系统的规模可想而知。而且这些软件从其应用范围和场景来看, 基本上都是和现实世界中的社会系统紧密耦合的, 软件系统和外界的交互相当复杂, 交互过程和场景一般是持久的, 同时又是不可预测的。

事后, 人们发现在引起项目失败的原因中, 需求阶段的问题占了相当的比重。PROMS 项目失败的一个主要原因, 是未能以客户能够理解和检查的形式去表达软件需求, 从而引起与客户沟通方面的困难, 使客户不愿意或者不能够紧密参与到需求的分析中来; RISP 项目之所以失败, 则是由于没有清楚地给出所能解决的问题的范围, 即没有明确地定义问题的边界; TAURUS 项目的许多问题, 来源于未能协调处理好那些不一致的需求, 同时也缺乏有效的、不一致需求的管理方法; Swanick 空中交通控制系统出现问题的主要原因是, 在需求还没有弄清楚、缺乏健全的需求规格说明的情况下, 就匆忙开始进行系统实现的后续工作, 从而引起不断返工。

为了更深刻地理解这些失败原因, 人们从许多方面对这些失败案例进行了深入的分析, 下面我们给出 (Macaulay, 1996) 中对 LAS 系统的一个分析结果。开发 LAS 系统的目的是建立一个尽可能全面自动化的救护车派遣系统, 来处理紧急呼救、并派遣救护车, 它需要根据电话号码来自动确定急救地点, 根据无线信号确定救护车的位置, 在控制室显示屏上显示救护车和事故发生地点, 为每个特定的呼救提出最合适的资源调度方案, 最后将任务分配命令通过无线发射装置传送到救护车的移动终端上。

这个项目的失败起码有两方面原因。

首先, LAS 系统要解决的问题属于社会服务领域, 对这类领域进行需求分析, 需要涉及方方面面的社会成员, 如 LAS 管理人员、工会、LAS 的技术经理以及政府部门等, 关于 LAS, 他们有不同的目的、起不同的作用、关心不同的兴趣点、有不同的期望。这种系统是否成功不能孤立地根据技术层面的成功来判断, 而必须关注各个群体的目标。表 1.1 罗列了这样一些可能的不同目标, 很显然这些目标有些是矛盾的, 这些矛盾揭示了不同群体之间兴趣上的冲突, 这些冲突不仅关系到组织本身, 而且影响到 LAS 系统以及它的开发。即使这些群体

对 LAS 系统达成过一致，也只能看做在特定的情形下，他们的目的得到了满足。他们对 LAS 的态度如果没有考虑周全，则有可能不能准确定义需求。

表 1.1 与 LAS 系统相关的四个不同群体的组织目标 (Macaulay, 1996)

群 体	组织目标	对技术的态度	对系统的兴趣
LAS 管理人员	改善信任状况 企业文化的改变 更低的开销 撤开工会	有帮助的	达到 LAS 范围内的 目的的方式
工会	更多资源 更好的工作条件 有义务、有责任维持 工会的地位	看在什么情景下使 用	改善工作条件的 方式，支持更好地 工作
系统经理	开发和管理好的系统	功能主义/工程化	为 LAS 提供好的 服务和系统
政府	NHS 政策的好案例 创造价值 降低工会的力量	政治上的需要	解决问题或者做 宣传的设备

其次，不同群体的目的和兴趣都是建立在组织、组织的政策以及社会环境等特定上下文中，不仅仅与技术相关，软件系统是成功了还是失败了不仅关系到“保证系统在技术上的正确”，甚至关系到“让它适合一个存在的或者计划中的组织结构”。例如，政府的全民健康服务 (NHS) 完全不在系统的控制范围之内，但它却对人们判定系统的成败起到某种程度上的决定作用。从这点上看，如果要想避免被看成是失败的话，技术人员不能假设在他们自己的技术范围和社会环境之间存在一个预定的边界，也就是说，这样的边界总是不能固定的。但是，如果是这种情况，技术人员就会直接牵扯到社会冲突中，这是技术人员所不愿看到的。同时，这一方面也隐含了在某些情况下，技术人员可能对系统被认为是成功还是失败这个结论没有太大的影响力。

我们再回到美国 Standish Group 在 1995 年所做的那项调查报告中，对这些项目进行分析的结果也揭示了当时美国 IT 项目应用开发的严峻局面。实际上，当时美国 IT 项目的最终结局是令人惊讶的，报告说，如果约定成功的项目指按时并按照规定预算完成了最初说明的所有系统特征和功能的话，则对大公司而言，其承担项目的成功率只有 9%，中型和小型公司承担项目的成功率分别是 16.2% 和 28%。许多项目是属于有问题的项目，即项目超时完成、超过预算或者改变了最初约定的系统功能。其中，大型公司的项目有 61.5% 是有问题的，中型和小型公司有问题的项目分别占 46.7% 和 50.4%。完全失败的项目的情况是，大型公司有 29.5% 的项目被取消，中型和小型公司被取消的项目分别占 37.1% 和 21.6%。总体上说，项目的成功率只有 16.2%，有问题的项目占