软件工程与软件质量分析

软件工程
与软件质量分析

李友仁　编著

電子工業出版社

软件工程与软件质量分析

李友仁　编著

电子工业出版社

内 容 简 介

本书系统地介绍了计算机软件系统的分析、设计、测试、维护和软件质量分析的理论与方法。

全书共分三部分。第一部分全面地讨论了软件系统分析与设计的理论和方法，介绍了许多有用的工具，并通过工资系统的分析与设计把理论与实现紧密地结合起来。第二部分详细地研究了近几年来国外在软件测试方面采用的有效的新技术以及软件测试的组织和管理方面的经验和措施。第三部分讨论了软件质量分析方法与软件维护技术，介绍了软件可靠性理论，多种错误预测模型，提高软件质量的方法以及怎样设计一个高质量的软件产品，还特别介绍了许多有效的维护技术。

本书取材广泛，内容丰富，注意理论联系实际，可供高等院校有关专业作为教材使用，也可供广大计算机软件设计、应用和管理人员阅读。

软件工程与软件质量分析

李友仁　编著

责任编辑　王昌铭

电子工业出版社出版（北京海淀区万寿路）

新华书店北京发行所发行　各地新华书店经售

山东电子工业印刷厂印刷

开本：787×1092毫米 1/16　印张：16.625　字数：349.2千字

1987年10月第一版　1987年10月第一次印刷

印数：1—10,000册　定价：4.00元

统一书号：15290·658

ISBN7-5053-0001-6/TN1

前　言

自本世纪六十年代以来，随着计算机应用领域的迅速扩大，计算机软件系统也越来越庞大和复杂。沿用从前的手工方式，分散地和无约束地研制大型软件系统遇到了巨大的困难。这些困难主要表现为：一个软件产品往往需要投入大量的人力和物力，因无法预先精确地估计它的经费，导致预算大大超支；软件人员在进行软件产品开发时，往往对“用户要求”是模糊的，完成的软件系统不能充分地符合用户的期望；在软件开发过程中未严格地进行软件测试与质量保证工作，往往不能保证软件的质量；软件的可维护性差。这就是所谓“软件危机”。软件危机的出现给人们提出了如何开发软件；怎样维护现有软件；怎样才能生产更多更好的软件产品以满足社会的需求等等问题。“软件工程”就是在这种背景下发展起来的一个软件分支学科。软件工程研究的目标是：如何使用理论知识、科学方法和工程上的设计规范来指导软件的开发，以达到用较少的时间、较小的开支获得高质量的软件产品。

如果把与这一目标有关的研究领域都归入软件工程研究内容的话，那么，软件工程的内容是十分庞大的。它不仅包括系统的分析与设计，结构程序设计，软件测试和维护，软件生产的组织与管理，而且还包括软件开发的工具和环境，以及软件形式开发方法，程序验证等。事实上，许多教科书和专著都冠以“软件工程”的名字，但其内容不尽相同，它们仅包含了上述研究范围的不同部分。本书研究的内容分为三个部分。第一部分：系统分析与设计；第二部分：系统测试；第三部分：软件质量分析与维护。本书包括了软件工程中最重要的内容，并且已在软件开发中得到了广泛的应用。学习和掌握这些软件开发方法和思想不仅对计算机类专业的学生是必须的，而且对从事计算机应用与开发的广大技术人员和管理人员也是十分必要的。

本书第八章的初稿由刘少英同志编写，卞雷同志审阅了本书，龚彬、李小琦同志在画图，文字修改和抄写方面做了大量工作，在本书的编写过程中还得到了西安交通大学计算机科学与工程系许多同志的帮助和支持。在此，对他们的辛勤劳动和帮助谨表示衷心的感谢。

西安交通大学　李友仁　1987.2

目　录

第一部分　系统分析与设计

第二部分 软 件 测 试

第三部分 软件质量分析与维护

第一部分　系统分析与设计

第一章　系　统　分　析

§1.1　概　　述

1.1.1　软件的生存期

通常，人们把软件的生存期划分为：系统分析、总体设计、详细设计、编码、测试、维护等六个阶段。

1. 系统分析

系统分析阶段的主要任务是收集，理解“用户要求”；对用户要求进行可行性分析；估价系统的投资和收益；作为系统分析阶段的结果给出系统说明书(或称为规格说明)。

2. 总体设计

总体设计又称系统设计，它的主要任务是决定系统的模块结构。也就是说，系统如何划分成模块；模块之间如何联系；以及对模块划分的评价等。

3. 详细设计

详细设计的主要任务是确定和描述模块的功能，模块的接口和实现方法。

4. 编码

由程序员编写程序和建立相应的文档。

5. 测试

检查和揭露程序的错误，以便改正和排除错误。

6. 维护

在使用中不断地改进系统，以满足变化了的用户需求。

1.1.2　什么是系统分析

系统分析是软件开发的第一个阶段。系统分析阶段的主要任务是：

1. 收集和分析用户的要求

用户的需求是指用户所期望的软件系统的全部性能和限制。它主要包括：功能要求；性能要求；可靠性要求；安全、保密要求；研制费用和时间，以及资源方面的限制等。

2. 进行可行性分析

可行性分析是研究在用户给定的条件下，软件系统可否实现。它主要包括澄清问题的定义(即准确地描述用户希望解决的问题)，确定系统的范围和目标；开发逻辑模型，

推荐各种可能的方案，提供可行性报告等。

3．做费用和收益分析

软件系统费用和收益的估价对一个软件系统是否值得继续开发是十分重要的。它主要包括对当前系统和未来系统的费用和收益进行分析对比。

4．系统说明书

作为系统分析阶段的结果，应提供系统说明书。系统说明书是一份精确和完整地表达用户要求的书面材料。它是用户和软件人员达成的协议或签订的合同。

系统说明书是软件生命期中极为重要的文档。在设计、编码、测试阶段中，软件人员以它为基础进行工作；在系统完成后，作为用户验收的标准；管理人员以它为基础，了解和掌握工作进度计划；它也是系统维护的基本文档。因此，系统说明书必须满足精确性，完整性、一致性，易于理解和易于维护性等。

1.1.3 分析员和用户的关系

所谓用户通常指企业的主管人，企业中各部门的负责人和具体工作人员三个阶层中的代表。因为企业主管人了解总的策略和今后的发展，部门负责人了解各部门的业务处理过程，而具体工作人员了解具体的业务细节，他们提供的情况在分析阶段应认真收集和分析，用户一方的责任是从他们的业务角度对系统提出要求，一般他们不干涉系统如何实现。

分析员是用户和软件系统实现者之间的中间人和桥梁。因此，他们应该了解用户业务领域的有关知识、熟悉计算机软件开发的方法和技术，能够在用户和计算机软件人员之间进行联系、交流信息。目前大多数分析员由软件开发部门的有经验的软件人员担任。随着计算机技术的普及，计算机应用领域的扩大，分析员将会由用户中熟悉计算机技术的人员承担。

分析员应访问用户，了解和理解用户的要求，召集各种形式的会议，协调和沟通用户和软件人员的思想和认识，最后对系统应“做什么”达成协议。这就是说明书。

分析阶段结束之后，分析员可从事软件系统的设计，编码或测试工作。当然，系统说明书不可能是绝对精确的和完善的。事实上，随着开发工作向后继阶段的延伸，就是对说明书进一步求精和改进的过程。系统说明书的改变是不可避免的。因此，分析员需要继续不断地同用户进行联系。

用户在分析阶段之后，也应了解软件研制工作的进展，随时将业务范围的细节告诉分析员，并准备参加系统验收。

1.1.4 什么是结构分析

所谓结构分析是指在分析阶段使用了“由顶向下”，逐层分解”的方法，即在顶层抽象地描述整个系统，然后逐层分解，到底层时再具体地描述系统的每一个细节。这就是从抽象到具体的逐步分解(求精)过程。对系统作了合理的分解之后，就可以分别讨论每一部分的细节。对每一部分编写说明，并把它们组织起来，就获得了整个系统的说明书。

结构分析是系统分析的一部分，此外，系统分析还包括以下内容：费用-收益分析、可行性分析、工程管理、性能分析、设备选择、人员挑选、开发政策等。

结构分析的过程可分四步进行：

（1）理解当前的现实环境，获得当前系统的具体物理模型。

（2）从当前系统的具体模型抽象出当前系统的逻辑模型。

（3）分析目标系统与当前系统在逻辑上的差异，建立目标系统的逻辑模型。

（4）对目标系统的逻辑模型作补充。

§1.2 可行性研究与费用-收益分析

1.2.1 可行性研究

1. 什么是可行性研究

可行性研究是整个系统分析和设计过程的一种简略形式或缩影。它从澄清问题定义开始，确定初始的系统范围（规模）和目标，识别强加在系统上的限制。在给出问题的定义之后，分析员便可着手开发系统的逻辑模型。然后，以这个逻辑模型为基础，研究各种可能的实现方法，分析它们的可行性。所谓可行性我们主要指在技术上，经济上和管理上系统实现的可能性。

（1）技术方面：利用当前的技术，该系统可以实现吗？

（2）经济方面：收益大于费用吗？

（3）管理或组织方面：在本组织（公司）中能够完成该系统吗？

对每个可行的方案，分析员提出一个粗略的实现计划。

可行性研究的结果是向管理部门和用户提供一个书面报告。如果分析员找到了可行的方案，那么，可行性研究应提供有关该工程项目的广泛的技术说明，即执行计划。否则，分析员将建议放弃该工程项目。

对可行性研究需要花费多少时间呢？这依赖于工程的规模。对一个非常小的应用系统，一个有经验的分析员也许仅通过简单交谈就足够了。对需要花费数十万元的信息系统，可能需要花费几周时间。对软件公司开发的价值数百万元的新系统，可能需要花费半年或一年时间。一般来说，可行性的研究费用占总工程费用的5%～10%。

2. 可行性研究的步骤

1）问题的定义、系统的范围和目标

在本步中，分析员应该进一步确定问题的定义（即准确地描述用户希望解决的问题），工程预期的范围和系统的目标。应该清楚地识别系统的一切限制条件。此外，需要同有关人员会谈，并检查和研究书面材料。

2）研究当前系统（如果它存在），建立当前系统的逻辑模型

当前的系统是重要的信息源。既然这个系统还在使用，它一定在做某些有用的工作。所以，它的基本功能必须加入到新的系统中。另一方面，如果当前系统能很好地工作，那么，也没有必要去设计一个新的系统。因此，必须研究和找出当前系统中存在的问题。最后，把当前系统运行的费用作为计算的基点，如果新系统未能提供额外的收益或减少费用，那么，应该保留当前系统。

分析员应研究当前系统的文档和执行过程，了解当前系统做什么和为什么以这种方

式来实现它的功能，弄清楚运行当前系统的费用，建立当前系统的逻辑模型。

一个普遍的错误是花费太多的时间去分析当前系统。分析员不要过多地关心当前系统如何工作，而应把注意力集中在研究“系统做什么？”上。

3）开发新系统的逻辑模型

分析员在对老系统的功能和限制有较深入的认识之后，他能够利用数据流图和数据词典构造新系统的逻辑模型。以后将利用这个逻辑模型设计新系统。换句话说，好的设计应从当前实际系统出发，开发它的逻辑模型。利用这个逻辑模型构造新系统的逻辑模型，然后再依据新系统的逻辑模型构造新的实际系统。

4）利用新知识重新定义问题

实际上，新系统逻辑模型的开发表示分析员已理解这个系统应该做什么。但是，这种理解是否符合用户的要求还需要进一步验证。为此，本步骤中分析员应该同用户一起检查问题定义、范围和目标。在检查中应该利用表示逻辑模型的数据流图和数据词典作为讨论问题的基础。如果分行析员有错误的理解，或用户有某些疏忽，那么，现在是发现它们的机会。我们把可行性研究的前四步看作一个循环。分析员定义问题，分析问题，提出一个暂时的解答，再次定义问题，再次分析它 修改解答，继续执行这个循环过程直到逻辑模型符合系统的目标。

5）开发和评价各种可能的方案

给出一个新系统的逻辑模型以后，分析员能够开发高层的 各种可能的物理（具体）方案。分析员如何产生这些可能的方案呢？一个最容易的方法是从技术可行性出发，提出多种可能的方案。例如，按照某种规则把数据流图划分成若干个部分，每一种划分对应着一种可能的物理方案。

另一种方法称之为“诸葛亮会议”。在给出系统的逻辑模型以后，由分析员和他的同事一起开会，由每个人提出各自的可能建议。在会上不对每个建议提出批评和进行评价。在会后，由分析员评价各种建议，并选择出少量的、似乎可行的方法。一旦已产生了一组可能的方案后，就按照技术的可行性进行分析，删除不合理的方案。

其次，从管理和组织的可行性方面考虑，即分析员必须研究管理和组织方面的某些偏爱、特长和要求，排除那些不满足这些要求的方案。

接着，再从经济可行性方面考虑。对剩余的每个方案做费用-收益分析，评估这些方案比当前系统能节省多少费用。对能收回费用的方案再作进一步分析。

最后，对通过技术、管理和经济可行性测试的每个方案制定实现进度表。在实现进度表中给出各开发阶段的完成日期。

6）对该系统是否值得继续开发提出建议

分析员应向用户和管理部门提出继续开发或放弃开发的明确建议。如果建议继续开发，分析员应选择出最好的方案，并论证这种选择。提出粗略的开发计划，例如，实现进度表，估计每个阶段需要安排多少软件人员，估计每个阶段的费用等。

7）写出可行性研究报告

可行性报告主要包括以下内容：报告标题、报告目录、问题定义、报告摘要、可行性研究的方法、新逻辑系统的分析、可选择的方案、建议、开发计划、附录等。

1.2.2 费用-收益分析

1. 什么是费用-收益分析

费用是指开发和(或)运行软件系统所支付的资金。而收益是指新系统增加的收入或减少的费用。开发软件系统是一种投资。这意味着当前向某一项目支付资金，希望将来某个时候获得收益。在软件生存期的每个阶段都需要投资。而期望的收益来自减少费用或增加收入。如果期望的收入小于费用，那么这个系统就不值得做下去。

投资有多种形式，不管哪种投资都有不同程度的风险。把资金投入开发新系统也要承担很大的风险。因为费用可能比预期的大，收益比期望的少。一个系统是否值得投资？费用-收益分析就回答了这个问题。

2. 影响费用和收益的因素

为了比较费用和收益，分析员必须首先估算它们。我们首先讨论系统开发的费用。在估算给定阶段的开发费用时，分析员应该考虑以下几个方面的因素：职员（包括分析员、程序员、操作员、办事员，管理人员和其它附属人员)；设备(主要包括基本设备、设备安装与调试、现有设备利用、文件转换、系统测试)；材料和物资储备；管理开支及其它(如资询费，特殊的培养费等)。

我们现在再来讨论操作(或系统运行)方面的费用。一旦系统被实现后，必须连续地付给操作方面的费用。它包括以下几方面的因素：硬件和设备维护费用；操作员、办事员和程序员的工资；消耗品的供应；管理开支等。开发和操作费用之间的差别在于开发费用是一次性付给。当系统交付使用后，开发就停止；这时，开始支付操作费用，并且在系统的整个使用期中连续支付操作费用。我们可以使用下面的公式计算收益

$$收益 = 收入 - 费用$$

增加收益的方法有两种：增加收入或减少开支(费用)。因此，在做费用-收益分析时，必须强调减少费用或增加收入。当增加收入是主要部分(即费用减少可以忽略不计)时，也就是说通过扩大新产品或改进产品的销售，增加了收入时，计算收益方法是从新收入中减去与扩大新销售有关的操作费用。当费用减少成为主要部分时(增加收入可以忽略不计时)，收益是利用老系统和新系统操作费用之差计算的。然而，当估算一个新系统的收入或费用时可能发生错误。这种错误会产生一定的风险。因此，一个好的费用-收益分析应清楚地说明系统的费用、收益和风险。

3. 费用-收益分析

费用-收益分析的第一步是估算新系统的开发费用，操作费用和收益。收益和操作费用在整个系统使用期限内都是有效的。使用期究竟多长为最好呢？因为一个研究和开发的项目很可能在项目使用数年之后才能产生效益，所以期望的使用期通常是十年或更长一些。但是系统的使用期越长，系统被废弃的危险也越大。因此，估算收益和费用的期限太长是充满危险的。在以下讨论中，我们预计系统的使用期为五年。

例如，我们考虑一个库存系统时，假定新系统因能缩短库存时间，每年可节省2,500元。开发新库存系统的费用估计为5,000元。把这些数字用图1.1说明。

1) 资金的时间价值

为了在一年内获得2,500元，现在愿意投资多少钱？回答是少于2,500元。为了在五

开发费用	5,000元
收　入：	
年	金额
1	2,500元
2	2,500元
3	2,500元
4	2,500元
	2,500元

图1.1　库存系统的费用和收益

年内获得2,500元，现在想投资多少钱？即使现在投资少于2,500元也不一定合算，因为今天的2,500元和五年后的2,500元是不等值的。资金有其时间价值，并表现为利息。资金的当前价值与资金的将来价值是不相等的。利用下面计算复利息的公式计算投资的将来价值：

$$F = P(1+i)^n$$

这里，F是投资的将来价值，P是投资的当前价值，i是复利计算周期的利率，n是复利计算的周期(通常以年为周期)数。

例如，把5,000元存入银行3年，年息为12%，那么三年后5,000元的价值为：

$$F = 5000 \times (1+0.12)^3 = 7024.64$$

在图1.1中，我们的投资为5,000元，收入估计每年为2,500元。然而，收入在将来才能获得，并且我们不能直接把当前的价值同将来的价值作比较，而不考虑资金的时间价值。例如，若在第五年后希望获得2,500元，那么现在愿意投资多少(按年利息12%计算)？答案应该是：把收入的当前价值同投资的当前价值做比较。

如何计算收入的当前价值呢？我们从基本的复利公式开始：

$$F = P(1+i)^n$$

解：当前的价值·

$$P = \frac{F}{(1+i)^n} = \frac{2500}{(1+0.12)^5} = 1418.57$$

这就是说，如果我们今年投资1,418.57元，那么五年后可收回2,500元(按年息12%计算)。把将来的价值转换为等价的当前价值称为折扣或贴现。

如果对每年都重复以上计算，其结果如图1.2所示，这样我们就得到了投资和收入的当前价值，并且能比较它们。

年	将来价值	$(1+i)^n$	现在价值	现在价值累计
1	2,500	1.12	2,234.14	2,234.14
2	2,500	1.2544	1,992.98	4,225.12
3	2,500	1.4049	1,779.45	6,004.57
4	2,500	1.5735	1,588.80	7,593.37
5	2,500	1.7623	1,418.57	9,011.94

图1.2　年收入的当前价值和累计价值

2) 偿还期

工程的偿还期是度量工程相对价值的一般方法。偿还期是指使累计的收入等于最初投资所需要的时间。显然，偿还期越短，开始获得收益的时间也就到来得越快。因此，投资也更令人满意。

例如，从图1.2中看出，到第三年底累计收入为6,004.57元，也就是说，在第三年中的某个时候到达了偿还期。在两年后我们获得的收入为4,225.12元，要到达偿还期还

差774.88元。第三年的收入是1,779.45元。因此，还需要第三年收入的44%作为补偿。所以，偿还期是2.44年。

偿还期是非常保守的度量值，不应单独使用它。但是，当同其他度量方法结合使用时，特别当一个工程项目在技术上出现废弃的危险时，它是非常有价值的。

3）当前纯价值

另外一种有用的度量是当前纯价值：它是收入的当前价值同投资的当前价值之差。例如，在图1.2表示的库存应用中，5,000元投资的累计收入是9,011.94元，当前纯价值是4,011.94元。如果当前纯价值大于零，那么这个项目是无危险的，经济上是合算的。如果当前纯价值等于零，那么这个项目是无危险的，但是，它可能是不值得开发的。如果当前纯价值是负的，那么这个项目肯定不值得开发。

为了同其他投资机会作比较，有时给出当前纯价值占投资的百分比。例如：

$$\frac{4013.44}{5000.00}=0.80 \text{ 或 } 80\%$$

如果项目的使用期为五年，那么每年平均为16%。把这个数字称为每年的投资利润率。当与其他同类数字做比较时它是有用的。不要把它同银行的利率混为一谈。

4）内部偿还率

现在来讨论内部偿还率，它是可以用作同最优惠的利率或其他金融市场统计数字做比较的数字。如果我们有投资的当前价值和一串估算的将来收入，那么内部偿还率定义如下：如果把投资作为存款，在每年底取回适当的金额，则内部偿还率是这样的复利率，即到偿还期结束时把存款取完。例如，对库存问题，现在存款为5,000元，每年取回2,500元，在五年后把存款取完，它的利率就是内部偿还率。

求内部偿还率的关键是在规定的期限内，把存款取完。因此，在时间到期时，投资与收入相低。利率的计算使用以下公式：

$$p=F_1\left[\frac{1}{(1+i)}\right]+F_2\left[\frac{1}{(1+i)^2}\right]+F_3\left[\frac{1}{(1+i)^3}\right]+\cdots+F_n\left[\frac{1}{(1+i)^n}\right]$$

或

$$0=-p+F_1\left[\frac{1}{(1+i)}\right]+F_2\left[\frac{1}{(1+i)^2}\right]+F_3\left[\frac{1}{(1+i)^3}\right]+\cdots+F_n\left[\frac{1}{(1+i)^n}\right]$$

这里，p表示投资的现在价值，F_i表示每年收入的将来价值，i表示所求的利率。例如库存问题有如下的方程式。

$$0=-5000+2500\left[\frac{1}{1+i}\right]+2500\left[\frac{1}{(1+i)^2}\right]+2500\left[\frac{1}{(1+i)^3}\right]$$
$$+2500\left[\frac{1}{(1+i)^4}\right]+2500\left[\frac{1}{(1+i)^5}\right]$$

从上述公式中所求的未知数i就是“内部偿还率”。计算结果i的值在41%～42%之间。换句话说，内部偿还率是每年偿还金额占投资的百分比。它在项目的经济效益中是最重要的参考数据。

4．风险

若比较三种完全不同的投资选择：银行存款、设计新的库存系统和赌博。那么，银行是可信赖的，库存系统有一定的风险，因为费用和收益估计可能有错，显然赌博的风险最大，尽管三种不同投资可能有相同的内部偿还率。然而投资者还希望对风险做比较，风险越大，希望的内部偿还率越高。

分析员必须把风险告诉管理部门。通常使用的方法是对每个不可靠的费用或收益提供乐观的估计，悲观的估计和最可能的估计。这三种估计分别加权为：20%，20%和60%。这种加权处理方法在全部费用-收益计算中使用。另一种选择是利用最坏情况的数字完成全部费用-收益分析（极大的费用和极小的收益），实际结果不可能比它更坏。在给出了一个合理的危险评估之后，管理部门就能够作出明智的决择。

5．估计费用

一般来说，对整个系统的费用估计比对它的组成部分的估计更困难一些。我们先计算每个部分的费用，然后再把这些结果加起来。对多数系统来说，大部分费用都集中在少数几个关键部分中。通过仔细地估计这些部分的费用，就可能以最少的代价比较准确地估算出系统的总费用。事实上，若能够认识占系统费用80%或90%的少数几个部分，并做精确的估算，那么对其他部分只需简单和粗略地估计或猜测就足够了。

历史数据是估算费用的重要信息源。努力找出已有的和类似的项目所耗费的资金，就能发现若干可供借鉴的计算费用的准则。当然，不能把当前系统的具体费用作为计算新系统费用的标准或依据。

另外，一些规律和标准也是重要的信息源。例如，维护费用在实际费用中所占的百分比是固定的，比如说20%。又如，人们常常用程序费用的10%作为程序测试和排错的费用等。

工作人员的费用是时间的函数。通常，如果我们能估计做一件工作所需的时间，就能估计出它的费用。例如，根据历史数据，程序员平均每天可编写（包括测试和排错）10行代码。一个中等程序一般有50个模块，每个模块平均有50行代码，每个模块需要5个程序员工作日，每个工作日的费用为100元。因此，这个程序的费用为 $50\times5\times100$元 $=25,000$元

§1.3 数据流图

1.3.1 什么是数据流图

1．数据流图的定义

数据流图是系统的逻辑模型。这个模型不依赖于硬件、软件、数据结构、或文件组织，在数据流图中没有物理含义。换句话说，数据流图是便于用户理解的数据流程的图形表示。它是分析员与用户之间非常好的通信工具，也是进一步系统设计的出发点。

2．数据流图的元素

数据流图由以下四种基本元素组成：

数据流（用箭头表示）；