会计电算化软件分析与设计

蔡传勋　主编

东北财经大学出版社

KUAIJIDIANSUANHUA　RUANJIANFENXI　YU　SHEJI

会计电算化软件分析与设计

蔡 传 勋 主 编
苏显阳 王孝忠 副主编

东北财经大学出版社

(辽)新登字 10 号

会计电算化软件分析与设计

蔡传勤　主编

苏显阳　王孝忠　副主编

东北财经大学出版社出版发行(大连黑石礁)
辽宁省新华书店经销　　沈阳市第二印刷厂印刷

开本:850×1168 1/32　　印张:16 3/4　　字数:420 000
1996 年 2 月第 1 版　　1996 年 8 月第 2 次印刷

责任编辑:　田世忠　丛树林　　责任校对:丛树林

印数:8001—12000

ISBN 7 - 81044 - 033 - 0/T · 29　　定价:19.80 元

内容简介

系统分析与设计是会计信息系统开发的重要方法。本书以软件工程技术为主体，从会计信息系统开发应用出发，比较系统地、完整地介绍了结构化系统分析、结构化系统设计、结构化程序设计和软件测试的基本思想、原则、方法和各种实用工具。同时，结合会计核算软件开发，较详细地全面地介绍了运用软件工程技术进行系统分析与设计的实例，介绍了会计电算化的管理。

全书共十二章，前四章系统地介绍了软件工程的基本技术，后七章是运用软件工程技术对工资、固定资产、材料、成本、销售核算子系统、帐务处理及会计报表子系统结构化系统分析和设计的实例，最后一章为会计电算化管理。各章均附有复习思考题。

本书内容系统、翔实、体系完整、深入浅出、简明易懂、实用性强，也便于自学。可作为大专院校财务计算机管理专业、会计电算化专业及上述专业的自学考试教材，也可作为会计专业、管理专业会计电算化、数据处理等课程的教材或教学参考书。对于从事会计电算化软件设计、维护和管理工作者也有阅读参考价值。

前　言

近几年来，我国会计电算化工作迅速发展，成绩喜人。商品化会计软件，定点开发和行业推广的会计软件并举，越来越多的企事业单位将计算机应用于会计实务，使广大财会人员从过去繁忙的手工记帐、算帐、报帐中解放出来。会计电算化提高了会计核算的及时性和核算结果的准确性，并充分利用会计信息，为领导经营管理、分析、预测、决策提供了数量更多、质量更高的会计信息。会计电算化有力地推动了企业管理工作水平和经济效益的提高。

电算化会计信息系统是企业管理信息系统(MIS)的一个组成部分，它涉及面广、结构复杂、逻辑(勾稽)关系严密，及时性、安全性、可靠性、准确性要求高，是MIS中最复杂、数据处理量和提供信息量最大的一个子系统。建立电算化会计信息系统，会计电算化的普及与提高，需要高质量的会计软件和大量不同层次的人材，特别是高层次人材。软件是智慧的代名词。高质量的会计软件需要优秀的软件设计者，会计软件的持久应用也离不开称职的软件维护人员。他们应该是既精通会计业务、又精通计算机和计算机数据处理技术的复合型人材。培养和造就大批既懂会计业务又懂计算机应用的复合型人材是会计电算化的迫切需要。根据财政部要求，到2000年在城市单位工作的会计人员有60%～70%接受会计电算化初级知识的培训，掌握会计电算化的基本操作技能；有15%接受中等专业知识的培训，基本掌握会计软件维护的技能；会计电算化的高级人材争取达到5%左右。这是历史赋予广大财会人员任重而道远的使命。

本书是为培养会计电算化中高级人材编写的，以会计软件开发技术为主，带有大量实例，通俗易懂，只要学过程序设计的人都能学懂。

软件工程技术对于设计小规模软件来说用处不大，但对开发大、中型软件来说则必不可少。研制、开发、维护结构复杂的会计软件必须使用软件工程技术。结构化系统分析与设计是目前软件工程技术的主要方法。因此在前面四章中较系统地、完整地介绍了结构化系统分析、结构化系统设计、详细设计、结构化程序设计及软件测试的基本思想、原则、方法及许多实用工具，为读者能系统全面地理解和掌握软件工程技术打下较扎实的基础。其后七章则是结构化分析与设计技术在开发会计核算子系统中的具体运用，这部分比较详细地提供了对这些子系统分析与设计的方法、过程和资料。会计电算化的管理是目前会计电算化工作的薄弱环节，因此最后介绍了会计电算化管理工作原则和一些行之有效的规章制度。

本书编者从事软件工程与会计电算化教学和科研工作多年，曾为一些企事业单位成功地开发过会计软件，因此本书是作者教学与实践的总结，书中实例都是从编者设计过的软件中筛选出来的。考虑到是会计电算化与财务计算机管理专业的主干专业课教材，读者有较高的专业知识，因此提供的实例，虽然在某些方面作了简化，但仍然很接进企业实际。而且资料也较完整、详细，实用性强，对于从事会计电算化工作的人员也有参考价值。

软件的质量主要是由设计决定的，我们的课程也是《会计电算化软件分析与设计》，加上我们提供的实例也比较大和复杂，所以书中没有提供源程序，但给出了主要文件及许多模块算法详细设计的问题分析图(PAD图)。读者可以根据自己程序设计的能力选择一些模块编写程序，也可结合本单位实际作适当修改并进行程序设计。编者认为只要认认真真地去做，肯定能使自己程序设计的

能力迅速提高。

本书共十二章，由蔡传勋主编，苏显阳、王孝忠为副主编。其中王孝忠编写了第六章，苏显阳编写了第五章和第十二章，其余各章由蔡传勋编写，刘重也编写了部分内容，最后由蔡传勋对全书进行了修改总纂。

本书编写过程中编者得到了东北财经大学会计系许多教师的帮助和支持；特别是注册会计师，原大连机床附件厂总会计师徐凤刚先生为编者提供了许多资料和帮助；东北财经大学出版社办公室主任黄玉鑫女士对于本书出版做了许多工作，在此一并表示感谢。

限于作者水平与经验，加之时间仓促，书中难免纰错，作者恳请有关专家和读者批评指正。

编　者

1996 年 1 月

目　　录

第一章　软件开发概述

电子计算机是一种能自动、高速、精确地完成信息存贮、数值计算、过程控制和数据处理的现代化电子设备，它由硬件和软件两部分组成。硬件是可以感知的设备，用于信息收集、存贮、加工处理和输出；软件是指挥、协调计算机按人们设想的步骤工作的各种程序及其技术文档。计算机的使用不同于一般机器，需要二次开发即软件开发，软件的功能通过硬件表现，硬件是基础，其性能的发挥又依赖于软件的开发，所以软件是灵魂，两者相辅相成，缺一不可。

软件特别是规模较大的软件结构复杂、逻辑严密、设计工作量大，开发是很不容易的，往往需要许多人合作且费用高、周期长。如果没有好的方法指导软件开发往往事倍功半，不但效率低，质量得不到保证，也难于控制和管理，甚至常常中途夭折，从历史上看曾经出现过软件危机。“软件工程”正是有效摆脱软件危机，大幅度提高软件生产效率和质量，指导软件生产的方法和技术，20 多年来得到了迅速发展。

§1.1　软件和软件工程

早先，人们仅仅把程序理解为软件。所以软件是抽象的逻辑性的无形产品，逻辑上很复杂，需要不断地修改，因为不是实物性的，所以“容易”修改。比如，好端端的一个或一组程序，只要更改、删除其中一个或若干个语句，其功能就可能发生很大的变化或导致不能正常运行。由于软件结构复杂，“牵一发动全身”，真正要修改好

程序又极其困难。复杂一点的软件又难于测试出绝大部分错误，其生产周期长，时间进度、开发费用和质量也难于管理和控制。另外，软件开发失败时往往又无可挽回，常常要推倒后从新设计。

1954 年美国通用电气公司设计软件，第一次在电子计算机上处理职工工资，开创了用计算机数据处理的新时代。60 年代以来，计算机硬件生产技术高速发展，机器功能和性能日趋完美，而价格急剧下跌。价格的下跌，意味着可以大量使用，于是计算机应用领域不断扩大。同时，软件的规模越来越庞大，项目难度增加，结构更加复杂，功能上需要不断修改、扩充和完善。然而，早先非规范化的凭个人经验、爱好、单枪匹马或作坊式的开发方法面临着困境。由于没有先进的技术指导，软件开发人员与用户知识结构又相差甚远，缺乏共同语言。设计人员对用户要求的了解往往相当模糊、粗糙和不全面，又急于编写程序。结果欲速则不达，造成大量返工或中途夭折，许多开发出来的软件因不符合用户要求，质量差、使用不可靠、不方便或维护非常困难，所以使用寿命短，没有生命力，而且开发周期长、费用高，生产效率十分低下，甚至影响了计算机应用的推广和普及。

如何以科学的理论和技术来指导软件生产，用较低的成本，较短的时间开发出能满足用户要求的高质量软件，为了解决这些问题，许多学者深入研究，软件工程正是在这种背景下产生的。60 年代末有些学者提出，提高软件生产效率的途径是软件开发也要“工程化”，即参照、借鉴机械工程，建筑工程中一些行之有效的方法和技术来指导和管理软件开发，变软件产品“无形”为“有形”，用适当的工具表达用户需求的模型，即先抽象出逻辑概念模型，得到用户确认后再转化（设计）为具体的物理模型，最后再编写程序并测试等等。有了正确的理论指导，许多技术和工具应运而生，软件工程学的理论和技术日趋成熟。

软件工程的要点是：

1. 面向用户的观点，使用户参与并了解系统，而不是首先考虑设计方便。

2. 严格区分工作阶段，每一阶段任务明确，工作有条不紊，工作成果要形成规范的技术文档，即用一些工具及文字资料表达出来，作为下阶段工作的起点。划分工作阶段能简化每一步工作的内容，使软件的复杂性较易控制和管理。

3. 复审技术，即对每一阶段的工作成果进行严格的审核，发现错误及时修改，以消除隐患，确认后才进行下一步工作。严格的复审似乎增加了工作量，会延误开发时间，其实则不然。因为错误发现得越早，排除它所化费的代价亦越小，尽早清除设计中的错误和隐患，也可有效制止错误的扩散，避免了今后不必要的返工，使损失最小，也使系统质量得到保证，所以反而加快了开发进度。

§1.2　软件的生存周期

与其它工业产品都有自己的生存周期一样软件也有生存周期。一个软件从开始计划起直到被废弃止称为它的生存或生命周期。软件的生存周期包括计划、开发与运行三个时期，每一时期又可细分为若干更小的阶段。软件生存周期的划分必须适应软件生产工程化的需要，用模型来表达，可归结为传统的瀑布模型和后来兴起的原型模型。目前普遍使用的是瀑布模型，下面先介绍这种模型。

软件生存周期的瀑布模型如图 1—1 所示。由图可见，瀑布模型中各阶段工作顺序展开。

一、计划时期

计划期的主要任务是分析用户要求和论证开发该系统的可行性。用户因对计算机应用不熟悉，提出开发电算化会计信息系统的目标和要求往往很笼统、不具体、也不完整。因此，在系统开发项目

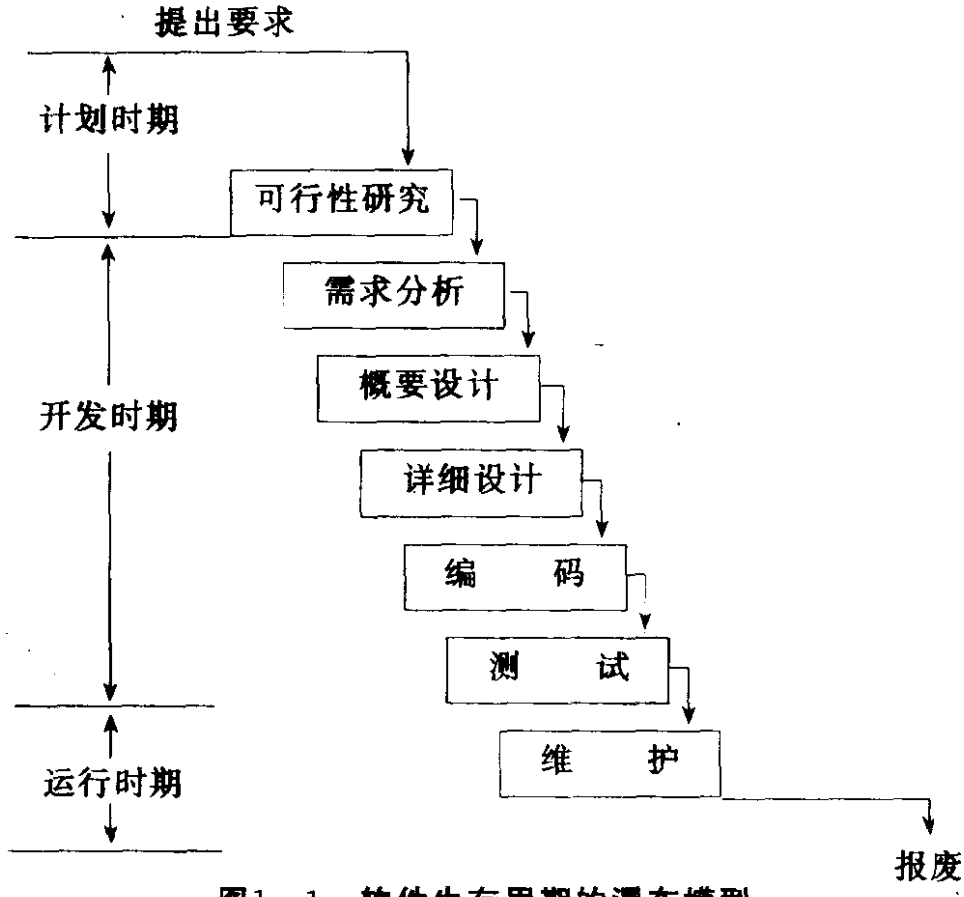

图1—1　软件生存用期的瀑布模型

确立之前，要进行初步调查，并在调查的基础上对该系统开发的必要性和可能性以及可能的候选方案进行分析与评估，为上级领导正确决策提供科学依据。若结论认为该项目值得开发，则应提出一种或数种候选方案，并制定项目初步实施计划，否则应提出终止开发该项目的建议。

二、开发时期

开发时期的任务是完成软件的设计和实现，前者又细分为需求分析、概要设计和详细设计三个阶段；后者又细分为编码(程)和测试两个阶段。每一阶段的任务明确，目的是在开发初期让软件人员全力以赴地搞好系统的逻辑模型和物理模型的设计。

1. 系统分析

又称需求分析，其任务是搞清楚用户对新系统的全部需求，并用工具准确地无二义地表达出来，即提出新系统的逻辑模型。采用

结构化系统分析方法时所用的工具有数据流图、数据字典、加工逻辑说明、数据立即存取分析等。这些文档资料即需求说明书，它既是新系统逻辑模型的描述，又是下一步物理设计的依据，还是将来验收的主要技术依据。软件的需求分析应该在用户和软件人员之间往复讨论和修改，使之逐步趋于准确和完善。

2. 概要设计

又称总体设计，其任务是确定软件的总体结构，即对系统进行分解，由数据流图导出并优化成由模块组成的软件结构图。结构图表达了该系统由哪些模块组成，每一个模块（非底层模块）管理哪些模块，即表达模块之间的调用关系和层次关系，以及模块调用中传递的参数，故又称模块结构图。概要设计的文档资料是模块结构图及模块说明书。

3. 详细设计

详细设计是对每个模块的算法设计，即确定模块内部的过程结构，详细描述实现该模块功能的算法和数据结构。会计信息系统要存贮大量信息，许多数据用代码表示，故详细设计还包括数据库及文件设计和代码设计。详细设计的表达工具有程序流程图、问题分析图、N—S 图或伪码等，其文档是各种设计说明书。

4. 编程

编程即编写源程序，也称编码，即用选定的程序设计语言或数据库管理系统把模块的过程性描述转换成源程序。由此可见，按软件工程开发软件，在这一阶段才真正编写程序。由于在编码之前已完成了大量设计，模块（程序）的输入、输出、使用的文件或数据库以及处理过程已很明确，因此，相对于前面阶段的工作来说，编程就比较简单了，其工作成果是可以上机运行的源程序。

6. 测试

测试是系统开发时期的最后一个阶段，通常又细分为单元（模块）测试、集成测试和验收测试。测试并非为了证明程序正确而进

行，而是为了检查出程序（模块）中、模块之间（接口）及系统中的错误，然后修改之。所以，测试是保证软件质量的重要手段。对较大规模的软件来说，彻底的测试（即系统中不含任何错误）是没有的。测试分静态测试（读并复审源程序）和动态测试（上机运行中调试）。单元测试常用的方法有黑盒法和白盒法；集成测试又有增殖（渐增）式测试和非增殖式测试，前者又可分为自顶向下、自底向上和混合式测试。验收测试是确认和验收软件是否符合软件需求说明书描述的对软件功能的要求与动态特性要求，以及整个系统能否协调、可靠地运行。测试中需设计大量的测试用例和某些辅助测试的虚模块（用于渐增式测试的驱动模块或桩模块），最后还要进行试运行。测试阶段的文档称为测试报告，包括测试计划、测试用例设计与测试结果。

6. 运行时期

为确保会计软件的合法性、安全性、可靠性、正确性、可变性、科学性和实用性，通过测试后的系统要进行一定时间的试运行，即对完整的会计业务，由手工和计算机并行进行处理以验证系统的质量。试运行后还要由财政部门进行评审，以确认该软件是否达到了会计软件的基本要求。评审通过后的会计软件才能正式使用，于是进入了软件生存周期的最后一个阶段，软件人员在这一阶段的工作是做好软件维护。如前所述对会计软件这样规模较大的软件来说，彻底的测试是不存在的，即使进行过严格的测试，免不了还有错误。这些错误因没有遇到适当的机会而隐蔽着，在正式使用中一旦遇到机会就会暴露出来。何况像会计这样的管理软件，使用环境或核算方法等还会有变化，甚至使用一段时间后，用户可能会提出某些新的功能，需要对系统进行扩充或完善，这些都需要不断地修改程序进行维护，不断维护才能延长软件的使用寿命。对会计软件进行修改应遵守规定的程序，填写和更改好有关文档资料，所以运行阶段的工作成果是改进后的系统。

软件的整体质量体现在维护的难易程度上，容易维护的软件一般来说其寿命就长，反之则短。当然，软件使用寿命的长短，并非仅仅取决于软件的质量，还与环境的变化程度，比如核算方法、管理模型等的变更，计算机的性能如处理速度、存贮容量等因素有关。当企业的会计信息处理量已超过系统的处理能力，或者环境变化太大，再作常规的维护已不能满足用户的新要求时，那么现用的系统便要报废，即软件的一个生存周期结束。这时用户又会提出开发新的信息系统的要求，于是软件又进入了另一个生存周期，因此，软件的生存周期是周而复始的。

上述2、3阶段的工作统称为系统设计，4、5阶段的工作统称为系统实施。在系统设计阶段还要进行系统硬件、系统软件（支撑软件）的配置设计；在系统实施阶段还要编写用户使用手册、维护手册以及组织人员培训等工作。

软件开发需要不同层次的技术人员，有系统分析员、系统设计员（高级程序员）和初级程序员。系统分析员是软件开发项目的负责人，负责可行性研究、需求分析和整个项目，因此应该既懂计算机硬件、软件及数据处理技术，通讯技术，又要懂管理业务（如会计）；系统设计员要精通计算机系统和数据处理技术，负责概要设计和详细设计工作，在我国又常常由系统分析员兼任；程序由初级程序员（又称编码员）编写，测试则由编码员及系统设计员及系统分析员共同进行，当然他们的工作各有所侧重。

§1.3　快速原型法

一、快速原型法的产生

传统的瀑布模型开发方法始于60年代末，这种方法一直强调对系统的充分理解（用户需求）、严格定义、预先说明及全面性、准确性的作用。它要求软件人员在设计之前，即开发初期就对整个系