

第3章

系统开发的组织管理与系统开发方法

本章学习目标

- 根据企业的现状和待开发系统的特点,选择适当的开发策略和开发方法。
- 了解管理信息系统的生命周期理论。
- 理解每种开发方法的开发过程,并能够运用结构化系统开发方法进行系统开发。

随着管理信息系统应用的逐渐普及,系统功能从最初的简单数据处理发展到用于支持管理者决策,其在组织中的作用越来越被认可。开发什么样的系统?采用什么样的步骤开发系统?由谁来开发系统?是不是只要会编写程序,就能开发出好的系统?……这系列的问题摆在了管理者的面前,管理信息系统的开发是复杂的系统工程,这个过程涉及的问题很多,如选择开发方式、开发策略和开发方法,合理组织开发工作的进程等诸多问题。

3.1 系统开发的组织管理

系统开发的组织管理主要是根据系统开发的条件和原则,选择适当的开发策略和开发方式。

3.1.1 系统开发的条件与原则

1. 系统开发的基本条件

1) 迫切的开发需求

组织现在使用的系统可能是手工系统,可能是计算机系统。随着组织自身的发展和变化,现行的系统可能会出现问题,当用户逐渐感觉到当前的系统不能满足要求时,便产生了开发新系统的需求。

2) 配备各方面人才

成立由组织高层管理者牵头,各部门负责人、用户单位信息技术主管和系统开发的项目负责人组成的开发领导小组。参与管理信息系统建设的专业技术人员包括项目主管、系统分析人员、系统设计人员、程序员、系统维护人员。

3) 资金支持

信息系统的建设内容复杂,属于智力密集型项目,需要足够的资金支持。信息系统开发过程中用到资金的地方很多,如开发费用、购置计算机软硬件费用、布置机房的费用、人员培

训费用等,企业要有承受这项高额投资的经济实力。

4) 规范化的管理

如果组织管理落后,没有完善的规章制度、科学的管理方法、完善准确的原始数据,这样的组织即便开发了先进的系统,系统也无法发挥最大的功效,甚至无法使用。因此,良好的管理基础是建设管理信息系统的必要条件。

5) 先进的技术

系统的开发要运用先进的工具,跟上软硬件技术、数据通信和网络技术的发展,合理应用这些技术手段,最大限度地挖掘系统的功能,提高系统的质量。这是成功开发信息系统的基本条件。

2. 系统开发的指导原则

系统开发有以下五个指导原则。

1) 目的性

在开发系统之前,首先确立系统目标,目标的正确与否直接影响系统开发的成败。管理信息系统服务于组织的管理过程,是给各级管理人员与业务人员使用的,其建设应该服从组织的整体目标和管理职能的需要,满足用户的信息需求。

2) 整体性

管理信息系统是由人、计算机软硬件组成的综合体,其功能不是各组成部分功能的简单叠加,要从整体出发,处理各功能模块之间的相互联系,注重功能和数据上的整体性,争取达到“1+1>2”的效果。

3) 实用性

管理信息系统要满足用户的要求,用户不仅要求系统功能正确,还希望系统拥有实用的用户界面、便捷的操作性和良好的安全性。许多信息系统失败的原因并不在于技术落后,而是因为系统缺乏实用性。

4) 经济性

管理信息系统的建设是投资巨大的系统工程,引进管理信息系统的复杂程度比引进一般生产设备的复杂程度还要高。在系统开发过程中,应该合理分配有限的资源,充分发挥人机结合优势,使系统的效益与费用比最大化。

5) 易维护性

管理信息系统应该结构清晰,各组成部分之间逻辑关系简单,为今后的发展留有接口,容易辨识和记忆,容易扩充功能、改进性能。

3.1.2 系统开发策略

信息系统的开发策略可分为以下几种。

1. “自下而上”的开发策略

“自下而上”的开发策略强调从基本业务入手,自下而上逐层建立管理信息系统。从组织的日常业务处理出发,先实现基层业务子系统,再实现上一层系统,这样逐层向上合并,不断增加管理/控制功能,最后将各部分组合成完整的系统。

“自下而上”的开发策略可以很好地避免一开始就研制和建立完整系统可能引发的不协调,开发灵活、快速,缺点是没有从整体上考虑问题,可能导致系统功能不完善、各子系统之间的接口被疏忽、数据的冗余和不一致问题。这种策略主要适合小型信息系统和局部信息系统的开发。

2. “自上而下”的开发策略

“自上而下”的开发策略强调由整体到局部、从上到下开发信息系统。先从整体出发,根据组织的总目标规划系统,确保全局的正确,然后确定完成任务需要实现哪些功能。在总体规划指导下,将各项具体的业务放到整体中考察,自上而下地将系统划分为若干子系统,再把子系统进一步分解成功能模块,这样逐层向下分解系统,最后实现系统开发。

“自上而下”的开发策略强调开发过程的整体性,能很好地协调组织内部的各种关系,使信息系统更具有实用性。这种策略工作量大,系统开发费用较高,适用于大型管理信息系统的开发。

3. 综合开发策略

在实际开发过程中,“自上而下”和“自下而上”的开发策略并不互相排斥,可以将两者综合起来使用。“自上而下”的开发策略适用于制定系统开发总体方案,进行系统的规划和分析;“自下而上”的开发策略适用于系统的设计和实施,按照自下而上的策略完成一个一个小模块,再将模块连接成大一些的模块,自下而上逐步构成整个系统。综合的开发策略,既保证系统整体性,又可以减少资源的消耗,得到一个比较理想的新系统。

3.1.3 系统开发方式

管理信息系统的开发方式有自行开发、委托开发、合作开发、购买商品化软件和应用软件包二次开发五种主要方式。这些方式都具有各自的特征,用户需根据自身的状况,主要是项目规模、技术力量和资金情况,选择适合的开发方式。

1. 自行开发

这是一种依靠组织自己的技术人员开发系统的方式。使用这种开发方式的组织通常拥有经验丰富的技术团队,这个团队中既有出色的领导,又有经验丰富的系统分析人员、系统设计人员和程序员。

自行开发的主要优点有:

- (1) 组织对自身的需求比较了解,可以将自身的管理思想融合到软件中去。
- (2) 开发过程沟通方便,实施的成功性更高些。
- (3) 经过完整的系统开发过程,本单位技术队伍得到锻炼。
- (4) 充分了解开发过程,拥有源代码,方便维护系统。
- (5) 开发费用少。

自行开发的主要缺点是:

- (1) 对组织中技术人员的能力要求高。
- (2) 技术人员的开发经验与软件公司的专业技术人员相比有一定的差距,将会导致系

统的不规范,难以摆脱组织长期以来形成的管理方式,不易开发出一个高水平的系统。

(3) 系统开发完成后,容易造成技术人员资源闲置。

2. 委托开发

委托开发是现在非常流行的软件外包的内容之一。外包源于 20 世纪 80 年代后期,是企业管理中的一种经营策略和管理手段,软件开发外包是目前软件行业的一大趋势。软件外包可以把整个开发工作委托给有开发经验的其他单位,也可以将部分开发工作外包出去。在选择开发单位时,应该考察其是否具有相关项目的开发经验。

委托开发的主要优点是:

(1) 对组织的开发能力要求不高,组织本身可以不必拥有自己的开发队伍。

(2) 虽然开发过程的一次性投资比较大,但是和成立专门的部门、招聘大量的信息技术专家相比较,成本还是较低的。

(3) 接受委托的科研单位或软件公司通常具有丰富的项目开发经验,开发出的系统规范性较好。

委托开发的主要缺点是:

(1) 开发方需要有充分的时间了解组织的情况,即使经过充分调查也可能存在盲点。

(2) 由于系统技术完全被开发方所掌握,系统的风险性高,对系统安全性要求高的企业不适合这种方式。

(3) 系统维护工作困难。

(4) 当系统需要升级或功能发生变更时,对开发方的完全依赖将导致维护成本升高。

3. 合作开发

合作开发是指用户和拥有丰富经验的第三方机构共同开发系统。通常,在组织开发能力较弱的情况下,组织通过与有经验的单位合作开发信息系统,让组织自身的员工也参与到系统的建设中,培养自己的技术骨干。组织选择开发伙伴时,重点考虑技术实力和类似系统的开发经历。

合作开发的主要优点是:

(1) 充分发挥双方的优势,开发方技术力量强,用户对管理业务熟悉,双方合作可以开发出既具有较高水平适用性又强的系统。

(2) 通过参与系统开发过程提高组织中的技术力量,便于系统维护。

合作开发的主要缺点是:在合作开发过程中,双方技术人员容易各自为政,达成共识需要更多的沟通。

4. 购买商品化软件

在信息系统的功能简单、常见的情况下,可以考虑购买商品化软件。例如,工资管理系统是每个组织都会用到的,基本功能比较相似,这时可以考虑从市场上购买成型的商品化软件。

购买商品化软件的主要优点是:

(1) 节省系统开发的时间。

- (2) 商品化软件专业化程度高。
- (3) 软件产品上市之前经过了专业的测试,性能稳定,安全性好。
- (4) 购买成本比开发成本低。

购买商品化软件的主要缺点是:

- (1) 商品化软件环境适应性差,难以完全贴合组织的实际需求。
- (2) 容易受到供应商企业稳定性的影响,例如,供应商经营不善、倒闭将有可能导致难以获得售后服务。
- (3) 系统维护困难。

5. 应用软件包二次开发

应用软件包是指与某一应用领域有关的、由开发商预先编写好的、完成特定任务的一组程序。很多组织之所以选择购买软件包是因为在某些管理上的应用是相似的,例如高校学籍管理、人力资源管理、库存管理这些职能的软件包已经非常成熟,并被广泛应用。这些应用软件包都是以特定的管理流程为基础的,虽然包含了一些从各行业中提炼出的通用流程,但每个组织的管理流程不完全相同,管理中有些“不通用”的东西,需要针对组织的自身特点进行二次开发,修改相应的功能才能顺利应用。

应用软件包二次开发的主要优点是:

- (1) 无须进行大规模的系统分析与设计工作,可以根据组织的需求进行补充设计,为组织量身定制系统。
- (2) 软件包产品在上市之前已经有厂家进行充分测试,系统的稳定性和安全性要比自行开发的系统优越。
- (3) 能在软件中真正体现、固化组织的核心理念。

应用软件包二次开发的主要缺点是:

- (1) 根据组织的特殊需求额外定制的功能,成本可能会很高。
- (2) 购买的软件包和其他子系统整合时发生的问题较多。
- (3) 需要软件开发商参与开发,没有开发能力的客户本身难以进行此种开发。

3.2 管理信息系统的开发方法

管理信息系统的开发是一项艰巨的工作,需要大量的人力、财力和时间的投入。系统开发的效率、质量、成本及用户的满意程度,除了管理和技术方面的因素外,很大程度上取决于系统开发方法的选择。传统的系统开发方法,在编程前不重视系统的分析与设计,不重视开发文档的完善与管理。20世纪70年代以来,西方在经历了“软件危机”以后,一些国家开始重视系统开发方法的研究,提出了许多新的系统开发方法。目前,常用的管理信息系统开发方法有:结构化系统开发方法、原型法、面向对象的开发方法和计算机辅助软件工程(CASE)方法。

3.2.1 结构化系统开发方法

早期的管理信息系统开发缺乏规范性、开发过程随意性大、无规律的开发逐渐成了信息

系统发展的瓶颈,于是人们开始探寻科学的系统开发方法。在这种背景下,结构化系统开发方法产生了。

1. 基本思想

结构化系统开发方法是建立在信息系统生命周期理论基础上的。

任何事物都有产生、发展、成熟、消亡(更新)的过程,信息系统也不例外。信息系统在使用过程中随着其生存环境的变化,要不断维护、修改,当它不再适应的时候就要被淘汰,就要由新系统代替老系统,这种周期循环称为信息系统的生命周期。信息系统的生命周期及相应的工作步骤如图 3.1 所示。

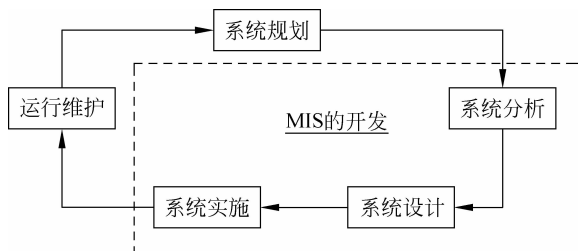


图 3.1 MIS 的生命周期图

从图 3.1 可见,信息系统的生命周期可以分为系统规划、系统分析、系统设计、系统实施、运行维护五个阶段。

结构化是指严格的、可重复的、可度量的,也就是将复杂的系统开发活动分解成一系列连续的步骤,每一步都建立在上一步的基础上。这种方法从整体出发,遵循标准化与模块化的设计思想,按照面向用户的原则,自上而下分析和设计系统,并自下而上地实施系统。具体地说,结构化的方法用一系列规范的步骤和工具进行系统开发,按照这种方法将系统开发过程划分为若干个阶段,每个阶段由若干个工作步骤组成,每个步骤都有其明确的目标和任务及预期要达到的阶段成果,步骤之间的次序不可以打乱。

2. 开发过程

结构化开发方法把系统的开发分为五个阶段,并明确规定了每个阶段的目的、工作内容、使用的方法和工具、参与人员及前后阶段的衔接关系等。

1) 系统规划阶段

系统规划阶段的任务是对企业的环境、目标及现行系统的状况进行初步调查,根据企业目标和发展战略,确定信息系统的发展战略,对建设新系统的需求做出分析和预测,同时考虑建设新系统所受的各种约束,研究建设新系统的必要性和可能性。根据需求和可能,给出拟建系统的备选方案。对这些方案进行可行性分析,写出可行性分析报告。可行性报告审议通过后,将新系统建设方案及实施计划编写成系统设计任务书。这个阶段弄清楚“要解决什么”的问题。

2) 系统分析阶段

系统分析阶段的任务是根据系统设计任务书所确定的范围,对现行系统进行详细调查,描述现行系统的业务流程,指出现行系统的局限性和不足之处,确定新系统的基本目标和逻辑

辑功能要求,即提出新系统的逻辑模型。这个阶段又称为逻辑设计阶段,是整个系统建设的关键阶段,也是信息系统建设与一般工程项目的重要区别所在。这个阶段的工作主要是解决系统“做什么”的问题。

系统分析阶段的成果体现在系统说明书中,这是系统建设的必备文件。它既是给用户看的,也是下一阶段的工作依据。因此,系统说明书要既通俗又准确。用户通过系统说明书可以了解未来系统的功能,判断是不是其要求的系统。系统说明书一旦讨论通过,就是系统设计的依据,也是将来验收系统的依据。

3) 系统设计阶段

系统分析阶段的任务是回答系统“做什么”的问题,而系统设计阶段要回答的问题是“怎么做”。该阶段的任务是根据系统说明书中规定的功能要求,考虑实际条件,具体设计实现逻辑模型的技术方案,即设计新系统的物理模型。这个阶段又称为物理设计阶段,分为总体设计和详细设计两个阶段,本阶段的技术文档是“系统设计说明书”。

4) 系统实施阶段

系统实施阶段是将设计的系统付诸实施的阶段。这一阶段的任务包括计算机等设备的购置、安装和调试,程序的编写和调试,人员培训,数据文件转换,系统调试与转换等。这个阶段的特点是几个互相联系、互相制约的任务同时展开,必须精心安排,合理组织。

系统实施是按实施计划分阶段完成的,每个阶段应写出实施进度报告。系统测试后写出系统测试分析报告。这个阶段的工作解决“具体实现”系统的问题。

5) 系统运行和维护阶段

系统投入运行后,需要经常进行维护和评价,记录系统运行的情况,根据一定的规格对系统进行必要的修改,评价系统的工作质量和经济效益。这个阶段的工作解决系统“运行管理”的问题。

结构化的开发方法把系统的开发分为以上五个阶段,如果将系统开发过程比作盖房子的过程,系统规划相当于项目前期调研和规划,系统分析和设计阶段相当于了解现场情况,进行房屋结构设计、画图纸,系统实施相当于施工建造房屋,系统运行和维护阶段相当于房屋的使用和维护。系统开发核心的三个阶段是系统分析、系统设计和系统实施,其实现过程可用图 3.2 表示。

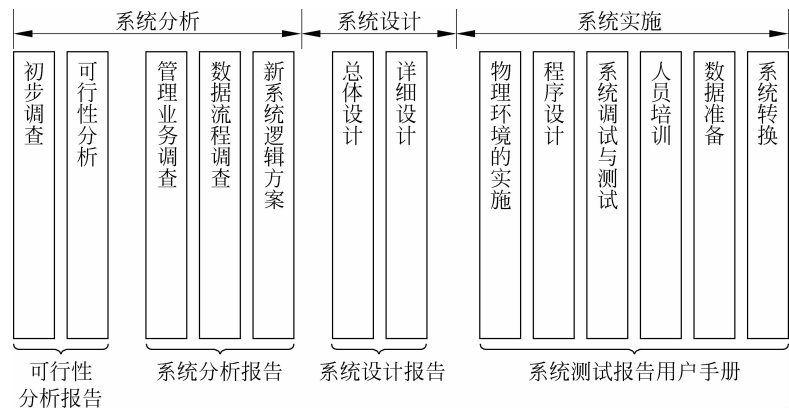


图 3.2 结构化系统开发方法核心三步骤

3. 结构化开发方法的评价

结构化系统开发方法,也称为系统开发生命周期法,它具有如下特点:

(1) 遵循 MIS 生命周期的基本原则,将整个开发工作分为系统分析、系统设计、系统实施三个阶段。

(2) 运用系统工程中解决复杂问题的基本策略——分解协调技术,从事作为社会技术系统的 MIS 开发。

(3) 按照预定准则,用标准工具(图、表)完成和表达各项工作的成果。

但是,结构化系统设计不能帮助系统设计人员建立一个直观的系统模型,使用户在实际得到并使用这个系统之前,就能够知道这个系统是不是所需要的计算机信息系统。用户关心的是这个系统的逻辑功能是否满足其需求,是否能解决要解决的问题。至于这个系统如何实现这些功能,并不是最关心的问题。为了使所设计的系统满足用户的要求,在设计之前,先要正确理解和准确表达用户的要求,这就是系统分析阶段的基本任务。结构化系统分析,强调系统分析员与用户一起按照系统的观点对企业活动由表及里进行分析,调查分析清楚系统的逻辑功能,并用数据流程图等工具把系统功能描述清楚。用户可以判断未来的系统是否满足其功能要求,而系统设计人员根据这些描述进行系统设计,保证系统功能的实现。

结构化系统开发适用于大型系统的开发,其优缺点如下:

- 优点——工作规范,符合工程化要求,具有严格的质量保证体系。
- 缺点——开发周期长,过程较为烦琐。用户在较长时间内不能得到一个可实际运行的物理系统,似乎“听得楼梯响,不见人下来”。

3.2.2 原型法

原型法是 20 世纪 80 年代初兴起的一种开发模式,其动机主要是针对传统生命周期法缺乏弹性的缺陷,为了缩短开发周期,减少开发风险,在限定的时间内,用最经济的方法,开发出一个可实际运行的系统原型,以便尽早澄清不明确的系统需求。在原型系统的运行中用户发现问题,提出修改意见,技术人员完善原型,使它逐步满足用户的需求。对于决策支持系统这种开发性而无固定需求范围的系统,更适合用原型法进行开发。原型法的开发过程如图 3.3 所示。

根据系统的特点、开发原型与最终系统之间的关系,原型可分为抛弃式原型、演化式原型和增量式原型。

1. 抛弃式原型

原型作为开发者和使用者之间的通信媒介,是一种试验工具,根据试验的结论开发新系统,原型只是功能上与实际系统相似,目的达到即被抛弃,是纯粹的“原型”。原型不作为实际系统投入使用,不作为最终产品。抛弃式原型工作重心在于完善需求的过程,而非开发过程。除了应用于确定需求,抛弃式原型也可以用于选择设计方案、评价设计方案或是用于系统测试。

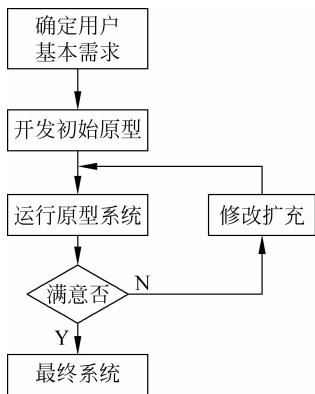


图 3.3 原型法的开发过程

2. 演化式原型

与抛弃式原型相反,演化式原型是按照使用者的基本需求开发出一个原型系统,原型开发完成即可使用,在使用过程中不断地修改、增加新功能,原型逐步演化成最终系统。演化式原型围绕系统的基本需求进行开发,允许在实施过程中不断地修改设计方案,因此必须加强对实施过程的管理和控制,否则容易引起无休止的反复。

3. 增量式原型

增量式原型根据用户的需求,开发出原型系统总体框架,然后不断地修改、完善具体功能,直到用户满意。增量式原型与演化式原型的区别在于增量式原型建立在软件总体设计的基础上,而演化式原型的设计方案是在不断发展的。显然,增量式原型修改的范围较小,与演化式原型相比其应变能力弱一些。

3.2.3 面向对象开发方法

面向对象(Object Oriented, OO)方法是由面向对象程序设计方法(Object Oriented Programming, OOP)发展起来的,面向对象的程序设计为人们提供了更有力的认识框架。这一认识框架迅速扩展到程序设计范围之外,相继出现了面向对象的数据库管理系统、面向对象的系统分析、面向对象的系统设计,逐步形成了一套完整的方法。

1. 基本思想

面向对象开发方法的基本思想可以归纳为以下四点:

- (1) 客观世界的任何事物都是对象(Object)。它们都有一些属性和操作,对象作为一个整体,对外不必公开这些属性与操作,称为对象的封装性。
- (2) 对象之间有抽象与具体、群体与个体、整体与部分几种关系,这些关系构成对象的网络结构。
- (3) 父类对象具有的性质,自然成为其子类的性质,这就是继承性。
- (4) 对象之间可以互送消息。消息可以传递一个参数,也可使对象开始某个操作。

2. 开发过程

面向对象的开发方法以对象为中心,采用面向对象的概念建立系统模型并开发和实现系统。面向对象的开发过程主要由分析、设计和实现三部分组成。

1) 面向对象分析

和其他的系统分析方法一样,面向对象的系统分析从分析用户的需求开始,通过对问题空间的分析,把现实问题分解成一些类和对象,并分析它们之间的关系,最终确定问题域模型。面向对象分析按照下列顺序进行:找出问题域的对象和类;确定结构;识别主题;定义属性和方法。

2) 面向对象设计

面向对象设计是在面向对象分析的基础上,根据用户需求、系统功能和系统环境,把分析阶段得到的对象和类做进一步规范整理,转化成符合要求的、抽象的系统实现方案。面

面向对象设计将数据对象和处理功能一起模块化,大多数系统逻辑上都由四部分组成,对应四个子系统的设计过程,它们分别是问题空间部分的设计、人机交互部分的设计、任务管理部分的设计、数据管理部分的设计。

3) 面向对象实现

面向对象实现包括程序编写和系统测试。程序编写是把面向对象设计结果用某种面向对象的程序设计语言转换成系统,描述类,定义对象的内部功能,实现系统的界面等。测试并调试面向对象程序时,除了使用传统的测试技术外,还可以应用与面向对象程序特点相适应的新测试技术。

3.2.4 计算机辅助软件工程方法

今天,代码变得日益简单,在模型(Model)的指导下、思想、设计、分析都变得异常重要。CASE(Computer-Aided Systems Engineering)——计算机辅助系统工程,原来指用来支持管理信息系统开发的、由各种计算机辅助软件和工具组成的大型综合性软件开发环境,随着各种工具和软件技术的产生、发展、完善和不断集成,逐步由单纯的辅助开发工具环境转化为一种相对独立的方法论。

CASE 并不是真正意义上的开发方法,而是由各种计算机辅助软件组合而成的综合性的开发环境。在高级程序设计语言的基础上,为提高软件开发的质量和效率,利用专门的 CASE 软件工具实现其开发方法,把原来由手工完成的开发方法变成自动化或半自动化开发过程,例如,利用 CASE 工具将 E-R 图转换成数据库模型。

CASE 工具能够减少成本、加速开发,提供用于将来维护和扩展的全面文档。常用的 CASE 工具包括以下几种。

1. 图稿绘制

Visio,这是目前国内用得最多的 CASE 工具。它提供了日常使用中的绝大多数框图的绘画功能(包括信息领域的各种原理图、设计图),同时提供了部分信息领域的实物图。Visio 的精华在于其使用方便,安装后的 Visio 既可以单独运行,也可以在 Word 中作为对象插入,与 Word 集成良好,其图生成后在没有安装 Visio 的 Word 环境下仍然能够查看。Visio 支持 UML 的静态和动态建模,对 UML 的建模提供了单独的组织管理。从 2000 版本后 Visio 被 Microsoft 公司收购,正式成为 Office 大家庭的一员,纳入名门的 Visio 2002 被 Microsoft 的风格所同化,形式上出现了一些华而不实的东西,但是功能不减从前,各种器件模板有了许多增进。

2. 原码浏览的工具

SourceInsight 以工程的方式管理原码,提供非常适合工程的浏览手段。整个面板分成三个部分:左边树提供工程内的所有变量、函数、宏定义,右边提供程序阅读和编辑,下边显示原码涉及的函数或者变量定义。最让人佩服的是,SourceInsight 不仅能高亮显示关键字,而且以近乎完美的版面编排让你看程序如看报纸。SourceInsight 提供函数交叉调用的分析,并以树状的形式显示调用关系。

3. 配置管理工具

配置管理的重要意义在于维护文档的统一和可追溯性。Microsoft 的 Studio 企业版包含的版本管理工具——Visual Source Safe,该工具包括一个服务器和一个通过网络可以连接服务器的客户端。VSS 提供了基本的认证安全和版本控制机制,包括 CheckIn(入库)、CheckOut(出库)、Branch(分支)、Label(标定)等功能,能够对文本、二进制、图形图像几乎任何类型的文件进行控制;提供历史版本对比,可以集成在 Studio 中。

4. 数据库建模工具

ERWin: CA 公司出品的拳头产品,强大的老牌数据库建模工具。它有一个兄弟是 BPWin,这个是 CASE 工具的一个里程碑产品。ERWin 的界面相当简洁漂亮,也是采用 E-R 模型,如果你要开发中小型数据库,极力推荐 ERWin,它的 Diagram 给人的感觉十分清晰。在一个实体中,不同的属性类型采用可定制的图标显示,实体与实体的关系也一目了然。

5. UML 建模工具

Rational Rose,代表 Rational 公司的面向对象分析和设计工具的一款力作。Rose 目前在国内正被越来越多的公司所使用,其原因一方面是随着软件规模的扩大,面向对象分析和设计的优势突现出来,软件企业正在从面向过程向面向对象过渡。另一方面,Rose 集中体现了统一软件建模(UML)的先进设计思想,能够通过一套统一的图形符号简洁有效地表达各种设计思想。当然,Rose 本身在设计上的完善和与 Rational CASE 家族的完美集成也是作为一款最成功的 CASE 产品的基础。

3.2.5 系统开发方法的比较选择

前面介绍多种系统开发方法,没有一种开发方法适合所有类型的系统开发,因此需要根据组织自身条件和待开发系统的特点决定采用何种方法。

1. 结构化系统开发方法是理论严谨、成功率较高的一种开发方法

结构化系统开发方法也称为瀑布法(Waterfall Method)。瀑布法是早期系统开发主要采用的方法。它遵循系统生命周期各个阶段的基本任务,对于科学的系统开发的全过程起到了有效和良好的效果,是系统开发方法的一个里程碑,并且有许多成功的案例。在一段时期内,在很大程度上解决了软件危机的问题。

瀑布法的特点是:按照传统的生命周期法开发系统,各阶段的工作自顶向下从抽象到具体顺序进行,阶段间具有顺序性和依赖性,是串行化的过程。任何一个阶段开始的前提是前一个阶段的成果。瀑布法给出了系统开发的明确定义(即文档加上代码),使系统开发各时期的管理都具有完备性。

结构化方法特别适用于管理基础好、能够较好定义需求的大型系统或复杂系统的开发,例如会计系统、银行系统,这些系统的工作流程清楚、有完善的规章制度。这种方法不适合需求不明确的系统开发,用户对系统需求的描述决定了系统功能,系统功能决定了系统的结构。需求模糊易变将会导致系统功能的变化,功能的变化又会引起系统结构的修改,容易导

致成本上升。

2. 原型法在提高用户满意程度、提高开发速度方面具有突出的特点

原型法的意义是可视化,强化沟通,降低风险,节省后期变更成本,提高项目成功率。一般来说,采用原型法后可以改进需求质量;虽然投入了较多先期的时间,但可以显著减少后期变更的时间;原型法投入的人力成本代价并不大,但可以节省后期成本;对于较大型的软件来说,原型系统可以成为开发团队的蓝图;另外,原型通过充分和客户交流,还可以提高客户满意度。

小型系统或局部系统适合采用原型法开发,如小型工资系统、人事系统等,这类系统功能比较简单,开发人员不多;原型法同样也适用于用户需求不确定的情况。对于大型的系统开发,涉及组织多个岗位、多个流程,开发系统时需要权衡各部门的要求,进行整体设计,没有充分的系统需求分析,很难构造出原型,因此大型系统不适合采用这种方法。另外,存在大量运算和逻辑性很强的模块与批处理系统,也不适合采用原型法。

3. 面向对象的方法近些年越来越多地被人们使用

面向对象的系统开发优点是与人类习惯的思维方式一致,稳定性好,可重用性好,可维护性好。

面向对象的系统开发工作重点应该放在分析阶段。这种方法在开发的早期阶段定义了一系列面向问题的对象,并且在整个开发过程中不断充实和扩充这些对象。由于在整个开发过程中都围绕对象进行,各个阶段使用统一的概念和表示符号,因此整个开发过程是一致的,降低了出错的可能。所以,面向对象的系统开发方法越来越流行。

4. 从严格意义上讲 CASE 方法只是一种自动化的开发技术

严格地说,CASE 只是一种集成化的开发环境,而不能算是完整的开发方法,一般与其他方法配合使用。

在开发系统的过程中,各种方法不是完全互斥的,建设一个系统应结合组织实际情况,灵活地将上述方法相结合。例如,当系统规模较大时,可以把结构化方法和原型法结合起来使用,以结构化方法为主,以原型法作为补充。综合性的系统开发既可以发挥结构化方法严格控制开发过程的特点,又能借助原型法“快”的特征加快系统开发速度。如果开发人员熟练掌握了某种面向对象的开发软件,则可以利用结构化方法分析和设计系统,用面向对象方法完成系统,缩短开发周期。

3.3 思考与练习

简答题

1. 管理信息系统的开发应遵循哪些原则?
2. 开发管理信息系统有哪些基本策略? 如何合理选择?
3. 什么是系统的生命周期理论?
4. 结构化的开发方法分哪几个阶段? 每个阶段有哪些工作内容?
5. 试比较常用的几种开发方法。