

刘 光 刘小东 编著

地理信息系统

二次开发实例教程

——C++ Builder 和 MapObjects 实现

► 本书以“北京市地理信息公众查询系统”为例，介绍了如何使用C++ Builder和MapObjects组件，实现地理信息系统应用程序的二次开发。

——C++ Builder 和 MapObjects 实现

——C++ Builder 和 MapObjects 实现



科学出版社
北京科海电子出版社

地理信息系统二次开发实例教程

——C++ Builder 和 MapObjects 实现

刘 光 刘小东 编著

科学出版社
北京科海电子出版社

内 容 提 要

本书以“北京市地理信息公众查询系统”为例,按照软件工程的思想与要求,介绍如何运用 C++ Builder 语言及地理信息系统二次开发组件 MapObjects,进行地理信息系统(GIS)的二次开发。

全书共分 8 章,首先概述了地理信息系统的软件工程设计方法,从第 2 章开始,以“北京市地理信息公众查询系统”为例,详细阐述了 GIS 系统的需求分析、总体设计、详细设计、主界面实现与主要功能的编码实现。

本书适用于政府、企业相关部门的 GIS 研究开发人员,也适合作为高等院校地理学、地理信息系统、房地产、环境科学、资源与城乡规划管理、区域经济学等专业学生的 GIS 实习教材和参考书,以及各种 GIS 培训学员的教材用书。

图书在版编目(CIP)数据

地理信息系统二次开发实例教程. C++ Builder 和 MapObjects

实现/刘光,刘小东编著.—北京:科学出版社,2004.9

ISBN 7-03-014422-8

I. 地… II. ①刘… ②刘… III. 地理信息系统—程序设计—教材
IV. P208

中国版本图书馆 CIP 数据核字(2004)第 099261 号

责任编辑:何武 / 责任校对:科海

责任印刷:科海 / 封面设计:林陶

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>

北京科普瑞印刷有限责任公司印刷

科学出版社发行 各地新华书店经销

*

2004 年 10 月第一版

开本:16 开

2004 年 10 月第一次印刷

印张:24.25

印数:1-4 000

字数:590 千字

定价:36.00 元(1CD)

(如有印装质量问题,我社负责调换)

前 言

以计算机为核心的信息处理系统技术是二次世界大战后科技革命的主要标志之一。在信息的诸多类型中,与空间相关的信息是十分重要的,人类赖以生存的地球是个三维空间,其中的万物无不与空间位置相关,如何利用计算机处理空间相关信息是地理信息系统(Geographic Information System, GIS)产生和发展的原动力。GIS起源于20世纪60年代,它作为有关空间数据管理、空间信息分析及其传播的计算机系统,在其40多年的发展历程中已经取得了很大成就,并广泛地应用于土地利用、资源管理、环境监测、交通运输、城市规划、经济建设以及政府各职能部门。并且随着计算机技术的不断发展,计算速度越来越快,使得地理信息系统技术应用领域越来越广泛。最近几年来,地理信息系统无论是在理论上还是应用上都处在一个飞速发展的阶段,并呈现出广阔的应用前景。“数字地球”概念的提出,更进一步推动了作为其技术支撑的GIS的发展。不管人们将21世纪称为什么世纪,GIS的广泛应用、普及必将是其一个重要的特征。

今天, GIS 已是一个全球拥有数十万开发人员和数十亿美元的产业。世界各国已设计出大量实用化的地理信息系统,常用的GIS软件已达400多种,比较著名的有美国环境系统研究所(ESRI)的ARC/INFO和ArcView,澳大利亚GENASYS公司开发的GENAMAP,美国Clark大学George Perkins Marsh研究所的IDRISI,中国地质大学开发的MapGIS,原武汉测绘科技大学开发的GeoStar,北京大学遥感与地理信息系统研究所开发的CityStar等等。另外,随着Internet/Intranet的迅猛发展,万维网地理信息系统(WebGIS)软件也开始走向市场,国内的产品主要有成都华好网景科技有限公司的OK Map、武汉测绘科技大学开发的Internet GeoStar(GeoSurf)、国家遥感应用工程技术研究中心网络与运行工程部开发的地网GeoBeans。

虽然GIS软件产品繁多,但是由于GIS软件具有专业性强的特点,它们不可能解决所有的问题,因此,针对某些具体问题,还必须由用户进行二次开发才能解决。为方便用户进行二次开发,各大GIS厂商在推出基础地理信息系统平台的同时,一般都提供专门的语言与二次开发组件,例如MapInfo公司的MapBasic、MapX, ESRI公司的AVENUE、MapObjects,以及RSI公司的IDL、IDLDrawWidget等。我国在这方面主要有北京超图地理信息技术有限公司的SuperMap。虽然目前介绍这方面知识的书籍逐渐开始多起来,但都只是零散地介绍各种组件的功能,至今国内市场上还没有介绍如何利用某一地理信息系统二次开发组件来实现一个功能相对齐全并且实用的系统。这正是本书编写的目的。

本书按照软件工程技术的要求,以“北京市地理信息公众查询系统”为例,介绍了如何利用C++ Builder语言以及地理信息系统二次开发组件MapObjects,进行地理信息系统的二次开发,包括系统的需求分析、总体设计、详细设计、主界面实现与主要功能的编码实现等几部分内容。

第1章“地理信息系统软件工程”讲述了软件工程技术在GIS系统开发中的几个应用方

面,包括需求分析、数据管理设计、用户界面设计、设计模式在GIS软件开发中的应用。在GIS开发过程中应用软件工程技术,可以提高软件开发效率和质量。第2章“需求分析”介绍了“北京市地理信息公众查询系统”的需求分析,包括需求概述、功能需求以及功能需求详细描述。第3章“系统总体设计”介绍了系统平台选择、系统总体框架、系统数据组织及系统开发进度安排等。第4章“系统详细设计”根据系统的总体设计结构分别从北京市地理信息公众查询系统的数据库设计和一些相关类的设计两方面来详细阐述GIS系统的设计。第5章“系统主界面的实现”首先简单介绍MapObjects的功能、特点、结构及其数据源,然后介绍如何设计系统的主界面及主要实现代码。第6章“选择与查询功能的实现”主要介绍如何通过查询与数据集有关的表从数据中获取信息,以及如何通过空间和逻辑的查询方法从数据中获取信息。第7章“系统其他辅助功能”介绍了系统中其他一些辅助功能的实现,例如当鼠标移动到某地物上并稍做停留后,出现一个小标签,显示该地物的名称,以及距离量算、面积量算的实现,等等。为了让读者更加全面地掌握MapObjects的开发,第8章介绍了“北京市地理信息公众查询系统”开发过程中没有涉及到的一些对象,包括投影对象、地址匹配对象、动态跟踪层(Tracking Layer)对象与地理事件(GeoEvent)对象。

本书配套光盘提供了书中涉及的所有源代码,以及实现书中内容所需的部分资料。

由于时间仓促,书中难免有一些错误、遗漏,恳请读者谅解,并提出批评和指正。

编者

2004年6月

目 录

第1章 地理信息系统软件工程	1
1.1 软件工程简介	1
1.1.1 基本概念	1
1.1.2 软件系统开发过程	2
1.1.3 开发过程模型	5
1.2 GIS需求分析	7
1.2.1 需求获取	7
1.2.2 需求规约	8
1.3 GIS数据管理设计	9
1.3.1 全部采用文件管理	9
1.3.2 文件结合关系数据库管理	9
1.3.3 全部采用关系数据库管理	10
1.3.4 采用面向对象数据库管理	11
1.4 GIS用户界面设计	12
1.4.1 界面设计原则	12
1.4.2 GIS界面设计中的要素	14
1.4.3 GIS界面样式	16
1.5 GIS应用模式与开发方式	18
1.5.1 GIS应用模式	18
1.5.2 GIS开发方式	20
1.6 “北京市地理信息公众查询系统”介绍	22
第2章 需求分析	24
2.1 需求概述	24
2.2 功能性需求	25
2.2.1 系统体系结构	26
2.2.2 用户描述	26
2.2.3 具体功能需求	27
2.3 非功能性需求	30
2.3.1 性能需求	30
2.3.2 安全性需求	30
2.4 功能需求详细描述	30

第3章 系统总体设计	41
3.1 系统平台选择	41
3.1.1 硬件平台	41
3.1.2 系统操作平台	41
3.1.3 数据库平台	41
3.1.4 系统开发模式与GIS组件选择	41
3.1.5 开发工具	44
3.2 系统总体框架	45
3.2.1 系统功能框架	45
3.2.2 系统数据库	46
3.2.3 系统的开发结构	46
3.2.4 系统界面组织	46
3.3 系统数据组织	48
3.3.1 系统数据的逻辑组织	48
3.3.2 系统的主要数据类型	49
3.4 进度规划	49
第4章 系统详细设计	50
4.1 数据库详细设计	50
4.1.1 地名分类编码	50
4.1.2 元数据表结构	67
4.1.3 电子地图数据	71
4.2 系统相关类的详细设计	76
4.2.1 辅助类的详细设计	76
4.2.2 TEnvironment类的详细设计	80
4.2.3 TPath类的详细设计	83
4.2.4 NetLayer类的详细设计	84
4.2.5 MapTip类的详细设计	92
第5章 系统主界面的实现	93
5.1 MapObjects简介	93
5.1.1 MapObjects的功能	93
5.1.2 MapObjects的特点	94
5.1.3 MapObjects的结构	94
5.2 导入MapObjects组件	97
5.3 系统主界面设计	98
5.3.1 创建资源	98
5.3.2 设计地图控制工具栏	99
5.3.3 设计地物类型工具栏	102
5.3.4 设计状态栏	104

5.3.5 设计“地图”页面.....	104
5.3.6 设计“查询”页面.....	105
5.3.7 其他辅助控件.....	111
5.4 TEnvironment类的初步实现.....	111
5.4.1 辅助类的实现.....	112
5.4.2 TEnvironment类的成员变量.....	114
5.5 读取元数据.....	116
5.6 “地图”页面的实现.....	123
5.7 图层的加入与控制.....	131
5.7.1 在地图中加入图层.....	131
5.7.2 依据比例尺控制图层显示.....	148
5.8 通过“地图”页面控制地图显示.....	154
5.8.1 控制显示的地物类型.....	154
5.8.2 控制地图显示区域.....	165
5.9 地图的放大、缩小、全图显示和漫游.....	171
5.10 其他辅助功能的实现.....	175
5.10.1 鹰眼功能的实现.....	175
5.10.2 显示经纬度与比例尺.....	176
5.10.3 资源释放.....	177
第6章 选择与查询功能的实现.....	178
6.1 选择地物.....	178
6.2 查询地物信息.....	188
6.3 地名查询.....	199
6.4 查找最近地物.....	206
6.5 公交查询.....	217
6.5.1 公交站点与线路查询.....	217
6.5.2 乘车路线查询.....	246
6.6 地名索引.....	273
6.7 查询结果的定位与更详细信息.....	285
6.8 最短路径查询.....	292
第7章 系统其他辅助功能.....	332
7.1 地名的快速显示.....	332
7.2 距离与面积量算.....	337
7.3 地图输出子系统的实现.....	342
7.4 在线帮助子系统的实现.....	343
7.5 其他快捷按钮功能的实现.....	344
7.6 程序封面的实现.....	346

第8章 MapObjects的其他对象	352
8.1 动态跟踪层对象与GeoEvent对象	352
8.1.1 TrackingLayer对象的属性	352
8.1.2 TrackingLayer对象的方法	353
8.1.3 GeoEvent对象的属性	353
8.1.4 GeoEvent对象的方法	353
8.1.5 实例应用	354
8.2 投影对象	356
8.2.1 坐标系	356
8.2.2 地图投影	358
8.2.3 投影转换	363
8.3 地理编码	368
8.3.1 用于地址匹配的专用文件	368
8.3.2 绘制街道文件	369
8.3.3 地理编码对象	370
8.3.4 地址定位对象	371
8.3.5 地址标准化对象	371
8.3.6 交互式地址匹配	373
8.3.7 批地址匹配	378

第 1 章 地理信息系统软件工程

在地理信息系统 (Geographic Information System, GIS) 开发过程中应用软件工程技术, 可以提高软件开发效率和质量。本章首先介绍了软件工程的基本概念、软件系统开发过程和开发过程模型。然后讲述了软件工程技术在GIS系统开发中的应用: 需求分析、数据管理设计、用户界面设计、设计模式在GIS软件开发中的应用。这些方面涉及了GIS开发过程中的不同阶段及不同层次, 有些方法之间是互斥的, 如UML和Code方法, 但是软件工程技术最重要的特点是实用, 开发者可以根据具体情况选用不同的技术。

1.1 软件工程简介

1.1.1 基本概念

计算机软件工程是一类求解工程。它应用计算机科学、数学、工程科学及管理科学等原理, 借鉴传统工程的原则、方法创建软件, 以达到提高软件质量、降低开发成本的目的。其中, 计算机科学、数学用于构造模型与算法, 工程科学用于制定规范、设计范型、评估成本及确定权衡, 管理科学用于管理计划、资源、质量、成本等。从学科角度来看, 软件工程是一门指导计算机软件开发和维护的工程学科。

软件工程的提出是为了解决20世纪60年代出现的软件危机, 当时在大型软件开发项目中存在着成本高、开发进程不易控制、开发工作量难于估算、软件质量低、项目失败率高等诸多问题, 给软件行业带来了巨大的冲击。软件工程提出了一系列理论、原则、方法及工具, 试图解决这种软件危机。

和其他工程一样, 软件工程有自己的目标、活动和原则, 其框架可以概括为图1.1所示的内容。

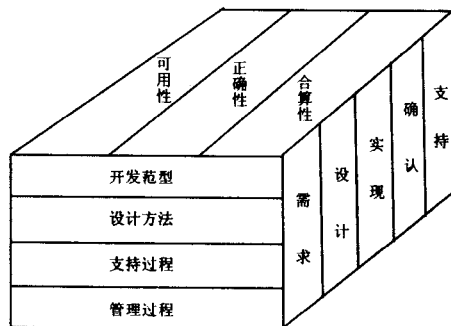


图 1.1 软件工程框架

软件工程的目标可以概括为“生产具有正确性、可用性及开销合宜的产品”，其活动包括需求、设计、实现、确认及支持等。围绕工程设计、支持及管理，软件工程有以下4条基本原则：

(1) 选取适宜的开发模型。选取适宜的开发模型可以利于认识需求易变性并加以控制，以保证软件产品满足用户的需求。

(2) 采用合适的设计方法。通常要考虑实现软件的模块化、抽象与信息隐藏、本地化、一致性及适应性等特征。

(3) 提供高质量的工程支持。在软件工程中，软件工具与环境对软件过程的支持颇为重要。

(4) 重视开发过程的管理。开发过程的管理直接影响可用资源的有效利用、最终的软件产品的满意度，软件组织的生产能力等问题。只有对开发过程实施有效管理，才能实现有效的软件工程。

GIS软件工程把软件工程的思想和方法应用于GIS软件的开发过程。如前所述，GIS软件工程活动包括需求、设计、实现、确认及支持等，它们对应于软件开发过程的不同阶段。一般来说，软件开发都要经历从分析设计到实现确认的过程。每个阶段按照相应的规范进行工作，并得到该阶段的成果，是保证整个开发过程成功的关键。

1.1.2 软件系统开发过程

前面讲过，软件工程活动包括需求、设计、实现、确认及支持，它们对应于整个软件开发过程的不同阶段。

1.1.2.1 需求分析

需求分析阶段处于软件开发过程的前期，其基本活动是准确定义未来系统的目标，确认为满足用户的需求必须做什么。需求分析又划分为两个阶段，即需求获取和需求规约，前者用自然语言清楚地描述用户的需求，而後者的目的是消除获取需求的二义性和不一致性。

在软件项目的生命周期中，一个错误发现得越晚，修复错误的代价也越高，所以，高质量的需求工程是软件项目得以正确、高效完成的前提。对于系统分析人员，建立需求面临着以下3个方面的困难：

- 问题空间的理解 系统开发人员通常是计算机专业人员，难以深入理解各种业务系统所要解决的问题空间。
- 人与人之间的通信 对于系统分析人员而言，通信主要包括同用户的通信以及同事之间的通信，由于自然语言的二义性，会给准确刻画需求造成障碍。
- 需求的不断变化 造成需求变化的原因很多，包括技术、用户方、市场等等。作为分析人员，必须采用一些策略以适应变化。

面向对象的分析方法被认为是解决上述困难的较好技术，但是完整、准确地刻画问题空间始终是分析人员所面临的挑战。



1.1.2.2 系统设计

一般来说,需求分析阶段的主要任务是确定系统“做什么”,而系统设计阶段则要解决“怎么做”的问题。系统设计的任务是将系统分析阶段提出的逻辑模型转化为相应的物理模型,其设计的内容随系统的目标、数据的性质和系统的不同而有很大的差异。一般而言,首先应根据系统的目标,确定系统必须具备的空间操作功能,称为功能设计;其次是系统的建模和产品的输出,称为应用设计。系统设计是系统整个开发工作的核心,不但要完成逻辑模型所规定的任务,而且要使所设计的系统达到优化。所谓优化,就是选择最优方案,使系统具有运行效率高、控制性能好和可变性强等特点。要提高系统的运行效率,一般要尽量避免中间文件的建立,减少文件扫描的遍数,并尽量采用优化的数据处理算法。为增强系统的控制能力,要拟定对数字和字符出错时的校验方法;在使用数据文件时,要设置口令,防止数据泄密和被非法修改,保证只能通过特定的通道存取数据。为了提高系统的可变性,最有效的方法是采用模块化的方法,即先将整个系统看成一个模块,然后按功能分解为若干个子模块。一个模块只执行一种功能,一种功能只用一个模块实现,这样设计出来的系统才能做到可变性好并具有生命力。

功能设计又称为系统的总体设计,它的主要任务是根据系统的目标来规划系统的规模,确定系统的各个组成部分,并说明它们在整个系统中的作用与相互关系,以及确定系统的硬件配置,规定系统采用的合适技术规范,以保证系统总体目标的实现。图1.2给出了通用GIS的总体设计结构图。

因此,系统的总体设计大致包括:

- 数据库设计
- 硬件配置与选购
- 软件设计等

应用设计又称详细设计。详细设计包括详细的算法、数据表示和数据结构、实施的功能和使用数据之间的关系。详细设计过程中采用了一些工具,以便对数据、算法等进行描述,包括流程图、PAD (Problem Analysis Diagram, 问题分析图)、盒图 (N-S图)、伪码等。

1.1.2.3 实现阶段

软件实现阶段将设计的结果变换成程序设计语言编写的程序。一般情况下在实现阶段,首先要确定程序设计语言,其影响因素包括:开发人员对语言的熟悉程度、语言的可移植性、编译程序的效率、编译工具的支持等等。目前,C++语言是被普遍采用的构造系统软件的编程语言,而Java则更多地应用于编写网络应用程序。

无论采用哪一种编程语言,都要求编写高质量的源程序代码,程序质量通常包含正确性、可读性、可移植性、程序效率等指标。考虑到系统的维护和演化,提高源程序的可读性是实现阶段的一个重要目标,其途径包括添加注释、规范书写格式、确定标识符命名原则、采用结构化的程序设计方法(不用或减少使用goto语句)等。



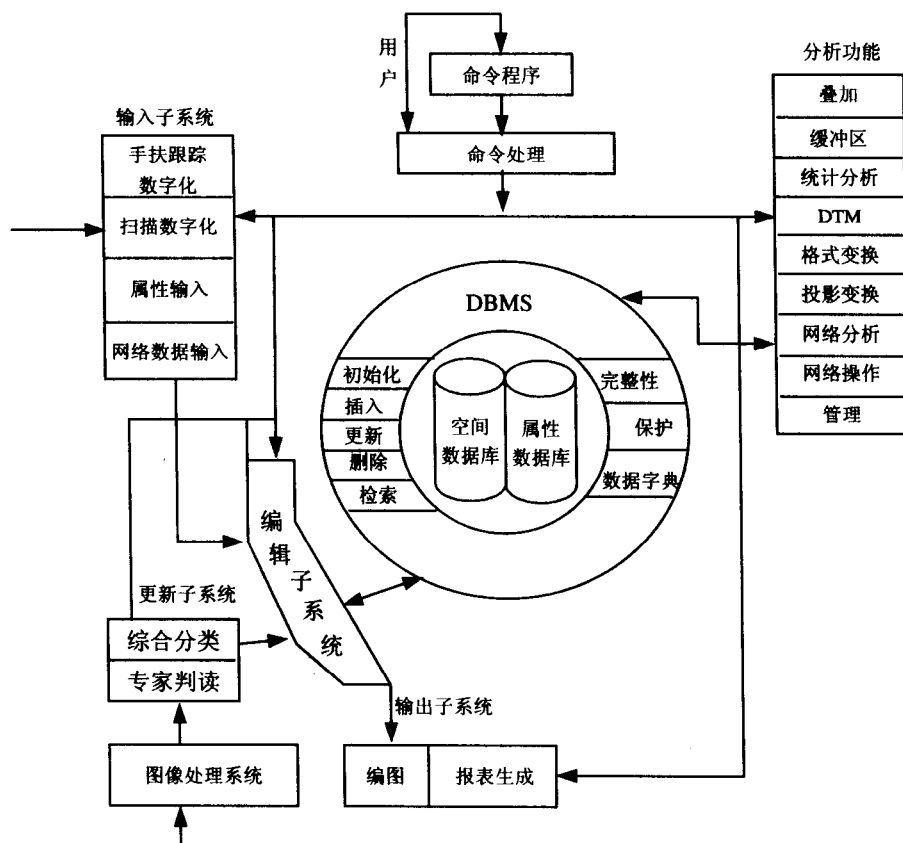


图 1.2 通用 GIS 的总体设计结构图

1.1.2.4 确认活动

确认活动贯穿于软件开发活动的始终，系统完成后的软件测试是主要的确认活动。软件测试是指按照特定规程发现软件错误的过程。软件测试技术大体上可以分为两类，即白盒测试技术和黑盒测试技术，前者依据的是程序逻辑结构，后者依据的是软件行为描述。根据测试的步骤，测试活动又可以分为单元测试、集成测试、确认测试和系统测试，其中确认测试是为了检验软件的功能和性能是否与用户需求一致，而系统测试主要是测试软件同硬件、其他支持软件、数据等结合在一起，在实际运行情况下同用户需求的匹配程度。

1.1.2.5 软件维护

当软件开发完成并交付用户使用后，就进入软件的运行/维护阶段。在此阶段对软件进行的修改称为软件维护。软件维护活动可以分为以下几类：

- 改正性维护 其目的是为了纠正运行阶段发现的软件错误、性能上的缺陷以及排除实施中的误用。
- 适应性维护 随着计算机的发展，软件的外部环境或者数据环境发生变化，为了适





应这种变化而对软件作出的修改称为适应性维护。

- 完善性维护 在使用过程中,用户往往会对软件提出新的功能和性能需求,为了满足这些需求,需要修改或再开发软件,这称为完善性维护。
- 预防性维护 预防性维护的目的是为了提高软件的可维护性、可靠性等,为进一步的软件维护打下良好的基础。预防性维护一般由开发单位主动进行。

1.1.3 开发过程模型

软件开发过程模型是软件开发的全部过程、活动和任务的结构框架。软件开发过程模型能够清晰、直观地表达软件开发过程,明确规定要完成的主要活动和任务,可以作为软件项目工作的基础。

随着软件工程的不断实践,人们相继提出了如下一系列开发模型。

1.1.3.1 瀑布模型

在瀑布模型中,各项活动依照固定顺序连接的若干阶段上工作,形如瀑布流水(见图1.3)。瀑布模型的特征是:每一阶段接受上一阶段的工作结果作为输入,其工作输出传入下一阶段;每一阶段工作都要进行评审,得到确认后,才能进入下一阶段的工作。瀑布模型较好地支持结构化软件开发,但是缺乏灵活性,无法通过软件开发活动澄清本来不够确切的需求。

1.1.3.2 演化模型

演化模型主要针对事先不能完整定义需求的软件开发项目。在这种模型中,用户可以先给出核心需求,当开发人员将核心需求实现后,用户提出反馈意见,以支持系统的最终设计和实现。

1.1.3.3 螺旋模型

螺旋模型是在瀑布模型以及演化模型的基础上,加入风险分析所建立的模型。在螺旋模型每一次演化过程中,都经历以下4个方面的活动:

- (1) 制定计划——确定软件目标,选定实施方案,弄清项目开发的限制条件。
- (2) 风险分析——分析所选方案,考虑如何识别和消除风险。
- (3) 实施工程——实施软件开发。
- (4) 客户评估——评价开发工作,提出修正建议。

每一次演化都开发出更为完善的一个新的软件版本,形成了螺旋模型的一圈。螺旋模型借助于原型获取用户需求,进行软件开发的风险分析。对于大型软件的开发,这是颇为实际的方法。



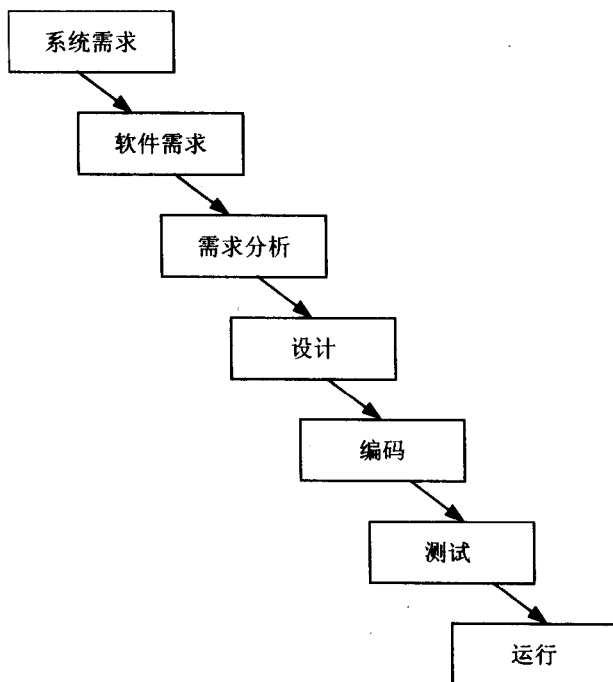


图 1.3 瀑布模型

1.1.3.4 喷泉模型

喷泉模型体现了软件开发过程中所固有的迭代和无间隙的特征（见图1.4）。喷泉模型表明了软件开发活动需要多次重复。例如，在编码之前，再次进行分析和设计，并添加有关功能，使系统得以演化。同时，该模型还表明各活动之间没有明显的间隙，例如在分析和设计之间没有明确的界限。

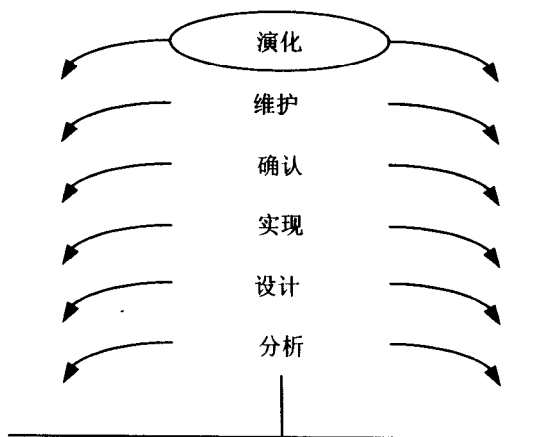


图 1.4 喷泉模型



在面向对象技术中,由于对象概念的引入,使分析、设计、实现之间的表达连贯而一致,所以,喷泉模型主要用于支持面向对象的软件开发过程。

目前,随着面向对象技术的发展和UML建模语言的成熟,人们总结并提出了统一软件开发过程(USDP, Unified Software Development Process)以指导软件开发,它是一个用例(use case)驱动的、体系结构为中心的增量迭代式开发过程模型,适合利用面向对象技术进行软件开发。

1.2 GIS需求分析

GIS需求分析是GIS产品在其生存期中重要的具有决定性的一步。只有通过GIS需求分析,才能把GIS的功能和性能的总体概念描述为具体的GIS产品需求规格说明,从而奠定GIS开发的基础。同时GIS需求分析是一个不断认识和逐步细化的过程。

GIS需求分析所要做的工作是深入描述GIS的功能和性能,确定GIS系统设计的限制和GIS同其他系统元素的接口细节,定义GIS的其他有效性需求。

1.2.1 需求获取

需求获取是软件开发活动的第一步,获得正确的需求描述是成功软件的前提。一般而言,用户需求分为两类:功能性需求(Functional Requirements)和非功能性需求(Nofunctional Requirements),前者定义系统做什么,包括输入、输出及其间的转换;后者定义系统工作时的特性,如效率、可靠性、安全性、可维护性、可移植性等要求。具体的需求获取内容包括:

- 物理环境 物理设备的位置及其分布的集中程度。
- 接口 与其他软件系统的接口以及对数据格式的要求。
- 用户或人的因素 包括系统用户熟练程度,使用系统需要接受的培训。
- 功能 系统要完成什么,性能如何。
- 文档 需要哪些文档及其针对的读者。
- 数据 数据格式、数据精度、数据量、接收和发送数据的频率。
- 资源 使用系统需要的设备,开发需要的人力资源、计算机资源、时间表。
- 安全性 对访问信息的控制程度,数据的备份等。
- 质量保证 对系统的可靠性要求、平均系统出错时间、可移植性、可维护性等。

地理信息系统的需求获取内容和上述内容基本一致,只是在数据、人员和组织等方面要进行额外的考虑。

(1) 数据

在一个GIS应用系统中,数据占有举足轻重的位置。进行需求获取时,与数据有关的因素包括:



- 数据的输出样式 包括屏幕显示、Web发布、出版、工程图等;
- 输出数据的内容和要求 输出数据要包括哪些内容,数据的精度、比例尺等;
- 数据的分布性 数据是集中管理还是分布管理;
- 现有的纸质地图 现有的纸质地图的内容,其比例尺、时效性、机密性;
- 现有的电子数据 数据形式(栅格/矢量/属性数据库)、数据格式、完整性、精度、投影方式、比例尺等因素。
- 数据录入 数据量大小、输入设备(包括数字化仪、扫描仪)、软件的支持程度、进行数据录入的人员数目、能否在预定时间内完成数据录入;
- 数据购买 数据量及价格。

(2) 人员

因为地理信息系统一般是针对专业领域,在进行需求获取时不仅要考虑一般用户,而且要听取领域专家的意见,将他们的理论、经验模型化,并应用于系统中。

(3) 组织

要考虑现有的组织机构、有关部门的职责、业务流程、GIS如何在其业务流程中体现,以及应用GIS可能引起的组织机构变化。

进行需求获取的方式是多种多样的,包括面谈、电话访谈、参观、问卷调查、获取领域相关资料等。在地理信息系统开发中,由于GIS的概念、功能等还没有被用户深入理解并接受,采用GIS专题报告可以很好地激励用户提出需求。如果时间和资金允许,开发原型系统也可以更好地挖掘用户需求。

1.2.2 需求规约

需求获取阶段得到了用自然语言描述的用户需求,但是其中存在着不一致性和二义性,这些问题要通过需求规约解决。目前有许多方法用于支持需求规约,如功能分解方法、数据流方法、信息模型方法(实体关系模型)、面向对象方法。每种方法都有相应的概念体系、符号表现和工具支持。

需求规约的结果是形式化或半形式化的,系统的需求报告必须完整地刻画问题域,能够适应需求变化。此外,它必须满足用户、分析人员、设计开发人员进行交流的需要,换言之,需求报告中的符号、描述,对各类相关人员的意义应是一致的。

正如一再强调的,面向对象的分析方法很好地解决了问题空间理解、需求易变性、交流等方面的问题。将通用的面向对象分析方法应用于GIS,固然可以描述系统需求,但是由于GIS更加关注空间对象和空间模型,在GIS中要经常处理一些特定的空间对象类以及特定的空间关系(拓扑关系、方位关系、度量关系等),所以对基本的面向对象方法进行特化扩展,采用特定的符号表示这些类及其关系,可以使需求报告的表述更加简捷,便于信息交流。