

Visual C++ 开发GIS系统

(第2版)

陈建春 编著



电子工业出版社

Publishing House of Electronics Industry
<http://www.phei.com.cn>

Visual C++ 开发 GIS 系统

(第 2 版)

陈建春 编著

电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书是GIS开发技术和Visual C++编程技术的集合体。全面介绍和分析了MFC程序的框架结构和各种MFC辅助工具的使用方法,实例剖析了MFC中与图形和数据库应用程序设计有关的类。介绍了GIS的开发方法和思路,结合具体代码开发实现了矢量图形系统、数据库管理系统、矢量图形系统和数据库管理系统之间的连接、空间信息查询、图形几何关系计算、区域重叠分析、缓冲区分析、网络分析等功能,实现了一个基本GIS。

本书是在作者2000年出版的《Visual C++开发GIS系统——开发实例剖析》一书的基础上,总结近几年实际开发的经验和技巧,进行修订并补充了很多新的内容而写成的。本书既保持了集开发思路、系统构筑、编程技术、实现代码于一体的特点,又在具体内容上进行了更新,融合了许多新的技术,更加强调了实用性。

本书适用于GIS、CAD、MIS等领域的软件开发人员,以及大专院校的师生。本书的代码适用于VC++5.0, VC++6.0及其新版本。

图书在版编目(CIP)数据

Visual C++ 开发 GIS 系统 / 陈建春编著. — 2 版. — 北京: 电子工业出版社, 2004.9
ISBN 7-121-00334-1

I .V... II .陈... III .C 语言—程序设计—应用—地理信息系统 IV .P208

中国版本图书馆 CIP 数据核字 (2004) 第 091897 号

责任编辑: 杜振民 特约编辑: 祖振升

印 刷: 北京天竺颖华印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

经 销: 各地新华书店

开 本: 787 × 1092 1/16 印张: 39.5 字数: 985 千字

印 次: 2004 年 9 月第 1 次印刷

印 数: 5000 册 定价: 75.00 元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。联系电话: (010) 68279077。质量投诉请发邮件到 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

前言

本书是2000年9月出版的《Visual C++ 开发GIS系统——开发实例剖析》的第二版。第一版出版以后，受到了欢迎，同时，读者也陆续提出了许多新的问题，希望得到解答。这些问题集中在地理信息系统（GIS）的一些功能上：例如，怎样计算区域之间的叠加关系？如何计算环形区域之间的叠加关系？如何用图形描述网络？如何计算点、线、面的缓冲区？如何实现网络分析功能？等等。在这种情况下，作者把近几年来自己在GIS系统开发过程中的认识、经验和技术的进一步做了总结和整理，决定出版此书的第二版。第二版对第一版的内容进行了全面整理、优化和修正，并增加了网络类图形元素、区域的重叠分析、缓冲区分析、网络分析等全新的内容。

近几年来，GIS技术在全球得到了迅猛的发展，它已经成为IT的重要组成部分。利用GIS技术开发的以多媒体、网络化、自动化为特色的信息处理、信息管理、指挥调度和决策支持系统，已经在各领域发挥着重大作用。GIS技术已经成为电子商务、电子政务、电子军务的重要技术支撑。正因为如此，用于开发GIS的专业开发工具也得到了飞速发展，出现了大量GIS的专业开发工具。应用专业开发工具开发GIS，具有开发起点高、可靠性好、相对简单易学等优点；但也存在着可扩展性差、适用性差、系统庞大、没有系统版权等缺点，尤其是，这些专业开发工具总体上来说还是一个难于真正修改的“黑匣子”，很难通过无缝集成实现实用系统的特殊功能。

应用VC++可以迅速开发出满足特殊要求的实用GIS系统，并在灵活性、可扩展性、系统版权、技术积累等各个方面具有无可比拟的优点。

编写本书的思路

编写本书的思路，在于以开发一个基本GIS为主线，一步步引导读者，去掌握面向对象程序设计的精髓，全面分析MFC程序框架的组成和结构，掌握GIS的开发思路和开发方法。结合VC++开发技术，实例开发组成GIS的矢量图形系统、数据库管理系统、矢量图形系统和数据库管理系统的连接、信息统计和检索、图形几何关系计算、区域重叠分析、缓冲区分析、网络分析等功能模块。使读者能够从程序设计思想、系统构筑、数据组织、MFC程序设计技术、VC++开发工具应用等各个方面，综合掌握MFC程序设计的精华。通过本书的内容，期望对读者能有所启发，从程序设计思想和编程技术等方面全面提高系统分析能力和设计能力，并在掌握本书实例开发的GIS基础上，能够自行完善各个功能，开发出各行各业实用GIS。

本书的特点

● 本书是作者多年利用 VC++ 从事开发实用 GIS、MIS、DSS、CAD 系统的实践经验和技术的提炼和总结,所以本书是理论与实践相结合的产物,是 GIS 开发方法与具体编程实现合二为一的成果,具有很强的实用性。

● 本书通过介绍一个实际的开发过程,并从一个利用 AppWizard 生成的程序框架出发,一步步展开,分析和组织 GIS 的数据结构,开发实现了包含基本功能的矢量图形系统、数据库管理系统、矢量图形系统和数据库管理系统的连接、信息统计和检索、图形几何关系计算、区域重叠分析、缓冲区分析、网络分析等功能模块。在开发过程中,结合实际代码进行了实例剖析,让读者在一个实际的开发过程中,掌握系统的设计思路和各种开发技术、开发方法。作者以实际代码结合 VC++ 设计技术向读者介绍系统的具体设计过程,避免了枯燥无味的讲解。

● 本书提供完整的源程序代码和全面翔实的代码注释。本书中的内容都是从实现思路、实现方法和具体实现等几个方面进行讲解。作者给出了以标准格式书写并经过全面调试的代码,并在代码中给出了完整的中文注释,使读者尤其是初学者能够轻松地掌握书中的内容。

本书概要

本书的第 1 章介绍了 GIS 的特点及其开发方法,介绍了使用 VC++ 开发 GIS 的思路。本书共分为四篇。

第一篇介绍了 VC++ 程序设计要点。

第 2 章介绍和剖析了 AppWizard、ClassWizard、ClassView、ResourceView、FileView 等各个 MFC 应用程序设计中必须或辅助使用的开发工具;第 3 章剖析了 MFC 应用程序框架,介绍了 MFC 程序框架的组成、基本类和构成思路。

第二篇具体开发实现了一个基本矢量图形系统。

第 4 章介绍了各种与图形系统设计有关的 MFC 类和与绘图有关的一些基本操作;第 5 章实现了基本矢量图形系统的文档和视图;第 6 章介绍了鼠标交互绘制图形元素的方法,实现了直线、连续直线、圆、圆弧、标注文本、多边形区域、网络点等的交互绘制功能;第 7 章实现了矢量图形系统的重画、放缩、移动、回溯、显示全图、滚动等操作功能;第 8 章介绍了鼠标点选图形元素的功能,实现了移动、旋转、删除图形元素等操作功能;第 9 章介绍了利用串行化存取文档的方法,实现了图形外部文件存取功能;第 10 章介绍了矢量图形系统的其他一些功能,介绍了子图块组织、图形参数管理、线型定制、子图填充、图形打印输出、与其他系统的接口等功能的实现方法或思路。

第三篇具体开发实现了一个数据数据库管理系统。

第11章介绍了开发数据库应用程序的基础知识,介绍了关系数据库、SQL语言中的基本子式,以及创建ODBC数据源的静态和动态创建方法;第12章剖析了数据库应用程序开发技术,介绍了用AppWizard创建基于ODBC数据库应用程序步骤,详细介绍记录集合类CRecordSet和视图类CDataView所能完成的功能,剖析了各种数据库管理功能的实现步骤;第13章介绍了数据库浏览功能,具体实现了一个数据库浏览视图;第14章具体实现了增加、修改、删除记录等数据库编辑功能;第15章介绍了通用数据库过滤功能,具体实现了一个通用过滤条件编辑器;第16章介绍了通用化数据库管理功能的方法,具体实现了对数据库管理功能的通用化。

第四篇开发实现GIS的一些常用功能。

第17章介绍了在矢量图形系统与数据库管理系统之间建立连接的方式和方法,实现了矢量图形系统与数据库管理系统之间的双向信息查询功能;第18章具体实现了各类图形元素的平面集合关系计算;第19章介绍了多边形区域的重叠分析,实现了多边形区域之间相交、相并、剪裁计算的具体步骤,介绍了环形区域之间相交、相并、剪裁计算的思路;第20章介绍了缓冲区分析功能,具体实现了点、线、面的缓冲区计算;第21章介绍了网络分析功能,具体实现了最佳路径分析、最佳游历路径分析、资源分配、连通分析等功能。

对读者的要求

阅读本书要求读者具有C和C++编程的一些基本知识。受篇幅所限,本书对具体的MFC类函数不做过多的说明,读者可以结合一些VC++的参考手册阅读。本书介绍的编程技术和开发实例是在VC++ 6.0企业版上实现的,对VC++ 5.0以及VC++新版本同样适用。

曾学贵教授、魏庆朝教授对于本书的编排进行了悉心指导并提供了宝贵意见,白雁、万传风、杨永平、王斌、高振宇、郭文军、周敏等同志参加了本书的编写工作,杜薇等同志参加本书的录入工作,在此一并感谢。

由于作者水平有限,书中难免有错误和疏漏之处,敬请读者给予批评指正。

陈建春

2004年3月于北京交通大学

目 录

第 1 章 VC++ 开发 GIS 的思路	(1)
1.1 GIS 简介	(1)
1.1.1 GIS 的应用领域和范围	(1)
1.1.2 GIS 与其他系统的结合	(1)
1.1.3 GIS 的专业开发工具	(2)
1.1.4 GIS 专业开发工具的优缺点	(3)
1.1.5 VC++ 开发 GIS 的优缺点及应用前景	(4)
1.1.6 选用 GIS 的开发方法	(5)
1.2 VC++ 开发 GIS 的思路	(6)
1.2.1 开发相对完整的矢量图形系统	(6)
1.2.2 开发属性数据库管理系统	(7)
1.2.3 建立矢量图形系统与数据库管理系统的连接	(7)
1.2.4 开发各种查询检索、空间分析、网络分析功能	(8)

第一篇 VC++ 程序设计技术要点

第 2 章 MFC 应用程序设计的开发工具	(11)
2.1 使用 MFC AppWizard	(12)
2.1.1 MFC AppWizard 的启动	(13)
2.1.2 利用 MFC AppWizard 产生 MFC 应用程序的步骤	(14)
2.2 使用 ClassView	(18)
2.2.1 启动 ClassView	(18)
2.2.2 编辑定位	(19)
2.2.3 增加类	(19)
2.2.4 增加类的成员	(22)
2.3 使用 ResourceView	(23)
2.3.1 ResourceView 的调用	(24)
2.3.2 ResourceView 工作分析	(24)
2.4 使用 FileView	(28)
2.4.1 打开一个程序项目	(28)
2.4.2 FileView 相关的操作功能	(28)
2.5 使用 ClassWizard	(30)
2.5.1 Windows 消息	(30)
2.5.2 消息处理函数	(32)
2.5.3 ClassWizard 的作用和启动	(34)
2.5.4 ClassWizard 管理消息处理函数	(35)
2.5.5 ClassWizard 管理重载虚函数	(39)

2.5.6	ClassWizard 管理映射变量	(41)
2.5.7	管理 Automation 和 ActiveX 事件	(45)
2.5.8	重新创建开发项目的 ClassWizard 工作信息	(45)
2.6	使用 DataView	(46)
2.7	管理多个程序项目	(47)
2.8	使用各设计工具进行编程的几点建议	(48)
第 3 章	MFC 应用程序框架	(50)
3.1	MFC 程序基本框架类和文件组成	(50)
3.1.1	基本程序框架类	(50)
3.1.2	应用程序主要源文件组成	(51)
3.2	MFC 应用程序中各个框架类的作用	(52)
3.3	MFC 应用程序的执行步骤分析	(54)
3.4	MFC 应用程序设计结构体系剖析	(56)
3.4.1	标准菜单的执行分析	(56)
3.4.2	修改标准菜单执行功能	(58)
3.4.3	MFC 应用程序多态性的实现	(59)
3.4.4	MFC 应用程序中各个框架类对象的相互调用关系	(61)

第二篇 实现基本矢量图形系统

第 4 章	与绘图操作有关的 MFC 类及操作	(65)
4.1	绘图类	(65)
4.1.1	绘图类简介	(65)
4.1.2	几个绘图类的主要差别	(66)
4.1.3	绘图类的使用方法	(67)
4.2	绘图设备类	(68)
4.2.1	CGdiObject 类	(68)
4.2.2	CPen 类	(68)
4.2.3	CBrush 类	(71)
4.2.4	CFont 类	(73)
4.2.5	CRgn 类	(77)
4.2.6	CPalette 和 CBitmap 类	(82)
4.3	坐标映像方式	(83)
4.3.1	Windows 映像方式及其设置方法	(83)
4.3.2	逻辑坐标和设备坐标的转换	(84)
4.4	窗口和视口	(85)
4.4.1	窗口和视口的概念	(85)
4.4.2	窗口和视口有关的操作函数	(85)
4.4.3	窗口逻辑坐标与视口设备坐标的转换	(86)
4.5	其他绘图类操作函数	(89)

4.5.1 选用绘图设备对象	(90)
4.5.2 主要绘图属性设置函数	(91)
第 5 章 实现矢量图形系统的文档和视图	(93)
5.1 组织矢量图形系统的图形元素类	(93)
5.1.1 图形元素基类的组织	(93)
5.1.2 点类图形元素组织	(94)
5.1.3 线类图形元素组织	(98)
5.1.4 面类图形元素组织	(100)
5.1.5 网络类图形元素的组织	(101)
5.1.6 图形参数类	(102)
5.2 组织矢量图形系统的文档	(104)
5.2.1 组织面向对象的文档管理机制	(104)
5.2.2 利用 MFC 模板创建管理图形元素对象指针的对象	(104)
5.2.3 实现矢量图形系统的文档	(105)
5.2.4 实现文档的管理功能	(106)
5.3 实现矢量图形系统的视图	(116)
5.3.1 建立坐标系	(116)
5.3.2 实现各类图形元素的绘制功能	(122)
5.3.3 视图类的 OnDraw 函数	(133)
5.3.4 实现视图	(135)
5.4 各类图形元素几何属性的计算	(136)
5.4.1 图形元素长度的计算	(136)
5.4.2 图形元素面积的计算	(138)
第 6 章 鼠标交互绘图	(140)
6.1 用鼠标绘图要解决的主要问题	(140)
6.1.1 捕捉鼠标操作消息	(140)
6.1.2 捕捉所有的鼠标输入	(142)
6.1.3 在屏幕上拖动图形	(142)
6.1.4 保存图形数据到文档	(142)
6.1.5 将图形以实际的形态重画	(143)
6.2 交互绘制各类图形元素	(143)
6.2.1 加入绘图菜单	(143)
6.2.2 获得图形元素的惟一识别号	(146)
6.2.3 绘制直线	(148)
6.2.4 绘制圆和填充圆	(151)
6.2.5 绘制圆弧	(154)
6.2.6 交互绘制标注文本	(162)
6.2.7 绘制连续直线和多边形区域	(170)
6.2.8 绘制网络点	(173)

6.2.9 绘制网络连接	(174)
第 7 章 矢量图形系统的操作功能	(177)
7.1 增加图形操作菜单	(177)
7.1.1 增加菜单资源	(177)
7.1.2 增加消息处理函数	(178)
7.2 图形重画	(179)
7.2.1 全屏重画	(179)
7.2.2 局部重画	(180)
7.3 图形放大和摆动	(181)
7.3.1 图形放大和摆动的计算方法	(181)
7.3.2 图形放大和图形摆动功能的实现	(183)
7.4 重画上屏和重画首屏	(186)
7.4.1 记录屏幕状态	(186)
7.4.2 实现重画上屏和重画首屏功能	(189)
7.5 显示全图	(190)
7.5.1 各类图形元素的边界矩形计算	(191)
7.5.2 实现显示全图功能	(195)
7.6 提高矢量图形系统重画速度的基本方法	(198)
7.6.1 提高图形重画速度的方法	(198)
7.6.2 提高图形重画速度的实现方法	(198)
7.6.3 提高图形元素的绘制速度	(200)
第 8 章 图形的选中、移动、旋转、删除	(202)
8.1 鼠标点选图形元素	(202)
8.1.1 各种图形元素的选中区域	(202)
8.1.2 各种图形元素的点选判断函数	(207)
8.1.3 选中图形元素的特殊显示	(212)
8.1.4 实现点选图形元素功能	(212)
8.1.5 放弃选中的图形元素	(218)
8.1.6 图形元素的显示和选中顺序	(219)
8.2 图形移动	(220)
8.2.1 各类图形元素的移动操作	(220)
8.2.2 实现图形移动操作功能	(222)
8.3 图形旋转	(225)
8.3.1 点与点的旋转操作	(225)
8.3.2 各类图形元素的旋转操作	(227)
8.3.3 实现旋转操作功能	(229)
8.4 图形元素的删除	(229)
8.4.1 删除操作的类型	(229)
8.4.2 从屏幕上删除图形元素的方法	(230)

8.4.3 各类图形元素的删除操作	(231)
8.4.4 实现图形元素删除操作	(235)
第 9 章 图形的存取功能	(237)
9.1 利用串行化存取文档	(237)
9.1.1 CArchive 类及 CArchive 对象的创建	(237)
9.1.2 Serialize 函数	(239)
9.1.3 CObArray 类的 Serialize 函数	(241)
9.2 实现矢量图形系统的存取功能	(242)
9.2.1 实现各个图形元素类的串行化	(242)
9.2.2 MFC 框架程序的文档串行化机制	(248)
9.2.3 实现矢量图形系统的文档存取功能	(249)
9.2.4 设置和取消文档修改标志	(251)
第 10 章 矢量图形系统的其他功能	(253)
10.1 子图块管理	(253)
10.1.1 组织管理子图块	(253)
10.1.2 创建子图块的方法	(255)
10.1.3 实现子图块的绘制功能	(255)
10.2 管理各种图形参数	(257)
10.2.1 管理颜色	(257)
10.2.2 管理图层	(260)
10.3 定制线型	(262)
10.3.1 线型的描述方式	(262)
10.3.2 实现线型所必须具备的功能	(263)
10.3.3 线型体的数据定义	(266)
10.4 区域的子图填充	(268)
10.4.1 管理填充子图块	(268)
10.4.2 绘制填充子图块	(268)
10.5 与其他图形系统的接口	(272)
10.5.1 读入其他图形系统的图形	(273)
10.5.2 输出可识别的接口文件	(279)
10.6 图形的打印输出	(281)
10.6.1 视图的打印机理和实现方法	(282)
10.6.2 打印功能的实现方法	(282)
10.6.3 实现矢量图形系统的打印功能	(285)

第三篇 实现数据库管理系统

第 11 章 开发数据库应用程序的基础知识	(293)
11.1 关系数据库	(293)

11.1.1 数据库的概念	(293)
11.1.2 关系数据库的基本术语	(295)
11.2 SQL 语言	(295)
11.2.1 选择子式	(296)
11.2.2 过滤子式	(298)
11.2.3 排序子式	(299)
11.3 ODBC 及创建 ODBC 数据源的方法	(301)
11.3.1 ODBC 的优势	(301)
11.3.2 VC++ 中使用 ODBC 的方法	(302)
11.3.3 创建 ODBC 数据源	(302)
11.3.4 动态创建 ODBC 数据源	(308)
第 12 章 数据库应用程序开发技术	(312)
12.1 创建基于 ODBC 的数据库应用程序	(312)
12.1.1 创建一个 ODBC 数据源	(312)
12.1.2 MFC AppWizard 创建一个数据库应用程序的步骤	(312)
12.2 CRecordSet 类功能分析	(314)
12.2.1 连接数据源和创建记录集合	(315)
12.2.2 与记录集合的数据交换	(318)
12.2.3 打开和关闭记录集合对象	(320)
12.2.4 过滤记录集合	(320)
12.2.5 排序记录集合	(321)
12.2.6 移动指向记录集合的记录指针	(323)
12.2.7 修改记录集合	(326)
12.2.8 CRecordSet 类的其他操作功能	(328)
12.2.9 MFC ClassWizard 管理 CRecordSet 派生类	(331)
12.3 CRecordView 视图类分析	(336)
12.3.1 CRecordView 视图类	(336)
12.3.2 CRecordView 视图对象与记录集合对象的联系	(337)
12.3.3 MFC ClassWizard 管理 CRecordView 派生类	(339)
12.4 基本数据库管理功能实现剖析	(344)
12.4.1 浏览功能	(344)
12.4.2 增加一个操作菜单	(350)
12.4.3 记录编辑功能	(352)
12.4.4 数据库过滤功能	(355)
12.4.5 数据库排序功能	(359)
第 13 章 实现数据库浏览功能	(361)
13.1 创建一个数据库浏览视图	(361)
13.1.1 视图的创建过程	(361)
13.1.2 创建另外一类视图的步骤	(363)

13.1.3 在矢量图形系统中创建数据库浏览视图	(366)
13.2 创建一个 CRecordSet 派生类对象	(373)
13.2.1 创建一个 ODBC 数据源	(374)
13.2.2 创建管理记录集合的 CRecordSet 派生类	(375)
13.2.3 创建记录集合对象	(376)
13.3 实现数据库浏览视图	(377)
13.3.1 建立视图与记录集合对象的联系	(377)
13.3.2 获得数据库浏览视图的滚动区域	(378)
13.3.3 实现数据库浏览视图	(379)
第 14 章 实现数据库编辑功能	(385)
14.1 在数据库浏览视图中建立记录指针	(385)
14.2 建立数据库编辑功能操作菜单	(389)
14.2.1 修改菜单资源	(389)
14.2.2 创建消息处理函数	(389)
14.3 建立编辑数据记录的对话框类	(391)
14.3.1 建立对话框资源	(391)
14.3.2 创建编辑数据记录的对话框类	(392)
14.3.3 设置对话框类的外部对象引用	(392)
14.3.4 建立对话框控制与记录集合类的映射变量的映射联系	(394)
14.4 增加和修改数据记录	(396)
14.4.1 获得记录的惟一识别号	(396)
14.4.2 增加记录	(397)
14.4.3 修改记录	(401)
14.5 删除记录	(402)
14.5.1 删除当前记录	(402)
14.5.2 删除所有记录	(403)
第 15 章 通用数据库过滤功能	(405)
15.1 过滤操作的实现方式	(405)
15.1.1 固定条件过滤操作	(405)
15.1.2 固定方式过滤操作	(405)
15.1.3 通用条件过滤操作	(406)
15.2 创建组织过滤条件的对话框类	(407)
15.2.1 创建对话框资源	(407)
15.2.2 创建组织过滤条件的对话框类	(408)
15.3 组织过滤条件编辑器的各种功能	(409)
15.3.1 过滤条件编辑器编辑的过滤条件式的语法格式	(409)
15.3.2 定义和初始化对话框类的变量	(410)
15.3.3 产生和分解各级条件子式	(413)
15.3.4 初始化对话框的状态	(418)

15.3.5 增加条件子式	(419)
15.3.6 删除和修改条件子式	(425)
15.3.7 条件子式的分解功能	(429)
15.3.8 实现过滤操作	(432)
15.4 在数据库浏览视图中实现过滤功能	(434)
15.4.1 增加一个操作菜单	(434)
15.4.2 实现数据库过滤操作功能	(435)
15.5 增强过滤条件编辑器的功能	(436)
15.5.1 增强过滤条件分解能力	(436)
15.5.2 增强交互式编辑界面	(437)
15.5.3 存储过滤条件	(438)
第 16 章 数据库管理功能通用化	(439)
16.1 各种数据库管理功能的可通用化分析	(439)
16.1.1 可通用化的数据库管理功能	(439)
16.1.2 不易通用化的功能	(440)
16.2 实现数据库管理功能通用化	(440)
16.2.1 通用化功能模块的方法	(441)
16.2.2 数据库操作功能模块通用化的步骤	(443)
16.2.3 选择操作不同的记录集合对象	(448)
16.2.4 增加应用程序管理的记录集合对象	(451)
16.2.5 增加通用数据库操作功能	(455)
16.3 数据库管理功能完全通用化	(457)

第四篇 实现 GIS 的常用功能

第 17 章 图形元素与属性数据的连接	(461)
17.1 连接方式及连接单元	(461)
17.1.1 连接方式	(461)
17.1.2 连接单元	(462)
17.1.3 连接的特性	(462)
17.2 属性数据的连接方法	(463)
17.2.1 隐含固定连接法	(463)
17.2.2 动态组织连接信息法	(464)
17.3 实现双向信息查询检索	(470)
17.3.1 通过属性数据检索矢量图形元素	(470)
17.3.2 通过图形元素检索属性数据	(472)
第 18 章 图形元素的平面几何关系计算	(475)
18.1 直线与多边形区域的关系	(475)
18.1.1 直线与直线的关系	(475)

18.1.2 直线与多边形区域的相交关系	(479)
18.1.3 连续直线与多边形区域的相交关系	(482)
18.2 圆弧与多边形区域的关系	(485)
18.2.1 圆与直线的关系	(486)
18.2.2 圆弧与直线的关系	(489)
18.2.3 圆弧与多边形区域的关系	(492)
18.3 各类图形元素与圆形区域的关系	(495)
18.3.1 圆与圆的关系	(495)
18.3.2 圆弧与圆形区域的关系	(497)
18.3.3 直线与圆形区域的关系	(499)
18.3.4 连续直线与圆形区域的关系	(502)
第 19 章 多边形区域叠加分析	(506)
19.1 两个多边形区域的交集	(506)
19.1.1 两个多边形区域交集的计算思路	(506)
19.1.2 计算两个多边形区域的交集	(507)
19.2 两个多边形区域的并集	(515)
19.2.1 两个多边形区域并集的计算思路	(515)
19.2.2 计算两个多边形区域的并集	(517)
19.3 一个多边形区域剪裁另外一个多边形区域	(524)
19.3.1 计算一个多边形区域剪裁另一个多边形区域的思路	(524)
19.3.2 计算一个多边形区域剪裁另一个多边形区域的补集	(525)
19.4 区域相交、相并、剪裁计算时的重合边界处理	(531)
19.4.1 相交、相并、剪裁计算时的重合边界处理情况分析	(531)
19.4.2 获得重合边界	(532)
19.4.3 相交、相并、剪裁计算时的重合边界处理	(536)
19.5 环形区域的相交、相并、剪裁计算的思路	(543)
19.5.1 实心区域与环形区域的相交、相并、剪裁计算	(544)
19.5.2 两个环形区域相交、相并、剪裁计算的思路和方法	(547)
第 20 章 缓冲区分析	(549)
20.1 点的缓冲区	(549)
20.2 直线的缓冲区	(550)
20.3 连续直线的缓冲区	(552)
20.3.1 用区域并集方法获取连续直线缓冲区	(552)
20.3.2 计算单侧连续直线缓冲区时的剪裁处理	(554)
20.3.3 加快连续直线缓冲区的计算速度	(561)
20.4 多边形区域的缓冲区	(563)
第 21 章 网络分析	(565)
21.1 网络基本信息计算	(565)

21.1.1 枢纽点的确定方法	(565)
21.1.2 节点连接信息的描述方法和处理	(566)
21.1.3 管理整个网络组织和网络连接信息	(568)
21.2 最佳路径分析	(577)
21.2.1 最佳路径分析方法	(578)
21.2.2 正向最佳路径搜索	(579)
21.2.3 逆向最佳路径搜索	(585)
21.2.4 确定搜索方向的方法	(589)
21.3 最佳游历路径分析	(590)
21.3.1 求解最佳游历路径的方法	(590)
21.3.2 实现最佳节点游历功能	(591)
21.4 资源分配	(600)
21.4.1 获取所有节点到中心节点距离法	(600)
21.4.2 中心节点向外延伸法	(604)
21.5 连通分析	(612)
附录 代码光盘与技术支持	(614)
参考书目	(615)

第1章 VC++ 开发GIS的思路

GIS是地理信息系统（Geographical Information System，简称为GIS）的简称，是一种将空间位置信息与属性数据结合在一起的系统。

近几年来，GIS技术在全球得到了迅速的发展，GIS技术已经成为IT的重要组成部分，在国外，用于开发GIS的工具得到了飞速发展。

1.1 GIS简介

1.1.1 GIS的应用领域和范围

从最初概念来看，GIS指的是将地球科学与属性数据相结合完成的一种大型的集成系统。但是，在实际应用中，以GIS技术为基础建立起来的系统已经应用到各种领域。各种GIS的复杂程度相差很大，大到针对某个地区（以至全球）的数字地面模型资料所建立起来的庞大系统，如美国在海湾战争、南联盟战争中采用的辅助指挥作战GIS；小到只是结合了简单平面位置坐标的小型管理系统，如一个住宅小区的平面管理系统，一个商场的触摸式导购系统等。从GIS中空间坐标的表示方法来看，有利用数字地面模型资料建立起来的三维图形为基础的GIS，也有使用二维平面示意图形的GIS。

GIS将普通管理信息系统（MIS）与矢量图形系统结合在了一起，克服了普通MIS的局限性，大大提高了系统的直观性和可操作性，同时提供了普通MIS所不能解决的功能，如空间信息统计、空间信息计算、网络分析等。

1.1.2 GIS与其他系统的结合

单纯的GIS实用价值受到限制。在实际应用中，GIS往往与其他系统结合在一起，GIS通常与遥感系统（RS）、全球定位系统（GPS）结合在一起，将三者一体化，也就是通常所说的“3S”系统。现在的GIS的应用范畴非常广泛，它已经与CAD、多媒体、通信、办公自动化（OA）、虚拟现实等技术相结合，形成了综合的信息技术。

GIS作为一个基础，与其他系统相结合，来设计实现各种实用系统。例如，作者开发的“铁路新线建设管理指挥系统”，利用GIS技术将施工地区的施工指挥平面图矢量化，用来直观地管理与工程建设有关的各种信息，如各种材料点、交通网、城市、地质情况、线路上的各种重点工程等，而系统的主要功能则是实现铁路建设中的管理功能，如施工组织计划的制定、施工组织进度图的辅助制定和绘制、重点工程工期的管理、施工质量、施工进度、工程投资的管理等等，整个系统应该属于MIS的范畴，但结合了GIS技术后，使MIS的操作功能具有了很大的改进，在实用性、直观性等方面达到了新的高度。再比如，作者