

Earth Sciences 地球科学类专业实验与实践系列教材

GIS程序设计教程

—— 基于ArcGIS Engine 的C#开发实例

◎ 张 丰 杜震洪 刘仁义 编著



ZHEJIANG UNIVERSITY PRESS
浙江大学出版社

GIS 程序设计教程

——基于 ArcGIS Engine 的 C# 开发实例

张 丰 杜震洪 刘仁义 编著



ZHEJIANG UNIVERSITY PRESS
浙江大学出版社

内容提要

这是一本非常适合于 GIS 组件式开发人员入门的教材,主要介绍了组件式 GIS 开发技术,重点是利用 ArcGIS Engine 开发组件库,在 .Net 环境下利用 C# 语言进行 GIS 程序开发。全书由浅入深,从组件式 GIS 基本概念入手,介绍了 ArcGIS Engine 10.0 的特性;从地图显示浏览入手,介绍了 GIS 数据的组织与访问、制图渲染与输出、空间数据编辑、GIS 分析及栅格图像处理,涵盖了 GIS 数据采集、编辑、处理、分析、输出等的基本功能;进一步地,分析了 ArcGIS 所提供的功能扩展模块,为 GeoProcessing 及 3D 分析等深入开发提供了案例。

本书适合地理信息系统、遥感等地理专业、测绘类、地质类、农林类、水利类等相关专业的本科生、研究生教学使用,也适合测绘、国土资源、城市规划、交通、环境保护等部门的研究和开发人员使用。

图书在版编目 (CIP)数据

GIS 程序设计教程:基于 ArcGIS Engine 的 C# 开发实例 / 张丰,杜震洪,刘仁义编著. —杭州:浙江大学出版社,2012.5

ISBN 978-7-308-09884-7

I. ①G… II. ①张… ②杜… ③刘… III. ①地理信息系统—程序设计—应用软件—教材②C 语言—程序设计 IV. ①P208T②P312

中国版本图书馆 CIP 数据核字 (2012) 第 069153 号

GIS 程序设计教程——基于 ArcGIS Engine 的 C# 开发实例

张 丰 杜震洪 刘仁义 编著

责任编辑 樊晓燕

封面设计 刘依群

出版发行 浙江大学出版社

(杭州市天目山路 148 号 邮政编码 310007)

(网址: <http://www.zjupress.com>)

排 版 杭州中大图文设计有限公司

印 刷 浙江良渚印刷厂

开 本 787mm×1092mm 1/16

印 张 14.25

字 数 347 千

版 印 次 2012 年 5 月第 1 版 2012 年 5 月第 1 次印刷

书 号 ISBN 978-7-308-09884-7

定 价 29.00 元

版权所有 翻印必究 印装差错 负责调换

浙江大学出版社发行部邮购电话 (0571)88925591

前 言

地理信息系统(Geographical Information Systems, GIS)作为当代空间信息科学技术的核心,已渗透到自然与社会科学的各个领域,并关系着国家诸多命脉产业。近年来,社会各领域对 GIS 专业人才(尤其是 GIS 应用与开发人员)的需求与日俱增。

GIS 应用开发型人才培养应以“基础适度、专业拓宽、加强计算机技能培养、提高实践操作能力”为原则,紧跟信息技术的发展步伐,提高 GIS 学习的实践强度和深度,培养其坚实的 GIS 应用与开发能力;以国内外最新研究成果和技术潮流为导向,鼓励接触并掌握前沿知识,用新思维、新方法、新视野解决问题,全面提升个人的专业水准与素质。

本教材最大优势在于编者的编写理念符合 GIS 技术的教学规律,在教材编写中体现了概念和实践并重;教材的另一优势则在于编者把自己多年教学与科研过程中收集和积累的 GIS 数据及其应用案例提供给读者共享,大大提高了其可读性。书中提供的 GIS 程序开发实例全面分析、讲解了 ArcGIS Engine 的相关组件接口,提供了详尽的可直接编译使用的程序代码,使 GIS 开发的学习过程立竿见影、收效显著。通过实际操作来体会,有助于对组件式 GIS 开发的原理、方法和应用的理解,初步掌握主流 GIS 组件开发平台的组件模型与接口使用。本教材是一套较完整的教学资源,可供不同程度的读者作为对照学习的辅助参考资料。

第 1 章介绍了组件式 GIS 程序开发基础知识,介绍、对比了当前国内外应用较广的几种 GIS 组件开发平台;第 2 章介绍了 ArcGIS Engine 的基本内容,提供了详细的开发环境部署过程;第 3 章结合地图显示介绍了 ArcGIS Engine 提供的系列地图控件及相关组件,详细阐述了空间书签功能模块的编写过程;第 4 章针对地图数据组织方式介绍了 ArcGIS 几种常用的空间数据格式及其访问方法,编写了地理属性数据显示功能模块;第 5 章介绍了地图制图相关组件,重点分析符号渲染模块,编写了制图输出功能模块;第 6 章就地理数据文件创建、空间和属性信息编辑进行了相应组件介绍与功能模块编写;第 7 章介绍 GIS 分析相关内容,给出了缓冲区分析功能实现;第 8 章阐述了栅格影像处理的相关功能组件、接口及其使用方法,提供了各类栅格空间分析代码实例;第 9 章是在基础 GIS 功能实现的基础上,讨论 GeoProcessing 与 3D 等功能扩展模块及其使用方法,最后将 ArcGIS 控件嵌入 Office 进行扩展应用开发。全书共 9 章,18 个实例,读者根据本教材提供的代码实例可以建立一个具有常规 GIS 显示、处理、分析、栅格操作的 GIS 应用软件。

本书的编著出版要感谢如下同仁给予的大力支持和协助。感谢冯天博士在“开发环境

部署”、“创建与调用 AOI 书签”、“创建 Shape 文件与添加要素”、“创建地理数据列表”、“简单渲染图层”、“打印页面布局”等实例编写中给予的帮助;感谢沈忠悦老师、浙江大学出版社樊晓燕以及我的研究生们给予的大力支持与帮助。感谢 ESRI 中国(北京)有限公司提供的软件和文档资料的支持。

受作者水平所限,错漏在所难免,欢迎读者不吝提出宝贵的批评和建议,以便再版和重印时改进。

张 丰

2012 年 4 月

目 录

第 1 章 导 论	1
1.1 GIS 程序设计	1
1.1.1 GIS 技术与发展	1
1.1.2 GIS 开发模式	3
1.1.3 GIS 开发模式比较与分析	4
1.2 组件化程序设计	4
1.2.1 COM 概述	5
1.2.2 COM 的特性	5
1.2.3 COM 的结构	6
1.3 组件式 GIS	8
1.3.1 组件式 GIS 体系结构	9
1.3.2 组件式 GIS 的特点	10
1.3.3 组件式 GIS 的不足	11
1.4 主流 GIS 组件平台	12
1.4.1 ArcGIS Engine	12
1.4.2 GeoMedia	13
1.4.3 MapX	14
1.4.4 TITAN GIS	15
1.4.5 SuperMap Objects	16
1.4.6 几种主要组件式 GIS 平台功能比较	17
第 2 章 ArcGIS Engine 开发初步	19
2.1 ArcGIS Engine 概述	19
2.1.1 ArcGIS Engine	19
2.1.2 ArcGIS Engine 的功能	20
2.1.3 ArcGIS Engine 包含的内容	21
2.2 使用 ArcGIS Engine 开发应用程序	22
2.3 软件安装	25
2.3.1 安装 Visual Studio 2010	25
2.3.2 .NET 与 C#	28
2.3.3 安装 ArcGIS Engine	33

2.4 ArcGIS Engine 类库介绍	36
2.4.1 对象模型图	36
2.4.2 常用类库概览	36
2.5 部署一个 ArcGIS Engine 应用程序	39
第 3 章 地图显示与浏览	42
3.1 地图控件	42
3.1.1 控件特性	43
3.1.2 地图控件	44
3.1.3 目录树控件	47
3.1.4 工具条控件	48
3.1.5 页面控件	51
3.2 地图及其相关组件	54
3.2.1 地图组件	54
3.2.2 地图常用接口	56
3.3 空间书签组件	56
3.4 创建与调用 AOI 书签	58
3.5 开发提示——如何判断添加类库引用	65
第 4 章 地图数据组织与访问	69
4.1 数据类型	69
4.1.1 Coverage	69
4.1.2 Shapefile	70
4.1.3 Geodatabase	71
4.1.4 ArcXML	71
4.2 Geodatabase 数据模型	72
4.2.1 Geodatabase 模型结构	72
4.2.2 Geodatabase 数据模型的优点	73
4.2.3 Geodatabase 数据模型的缺点	74
4.3 Geodatabase 类型	75
4.3.1 文件地理数据库	76
4.3.2 个人地理数据库	76
4.3.3 ArcSDE 地理数据库	77
4.3.4 三种类型的地理数据库比较	77
4.4 数据访问	78
4.4.1 工作空间工厂及其相关组件	78
4.4.2 打开一个 Shapefile	79
4.4.3 打开一个 Access Geodatabase 要素类	81
4.4.4 图层组件 ILayer	81
4.4.5 地理数据集组件	84

4.5 地理数据列表显示	86
4.6 数据格式转换	91
4.6.1 地理数据转换组件	91
4.6.2 数据转换示例	92
第5章 地图渲染与制图输出	97
5.1 地图制作	97
5.1.1 地理对象的符号化表达方式	97
5.1.2 地图制图的要求	98
5.1.3 地图数据准备	99
5.1.4 地图整饰与输出	99
5.2 地图显示及其相关组件	99
5.3 符号渲染	100
5.3.1 ArcMap 中的地图渲染	100
5.3.2 特征渲染器 Render	105
5.3.3 图层基本渲染	110
5.4 制图输出	117
5.4.1 制图输出相关组件	117
5.4.2 打印页面布局	122
5.4.3 制图文件输出	127
第6章 空间数据处理	129
6.1 数据创建	129
6.1.1 创建工作空间	129
6.1.2 要素工作空间及其相关组件	130
6.1.3 字段相关组件	130
6.1.4 地理要素类的创建	132
6.1.5 创建一个 Shapefile 文件	133
6.2 地理要素编辑	137
6.2.1 地理要素相关组件	137
6.2.2 创建新要素	141
6.2.3 地理要素交互编辑	144
6.3 地图元素编辑	150
6.3.1 地图元素相关组件	150
6.3.2 地图的整饰元素	154
6.3.3 添加地图元素编辑工具	157
第7章 GIS 分析	159
7.1 空间关系查询	159
7.1.1 数据查询相关组件	159

7.1.2 空间关系	163
7.1.3 空间关系示例	164
7.2 空间拓扑分析	167
7.2.1 拓扑操作	167
7.2.2 缓冲区分析	169
7.3 数据统计	171
7.3.1 数据统计	172
7.3.2 要素统计实例	173
第 8 章 栅格数据处理	176
8.1 栅格数据模型	176
8.2 栅格数据访问	177
8.2.1 打开栅格工作空间	177
8.2.2 获得栅格数据集	178
8.2.3 获得栅格目录	179
8.2.4 创建栅格数据集	180
8.3 栅格数据处理	183
8.3.1 栅格数据格式转换	183
8.3.2 栅格影像镶嵌	184
8.3.3 栅格转换相关组件	189
8.4 栅格空间分析	191
8.4.1 栅格计算	191
8.4.2 栅格插值	193
8.4.3 地形分析	193
8.4.4 栅格统计	193
第 9 章 ArcEngine 深入开发	197
9.1 ArcGIS 扩展模块	197
9.2 利用 GeoProcessing 实现流程式空间处理	201
9.2.1 GeoProcessing	201
9.2.2 利用 ModelBuilder 建立空间处理工具	201
9.2.3 地理处理相关类库与接口	203
9.2.4 在程序中添加 GeoProcessing 处理模型	207
9.3 3D 分析开发	208
9.3.1 ArcScene 相关组件与接口	209
9.3.2 3D 分析与显示实例	211
9.4 在 Office 中嵌入 ArcGIS Engine 开发	214
参考文献	219

第1章 导 论

地理信息系统(Geographic Information System, GIS)的理论和技術方法已经渗透到自然科学和社会科学的各个领域,社会对地理信息系统专业人才的需求很大,尤其是对 GIS 应用开发人员的需求越来越大。

GIS 应用开发具体表现为一个计算机软件信息系统,既有一般信息系统的普遍特征,又不乏自身的特殊性,主要表现在空间数据的大容量和复杂性方面。本书为组件式 GIS 程序开发,特别在 .NET 环境下利用 C# 语言实现基于 ArcGIS Engine 的 GIS 程序开发教程,可为 GIS 程序开发初学者、兴趣爱好者提供指导。

面向对象的程序设计语言和面向对象的程序设计方法,已经渗入到计算机软件科学的各个领域。随着软件科学的不断发展,新的应用系统越来越复杂。尤其是近几年 Intranet/Internet 的飞速发展,使软件应用置身于更加复杂的环境中,从而对应用软件提出了更高的要求,这就使得软件设计更加困难。在这样的情况下,面向对象的思想已经难以适应这种分布式软件模型,于是组件化程序设计思想得到了迅速的发展。以往的 GIS 软件像其他软件一样,一直是由软件开发商提供全部系统或者具有二次开发功能的软件,不能脱离平台。现在除了提供功能强大的集成式 GIS 平台外,还提供丰富的组件工具,用户可以根据需要,采用高级开发语言自行开发。

1.1 GIS 程序设计

1.1.1 GIS 技术与发展

自上世纪 60 年代加拿大第一个地理信息系统问世以来,经过 40 多年的发展,GIS 已经广泛应用于测绘、规划、土地管理、地质、矿山、环境保护、资源调查、城市评估、区域发展、疾病预防、电力、通讯、商业、公安、交通、房地产等诸多领域,并在其中发挥越来越大的作用。随着时间的推移和理论技术的发展,特别是计算机技术的发展,地理信息研究与应用已经由系统开发阶段发展到信息共享阶段,再到信息服务阶段,如图 1-1 所示。

20 世纪 60 年代属于 GIS 的起步阶段,由于受当时计算机性能的限制,这个时期 GIS 的主要任务是地图数据输入、地图数据管理、空间数据统计、自动化制图等,地学分析功能极为简单。发展到 70 年代,随着计算机硬件和软件技术的飞速发展,GIS 朝着应用方向迅速发展。这阶段提出了混合数据模型,增强了空间综合分析能力。一些发达国家先后在资源、环境、国土等领域建立了许多专业性的信息系统;一些商业公司开始活跃起来;许多大学和研

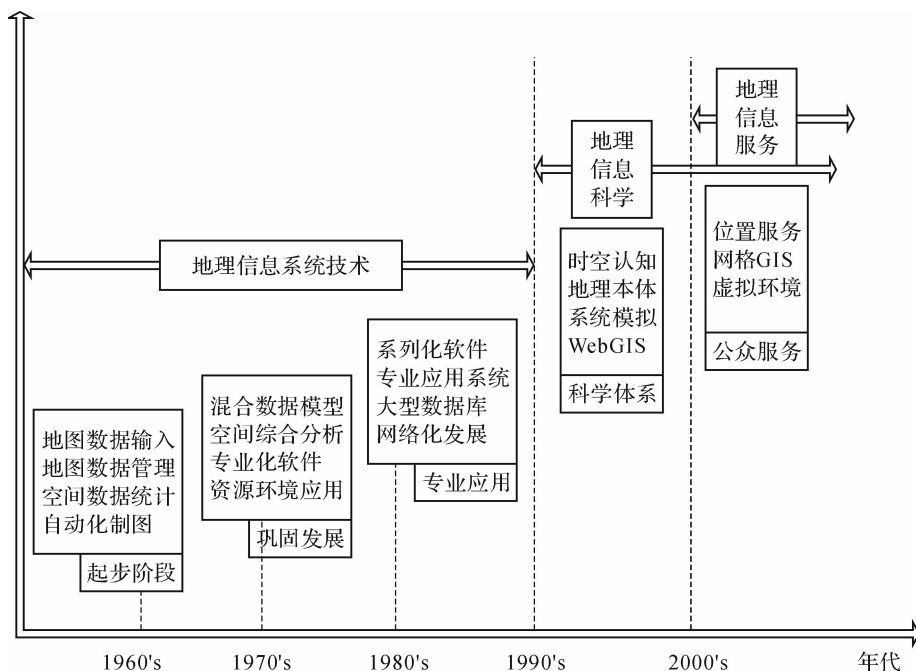


图 1-1 GIS 发展历程

研究机构也开始重视 GIS。那一时期 GIS 处于一个巩固发展阶段。80 年代,随着计算机性能的不断提高,GIS 的研制和开发也取得了更大的成绩,涌现出了系列化软件平台、专业应用系统,大型数据库开始在 GIS 中得到应用。经过 30 余年的发展,到 20 世纪 90 年代,地理信息系统及遥感等领域已经积累了从理论、技术到应用研究的非常丰富的内容,GIS 事实上已经形成一个新的学科领域(刘南,2002)。

地理信息科学是 1992 年由 GoodChild 提出的。与地理信息系统相比,它更加侧重于将地理信息视作一门科学,而非仅仅是一个技术实现。地理信息科学已经形成一个重要的新兴交叉学科,其主要研究在应用计算机技术对地理信息进行处理、存储、提取以及管理和分析过程中提出的一系列基本问题,包括分布式计算、地理信息的认知、地理信息的互操作、地理数据的不确定性以及 GIS 和社会等。

随着 Internet 应用在全球的迅猛发展,主要采用 HTML、HTTP 与 WWW 等技术实现的 Internet 应用所带来的优越性已经开始在 GIS 的研究领域中逐步被认识到。WebGIS 是 Internet 技术应用于 GIS 开发的产物。通过因特网上的任意一个节点,用户可以浏览 WebGIS 站点中的空间数据、制作专题图、进行各种空间检索和空间分析。随着分布式网络技术的发展以及“数字地球”概念的提出,传统的 GIS 正向着信息共享和开放的网络 GIS 发展,Internet 技术正在改变整个世界。

1996 年以前出现的由 WebGIS 提供的地理信息服务仅仅是地理信息分发,是单纯的地图服务器。1996 年以后,随着 Internet 技术的飞速发展,地理信息服务也从地理信息分步入地理信息共享。到了 21 世纪,Web Services 概念、面向服务的软件体系结构思想的兴起为实现真正意义上的地理信息服务开辟了新的道路。基于 Web 服务技术的 GIS 门户已经开始具备进行多个异构系统交互、完成更为高级 GIS 处理与分布式地理信息服务等功

能。地理信息服务改变了 GIS 的设计和应用模式,有效地解决了空间信息共享与互操作的难题,是 GIS 发展的必然趋势。

GIS 的发展得益于计算机技术的进步,地球信息科学的发展已经经历了从地理信息系统—地理信息科学—地理信息服务的不同阶段的“质”的飞跃。

1.1.2 GIS 开发模式

地理信息系统根据其内容可分为两大基本类型:一是应用型地理信息系统,包括专题地理信息系统和区域综合地理信息系统;二是工具型地理信息系统,即 GIS 工具软件包,如 ArcGIS、SuperMap 等,具有空间数据输入、存储、处理、分析和输出等功能。随着地理信息系统应用领域的扩展,应用型 GIS 的开发工作日益重要。如何针对不同的应用目标,高效地开发出既合乎需要又具有方便、美观、丰富的界面形式的地理信息系统,是 GIS 开发者非常关心的问题。

GIS 软件走过了几十年的发展历程,发展到了组件式 GIS 阶段。传统 GIS 虽然在功能上已经比较成熟,但是由于这些系统多是基于十多年前的软件技术开发的,受当时技术条件的限制,存在许多固有的缺陷,这些系统基本上属于独立封闭的系统。同时,这些 GIS 软件由于用户需求不断增加,功能一再扩充,变得日益庞大。其结果是用户难以掌握,费用昂贵,阻碍了 GIS 的普及和应用。组件式 GIS 的出现为集成式 GIS 面临的多种问题提供了全新的解决思路。

GIS 常用开发模式主要有三种:独立开发、单纯二次开发、集成二次开发。

1. 独立开发模式

独立开发模式不依赖于任何 GIS 工具软件,从空间数据的采集、编辑到数据的处理分析及结果输出,所有的算法都由开发者独立设计,然后选用某种程序设计语言在一定的操作系统平台上编程实现。这种方式的好处在于无须依赖任何商业 GIS 工具软件,减少了开发成本。但另一方面,对于大多数开发者来说,由于能力、时间、财力方面的限制,其开发出来的产品很难在功能上与商业化 GIS 工具软件相比。

2. 单纯二次开发模式

单纯二次开发模式指完全借助于 GIS 工具软件提供的开发语言进行应用系统开发。GIS 工具软件大多提供了可供用户进行二次开发的宏语言,如 ESRI 的 ArcView 提供了 Avenue 语言、MapInfo 公司研制的 MapInfo Professional 提供了 MapBasic 语言等。用户可以利用这些宏语言,以原 GIS 工具软件为开发平台,开发出自己的针对不同应用对象的应用程序。这种开发方式继承了平台软件的所有 GIS 功能,容易掌握,开发起来省时省力。但受二次开发的宏语言限制,所开发的应用系统功能扩展能力极弱,难以解决复杂问题,开发的应用程序不尽如人意。

3. 集成二次开发模式

集成二次开发模式是指利用专业的 GIS 工具软件,如 ArcInfo、MapInfo、SuperMap 等,实现 GIS 的基本功能,以通用软件开发工具尤其是可视化开发工具,如 C#、.NET、Visual

C++、Delphi、Visual Basic、Java 等为开发平台,进行二者的集成开发。

集成二次开发模式目前主要有两种方式:

(1) OLE/DDE

此方法采用 OLE Automation 技术或利用 DDE 技术,用软件开发工具开发前台可执行应用程序,以 OLE 自动化方式或 DDE 方式启动 GIS 工具软件在后台执行,利用回调技术动态获取其返回信息,实现应用程序中的地理信息处理功能。

(2) GIS 控件

此方法利用 GIS 工具软件生产厂家提供的建立在 OCX 技术基础上的 GIS 功能控件,如 ESRI 的 ArcGIS Engine、MapInfo 公司的 MapX、SuperMap 的 SuperMap Objects 等,在高级编程语言环境中,直接将 GIS 控件嵌入其中,调用组件实现地理信息系统的各种功能。

1.1.3 GIS 开发模式比较与分析

由于独立开发难度太大,开发出来的产品在功能上很难与商业化工具软件相比,这种低水平重复开发的结果往往是人力、物力、财力的极大浪费。单纯二次开发受 GIS 工具提供的编程语言的限制,功能效果差强人意。相比之下,结合 GIS 工具软件与当今可视化开发语言的集成二次开发方式就成为 GIS 应用开发的主流方式。它的优点是既可以充分利用 GIS 工具软件对空间数据的管理、分析功能,又可以利用其他可视化开发语言具有的高效、方便等编程优点,集二者之所长,不仅能大大提高应用系统的开发效率,而且使用可视化软件开发工具开发出来的应用程序具有更好的外观效果,更强大的数据库功能,可靠性好、易于移植、便于维护。

相比较三种开发模式,集成二次开发模式成为 GIS 开发的主流方向。与利用 OLE Automation 技术作为服务器的 MapInfo 相比,利用控件开发速度快,占用资源少,而且易实现许多底层的编程和开发功能。目前许多软件公司开发了很多 ActiveX 控件。合理选择和运用现成的控件,可减少开发者的编程工作量,使开发者避开某些应用的具体编程,直接调用控件,实现具体应用。这不仅可以缩短程序开发周期,使编程过程更简洁、用户界面更友好,而且可以使程序更加灵活、简便。

1.2 组件化程序设计

组件是指已经编译、链接好并可以使用的二进制代码模块,每一个模块可以运行在同一台机器上,也可以运行在局域网、广域网及 Internet 上的不同机器上。多个组件粘合起来可以形成单独而复杂的应用程序。组件间通过组件提供的接口进行通信。

组件化程序设计思想的出现是用来解决传统软件开发中周期长、维护困难、难以扩展等问题。它不同于传统的结构化程序设计方法,也不同于面向对象程序设计方法,更侧重于应用系统的全局。

组件技术中的关键技术之一是接口通信问题。在同一软件中的组件必须使用同样的接口标准才能保证组件之间可以进行通信。为此,国际上 OMG 对象管理组织和 Microsoft 公司分别提出了 CORBA(Common Object Request Broker Architecture,公共对象请求中介

体系结构)和 COM(Component Object Model,组件对象模型)标准。目前,CORBA 主要应用于 UNIX 操作系统平台上;而 COM 则主要应用于 Microsoft Windows 操作系统平台上。

1.2.1 COM 概述

COM 是 Component Object Model(组件对象模型)的缩写。从起源上讲,COM 标准是在 OLE 技术发展过程中产生的。OLE 技术以 COM 规范为基础,充分发挥 COM 标准的优势,使 Windows 操作系统上的应用程序具有极强的可交互性。这几年,网络技术飞速发展,OLE 技术在进行网络互联时显示出了很大的局限性,而 COM 则表现出了极强的适应能力。因此,伴随着网络的发展,COM 也得到了展示的机会。

COM 不仅定义了组件程序之间进行交互的标准,而且也提供了组件程序运行所需要的环境。COM 提供了一个 COM 库(COM library),用户使用其应用程序编程接口(API)能实现组件的查询,以及组件的注册/反注册等一系列服务。COM 库主要应用于 Microsoft Windows 操作系统平台上。COM 通常以 win32 动态链接库(DLL)或可执行文件(EXE)的形式发布。

COM 不是一种面向对象的语言,而是一种二进制标准。它定义了组件对象之间基于这些技术标准进行交互的方法。简单说来,COM 是一种以组件为发布单元的对象模型,这种模型使各软件组件可以用一种统一的方式进行交互。COM 是一种网络标准,可用于软件组件间跨越多个进程、机器、硬件和操作系统进行互操作。COM 组件对象之间交互的规范不依赖于任何特定的语言,COM 也可以是不同语言协作开发的一种标准。换言之,它允许任意两个组件互相通信,而不管它们是在什么计算机上运行,不管各计算机运行的是什么操作系统,也不管该组件是用什么语言编写的。

组件技术有以下优点:

(1)简化应用开发。组件开发商已经编制好了大量的组件模型,因此减少了用户开发的工作量,使开发周期大大缩短,开发成本大大减低。

(2)增加应用软件的灵活性。应用软件是在组件上编制的,对于使用者的不同要求,往往只要通过更换、修改一个或几个组件就可以实现。

(3)维护方便。

1.2.2 COM 的特性

面向对象技术有三个最基本的特性:封装性、多态性、重用性。COM 是一种面向对象的二进制标准,可以实现不同语言开发的对象之间的交互。COM 定义了一种提供和享用通用的软件服务的方法,简化了软件开发,统一了传统的以库程序、其他独立的当地过程系统调用以及远程过程等的服务方式,改变了软件的生产方式。

1. 封装性

所谓封装,就是把客观事物的本质特性封装成抽象的类,通过类的定义并给类的属性和方法加上访问控制。从语法上讲封装就是加上 private、protected、public 等关键词,只让可信的类或者对象操作,对不可信的进行信息隐藏。COM 对象的封装特性是很彻底的,所有的对象状态信息必须通过接口才能访问。

2. 多态性

多态是指为同名的方法提供不同的实现的能力。它从另外一个角度分割了接口和实现,即把“什么”和“如何”两个概念分离开来。COM 的多态性完全通过接口体现出来。而且,COM 分别在三个层次上体现了多态性:接口成员函数、单个接口、一组接口。

3. 重用性

所谓重用性是指,当一个程序单元能够对其他的程序单元提供功能服务时,尽可能地重用原先程序单元的代码,既可以在源代码一级重用,也可以在可执行代码一级重用。

C++语言的重用性位于源代码一级,一个类可以继承于另一个类,从而把父类的功能重用。但对于 COM 组件则情形有所不同,因为 COM 是建立在二进制代码基础上的标准,所以其重用性也必然建立于二进制代码一级。

COM 重用性是指一个 COM 对象如何重用已有的 COM 对象的功能,而不是重复实现老的功能服务。按照 COM 的标准,实现这种重用性有两条途径:包容和聚合。

4. 语言无关性

COM 是一套二进制对象的规范,COM 建立在严格的标准之上,只要提供的组件符合一定的准则,并不关心用什么语言开发系统。不管是什么开发语言,只需按照 Microsoft 的 ActiveX 控件标准开发接口就可以实现 COM 提供的基本功能函数。对于应用开发者而言,只需熟悉基于 Windows 平台的通用集成开发环境,以及控件的属性、方法和事件,就可以完成应用系统的开发和集成。目前,可供选择的开发环境很多,如 Visual C++、Visual Basic、Visual FoxPro、Borland C++、Delphi、C++Builder 以及 Power Builder 等。

5. 进程透明性

客户程序创建 COM 对象具有进程透明特性,不管是进程内组件还是进程外组件,客户程序可以使用一致的方法创建 COM 对象。对于进程内组件,无论是创建过程,还是客户程序对接口函数的调用过程,都可以按照一般的同一进程内部函数调用的过程来理解组件和客户之间的交互操作;但对于进程外组件,实际的情形要复杂得多,因为组件用户程序拥有不同的进程空间,所以,它们之间所有的交互过程都涉及进程之间的通信过程。然而,COM 客户程序创建进程外组件程序成功后,它就得到了组件对象的一个接口指针,通过该指针间接调用组件对象的成员函数,如同调用本进程内的函数一样,这正是 COM 所期望达到的透明效果。

1.2.3 COM 的结构

COM 所建立的是一个软件模块与另一个软件模块之间的链接。当这种链接建立之后,模块之间就可以通过称为“对象接口”(interface on object)的机制来进行通信,进而实现 COM 对象与同一程序或者其他程序甚至远程计算机上另一个对象之间进行交互,而这些对象可以是使用不同的开发语言、以不同的组织方式开发而成的。COM 定义了一种基础性接口,这种接口为所有以 COM 为基础的技术提供了公共函数。COM 允许组件对其他组

件开放其功能调用,既定义了组件如何开放自己以及组件如何跨程序、跨网络实现这种开放,也定义了组件对象的生命周期。

在 COM 规范中,对象与接口是最核心的部分,如图 1-2 所示。COM 接口是指向由对象实现的函数的指针表。COM 对象被很好地封装起来,客户访问 COM 对象的唯一途径是通过 COM 接口。COM 不是一种计算机语言,不是 DLL,不是函数集,也不是类库,而是一种标准和技术方法。

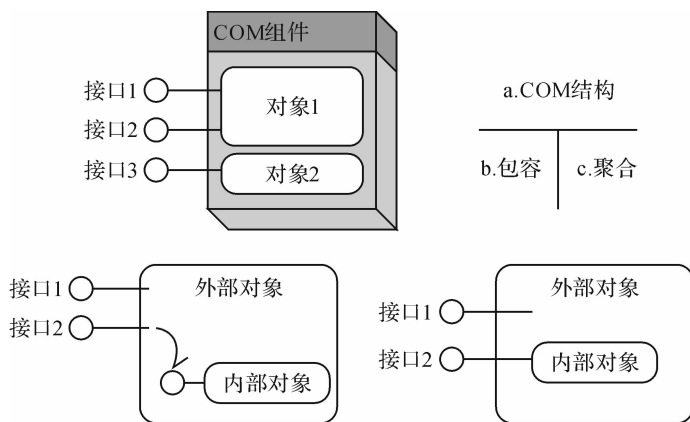


图 1-2 COM 结构与重用模式(刘仁义、刘南,2006)

1. COM 对象

对象是 COM 的基本要素之一。与 C++ 中对象的概念类似,对象是某个类的实例,而类则是一组相关的数据和操作的组合。与 C++ 中对象不同的是,COM 对象的数据完全封装在对象内部,对于对象使用者(通常称为客户)而言是不可见的。COM 服务器和客户可能不同的模块、进程甚至机器上运行,客户通过接口成员函数访问服务器对象的数据。COM 对象的可重用性表现在 COM 对象的包容和聚合,一个对象可以完全使用另一个对象的所有功能,而 C++ 对象的可重用性表现在 C++ 类的继承性。

每一个 COM 对象(的类)由一个 128 位的全局唯一标识符(Globally Unique Identifier, GUID)来标识,COM 客户通过这一标识符来调用并实例化 COM 对象。GUID 通常由系统随机生成,可以在概率上保证其全球唯一性。

2. COM 接口

COM 接口是一组逻辑上相关的函数集合,其函数也被称为接口成员函数。习惯上在接口名的前面冠以“I”表示接口(Interface),如 IUNKNOWN。COM 规范允许一个对象实现多个接口,因此 COM 对象的多态性可以在每个接口上得以实现。利用 COM 对象的多态性,COM 客户可以用一种方法调用不同的 COM 服务。

同 COM 对象一样,每一个 COM 的接口都由一个 128 位的 GUID 来标志。COM 客户通过 GUID 获得接口的指针,再通过接口指针调用相应的成员函数。接口设计必须满足:

- (1)必须直接或间接地从 IUNKNOWN 接口继承;
- (2)接口必须有唯一的标识符号。

COM 具有接口不变性,即一旦分配和公布了 GUID,接口定义的任何因素都不能改变。

3. 类和接口

用 COM 开发意味着使用接口,也可以称为基于接口的设计模型。对象间的所有通信都是通过它们的接口来进行的。COM 接口是抽象的,意味着相关的接口没有实现,和接口相关的代码来自于一个类实现。如何实现接口,对于不同对象是不同的,因此对象只是继承接口的类型,而不是它的实现,这称为类型继承。功能用接口被抽象地构造,并且用类去真正实现。

在 COM 中接口和类通常被当做“做什么”(what)和“怎么做”(how)。接口定义一个对象能做什么,类定义它怎么去做。COM 组件的使用者只需要知道这个接口能做什么,而不需要知道实现的过程。

COM 类提供了一个或多个接口相关的代码,因此功能实体封装在类中。几个类可以有同样的接口,但是它们的实现可能是极不相同的。通过实现这些接口,COM 实现了面向对象的多态性,COM 不支持多重继承概念,然而,这不是一个缺点,因为一个类可以实现多个接口,不同接口可以有不同的功能和实现。

1.3 组件式 GIS

传统的 GIS 虽然在功能上已经比较成熟,但是由于这些系统多是基于较早期的软件技术体系设计开发的,在很大程度上限制了 GIS 软件进一步的发展和应用。在信息技术日益复杂化和综合化的今天,传统 GIS 技术由于开发负担重、集成困难、二次开发语言复杂、普及困难等缺点,面临越来越严峻的挑战。因此,迫切需要一种新型的 GIS 软件技术体系,以满足日益增长的 GIS 应用需求,并跟上软件技术发展的潮流。组件式 GIS 技术正是这样一种全新的 GIS 软件技术体系,它的出现为传统 GIS 面临的多种问题提供了全新的解决思路。

组件式 GIS(COM GIS)是 GIS 技术与组件技术结合的产物。组件式 GIS 的基本思想是:把 GIS 的各种功能模块进行分类,划分为不同类型的控件,每个控件完成各自相应的功能;各个 GIS 控件之间以及 GIS 控件与其他非 GIS 控件之间可以方便地通过可视化的软件开发工具集成起来,形成满足用户特定功能需求的 GIS 应用系统。控件如同一堆各式各样的“汽车部件”,它们分别实现不同的功能(包括 GIS 和非 GIS 功能)。根据需要把实现各种功能的“部件”搭建起来,就构成一辆能够完成各种功能的“信息汽车”(应用系统),前提条件是这些汽车部件必须符合一定的标准规范。

长期以来,由于 GIS 开发周期长、难度大,在一定程度上制约了 GIS 的发展。组件式 GIS 的出现为新一代 GIS 应用提供了工具。组件式 GIS 系统与传统各类 GIS 系统相比具有许多特点。把 GIS 功能适当抽象,以组件形式供开发者使用,将会带来许多传统 GIS 工具无法比拟的优点。