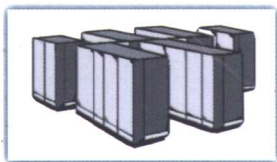


可视化软件开发工具应用丛书



Visual C++ 项目案例导航



特点：从一个标准的软件项目开发过程来讲述软件开发方法，使读者更加容易理解开发技术，掌握一般软件系统的常规实现过程和思路

杨小平 编著

 科学出版社

提供标准的
软件项目
开发过程

可视化软件开发工具应用丛书

Visual C++ 项目案例导航

杨小平 编著

科学出版社

2002

内 容 简 介

作为 Microsoft 公司的重要产品, Visual C++ 被认为是目前最好的软件开发工具之一, 受到软件开发人员的青睐, 包括微软在内的许多软件公司都把 Visual C++ 作为自己产品的开发平台。

本书是从应用出发, 以实际项目开发为背景, 对 C/S 结构、三层分布式结构和网络应用项目开发实例进行详细讨论。在项目实现时深入地讲解概念和技巧, 使读者更自然地掌握那些难以理解的内容。

本书可作为专业技术人员、大专院校计算机专业的本科生、研究生进行项目开发或课程实践、毕业设计时的参考书或教材。

图书在版编目 (CIP) 数据

Visual C++ 项目案例导航/杨小平编著. —北京: 科学出版社, 2002

(可视化软件开发工具应用丛书)

ISBN 7-03-010530-3

I . V… II . 杨… III . C 语言—程序设计 IV . TP312

中国版本图书馆 CIP 数据核字 (2002) 第 041273 号

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>

新 蕾 印 刷 厂 印 刷

科学出版社发行 各地新华书店经销

*

2002 年 6 月第 一 版 开本: 720×1000 1/16

2002 年 6 月第一次印刷 印张: 25

印数: 1—4 000

字数: 476 000

定价: 40.00 元

(如有印装质量问题, 我社负责调换(环伟))

可视化软件开发工具应用丛书

编委会名单

顾问：杨一平 杨小平

主编：郑 宏

编委：（按照姓氏笔画为序）

王立丰 乔建行 许向荣 杨一平 杨小平

李明晴 郑 宏 陶 川 聂慧静 谢 红

鲁荣江 窦 昱 潘 清

丛书前言

当前, 计算机软件技术和网络技术正在飞速发展, 软件系统日趋大型化、复杂化。因此, 社会对软件开发者提出了比以往更高的要求。这个要求就是软件开发者不仅需要熟悉开发工具, 更重要的是必须掌握系统开发的观点和能力。软件开发者不仅要从自身的开发任务考虑软件开发, 更要有项目大局观并成为一个多面手。因此, 每一个软件开发者不仅需要掌握一些基础知识和编程技巧, 更需要细致地了解软件项目开发的整个过程, 在熟悉某个软件开发工具的基础上, 掌握进行软件项目设计和开发全过程的能力。

我们在实际从事计算机软件教学中也感到, 许多计算机和信息专业的学生和初级开发人员虽已具备了一定的开发能力, 但只是掌握了一些初级开发技巧, 由于缺乏实际的、完整的软件项目开发经验, 从事软件系统的分析、设计和开发的能力还十分欠缺, 对项目设计、系统开发理论的理解还不十分深刻。

项目案例教学为解决这一问题提供了较好的手段。由中国信息协会、中国人民大学、首都经贸大学、首都医科大学等从事计算机教学和研究开发工作的知名教授、专家组成的“可视化软件开发工具应用丛书”编委会, 经过精心研究和编写, 将这套软件项目案例丛书奉献给广大读者。这些教授、专家不仅具有丰富的教学经验, 也都具备开发、集成大型应用系统的实际经验, 在计算机教学和开发两个领域都有很高的造诣。

本丛书目前由 6 本图书组成, 覆盖了当前一些主要的可视化软件开发工具, 包括: Visual C++、Visual Basic、Visual FoxPro、Java、Delphi、Lotus Notes 等。丛书通过具体的项目案例分析, 引导读者全面地认识软件项目开发和设计, 使初学者能尽快地熟悉开发工具的主要功能并具备设计、编制软件系统的能力。书中案例都经过了精心选择, 体现了开发工具在不同项目开发中的特点, 注重与网络和数据库应用的结合, 贴近实际应用, 读者通过学习这些案例中很快可以领悟软件项目开发的精髓, 并将系统设计、系统开发的思想和方法应用到自己的实际工作中去。此外, 丛书也介绍了开发工具的编程和使用技巧, 案例的源代码可到科学出版社网站(<http://www.sciencep.com>)的下载区下载。我们用心编写的这套丛书是项目案例教学在软件项目开发领域的一次有益尝试。

本丛书适合刚刚掌握某一软件开发工具基本功能的初学者, 高年级本科生、研究生、初级程序员和高校教师在教学、毕业设计、实习和工作中非常适合使

用，具有中级水平的程序员在实际开发过程中也可参考使用。

由于软件项目设计、开发是一项复杂细致的工作，我们欢迎读者就书中存在的不足进行指正，以更好地服务于广大读者，服务于中国的软件事业。

可视化软件开发工具丛书编委会

2002 年 3 月

前 言

作为 Microsoft 公司的重要产品, Visual C++被认为是目前最好的软件开发工具之一, 受到软件开发人员的青睐, 包括微软在内的许多软件公司都把 Visual C++作为自己产品的开发平台。作为一种程序设计语言, Visual C++不仅支持传统的软件开发方法, 更重要的是它能支持面向对象、可视化的开发风格。因此 Visual C++更应该称作是一个集成开发工具, 它提供了软件代码自动生成和可视化的资源编辑功能, 在使用 Visual C++开发应用程序的过程中, 系统还为我们生成了大量的各种类型的文件。

Visual C++具有以下优点: 提供了面向对象的应用程序框架 MFC (Microsoft Foundation Class), 简化了程序员的编程工作, 提高了模块的可重用性; 提供了基于 CASE 技术的可视化软件自动生成和维护工具 AppWizard、ClassWizard、Visual Studio、WizardBar 等, 实现了直观、可视的程序设计风格, 方便地编写和管理各种类, 维护程序源代码; 已成为工业标准的 MFC 类库得到了开发商和软件开发工具的支持; 封装了 Windows 的 API 函数、USER、KERNEL、GDI 函数, 简化了编程时创建、维护窗口的许多复杂的工作。

由于 Visual C++本身的复杂性和概念上的广泛性, 编程人员难以把握它的复杂机制。本书并不想写成是 Visual C++程序员的观念手册, 只是为了理解上的要求, 在开始时对必要的概念作解释。同时, 本书也不是传统意义上的教科书, 结构上没有刻意安排“从入门到精通”的课堂式学习过程, 这些书已经很多了; 另外, 本书也不是单纯介绍 Visual C++的编程技巧, 虽然对程序员来说, 许多技巧是那么的引人入胜。本书是从应用出发, 以实际项目开发为背景, 对 C/S 结构、三层分布式结构和网络应用项目开发实例进行详细讨论。在项目实现中再反过来讲解概念和技巧, 使读者更自然地掌握那些难以理解的内容。

本书分成四个部分, 第一部分为基础知识介绍, 包括三章, 对面向对象的概念、C++的必要知识做了介绍, 并在第三章中对线程编程的概念进行详细介绍, 目的在于为后面的学习做准备。第二部分讨论了在传统的 C/S 模式下的数据库应用系统的开发, 用三章内容介绍一个图书管理信息系统的实现。第三部分讲述的是三层分布式结构下的应用系统实现, 介绍了分布式系统特点和组件概念及应用, 该部分共五章, 主要以考试系统的设计和实现作为例子, 讲述在 Visual C++环境下如何开发 COM+应用程序。第四部分讲述网络环境下的应用, 分四章详细介绍了 TCP/IP 的协议, Socket 函数和 MFC SOCKET 以及它们的使用方法,

以网上聊天室的实例介绍它们的应用。同时，为了便于读者阅读，本书还在附录部分将 Visual C++ 的重要的编程环境部分，以手册形式提供出来。

本书可作为专业技术人员，大专院校计算机专业的本科生、研究生进行项目开发或课程实践、毕业设计时的参考书。

本书由杨小平编著，窦昱、李明晴参加了文字编写工作。北京理工大学计算机系李凤霞教授认真审阅了书稿，并提出宝贵的修改意见；同时得到首都经贸大学信息学院杨一平教授、北京航空航天大学王强副教授的热情帮助，在此表示衷心感谢！

由于作者水平所限，谬误之处在所难免，望读者批评指正。

编 者

2002.5

目 录

第一部分 Visual C++基础

第一章 C++和面向对象设计	3
1.1 类与对象	3
1.2 封装	3
1.3 继承	4
1.4 虚函数与多态	7
第二章 Visual C++基础	11
2.1 Visual C++工程	11
2.2 MFC 特点	12
2.3 功能强大的 Wizard	13
2.4 Visual C++的调试技术	24
第三章 多线程编程	43
3.1 线程基本知识	44
3.2 线程的调度	54
3.3 MFC 的多线程编程	62

第二部分 传统的 C/S 模式下的数据库系统

第四章 图书管理系统的需求分析	67
4.1 图书管理系统的需求分析图	67
4.2 图书管理系统的数据分析	69
第五章 图书管理系统的模块划分	71
第六章 图书管理系统的具体实现	73
6.1 建立数据库	73
6.2 图书管理系统的整体介绍	79
6.3 建立工程	79
6.4 登录模块的实现	82
6.5 登录模块的具体实现	90
6.6 读者资料维护模块的实现	95

6.7 还书模块	109
----------------	-----

第三部分 分布式应用系统——考试系统

第七章 传统的 C/S 结构与基于组件的分布式应用结构的比较	119
第八章 基于组件的三层分布式应用系统的设计	122
8.1 三层体系结构	122
8.2 商业组件	125
8.3 分布式商业组件	130
第九章 COM、DCOM 和 COM+	134
9.1 COM 和 DCOM	134
9.2 COM+	136
第十章 考试系统的应用需求分析	152
10.1 问题描述	152
10.2 面向对象的设计	152
10.3 组件的实现	160
第十一章 考试系统的具体实现	162
11.1 Question 组件	162
11.2 UserInfo 组件的实现	179
11.3 Exam 组件的实现	192
11.4 客户程序的实现	220

第四部分 网络应用系统

第十二章 网络基本原理与协议	227
12.1 Internet	227
12.2 TCP/IP 协议	228
第十三章 WinSock 编程机制	235
13.1 WinSock 描述字	235
13.2 Socket 的系统数据结构	235
13.3 Socket 的地址说明	236
13.4 Socket API 的函数说明	236
13.5 Socket 的编程框架	251
第十四章 MFC Socket 编程机制	266
14.1 CAsyncSocket 类	266

14.2 CSocket 类	269
第十五章 实例	274
15.1 设计思路	274
15.2 网络聊天程序	278
15.3 程序代码	280
附录一 本书涉及的 MFC 类的层次图	346
附录二 数据库相关知识介绍	346
附录三 C++ 的异常类	368
附录四 Windows 网络知识介绍	369
参考文献	387

第一部分 Visual C++基础

Visual C++ 作为一个优秀的应用开发工具，得到了无数程序员的喜爱。Visual C++ 不仅仅是一个编译器，也是一个全面的应用程序开发环境和工具，使用它，可以充分利用 C++具有的面向对象特性开发出专业级的 Windows 应用程序。为了能充分利用这些特性，必须理解 C++ 程序设计语言。而要掌握 C++，就必须掌握 Microsoft 基本类库 (MFC) 的层次结构。该层次结构包容了 Windows API 中的用户界面部分，使用户能够很容易地以面向对象的方式建立 Windows 应用程序。这种层次结构适用于所有版本的 Windows 并彼此兼容。用 MFC 所建立的代码是完全可移植的。

本部分主要介绍 C++面向对象设计、Visual C++的特点和使用方法以及多线程编程的原理与方法。请读者注意的是，本书第一部分是介绍 Visual C++ 的基础知识为主，目的在于帮助读者能顺利理解实例，而具体的实例将在后面各个部分分别进行详细讨论和说明。

第一章 C++和面向对象设计

20 世纪 90 年代，随着面向对象概念的提出，许多在过程式语言中无法或很难解决的问题都得到了相当完美的解决。从而使得面向对象编程技术得到空前的发展和普及。出现了很多的面向对象语言，如：Small Talk, Eiffel 以及 C++等等。C 语言自从它诞生那天起，就一直得到广大程序员的喜爱，主要在于它的灵活和高效。因而 C++一出来，就立即得到很多 C 程序员的喜爱。

对于面向对象，作者认为重点是要理解面向对象的概念，很多人一听到面向对象，就会将其和继承联系起来或者等同起来，实际上面向对象所涉及的概念要复杂得多，而继承只不过是面向对象的基础而已。

关于面向对象有几个非常基础的概念：类与对象、封装、继承、虚函数与多态。理解这些概念，有助于提高程序的质量，特别是提高软件的可复用性与可扩充性。

本章主要阐述在 C++环境下，如何设计和编写面向对象的程序。

1.1 类与对象

对象 (Object) 是类 (Class) 的一个实例 (Instance)。所谓类就是许多具有共同属性和操作特性的事物集合的抽象表示，而对象是该类中的一个具体的个体而已。如果将对象比作房子，那么类就是房子的设计图纸。所以面向对象程序设计重点是类的设计，而不是对象的设计。

设计孤立的类是比较容易的，难的是正确设计基类及其派生类。因为有些程序员搞不清楚“封装”、“继承”、“多态”这些概念。这里就对每个概念做简要的介绍。要详细了解如何使用 C++来进行面向对象的编程，可以参考相关书籍。

1.2 封 装

类可以将数据和函数封装在一起，其中函数表示了类的行为或者动作。类提供关键字 `public`、`protected` 和 `private`，用于声明哪些数据和函数是公有的、受保护的或者是私有的。这样可以达到信息隐藏的目的，因而可以使得类仅仅公开外界需要和应该知道的内容，而隐藏其他实现细节和内容。我们不可以滥用类的封

装功能，不要把它当成杂货铺，什么东西都往里扔。

如类 **Person** 是一个表示人的类，那么它的结构如下：

```
class Person
{
private:
    char      *Name;
    char      *Address;
    unsigned long SecuritySocietyNo;
    CDate      BirthDate;
    .
    .
    .
public:
    void SetName(const char theName[]);
    char* GetName() const;
    void SetAddress(const char theAddress[]);
    char* GetAddress() const;
    CDate GetBirthDate() const;
    Unsigned long GetIdentification() const;
    Void SetIdentification(unsigned long newID);
    Void Print() const;
    .
    .
    .
};
```

1.3 继 承

如果 **A** 是基类，**B** 是 **A** 的派生类，那么 **B** 将继承 **A** 的数据和函数。示例程序如下：

```
class A
{
```

```
public:
    void Func1(void);
    void Func2(void);
};

class B : public A
{
public:
    void Func3(void);
    void Func4(void);
};

// Example
main()
{
    B b;
    b.Func1();
    b.Func2();
    b.Func3();
    b.Func4();
}
```

这个简单的示例程序说明了一个事实：C++的继承特性可以提高程序的可复用性。正因为继承太有用、太容易用，才要防止乱用继承。有人在使用继承时界定了一些规则：

(1) 如果类 A 和类 B 毫不相关，不可以仅为了使 B 的功能更多些而让 B 继承 A 的功能。

(2) 如果类 B 有必要使用 A 的功能，则要分两种情况考虑：

① 若在逻辑上 B 是 A 的“一种”(a kind of)，则允许 B 继承 A 的功能。如男人(Man)是人(Human)的一种，男孩(Boy)是男人的一种。那么类 Man 可以从类 Human 派生，类 Boy 可以从类 Man 派生。其示例程序如下：

```
class Human
{
```

```
...
};
class Man : public Human
{
    ...
};
class Boy : public Man
{
    ...
};
```

② 若在逻辑上 A 是 B 的“一部分”(a part of)，则不允许 B 继承 A 的功能，而是要用 A 和其他东西组合出 B。例如眼 (Eye)、鼻 (Nose)、口 (Mouth)、耳 (Ear) 是头 (Head) 的一部分，所以类 Head 应该由类 Eye、Nose、Mouth、Ear 组合而成，不是派生而成。示例程序如下：

```
class Eye
{
public:
    void Look(void);
};
class Nose
{
public:
    void Smell(void);
};
class Mouth
{
public:
    void Eat(void);
};
class Ear
{
public:
```