



新 Visual C++ 游戏设计

荣钦科技 编著



北京科海电子出版社

内 容 简 介

当今,游戏设计已经成为许多玩家的梦想。本书用浅显生动的语言、经典的案例带您轻轻松松进入游戏设计的美妙世界。

本书参照开发《仙剑奇侠传》所需要技术,如镂空动画、半透明动画、碰撞检测、斜角卷动的地图、人工智能、3D 动画、3D 音效等,对游戏的开发原理和过程进行了系统而又详细的介绍。看完本书后,读者能清楚地了解游戏设计的各种知识,包括 Visual C++ 的基本操作、游戏画面的坐标系统、规划游戏的主要架构,以及电脑游戏的画面显像、结合 DirectX 来设计游戏等游戏设计中会使用到的各种理论。

本书理论与实例并重,还包括多种设计技巧,非常适合作为游戏设计的入门图书,也可供多媒体开发人员参考使用。

盘 名: 新 Visual C++ 游戏设计

作 者: 荣钦科技

责任编辑: 马首鳌

排 版: 房利萍

光盘制作: 马首鳌

咨询电话: (010) 82896445-8407

出 品: 北京科海电子出版社

印 刷 者: 北京科普瑞印刷有限责任公司

发 行 者: 新华书店总店北京发行所

开 本: 异 16 印张: 字数: 千字

版 次: 2003 年 5 月第 1 版 2003 年 5 月第 1 次印刷

印 数: 0001 ~ 0000

盘 号: ISBN 7-900107-91-6

定 价: 00.00元 (1CD / 配套手册)

改编者推荐序

少时曾流连忘返于街机，但本人非常愚钝，花钱无数却不得入其道，屡被街机中的恶人打败，可以说是乘兴而来，败兴而归。偶尔也对街机中的游戏是如何进行的比较感兴趣，但限于当时的环境和学到的知识，百思不得其解。

后来进了大学，攻读计算机专业，总算了解了街机其实就是一种专用于游戏的计算机。为了一解少时的谜团，经常出没于计算机房和图书馆，也用 Turbo C 编写一些当时自认为比较高明的小程序。后又与舍友节衣缩食合买了一台 486，恰逢当时《仙剑奇侠传》非常流行，因此屡次在寝室里通宵达旦地战斗，并在一帮朋友面前胡侃地李逍遥的故事。

也许是年少轻狂，学了一些编程知识后便觉得游戏不过如此，总想一展身手，编写一两套游戏。于是读遍金庸古龙的大作，并借人家的情节编写了一个小故事，然后试着着手编写程序。可坐在计算机前却不知从何下手，特别是图形的显示（那时还是 DOS 环境），这时才觉得自己所学有限，于是跑遍武汉所有书店，总算买到一本《用 Turbo C 设计游戏》一书。回来后如获至宝，一口气读完，并照书中的范例编写了一些小游戏。严格说来，那算不上是游戏，只能说是几个图形程序而已，因为全书讲解的都是图形的显示，至于游戏的机理，几乎没怎么讲。后来又由于临近考试，再加上计算机技术发展一日千里，所以开发游戏只能是有头无尾了。不过，我至今仍感激那段经历，虽然尝试没有成功，但确实学到了很多知识，后来所有的专业考试都顺利通过，应该还有它的一点功劳吧。

如今，看了手头的这本书，总算恍然大悟，特别是以前不知道的碰撞检测、人工智能、镂空和半透明动画、卷动的地图、斜角背景等。也偶尔会产生一些幻想，要是我当时有这本书，现在又是怎样呢？但有了这本书，我又能如何呢？那些只是客观因素，只有不停的奋斗，才会有丰硕的果实。

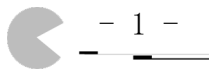
但我们不得不承认，在游戏方面，我们确实落后于我们的台湾同胞，更别说日本了。尤其是在知识的应用和系统的整合上，所以在这里真心的向大家推荐这本书，希望能让广大游戏设计爱好者不要留下我那样的遗憾。

在这本书中，将很多开发游戏所需的知识整合在一起，如物理中的运动分析，数学中的轨迹方程等。而且还比较系统地讲解了游戏的整体设计规划，是一本非常适合游戏设计爱好者入门的好书。无论是 RPG、即时战斗，还是网络游戏等中用到的某些技术在书中都进行了讲解，如人物的移动、碰撞的检测等。

好了，不占用大家的宝贵时间了，还是一起来看这本书吧！

改编者

2003. 3. 24 日



- 1 -

开启游戏设计的筑梦王国

游戏设计已经成为当今许多玩家的梦想。笔者相信有许多玩家凭着自己的崇高游戏理念与构思，想在游戏设计的领域里闯出一片天空，但自己却苦无能力将这种构思理念变成游戏中的故事、剧情、玩法与机制，以至于造成半途而废的人也越来越多了。

基于这个出发点，为了让这些人能重拾游戏设计的信心，本书将以各种层面的技术与范例，来告诉您游戏设计的知识与技巧，邀您一同进入游戏设计这个奇妙的领域，把这些认为不可能的梦想变为现实吧！

看了这本书之后您能清楚了解游戏设计的基本知识。如 Visual C++ 的基本操作、游戏画面的坐标系统、如何设定游戏的主要架构等，这些知识对于刚开始游戏设计的新手有极大的帮助。

此外，我们也会谈到各类的设计技巧及技术，包括电脑游戏的画面显像、如何结合 DirectX 来设计游戏及游戏设计中会使用到的各类理论与实例。

在专题制作部分，我们以常见的游戏作为书中的游戏范本，利用简单又明了的范例程序，来让您可以轻轻松松地学习游戏设计的技术与乐趣。笔者深信这是一本非常实用的游戏设计宝典，只要您肯花时间好好研读，它绝对能带您一窥游戏设计的美妙世界。最后，祝您在学习游戏设计的旅途上处处充满喜悦。

荣钦科技 敬上



目 录

第 1 章 开始设计游戏	1
1.1 电脑游戏设计导论	2
1.1.1 为什么使用 Visual C++ 来设计游戏	2
1.1.2 与其他函数库的结合	3
1.1.3 游戏舞台的坐标系统	6
1.2 游戏架构与规划	8
1.2.1 游戏的戏剧表现	8
1.2.2 游戏主题的确立	9
1.2.3 剧情的发展过程	9
1.2.4 主要人物的突出	10
1.2.5 悬念的安排	10
1.2.6 节奏的快慢	11
1.2.7 风格的呈现	12
1.2.8 摄影机的操控	12
1.2.9 视角的应用	13
1.3 游戏剧本的规划与设计	13
1.3.1 游戏规划的细则	14
1.3.2 界面的设计	14
1.3.3 道具的设计	14
1.3.4 真实与虚构	15
1.3.5 游戏的死角	15
1.3.6 游戏的交互性	16
1.3.7 游戏中的奖励	16
1.3.8 游戏中的对话	16
1.4 构建游戏窗口	17
1.4.1 设置 Visual C++ 应用程序中的头文件	17
1.4.2 建立程序项目	18
1.4.3 应用程序与窗口	24

第 2 章	Visual C++基本绘图技巧.....	29
2.1	MFC 绘图类	30
2.1.1	MFC 的 GDI 类.....	30
2.1.2	MFC 的 CDC 类.....	30
2.2	位图的使用技巧.....	39
2.2.1	在窗口中显示位图.....	39
2.2.2	镂空图的制作技巧.....	49
2.2.3	半透明效果的制作.....	54
第 3 章	动画的制作	63
3.1	控制动画显示时间.....	64
3.1.1	定时器的运用.....	64
3.1.2	使用单个定时器显示动画.....	65
3.1.3	使用多个定时器显示动画.....	67
3.1.4	利用回调函数显示动画.....	70
3.2	特殊动画效果.....	72
3.2.1	镂空动画技巧.....	72
3.2.2	半透明动画技巧.....	80
3.3	游戏背景的卷动.....	87
3.3.1	单一背景卷动技术.....	87
3.3.2	多重背景卷动技术.....	89
第 4 章	物理动作设计	95
4.1	物体的移动.....	96
4.1.1	匀速运动	96
4.1.2	变速运动	99
4.1.3	动量	100
4.1.4	重力	101
4.1.5	重力加速度.....	103
4.1.6	摩擦力	103
4.1.7	下落与弹跳.....	108
4.2	物体间的碰撞.....	111
4.2.1	通过范围检测碰撞.....	112
4.2.2	通过颜色检测碰撞.....	115
4.2.3	通过行进路线检测碰撞.....	121
4.2.4	与斜面碰撞后的速度.....	122



4.3	粒子的应用.....	127
4.3.1	粒子的定义.....	127
4.3.2	雪花纷飞.....	128
4.3.3	放焰火.....	131
第 5 章	管理程序中的消息.....	137
5.1	键盘消息.....	138
5.1.1	处理键盘输入消息.....	138
5.1.2	使用键盘控制人物移动.....	139
5.2	鼠标消息.....	146
5.2.1	处理鼠标输入消息.....	146
5.2.2	鼠标版“电流急急棒”.....	148
5.2.3	改变鼠标指针.....	155
第 6 章	剖析人工智能.....	161
6.1	移动型人工智能.....	162
6.1.1	固定模式移动.....	162
6.1.2	追逐移动.....	169
6.1.3	躲避移动.....	176
6.2	行为型人工智能.....	177
6.2.1	物体的动作行为.....	177
6.2.2	路径查找的概念.....	188
6.2.3	查找最佳路径.....	189
6.3	策略型人工智能.....	206
6.3.1	计算机的决策方式.....	207
6.3.2	五子棋游戏设计.....	211
第 7 章	进入 3D 的世界.....	229
7.1	初探 DirectX.....	230
7.1.1	DirectX SDK 简介.....	230
7.1.2	DirectX 的特色.....	231
7.2	DirectX 8.0.....	231
7.2.1	浅谈 Direct3D.....	232
7.2.2	建立 Direct3D 环境.....	232
7.3	绘制基本几何图形.....	238
7.3.1	描绘三角形.....	238

7.3.2	旋转的三角形.....	242
7.3.3	描绘立方体.....	246
7.4	打光.....	252
7.4.1	打光的概念.....	252
7.4.2	光的属性.....	254
7.4.3	光的色彩属性.....	255
7.4.4	光线投射的范例.....	257
7.5	材质贴图.....	261
7.5.1	材质的基本概念.....	261
7.5.2	材质过滤.....	263
7.5.3	材质 Stage.....	264
7.5.4	材质贴图的范例.....	266
7.6	模型.....	270
7.6.1	构成三角形的方法.....	270
7.6.2	加载.x 文件.....	272
7.6.3	绘制模型文件.....	273
第 8 章	使用 DirectDraw	277
8.1	建立 DirectDraw 程序.....	278
8.1.1	建立 DirectDraw 的第一步.....	278
8.1.2	建立 DirectDraw 对象.....	279
8.1.3	设置程序协调层级.....	280
8.1.4	设置屏幕显示模式.....	281
8.1.5	绘图页的概念.....	281
8.1.6	建立绘图页.....	282
8.2	绘图页的基本应用.....	285
8.2.1	在幕后暂存区中加载位图.....	285
8.2.2	贴图与翻页.....	287
8.2.3	清除绘图页.....	294
8.3	DirectDraw 进阶应用.....	295
8.3.1	颜色键.....	296
8.3.2	DirectDraw 动画.....	300
8.3.3	存取绘图页的方式.....	304
第 9 章	使用 DirectSound	309
9.1	建立 DirectSound.....	310



9.1.1	建立 DirectSound 程序的第一步	310
9.1.2	建立 DirectSound 对象	311
9.1.3	设置程序协调层级	312
9.1.4	缓冲区的概念	312
9.1.5	建立主缓冲区	314
9.1.6	加载 WAVE 声音文件	315
9.1.7	建立次缓冲区	319
9.1.8	加载声音到次缓冲区	320
9.2	声音的播放与控制	322
9.2.1	播放声音	322
9.2.2	制作混音效果	322
9.2.3	控制声音	325
9.3	使用 3D 音效	332
9.3.1	3D 音效概念	332
9.3.2	建立倾听者	334
9.3.3	建立发声者	335
第 10 章	使用 DirectInput	341
10.1	建立 DirectInput 程序	342
10.1.1	建立 DirectSound 的第一步	342
10.1.2	建立 DirectInput 对象	343
10.1.3	建立输入设备对象	344
10.1.4	设置数据格式	345
10.1.5	设置程序协调层级	345
10.1.6	取得输入设备的控制权	346
10.2	使用键盘与鼠标	346
10.2.1	取得键盘输入	346
10.2.2	取得鼠标输入	351
10.3	使用摇杆	357
10.3.1	取得摇杆设备	357
10.3.2	列出摇杆组件	359
10.3.3	取得摇杆输入	364
10.3.4	设置无效范围	368
第 11 章	使用 DirectPlay 和 DirectShow	371
11.1	DirectPlay	372

11.1.1	为什么要使用 DirectPlay.....	372
11.1.2	DirectPlay 的网络拓扑.....	372
11.1.3	网络联机游戏的组成.....	374
11.1.4	DirectPlay 的使用方法.....	375
11.1.5	范例介绍.....	382
11.2	DirectShow.....	395
11.2.1	DirectShow 的架构.....	396
11.2.2	播放影片.....	397
11.2.3	播放 MP3.....	399
第 12 章	游戏制作集锦.....	401
12.1	程序中的自定义函数.....	402
12.1.1	初始化 DirectX 对象.....	402
12.1.2	建立 DirectDraw 幕后暂存区.....	408
12.1.3	建立 DirectSound 次缓冲区.....	410
12.1.4	设置颜色键.....	412
12.2	豪华版“电流急急棒”.....	413
12.2.1	游戏功能介绍.....	413
12.2.2	实现游戏功能的方法.....	415
12.2.3	程序内容说明.....	417
12.3	太空射击游戏.....	425
12.3.1	游戏功能介绍.....	425
12.3.2	卷动背景的设计.....	427
12.3.3	怪物的产生与移动.....	427
12.3.4	子弹的产生.....	428
12.3.5	检测碰撞的方法.....	429
12.3.6	程序编写.....	431
第 13 章	游戏专题——2D 斜角游戏.....	453
13.1	游戏制作前的工作.....	454
13.1.1	游戏规划.....	454
13.1.2	游戏流程.....	455
13.1.3	工作分配.....	456
13.1.4	整合与维护.....	457
13.2	背景横向卷动.....	457
13.2.1	横向卷动的原理.....	457



13.2.2	横向卷动的算法.....	459
13.3	斜角地图贴图.....	462
13.3.1	游戏地图拓扑.....	462
13.3.2	地图索引.....	463
13.3.3	斜角地图的实现.....	464
13.4	障碍物贴图.....	469
13.4.1	障碍物索引.....	469
13.4.2	障碍物贴图算法.....	471
13.5	人物贴图.....	473
13.5.1	人物移动的模式.....	473
13.5.2	索引图的规则.....	474
13.5.3	人物贴图算法.....	475
13.6	排序贴图.....	479
13.6.1	排序索引.....	479
13.6.2	障碍物与人物的贴图方式.....	481
13.7	障碍物判断.....	486
13.7.1	障碍物的碰撞.....	485
13.7.2	使用障碍物索引.....	485
13.7.3	碰撞处理.....	488
13.8	斜角游戏范例程序.....	493
附录 A	DirectInput 按键索引值与按键列表.....	499

第 1 章

开始设计游戏

- ▶ 1.1 电脑游戏设计导论
- ▶ 1.2 游戏架构与规划
- ▶ 1.3 游戏剧本的规划与设计
- ▶ 1.4 构建游戏窗口

1.1 电脑游戏设计导论

在游戏所创造出的虚幻世界里，我们可以进行一趟惊心动魄的冒险行动，或者是与众多的武功高手对决，或者是为了保卫地球而与外星入侵者打场星际大战等等。

但您知道游戏是怎样设计出来的吗？在这里，就让我们来为您揭开游戏程序设计的序幕。

1.1.1 为什么使用 Visual C++ 来设计游戏

C 语言是一种高效率的计算机程序设计语言，而 Visual C++ 则是在 C 语言的基础上开发出一套功能强大的程序设计工具。对于追求运行速度与表现效果的游戏程序，Visual C++ 可以说是最佳的开发工具。

在目前的商业游戏程序设计领域中，都是使用 Visual C++ 作为开发工具，下面就来说明 Visual C++ 在游戏程序开发上所具备的优点。



优异的运行速度

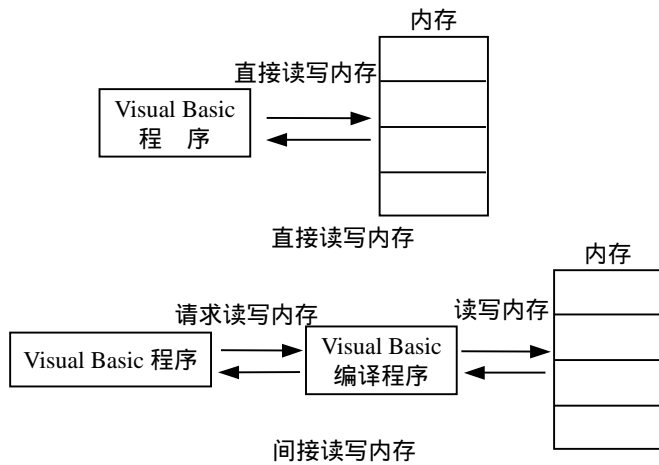
如果您使用过 Visual Basic，那么当您在刚开始使用 Visual C++ 来编写程序的时候，可能会不知从何下手，因为 Visual C++ 没有像 Visual Basic 一样有许多现成的对象和函数可以使用。Visual Basic 中的现成对象和函数，都是为了让 Visual Basic 程序员们能够更轻松地开展程序而事先准备的，但这些功能完整的对象和函数在执行时会占用很多计算机资源，降低程序的运行效率。

而 Visual C++ 因为没有这些通用的对象和函数，所以开发出的程序运行速度非常快。在后面的章节中，您将会发现即使是需要进行复杂的计算的图形处理程序，在运行时仍然非常流畅。



直接访问内存

在 C 语言中可以直接用指针来读写计算机内存，而且还提供了许多有关内存处理的函数，直接读写内存的最大好处就是速度快。为什么呢？其实在任何程序中都必须读取内存，只不过其他程序设计语言都为了防止发生内存读写不当（读写了系统和其他程序正在使用中的内存，造成程序运行错误或者系统死机）而使用间接管理内存的方式，这样虽然安全，但却比较耗费时间。读写内存的过程如下图所示：



直接读写内存的方式可以说是一把双刃剑，使用得当将大大提升程序的运行速度，但若是使用不当，将会造成程序运行出错或者计算机死机。

MFC 函数库

MFC (Microsoft Foundation Class) 是 Visual C++ 中的一套完整的函数库，使用 MFC 可以让程序员很轻松地构建应用程序的框架，降低使用 Windows API 开发应用程序的难度，非常适合初学 Visual C++ 或者游戏程序设计的人。因此，本书将以使用 MFC 编程为主，讲解电脑游戏的设计与开发。

使用 Windows API

在 Visual C++ 中使用 Windows API 是相当容易的，只要在项目中设定所要使用的函数库并引用正确的头文件，就可以使用 Windows API 中的函数与数据类型。有关设定项目连接的函数库以及引用头文件的方法，我们将会在后面的章节中介绍。

在 Visual C++ 开发环境之下，可以用具有完整面向对象属性的 C 语言作为程序设计的主体，而且可以很容易地使用 MFC、Windows API，甚至整合其他技术（如 DirectX 和 OpenGL）来开发游戏程序。因此，在未来的一段时间之内，Visual C++ 仍然会是开发程序的最佳工具。

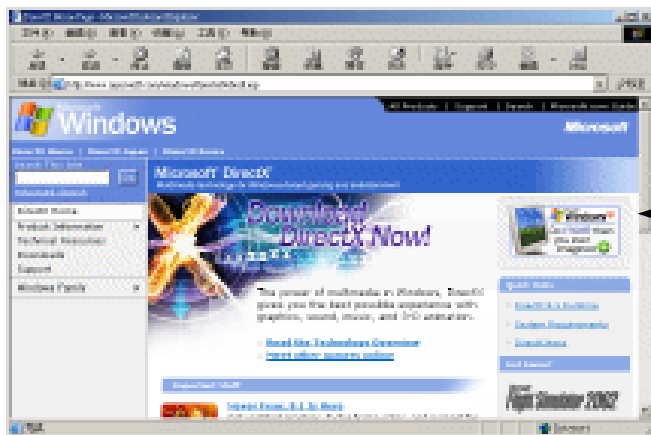
1.1.2 与其他函数库的结合

单纯使用 Visual C++ 是无法设计出优美且高效率的游戏的，目前的游戏开发中都运用了 DirectX 来加强多媒体功能，或者是使用 OpenGL 来制作更流畅的 3D 效果。在本小节中，

将介绍这两种开发游戏程序不可或缺的多媒体技术。

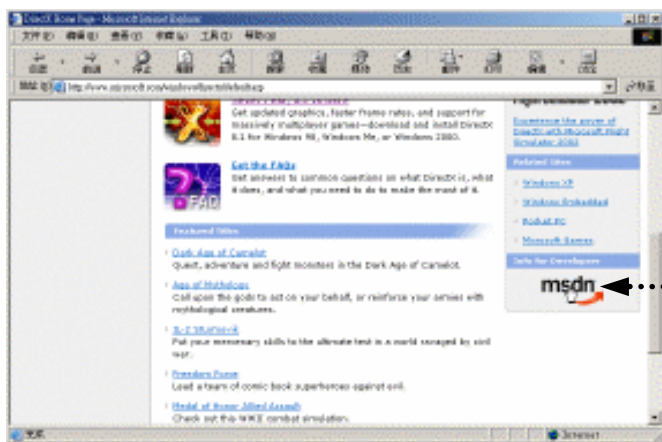
DirectX SDK

DirectX SDK (DirectX Software Development Kit) 是微软推出的一套在 Windows 中开发多媒体程序的 API 函数集, 目前最新版本是 9.0 版, 本书将以 8.0 版来讲解。在 DirectX 中, 包含了很多组件用来处理 2D 与 3D 影像、声音、网络, 以及控制各种输入输出设备, 可以说是开发游戏程序时不可或缺的工具包。在本书后面的章节中, 将会介绍 DirectX 中的 DirectDraw、DirectSound 和 DirectInput 工具, 您还可以通过访问微软的网站 <http://www.microsoft.com/windows/directx/default.asp> 来获得更多有关 DirectX 的信息, 如下图所示:



按住鼠标左键,
向下拖曳

如果您的系统中还没有安装 DirectX SDK, 则可以按照下列步骤来下载 8.0 版的 DirectX SDK。



点击此链接



下载完成之后，就可以将 DirectX SDK 安装到您的计算机上。在后面的章节里，将介

绍如何引用 DirectX 的头文件和函数库。



OpenGL API

OpenGL (Open Graphic Library) 是显示卡硬件厂商共同制定出的一套绘图函数库, 它可以说是专门为开发 3D 游戏程序而设计的, 使用 OpenGL 的 API 函数可以制作出更为流畅的 3D 画面与场景。与 DirectX 相比, 使用 OpenGL 开发出的 3D 游戏在执行速度上更胜一筹, 而且许多操作系统也支持 OpenGL, 所以它还拥有可跨平台作业的优点。但由于 3D 加速卡的厂商制定了太多的规格, 以至于一套软件在开发的时候, 就必须使用各厂商的 OpenGL, 才能确保游戏能够在各种 3D 加速卡上正常运行, 所以这种方式也扼杀了其兼容性, 并且增加了开发成本。因此在 Windows 操作系统中编写多媒体应用程序时, DirectX 还是一个不错的选择。

对游戏程序开发所必须结合的相关技术有了基本的认识之后, 我们将介绍在 2D 游戏程序中占重要地位的游戏舞台坐标系统。

1.1.3 游戏舞台的坐标系统

在开始设计游戏时, 就要先确定游戏中各对象的位置, 这都是通过确定其坐标来完成的, 所以在编写游戏程序之前, 先讲解有关程序中的窗口区域与坐标系统的概念。



屏幕区、窗口区与客户区

任何一个游戏程序, 不论是使用全屏模式还是窗口模式, 都要建立一个窗口。当窗口建立之后, 对程序而言, 显示屏幕便被划分为三个区域:

- 屏幕区 (Screen)
- 窗口区 (Window)
- 客户区 (Client)

这三个区域在 Windows 操作系统中的位置如下图所示:

