



Sams Teach Yourself
DirectX 7 in 24 Hours

DirectX 7

- 创建3D游戏和图像的方法
- 创建滚动背景
- 用DirectSound增加音效
- 增加光效
- 运用3D对象
- 检测冲突
- 创建游戏走廊
- 加入视频
- 创建3D声音
- 在网页中应用多媒体

速成教程

时光飞逝 时不我待

Robert Dunlop
(美) Dale Shepherd 著
Mark Martin

孙守迁 王剑 杨勤 等译



机械工业出版社
China Machine Press

SAMS

DirectX 7 速成教程

Robert Dunlop

(美) Dale Shepherd 著

Mark Martin

孙守迁 王 剑 杨 勤 等译

匡继勇 罗仕鉴 审校



机械工业出版社
China Machine Press

DirectX 是多媒体应用软件, 由 Microsoft 公司提供, 用于创建娱乐和其他基于 Windows 的多媒体应用。本书介绍了目前 DirectX 的所有内容, 其内容包括: DirectX 的组成和各部分作用、DirectDraw 组件的介绍、系统的控制、游戏循环的创建、DirectDraw 动画技术、DirectSound 介绍以及 DirectMusic 的应用等。本书条理清晰、由浅入深地介绍了 DirectX 7, 并结合大量实例, 帮助读者有效地使用 DirectX 7。

本书所附光盘包括了 Microsoft 的 DirectX 7 SDK 以及作者的源代码和示例。

Robert Dunlop, et al: Sams Teach Yourself DirectX 7 in 24 Hours.

Authorized translation from the English language edition published by Sams, an imprint of Macmillan Computer Publishing U.S.A.

Copyright © 2000 by Sams Publishing. All rights reserved.

Chinese simplified language edition published by China Machine Press.

Copyright © 2001 by China Machine Press.

本书中文简体字版由美国麦克米伦公司授权机械工业出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

本书版权登记号: 图字: 01-2000-0080

图书在版编目 (CIP) 数据

DirectX 7 速成教程 / (美) 丹鲁波 (Dunlop, R.) 等著; 孙守迁等译. - 北京: 机械工业出版社, 2001.10

书名原文: Sams Teach Yourself DirectX 7 in 24 Hours

ISBN 7-111-09356-9

I. D... II. ①丹...②孙... III. 多媒体-软件工具, DirectorX 7-教材 IV. TP311.56

中国版本图书馆 CIP 数据核字 (2001) 第 065529 号

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑: 华章

北京忠信诚胶印厂印刷·新华书店北京发行所发行

2002 年 1 月第 1 版第 1 次印刷

787mm×1092mm 1/16·24.25 印张

印数: 0 001-4 000 册

定价: 53.00 元 (附光盘)

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

前 言

参与游戏设计和多媒体应用是多么激动人心！由于计算机硬件的优化，计算机速度提升的周期越来越短，其中视频卡在计算机硬件中是发展最快的。

当然，多媒体并不只意味着视频，还有声音、用户输入和音乐。网络为视频和动画创造了良机。

多媒体应用开发者的工作并没变得更容易，技术越来越复杂和新颖，但帮助我们应用这些新技术的工具也应运而生。Microsoft 的 DirectX 平台已经发展得非常优秀。

市场上有许多工具和 SDK 处理多媒体应用的不同方面，但没有一种像 DirectX 这样强大和全面。拥有 DirectX，你可以使用完整的 API 写出新一代的多媒体游戏和应用。其 3D 部分可以为用户创造出标准的 3D 世界。DirectMusic 和 DirectSound 可满足你的所有音乐需求，DirectInput 保证用户的输入请求，对于多媒体播放器，有使用方便的 DirectPlay。

DirectX 包括许多部分，但并没有多少书能将所有这些部分包容在一起或以一种容易使用的风格介绍它们。这就是你要阅读本书的原因。我们将首次以特殊的方式向你全面介绍 DirectX 7，从基础到精深。我们也提供了许多真实的例子，以帮助你有效地使用 DirectX，并丰富你的经验。

即使毫无使用 DirectX 7 的工作经验，也同样可以阅读本书。Microsoft 已经为其最新的版本赋予了许多强大的功能，使程序员和设计师可以比以前更为容易地使用它。

那么就请坐下来，打开一听饮料，感受我们短暂的愉快之旅。只需要 24 个学时，就可以学到全部的 DirectX 7。你将喜欢它的每一分钟，就像我们在写作本书时的感受一样。你可以按照自己的方式制作出极为精彩的 3D 游戏。

学时	标 题	内 容
1	第 1 章 DirectX 组件及其作用	了解组成 DirectX 7 的模块
2	第 2 章 Windows 应用程序中的 DirectDraw	用 DirectDraw 在应用窗口里创建一个图像
3	第 3 章 系统的控制	创建全屏 DirectDraw 应用
4	第 4 章 创建游戏循环	浏览基本的游戏机的构造和高速游戏控制的消息循环
5	第 5 章 动起来: DirectDraw 动画技术	通过把复杂的对象加入场景来扩展边缘滚动条
6	第 6 章 DirectSound: 为你的游戏增加情调和声音效果	学习 DirectSound 的原理
7	第 7 章 应用 DirectSound	学习怎样从 WAV 文件里边装载声音和用 DirectSound 来填充滚动引擎
8	第 8 章 DirectMusic: 交互式音乐	遨游交互音乐的世界, 浏览 Microsoft DirectMusic API
9	第 9 章 应用 DirectMusic	用 DirectMusic Composer 组织一个音乐的例子并学习如何动态地运用
10	第 10 章 3D 概念介绍	学习创建 3D 对象的基本观点
11	第 11 章 生成 3D 图像	浏览建立 Direct3D 的接口以及组建 3D 几何图形
12	第 12 章 创建第一个 D3D 的应用程序	创建 3D 应用, 通过城市场景来导航
13	第 13 章 在布景中加入纹理和 Z-buffer	补充上一学时的内容, 并添加细节
14	第 14 章 通过光照增加真实感	通过添加动态光源, 增添真实性
15	第 15 章 向场景中输入 3D 对象和动画	通过置入模型增添 3D 的强大功能
16	第 16 章 模拟一个复杂的世界: 物理学和对象层次的应用	在复杂 3D 场景中模拟动画和交互效果
17	第 17 章 DirectInput 简介: 获取用户输入	学习 DirectInput 结构
18	第 18 章 用户可以感觉到的: 力反馈	学习如何控制力反馈, 以及使用栈效果, 或者创建自定义效果
19	第 19 章 3D 声音: 从混音到多普勒效应	通过应用 3D 图形拓展 3D 声音知识
20	第 20 章 将你的游戏放入网络: 编写多用户标题	了解多播发行应用软件, 学习发布网络游戏
21	第 21 章 游戏中枢: 创建通道	通过创建通道了解游戏中枢
22	第 22 章 用 DirectShow 添加视频	用 DirectX 学习 Windows 媒体的强大功能
23	第 23 章 用 DirectX Transform 为面带来活力	探索 DirectX 转换功能, 学习 DirectDraw
24	第 24 章 用 DirectAnimation 将媒体集成到网页和应用程序中	用脚本增加 DirectX 网络功能

目 录

前言

第一部分 DirectX 介绍

第 1 章 DirectX 组件及其作用	1
1.1 什么是 DirectX	1
1.2 DirectX 组件	1
1.2.1 DirectX SDK	1
1.2.2 DirectX Media SDK	2
1.3 准备使用 DirectX	2
1.3.1 准备编译器	2
1.3.2 在程序中包含 SDK	3
1.4 COM 简介	3
1.4.1 可重用的接口和兼容性	4
1.4.2 IUnknown 基类	4
1.4.3 查询接口	4
1.4.4 COM 参考计算器	5
1.5 课时小结	5
1.6 专家答疑	5
1.7 课外作业	6
1.7.1 思考题	6
1.7.2 练习题	6

第二部分 从 DirectDraw 开始

第 2 章 Windows 应用程序中的 DirectDraw	7
2.1 视频系统	7
2.2 DirectDraw 组件	8
2.2.1 IDirectDraw7 接口	8
2.2.2 DirectDraw 界面	9
2.2.3 与 DirectDraw 剪贴板的内部保持 一致	10
2.3 在主界面上绘图	11
2.4 第一个 DirectDraw 应用程序	14

2.4.1 建立项目	14
2.4.2 生成窗口框架	14
2.4.3 初始化 DirectDraw 并生成剪贴板	17
2.4.4 生成主界面	18
2.4.5 加载图像	19
2.4.6 把图像传到屏幕上	20
2.4.7 在消息循环中将它绑到一起	20
2.4.8 释放接口	21
2.5 课时小结	22
2.6 专家答疑	22
2.7 课外作业	23
2.7.1 思考题	23
2.7.2 练习题	23
第 3 章 系统的控制	24
3.1 全屏图形	24
3.1.1 得到系统的合作	24
3.1.2 改变屏幕分辨率	25
3.2 翻页	26
3.2.1 创建触发链	26
3.2.2 使用页触发	27
3.3 幻灯片显示：一个简单的界面——触发 应用程序	28
3.4 设置应用程序	28
3.4.1 初始化应用程序	29
3.4.2 清除	31
3.4.3 绘制一张幻灯	31
3.4.4 处理幻灯片导航	33
3.4.5 例程输出	34
3.5 课时小结	34
3.6 专家答疑	34
3.7 课外作业	35
3.7.1 思考题	35
3.7.2 练习题	35
第 4 章 创建游戏循环	36
4.1 游戏循环的概念	36

4.2 写一个更好的消息循环	37	6.1.4 DirectSound 与 Windows 音频	67
4.3 获取平滑的回放	38	6.2 内部 DirectSound	67
4.3.1 在循环中使用计时器	38	6.2.1 DirectSound 对象	67
4.3.2 选择计时器	38	6.2.2 创建一个声音缓冲区	68
4.4 第一个动画：一个活动的背景	39	6.2.3 获取设备能力	69
4.4.1 设置应用程序	40	6.2.4 释放 DirectSound	69
4.4.2 设置初始化	41	6.2.5 设置合作级别	69
4.4.3 通过键盘输入控制动作	42	6.2.6 DirectSoundBuffer 对象	70
4.4.4 清除并退出	43	6.2.7 锁定与解锁一个声音缓冲区	71
4.4.5 创建游戏循环	43	6.2.8 播放与停止一个声音缓冲区	72
4.4.6 表演一个活动的背景	47	6.2.9 获取声音缓冲区的状态	73
4.5 课时小结	48	6.2.10 设置声音缓冲区的音量	73
4.6 专家答疑	49	6.2.11 控制声音缓冲区的颤动	73
4.7 课外作业	49	6.2.12 改变声音缓冲区的频率	74
4.7.1 思考题	49	6.2.13 静态和流声音缓冲区	74
4.7.2 练习题	49	6.3 课时小结	74
第 5 章 动起来：DirectDraw 动画技术	50	6.4 专家答疑	74
5.1 位块传送	50	6.5 课外作业	75
5.1.1 颜色锁结的两个方法	50	6.5.1 思考题	75
5.1.2 选择一种锁结色	51	6.5.2 练习题	75
5.1.3 转换锁结颜色	51	第 7 章 应用 DirectSound	76
5.1.4 在 DirectDraw 中设置锁结色	53	7.1 DirectSound 和游戏	76
5.2 使它看上去像 3D	54	7.2 用波工作	77
5.2.1 Z 轴排序	54	7.2.1 波的高级用法	77
5.2.2 视差：移动物体的深度感知	54	7.2.2 创建 CWave 类	78
5.3 将它们放在一起	55	7.2.3 使用 CWave 类	84
5.4 课时小结	62	7.3 DirectSound 播放声音效果	84
5.5 专家答疑	62	7.4 课时小结	90
5.6 课外作业	63	7.5 专家答疑	90
5.6.1 思考题	63	7.6 课外作业	90
5.6.2 练习题	63	7.6.1 思考题	90
		7.6.2 练习题	90

第三部分 添加音乐和声音

第 6 章 DirectSound：为你的游戏增加情调

和声音效果

6.1 DirectSound 的基础	65
6.1.1 低延迟的音频混合	66
6.1.2 硬件加速	66
6.1.3 3D 声音	66

第四部分 欢迎来到 3D 世界

第 8 章 DirectMusic：交互式音乐

8.1 DirectMusic 的特点	93
8.1.1 Microsoft Synthesizer	94
8.1.2 交互式音乐的时代	95
8.1.3 音乐的动态生成	95

8.2	数字领域的音乐创作	96
8.2.1	音乐结构的快速启蒙	97
8.2.2	用 MIDI 形式存储乐曲	98
8.2.3	多轨音乐合成器	98
8.3	DirectMusic 体系结构	98
8.3.1	IDirectMusicPerformance 接口	98
8.3.2	执行对象的接口	100
8.3.3	乐团的组合	100
8.3.4	音乐模板	101
8.4	课时小结	102
8.5	专家答疑	102
8.6	课外作业	102
8.6.1	思考题	103
8.6.2	练习题	103
第 9 章	应用 DirectMusic	104
9.1	用 DirectMusic 产生器应用程序创作一首简单的谱子	104
9.1.1	选择工具	105
9.1.2	创建节奏	106
9.1.3	制作简单的旋律	107
9.1.4	生成音乐模板	108
9.1.5	保存演奏	110
9.2	创立 DirectMusic	110
9.2.1	初始化演奏接口	110
9.2.2	载入作曲文件	111
9.2.3	确定乐器	112
9.3	DirectMusic 演奏	114
9.3.1	演奏段	114
9.3.2	在线改变节奏和结构	115
9.4	在游戏中加入 DirectMusic	116
9.4.1	创建接口	117
9.4.2	加载执行组件	118
9.4.3	让音乐烘托气氛	119
9.5	课时小结	121
9.6	专家答疑	121
9.7	课外作业	122
9.7.1	思考题	122
9.7.2	练习题	122
第 10 章	3D 概念介绍	124
10.1	3D 处理综述	124

10.1.1	物件定义	124
10.1.2	3D 管道	124
10.2	在 3D 空间中定义位置	126
10.3	在二维环境中创建对象	126
10.4	矩阵: 使世界运动起来	127
10.4.1	“世界”变换	128
10.4.2	视窗变换	132
10.5	课时小结	132
10.6	专家答疑	133
10.7	课外作业	133
第 11 章	生成 3D 图像	134
11.1	D3D 中直接模型的继承关系	134
11.1.1	IDirectD7 接口	134
11.1.2	IDirect3DDevice7 接口	134
11.2	视域锥面	135
11.2.1	在视域中剪切平截面	135
11.2.2	剪切面的前面和后面	135
11.2.3	工程矩阵	136
11.2.4	定义视见区	137
11.3	隐藏面的消除	138
11.4	背面的选取	138
11.5	Z 缓冲	139
11.6	课时小结	140
11.7	专家答疑	140
11.8	课外作业	140
11.8.1	思考题	140
11.8.2	练习题	140

第五部分 输出设备

第 12 章	创建第一个 D3D 的 应用程序	143
12.1	在 D3D 中建立对象	143
12.2	三种网阵格式	144
12.2.1	三角链法	144
12.2.2	三角带链法	144
12.2.3	三角扇形法	145
12.2.4	画一个网状体	145
12.3	标记网状图	146
12.4	开始第一个 Direct3D 图形	147

12.4.1	生成一个简单的 3D 物体	147
12.4.2	物体的处理	150
12.4.3	开始工作	151
12.4.4	球类表面指针	152
12.4.5	初始化操作	153
12.4.6	用 Direct3D 初始化 DirectDraw	155
12.4.7	初始化 Direct3D	156
12.4.8	设置视窗部分	157
12.4.9	创建物体	158
12.4.10	使物体运动	158
12.4.11	作出布景	160
12.4.12	处理用户输入	162
12.4.13	完成涉及项目	164
12.4.14	程序运行	166
12.5	课时小结	167
12.6	专家答疑	167
12.7	课外作业	168
12.7.1	思考题	168
12.7.2	练习题	168
第 13 章	在布景中加入纹理和 Z-buffer	169
13.1	准备一个 Z-buffer 设备	170
13.1.1	选择一个 Z-buffer	170
13.1.2	创建并结合 Z-buffer	171
13.1.3	开启与关闭 Z-buffer	172
13.2	加纹理	173
13.2.1	读入一个纹理	173
13.2.2	准备几何图	177
13.2.3	建立设备与操作	179
13.3	课时小结	181
13.4	专家答疑	181
13.5	课外作业	181
13.5.1	思考题	181
13.5.2	练习题	181
第 14 章	通过光照增加真实感	182
14.1	建立一个光源	182
14.2	光源类型	182
14.2.1	点光源	183
14.2.2	聚光灯光源	183
14.2.3	方向光源	184

14.3	在项目中添加光照	184
14.3.1	创建光源	188
14.3.2	激活光照管道	191
14.3.3	在场景中实现光源的动画	191
14.3.4	删除光源	193
14.4	课时小结	193
14.5	专家答疑	193
14.6	课外作业	194
14.6.1	思考题	194
14.6.2	练习题	194

第六部分 Direct3D 直接模式

第 15 章	向场景中输入 3D 对象和动画	195
15.1	3D 建模软件包简介	195
15.2	Direct3D X 文件格式	195
15.2.1	顶点存储	200
15.2.2	顶点颜色	201
15.2.3	顶点法线	202
15.2.4	纹理映射	203
15.2.5	框架层次	203
15.2.6	动画路径	205
15.3	转换 3D Studio 文件: CONV3DS 实用库	206
15.4	读取一个 X 文件格式	207
15.5	在我们的应用程序中添加传播手段	207
15.5.1	读取模型	207
15.5.2	建立层次	208
15.5.3	对象加入到场景中	209
15.5.4	清除	210
15.6	课时小结	211
15.7	专家答疑	211
15.8	课外作业	211
15.8.1	思考题	212
15.8.2	练习题	212
第 16 章	模拟一个复杂的世界: 物理学和对象层次的应用	213
16.1	反作用 and 影响: 定义真实世界的关系	213

16.1.1 运动动力学	214	18.2 内部 DirectInput 概览	250
16.1.2 前向运动	214	18.3 使用力反馈	252
16.1.3 支点和其他运动特性	215	18.3.1 列举和创建力反馈设备	253
16.1.4 反向运动	216	18.3.2 列举力反馈效果	255
16.1.5 对象层次	216	18.4 创建力反馈效果	256
16.1.6 用矩阵来合并运动	217	18.4.1 运行力反馈效果	259
16.2 对象间的相互作用	219	18.4.2 改变力反馈效果	259
16.2.1 碰撞检测	220	18.5 力反馈效果特例	260
16.2.2 对象的反作用力	221	18.6 课时小结	261
16.3 在我们的应用程序中添加碰撞检测	223	18.7 专家答疑	261
16.3.1 定义对象的范围	224	18.8 课外作业	262
16.3.2 碰撞测试	227	18.8.1 思考题	262
16.3.3 反作用力向量的动画实现	228	18.8.2 练习题	262
16.4 课时小结	228	第 19 章 3D 声音：从混音到多普勒	
16.5 专家答疑	229	效应	263
16.6 课外作业	229	19.1 3D 声音入门	263
16.6.1 思考题	229	19.1.1 声音物理学	263
16.6.2 练习题	229	19.1.2 我们如何感觉声音	264
第 17 章 DirectInput 简介：获取		19.1.3 DirectSound3D 的结构	266
用户输入	231	19.2 DirectSound 3D 的组成	266
17.1 DirectInput 基础	231	19.2.1 DirectSound 3D 缓冲区	266
17.2 内部 DirectInput	232	19.2.2 DirectSound 3D 接收器	270
17.2.1 DirectInput 对象	232	19.3 向应用程序中添加 DirectSound 3D	274
17.2.2 DirectInputDevice 对象	234	19.3.1 创建听者	274
17.3 DirectInput 内部代码	235	19.3.2 载入声音	276
17.3.1 启动 DirectInput	235	19.3.3 创建 3D 缓冲区	276
17.3.2 清除 DirectInput	235	19.3.4 仿真音效	277
17.4 处理键盘输入	236	19.3.5 释放接口	278
17.5 处理鼠标输入	239	19.4 课时小结	279
17.6 处理游戏杆的输入	241	19.5 专家答疑	279
17.7 课时小结	245	19.6 课外作业	279
17.8 专家答疑	246	19.6.1 思考题	279
17.9 课外作业	246	19.6.2 练习题	280
17.9.1 思考题	246		
17.9.2 练习题	247		
第 18 章 用户可以感觉到的：		第七部分 DirectPlay	
力反馈	248		
18.1 认识力反馈	248	第 20 章 将你的游戏放入网络：编写多	
18.1.1 力反馈术语	248	用户标题	281
18.1.2 效果的类型	250	20.1 DirectPlay 结构	281
		20.2 创建一个 DirectPlay 对象	286

20.3 选取一个网络联接	286	22.3 互联网流型视频应用程序	316
20.4 连接一个会话	289	22.4 图表过滤器	319
20.5 与玩家通信	291	22.5 DirectShow 应用程序示例	320
20.5.1 玩家与分组	292	22.5.1 初始化 DirectShow	320
20.5.2 DirectPlay 消息	294	22.5.2 建立过滤器	321
20.6 处理网络等待时间	294	22.5.3 建立文件流	322
20.6.1 确定性/非确定性数据	296	22.5.4 播放影片	324
20.6.2 被担保的消息	296	22.5.5 清除	325
20.6.3 消息管理	297	22.6 课时小结	325
20.7 课时小结	297	22.7 专家答疑	325
20.8 专家答疑	298	22.8 课外作业	326
20.9 课外作业	298	22.8.1 思考题	326
20.9.1 思考题	298	22.8.2 练习题	326
20.9.2 练习题	299	第 23 章 用 DirectX Transform 为面带来	
第 21 章 游戏中枢：创建通道	300	活力	327
21.1 DirectPlay 通道简介	300	23.1 DirectX Transform 的功能	327
21.1.1 将玩家集中到一个场所	301	23.2 通用的 DirectDraw 面：IDXSurface	328
21.1.2 游戏服务器	302	23.2.1 创建 DXSurface	328
21.2 使一个 DirectPlay 应用程序通道化	303	23.2.2 从各种文件格式中读取图形	329
21.3 IDirectPlayDobby3 接口	303	23.2.3 自动颜色转换	330
21.3.1 通道对游戏客户端的支持	304	23.3 特殊效果：DirectX Transform	330
21.3.2 将游戏注册成通道型的	305	23.3.1 创建 Transform Factory	330
21.4 用户、分组和会话	307	23.3.2 用工厂来访问转换	331
21.5 提供交谈服务	308	23.3.3 2D 转换	332
21.6 创建一个会话	310	23.3.4 过程型面	333
21.7 启动一个 DirectPlay 通道应用程序	311	23.3.5 3D 转换	335
21.7.1 使游戏通道化	312	23.4 DirectX Transform 应用程序示例	337
21.7.2 发送和接收通道消息	312	23.4.1 创建 DXSurface 对象	337
21.7.3 清除	313	23.4.2 载入图形	338
21.8 课时小结	313	23.4.3 创建转换	338
21.9 专家答疑	313	23.4.4 实现图形转换动画	339
21.10 课外作业	314	23.4.5 清除	340
21.10.1 思考题	314	23.5 课时小结	340
21.10.2 练习题	314	23.6 专家答疑	341
		23.7 课外作业	341
		23.7.1 思考题	341
		23.7.2 练习题	341
第 22 章 用 DirectShow 添加视频	315	第 24 章 用 DirectAnimation 将媒体集	
22.1 DirectShow 简介	315	成到网页和应用程序中	342
22.2 视频重放能力	316	24.1 DirectAnimation：一个 API，多个	

第八部分 DirectMedia SDK

第 22 章 用 DirectShow 添加视频	315
22.1 DirectShow 简介	315
22.2 视频重放能力	316

第 24 章 用 DirectAnimation 将媒体集	
成到网页和应用程序中	342
24.1 DirectAnimation：一个 API，多个	

用户	342	24.4.6 添加声音	353
24.2 深入面：接口一览	343	24.4.7 实现场景动画	353
24.2.1 DirectAnimation 结构	344	24.4.8 清除	354
24.2.2 DirectAnimation 模型	344	24.4.9 用 DirectDraw 方式完成工作	354
24.3 多功能的编程环境	345	24.5 使用 DirectAnimation 的网页示例	355
24.3.1 在 C++ 中编写 DirectAnimation ...	346	24.6 课时小结	357
24.3.2 用脚本在 Web 上使用		24.7 专家答疑	357
DirectAnimation	346	24.8 课外作业	357
24.4 C++ 的 DirectAnimation 应用程序		24.8.1 思考题	357
示例	347	24.8.2 练习题	358
24.4.1 初始化 DirectAnimation	347	附录	359
24.4.2 载入 2D 图形	351	附录 A 参考答案	359
24.4.3 创建 3D 对象	351	附录 B 用 DirectSetup 为应用程序的发行	
24.4.4 添加某些运动	352	做准备	374
24.4.5 创建照相机和光照	352		

测。参考计数很重要，因为在 COM 运行时要卸载一个并不在当前使用的 COM 对象。当你使用对象，要增加一个计数器；当完成了对象，要去掉这个计数器。当计数器为零，COM 将清除所有的东西。接口检测也很重要，因为 COM 仅在 IUnknown 没用时关联。如果 IUnknown 支持一个特定的接口，IUnknown 不会访问 COM 对象。如果可以，COM 对象将提供一个指向那个接口的指针。如果不可以，将收到一个 NULL 指针（一个错误的返回码）。现在我们进行深入学习。

1.4.1 可重用的接口和兼容性

COM 定义的一个缺点在于它不允许对存在接口进行修改。这意味着当一个新版本的 DirectX 被释放掉，已存在的接口不能被修改。而新的接口必须创建允许新增的功能。

尽管这导致了提供同样功能接口的增殖，COM 与过去版本完全兼容。例如，如果你写了一个应用程序，在 DirectX 3 下使用基本的 IDirectDraw 接口，你能确定如果 DirectX 的开发者正确地做了他们的工作，同样的代码将在 DirectX 7 下面也能操作。

这是因为 DirectX 7 提供的 IDirectDraw 接口与早期版本的是一样的。为了使用由较新版本提供的较新的特征，必须使用新的接口，如 IDirectDraw2 或者 IDirectDraw7。

这不意味着能忽视在新的应用程序中的早期版本的接口。例如，当初初始化 DirectDraw，将收到一个指向旧版本的 IDirectDraw 接口的指针。这个接口被用来指示想使用的更新的接口，这个在以下内容将看到。

1.4.2 IUnknown 基类

所有的 COM 接口是基于 IUnknown 接口的。IUnknown 更像作为一个 C++ 中抽象的基类，提供一套罪小的函数，每个接口必须完成这些功能或允许增加功能。

三个函数在 IUnknown 定义下定义：

函 数	描 述
QueryInterface	用来获取指向从现有接口派生的接口的指针
AddRef	增加接口的参考计数。参见 1.4.4 节
Release	当不再需要时释放接口。如果参考计数为零，减少参考计数并摧毁对象

1.4.3 查询接口

DirectX 提供了函数，可以用来获取指向最基础接口的指针，例如 IDirectDraw。增加的接口，包括一个接口的更晚的迭代，如 IDirectDraw7，能够通过简单地要求一个已经存在的接口来获取一个指针。

这些运用 QueryInterface 方法就可以实现，这一点已经在 IUnknown 基类中进行了定义。要获得一个接口指针，Globally Unique Identifier (GUID) 比较适合 QueryInterface。

新术语

Globally Unique Identifier，或者 GUID，是用来代表 COM 接口的唯一值。例如，假如你已经在 lpDD 中得到了一个 IDirectDraw 接口，还要获得关于 IDirectDraw 7 接口，你就可以

在存在的接口中应用 QueryInterface 功能。如下所示：

程序清单 1-1 查询新接口

```
// query for IDirectDraw7 interface

LPDIRECTDRAW7 lpDD7;
HRESULT ddVal;

if (FAILED(ddVal=lpDD->QueryInterface(IID_IDirectDraw4,
                                     (void **) &lpDD7))) {
    // error retrieving interface, handle error code stored in ddVal
}
```

很幸运，QueryInterface 回车将会显示 S_OK，否则，返回特定接口的错误代码。FAILED() 宏能够决定 COM 接口的成功与失败。在 Visual C++ 中，如果回车 S_OK，宏就会显示 False；如果出错，则显示 True。

1.4.4 COM 参考计算器

COM 的另一有趣特征是，它可以兼容多元素同时使用同一界面，而不必担心对象的寿命和域。实质上，它允许多个主人同时拥有这一对象。

这一功能可以通过参考计算（reference counting）实现。当一接口被创建后，它就会包含一个内部计算器，叫做参考计算器，其原始值是 1。它代表了这一接口的指针数。

当由 QueryInterface() 提供对象接口的指针时，或者利用 DirectX 功能返回接口指针时，参考计算器就会增加 1。当使用 Release() 功能时，参考计算器就会减 1。然后，参考计算器被检测，当它到 0 时，对象被破坏。

参考计算器也可以通过 AddRef() 功能增加，这将在 IUnknown 基类中讲述。当程序中想要获得多个指针备份时，这一点非常有用。例如，当工作线程获取其接口指针时。



一定要使对象的每一个参考有且仅有一个 Release() 调用，不管是通过 QueryInterface() 回到 DirectX，还是通过 AddRef() 增加。在对象被释放后，成熟的 COM 程序员一般会把指针设置为 NULL 以备将来使用。

1.5 课时小结

在本章中，你第一次看到了 DirectX。看到了 DirectX SDK 和 DirectX Media SDK 的概况和性能。学会了怎样为使用 DirectX SDK 建立 Visual C++ 编译器，和访问 DirectX 所要求的 COM 的基础。

1.6 专家答疑

问题：Direct 是怎样使软件与将来的硬件很好的工作的？

解答：当你用 DirectX 工作时，不用直接对硬件进行操作。相反，可以通过 DirectX 中的一

系列标准软件接口来访问硬件。硬件销售者开发驱动程序以便允许 DirectX 操作硬件，并且执行 DirectX 需要的每一个接口。虽然实际的硬件元件可能有所变化，但开发者必须使软件接口保持不变。

问题：为现在的 DirectX 版本编写的软件需要重写以便支持将来的 DirectX 版本吗？

解答：不需要。隐藏的 COM 定义需要将来的 DirectX 版本提供与现在相同的接口。实际上，对于新增的 DirectX 功能，必须提供新的接口。现存的接口不能改动，以保证软件与未来的版本兼容。

1.7 课外作业

课外作业帮你回答可能会出现的问题，评价所学的效果，将所学的知识应用到实践之中。思考题的答案参考附录 A。

1.7.1 思考题

1. COM 是什么？
2. 什么宏可被用来测试 `QueryInterface()` 的结果？
3. 哪个 DirectX 接口支持游戏控制器？
4. 构成所有 COM 对象的基类是什么？

1.7.2 练习题

本章没有习题。

第二部分 从 DirectDraw 开始

第 2 章 Windows 应用程序中的 DirectDraw

本章将初次建立一个 DirectX 程序。在开始编码之前，需要理解在 DirectDraw 的内部结构。本章目标：

- 视频系统是怎样工作的。
- DirectDraw 的组件。
- 在表面画图的过程。
- 创建第一个 DirectDraw 应用程序。

2.1 视频系统

当开发一个使用视频屏幕的应用程序时，实质就是控制 CPU 和视频适配器之间的数据流，这些数据流在屏幕上产生图像，图 2-1 显示了这个过程中所包含的组件。

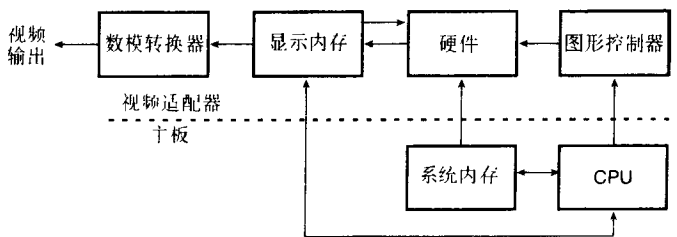


图 2-1 视频组件

可见，在这个模型中存在着两个独立的存储器。第一个是你很熟悉的系统内存，它连接到 CPU 并用于存储应用程序和数据。

第二种内存是显示内存。它是在显示适配器上的存储器，被用于存储显示的图像。这样的存储器称为“帧缓存器”，包含了现在正在显示的图像。

在屏幕上看到的图像由数模转换器 DAC 创建。这个模块负责将存储在帧缓存器中的数字值转换为相应的电压值，创建驱动显示器的模拟信号。

为了获取屏幕上的一幅图像，应用程序将图像数据转换到帧缓存器并在显示器上显示。这一步将大大地影响开发产品的表现。

充分发挥视频硬件的作用是很重要的。尽管数据能被 CPU 逐块转换，很多视频适配器包

含了专门的硬件能有效地将数据块转移到显存中，不管是来自系统内存的还是来自显存的另一个区域都如此。

幸运的是，DirectDraw 能为你处理很多这样的事情，下面将介绍这些功能。

2.2 DirectDraw 组件

DirectDraw 用于访问视频屏幕。当你直接用 DirectDraw 进行 2D 图形操作或者使用 Direct3D 或 DirectMedia 的一个模块时，所有的这些都是通过 DirectDraw 来访问的。这就需要对这个接口从头理解。

DirectDraw 包含以下五个基本组件或者接口

- IDirectDraw7 提供了访问 DirectDraw 的入口并允许创建各种不同的 DirectDraw 对象。
- IDirectDrawClipper 维持一系列剪贴板矩形来控制显示在一个窗口中的 DirectDraw 应用程序。
- IDirectDrawSurface7 对象代表了画一幅图像的一块系统内存或显示内存的区域。
- IDirectDrawPalette 对象被用来存储一个颜色索引阵列并在执行 8 位图像模式时创建用于传输颜色的调色板。
- IDirectDrawVideoPort 提供视频适配器的控制，这种适配器有动态视频功能如电视调谐器和视频采集卡。

2.2.1 IDirectDraw7 接口

所有 DirectDraw 的操作都要通过 IDirectDraw7 接口。先获取一个指向接口的指针，接着是写任何一个使用视频的 DirectX 应用程序的第一步。因为应用程序初始时没有任何能查询的来自 DirectX 的接口，所以可使用由 DirectDraw 启用程序提供的一个帮助函数。

1. DirectDrawCreateEx 的语法

```
HRESULT WINAPI DirectDrawCreateEx(  
    GUID FAR *lpGUID,  
    LPDIRECTDRAW7 FAR *lpDD,  
    REFIID iid,  
    IUnknown FAR *pUnkOuter  
);
```

DirectDrawCreateEx () 函数被用于获取一个指向 IDirectDraw7 界面的指针。如果函数执行成功，返回 DD_OK。

参数：

lpGUID：全局唯一的一个视频显示驱动器的识别符。使用 NULL 来标记缺省的驱动器，或者使用下列值之一：

DDCREATE_EMULATIONONLY：限制驱动器使用任何硬件加速的特性。

DDCREATE_HARDWAREONLY：限制驱动器使用非硬件加速的仿真特性。

lpDD：变量的地址，接收一个指向 IDirectDraw7 接口的指针。

iid：GUID 指定的返回接口的类型，必须设到 IID_IDIRECTDRAW7。