

游戏软件开发专家系列



DirectX 9.0 3D 游戏开发编程基础

(美) Frank D. Luna 著

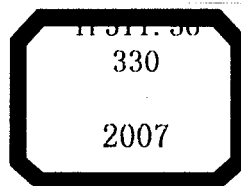
(美) Rod Lopez 技术审校

段菲译



清华大学出版社

游戏软件开发专家系列



DirectX 9.0 3D 游戏开发编程基础

(美)Frank D. Luna 著
(美)Rod Lopez 技术审校
段 菲 译

清华大学出版社

北 京

内 容 简 介

本书主要介绍如何使用 DirectX 9.0 开发交互式 3D 图形程序,重点是游戏开发。全书首先介绍了必要的数学工具,然后讲解了相关的 3D 概念。其他主题几乎涵盖了 DirectX3D 中的所有基本运算,例如图元的绘制、光照、纹理、Alpha 融合、模板,以及如何使用 DirectX3D 实现游戏中所需的技术。介绍顶定点着色器和像素着色器的章节(包含了效果框架和新的高级着色语言的内容)对这些关键运算进行了较为集中的讨论。

本书内容深入浅出,内容广泛,可供从事 3D 游戏程序设计、可视化系统设计或其他图形应用程序开发的开发人员和大专院校学生参考,也极适合各种游戏开发培训机构作为 DirectX3D 编程的培训教程。

DirectX 9.0 3D 游戏开发编程基础

Introduction to 3D Game Programming with DirectX 9.0(ISBN 1-55622-913-5)

Frank D.Luna

Copyright © 2003 by Wordware Publishing, Inc.

Original English Language Edition Copyright © 2003 by Wordware Publishing, Inc.

All Rights Reserved 2320 Los Rios Boulevard Plano, Texas 75074, U.S.A.

本书中文简体版由 Wordware Publishing 授权清华大学出版社出版。

北京市版权局著作权合同登记号 图字:01-2005-0656

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13501256678 13801310933

图书在版编目(CIP)数据

DirectX 9.0 3D 游戏开发编程基础/(美)Frank D. Luna 著;(美)Rod Lopez 技术审校;段菲译. —北京:清华大学出版社,2007.4

(游戏软件开发专家系列)

书名原文: Introduction to 3D Game Programming with DirectX 9.0

ISBN 978-7-302-13025-3

I. D... II. ①F... ②R... ③段... III. ①多媒体—软件工具, DirectX 9.0 3D②游戏—应用程序—程序设计 IV. ①TP311.56②G899

中国版本图书馆 CIP 数据核字(2006)第 049566 号

责任编辑:宣 颖

封面设计:杨玉兰

责任印制:何 芊

出版发行:清华大学出版社 地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编:100084

c-service@tup.tsinghua.edu.cn

社 总 机:010-62770175 邮购热线:010-62786544

投稿咨询:010-62772015 客户服务:010-62776969

印 刷 者:北京市清华园胶印厂

装 订 者:三河市春园印刷有限公司

经 销:全国新华书店

开 本:185×230 印 张:24.75 字 数:535 千字

版 次:2007 年 4 月第 1 版 印 次:2007 年 4 月第 1 次印刷

印 数:1~4000

定 价:46.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:014854-01

译者序

DirectX 是微软公司开发的一套功能丰富的底层 API，其功能涵盖了对 2D/3D 图形加速的支持、对各种输入设备的控制，以及对声音和音乐输出的混音和采样、对多玩家网络游戏的支持、对各种多媒体流模式的控制等。DirectX 设计者的初衷是让 Windows 成为理想的游戏开发平台，如今他们正逐步实现这一理想。实际上，虽然不断遭遇同样出色的竞争对手(如 OpenGL)的挑战，但目前的形势是——DirectX 在游戏开发领域已经无可争议地占据王者地位，而且这种趋势极有可能保持下去。当然，这并不是说 DirectX 能够完全取代其他 API，只是“术业有专攻”，各自有不同的擅长领域罢了。

DirectX 在体系结构上很突出的优点是它使用了两层驱动——HAL 和 REF，那些为硬件所支持的特定功能可由 HAL 来控制(即可以充分地发挥硬件的加速功能)，从而可以获得理想的运行速度。而对于那些不为硬件所支持的功能，借助 REF，程序能够以软件方式来模拟相应的运算或处理过程，从而使得无论硬件配置是否高端，开发人员都可以完整地体验和测试 DirectX 所提供的全部功能。

关于 DirectX 原理及编程的书籍，目前国内还很少，能体现出 DirectX 9.0 版本新特性的书籍更是少之又少。这本《DirectX 9.0 3D 游戏开发编程基础》可以说从一定程度上弥补了国内图书市场的缺憾。这本书专注于讲解 DirectX 与图形绘制及渲染相关的一个核心子集——Direct3D，是 DirectX 3D 图形编程领域的一本经典著作，在国外自出版以来，就一直雄踞此类书的销量榜首。该书有一些很鲜明的特点：

(1) 授人鱼，更授人与渔

本书在注重讲解原理和方法的同时，也注重引导读者如何获取知识，如何利用 DirectX SDK 帮助文档，非常有利于提高读者的学习能力。尤其需要指出的是，作者专门为本书维护了一个网站 www.moon-labs.com，该网站上不但有全书所有例程的源代码，还在此提供了与之相关但由于篇幅所限而略去的重要主题文档。同时，作者设立了一个 DirectX 技术的讨论区(作者的电子邮件地址是 frank@gameinstitute.com)，并时常亲自出面答疑。这些资源对于读者的作用无疑是不可估量的。

(2) 内容全面，由浅入深

本书根据由易到难的顺序合理安排各章，刚入门的读者只需按书中的章节顺序安排学习，即可顺利掌握各章内容。本书虽属导论性质的著作，但是内容的跨度相当大，从基础的数学原理讲起，涵盖了 Direct3D 初始化、绘制流水线、颜色、光照、纹理、融合、模板、字体、网格、地形绘制、粒子系统、拾取技术直到高级着色语言和效果框架，基本上涵盖了 Direct3D 常用的方方面面。书中某些章节的实例虽然算法比较简单，甚至不够实用，但从理解原理的角度来看还是很适宜的，而且这样可以为读者提供更多的思考空间。

(3) 提供了一个清晰的程序框架

本书中所有的例程都用到了作者自编的一个程序框架,层次非常清晰,降低了程序中各模块的耦合性。该框架将所有程序都需要进行的公共操作封装起来,使得读者在学习新知识点时,精力不必分散到与当前主题无关的那些工作中。

(4) 实例丰富、理论与实践并重

本书的每一章都是在讲解原理的同时插入一小段代码片段来形象地进行辅助说明,而且每章或每个大主题的最后,都会附上一个完整的综合性实例,非常有助于读者加深对当前主题的理解。

本书的作者 Frank D. Luna 是位著名的游戏开发专家,他从事交互式图形程序的研究与开发已有 8 年,而且有超过 10 年的 C++ 编程经验。从 DirectX 5.0 起, Luna 就一直在关注和使用 DirectX。

作为译者,本人在翻译过程中力求忠于原作。书中所涉及的所有专业术语,都查阅了大量国内外相关文献,并请教了许多专家学者,力求找到最完美、最准确的译法,并在可能引起歧义和冲突的地方做了适当调整。同时,在专业术语第一次出现或在容易混淆的环境中出现时,都注上了该术语的英文原文,以便读者进行对比理解。为保证正确性,本书所有例程都在 DirectX October 2006 版和 Visual C++ 2005 中进行了调试和验证。另外在翻译过程中,修正了原书中的若干小错误和纰漏,并参照 DirectX Oct 2006 SDK 文档进行了修正。

本书全部章节的翻译都由译者一人完成,其间得到了清华大学出版社的大力支持,特别是文开琪编辑和宣颖编辑,她们投入了大量时间和精力,为该书的译稿提出了许多宝贵的修改意见,在此向她们表示深深的感谢。限于译者的水平,书中的错误和不妥之处在所难免,恳请读者批评指正,本人的邮箱 d3dbook@gmail.com 或 d3dbook@163.com,热切期待着读者朋友的来信!

段 菲

2007-03-04

前言

本书主要讲述如何使用 DirectX 9.0 开发 3D 交互式计算机图形程序，重点是游戏开发。我们将向您讲解 Direct3D 的基本原理，在您掌握了这些原理之后，您将有能力进一步学习并运用更多的高级技术。我们假定您阅读此书之前已经对 DirectX 有大致了解。从开发者的角度看，DirectX 是一套用于开发 Windows 平台多媒体应用程序的 API(application programming interface, 应用程序编程接口)。本书主要关注 DirectX 的一个特定子集——Direct3D。正如其名称所寓意的，Direct3D 主要用于开发 3D 应用程序。

本书分为四部分。第 I 部分介绍了贯穿全书的数字工具。第 II 部分涵盖了基本的 3D 技术，例如光照(lightning)、纹理(texturing)、Alpha 融合(alpha blending)和模板(stenciling)等。第 III 部分主要是关于如何运用 Direct3D 实现一些有趣的技术和应用程序，例如拾取(picking)、地形绘制(terrain rendering)、粒子系统(particle system)、灵活虚拟摄像机(flexible virtual camera)以及加载和绘制 3D 模型(XFiles)。第 IV 部分的主题是顶点(vertex)和像素着色器(pixel shader)，包括效果框架和新的(到 DirectX 9.0 版)高级着色语言。目前，着色器已在 3D 游戏编程中得到了广泛运用，未来局面依然会如此。我们将第 IV 部分的内容全部用于介绍着色器，以使读者能够跟上现代游戏编程发展的步伐。

对于初学者，最好按章节顺序从头到尾阅读本书。本书的章节编排经过了精心设计，保证了难度的循序渐进，同时在复杂度上也不会有跳跃性的突变。一般来说，对于某一特定章节我们都会使用前一章所介绍的技术和概念。这样，在您进入下一章学习新一章内容之前，请务必确保已经熟练掌握了前一章。有经验的读者可以有选择地阅读感兴趣的章节。

最后，您可能非常想知道阅读完本书之后您将能够制作什么样的游戏。我的回答是您最好大致浏览一下本书中例程的类型。根据本书介绍的技术以及您自身的灵感，您会对将要开发的游戏程序有一个大概的构想。

预备知识

本书是一本入门级的教科书。但是，这并不意味着没有编程经验的读者也能够轻松地阅读本书。读者需要已经熟练掌握了代数学、三角学、开发环境(例如 Visual Studio)、C++，以及基本的数据结构(如数组和列表)。熟悉 Windows 编程会有所帮助但不必需；关于 Windows 编程的介绍请参见附录。

所需的开发工具

本书采用 C++ 作为例程的编程语言。DirectX 文档中提到“DirectX 9.0 只支持 Visual C++ 6.0 及后续版本”。因此,要使用 DirectX 9.0 编写 C++ 应用程序,您应有 Visual C++(VC++)6.0、VC++7.0 或 VC++ 2005。



注意 本书的例程都是在 Visual C++ 2005 下编译和调试。大部分章节的例程也可以在 Visual C++ 6.0 下运行,但要注意下列细微差别。在 VC++7.0/VC++2005 中,下列代码合法并能够通过编译,因为变量 cnt 被认为是 for 循环的局部变量。

```
int main()
{
    for(int cnt = 0; cnt < 10; cnt++)
        std::out<<"Hello"<<std::endl;
    for(int cnt = 0; cnt < 10; cnt++)
        std::cout<<"Hello"<<std::endl;
}
return 0;
}
```

但在 VC++ 6.0 中,上述代码不能成功编译。所给出的错误信息为“error C2374: 'cnt': redefinition; multiple initialization”,这是因为在 VC++ 6.0 中,变量 cnt 不被认为是 for 循环的局部变量。因此,在 VC++ 6.0 中调试本书的例程时,您可能需要做些小改动。

推荐的硬件配置

如果希望以可容忍的帧频运行本书的例程,推荐您使用以下硬件配置。如果使用 REF 设备(以软件方式模拟 Direct3D 的功能),所有的例程都可以运行,但是程序也会因此运行十分缓慢。我们将在第 1 章中更多地讨论 REF。

本书第 II 部分的例程相当基本,在如 Riva TNT 及性能相近的低端图形卡上应该能顺利运行。第 III 部分的例程涉及了更复杂的几何运算,并使用了一些较新的特性,如点精灵。要顺利运行这些例程,我们推荐您至少使用 GeForce 2 一级的图形卡。第 IV 部分的例程使用了顶点着色器和像素着色器,所以,要想看到这些例程的实时效果,您需要配备支持着色器的图形卡(如 GeForce 3)。

面向的读者

本书面向的读者主要分为以下三类。

- 希望了解 Direct3D 9.0 编程的中级 C++程序员。
- 使用过不同于 DirectX 的 API(例如 OpenGL)，希望了解 Direct3D 9.0 的 3D 程序员。
- 希望了解 Direct3D 9.0 新特性(包括顶点和像素着色器、高级着色语言以及效果框架)的 Direct3D 高级程序员。

安装 DirectX 9.0

为了编写和执行 DirectX 9.0 程序，计算机中需要安装 DirectX 9.0 运行时(runtime)以及 DirectX 9.0 SDK(Software Development Kit, 软件开发工具包)。注意，安装 DirectX SDK 时，运行时会自动安装。DirectX SDK 可以从 <http://msdn.microsoft.com/directx/sdk/> 获得最新版本的 DirectX 开发包(目前最新版本为 DirectX Oct 2006，即是本书所采用的版本 DirectX October 2006)。安装过程比较简单，您只要按照向导提示进行即可。

开发环境的配置

我们用来编写 DirectX 应用程序的工程类型是 Win32 Application。在 VC++ 6.0 以及更高版本中，您需要指定 DirectX 头文件和库文件所在的路径，以便 VC++ 能够找到这些文件。默认情况下，DirectX 头文件和库文件分别位于路径 C:\Program Files\Microsoft DirectX SDK (Oct 2006)\Include 和 C:\Program Files\DirectX SDK (Oct 2006)\Lib\x86 下。



注意 DirectX 的路径在您的计算机上可能与上述路径有些差异，这要看您在安装过程中是否指定了其他安装路径。

通常情况下，DirectX SDK 会自动将这些路径添加到 VC++ 中。但有时需要您手动添加，方法如下。

在 VC++ 6.0 中选择 Tools | Options | Directories 命令，然后输入 DirectX 头文件和库的路径，如图 1.1 所示。

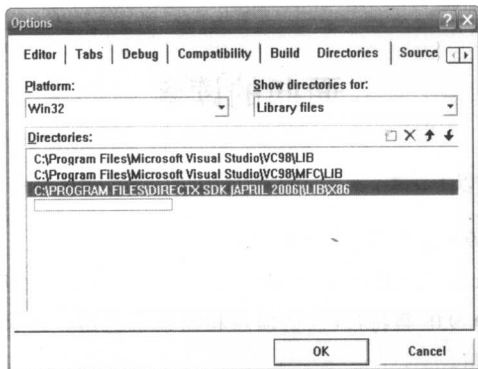


图 1.1 将 DirectX 的 include 和 library 路径添加至 VC++ 6.0

在 VC++ 2005 中, 选择 Tools | Options | Projects and Solutions | VC++ Directories 命令, 然后输入 DirectX 头文件和库的路径, 如图 1.2 所示。

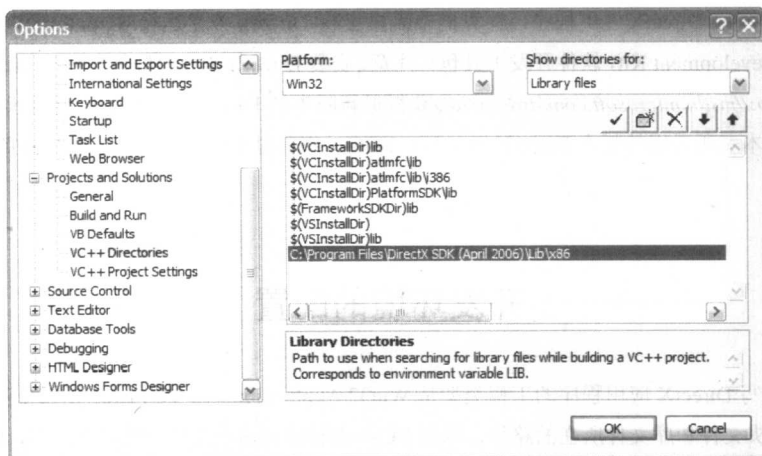


图 1.2 将 DirectX 的 include 和 library 路径添加至 VC++ 2005

接下来, 为了建立(Build)例程, 您需要将库文件 d3d9.lib, d3dx9.lib 以及 winmm.lib 连接到您的工程中。请注意, winmm.lib 并非 DirectX 库文件; 它是 Windows 多媒体库文件, 我们主要使用其定时功能。

在 VC++ 6.0 中, 您可通过选择 Project | Settings | Link 命令来连接您所需要的库文件, 然后输入库名, 如图 1.3 所示。

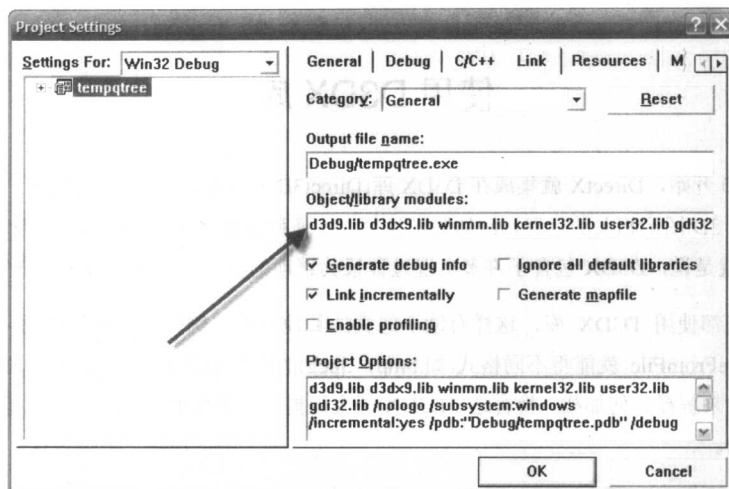


图 1.3 指定需要连接到 VC++ 6.0 中工程的库文件

在 VC++ 7.0 或 VC++ 2005 中，您可以通过选择 Project | Properties | Linker | Input Folder 命令并输入库名来指定所要连接的库，如图 I.4 所示。

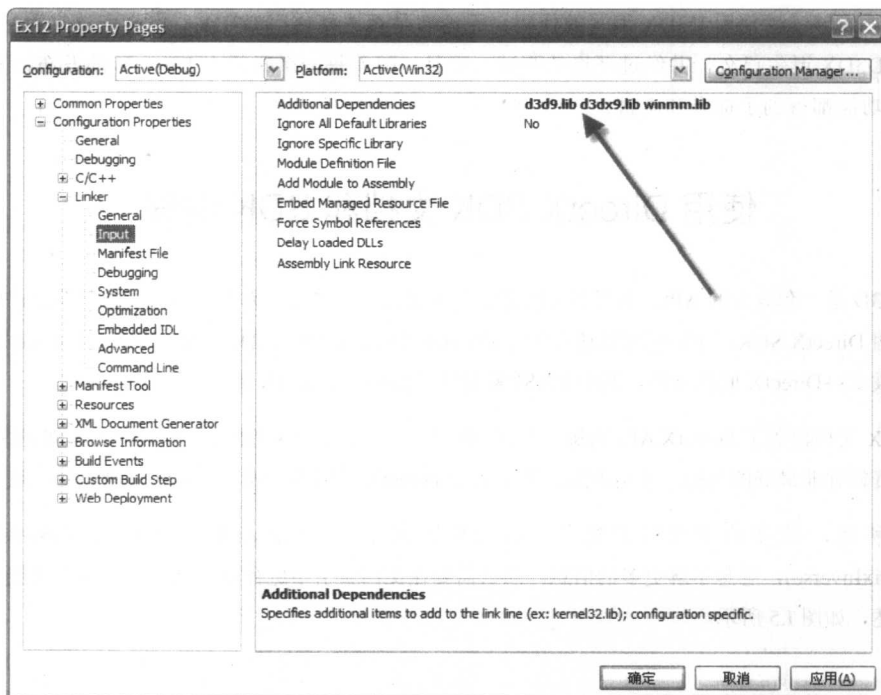


图 1.4 指定需要连接到 VC++ 2005 中工程的库文件

使用 D3DX 库

从 DirectX 7.0 开始, DirectX 就集成在 D3DX 库(Direct3D 扩展库)中。该库提供了一些函数、类和接口,极大地简化了 3D 图形相关的运算,例如数学运算、纹理和图像运算、网格运算以及着色器运算(例如编译和汇编)。也就是说, D3DX 包含了许多实现过程极其繁琐的特性,极大地方便了开发人员。

本书自始至终都使用 D3DX 库,这样有助于您关注其他方面。例如,当我们借助一个 D3DX 函数 `D3DXCreateTextureFromFile` 就能将不同格式(如.bmp, .jpeg)的图像加载到 Direct3D 纹理接口时,我们便不必花很大的篇幅来解释如何加载。换言之, D3DX 极大地提高了我们的效率,使我们能够将精力集中于更多有意义的内容。

使用 D3DX 的其他理由还包括:

- D3DX 通用性强,可用于开发类型广泛的 3D 应用程序。
- D3DX 速度快。
- 其他的开发者也可以使用 D3DX,这样您会更频繁遇到使用 D3DX 编写的程序,所以,为了能够读懂使用 D3DX 编写的代码,无论您是否选择使用 D3DX,您都应该掌握它。
- D3DX 诞生已久,并经过严格的测试。而且,随着每一个新版本 DirectX 的发布,其性能和功能都得到了提升和丰富。

使用 DirectX SDK 文档和 SDK 例程

Direct3D 是一套庞大的 API,本书不可能覆盖其全部内容。所以,为了获得更多外延信息,您必须掌握如何使用 DirectX SDK 文档。您可以通过运行 `DXSDK\Documentation\DirectX9` 路径下的 `directX9_c.chm` 文件来加载 C++DirectX 联机文档,其中 DXSDK 是您计算机中 DirectX 的安装路径。

DirectX 文档覆盖了 DirectX API 的每一部分,所以,它有很好的参考价值。但是因为其叙述深度不够,作为学习资料并非最好的选择。幸运的是,随着每个 DirectX 新版本的发布,它正逐渐得到改进。

如前所述,该文档主要用于参考。假定您看到了一个 DirectX 相关的类型或函数(例如 `D3DXMatrixInverse`),您想了解更多的信息。您只需要在文档索引中进行查找就可获得该对象类型或函数的详细描述,如图 I.5 所示。



图 1.5 C++ SDK 文档浏览器



注意 本书中，我们可能会不时地指导您查询该文档。

该 SDK 文档也包含一些入门教程，您可从文档浏览器左边的目录中找到。这些教程对应于本书第 II 部分的一些主题。建议您在阅读本书的同时学习一下相关的教程，这样您会对一些概念和原理获得不同角度的认识，并接触到其他的例程。

我们将指出随 DirectX SDK 一起安装的 Direct3D 例程。这些 C++ Direct3D 例程位于目录 DXSDK\Samples\C++下。每个例子都阐释了如何实现 Direct3D 的一种特定效果。这些例程对于刚入门的图形程序员来说相当复杂，但当您读完本书再来研究这些例程时，便会倍感轻松。仔细研究这些例程是读完本书之后得以提升的绝佳选择。

代码约定

本书例程代码的约定相当明确。需要特别指出的有两点。首先，所有的成员变量都以下划线为前缀。例如：

```
class C {
public:
    //...define public interface
```

```
private:
    float_x;    //prefix member variables with an underscore.
    Float_y;
    Float_z;
};
```

其次，全局变量和函数名都以大写字母开头，而局部变量和方法名都以小写字母开头。您将发现这种约定对于判断变量或函数的作用域很有帮助。

异常处理

通常情况下，在例程中我们并没有做异常处理，这是因为我们想将您的注意力集中在阐释一个具体概念或技术的更重要的代码上。也就是说，觉得没有异常处理的例程代码能够更清晰地表达某一概念。如果您要在实际的工程中引用例程中的代码，请不要忘记这一点。

可读性

有一点要强调的是，在编写本书中的例程时，我们注重可读性更甚于性能。所以，许多例程的执行效率可能并不是最高的。如果您要在实际工程中引用例程中的代码，请不要忘记这一点，您需要进行一定的改写以提高程序的执行效率。

例程和在线补充材料

本书的网站(www.moon-labs.com)为充分发掘本书的功能发挥了重要作用。该网站提供了本书中所有例程的完整代码。我们建议读者阅读某章时或阅读完毕之后，仔细研究一下相应章节的例程。读者阅读完某一章后，应能自己实现该章的例程并对例程代码进行仔细琢磨。通过参考书中的例程，您若能动手实现它们，将会受益无穷。

除了这些例程，该网站中还包含有公告栏和聊天室。我们鼓励读者对其不理解或需要澄清的问题和概念与其他读者进行交流。因为我们一般都有这样的体会，从不同的视角看待一个概念会加速对该概念的理解。

最后，我们打算在该网站上就由于种种原因没有写入本书的主题增加一些附加例程和教程。如果我们从读者的反馈中发现某些特定概念较难理解时，我们也会上传一些例子和对这类概念分析的文档。

本书的配套文件也可从 www.wordware.com/files/dx9 下载。

致 谢

我要感谢我的技术编辑 Rod Lopez 为本书的审阅投入了大量精力,使得本书的准确性和质量都得到了提升。我也要感谢 Jim Leiterman(*Vector Game Math Processors* 一书的作者,该书已由 Wordware 出版公司出版)和 Hanley Leung(Kush 游戏的开发者),他们共同审阅了本书的部分章节。其次,我要感谢 Adam Hault 和 Gary Simmons 给我的诸多帮助,他们都在 www.gameinstitute.com 从事 BSP/PVS 课程的教学工作。此外,我还想感谢 William Chin 多年前曾给予我的帮助。最后,我要特别感谢 Wordware 出版公司的员工 Jim Hill、Wes Beckwith、Beth Kohler、Heather Hill、Denise McEvoy 以及 Alan McCuller。

目 录

第 I 部分 基础知识

必备的数学知识	3	基本变换	19
3D 空间中的向量	3	平移矩阵	20
向量相等	7	旋转矩阵	21
计算向量的长度	7	比例变换矩阵	23
向量的规范化	8	几何变换的组合	24
向量加法	9	向量变换的一些函数	25
向量减法	9	平面(选读)	26
数乘	10	D3DXPLANE	27
点积	11	点和平面的空间关系	27
叉积	11	平面的创建	28
矩阵	12	平面的规范化	29
矩阵相等、矩阵数乘和矩阵加法	13	平面的变换	29
矩阵乘法	14	平面中到某一点的最近点	30
单位矩阵	15	射线(选读)	31
逆矩阵	15	射线	31
矩阵的转置	16	射线与平面的相交	32
D3DX 矩阵	16	小结	32

第 II 部分 Direct3D 基础

第 1 章 初始化 Direct3D	37	1.3.2 多重采样	41
1.1 Direct3D 概述	37	1.3.3 像素格式	42
1.1.1 REF 设备	38	1.3.4 内存池	42
1.1.2 D3DDEVTYPE	38	1.3.5 交换链和页面置换	43
1.2 COM(组件对象模型)	38	1.3.6 深度缓存	44
1.3 预备知识	39	1.3.7 顶点运算	45
1.3.1 表面	39	1.3.8 设备性能	45

1.4 Direct3D 的初始化.....46	索引缓存的信息.....77
1.4.1 获取接口 IDirect3D9 的指针.....47	3.2 绘制状态.....78
1.4.2 校验硬件顶点运算.....47	3.3 绘制的准备工作.....78
1.4.3 填充 D3DPRESENT_	3.4 使用顶点缓存和索引缓存进行绘制.....79
PARAMETER 结构.....48	3.4.1 IDirect3DDevice9::
1.4.4 创建 IDirect3DDevice9	DrawPrimitive.....79
接口.....50	3.4.2 IDirect3DDevice9::
1.5 例程: Direct3D 的初始化.....51	DrawIndexedPrimitive.....80
1.5.1 d3dUtility.h/cpp.....52	3.4.3 Begin/End Scene.....81
1.5.2 例程框架.....53	3.5 D3DX 几何体.....81
1.5.3 例程: D3D 初始化.....54	3.6 例程: 三角形、立方体、
1.6 小结.....56	茶壶、D3DXCreate*.....83
第 2 章 绘制流水线.....58	3.7 小结.....87
2.1 模型表示.....59	第 4 章 颜色.....89
2.1.1 顶点格式.....60	4.1 颜色表示.....89
2.1.2 三角形单元.....60	4.2 顶点颜色.....92
2.1.3 索引.....61	4.3 着色.....92
2.2 虚拟摄像机.....62	4.4 例程: 具有颜色的三角形.....93
2.3 绘制流水线.....63	4.5 小结.....95
2.3.1 局部坐标系.....64	第 5 章 光照.....97
2.3.2 世界坐标系.....64	5.1 光照的组成.....97
2.3.3 观察坐标系.....65	5.2 材质.....98
2.3.4 背面消隐.....67	5.3 顶点法线.....100
2.3.5 光照.....68	5.4 光源.....102
2.3.6 裁剪.....68	5.5 例程: 光照.....105
2.3.7 投影.....69	5.6 一些附加例程.....108
2.3.8 视口变换.....70	5.7 小结.....108
2.3.9 光栅化.....72	第 6 章 纹理映射.....109
2.4 小结.....72	6.1 纹理坐标.....110
第 3 章 Direct3D 中的绘制.....73	6.2 创建并启用纹理.....111
3.1 顶点缓存与索引缓存.....73	6.3 纹理过滤器.....112
3.1.1 创建顶点缓存和索引缓存.....73	6.4 多级渐进纹理.....113
3.1.2 访问缓存内容.....76	6.4.1 多级渐进纹理过滤器.....114
3.1.3 获取顶点缓存和	6.4.2 使用多级渐进纹理.....114

6.5 寻址模式	114	8.1.2 模板测试	130
6.6 例程: 纹理四边形	116	8.1.3 模板测试的控制	130
6.7 小结	118	8.1.4 模板缓存的更新	132
第 7 章 融合技术	119	8.1.5 模板写掩码	132
7.1 融合方程	119	8.2 例程: 镜面效果	133
7.2 融合因子	121	8.2.1 成像中的数学问题	133
7.3 透明度	122	8.2.2 镜面效果实现概述	134
7.3.1 Alpha 通道	122	8.2.3 代码解析	135
7.3.2 指定 Alpha 来源	123	8.3 例程: Planer Shadows	139
7.4 用 DirectX Texture Tool 创建 Alpha 通道	123	8.3.1 平行光阴影	139
7.5 例程: 透明效果	125	8.3.2 点光源产生的阴影	140
7.6 小结	127	8.3.3 阴影矩阵	141
第 8 章 模板	128	8.3.4 使用模板缓存 防止二次融合	142
8.1 模板缓存的使用	129	8.3.5 代码解析	143
8.1.1 模板缓存格式的查询	129	8.4 小结	144

第Ⅲ部分 Direct3D 的应用

第 9 章 字体	149	10.2 子集和属性缓存	157
9.1 ID3DXFont 接口	149	10.3 绘制	159
9.1.1 创建一个 ID3DXFont 接口对象	149	10.4 网格优化	159
9.1.2 绘制文本	150	10.5 属性表	161
9.1.3 计算每秒绘制的帧数	151	10.6 邻接信息	163
9.2 CD3DFont	152	10.7 克隆	164
9.2.1 创建 CD3DFont 类的实例	152	10.8 创建网格(D3DXCreateMeshFVF)	165
9.2.2 绘制文本	152	10.9 例程: 网格的创建与绘制	167
9.2.3 清理	153	10.10 小结	172
9.3 D3DXCreateText 函数	153	第 11 章 网格 (二)	174
9.4 小结	155	11.1 ID3DXBuffer	174
第 10 章 网格 (一)	156	11.2 XFile	175
10.1 几何信息	156	11.2.1 加载 XFile 文件	175
		11.2.2 XFile 材质	176
		11.2.3 例程: XFile	177