第一图书网, tushu007.com

<<DirectX游戏开发终极指南>>

图书基本信息

书名：<<DirectX游戏开发终极指南>>

13位ISBN编号：9787302172864

10位ISBN编号：7302172862

出版时间：2008-6

出版时间：第1版 (2008年6月1日)

作者：谢里德 (Allen Sherrod)

页数：692

字数：1017000

译者：狄东宁

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：http://www.tushu007.com

前言

3D游戏开发已经成为一个热门的研究领域。
对于那些想要涉足这方面学习、研究的人而言，拥有一本这方面的教材已经迫在眉睫。
如今市面上这类书很多，但在众多书籍之中这本《DirectX游戏开发终极指南》却有它的独到之处，它围绕一个实例由浅入深地将游戏开发这个高深的主题娓娓道来。
本书主要介绍在Microsoft公司的DirectX软件开发包上开发3D游戏的全过程。
作为一本抛砖引玉的游戏编程书籍，本书尽可能避免对高深的游戏编程理论的介绍，这有利于引起初学者的兴趣。
同时整本书结合了一个游戏开发实例，这也可以很大程度地提高读者的学习兴趣。
读者学完本书后，应该对如何在Windows平台上开发3D游戏有一个完整的了解，这对于今后从事游戏开发工作或是深入研究都大有裨益。
本书分为3个主要部分，从头至尾介绍了整个3D游戏的开发。
第一部分重点介绍了使用Direct3D API生成DirectX图形的内容。
第二部分重点介绍了游戏数学、碰撞检测、输入检测和游戏声音等内容。
第三部分重点介绍了模型加载和3D场景中的动画与场景管理。
本书还在附录中介绍了游戏开发的相关书籍和推荐的一些资源。
本书既可作为游戏开发爱好者的入门书籍，也可作为游戏开发人员的培训教材，对那些高级游戏程序员而言，本书也可作为他们的案头参考书。
欢迎各位读者对本书提供反馈意见。
我希望读者能从本书中受益，也希望通过读者的反馈意见来了解自己的不足，以求在今后的译作中更多更切实际地考虑读者的需要。

内容概要

开发计算机游戏充满了挑战性，需要开发人员具备大量的技能，富有奉献和创新精神。
本书向读者介绍了如何将自身的技能与一些基本的C++编程技能结合起来，开发出自己的游戏。
本书涵盖了游戏开发过程和方方面面，书中以一个名为Straned的第一人称射击游戏开发过程为主题展开介绍，首先介绍了DirectX 和Direct3D图形学，然后逐章介绍了游戏引擎和完整游戏的开发过程。
主要内容涉及到基本的图形学技术、最基本的数学知识、碰撞检测和响应、声音回放、场景管理、动画以及模型/角色的加载和绘制。

一旦读者开发出该游戏，就可以轻松地对其功能进行扩展，并定制该游戏，添加一些特有的功能。
如果读者已经具备了基本编程 技能且热爱游戏开发出自己的游戏，那么可以在本书中找到大量所需的素材。

本书适合于想实践游戏开发的学生、游戏编程的初学者以及想要学习游戏开发的C++程序员。

作者简介

Allen Sherrod毕业于DeVry大学，获得了计算机信息系统专业学士学位，是www
. UltimateGameProgramming.com游戏编程网站的创建者，这是一个面向游戏开发初学者和游戏编程爱好者的网站。
Allen为这个广受欢迎的网站定期撰写文章并开发了大量的代码。
在过去9年中，他一直在使

书籍目录

第1章 DirectX导论 1.1 本书概述 1.1.1 编写本书的目的 1.1.2 读者对象 1.1.3 工具和资源
1.2 游戏规划 1.2.1 项目Stranded概述 1.2.2 设计概述 1.2.3 引擎设计概述 1.2.4 渲染
统 1.2.5 输入系统 1.2.6 声音系统 1.2.7 物理系统 1.2.8 动画系统 1.2.9 人工智能
）系统 1.3 DirectX背景 1.3.1 DirectGraphics 1.3.2 DirectInput 1.3.3 DirectPlay 1.3.4
DirectMusic 1.3.5 DirectSound 1.3.6 安装DirectX 9.0 SDK 1.4 手动设置窗口 1.4.1 创建和
示Direct3D窗口 1.4.2 使用Direct3D绘制图元 1.4.3 Direct3D顶点缓存 1.4.4 坐标系 1.5 演
程序 1.5.1 Lines演示程序 1.5.2 Triangle演示程序 1.5.3 Quad演示程序 1.5.4 Ortho Matr
演示程序 1.5.5 Perspective Projection Matrix演示程序 1.5.6 World Matrix演示程序 1.5.7 View
Matrix演示程序 1.5.8 深度测试 1.5.9 模板源文件 1.5.10 使用DirectX框架设置Direct3D 1
总结 第2章 游戏：Stranded 2.1 游戏规划导论 2.2 游戏规划 2.1.1 角色模型 2.1.2 菜单和
2.1.3 环境 2.1.4 游戏剧本 2.3 引擎规划 2.3.1 渲染系统 2.3.2 输入系统 2.3.3
系统 2.3.4 人工智能 2.3.5 数学库 2.4 游戏项目概述 2.4.1 游戏项目第1部分 2.4.2
项目第2部分 2.4.3 游戏项目第3部分 2.4.4 游戏项目第4部分 2.4.5 游戏项目第5部分
2.4.6 游戏项目第6部分 2.4.7 游戏项目第7部分 2.4.8 游戏项目第8部分 2.4.9 游戏项目第9
部分 2.4.10 游戏项目第10部分 2.4.11 游戏项目第11部分 2.4.12 游戏项目第12部分 2.5 游
戏项目第1部分：启动项目 2.5.1 游戏源文件 2.5.2 引擎源文件和头文件 2.5.3
D3DRenderer.cpp 2.6 总结 第3章 Direct3D光照和物体 3.1 Direct3D光照导论 3.1.1 光源 3
反射模型 3.1.3 光照和绘影技术 3.2 使用Direct3D函数创建物体 3.3 在Direct3D中创建光照 3
游戏项目第2部分：增加对硬件光照的支持 3.5 总结 第4章 纹理 4.1 Direct3D中的纹理介绍 4.2 创
建和使用纹理 4.2.1 纹理坐标 4.2.2 Mipmap 4.2.3 纹理质量 4.2.4 Textures演示程序
4.2.5 多纹理贴图 4.2.6 Multitexture贴图演示程序 4.2.7 透明度 4.2.8 Transparency演示程序
4.2.9 立方体贴图纹理 4.3 sprite（子图形） 4.3.1 点状sprite 4.3.2 Point Sprites演示程序
4.4 凸凹贴图 4.5 保存纹理 4.6 幕外渲染 4.7 游戏项目第3部分：增加对纹理的支持 4.8 总结
第5章 Direct3D文本和图形用户界面 5.1 在屏幕上显示文本 5.2 计算帧率 5.3 创建和显示图形用户
界面 5.3.1 状态显示界面（HUD） 5.3.2 GUI演示程序 5.3.3 main源文件 5.4 游戏项目第
部分：添加文本和GUI支持 5.4.1 游戏源文件 5.4.2 游戏项目引擎文件 5.5 总结 第6章 特效
6.1 多采样 6.2 雾 6.3 细节映射 6.4 粒子系统 6.5 游戏项目第5部分：增加特效 6.6 总结
基本脚本系统 7.1 脚本介绍 7.2 属性脚本系统 7.3 命令脚本系统 7.4 令牌流 7.5 其他类型
本系统 7.6 游戏项目第6部分：增加对脚本的支持 7.7 总结 第8章 游戏数学回顾 8.1 游戏数学介
8.2 矢量数学和回顾 8.3 矩阵数学 8.3.1 矩阵复习 8.3.2 Direct3D矩阵 8.4 四元组数学
射线数学 8.6 平面数学 8.6.1 平面复习 8.6.2 Direct3D平面 8.7 三角形和多边形 8.8 物
8.9 游戏项目第7部分：创建数学库 8.10 总结 第9章 碰撞检测 9.1 碰撞介绍 9.2 边界框 9.3
界球 9.4 平面碰撞 9.5 演示程序 9.5.1 边界框碰撞 9.5.2 球碰撞 9.5.3 平面碰撞
戏项目第8部分：添加碰撞检测 9.7 总结 第10章 输入检测和响应 10.1 使用DirectInput 10.2
DirectInput演示程序 1 0.3 游戏项目第9部分：添加输入系统 10.4 总结 第11章 声音 11.1 声音介
11.1.1 DirectSound 11.1.2 DirectMusic 11.2 使用DirectMusic和DirectSound 11.3 声音演示程序
11.3.1 main源文件 11.3.2 演示程序的头文件和源文件 11.4 游戏项目第10部分：添加声音
11.5 总结 第12章 模型加载 12.1 模型加载介绍 12.2 使用X文件 445 12.2.1 X文件格式介绍
12.2.2 Material模板 12.2.3 Mesh模板 12.2.4 MeshMaterialList模板 12.2.5 MeshTextureCoords
模板 12.2.6 加载和渲染X模型 12.2.7 X Model Loading演示程序 12.3 使用OBJ文件 12.3.1
OBJ文件格式介绍 12.3.2 OBJ Model Loading演示程序 12.4 使用UMF文件 12.4.1 UMF文件格式
介绍 12.4.2 UMF模型加载演示程序 12.5 游戏第11部分：加载模型 12.5.1 添加模型加载支持
12.5.2 加载和渲染等级 12.6 总结 第13章 模型动画 13.1 动画介绍 13.2 动画路径 13.2.1
线路径 13.2.2 曲线路径 13.2.3 圆形路径 13.2.4 路线 13.3 骨骼动画 13.4 X模型动画
13.5 游戏项目第12部分：添加对动画的支持功能 13.6 总结 第14章 场景管理 14.1 场景管理介绍

14.2 场景管理技术 14.2.1 状态管理 14.2.2 闭塞物和平截头体选择 14.2.3 八叉树和四叉树 14.2.4 二叉搜索树 14.2.5 可能可见度集合 14.3 总结 第15章 完成游戏引擎设计 15.1 日志系统 15.2 3D摄像机系统 15.3 平截头体选择 15.4 游戏项目第13部分：完成引擎设计 15.5 总结第16章 开发游戏：Stranded 16.1 本章概述 16.2 组织游戏资源 16.3 加载和显示最终的游戏 16.4 摄像机移动和碰撞检测 16.5 游戏元素 16.6 代理角色 16.7 打包 16.8 总结 第17章 结束语 17.1 的思考 17.2 下一步的工作 附录A 推荐的书籍和网站 A.1 推荐的书籍 A.1.1 游戏和图形编程 A.1.2 通用编程 A.1.3 人工智能（AI）和数学 A.2 推荐的网站 附录B C++入门 B.1 C++基础 B.2 动态内存和文件输入/输出 B.3 结构和类 B.4 总结 附录C 关于光盘 C.1 文件夹 C.2 一般系统要求

章节摘录

第1章　DirectX导论　　从某种程度上讲，游戏编程和开发是极具潜力且激动人心的行业。
它发展迅猛、富有挑战性并带给那些为之努力工作并且超越极限的人非常丰厚的回报。
游戏开发需要一批极富创造力的人协同工作，从而开发出新的游戏产品。
这些人包括软件开发人员、作家、音效师、数学家、物理学家、艺术家以及许多其他人员。
每个人面临的职责、应对的挑战和得到的回报也各不相同。
游戏产业自诞生以来，规模已得到长足发展。
现在它已经可以和诸如电影、音乐、电视这样的产业竞争。
这个每年可以带来数百亿美元利润的产业随着时间的推移、技术的强大和效率的提高也在不断地增长。
视频游戏已经存在数十年了，而且还在快速增长。
随着越来越多游戏玩家的成长，游戏产业带来的经济利润就越来越大。
大部分购买游戏的人都是成年人。
这没有什么值得奇怪的，因为大多数孩子没有很多的钱去购买游戏，而等到他们长大成人后，就有钱买游戏了。
　　游戏开发并不总是充满乐趣和令人兴奋。
开发交互式软件是一件艰苦且充满挑战的事情，对此从来都不要掉以轻心。
开发游戏像玩游戏一样有趣，但同时需要做大量的工作和付出很多的心血。
许多人一想到这一点就会因气馁而放弃。
游戏开发中最困难也是最让人有挫折感的部分是学习如何开发游戏的方法。
学习基础知识就会为你今后提高能力和技术水平打下坚实的基础。
如同在任何领域中一样，通常需要耗费几年的时间才能开发出专业级别的游戏，只要有耐心、冲劲和决心，就能做到这一点。
　　本书为读者介绍使用DirectX开发游戏的方法。
本章将介绍DirectX，更确切地说是Direct3D图形知识。
随着本书的深入，读者同样会学到绘图技巧、输入设备检测和响应、声音回放、场景管理、动画、模型／角色的加载和绘制等内容。
　　1.1　本书概述　　本书将使用DirectX软件开发包（SDK）开发一个名为Stranded的第一人称射击游戏。
本书从头至尾将为该游戏编写代码，这样可以给读者一个清晰的概念，即在DirectX基础上开发游戏所需掌握的内容。
某些章节的结尾处有一些名为“游戏项目”的内容。
这样可以将在当前或前面章节中学到的知识运用到本书的游戏项目中。
到学完本书时，读者会在开发自己的第一个3D游戏方面取得进步。
　　本书分为5个部分。
第一部分涉及到DirectX图形学的基础知识，包括Direct3D的安装和使用，DirectX中控制屏幕渲染等内容。
读者具备这些基础知识后，就可以立即开发出一个短小而且包含Direct3D场景的演示程序。
这些基本技能包括绘制基本图形和对象，在几何图形表面显示图像，在屏幕上创建和显示文本以及学会3D场景中的游戏开发方法。
　　本书第二部分涉及到游戏中共用的编程数学知识和碰撞检测。
这部分内容非常重要，因为游戏以及其他交互娱乐平台都涉及了大量的数学知识。
从渲染图形到移动人物等，所有这些都需要数学知识。
使用本书或学习游戏开发时，读者不必具备很高的数学水平，但经验越多，事情做起来就越容易。
　　本书第三部分将介绍输入和声音这部分内容。
在此，将介绍DirectX家族中的两部分内容：Direct Input和Direct Sound。

这些章节同样也很重要，因为能够和软件进行交互（输入）非常重要，这会让产品更像一款真正的游戏。
声音增加了游戏真实感，并通过影响场景中的状态和氛来增强玩家的体验。
本书第四部分讨论了模型加载和动画制作。
这几章集中讨论了从文件中加载复杂模型和在3D中将数据用动画形式逼真地显示出来的内容。
了解游戏数学和DirectX（Direct3D）渲染知识，就可以让人物逼真地在环境中移动。
本书最后一部分将完成整个游戏的开发工作。
此时，读者已经做了大量的工作。
这些章节将“整理”在前面章节中学习过的知识，并最终开发出成品游戏。
尽管从编码角度而言，该游戏非常简单，但它将会让读者终身获取非常宝贵的学习经验。
读者学习完本书后，一定要查看附录A中推荐的书籍和网站。
从中可以获取将来在游戏开发中加深学习而要用到的补充资源和参考资料。
1.1.1 编写本书的目的 本书旨在使用入门级游戏程序员知识，开发一个简单的第一人称射击游戏。
读者通过编写该游戏，将了解到游戏设计和开发的整个过程。
此外，读者同样可以开发一个让自己的朋友和家人留下深刻印象的游戏。
虽然读者在学习本书时，不需要有任何的游戏开发和DirectX使用经验，但是应该有一些编程经验。
本书是一块垫脚石，它将带领读者进入到游戏开发的广阔空间中。
本书并不是一本涉及到游戏开发全部内容的“终极指南”，它只是一个开端。
读者可以将其作为一本参考资料加以收藏，在将来需要时可以在其中查找对自己有用的信息。
1.1.2 读者对象 本书面向的读者是那些希望在Windows操作系统上使用DirectX进行游戏开发的C++程序员。
虽然读者不一定要是C++大师，但掌握这方面的知识对读者大有裨益，例如掌握面向对象编程。
另外，读者不必有任何游戏编程或图形编程经验，就可以享受到游戏开发的乐趣，并从本书获取大量有用的信息。
本书仅涉及面向对象编程和软件开发的一般主题，非常适合那些以此为爱好的程序员、学生以及希望在这个行业中找到工作的人阅读。
如果读者想要学习DirectX，但是又没有任何编程经验，则可以参阅附录A中列出的一些书籍。
这些书籍有助于读者在编程方面的入门。
同样，附录中还推荐了一些该领域的书籍和资源，它们有助于提高读者的技能水平，这些资源已超出了本书的讨论范围，包括编程、软件开发、数学、人工智能、物理学以及高级图形学和引擎编程等。
1.1.3 工具和资源 本书基于DirectX 9.0进行开发工作，所以读者务必要有一套最新版的DirectX 9.0软件开发包（DirectX 9.0 SDK）。
读者可以从Microsoft公司的www.Microsoft.com／DirectX网站上下载该软件开发包。
本书的代码示例同样可以使用Microsoft公司的VisualStudi0.NET 2003集成开发环境编译。
读者也可以使用以前版本的Visual Studi0集成开发环境，但要在该环境中创建自己的项目，以便对代码示例进行编译。
如果读者有最新版本的Visual Studio，就可以直接打开用Visual Studio.NET 2003开发的项目而不会出现任何问题。
除了附录A中提到可以帮助读者提高编程能力的资源外，读者还可以登录到www.UltimateGameProgrammin9.com网站，使用该网站提供的资源。
读者在此可以找到一个提供大量游戏开发知识和文章的社区，这些知识和文章涉及多个不同专题，供读者学习使用，以帮助读者解决学习过程中遇到的问题。
像可以随意下载代码示例一样，读者可以自由参与网站上的各种讨论。
1.2 游戏规划 如前所述，本书旨在开发一款简单的DirectX第一人称射击游戏。
通过本书大部分章节中最后一部分“游戏项目”的渐进学习，读者就可以完成这款游戏的开发工作。

读者通过本书前面四分之三内容的学习，可以编写简单的引擎代码，它可以作为最终游戏的开发基础。
剩下四分之一的内容将编译和完成最终的游戏。
随书光盘提供了本书中所有的开发代码。
1.2.1 项目Stranded概述 游戏Stranded的风格与《雷神之锤》（Quake）和《虚幻》（Unreal）这类游戏很相似。
游戏是在名为Stranded的游戏引擎基础上开发的。
本书通篇都是同时进行游戏引擎和游戏的开发工作。
每章将会解开游戏开发的一部分难题，直到完成一个简单的3D游戏为止。
当读者学完本书时，如果想要将自己的工作提高一个层次，那么书中提供的任何代码都很容易扩展，或是可以毫不费力地更换其中的代码。
可以使用键盘和鼠标或游戏垫（game pad）等输入设备玩游戏。
如果喜欢的话，读者可以添加其他设备，因为DirectInput作为DirectX的输入，可以方便高效地检测所有输入设备。
同样，游戏可以为武器提供3D声效，如同在点击游戏主菜单时出现的环境音乐。
3D声音似乎是从游戏中某个位置发出的声音。
声音文件主要采用.wav格式，可以使用如Reason 3.0这样的声音／音乐开发软件编制。
Stranded在环境和环境物体的重力和碰撞中用到了简单的物理学知识。
由于物理学是一个不可能详细介绍的庞大领域，所以只能对简单的内容加以介绍。
书中同样还介绍了简单的脚本系统，这样初学者可以涉足脚本系统这一领域的学习。
同样，附录A还推荐了可以帮助读者开发更复杂脚本系统的资源，当然也可以使用现有的脚本系统。
在学完本书后，有些程序员可能希望开发出更复杂的脚本系统。
从图形角度而言，游戏非常简单，因为本书目标只是对游戏编程做一个入门级介绍。
如果希望深入开发工作，读者在学完本书后，添加更高级的图形也是不难实现的。
实际上，可以使用本书的源代码创建更高级的图形，一方面由于本书空间所限，另一方面因为这也超出了本书的讨论范围，所以本书删除了这部分内容。
游戏中使用的人工智能（AI）非常简单。
人工智能是一个巨大的研究领域，它能真正让游戏给人留下深刻印象，让人愿意去观看和进行交互。
人工智能使读者可以和游戏自身进行交互，但它并不能模仿最新最好的游戏。
任何学完本书的人都可以在编写代码基础上，继续开发更高级和效率更高的人工智能。
游戏本身很简单。
主角是一个陷落在地球上的外星人。
第一个任务是主角为了返回家乡，要找到一个像飞船一样的物体。
在寻找该物体的过程中，主角会遭遇试图抓他并获取其技术的敌人。
这意味着有两类角色：敌人和外星人。
每一类角色可以共用一种武器。
同样，游戏只有单场景：一座荒岛。
一旦开发完这个基本游戏，读者就可以为其添加更多的武器、角色、场景等。
1.2.2 设计概述 如果读者计划涉足游戏开发领域，那么游戏设计是要面对的一个重要而广泛的领域。
关于这一主题，有很多文章和书籍可以引导读者进入该领域的学习。
然而，本书只是开发一个短小而简单的游戏，因此不需要很多的游戏设计知识就可以完成开发工作。
这并不意味着无需进行游戏设计。
实际上，由于本书关注的是DirectX和游戏编程，因此它采用了一种非常规的方法替代专业的游戏设计。
附录A推荐了许多游戏设计资源。
因为游戏设计本身也可以用一本书来诠释，所以本书像对待其他部分一样，并未用太多的篇幅来讨论

这个主题。
第2章将讨论游戏规划，但目前仅对该主题稍作介绍。
　　在开始真正的游戏开发前，需要编写大量的代码来构建本书的项目游戏。
所有现代游戏都运行在一个称为游戏引擎的平台上。
游戏引擎指的是可以确保在较高层面上运行游戏的代码集。
例如，可以编写负责渲染工作的代码，而不是直接和Direct3D（或OpenGL等）打交道，这样可以构建一个封装在场景后面的渲染引擎，而任何使用该代码的人都只需考虑渲染引擎即可，而不用考虑它要用到的图形用户界面（GUI）。
这样在底层改动代码.时，例如用OpenGL替代Direct3D，那些使用游戏引擎的人会对此毫无察觉。
同样，在学习过程中，本书中的项目需要一些可以使编译、调试和纠错等工作简化的功能。
这包括错误日志记录系统、简单的脚本运行等。
　　1.2.3　引擎设计概述　　如前所述，游戏引擎是一系列高级代码，读者可以以它为基础开发自己的游戏。
现代游戏引擎已经对那些使用它的人隐藏了底层实现的细节和规范。
例如，可以在OpenGL和Direct3D的基础上开发渲染引擎，这样，引擎用户就不需要知道使用的是哪一个渲染引擎，尽管用到了一些底层的东西。
读者可能会说，那样游戏引擎会变得很复杂。
游戏引擎包括：渲染引擎、物理引擎、声音引擎等。
游戏引擎本身只是一个由更小的引擎组成的集合。
游戏引擎或它涵盖的内容并没有一个精确的定义。
对视频游戏而言，它的游戏引擎至少要包含渲染引擎和输入引擎，这是必需的，否则，就不能称其为交互式游戏。
　　由于游戏引擎开发这个主题很复杂，所以本书并不会试图去开发一个很庞杂的游戏引擎。
实际上，本书将在构成DirectX的不同APl基础上开发出一个简短的游戏引擎。
学完本书足以让读者了解引擎，读者不只是可以编写出一个简单的游戏引擎，还可以在这个引擎上继续开发工作或对其改动以满足自己的需求。
对于那些讨论引擎体系结构和开发的全部资源而言，这不是问题。
虽然没有一本书讨论了类似的游戏引擎的开发，但它们都会给读者提供宝贵的提示、想法，并引领读者深入考虑开发代码的方法。
　　本书的游戏引擎包括构建在Direct3D基础上的渲染引擎、构建在Directtnput基础上的输入系统、构建在DirectSound基础上的声音系统、处理碰撞和运动的物理系统、动画系统和一个简单的人工智能系统。
该引擎代码还包括创建错误日志、使用命令和属性脚本的基本脚本能力和图形用户界面（GUI）系统。
GUl系统用于创建主菜单、加载场景等。
　　Stranded游戏引擎中并未涉及网络系统，这是因为网络是个很庞大的主题，它超出了本书为初学者介绍的知识范围。
其他一些系统至少可以做一些简化，而网络系统编写起来并不容易。
当然并非所有的网络系统代码都很高深，或是很庞大，但要了解的内容已经超出了本书的讨论范围。
由于本书不是一本讨论引擎开发的书籍，所以只关注DirectX的不同组成，并将其集成在一起开发出一个简单的游戏。
游戏程序员和游戏引擎程序员完成的是两种不同的工作。
游戏程序员只与开发游戏的游戏引擎打交道，而游戏引擎程序员要开发出其他人用于开发游戏的引擎。

编辑推荐

《游戏开发经典丛书·DirectX游戏开发终极指南》适合于想实践游戏开发的学生、游戏编程的初学者以及想要学习游戏开发的C++程序员。

《游戏开发经典丛书·DirectX游戏开发终极指南》配套光盘中包含了每一章的所有源代码、项目以及演示程序，还包含所有的图形、Microsoft DirectX SDK以及游戏Stranded的最终版。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:http://www.tushu007.com