

- 全部的 Unity 3D 知识：集成开发环境搭建，界面的详细介绍，脚本的编写等
- 完整的物理引擎体系、天空盒、虚拟按钮与摇杆、声音、光源、地形引擎、角色动画以及角色控制器等开发，着色器和着色器语言以及加速度传感器
- 两个经典案例，帮助读者尽快进入实战角色



Unity 3D

游戏开发技术详解与典型案例

吴亚峰 于复兴 编著
百纳科技 审校

独享两大获奖商业案例



源程序

Unity 3D

游戏开发技术详解与典型案例

吴亚峰 于复兴 编著
百纳科技 审校



人民邮电出版社
北京

图书在版编目 (CIP) 数据

Unity 3D 游戏开发技术详解与典型案例 / 吴亚峰,
于复兴编著. — 北京: 人民邮电出版社, 2012. 11
ISBN 978-7-115-29293-3

I. ①U… II. ①吴… ②于… III. ①游戏程序—程序
设计 IV. ①TP311.5

中国版本图书馆CIP数据核字 (2012) 第207494号

内 容 提 要

本书对 Unity 3D 集成开发环境界面、脚本的编写和众多高级特效的实现进行了详细的介绍, 内容深入浅出, 是一本适合不同需求、不同开发水平读者的技术宝典。

全书共分 10 章。第 1 章主要介绍了 Unity 3D 的诞生、特点、开发环境的搭建以及运行机制; 第 2 章对 Unity 3D 集成开发环境进行了详细的介绍; 第 3 章介绍了 Unity 3D 中脚本的编写; 第 4 章主要对 Unity 3D 开发过程中经常使用的组件及对象进行了详细的介绍; 第 5 章介绍了 Unity 开发平台的完整的物理引擎体系; 第 6 章介绍了天空盒、虚拟按钮与摇杆、声音、光源、地形引擎、角色动画, 以及角色控制器等开发常用的技术; 第 7 章介绍了 Unity 3D 中的着色器和着色器语言——ShaderLab; 第 8 章介绍了 PlayerPrefs 类、Network Class (网络类), 以及加速度传感器的使用; 第 9 章介绍了完整的大型 3D 游戏案例——3D 保龄球; 第 10 章介绍了完整的大型 3D 游戏案例——火力篮球。同时为了便于读者的学习, 本书附赠的光盘中包含了书中所有案例的完整源代码, 最大限度地帮助读者快速掌握开发技术。

本书适合从事游戏开发的编程人员学习。

Unity 3D 游戏开发技术详解与典型案例

◆ 编 著 吴亚峰 于复兴

审 校 百纳科技

责任编辑 张 涛

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京鑫正大印刷有限公司印刷

◆ 开本: 800×1000 1/16

印张: 27.75

彩插: 2

字数: 834 千字

2012 年 11 月第 1 版

印数: 1—4 000 册

2012 年 11 月北京第 1 次印刷

ISBN 978-7-115-29293-3

定价: 69.00 元 (附光盘)

读者服务热线: (010)67132692 印装质量热线: (010)67129223

反盗版热线: (010)67171154

广告经营许可证: 京崇工商广字第 0021 号

前言

为什么要写这样的一本书

近几年，Android 平台游戏、iPhone 平台游戏以及 Web 的网页游戏发展迅猛，已然成为带动游戏发展的新生力量。遗憾的是，目前除了少数的成功作品外，大部分的游戏都属宣传攻势大于内容品质的平庸之作。面对这种局面，3D 游戏成为独辟蹊径的一种选择，而为 3D 游戏研发提供强大技术支持的 Unity 3D 引擎，以其创造高质量的 3D 游戏和真实视觉效果的核心技术，为开发 3D 游戏提供了强大的源动力。

Unity 3D 是由 Unity Technologies 开发的一个轻松创建三维视频游戏、建筑可视化、实时三维动画等类型互动内容的多平台的综合型游戏开发工具，是一个全面整合的专业游戏引擎。

本书通过对 Unity 3D 集成开发环境的搭建，集成开发环境各个界面的详细介绍，再到脚本的编写、开发过程经常应用的技术和对象的讲解，以及最后用两个经典案例进行实战介绍，给读者以由浅入深、循序渐进的学习过程，相信每一位读者都会通过本书得到意想不到的收获。

未来几年内必定是 Unity 3D 大行其道的时代，因其开发群体的迅速扩大，Web player 装机率的快速上升，使 Unity 3D 迅速爆发的时机已经到了。在此引用业内知名人士的一句话：“不要再对所谓的 Flash 3D 抱有什么希望，也不要再去花心思学习那些杂七杂八的 Flash 3D 插件，赶紧学习 Unity 3D 才是正经。”

经过半年多见缝插针式的奋战，本书终于交稿了。回顾写书的这半年多时间，不禁为自己能最终完成这个耗时费力的“大制作”而感到欣慰。同时也为自己能将从事游戏开发近 10 年来积累的宝贵经验以及编程感悟分享给正在开发阵线上埋头苦干的广大编程人员而感到高兴。

本书特点

1. 内容丰富，由浅入深

本书组织上本着“起点低，终点高”的原则，内容覆盖了从学习 Unity 3D 必知必会的基础知识，到基于着色器语言所实现的高级特效，最后还给出了两个完整的大型 3D 游戏案例。这样的内容组织使得懵懂的菜鸟可以一步一步成长为 3D 开发的达人，符合绝大部分想学习 3D 开发的学生与技术人员，以及正在学习 3D 开发人员的需求。

2. 结构清晰，讲解到位

本书中配合每个需要讲解的知识点都给出了丰富的插图与完整的案例，使得初学者易于上手，

有一定基础的读者便于深入。书中所有的案例均是根据笔者多年的开发心得进行设计的，结构清晰明朗，便于读者进行学习与参考。同时书中还给出了很多笔者多年来积累的编程技巧以及心得，具有很高的参考价值。

3. 实用的光盘内容

为了便于读者的学习，本书附赠的光盘中包含了书中所有案例的完整源代码，最大限度地帮助读者掌握开发技术。

内容导读

本书总共分为 10 章，内容按照必知必会的基础知识、基于 Unity 集成开发环境及真实大型游戏案例的顺序进行详细的讲解。

章 名	主 要 内 容
第 1 章 Unity 3D 基础以及开发环境的搭建	简要介绍了 Unity 3D 的诞生、特点、开发环境的搭建以及其运行机制
第 2 章 Unity 集成开发环境详解	主要对 Unity 集成开发环境进行详细的介绍
第 3 章 Unity 3D 的脚本概述	介绍 Unity 中脚本的编写，主要讲解的是特定于 Unity 的 JavaScript 脚本编写的语法和技巧
第 4 章 Unity 3D 的常用组件及对象	主要对 Unity 开发过程中经常使用的组件及对象进行详细的介绍
第 5 章 物理引擎	介绍 Unity 开发平台的完整的物理引擎体系，包括刚体、碰撞器、粒子系统以及关节等知识
第 6 章 3D 游戏开发的常用技术	介绍了天空盒、虚拟按钮与摇杆、声音、光源、地形引擎、角色动画，以及角色控制器等开发常用的技术
第 7 章 着色器——Shaders	介绍了 Unity 中的着色器和着色器语言——ShaderLab，为各种高级特效的开发打下良好的基础
第 8 章 杂项	介绍了 PlayerPrefs 类，Network Class（网络类）以及加速度传感器的使用
第 9 章 综合案例——3D 保龄球	详细介绍了 3D 保龄球项目的开发过程及用到的各种相关技术
第 10 章 综合案例——火力篮球	详细介绍了火力篮球项目的开发过程及用到的各种相关技术

本书内容丰富，从基本知识到高级特效，从简单的应用程序到完整的 3D 游戏案例，适合不同需求、不同水平层次的各类读者。

- 初学 Unity 3D 应用开发的读者

本书包括在 Android 平台下进行 3D 应用开发各方面的知识，内容由浅入深，配合详细的案例。非常适合初学者循序渐进地学习，最终成为 3D 游戏应用开发达人。

- 有一定 3D 开发基础读者，可以进一步深入学习 Unity 3D 高级开发技术

本书不仅包括了 Unity 3D 开发的基础知识，同时也包括基于着色器语言、关节、动画等技术所实现的高级特效，以及 Unity 3D 强大的物理引擎与完整的游戏案例，有利于有一定基础的开发人员进一步提高开发水平与能力。

- 其他平台的 3D 开发人员

由于 Unity 3D 是跨平台的，可以开发基于各个平台的项目，因此适合其他各种平台的 3D 开发人员。

本书作者

吴亚峰，毕业于北京邮电大学，后留学澳大利亚卧龙岗大学取得硕士学位。1998 年开始从事 Java 应用的开发，有 10 多年的 Java 开发与培训经验。主要的研究方向为 OpenGL ES、手机游戏、Java EE 以及搜索引擎。同时为手机游戏、Java EE 独立软件开发工程师，并兼任百纳科技 Java 培训中心首席培训师。近 10 年来为多家著名企业培养了上千名高级软件开发人员，曾编写过《菜鸟成长之路——Java 程序员职场全攻略》、《Android 2.0 游戏开发实战宝典》、《Android 3D 游戏开发技术详解与典型案例》、《Android 应用案例开发大全》、《Android 游戏开发大全》、《Android 3D 游戏开发技术宝典——OpenGL ES 2.0》等多本畅销技术书籍。2008 年年初开始关注 Android 平台下的 3D 应用开发，并开发出一系列优秀的 Android 应用程序与 3D 游戏。

于复兴，北京科技大学硕士，从业于计算机软件领域 10 年，在软件开发和计算机教学方面有着丰富的经验。工作期间曾主持科研项目“P2P 流量可视化检测系统研究与实现”，主研省、市级项目多项，同时为多家企事业单位设计开发了管理信息系统，并在各种科技刊物上发表多篇相关论文。2008 年开始关注 Android 平台下的应用开发，参与开发了多款手机 3D 游戏应用。

本书在编写过程中得到了唐山百纳科技有限公司 Java 培训中心的大力支持，同时刘文洲、陈伟、赵鑫磊、谷新蕾、章雅卓、付鹏、白冰、郑培阳、郭小月、李雪晴、李俊熠、宋盼盼以及作者的家人为本书的编写提供了很多帮助，在此表示衷心的感谢！光盘中包含所有源程序。

由于编者的水平和学识有限，且书中涉及的知识较多，难免有错误疏漏之处，敬请广大读者批评指正，并提出宝贵意见，反馈邮箱 javase6_guide@yahoo.cn。编辑联系邮箱为 zhangtao@ptpress.com.cn。

编者

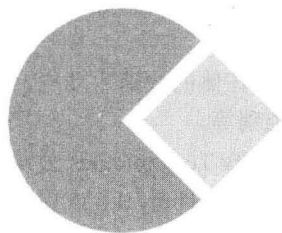
目 录

第 1 章	Unity 3D 基础以及开发环境的搭建	1
1.1	Unity 3D 基础知识概览	1
1.1.1	初识 Unity 3D	1
1.1.2	Unity 的诞生及发展	1
1.1.3	Unity 3D 广阔的市场前景	2
1.1.4	独具特色的 Unity 3D	2
1.2	开发环境的搭建	7
1.2.1	Unity 集成开发环境的安装	7
1.2.2	目标平台的 SDK 与 Unity 3D 的集成	11
1.3	第一个 Unity 3D 程序	16
1.4	本章小结	23
第 2 章	Unity 集成开发环境详解	24
2.1	Unity 集成开发环境的整体布局	24
2.1.1	Unity 集成开发环境的整体布局	24
2.1.2	标题栏	24
2.1.3	菜单栏	25
2.1.4	工具栏	26
2.1.5	场景设计面板	26
2.1.6	游戏预览面板	31
2.1.7	游戏组成对象列表	32
2.1.8	项目资源列表	34
2.1.9	属性查看器	35
2.1.10	状态栏与控制台	36
2.1.11	动画视图	37
2.2	菜单栏	37
2.2.1	文件 (File)	37
2.2.2	编辑 (Edit)	40
2.2.3	资源 (Assets)	46
2.2.4	游戏对象 (GameObject)	56
2.2.5	组件 (Component)	65
2.2.6	地形 (Terrain)	68
2.2.7	窗口 (Window)	70
2.2.8	帮助 (Help)	75
2.3	本章小结	79
第 3 章	Unity 3D 的脚本概述	80
3.1	Unity 脚本概述	80
3.2	Unity 中专用 JavaScript 与通用 JavaScript 的区别	80
3.2.1	在运行过程中性能的差异	80
3.2.2	在语法上的区别	81
3.3	Unity 的基本语法	83
3.3.1	常用操作	83
3.3.2	记录时间	84
3.3.3	访问其他组件	85
3.3.4	访问其他游戏对象	86
3.3.5	向量	89
3.3.6	成员变量和全局变量	89
3.3.7	实例化	90
3.3.8	协同程序和中断	91
3.3.9	一些重要的类	92
3.3.10	性能优化	95
3.3.11	脚本编译	97
3.3.12	泛化方法	98
3.4	一个简单的案例	98
3.4.1	球转场景实现	98
3.4.2	球走场景实现	104
3.5	本章小结	108

第4章 Unity 3D 的常用组件及对象	109
4.1 图形用户界面组件	109
4.1.1 图形用户界面组件的 变量	109
4.1.2 图形用户界面的具体 组件	118
4.1.3 图形用户界面组件 综合案例	150
4.2 预制件 prefab 资源的应用	154
4.2.1 预制件 prefab 资源的 创建	155
4.2.2 通过 prefab 资源进而 实例化对象	156
4.3 常用的输入对象	158
4.3.1 Touch 输入对象	158
4.3.2 Input 输入对象	159
4.4 销毁相关方法	168
4.4.1 Object.Destroy 方法	169
4.4.2 Network.Destroy 方法	169
4.4.3 EditorWindow.OnDestroy 方法	170
4.4.4 MonoBehaviour.OnDestroy 方法	170
4.4.5 ScriptableObject.OnDestroy 方法	171
4.4.6 Object.DestroyImmediate 方法	171
4.5 本章小结	171
第5章 物理引擎	172
5.1 刚体	172
5.1.1 刚体特性	172
5.1.2 物理管理器	176
5.1.3 碰撞与休眠	180
5.2 碰撞器	181
5.2.1 碰撞器的添加	182
5.2.2 碰撞过滤	186
5.2.3 物理材质	191
5.3 粒子系统	193
5.3.1 粒子系统的创建	194
5.3.2 粒子系统的特性	195
5.3.3 粒子系统的组合使用	198
5.4 关节	201
5.4.1 铰链关节	201
5.4.2 固定关节	206
5.4.3 弹簧关节	207
5.4.4 角色关节	209
5.4.5 可配置关节	211
5.4.6 关节综合案例—— 机械手	216
5.5 游戏实例——摧毁立方体	223
5.6 本章小结	231
第6章 3D 游戏开发的常用技术	232
6.1 天空盒的应用	232
6.1.1 Unity 内置的天空盒 资源	233
6.1.2 开发实际需要的天空盒	235
6.2 虚拟按钮与摇杆使用	237
6.2.1 Unity 内置的虚拟按钮与 摇杆资源	237
6.2.2 虚拟按钮与摇杆使用的 案例	238
6.3 声音	241
6.3.1 音频管理器 (Audio Manager)	242
6.3.2 音频监听器 (Audio Listener)	242
6.3.3 音频源 (Audio Source)	243
6.3.4 音频效果 (Audio Effects)	245
6.3.5 音频混响区 (Audio Reverb Zones)	249
6.3.6 简单的声音控制案例	250
6.4 光源	252
6.4.1 点光源 (Point light)	252
6.4.2 平行光光源 (Directional light)	254

6.4.3	聚光灯光源 (Spotlight)	255	7.4	表面着色器和顶点片元着色器	294
6.4.4	区域光光源 (Area Light)	257	7.4.1	表面着色器	294
6.4.5	光照过滤	257	7.4.2	顶点片元着色器	300
6.5	3D 拾取技术	259	7.5	本章小结	303
6.6	地形引擎	262	第 8 章	杂项	304
6.6.1	创建地形	262	8.1	PlayerPrefs 类	304
6.6.2	编辑地形	264	8.2	网络—Network	306
6.6.3	添加树木和草	268	8.2.1	消息发送—— Messages Sent	306
6.6.4	添加水	271	8.2.2	类变量—— Class Variables	310
6.6.5	地形参数	272	8.2.3	类方法—— Class Functions	317
6.7	角色动画	272	8.3	加速度传感器	326
6.7.1	导入角色动画	273	8.4	本章小结	329
6.7.2	动画脚本	274	第 9 章	综合案例——3D 保龄球	330
6.8	角色控制器	275	9.1	游戏背景以及功能概述	330
6.8.1	添加角色控制器	275	9.1.1	背景概述	330
6.8.2	角色控制器的使用	276	9.1.2	功能简介	330
6.9	本章小结	276	9.2	游戏的策划及准备工作	332
第 7 章	着色器—Shaders	277	9.2.1	游戏的策划	332
7.1	着色器概述	277	9.2.2	使用 Unity 开发游戏前的 准备工作	333
7.2	ShaderLab 语法基础	278	9.3	游戏的架构	336
7.2.1	Shader	278	9.3.1	各个场景的简要介绍	336
7.2.2	Properties	278	9.3.2	游戏架构简介	337
7.2.3	SubShader	279	9.4	主菜单界面	338
7.2.4	SubShader Tags	280	9.5	游戏界面	346
7.2.5	Pass	281	9.5.1	场景搭建	346
7.2.6	Fallback	283	9.5.2	保龄球瓶及其脚本	350
7.2.7	Category	283	9.5.3	初始化与得分计算	354
7.3	通道渲染命令 (ShaderLab)	283	9.5.4	瓶子碰撞体的组装	361
7.3.1	颜色、材质和光照	283	9.5.5	得分的绘制, 储存和 返回键的开发	365
7.3.2	剪裁和深度测试	285	9.5.6	瓶子下降和下降台控制	368
7.3.3	纹理处理	287	9.6	设置界面	370
7.3.4	雾	289	9.7	统计界面	373
7.3.5	Alpha 测试	289	9.8	帮助界面	379
7.3.6	混合	291			
7.3.7	通道标签	293			
7.3.8	绑定通道	293			

9.9	游戏结束界面	382	10.5.1	场景搭建	399
9.10	游戏的优化与改进	385	10.5.2	篮球及其脚本	406
第 10 章	综合案例——火力篮球	386	10.5.3	计分触发器开发	410
10.1	游戏背景以及功能概述	386	10.5.4	主摄像机设置以及 脚本开发	414
10.1.1	游戏背景概述	386	10.6	设置界面	416
10.1.2	功能简介	386	10.7	关于界面	418
10.2	游戏的策划及准备工作	388	10.8	帮助界面	420
10.2.1	游戏的策划	388	10.8.1	场景搭建	420
10.2.2	使用 Unity 开发游戏前的 准备工作	388	10.8.2	篮球脚本开发	421
10.3	游戏的架构	391	10.8.3	计分触发器的脚本开发	423
10.3.1	各个场景的简要介绍	391	10.8.4	主摄像机脚本开发	425
10.3.2	游戏架构简介	392	10.9	记录界面	427
10.4	主菜单界面	393	10.10	游戏结束界面	433
10.5	游戏界面	398	10.11	游戏的优化与改进	434



第1章 Unity 3D 基础以及开发环境的搭建

本章主要向读者介绍 Unity 3D 的相关知识以及 Unity 集成开发环境的搭建，通过本章的学习，读者会对 Unity 3D 有一个大致的了解。

1.1 Unity 3D 基础知识概览

本节主要向读者介绍 Unity 3D 的相关知识，主要内容包括 Unity 3D 的简介、Unity 3D 的发展和 Unity 3D 的特点等。通过本节的学习，读者将对 Unity 3D 有一个基本的认识。

1.1.1 初识 Unity 3D

Unity 3D 是由 Unity Technologies 开发的一个轻松创建三维视频游戏、建筑可视化、实时三维动画等互动内容的、多平台的综合型游戏开发工具，是一个全面整合的专业游戏引擎。

Unity 类似于 Director、Blender game engine、Virtools 或 Torque Game Builder 等利用交互的图形化开发环境为首要方式的软件。其编辑器运行在 Windows 和 Mac OS X 下，可发布游戏至 Windows、Mac、Wii、iPhone 和 Android 平台，也可以利用 Unity web player 插件发布网页游戏，支持 Mac 和 Windows 的网页浏览，并且 Unity 的网页播放器也被 Mac widgets 所支持。

1.1.2 Unity 的诞生及发展

通过前面小节的学习，相信读者对 Unity 有了一个简单的认识。而本节为了让读者对 Unity 有更进一步的了解，将为读者介绍 Unity 的发展史。

- 2005 年 6 月，Unity1.0 发布。Unity1.0 是一个轻量级、可扩展的依赖注入容器，有助于构建松散耦合的系统。它支持构造子注入（Constructor Injection）、属性/设值方法注入（Property/Setter Injection）和方法调用注入（Method Call Injection）。

- 2009 年 3 月，Unity 2.5 加入了对 Windows 的支持。Unity 发展到 2.5 完全支持 Windows Vista 与 Windows XP 的全部功能和互操作性，而且 Mac OS X 中的 Unity 编辑器也已经重建，在外观和功能上都相互统一。Unity 2.5 的优点就是 Unity 3D 可以在任一平台建立任何游戏，实现了真正的跨平台。

- 2009 年 10 月，Unity 2.6 独立版开始免费。Unity 2.6 支持了许多的外部版本控制系统，例如 Subversion、Perforce、Bazaar，或是其他的 VCS 系统等。除此之外，Unity 2.6 与 Visual Studio 完

整的一体化也增加了 Unity 自动同步 Visual Studio 项目的源代码, 实现所有脚本的解决方案和智能配置。

- 2010 年 9 月, Unity 3.0 支持多平台。新增加的功能有: 方便编辑桌面左侧的快速启动栏、增加支持 Ubuntu 12.04、更改桌面主题和在 dash 中隐藏“可下载的软件”类别等。

- 2012 年 2 月, Unity Technologies 发布 Unity 3.5。纵观其发展历程, Unity Technologies 公司一直在快速强化 Unity, Unity 3.5 版提供了大量的新增功能和改进功能。所有使用 Unity 3.0 或更高版本的用户均可免费升级到 Unity 3.5。

1.1.3 Unity 3D 广阔的市场前景

近几年来, Android 平台游戏、iPhone 平台游戏以及 Web 的网页游戏发展迅猛, 已然成为带动游戏发展的新生力量。遗憾的是目前除了少数的作品成功外, 大部分的游戏都属宣传攻势大于内容品质的平庸之作。面对这种局面, 3D 游戏成为独辟蹊径的一种选择, 而为 3D 游戏研发提供强大技术支持的 Unity 3D 引擎, 以其创造高质量的 3D 游戏和真实视觉效果的核心技术, 为开发 3D 游戏提供了强大的源动力。

Unity 3D 游戏引擎技术研讨会最早于 2011 年 5 月在韩国举行。据悉, 现在 10 种以上的新引擎开发, 都是采用了 Unity 3D 游戏引擎技术。现已有部分开发商利用 China Joy 展会的契机, 展示了该引擎的运行效果, 目前已有不少厂商与开发商签订了提前预定引擎的协议。

提示

Unity 引擎可以帮助开发人员制作出炫丽的 3D 效果, 并实时生成查看, 目前已推出了对应 iPhone、iPad、PC、MAC、Android、Wii、PS3、XB360 等平台的版本, 促进了游戏跨平台的应用。读者要做的, 只是在编辑器中选择使用哪一个平台来预览游戏作品。

未来几年内必定是 Unity 3D 大行其道的时代, 因其开发群体的迅速扩大, web player 装机率的快速上升, 使 Unity 3D 迅速爆发的时机已经到了。在此引用业内知名人士的一句话: “不要再对所谓的 Flash 3D 抱有什么希望, 也不要再去花心思学习那些杂七杂八的 Flash 3D 插件, 赶紧学习 Unity 3D 才是正经。”

1.1.4 独具特色的 Unity 3D

通过前面两个小节的学习, 相信读者对 Unity 3D 有了一个基本的认识, 本小节将为读者介绍 Unity 3D 的特点, 帮助读者进一步学习 Unity 3D。

1. Unity 3D 本身所具有的特点

- 综合编辑

Unity 简单的用户界面是层级式的综合开发环境, 具备视觉化编辑、详细的属性编辑器和动态的游戏预览特性。由于其强大的综合编辑特性, 因此, Unity 也被用来快速地制作游戏或者开发游戏原型, 如图 1-1 所示。



▲图 1-1 综合编辑

- 图形引擎

Unity 的图形引擎使用的是 Direct3D (Windows)、OpenGL (Mac, Windows) 和自有的 APIs (Wii)。可以支持 Bump mapping、Reflection mapping、Parallax mapping、Screen Space Ambient Occlusion、动态阴影所使用的 Shadow Map 技术与 Render-to-texture 和全屏 Post Processing 效果。

- 资源导入

项目中的资源会被自动导入, 并根据资源的改动自动更新。虽然很多主流的三维建模软件为 Unity 所支持, 不过对于 3ds Max、Maya、Blender、Cinema 4D 和 Cheetah3D 的支持比较好, 并支持一些其他的三维格式。

- 一键部署

Unity 可开发微软 Microsoft Windows 和 Mac OS X 的可执行文件, 在线内容通过 Unity Web Player 插件支持 Internet Explorer、Firefox、Safari、Mozilla、Netscape、Opera 和 Camino、Mac OS X 的 Dashboard 工具, 但是 Wii 程序和 iPhone 应用程序的开发需要用户购买额外的授权, 在价格上有所不同, 如图 1-2 所示。



▲图 1-2 一键部署

- 着色器 (shaders)

shaders 编写使用 ShaderLab 语言, 同时支持自有工作流中的编程方式或 Cg/GLSL 语言编写的 shader。一个 shader 可以包含众多变量及一个参数接口, 允许 Unity 去判定参数是否为当前所支持并适配最适合参数, 并选择相应的 shader 类型以获得广大的兼容性。因此, Unity 的着色器系统具有易用、灵活和高性能的特性。

- 地形编辑器

Unity 内建强大的地形编辑器, 支持地形创建和树木与植被贴片, 而且还支持水面特效, 尤其是低端硬件亦可流畅运行广阔茂盛的植被景观, 如图 1-3 和图 1-4 所示。



▲图 1-3 地形 1



▲图 1-4 地形 2

- 联网

现在大部分的游戏都是联网的, 令人惊喜的是, Unity 内置了强大多人联网游戏引擎, 具有 Unity 自带的客户端和服务端, 省去了并发、多任务等一系列繁琐而困难的操作, 可以简单地完成所需的任务。其多人网络连线采用 Raknet, 可以从单人游戏到全实时多人游戏。

- 物理特效

物理引擎是一个计算机程序模拟牛顿力学模型, 使用质量、速度、摩擦力和空气阻力等变量。其可以用来预测各种不同情况下的效果。Unity 内置 NVIDIA 强大的 PhysX 物理引擎, 可以方便、准确地开发出所需要的物理特效。

PhysX 可以由 CPU 计算, 但其程序本身在设计上还可以调用独立的浮点处理器 (如 GPU 和 PPU) 来计算, 也正因为如此, 它可以轻松完成像流体力学模拟那样的大计算量的物理模拟计算。并且 PhysX 物理引擎还可以在包括 Windows、Linux、Xbox360、Playstation3、Mac、Android 等在内的全平台上运行。

- 音频和视频

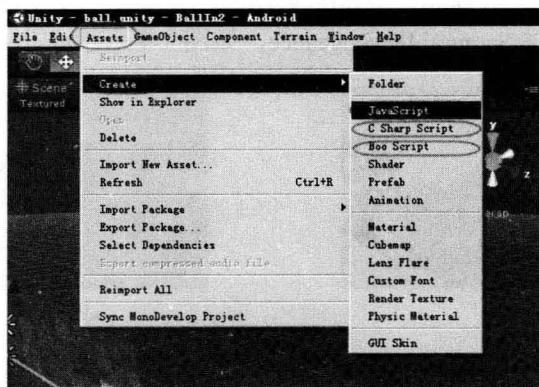
音效系统基于 OpenAL 程式库, 可以播放 Ogg Vorbis 的压缩音效, 视频播放采用 Theora 编码, 并支持实时三维图形混合音频流和视频流。

OpenAL 主要的功能是在来源物体、音效缓冲和收听者中编码。来源物体包含一个指向缓冲区的指标、声音的速度、位置和方向, 以及声音强度。收听者物体包含收听者的速度、位置和方向, 以及全部声音的整体增益。缓冲里包含 8 或 16 位元、单声道或立体声 PCM 格式的音效资料, 表现引擎进行所有必要的计算, 如距离衰减、多普勒效应等。

不同于 OpenGL 规格, OpenAL 规格包含两个 API 分支, 分别为以实际 OpenAL 函式组成的核心和 ALC API, 其中 ALC 用于管理表现内容、资源使用情况, 并将跨平台风格封在其中。OpenAL 还有“ALUT”程式库, 提供高阶“易用”的函式, 其定位相当于 OpenGL 的 GLUT。

- 脚本

游戏脚本为基于 Mono 的 Mono 脚本，是一个基于 .NET Framework 的开源语言，因此，程序员可用 JavaScript、C# 或 Boo 加以编写，如图 1-5 所示。



▲图 1-5 脚本

提示

由于 JavaScript 是开放式语言，考虑到脚本语言的通用性，本书采用 JavaScript 编写脚本。

- Unity 资源服务器

Unity 资源服务器具有一个支持各种游戏和脚本版本的控制方案，使用 PostgreSQL 作为后端。其可以保证在开发过程中多人并行开发，保证不同的开发人员在使用不同版本的开发工具所编写的脚本能够顺利的集成。

- 真实的光影效果

Unity 提供了具有柔和阴影与 lightmaps 的高度完善的光影渲染系统。光照图 (lightmap) 是包含了视频游戏中面的光照信息的一种三维引擎的光强数据。光照图是预先计算好的，而且要用在静态目标上。

说明

静态目标 static objects 在三维引擎里是区别于动态目标 dynamic object 的一种分类。

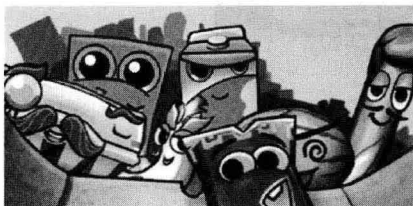
2. Unity 3D 的跨平台特性

Unity 类似于 Director、Blender game engine、Virttools 或 Torque Game Builder 等利用交互的图形化开发环境为首要方式的软件，其编辑器运行在 Windows 和 Mac OS X 下，可发布游戏至 Windows、Mac、Wii、iPhone 和 Android 平台，也可以利用 Unity web player 插件发布网页游戏，支持 Mac 和 Windows 的网页浏览。

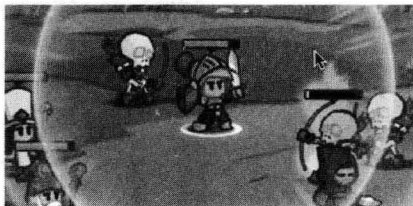
现在市面上已经推出了很多由 Unity 开发的基于 Android 平台、iPhone 平台以及大型的 3D 网页游戏，这些游戏都得到了很高评价。接下来将分别为读者介绍这 3 种游戏。

- 基于 Android 平台的游戏

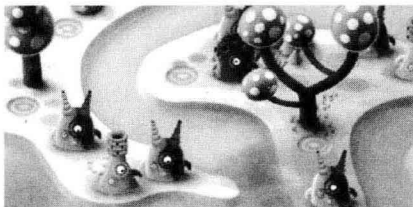
Unity 可以基于 Android 平台进行游戏开发, 由于其自身存在的优势, 因此开发的游戏也让人赏心悦目, 赞不绝口。例如, 由 Hidden Variable Studios 开发的 Bag It!, 如图 1-6 所示; 由 Mika Mobile, Inc 开发的 Battleheart, 如图 1-7 所示; 由 Infinite Dreams 开发的 Jelly Defense, 如图 1-8 所示; 由 MADFINGER Games 开发的 Samurai II: Vengeance, 如图 1-9 所示。



▲图 1-6 Bag It!



▲图 1-7 Battleheart



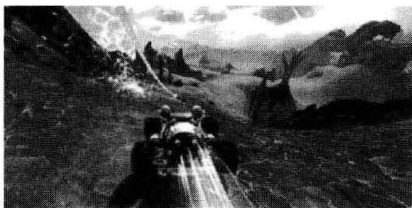
▲图 1-8 Jelly Defense



▲图 1-9 Samurai II: Vengeance

- 基于 iPhone 平台的游戏

Unity 依然可以基于 iPhone 平台进行游戏开发, 由于其自身存在的优势, 可以制作出绚丽多彩的 iPhone 平台游戏, 例如, 由 Punchers Impact 开发的 Crasher, 如图 1-10 所示; Warner Bros & Sticky Studios 开发的 Sucker Punch, 如图 1-11 所示; 由 Deemedya 开发的 Trial Xtreme 2, 如图 1-12 所示; 由 Bigpoint 开发的 Battlestar Galactica Online, 如图 1-13 所示。



▲图 1-10 Crasher



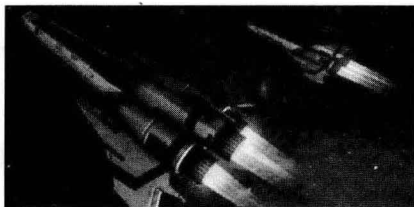
▲图 1-11 Sucker Punch

- 基于 Web 的大型 3D 网页游戏

同样 Unity 也可以开发基于 Web 的大型 3D 网页游戏, 市面上已经推出了很多这样的 3D 网页游戏, 例如《胸怀三国志》是曾开发《巨商》和《欢乐君主》等战略网络游戏的 Ndoors 公司金泰坤常务的最新力作, 如图 1-14 所示;《骑士的远征》是旭游网络继《兄弟篮球》之后的又一款力作, 如图 1-15 示;《木乃伊 OL》(The Mummy Online) 是由德国知名网页游戏开发商 Bigpiont 开发的一款全新网页游戏, 如图 1-16 示;《图腾王》是由上海 Tip Cat 开发的 3D 休闲对战网页游戏, 如图 1-17 所示。



▲图 1-12 Trial Xtreme 2



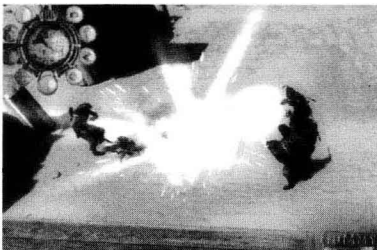
▲图 1-13 Battlestar Galactica Online



▲图 1-14 《胸怀三国志》



▲图 1-15 《骑士的远征》



▲图 1-16 《木乃伊 0L》



▲图 1-17 《图腾王》

12 开发环境的搭建

本节介绍 Unity 集成开发环境的搭建，开发环境的搭建分为两个步骤：Unity 集成开发环境的安装和目标平台的 SDK 与 Unity 3D 的集成。

1.2.1 Unity 集成开发环境的安装

本小节主要讲述如何构建 Unity 3D 的开发环境，之后对开发环境进行测试并创建第一个 Unity 3D 程序。前面已经对 Unity 3D 这个游戏引擎进行了简单的介绍，从本小节开始，将带领读者逐步搭建自己的开发环境，具体的步骤如下。

(1) 登录到 Unity 官方网站 <http://unity3d.com/unity/download> 下载最新的 Unity 安装程序，如图 1-18 所示，单击“Download Unity 3.5.1”按钮下载 Android 平台下的 Unity 安装程序。

(2) 若想下载 iPhone 平台下的 Unity 安装程序，即可单击 Developing On Mac OS X 超链接，跳转到下载 iPhone 平台下的 Unity 安装程序的界面；单击“Download Unity 3.5.1”按钮下载即可，