



PANDA 3D

刘 晖 林 欣 李 强 编著

开源商业化 三维游戏引擎

```
mercury.set_pos( 0.38 * orbitscale, 0, 0);mercury.set_scale(0.385 * sizescale);
//载入金星venus
```

```
us = window->load_model(framework.get_models(),"../models/planet_sphere");
```

```
PT(Texture) venus_tex = myTexturePool->load_texture(
```

```
"../models/venus_1k_tex.jpg");venus.set_texture(venus_tex, 1);
```

```
venus.reparent_to(orbit_root_earth);venus.set_pos( 0.72 * orbitscale, 0,
```

```
0);venus.set_scale(0.2 * sizescale);//载入火星mars
```

```
mars = window->load_model(framework.get_models(),"../models/planet_sphere");
```

```
PT(Texture) mars_tex = myTexturePool->load_texture(
```

```
"../models/mars_1k_tex.jpg");mars.set_texture(mars_tex, 1);
```

```
mars.reparent_to(orbit_root_earth);mars.set_pos( 1.52 * orbitscale, 0,
```

```
0);mars.set_scale(0.515 * sizescale);
```

```
//载入地球earth
```

```
earth = window->load_model(framework.get_models(),
```

```
"../models/planet_sphere");
```

```
PT(Texture) earth_tex = myTexturePool->load_texture(
```

```
"../models/earth_1k_tex.jpg");
```

```
earth.set_texture(earth_tex, 1);
```

```
earth.reparent_to(orbit_root_earth);
```

```
earth.set_scale(sizescale);
```

```
earth.set_pos(orbitscale, 0, 0);
```

```
void rotatePlanets(void){
```

```
//创建各种幕，根据层次旋转太阳、地球、月亮，大致模拟太阳系
```

```
day_period_sun = new CLerpNodePathInterval("day_period_sun", 20.0,
```

```
CLerpInterval::BT_no_blend, true, false, sun, NodePath());
```

```
day_period_sun->set_end_hpr(LPoint3f(360, 0, 0));
```

```
orbit_period_mercury = new CLerpNodePathInterval("orbit_period_mercury",
```

```
(0.241 * yearscale), CLerpInterval::BT_no_blend, true, false, orbit_root_mercury,
```

```
NodePath());
```

```
orbit_period_mercury->set_end_hpr(LPoint3f(0, 0, 0));
```

```
day_period_mercury = new CLerpNodePathInterval("day_period_mercury",
```

```
(59 * dayscale), CLerpInterval::BT_no_blend, true, false, mercury, NodePath());
```

```
day_period_mercury->set_end_hpr(LPoint3f(0, 0, 0));
```

```
orbit_period_earth = new CLerpNodePathInterval("orbit_period_earth",
```

```
(1.0 * yearscale), CLerpInterval::BT_no_blend, true, false, orbit_root_earth,
```

```
NodePath());
```

```
orbit_period_earth->set_end_hpr(LPoint3f(0, 0, 0));
```

清华大学出版社

开源商业化三维游戏引擎大揭秘

刘 晖 林 欣 李 强
陈小东 王学光 编著

清华大学出版社
北 京

内 容 简 介

Panda3D 是美国卡内基梅隆大学为娱乐巨头华特迪斯尼研发的开源、商业化三维游戏引擎，是世界十大开源游戏引擎中功能最完整、性能最稳定、商业化限制最少的一款。目前，迪斯尼仍在使用的卡内基梅隆大学在校学生及世界各地的业界人员也以开放、共享全部源代码方式不断促进这款游戏引擎的持续发展。在竞争激烈的国外游戏引擎市场中，Panda3D 始终引人注目，技术领先。

本书为中文读者深入、完整掌握游戏引擎 C++、Python 核心内容提供了“先天下之闻而阅”的揭秘地图。全书图文并茂、实例丰富，配书光盘中有 80 多段演示效果视频，为 Windows、Linux、Mac 操作系统中的 C++、Python 游戏引擎掘金者提供了价值无穷的资源。

作者结合深入的专业知识及多年的工作实践开发经验，重点针对市场、读者技术实践需求编写此书，通过全面学习，相信读者能够轻松掌握 Panda3D 的使用技巧，简单快捷地步入 3D 应用开发新天地。

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 13701121933

图书在版编目(CIP)数据

开源商业化三维游戏引擎大揭秘/刘晖，林欣，李强编著. —北京：清华大学出版社，2013
ISBN 978-7-302-30609-2

I. ①开… II. ①刘… ②林… ③李… III. ①三维动画软件—游戏程序—程序设计 IV. ①TP391.41

中国版本图书馆 CIP 数据核字(2012)第 274891 号

责任编辑：张彦青 桑任松

装帧设计：杨玉兰

责任校对：周剑云

责任印制：沈 露

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座 邮 编：100084

社 总 机：010-62770175 邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载：<http://www.tup.com.cn>, 010-62791865

印 刷 者：北京鑫丰华彩印有限公司

装 订 者：北京市密云县京文制本装订厂

经 销：全国新华书店

开 本：185mm×260mm 印 张：60.25 字 数：1462 千字
(附 DVD1 张)

版 次：2013 年 1 月第 1 版

印 次：2013 年 1 月第 1 次印刷

印 数：1~4000

定 价：130.00 元

前言

当英语论坛上还在招募志愿者整理 C++ 文档、调试 C++ 实例时，中文读者已经可以“先天下之阅而阅”，深入 Panda3D 游戏引擎 C++/Python 核心内幕，了解、学习、使用、分析、定制、扩展这款娱乐巨头麾下的开源商业化游戏引擎了。

Panda3D 游戏引擎是 US News 排名中计算机水平第一的美国著名学府“卡内基梅隆大学”为娱乐巨头“华特迪斯尼”研发的产品。目前，迪斯尼仍然使用这款游戏引擎，而卡内基梅隆娱乐技术中心也不断以开源方式吸引在校学生及世界各地的业界人员共同促进这款游戏引擎的持续发展。因此，如果读者静下心来掌握 Panda3D 游戏引擎，无疑会使自己能够驾驭的 3D 技术与娱乐巨头的 3D 技术比肩，与世界领先的水平看齐。更为重要的是，读者可以毫无遗漏地获得全部源代码！

Panda3D 游戏引擎是所有著名开源游戏引擎中最为完整、商业化运作几乎没有限制的一款，是一个无比丰富的金矿。既可以拿来开发各种 3D 应用，又可以作为产品原型系统进行客户定制。目前，除了官方网站上的英语资源外，主要图书包括 Panda3D 1.6 Game Engine Beginner's Guide(《Panda3D 1.6 游戏引擎初学指南》)及 Panda3D 1.7 Game Developer's Cookbook(《Panda3D 1.7 游戏引擎开发者实践》)两本，分别于 2011 年 2 月、3 月出版。而本书是第一本关于 Panda3D 游戏引擎的中文图书。

Panda3D 游戏引擎支持 Unix、Linux、Windows、Mac 等操作系统，采用 C++ 开发，与 Python 脚本语言进行了有效绑定。因此，开发者既可以使用 C++，也可以使用 Python；既可以选用 Unix/Linux 操作系统，也可以选用 Windows 操作系统。为了满足开发者多元化的需求，本书选用 Panda3D 1.7.x 版本，相关开发实例验证于 Windows、Fedora(Linux)操作系统，推荐的开发环境是 Windows + Visual Studio C++ 2008、Fedora + Eclipse、Fedora + g++、Windows + Eclipse PyDev(Python)、Fedora + Eclipse PyDev(Python)。当然，如果开发者喜欢 Ubuntu、Debian、Mac OS X，除了需要自行安装适当配置以外，有关开发指南同样有效。本书推荐使用 Autodesk 3ds Max 2010、Autodesk Maya 2009、Blender 处理游戏资产，对如何与 Panda3D 游戏引擎进行资产整合进行了必要说明。

尽管 3D 游戏引擎涉及大量的理论知识和数学方法，但是本书主要偏重于编程实践和应用开发。全书共分 22 章。

第 1 章总结了国内目前较受关注游戏的游戏引擎特征，介绍了游戏引擎发展历程，比较了开源游戏引擎总体差距、商业化运作案例，提供了游戏引擎成本及受益计算方法，归纳了游戏引擎选择尺度，深入分析了十大开源游戏引擎，解释了软件许可证方面的限制，最后指导读者在 Windows 及 Fedora 上安装 Panda3D 游戏引擎。

第 2 章指导读者安装、熟悉 Panda3D 游戏引擎开发环境。选定的集成编译环境是 Linux 中 Eclipse C/C++ 的 Helios 版及 Eclipse & PyDev 1.6.3，Windows 中 Visual Studio 2008 及 Eclipse & PyDev 1.6.3。在介绍了安装过程后，读者能够根据说明步骤调试 Python、C++ 版“你好，大熊猫”程序。编译过程与方法适用于全书所有实例。

第 3 章全面介绍了游戏引擎的配置变量，举例说明了通过配置文件、代码访问处理变

量的方法。同时说明了游戏引擎自带工具集，重点学习使用 Pview。

第4章说明了 Egg 模型文件定义语法、骨架及变形动画约定结构，演示了引擎自带的场景浏览器，总结了导演工具鼠标快捷键语义，举例介绍了导入、转换、管理 Autodesk Maya、Autodesk 3ds Max、Blender 游戏资产的方法。主要涉及了游戏编程以外的工作流程。

第5章深入解释游戏引擎场景处理设计理念、重要类的继承关系及函数功能，通过“合唱团”场景说明场景树形结构的灵活性，着重介绍了共同抽象基类 PandaNode、节点类 NodePath、窗口框架类 WindowFramework、引擎框架类 PandaFramework、窗口属性类 WindowProperties、Python 启动类 ShowBase，涉及场景属性调整、场景检索、占位符节点多层次管理等功能。开发实战主题为太阳系程序。

第6章围绕模型、地形、角色展开。介绍了文件命名跨平台兼容处理、环境变量动态管理及设置。重点是角色静态模型与动画模型的绑定、控制、管理，支持自动处理，也支持自定义处理。不同动画效果可以通过4种方式进行混合，各帧动画之间允许进行平滑插值。Character 是一种特殊的动画角色，多用于实现具有关节的组合对象。游戏引擎支持通过高度场图像自动生成地形，提供两个演示实例。开发实战主题为角色控制、关节处理。

第7章集中讨论渲染属性。引擎允许使用点光源、定向灯、环境灯以及聚光灯4种类型的灯。材质主要模拟环境色散、漫反射、镜面反射、发射率。景深测试能够渲染透明、立体阴影。雾产生指数型、线性、指数平方型3种渲染效果。透明测试多用于实现镂空效果。色彩写入掩码渲染图像时结合透明通道生成3D或RGB色彩过滤效果。抗走样模式采用配置文件、图形卡属性等参数设置，提供多种质量性能选择。借助重新配色或颜色调节，渲染过程中可以实时控制物体表面、顶点颜色。前后表面拣选控制渲染表面指向，以此呈现蜡版、阴影、内部等视觉效果。灯光缓变支持以HDR色调映射或卡通着色方法实现独特的灯光渲染特效。蜡版缓冲器是一种灵活的渲染数据处理模板，经过设计，渲染将输出奇特的效果。另外，游戏引擎提供公告牌、指南针、贴花、灯光组合、显示边界等渲染效果。开发实战主题为迪斯科灯光、无底洞。

第8章处理各种纹理，涉及1D、2D、3D(立方体)纹理，支持JPG、PNG、TIF、BMP以及具有变频译码(Mipmaps)能力的DDS文件。纹理的主要特性是尺寸、自动填充、换行模式、过滤器。在运行过程中，开发者需要对各种纹理进行载入、替换、动态生成、管理、变换，多采用纹理池、纹理层实现。特殊的纹理效果可通过多纹理机制完成，经过调节色彩纹理、法线、光泽、发光等属性，产生调节、贴花、混合、凹凸映射、高度视差、光泽、发光。高级纹理技术包括自动纹理、投影纹理、3D纹理、球体映射、立方体映射、动画纹理、视频纹理。开发实战主题为凹凸贴图、纹理交换。

第9章涉及商业化扩展的重点主题着色器。游戏引擎支持 Nvidia 与微软一起制定的着色器语言 Cg，包括顶点着色器、几何着色器、片段着色器(像素着色器)。着色器输入类型覆盖 Cg 语言约定，同时加以扩展，允许定制、注册新的类型。游戏引擎内部支持模型空间、游戏世界空间、视线空间、API 视线空间、剪裁空间、API 剪裁空间之间进行有效转换。着色器有时与相机、镜头状态密切相关，游戏引擎支持透视镜、正交镜头、鱼眼镜、球镜、矩阵镜、柱面镜。透视镜控制参数最多，包括焦点、视野、近平面距离、远平面距离、胶片尺寸、焦距长度、宽高比。调试时，可以使用体外体验 OOB 模式。提供了6个着色器开发程序。开发实战主题为卡通着色、发光滤镜。

第10章介绍将渲染、动画、音频、交互等构件组合为幕，进一步构造自动运行的串行

情节、并行情节的技术。是游戏关卡的基本技术,涉及全局时钟、事件。比较重要的幕包括插值幕、插值函数幕、弹道曲线幕、声音幕、运动路径幕、粒子幕。除此之外,还可以采用异步任务、事件处理自主组织游戏构件。它们都支持多线程机制。开发实战主题为小行星、旋转木马、电视机中的茶壶。

第 11 章侧重介绍了 OpenAL、FMOD、Miles 中的前两种开源音频库。音频处理相对简单,仅涉及音频系统的结构、循环控制、音频管理器、DSP 效果、3D 效果以及 Python 音频自动化幕等内容。开发实战主题为播放多媒体、扭曲、萤火虫。

第 12 章说明游戏引擎自带的高级交互功能。为了便于国际化集成,引擎支持 ISO8859、UTF-8 及 Unicode 字符集,提供互相转换函数,允许根据字库及 Egg 文件选择字体,通过文本节点、图像文本、屏幕文本、屏幕图像等组件进行显示、模型生成、样式调整。基本交互组件包括 C++ 中的 PGui 按钮、文本输入、滑块、框架、滚动条、进度条,以及 Python 中的 DirectGUI 按钮、复选框、单选框、对话框、输入框、框架、标签、菜单、滚动列表、进度条、滑块、滚动条、滚动框架。开发实战主题为音乐盒。

第 13 章深入游戏引擎内部构造,讨论了顶点、几何形体的存储;3D 模型的生成、细节调整、读取、存盘;边界的计算、扩展、相交检查;矩阵基本运算;纹理图像内存处理等内容。开发实战主题为动态构造立方体模型。

第 14 章剖析了渲染处理过程。涉及渲染的基本流程,流程的集成性,流程的扩展性,流程功能模块的划分。由图形管道、图形引擎、图形状态监控、图形输出、显示区域,图形上下文、图像过滤器共同实现。渲染输出可以进一步用于动态纹理、过滤器以产生特殊效果。游戏引擎支持 3D 显示(左右眼不同视觉体验)。开发实战主题为影子处理、Python 过滤器。

第 15 章说明如何通过力场产生粒子,某一时刻在特定空间按一定规律运动形成爆炸、烟雾、气泡、水泡、浮云等物理仿真效果。具体由粒子渲染器、粒子发射器、粒子工厂实现。游戏引擎带有粒子面板,可以设计、调试各种粒子文件,包括尘土飞扬、云蒸雾绕、喷泉、烟雾、烟圈、火炬等实例。开发实战主题为分形树。

第 16 章重点介绍碰撞检测,可以通过自定义的球体或多边形进行检测,也可以对所有几何形体进行检测。碰撞形体具体类型包括碰撞球、管道、逆置球、平面、多边形、射线、直线、分段、抛物线、地板网格、双球体、盒等。碰撞类型分为 from 类检测、into 类检测。碰撞事件由碰撞处理器队列、碰撞处理器事件、碰撞物理处理器、碰撞推处理器、物理碰撞处理器、流式碰撞推处理器、地面碰撞处理器、重力碰撞处理器构成,适于不同碰撞场合。发生碰撞后,能够处理的事宜主要包括获取有关形体的指针及对应节点,设置获取碰撞相对时间,设置获取有关物体的碰撞点、可穿越深度、碰撞法线等。碰撞掩码有助于优化碰撞遍历检测。引擎也能够处理高速运动物体。开发实战主题为漫游者。

第 17 章重点说明物理系统。支持内建物理系统、ODE、NVIDIA PhysX 以及 Bullet(针对 1.8 版本以上)。多用于模拟物体速率、加速度、动力、力场、摩擦、撞击;实现角色自动行进;仿真机动物体运行;处理水、浪等流动物体;展现服装、织物质感等效果。开发实战主题为迷宫滚球。

第 18 章介绍通过人工智能、有限状态机控制非玩家角色的方法。允许非玩家角色实现寻找、逃离、追逐、逃避、游逛、羊群、避障、路径跟踪等自动督导行为,也允许完成最短寻径。FSM 是 Python 专用的方法,以状态机模型式的状态迁移、状态输入过滤等方式定

义游戏状态切换。开发实战主题为国际象棋。

第 19 章集中补充了很多高级开发杂项。涉及键盘、鼠标、流级别网络 API、对象级别网络 API、多媒体文件包的子文件、流文件、加密文件、补丁处理、调试日志、开发集成环境调试、跨语言子类派生、脚本语言代码质询等内容。开发实战主题为通过鼠标和键盘控制 FPS 相机。

第 20 章讨论 1.7 版本以后新增的分布式应用功能，通过打包、分布式部署、浏览器插件等机制，支持网页游戏或者独立的分布式应用。考虑安全问题，允许使用数字签名、数据源认证等方式加以保护。开发实战主题为运动踪迹。

第 21 章介绍了性能监测的工具及方法。发现性能瓶颈是一个非常复杂的话题，甚至比设计、开发程序更富有挑战性。

第 22 章说明游戏引擎源代码编译方法及主要步骤，适合高级开发者深入使用。

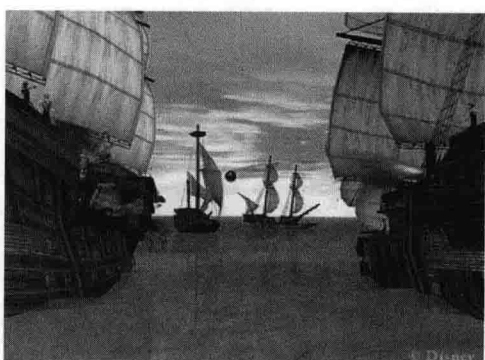
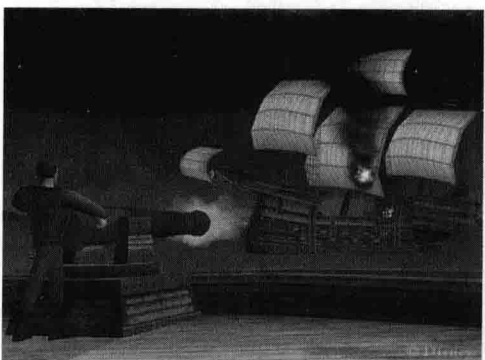
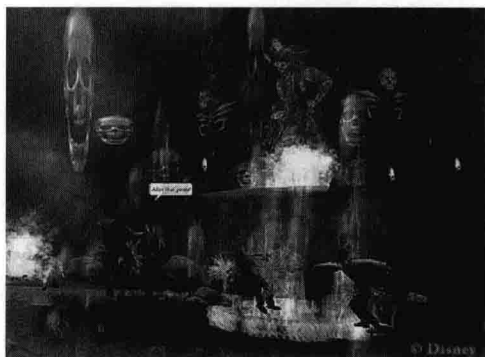
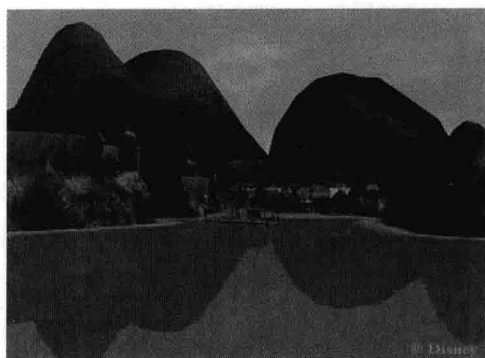
全书由刘晖、林欣、李强、陈小东、王学光参与编写。由刘晖、林欣负责统校全书。实例开发由刘晖、林欣、陈小东负责完成。

另外，本书使用的开源软件许可证已经放在 DVD 光盘相应软件目录之中，读者进一步开发时务必遵守相关许可证约定。本书使用的插图或者自行制作，或者来自互联网。来自互联网的部分插图主要是由于与游戏引擎机制原理、输出结果有关，仅作为科技事实提供给读者。作者没有参与有关游戏运作、宣传、评论等商业活动。特此声明。

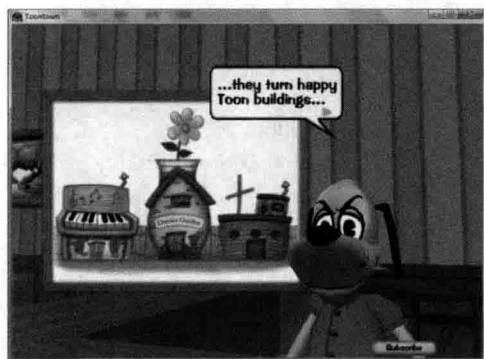
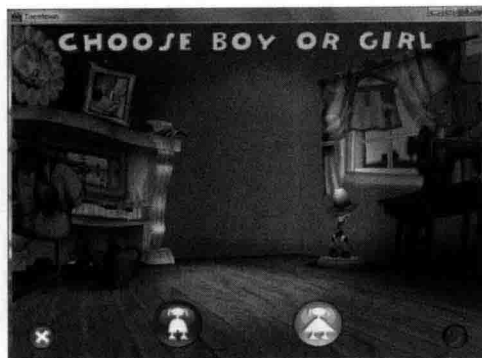
由于作者水平有限，时间紧迫，书中难免有不足之处，恳请读者批评指正。

如果说 Panda3D 是一座活跃的金矿，现在，本书已经绘制出一幅完整的金矿地图。让我们一起去挖掘宝藏吧。

(截止发稿时，Panda3D 游戏引擎中文支持网站第一版也已经开发完毕，正在加紧内测、准备上线，争取第一时间奉献给大家，与广大读者、开发者、技术爱好者、游戏厂商深入地交流互动，请访问 www.panda3d.org.cn。)



迪斯尼：加勒比海盗(采用 Panda3D 引擎)



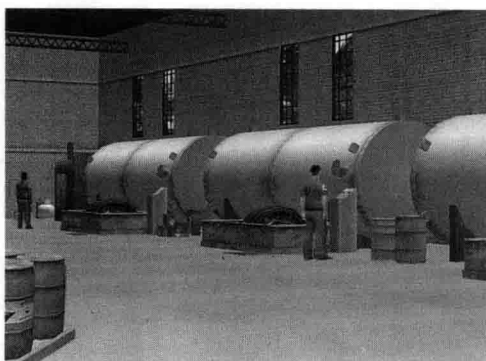
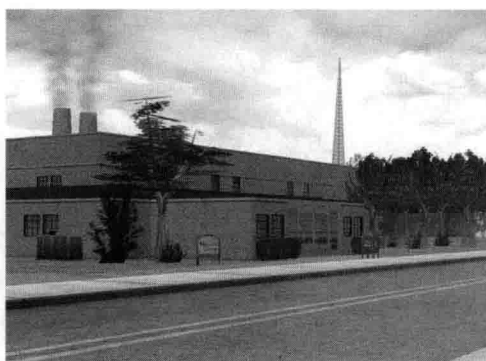
迪斯尼：卡通城(采用 Panda3D 引擎)



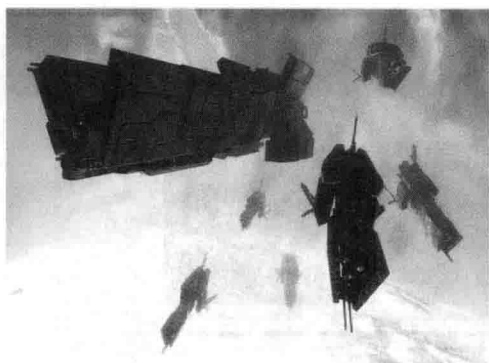
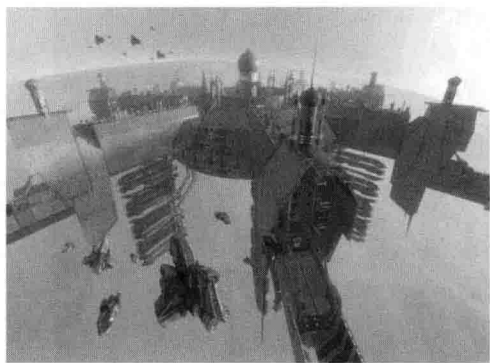
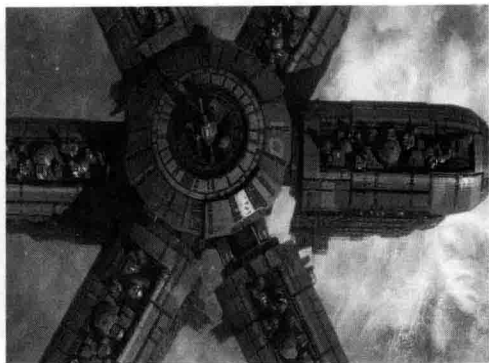
秋月娱乐：吸血鬼传奇(采用 Panda3D 引擎)



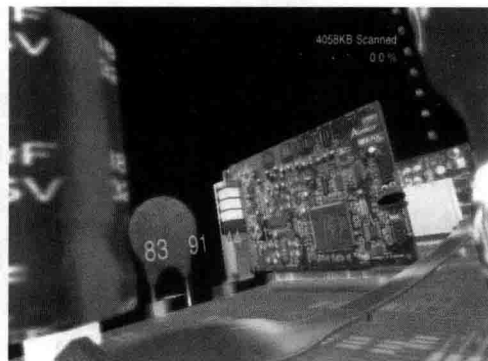
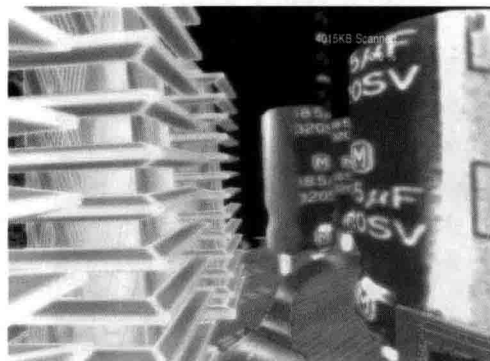
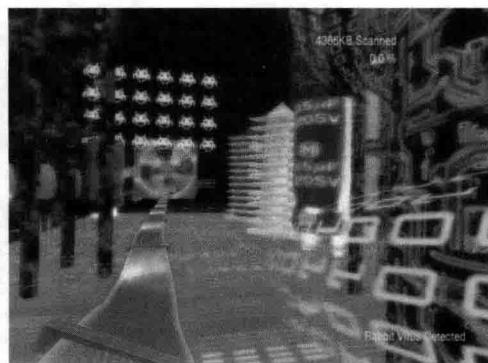
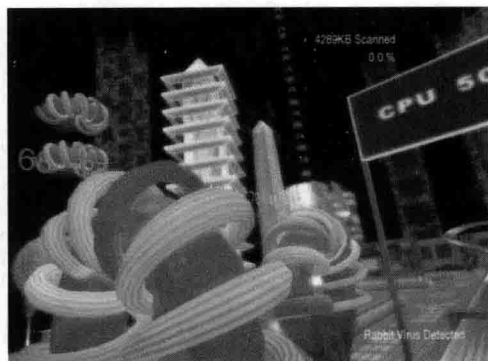
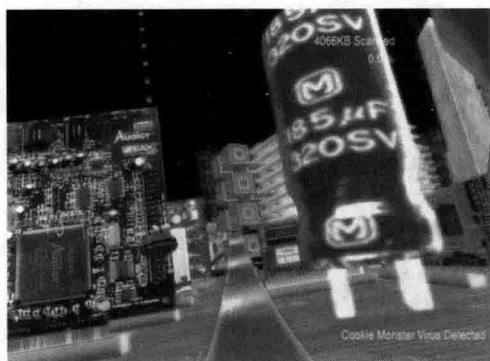
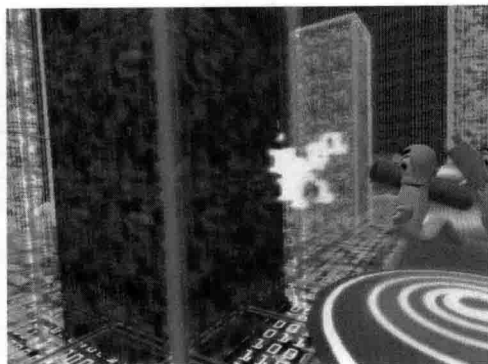
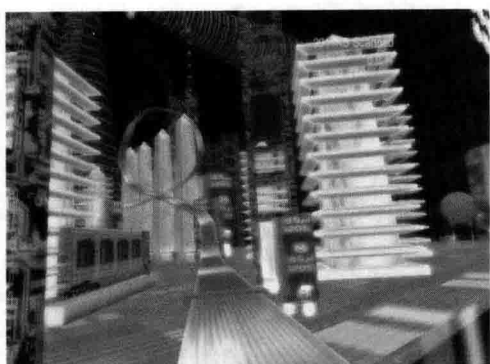
秋月娱乐：巫胡岛的幽灵海盗(采用 Panda3D 引擎)



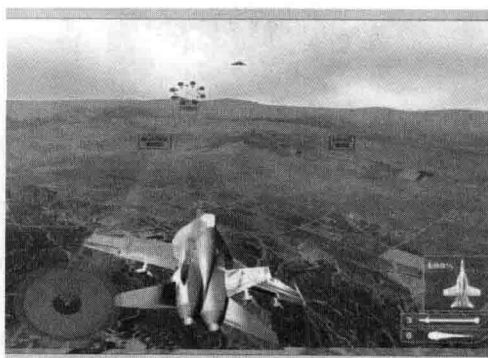
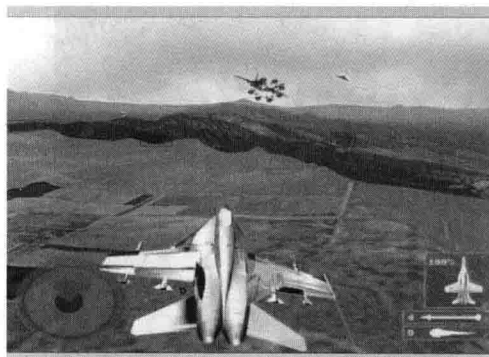
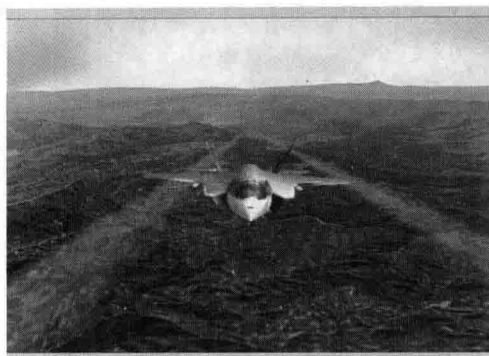
Sim Ops Studio: Code3D(采用 Panda3D 引擎)



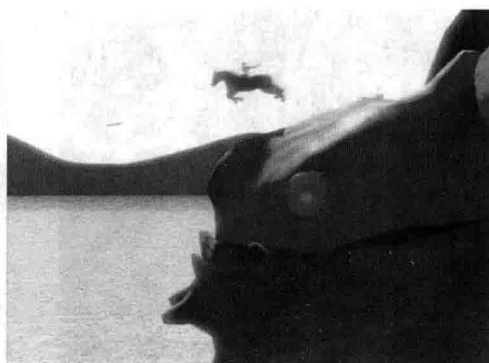
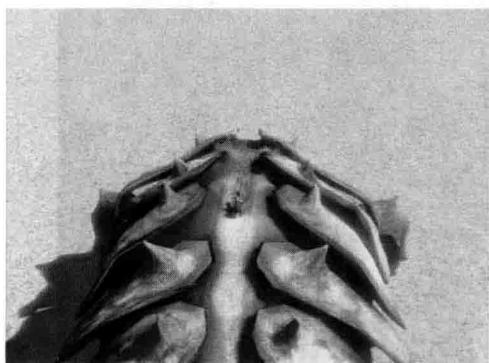
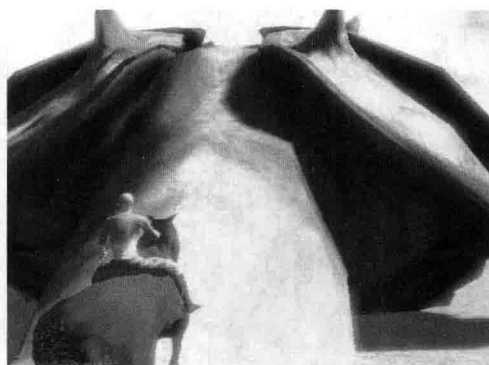
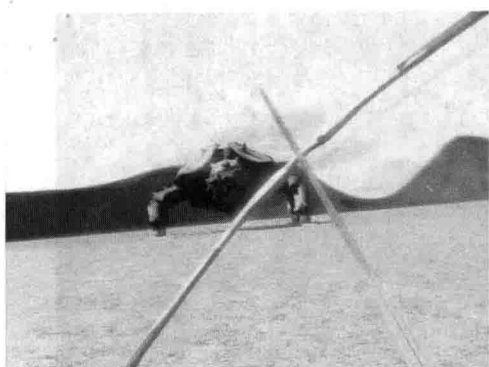
Treeform: 敢死天使 第二次星宿战争(采用 Panda3D 引擎)



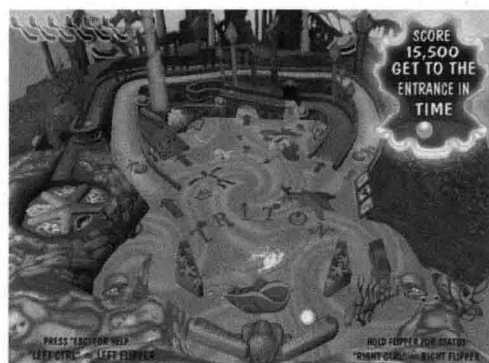
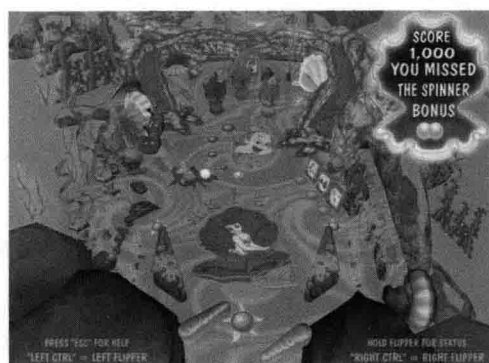
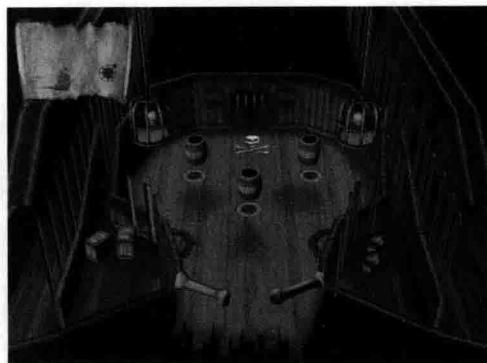
卡内基梅隆大学学生：挽救中毒电脑(采用 Panda3D 引擎)



卡内基梅隆大学学生：飞行模拟器(采用 Panda3D 引擎)



卡内基梅隆大学学生：庞然大物(采用 Panda3D 引擎)



迪斯尼：符合物理动力特性的弹球系列(采用 Panda3D 引擎)