

可定制军用三维图形绘制 引擎系统理论及运用

张 森 曾艳阳 刘 刚 著



可定制军用三维图形绘制 引擎系统理论及运用

张 森 曾艳阳 刘 刚 著



科学出版社

北 京

内 容 简 介

本书全面地论述军用三维图形绘制引擎设计的系统理论和相关技术,并系统地总结作者在军用三维视景仿真引擎方面的研究经验和最新进展,给出视景仿真引擎系统的设计开发方法和采用 Vega Prime 视景仿真软件开发分布式对海作战视景仿真引擎系统的典型应用案例,既包括理论分析,又包括工程实现技术。本书选材广泛、内容新颖、研究思路独特、实用性强。

本书可供从事视景仿真尤其是对海作战视景仿真等领域研究和应用的广大科技工作者参考使用,也可以作为高等院校虚拟现实技术、计算机应用、武器系统仿真等专业的教师、研究生进行有关课题研究实践或课程学习时的参考书。

图书在版编目(CIP)数据

可定制军用三维图形绘制引擎系统理论及运用 / 张森,曾艳阳,刘刚著. —北京:科学出版社,2016. 8

ISBN 978-7-03-049620-1

I. ①可… II. ①张…②曾…③刘… III. ①计算机图形学
IV. ①TP391. 411

中国版本图书馆 CIP 数据核字(2016)第 203726 号

责任编辑:杨向萍 张晓娟 / 责任校对:蒋 萍

责任印制:张 伟 / 封面设计:迷底书装

科学出版社 出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

北京教图印刷有限公司 印刷

科学出版社发行 各地新华书店经销

*

2016 年 8 月第 一 版 开本:720×1000 B5

2016 年 8 月第一次印刷 印张:15

字数:303 000

定价:88.00 元

(如有印装质量问题,我社负责调换)

前 言

三维图形绘制引擎是视景仿真软件的核心,它主要负责虚拟场景的图形绘制,其优劣直接决定显示效果。但是半开放式高级引擎对外屏蔽源代码,开放式高级引擎的架构比较臃肿,需要设计一套架构简洁、效率较高的三维图形绘制引擎。

本书总结近几年内三维图形绘制引擎技术的研究现状及存在问题,针对目前三维图形绘制引擎存在的不足,从军用三维图形绘制的多样化需求出发,深入论述并分析图形引擎的总体技术、绘制流水线技术、光场绘制技术、大规模战场地形实时绘制技术及引擎平台的接口技术。在这些技术的基础上,论述一套虚拟化装配可定制仿真算法组件的引擎平台设计方法,设计系统所需要的多个功能模块,突破多项图形引擎的底层支撑技术,为三维图形绘制引擎的军事化及通用化提供技术支持,使研发工作有据可循、易于开展。

本书共 8 章。第 1 章为概论,综述三维图形绘制引擎的发展历史及存在的问题。较全面地概括近年来国内外在三维图形绘制引擎开发方面的最新进展,主要包括引擎的总体技术、绘制流水线技术、光场渲染技术、大地形实时绘制技术及应用平台技术等。第 2 章为 MCGRE 的总体技术。针对引擎的功能需求进行分析,通过研究引擎的框架设计、可定制功能机制、仿真资源装配关系、数据流、扩展机制、各组成单元的模块设计及中间件等总体技术,提出一种可定制三维图形绘制引擎的结构模型,为底层关键技术的深入研究做铺垫。第 3 章为核心子系统开发-绘制流水线设计。针对图形绘制全过程中数据流交互复杂、模块间耦合性强、新算法无法快速嵌入等难题,提出一种三段式绘制循环模式。第 4 章为光场绘制技术。由于动态海面影响太阳光在水中的干射、衍射现象,针对该现象的光照模型建模困难、动态光场难以渲染问题,搜集与整理引擎中常规光场绘制模型,分析风海波下光场的物理特性,建立水下光场绘制的数学模型,提出一种水下光场渲染方法。第 5 章为大规模战场地形实时绘制技术。针对地形拼接的过渡带存在缝隙、对比度强及地形失真等问题,提出一套军用大地形实时绘制数据处理技术。通过研究地形高程数据的组织方法、色彩空间的图像边界检测、像素特征点集对的提取与匹配等关键技术,建立一种基于视域的无缝拼接算法,实现过渡带的平滑处理。第 6 章为 MCGRE 接口技术。根据平台设计思想,讨论可定制功能策略和可定制人机接口封装方法,封装仿真组件,设计引擎平台的可定制功能引导界面,并对引导界面进行实例测试。第 7 章为军用视景仿真引擎系统设计与开发。从系统的模型需求分析出发,详细论述三维视景模型的建立方法、三维视景系统的一般开发流程及关

键技术配置。第 8 章为分布式对海作战视景仿真引擎系统开发与应用。以对海作战视景仿真系统为例,介绍海战场的二三维规划与同步、作战想定规划与解析、推演规则动态建模、参战实体规划与建模、实体特效开发及大数据调度规则、场景图元的管理及视景仿真系统开发评估结果。

在本书的撰写过程中,张森负责第 1~5 章的撰写及全书的统稿工作,曾艳阳负责第 7 章、8 章的撰写,刘刚负责第 6 章的撰写。研究生尚根锋、张元亨、金超和杨婷婷负责校对,在此表示感谢!

由于作者水平有限,书中难免存在不当之处,恳请广大读者批评和指正。

作 者

2015 年 10 月

目 录

前言

第 1 章 概论	1
1.1 三维图形绘制引擎技术概述	1
1.2 引擎系统简介	2
1.2.1 游戏引擎	2
1.2.2 仿真引擎	14
1.2.3 引擎的底层绘制编程语言	39
1.3 三维图形绘制引擎核心技术	40
1.3.1 三维图形绘制引擎总体技术	40
1.3.2 绘制流水线技术	43
1.3.3 光场绘制技术	45
1.3.4 大地形实时绘制技术	47
1.3.5 三维图形绘制引擎的其他技术	51
第 2 章 MCGRE 的总体技术	54
2.1 概念辨析	54
2.2 MCGRE 的设计方法	54
2.3 设计目标与功能需求	56
2.3.1 设计目标	56
2.3.2 功能需求	57
2.3.3 MCGRE 软件编程要求	57
2.4 总体设计	59
2.4.1 开发方案	59
2.4.2 框架设计	61
2.4.3 引擎模型	67
2.4.4 组成单元	70
2.4.5 装配关系	81
第 3 章 核心子系统开发-绘制流水线设计	83
3.1 绘制流水线的简介	83
3.1.1 固定绘制流水线	83
3.1.2 可编程绘制流水线	85
3.2 三段式绘制流水线设计	85

3.2.1	三段式绘制循环	86
3.2.2	场景组织	87
3.2.3	绘制预处理	90
3.2.4	绘制器	93
3.3	绘制流水线设计中的调度机制	97
3.3.1	一致变量回调	98
3.3.2	裁减回调	98
3.3.3	动画更新回调	98
3.3.4	相机回调	99
3.3.5	文件读写回调	99
3.4	绘制流水线设计中的组件封装机制	99
3.5	绘制流水线设计中的扩展机制	101
3.5.1	基于场景图的扩展机制	101
3.5.2	基于回调的扩展机制	102
3.5.3	Shader 渲染扩展	102
第4章	光场绘制技术	103
4.1	光场可视化简介	103
4.2	引擎中常用光场渲染模型	103
4.2.1	Gouraud 明暗处理	104
4.2.2	Phong 明暗处理	105
4.2.3	Z 缓冲器算法	106
4.2.4	光线跟踪算法	107
4.2.5	辐射度算法	108
4.2.6	光线跟踪与辐射度结合算法	109
4.3	一种水下光场的绘制方法	110
4.3.1	风海波下光场渲染	110
4.3.2	物理交互光场的绘制	114
4.4	光照渲染模块	117
4.4.1	光照渲染模块设计	117
4.4.2	创建一个聚光灯示例	118
4.5	基于加速技术的像素级光照渲染	119
第5章	大规模战场地形实时绘制技术	121
5.1	大规模战场地形实时绘制技术简介	121
5.2	战场地形高程数据的统一规范	122
5.2.1	地形高程数据不统一的表现	122
5.2.2	水平测量基准的统一	123
5.2.3	垂直测量基准的统一	123

5.2.4 多源数据记录与转换格式的统一	124
5.2.5 统一的海陆高程数据的建立方法	124
5.3 军用大地形无缝拼接技术	126
5.3.1 拼接需求	126
5.3.2 拼接方案	127
5.3.3 拼接算法	127
5.3.4 拼接测试	133
5.4 军用大地形自适应混合加权数据处理技术	134
5.4.1 加权内部插值算法及数学分析	135
5.4.2 加权外部推算算法及数学分析	136
5.4.3 混合加权动态校正方法	138
5.4.4 自适应分级调度策略	138
5.4.5 算法实施步骤	139
5.4.6 算法实施实例	140
5.5 大地形管理模块设计	142
5.5.1 地形数据转换	142
5.5.2 地形组织与存储调度	142
5.5.3 地形实时绘制算法	143
第6章 MCGRE 接口技术	145
6.1 可定制策略	145
6.1.1 插件可定制功能设计	145
6.1.2 算法可定制功能设计	145
6.1.3 可定制功能接口设计	146
6.2 接口封装	147
6.2.1 接口封装方法	147
6.2.2 基于共享内存的通用化柔性接口设计	149
6.3 引导界面	150
6.3.1 界面设计	150
6.3.2 内部数据接口	152
6.4 实例测试	153
第7章 军用视景仿真引擎系统设计与开发	154
7.1 系统模型的需求分析	154
7.2 三维视景模型建立	155
7.3 三维视景开发架构	159
7.4 系统的关键技术配置	160
7.4.1 三维模型表现形式转换技术	160
7.4.2 场景显示	162

7.4.3 运动模型	164
7.4.4 环境构成	165
7.4.5 大场景管理	165
第 8 章 分布式对海作战视景仿真引擎系统开发与应用	168
8.1 海战场的二三维规划与同步	168
8.1.1 海战场的二三维规划	168
8.1.2 二三维同步实现	169
8.2 海洋环境配置	170
8.3 作战想定规划与解析	172
8.3.1 作战想定文件的描述	172
8.3.2 使用 ADO.NET 存取 XML	174
8.3.3 关联仿真数据库	177
8.3.4 关联实体属性	179
8.4 推演规则动态建模	179
8.5 参战实体规划与建模	180
8.5.1 基于特征约束的随机误差建模	180
8.5.2 典型应用	181
8.6 实体特殊效果开发	182
8.6.1 通信链路可视化	183
8.6.2 指控关系可视化	187
8.6.3 雷达探测可视化	188
8.6.4 导弹尾焰可视化	192
8.6.5 爆炸特效可视化	195
8.6.6 声音效果	197
8.6.7 其他特殊效果可视化	199
8.7 大数据调度规则	200
8.7.1 Vega Prime 中的视点管理	200
8.7.2 智能化视点与窗口的控制	201
8.8 场景图元的管理	204
8.8.1 三维实体模型的简化	204
8.8.2 实时连续 LOD 场景模型的绘制与显示	204
8.8.3 碰撞检测	205
8.9 视景仿真系统开发结果	205
参考文献	207
附录 A	217
附录 B	227

第 1 章 概 论

1.1 三维图形绘制引擎技术概述

随着虚拟技术的推进,三维图形绘制引擎技术得到了长足的进步,在军用视景仿真行业中的应用也得到了迅速的推广和深入,并成为控制科学与工程领域最活跃的研究方向之一。从武器装备设计到武器系统作战效果评估虚拟展示,从作战指挥想定规划到作战指挥流程模拟,人们都感受到了图形引擎技术的独特魅力。

以往,绘制引擎技术研究注重算法的实时能力和更真实的图形输出效果(实时性与逼真度),而图形处理算法又具有很高的计算复杂度,且需要在一些大型工作站上实现;大众使用的个人电脑由于内存较低,只能得到相当少的 CPU 处理能力。因此,导致早期的视景仿真应用研究从低到高存在两种走向:一是基于现有的视景仿真软件开发应用系统;二是利用图形处理算法结合计算机底层硬件提高运行效率。随着计算机硬件的不断更新与升级,高速发展的图形硬件技术促进了图形学的发展。大众电脑开始具备强大的计算能力和强劲的内存,开发人员也开始使用高级语言开发图形应用软件,并且从计算机图形学研究领域获得理论支持和应用拓展。但同时也产生了日益增长的用户需求与软件开发难度大、开发速度缓慢、维护困难和功能较少之间的矛盾,尤其表现在需求多样化与软件单一化。针对该问题出现了绘制引擎的概念。基于此,视景仿真又分化成两种走向:一是基于经典模板开发自己的底层三维图形绘制引擎软件;二是基于已有的引擎开发自己的军用模块进而封装成具有特定军事功能的视景仿真软件。

在开发三维图形应用程序的过程中,为了确保核心算法模块的通用性与扩展性,并尽可能地降低开发成本,通常把与核心技术密切相关的内容提取出来进行独立开发。而三维图形引擎正是这样的一个独立模块,相当于程序开发人员的工具包,提供了一系列初始化、图形处理、碰撞检测等相关的 API 函数或者 DLL 形式。它像汽车引擎一样,向上层图形应用提供驱动力,驱动视景数据实现绘制。目前,计算机图形学已进入三维时代,三维图形无处不在。三维图形引擎作为上层图形应用的核心,决定着三维图形的表现、稳定性、速度和真实感等特性,这些特性都与三维图形引擎密切相关,并建立在引擎基础之上。

通过对三维图形绘制引擎关键技术的突破,使底层图形绘制引擎的设计方法流程化。根据用户开发需求,设计一个可定制的三维图形绘制引擎。用户在此引

擎基础上,能够快速组建一个自己的三维图形绘制引擎,为深刻理解引擎行为及指导其理论设计提供坚实的技术支撑,为军用视景仿真软件的工程化、实用化奠定基础。

1.2 引擎系统简介

1.2.1 游戏引擎

1. 游戏引擎的概念

游戏引擎是指一些已编写好的可编辑电脑游戏系统或者一些交互式实时图像应用程序的核心组件。这些系统为游戏设计者提供编写游戏所需的各种工具,目的在于让游戏设计者能容易和快速地做出游戏程式而不用从零开始。简单地说,游戏引擎就是“用于控制所有游戏功能的主程序”,从计算碰撞、物理系统和物体的相对位置到接受玩家的输入以及声音的输出等功能,都是游戏引擎需要负责的事情。它扮演着中场发动机的角色,把游戏中的所有元素捆绑在一起,在后台指挥它们有序地工作。无论 2D 游戏还是 3D 游戏,无论角色扮演类游戏、即时策略游戏、冒险解谜游戏还是动作射击游戏,哪怕是再小的一个游戏,都是由一段控制代码来控制的。经过不断地进化,如今的游戏引擎已经发展为由多个子系统共同构成的复杂系统,从建模、动画到光影、粒子特效,从物理系统、碰撞检测到文件管理、网络特性,还有专业的编辑工具和插件,几乎涵盖了开发过程中的所有重要环节。游戏引擎大部分都支持多种操作平台,如 Linux、Mac OSX、微软 Windows。一个典型的游戏引擎一般包含以下几个组件:

(1)光影计算。它决定了场景中的光源对处于其中的人和物的影响方式。游戏的光影效果完全是由引擎控制的,折射、反射等基本的光学原理以及动态光源、彩色光源等高级效果都是通过引擎技术实现的。

(2)动画技术。目前游戏所采用的动画系统可以分为两种:一种是骨骼动画系统,另一种是模型动画系统。前者用内置的骨骼带动物体产生运动,比较常见;后者则是在模型的基础上直接进行变形。

(3)物理系统。它可以使游戏世界中的物体运动遵循客观世界规律。例如,当游戏人物跳起的时候,系统内定的重力值将决定他能跳多高,以及他下落的速度有多快,子弹的飞行轨迹、车辆的颠簸方式等也都是由物理系统决定的。另外,碰撞检测也是物理系统的另一个核心部分,它可以检测游戏中各物体的动态交互情况。

(4)实时渲染。它是引擎最重要的功能之一,也是引擎所有部件中最复杂的,它的强大与否直接决定着最终的游戏画面质量。

(5)人机交互。它负责玩家与电脑之间的沟通,处理来自键盘、摇杆和其他外设的信号。

(6)网络接口。对网络游戏引擎来说,网络代码则会被集成在引擎中,用于客户端与服务器之间通信的管理。

游戏引擎相当于游戏的框架,只要搭好了这个框架,剩下的工作就是由关卡设计师、建模师、动画师往里填充内容了。

2. 世界游戏引擎发展概况

曾经有一段时期,开发游戏的技术人员关注的重点仅仅是怎样尽量多地开发出新的游戏并将它们推销给玩家。虽然那个时候的游戏大多是一些简单粗糙的类型,但是每款游戏的平均开发周期也都在8~10个月以上,产生这种情况一方面是技术的原因,另一方面则是因为几乎每款游戏的代码都要从头编写,这就会造成大量的重复劳动。渐渐地,一些有经验的开发者摸索出了一种偷懒的方法,他们借用上一款类似题材游戏中的部分代码作为新游戏的基本框架,以节省开发时间和开发费用。这样,游戏引擎的雏形就初步显现出来了。

现在,已经找不到每开发一款游戏就从头编写一遍代码的开发商了,在游戏商业化的驱使下,游戏开发的周期成了开发商关注的主要问题,而游戏引擎恰恰能大大缩短游戏的开发周期,几乎每款游戏都有自己的引擎,但真正的优秀引擎屈指可数。好的游戏引擎和游戏开发之间的关系就相当于性能出色的发动机和赛车之间的关系,只有在好的游戏引擎的基础上,开发商才能开发出惊世的游戏大作。纵观世界上游戏引擎十多年的发展历程,我们会发现游戏引擎最大的驱动力来自3D游戏,尤其是3D射击游戏,动作射击游戏和3D游戏引擎之间的关系就像是战略合作伙伴一样,它们相互促进,相互发展。接下来将围绕动作射击游戏的变迁来对引擎历史进行回顾。

1)引擎的诞生(1992~1993年)

1992年,3D Realms公司/Apogee公司发布了一款只有2M多的3D小游戏——《德军司令部》(Wolfenstein 3D),稍有资历的玩家可能都还记得初接触它时的兴奋心情,用“革命”这一极富煽动色彩的词语也无法形容它在整个电脑游戏发展史上占据的重要地位。这部游戏开创了第一人称射击游戏的先河,更重要的是,它在X轴和Y轴的基础上增加了一根Z轴,在由宽度和高度构成的平面上增加了一个向前向后的纵深空间,这根Z轴对那些看惯了2D游戏的玩家造成的巨大视觉冲击可想而知。

Wolfenstein 3D引擎的作者是鼎鼎有名的约翰·卡马克,这位Id Software公司的首席程序师正是凭借这款Wolfenstein 3D引擎在游戏圈里站稳了脚跟。事实上,《德军司令部》并非第一款采用第一人称视角的游戏,在它发售前的几个月,

Origin 公司就已经推出了一款第一人称视角的角色扮演游戏——《创世纪：地下世界》(Ultima Underworld)，这款视角游戏采用了类似的技术，但它与 Wolfenstein 3D 引擎之间有着相当大的差别。例如，《地下世界》的引擎支持斜坡，地板和天花板可以有不同的高度，分出不同的层次，玩家可以在游戏中跳跃，可以抬头、低头，这些特性 Wolfenstein 3D 引擎都无法做到，而且从画面上看，《德军司令部》更接近漫画风格而不是传统的像素画面。

尽管从技术细节上看，Wolfenstein 3D 引擎比不上《创世纪：地下世界》的引擎，但它却更好地利用了第一人称视角的特点，快速火爆的游戏节奏使人们一下子记住了“第一人称射击游戏”这个单词，而不是“第一人称角色扮演游戏”。《德军司令部》后来还发布过一款名义上的续集——《三元的崛起》(Rise of the Triad)，这款游戏在 Wolfenstein 3D 引擎的基础上增加了许多重要特性，包括跳跃和抬头低头等动作。

引擎诞生初期的另一部重要游戏同样是出自 Id Software 公司的一款非常成功的第一人称射击游戏——《毁灭战士》(Doom)。Doom 引擎在技术上大大超越了 Wolfenstein 3D 引擎，《德军司令部》中的所有物体大小都是固定的，所有路径之间的角度都是直角，也就是说你只能笔直地前进或后退，这些局限在《毁灭战士》中都得到了突破。尽管游戏的关卡还是维持在 2D 平面上进行制作，没有“楼上楼”的概念，但墙壁的厚度可以为任意，并且路径之间的角度也可以为任意，这使得楼梯、升降平台、塔楼和户外等各种场景成为可能。

由于 Doom 引擎本质上依然是二维的，因此可以做到同时在屏幕上显示大量角色而不影响游戏的运行速度，这一特点为游戏创造出了一种疯狂刺激的动作风格，迄今为止，在这方面大约只有《英雄萨姆》(Serious Sam)系列能与之相比，除此之外，还没有哪款 3D 引擎能在大批敌人向你涌来的时候依然保持游戏的流畅，这也是为什么如今市面上的大部分第一人称射击游戏都在积极地培养玩家的战术运用能力、提高玩家的射击准确率而拒绝滥砍滥杀的主要原因之一。值得一提的是，尽管 Doom 引擎缺乏足够的细节度，但开发者仍然在《毁灭战士》中表现出了惊人的环境效果，其纯熟的设计技巧实在令人赞叹。

不过更值得纪念的是，Doom 引擎是第一个被用于授权的引擎。1993 年底，Raven 公司采用改进后的 Doom 引擎开发了一款名为《投影者》(Shadow Caster)的游戏，这是游戏史上第一例成功的嫁接手术。1994 年，Raven 公司采用 Doom 引擎开发《异教徒》(Heretic)，为引擎增加了飞行的特性，成为跳跃动作的前身。1995 年，Raven 公司采用 Doom 引擎开发《毁灭巫师》(Hexen)，加入了新的音效技术、脚本技术以及一种类似集线器的关卡设计，使玩家可以在不同关卡之间自由移动。Raven 公司与 Id Software 公司之间的一系列合作充分说明，引擎的授权无论对使用者还是开发者来说都是大有裨益的，只有把自己的引擎交给更多的人去使

用,才能使引擎不断地成熟起来。

《毁灭战士》系列本身就相当成功,大约卖了 350 万套,而授权费又为 Id Software 公司带来了一笔可观的收入。在此之前,引擎只是作为一种自产自销的开发工具,从来没有哪家游戏商考虑过依靠引擎赚钱,Doom 引擎的成功无疑为人们打开了一片新的市场。

2)引擎的转变(1994~1997 年)

在引擎的进化过程中,1994 年肯·西尔弗曼为 3D Realms 公司开发的 Build 引擎是一个重要的里程碑,Build 引擎的“肉身”就是那款家喻户晓的《毁灭公爵》(Duke Nukem 3D)。《毁灭公爵》已经具备了今天第一人称射击游戏的所有标准内容,如跳跃、360°环视以及下蹲和游泳等特性,此外还把《异教徒》里的飞行换成了喷气背包,甚至加入了角色缩小等令人耳目一新的内容。在 Build 引擎的基础上先后诞生过 14 款游戏,如《农夫也疯狂》(Redneck Rampage)、《阴影武士》(Shadow Warrior)和《血兆》(Blood)等,还有台湾艾生资讯开发的《七侠五义》,这是当时(至今依然是)国内不多的几款 3D 射击游戏之一。Build 引擎的授权业务为 3D Realms 公司带来了约一百多万美元的额外收入,3D Realms 公司也由此而成为了引擎授权市场上的第一个“暴发户”。不过从总体来看,Build 引擎并没有为 3D 引擎的发展带来任何质的变化,突破的任务最终由 Id Software 公司的《雷神之锤》(Quake)完成。

《雷神之锤》紧跟在《毁灭公爵》之后发售,一时之间,两者孰优孰劣成为玩家的热门话题。从内容的精彩程度来看,《毁灭公爵》超过《雷神之锤》不少,但从技术的先进与否来看,《雷神之锤》是毫无疑问的赢家。Quake 引擎是当时第一款完全支持多边形模型、动画和粒子特效的真正意义上的 3D 引擎,而不是 Doom、Build 那样的 2.5D 引擎。此外,Quake 引擎还是连线游戏最先使用的引擎,尽管几年前的《毁灭战士》也能通过调制解调器连线对战,但最终把网络游戏带入大众视野之中的是《雷神之锤》,是它促成了电子竞技产业的发展。

一年之后,Id Software 公司推出《雷神之锤 2》,一举确定了自己在 3D 引擎市场上的霸主地位。《雷神之锤 2》采用了一套全新的引擎,可以更充分地利用 3D 加速和 Open GL 技术,在图像和网络方面与前作相比有了质的飞跃,Raven 公司的《异教徒 2》(Heretic II)和《军事冒险家》(Soldier of Fortune)、Ritual 公司的《原罪》(Sin)、Xatrix 娱乐公司的《首脑:犯罪生涯》(Kingpin:Life of Crime)以及离子风暴工作室去年夏天刚刚发布的《安纳克朗诺克斯》(Anachronox)都采用了 Quake II 引擎。

Quake II 引擎的授权模式大致如下:基本许可费从 40 万美元到 100 万美元不等,版税金视基本许可费的多少而定,40 万美元的许可费大约需提取 10%以上的版税金,100 万美元的许可费则提取很少一部分版税金。这样算下来,《雷神之锤 2》通

过引擎授权所获得的收入至少有 1000 万美元,尽管游戏本身的销售业绩比起《毁灭战士》来要差很多,大约卖了 110 多万套,收入在 4500 万美元左右,但在授权金这一块它所获得的盈利显然要远远高于《毁灭战士》。

俗话说“一个巴掌拍不响”,没有实力相当的竞争者,任何市场都是无法发展起来的。正当 Quake II 独霸整个引擎市场的时候,Epic Megagames 公司(现在的 Epic 游戏公司)的《虚幻》(Unreal)问世了。可以这么说,第一次运行这款游戏的时候,很多人会被眼前的画面所惊呆,尽管当时只是在 300×200 的分辨率下运行的(四大悲事之一:玩游戏机器不够劲)。除了精致的建筑物外,游戏中的许多特效即便在今天看来依然很出色,荡漾的水波,美丽的天空,庞大的关卡,逼真的火焰、烟雾和力场等效果。从单纯的画面效果来看,《虚幻》是当之无愧的佼佼者,其震撼力完全可以与人们第一次见到《德军司令部》时的感受相比。

Unreal 引擎可能是使用最广的一款引擎,在推出后的两年之内就有 18 款游戏与 Epic 公司签订了许可协议,这还不包括 Epic 公司自己开发的《虚幻》资料片《重返纳帕利》,其中,比较近的几部作品如第三人称动作游戏《北欧神符》(Rune)、角色扮演游戏《杀出重围》(Deus Ex)以及永不上市的第一人称射击游戏《永远的毁灭公爵》(Duke Nukem Forever),这些游戏都曾经或将要获得不少好评。

Unreal 引擎的应用范围不限于游戏制作,还涵盖了教育、建筑等其他领域。Digital Design 公司曾与联合国教科文组织的世界文化遗产分部合作,采用 Unreal 引擎制作过巴黎圣母院的内部虚拟演示,Zen Tao 公司采用 Unreal 引擎为空手道选手制作过武术训练软件,另一家软件开发商 Vito Miliano 公司也采用 Unreal 引擎开发了一套名为“Unrealty”的建筑设计软件,用于房地产的演示。

这款与《雷神之锤 2》同时代的引擎经过不断地更新,至今依然活跃在游戏市场上,丝毫没有显出老迈的迹象,实属难得。

3)引擎的革命(1998~2000 年)

游戏的图像发展到《虚幻》已经达到了一个天花板的高度,接下去的发展方向很明显不可能再朝着视觉方面进行下去。前面说过,引擎技术对于游戏的作用并不仅局限于画面,它还影响到游戏的整体风格。例如,所有采用 Doom 引擎制作的游戏,无论《异教徒》还是《毁灭巫师》,都有着相似的内容,甚至连情节设定都如出一辙。玩家开始对端着枪跑来跑去的单调模式感到厌倦,开发者不得不从其他方面寻求突破,由此掀起了第一人称射击游戏的一个新的高潮。

两部划时代的作品同时出现在 1998 年——Valve 公司的《半条命》(Half-Life)和 Looking Glass 工作室的《神偷:暗黑计划》(Thief: The Dark Project),尽管此前的《系统震撼》(System Shock)等游戏也为引擎技术带来过许多新的特性,但没有哪款游戏能像《半条命》和《神偷》那样对后来的作品以及引擎技术的进化产生如此深远的影响。

曾获得无数大奖的《半条命》采用的是 Quake 和 Quake II 引擎的混合体, Valve 公司在这两部引擎的基础上加入了两个很重要的特性:一是脚本序列技术,这一技术可以令游戏以合乎情理的节奏通过触动事件的方式让玩家真实地体验到情节的发展,这对诞生以来就很少注重情节的第一人称射击游戏来说无疑是一次伟大的革命;二是对人工智能引擎的改进,与以往相比,敌人的行动明显有了更多的狡诈,不再是单纯地扑向枪口。这两个特点赋予了《半条命》引擎鲜明的个性,在此基础上诞生的《要塞小分队》《反恐精英》和《毁灭之日》等优秀作品又通过加入网络代码,使《半条命》引擎焕发出了更为夺目的光芒。

在人工智能方面真正取得突破的游戏是 Looking Glass 工作室的《神偷:暗黑计划》,游戏的故事发生在中古年代,玩家扮演一名盗贼,任务是进入不同的场所,在尽量不引起别人注意的情况下窃取物品。《神偷》采用的是 Looking Glass 工作室自行开发的 Dark 引擎,Dark 引擎在图像方面比不上《雷神之锤 2》或《虚幻》,但在人工智能方面它的水准却远远高于后两者,游戏中的敌人懂得根据声音辨认你的方位,能够分辨出不同地面上的脚步声,在不同的光照环境下有不同的目力,发现同伴的尸体后会进入警戒状态,还会针对你的行动做出各种合理的反应,你必须躲在暗处不被敌人发现才有可能完成任务,这在以往那些纯粹的杀戮游戏中是根本见不到的。如今的绝大部分第一人称射击游戏都或多或少地采用了这种隐秘的风格,包括新近发布的《荣誉勋章:盟军进攻》(*Medal of Honor: Allied Assault*)。遗憾的是,由于 Looking Glass 工作室的过早倒闭,Dark 引擎未能发扬光大,除了《神偷:暗黑计划》外,采用这一引擎的只有《神偷 2:金属时代》(*Thief 2: The Metal Age*)和《系统震撼 2》等少数几款游戏。

受《半条命》和《神偷:暗黑计划》两款游戏的启发,越来越多的开发者开始把注意力从单纯的视觉效果转向更具变化的游戏内容,其中比较值得一提的是离子风暴工作室出品的《杀出重围》,《杀出重围》采用的是 Unreal 引擎,尽管画面效果十分出众,但在个体的人工智能方面它无法达到《神偷》系列的水准,游戏中的敌人更多的是依靠预先设定的场景脚本做出反应,例如,砸碎弹药盒可能会引起附近敌人的警惕,但这并不代表他听到了什么,打死敌人后周围的同伙可能会朝你站立的位置奔过来也可能无动于衷,这些不真实的行为即便在《荣誉勋章:盟军进攻》里也依然存在。图像的品质抵消了人工智能方面的缺陷,而真正帮助《杀出重围》在众多射击游戏中脱颖而出的则是它的独特风格,游戏含有浓重的角色扮演成分,人物可以积累经验、提高技能,还有丰富的对话和曲折的情节。同《半条命》一样,《杀出重围》的成功说明了叙事对第一人称射击游戏的重要性,能否更好地支持游戏的叙事能力成为衡量引擎的一个新标准。

从 2000 年开始,3D 引擎朝着两个不同的方向分化,一是如《半条命》《神偷》和《杀出重围》那样通过融入更多的叙事成分和角色扮演成分以及加强游戏的人工智

能来提高游戏的可玩性,二是朝着纯粹的网络模式发展,在这一方面,Id Software 公司再次走到了整个行业的最前沿,他们意识到与人斗才是其乐无穷,于是在 Quake II 出色的图像引擎的基础上加入更多的网络成分,破天荒地推出了一款完全没有单人过关模式的纯粹的网络游戏——《雷神之锤 3 竞技场》(Quake III Arena),它与 Epic 公司稍后推出的《虚幻竞技场》(Unreal Tournament)一同成为引擎发展史上的一个转折点。

随着 Quake III 引擎的大获成功,Id Software 公司在引擎授权市场上也大赚了一笔。Raven 公司再次同 Id Software 公司合作,采用 Quake III 引擎制作了第一人称射击游戏《星际迷航:精英部队》(Star Trek Voyager: Elite Force),此外这部引擎还被用于制作第三人称动作游戏《重金属 F. A. K. K. 2》(Heavy Metal F. A. K. K. 2)和《艾丽丝漫游魔境》(American McGee's Alice)、最近的两款第二次世界大战题材的射击游戏《重返德军总部》(Return to Castle Wolfenstein)和《荣誉勋章:盟军进攻》,以及开发中的《绝地放逐者:绝地武士 2》(Jedi Outcast: Jedi Knight II)。从地牢到外太空,从童话世界到第二次世界大战,从第一人称视角到第三人称视角,充分显示了 Quake III 引擎的强大潜力。

Epic 公司的《虚幻竞技场》虽然比《雷神之锤 3 竞技场》落后了一步,但如果仔细比较一下,就会发现它的表现要略高后者一筹。从画面方面看,两者差不多打成平手,但在联网模式上,它不仅提供有死亡竞赛模式,还提供团队合作等多种激烈火爆的对战模式,而且 Unreal Tournament 引擎不仅可以应用在动作射击游戏中,还可以为大型多人游戏、即时策略游戏和角色扮演游戏提供强有力的 3D 支持。Unreal Tournament 引擎在许可业务方面的表现也超过了 Quake III,迄今为止,采用 Unreal Tournament 引擎制作的游戏已经有 20 多款,其中包括《星际迷航深度空间九:坠落》(Star Trek Deep Space Nine: The Fallen)、《新传说》(New Legend)和《塞拉菲姆》(Seraphim)等。

在 1998~2000 年迅速崛起的另一款引擎是 Monolith 公司的 LithTech 引擎,这款引擎最初是用在机甲射击游戏《升刚》(Shogo)上的。前面说过,LithTech 引擎的开发花了整整五年时间,耗资 700 万美元,经过多批技术人员的攻关,1998 年 LithTech 引擎的第一个版本推出之后立即引起了业界的注意,为当时处于白热化状态下的《雷神之锤 2》和《虚幻》之争泼了一盆冷水。

正是由于过于高昂的开发代价,2002 年,Monolith 公司决定单独成立一个 LithTech 公司,以 LithTech 引擎的授权许可作为主要业务,希望借此赚回一些成本。采用 LithTech 第一代引擎制作的游戏包括《血祭 2》(Blood 2)和《清醒》(Sanity)等。2000 年,LithTech 公司推出了引擎的 2.0 版本和 2.5 版本,加入了骨骼动画和高级地形系统,给人留下深刻印象的《无人永生》(No One Lives Forever)以及即将上市的《全球行动》(Global Operations)采用的就是 LithTech 2.5 引擎,此时