

《虚拟现实技术及其应用系列丛书》

Cryengine 使用宝典

(基础篇)

■ 许仁杰 编著



本书特色:

集资深游戏研发人员多年使用经验
以应用实例为线索,讲解使用技巧
以游戏开发为主线,奉献经验心得



请扫二维码,
索取配套资源。



国防工业出版社
National Defense Industry Press

虚拟现实技术及其应用系列丛书

Cryengine 使用宝典

(基础篇)

许仁杰 编著

国防工业出版社

· 北京 ·

内 容 简 介

本书主要介绍了 Cryengine3 游戏引擎的基本使用方法, 首先介绍了当今主流三维游戏引擎的功能特点, 然后结合实例详细阐述了 Cryengine3 游戏引擎中 SandBox 编辑器的使用, 并在这些基础上以实例说明了如何进行游戏环境编辑、游戏场景资源管理、游戏中的载具制作、声音设计, 如何使用 Flow Graph 流图和 LUA 脚本语言开发游戏运行逻辑。

图书在版编目(CIP)数据

Cryengine 使用宝典. 基础篇/许仁杰编著. —北京: 国防工业出版社,
2015.1

ISBN 978-7-118-09747-4

I. ①C… II. ①许… 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 (2014) 第 278350 号

※

国防工业出版社出版发行

(北京市海淀区紫竹院南路 23 号 邮政编码 100048)

北京奥鑫印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 17¼ 字数 390 千字

2015 年 1 月第 1 版第 1 次印刷 印数 1—3000 册 定价 79.00 元

(本书如有印装错误, 我社负责调换)

国防书店: (010) 88540777

发行邮购: (010) 88540776

发行传真: (010) 88540755

发行业务: (010) 88540717

编写人员

主 编 许仁杰

副主编 钟海波

编 写	杨学会	鲁鹤松	蒲 伟	姚凡凡
	杜 君	赵露华	许永晗	曹贻鹏
	许 飞	李立峰	刘俊红	杨 明

前 言

本书在分析了当前主流三维游戏引擎的基础上，详细介绍了 Cryengine 游戏引擎的特点，并针对该游戏引擎使用进行了系统的说明，结合作者在制作游戏过程中的心得，给出了更细致的讲解，并以案例的形式使读者了解 Cryengine 引擎的具体细节，本书结合 Cryengine 使用宝典（实战篇）使用效果更佳。

本书适用于普通高校数字媒体专业的本科生、研究生、教师，也适用于从事游戏行业的游戏设计人员，还可作为游戏学院等相关培训机构的游戏设计专业的培训教材。

本书主要由装甲兵工程学院的许仁杰、钟海波、杨学会、鲁鹤松、蒲伟、姚凡凡、杜君、赵露华、许永晗、曹贻鹏、许飞、李立峰、刘俊红和边防学院的杨明编写。读者在阅读过程中如对本书有意见和建议，欢迎来信交流，邮箱：jackyxu@bit.edu.cn。

在完成此书的过程中，编者参考了国内外相关游戏引擎的参考文档，在此对相关作者一并表示感谢。

编著者
2014 年 8 月

目 录

第1章 游戏引擎概述	1
1.1 游戏引擎的发展史	1
1.2 主流引擎介绍	2
1.2.1 ID 公司的 Quake/DOOM 游戏引擎	2
1.2.2 虚幻竞技场引擎 Unreal	5
1.2.3 Source 引擎	7
1.2.4 Cryengine 引擎	8
1.3 Cryengine 3 新特点	10
1.3.1 Cryengine 3 SandBox	10
1.3.2 实时动态光照	10
1.3.3 延迟光照	10
1.3.4 POLYBUMP 技术支持工具对应多核心与 64bit	10
1.3.5 屏幕空间环境光遮蔽	11
1.3.6 综合植被和地形覆盖生成系统	12
1.3.7 HDR 渲染	12
1.3.8 动态光照系统	13
1.3.9 高品质的水面效果	14
1.3.10 即时神圣光芒表现	15
1.3.11 Cryengine 3 角色表现动画系统	16
1.3.12 Cryengine 3 中整合的内置物理引擎	17
第2章 SandBox 入门	19
2.1 SandBox 初体验	19
2.1.1 在 SandBox 中打开一个场景	19
2.1.2 在场景中进行漫游观察	20
2.1.3 进入游戏模式	21
2.2 SandBox 的基本人机交互接口功能与操作	22
2.2.1 SandBox 人机交互接口概述	22
2.2.2 窗口交互与布局	33
2.2.3 菜单功能与操作	36
2.2.4 工具栏功能与操作	45
2.2.5 浮动工具栏 RollupBar 功能与操作	47

2.2.6	数据库窗口功能与操作	49
2.2.7	对象选择窗口功能与操作	50
2.3	SandBox 关卡创建基础	51
2.3.1	新建一个关卡	51
2.3.2	产生随机地形并修改	54
2.3.3	设置地形纹理	57
2.3.4	在场景中放置和编辑物体	60
2.3.5	使用数据层管理物体	62
第3章	环境编辑	63
3.1	地形环境	63
3.1.1	地形 Surface 创建	63
3.1.2	配置地形纹理图层	71
3.1.3	绘制地形纹理	74
3.1.4	植被	78
3.1.5	道路	83
3.1.6	河流	87
3.1.7	洞穴	91
3.2	天空环境	94
3.2.1	更换天空盒纹理	94
3.2.2	天气效果——云	97
3.2.3	天气效果——雨	101
3.2.4	天气效果——风	105
3.3	TOD 设置	107
3.4	制作 Mini 地图	107
第4章	资源管理	114
4.1	资源结构介绍	115
4.2	游戏主要资源结构	116
4.2.1	关卡文件结构	116
4.2.2	Pak 文件的创建方法	117
4.3	美术资源管理	117
4.3.1	美术资源的分类	118
4.3.2	美术资源管理工具	122
4.4	数据资源管理	126
4.4.1	DataBase View 界面及功能介绍	126
4.4.2	Archetype Entities 数据类	128
4.4.3	Prefabs 数据类	132
4.4.4	Vegetation 数据类	135

4.4.5	Particles 数据类	138
4.4.6	Music 数据类	140
4.4.7	Reverb Preset 数据类	142
4.4.8	SoundMoods 数据类	144
4.4.9	GameTokens 数据类	146
4.5	其他资源管理	147
4.5.1	Script 文件	147
4.5.2	Config 文件	147
4.5.3	Material Effects	147
第 5 章	使用流程图进行脚本开发	150
5.1	Flow Graph 介绍	150
5.1.1	Flow Graph 的组成	150
5.1.2	Flow Graph 编辑器	155
5.1.3	Flow Graph 基本操作	159
5.2	Flow Graph 脚本开发实践	163
5.2.1	Flow Graph 脚本设计	163
5.2.2	Flow Graph 脚本调试	167
5.2.3	小结	169
第 6 章	使用 Lua 语言进行脚本开发	170
6.1	Lua 语言介绍	170
6.2	Lua 的基本语法	171
6.2.1	注释	171
6.2.2	语句	171
6.2.3	关键字	173
6.2.4	变量类型	173
6.2.5	变量定义	174
6.2.6	小结	178
6.3	Lua 脚本开发实践	178
6.3.1	创建一个 Entity 类	178
6.3.2	在脚本中使用实体插槽和状态	181
6.3.3	创建一个实体流节点	182
6.3.4	使用 Lua 的 xml 加载器	184
6.4	Lua 脚本调试	187
6.4.1	调试器的启用	188
6.4.2	断点的使用	188

第7章 载具制作	189
7.1 3d Max 中载具制作	189
7.1.1 载具的结构剖析	189
7.1.2 在 Sandbox 中的设置	190
7.1.3 3d Max 中载具的导出选项	193
7.1.4 相关的调试	194
7.2 坦克的制作	194
7.3 直升机的制作	196
7.3.1 螺旋桨的制作	196
7.3.2 挂载武器	196
7.4 载具附件的制作	197
7.4.1 可破坏车窗	197
7.4.2 转速表与速度表	197
7.4.3 车灯以及排气孔	197
7.5 车辆编辑器	198
7.5.1 载具的编辑	198
7.5.2 新载具的创建	198
7.5.3 载具编辑器的组成	199
7.5.4 运动参数的编辑	200
7.5.5 结构的编辑	201
7.5.6 轮子的编辑	202
7.5.7 辅助点的编辑	202
7.5.8 座位的编辑	203
7.5.9 武器的编辑	204
7.5.10 毁坏部件的编辑	205
7.6 特效编辑	207
7.6.1 排烟的粒子效果	207
7.6.2 毁坏特效	208
第8章 流图编辑器	210
8.1 流图编辑器	210
8.1.1 流图编辑器的界面	210
8.1.2 流程图节点合成	213
8.1.3 添加编辑节点	214
8.1.4 导入/导出	215
8.2 流程图节点分类	217
8.2.1 Entity Nodes 实体节点	217
8.2.2 Game Nodes 游戏节点	222

8.2.3	HUD Nodes HUD 节点	226
8.2.4	Image Nodes 图像节点	235
8.3	流图实例	243
8.3.1	行进的流图控制	243
8.3.2	屏幕效果的流图控制	244
8.3.3	触发的流图控制	246
第9章	声音制作	248
9.1	声音制作简介	248
9.1.1	了解 Cryengine 声音事件系统	248
9.2	创建一个环境声音事件	249
9.3	声音制作实例	253
9.3.1	为武器添加声音	253
9.3.2	为粒子效果创建声音	254
9.3.3	声音与物理系统	256
9.3.4	为载具添加声音	259
参考文献		263

第1章 游戏引擎概述

1.1 游戏引擎的发展史

早期的游戏开发没有游戏引擎这个概念，通常一款游戏开发周期在8~10个月，都是一个游戏一套代码，随着游戏越来越多，开发有些游戏时，大家发现有很多可重用的代码，逐渐把这些重用代码封装起来，这样就可以大大减少游戏开发周期和开发费用，这就是早期的引擎。

现今的游戏引擎基本包括图形渲染引擎、声音引擎、网络引擎、脚本引擎、GUI、人工智能引擎、物理引擎等核心功能模块，为了方便游戏制作者开发游戏，还会有一系列的场景编辑器、角色编辑器、动画编辑器、脚本编辑器、资源编辑器、第三方资源转换工具、调试工具等相配合的开发工具组件等。

经过十多年的发展，游戏引擎的功能也越来越强大，涌现出一批比较知名的引擎（如Quake/DOOM、Unreal 虚幻、Cryengine），那么这些引擎的成功之处在哪里呢？什么才是好的游戏引擎？

时至今日，游戏引擎已从早期游戏开发的附属变为今日的当家角色，对于一款游戏来说，能实现什么样的效果，很大程度上取决于使用的引擎有多么强力。优秀的游戏引擎通常具有如下优点：

1. 完整的游戏功能

随着游戏要求的提高，现在的游戏引擎不再是一个简单的3D图形引擎，而是涵盖3D图形、音效处理、AI运算、物理碰撞等游戏中的各个组件，组件设计也应该是模块化的，可以按需购买。以这两年最成功的虚幻3引擎为例，虽然全部授权金（不包括售后技术服务）高达几十万甚至上百万美元，但是可以分别购买相关组件，降低授权费用。

2. 强大的编辑器和第三方插件

优秀的游戏引擎还要具备强大的编辑器，包括场景编辑、模型编辑、动画编辑等，编辑器的功能越强大，美工人员可发挥的余地就越大，做出的效果也越多。而插件的存在，使得第三方软件（如3ds Max、Maya）可以与引擎对接，无缝实现模型的导入、导出。

3. 简洁有效的SDK接口

优秀的引擎会把复杂的图像算法封装在模块内部，对外提供的则是简洁有效的SDK接口，有助于游戏开发人员迅速上手，这一点就如各种编程语言一样，越高级的语言越容易使用。

4. 其他辅助支持

优秀的游戏引擎还提供网络、数据库、脚本等功能,这一点对于面向网游的引擎来说更为重要,网游还要考虑服务器端的状况,要在保证优异画质的同时降低服务器端的极高压力。

以上4条对于今天的游戏引擎来说不成问题,当我们回头历数过去的游戏引擎,便会发现这些功能也都是从无到有慢慢发展起来的,早期的游戏引擎在今天看来已经没有什么优势,但是正是这些先行者推动了当今的发展。

1.2 主流引擎介绍

1.2.1 ID 公司的 Quake/DOOM 游戏引擎

谈及游戏引擎,有一家游戏公司最应该被提起,也有一个人最不应该忘记,那就是 ID Software (以下简称 ID 公司)和它的创始人之一约翰·卡马克(John D. Carmack)。据说这家公司仅有13名固定员工,而且停车场也只停13辆法拉利跑车,从这一“八卦新闻”中可以看出大家都很关注这家公司是如何的盛产富翁,但是这群“狂人”开发的游戏技术更为傲人。1990年,卡马克和 ID 公司的另一巨头罗梅洛制作了一款小游戏《Commander Keen》(《指挥官基恩》),在 PC 机上首次实现了卷轴类游戏背景的流畅效果,后来他们又花了一个晚上的时间把当时的街机游戏《超级玛丽》移植到 PC 机上,实现了流畅的横板效果。1991年,卡马克和罗梅洛成立了 ID Software 公司,开始自主创业。凭借着过人的技术和狂热的信念, ID 公司创造了一个又一个技术巅峰。没多久,他们推出了《Wolfenstein 3D》(《德军总部 3D》)游戏。这款游戏的画面在现在看来不值一提,但是却是 3D 射击游戏之祖,卡马克也因此获得了“FPS 游戏之父”的称号。

1993 年 ID 公司推出的《DOOM》(《毁灭战士》)游戏引擎是引擎技术的代表,如图 1-1 所示。DOOM 引擎改善了《Wolfenstein 3D》引擎中的一些缺陷,如所有房间的高度都是相同的,所有墙壁都是垂直的,运动也只能直线前进或后退等,并且加入了许多新效果。

在游戏中,角色与游戏中的物品间互动性进一步增强,楼梯、楼梯上的花朵以及路桥已经可升降。游戏中的光照效果也不再单一,不同的单位也有了不同的亮度,此外 DOOM 引擎还支持立体声音效,环境的定位感更真实。这些效果在 3D 显卡都没有问世的 1993 年毫无疑问是非常激动人心的进步。

凭借自身的过人之处,《DOOM》游戏获得了 350 万的销量,为 ID 公司带来了滚滚财源。更重要的是,DOOM 引擎成为 ID 公司第一款用于商业授权的引擎,从《DOOM》

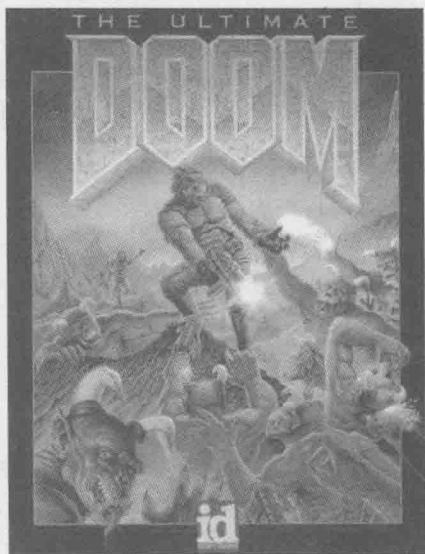


图 1-1 ID 的引擎渲染效果

游戏开始，游戏公司又多了一条创收之路，DOOM 发售一年之后，ID 又推出了 DOOM 系列的第二部作品《DOOM: Hell on Earth》。凭借 ID 的号召和一代的影响力，DOOM II 获得了 1994 年度的最佳原创幻想/科幻电脑游戏大奖，在销量和口碑上再创新高。

就在 DOOM 系列热火朝天的时候，ID 公司又于 1996 年发售了另一款新型游戏《Quake》，相比《DOOM》游戏使用的将拥有高度信息的二维地图渲染成 3D 图像的伪 3D 引擎，《Quake》使用的 Quake 引擎是实实在在的 3D 引擎，Quake 引擎是当时第一款完全支持多边形模型、动画和粒子特效的引擎，而在技术之外，《Quake》游戏的操控方式也树立了 FPS 游戏的标准。游戏采用流动控制方案（Fluid Control Scheme），它使用鼠标来观看/瞄准/定向以及用键盘前进/后退/侧移，这也成了 FPS 游戏最普遍的操控模式，直到今日仍没有变化。Quake 的出现为 FPS 游戏建立了经典的操控方式。

使用 Quake 引擎的主要是 ID 自己的《Quake》和《Quake World》系列、《HeXen II》（《异教徒》）、《MageSlayer》、《Nexuiz》（《鏖战废土》）、《Day of Defeat》（《胜利之日》），还有大名鼎鼎的《Half life》（《半条命》）以及《Half life》扩展出来的《反恐精英（Counter Strike, CS）》。大名鼎鼎的“CS”游戏使用的便是 Quake 引擎，《Quake》发布一年之后，ID 公司又发售了《Quake II》，如图 1-2 所示，采用的引擎也升级为 ID Tech2。



图 1-2 Quake II

1997 年，3D 加速卡（当时显卡称为 3D 加速卡）已经兴起，ID Tech2 引擎已经支持硬件 3D 加速，当时较为知名的 3D API 是 OpenGL，ID Tech2 也因此重点优化了 OpenGL 性能，这也奠定了 ID 公司系列游戏多为 OpenGL 渲染的基础。

ID Tech2 引擎添加了 256 色材质贴图，首次实现了彩色光影效果，使得游戏中的物体在反射光影时能展现出不同的色彩效果，另外一个技术特性则是通过支持 DLL（Dynamic-Link Libraries，动态链接库）文件实现了对软件渲染和 OpenGL 渲染的支持。基于 ID Tech2 引擎的游戏有《Quake II》、《Anachronox》（《时空传奇》）、《大刀》（ID 另一强人罗梅洛离开 ID 公司之后开发的游戏）、《Soldier of Fortune》（《命运战士》）、《半条命》（Quake 和 Quake II 的代码都有）。

1999 年，ID 公司又发布了《Quake III: Arena》，这是《Quake》系列的第三步，

也标志着 ID 的引擎技术到了 ID Tech3 时代。在这一时代, ID Tech3 已经不再支持软件渲染,必须要有一款支持硬件加速的显卡才能运行。ID Tech3 引擎增加了 32bit 材质的支持,还直接支持高细节模型和动态光影。同时,引擎在地图中的各种材质、模型上,都表现出了极好的真实光线效果。《Quake III》游戏中使用了革命性的 .MD3 格式的人物模型,模型的采光使用了顶点光影(Vertex Animation)技术,每一个人物都被分为不同段(头、身体等),并由玩家在游戏移动而改变实际的造型,游戏中真实感更强烈。

ID Tech3 引擎拥有游戏内命令行的方式,几乎所有使用这款引擎的游戏都可以用键调出游戏命令行界面,通过指令的形式对游戏进行修改,增强了引擎的灵活性,它是一款十分优秀的游戏引擎,即使是放到今天来讲,这款引擎仍有可取之处,即使画质可能不是一流的了,但是其优秀的移植性、易用性和灵活性使得它作为游戏引擎仍能发挥余热,使用 ID Tech3 引擎的游戏数量众多,比如早期的《使命召唤》系列、《荣誉勋章》、《绝地武士 2》、《星球大战》、《佣兵战场 2》、《007》、《重返德军总部 2》等。第一代的《使命召唤》使用了 ID Tech3 引擎。

时间很快就到 2004 年,这一时代已经是 DX9 显卡的时代,但是人们关注的年度 3D 大作仍是基于 OpenGL 渲染的《DOOM3》,如图 1-3 所示,这也是 ID 公司的第四代引擎——ID Tech4 的代表作。



图 1-3 DOOM3 画面

当时的《DOOM3》到底强到什么程度? ID 公司自家人恐怕最清楚,《DOOM3》的首席程序员罗布·达菲曾经说过:“DOOM3 从任何角度而言都是最完美的,而且它几乎完全追随并引导了硬件的发展。以 ID 公司的经验来看,DOOM3 估计又会被追捧至少 5 年,而在这 5 年里测试显卡 FPS 的基准估计都会是它了。”在《DOOM3》中,即时光影效果是主旋律,它不仅实现了静态光源下的即时光影,最重要的是通过 Shadow Volume(阴影锥)技术,如图 1-4 所示, ID Tech4 引擎实现了动态光源下的即时光影在游戏中大规模的使用。

当然了,除了 Shadow Volume(阴影锥)技术之外,《DOOM3》中的凹凸贴图、多边形、贴图、物理引擎和音效也都是非常出色的,2004 年《DOOM3》一出,当时的显卡市场可谓一片“哀嚎”, GeForce FX 5800/Radeon 9700 以下的显卡基本丧失了高画质

下流畅运行的能力，强悍能力也只有现在的《Crysis》能与之相比了。



图 1-4 Shadow Volume (阴影锥) 关闭 (左) 与开启 (右) 的效果对比

由于 ID Tech4 引擎的优秀，后续有一大批游戏都使用了这款引擎，包括《DOOM3》资料片《Resurrection of Evil》(《邪恶复苏》)、ID 公司自家的《Quake4》(实际是由 Raven 公司捉刀)、Human Head Studios 公司的《Prey》、Splash Damage 公司的《Enemy Territory: Quake Wars》(敌占区：雷神战争) 和《Brink》、Raven 公司的《Wolfenstein》(重返德军总部) 等。

现在，ID 公司还在进行着《DOOM4》的开发，引擎也升级到 ID Tech5，虽然详细的技术规格仍然是迷，但是卡马克已经放言：“DOOM4 是一款为 30 帧而拼搏的作品，就像是应用了全新的引擎，它基于我们这四年来的研发成果。”如果他指的是游戏发布时的硬件水平，那么 ID Tech5 引擎恐怕又是高端显卡的“坟墓”了。

1.2.2 虚幻竞技场引擎 Unreal

如果要找一个能跟 ID 公司的 Quake/DOOM 游戏引擎相比的对手，那么 EPIC 公司的 Unreal 引擎无疑是最合适的。而且相比只负责 3D 图像处理的 Quake/DOOM 引擎来讲，Unreal (虚幻) 涵盖的方面更多，涉及物理特性、动画演示、音频效果和碰撞检测等游戏的所有组件，也就是说 Unreal (虚幻) 引擎的集成度更高、通用性更强。1998 年 5 月 22 日，在《Quake II》发布后半年左右，由 Epic Games 开发 GT Interactive 发行的《Unreal》问世，游戏中除了精致的建筑物外，还拥有许多游戏特效，比如荡漾的水波，美丽的天空，逼真的火焰、烟雾和力场，单纯从画面效果来看，《Unreal》是当之无愧的佼佼者，如图 1-5 所示。

虽然在彩色光照效果上被 Quake II 引擎抢了先，但是 Unreal 引擎依然拥有自己的独门秘籍。Unreal 引擎在彩色光照和纹理过滤上的软件渲染性能已经接近硬件级的加速，而且 Unreal 引擎还支持当时 CPU 刚刚集成的 SIMD (单指令多数据) 指令，比如 AMD 新增的 3DNow，Intel 的 MMX 和 SSE 指令集，这让它的性能进一步增强。

Unreal 问世的时候，大行其道的图形接口还是 3Dfx 的 Glide，性能最强的显卡也是 3Dfx 的 Voodoo 5，微软的 DX 规范占据主流之后，Unreal 引擎又很快支持 DX，而对 OpenGL 规范虽然也有支持，但是一直不太顺，性能也不够好，所以 Unreal 引擎慢慢就变成了专司 DX 规范了。

得益于画面精美和 DX 规范的通用性，Unreal 很快就得到了 18 款游戏的支持，包括《Unreal》《Unreal Tournament》《哈利波特》《Rune》《Deus Ex》等，它们都用 Un-



图 1-5 最早的《Unreal》画面

real 引擎实现了不同的游戏效果,而且,由于 Unreal 引擎的通用性,除了游戏,它还广泛使用在 3D 建模、建筑设计、动作捕捉、电影特效等领域。

Unreal 引擎一炮打响,EPIC 又再接再厉推出了 Unreal2 引擎。相比第一代 Unreal 引擎,Unreal2 的全部代码几乎重写,并集成了最新的编辑器。随着游戏开发的需要,游戏引擎的内涵也在不断扩大,Unreal2 引擎拥有了更多的功能。

当时物理加速技术已然成形,Unreal2 引擎便集成了 Karma 物理加速技术开发包,引擎中的物理效果得以增强,并且开始支持 XBOX、PS2 等主机平台。Unreal2 引擎期间有过一次小幅升级,被称为 UE2.5,渲染性能有了一定提高。

有了第一代引擎的铺垫,采用 Unreal2 及其升级型引擎的游戏数量也大幅提高,其中的知名游戏包括《汤姆·克兰西之细胞分裂 2:明日潘多拉》《部落:复仇者》《越战英豪》《天堂 2》《杀手 13》《彩虹六号:雅典娜之剑》《荒野大镖客》《虚幻竞技场 2003》《手足兄弟连》等。

到了 DX9 时代,EPIC 公司的 Unreal2 引擎便跟不上时代了,急需一款性能更强大、更灵活的新时代引擎,这便是 EPIC 当前的主力干将——Unreal3 引擎,融合了众多新技术、新特性的 Unreal3 也是当前使用最广泛的引擎之一,在主机、单机、网游全面开花。

Unreal3 支持 64 位 HDRR 高精度动态渲染、多种类光照和高级动态阴影特效,可以在低多边形数量(通常在 5000~15000 多边形)的模型实现通常数百万个多边形模型才有的高渲染精度,这样就可以用最低的计算资源做到极高画质,Unreal3 引擎还提供了强大的编辑工具,让开发人员随意调用游戏对象,真正做到所见即所得。

Unreal3 引擎之中还集成了许多现成的游戏技术,包括 PhysX 物理引擎、SpeedTree 植被引擎、EAX5.0 音效引擎、AI 引擎等,其中的 PhysX 物理加速技术是 NVIDIA 公司力推的,在 PhysX 游戏中会拥有更好的效果。

优秀的画质表现、适中的显卡要求、强大的自定义工具和一站式配套开发使得 Unreal 具备了各个平台各种游戏的适应性,所以自问世以来,已经有越来越多的游戏采用 Unreal3 引擎,势头直追当前的 Quake/DOOM 引擎。

在采用 Unreal3 引擎的游戏中,《生化危机》《Alpha Protocol》《战争机器》《荣誉

勋章：空降神兵》《枪神》《质量效应》《镜之边缘》都是知名大作，尤其是《战争机器》一度创下 PC 游戏的画质神话。

2011 年，我国推出的第一款国产军事游戏《光荣使命》就是使用 Unreal3 引擎制作。

1.2.3 Source 引擎

Valve 公司的《Half Life》使用的是 Quake/Quake II 引擎，当他们开发续作《Half Life2》时，如图 1-6 所示，Quake 引擎已经略显老态，于是他们决定自己开发游戏引擎，这也成就了另一款知名的引擎——Source 引擎。



图 1-6 《Half Life2》系列

与 Unreal 引擎一样，Source 也不是一款简单的 3D 引擎，它同样包括了 3D 图像渲染、材质系统、AI 人工智能计算、Havok 物理引擎、游戏界面、游戏声效等各个组件，而且创造性地使用了模块化理念，结合 Valve 公司的 Stream 平台，游戏引擎的修改和升级都非常简单，这也使得 Source 引擎可以长盛不衰，至今仍是主流引擎之一。

Source 引擎最让人惊叹的特效当属丰富的人物表情。人的面部有 42 块肌肉，可以做出许多不同种类的表情，喜怒哀乐愁不一而足，要想在游戏中实现拟人化的表情绝非易事。为此，Valve 专门开发了一套面部肌肉模拟系统，这套系统中包含了一系列预先设定的表情脚本，能控制角色面部肌肉群产生相应的面部动画，并进行相加/混合/插值运算以创建现实生活中所见的自然真实的角色表情，如图 1-7 所示。



图 1-7 Source 引擎的面部表情