



普通高等教育
软件工程

“十二五”规划教材

12th Five-Year Plan Textbooks
of Software Engineering

游戏程序设计教程

——从游戏引擎构建到实际应用

万立中 万立新 ◎ 编著

*Windows Game Programming
with C++*



人民邮电出版社
POSTS & TELECOM PRESS

014013143

TP311.5-43
193



普通高等教育

软件工程

“十二五”规划教材

12th Five-Year Plan Textbooks
of Software Engineering

游戏程序设计教程

——从游戏引擎构建到实际应用

万立中 万立新 ◎ 编著



Windows Game Programming
with C++



北航

C1700437

人民邮电出版社

北京

图书在版编目(CIP)数据

游戏程序设计教程：从游戏引擎构建到实际应用 / 万立中, 万立新编著. — 北京：人民邮电出版社, 2014.2

普通高等教育软件工程“十二五”规划教材
ISBN 978-7-115-34258-4

I. ①游… II. ①万… ②万… III. ①游戏—程序设计—高等学校—教材 IV. ①TS952.83

中国版本图书馆CIP数据核字(2014)第003918号

内 容 提 要

本书从游戏编程实践的角度, 首先介绍了游戏编程要掌握的 Windows 编程和图形编程基础, 然后以 TinyEngine 游戏引擎构建为贯穿全书主线, 详细介绍了该引擎的完整创建过程, 最后以实际游戏案例应用所构建的游戏引擎结束。

全书分为三个部分, 第1~4章为第一部分, 该部分介绍游戏编程需要掌握的 Windows 编程基础和图形编程基础; 第5~10章为第二部分, 它是全书的核心, 该部分详细阐述了 TinyEngine 游戏引擎各组成模块的构建过程, 每章都有应用实例; 第11、12章为第三部分, 这部分介绍了两个完整的引擎应用案例游戏, 详细阐述了整个代码的实现过程。

本书适用于对 C++ 语言有基本了解的读者, 适合作为游戏程序设计课程理论与实践配套的教材, 也适合作为读者进入游戏开发领域的技术入门学习用书。

-
- ◆ 编 著 万立中 万立新
责任编辑 刘 博
责任印制 彭志环 杨林杰
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
大厂聚鑫印刷有限责任公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 21
字数: 554千字
-

定价: 54.00 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京崇工商广字第 0021 号

前 言

Windows 游戏程序设计, 需要较强的动手能力, 特别强调对理论知识的实际运用。实践证明, 要做到理论与实际相结合, 学习者必须在充分的理论指导下, 经过一个完整、系统的游戏编程训练才能掌握。本教材将游戏编程必须掌握的理论知识和游戏编程实训有机地结合在一起, 适合理论和实践紧密结合的教学需求。

全书从游戏编程实践的角度, 首先介绍了游戏编程必须掌握的游戏理论、Windows 编程基础、Windows 图形编程基础, 然后以 TinyEngine 游戏引擎的构建作为贯穿全书的主线, 详细地描述了该游戏引擎各组成模块的构建过程及理论。每个章节的知识点以自然、逐步进阶的方式安排。除导论部分外, 每章都有对应的案例, 各案例之间相互关联, 难度上不断加深, 知识点运用上逐步进阶。全部案例都有详细、完整的实现过程描述和相应的实现代码。最后通过编写完整的游戏程序来实际应用所构建的 TinyEngine 游戏引擎。

全书共 12 章, 分为三个部分: 第一部分为游戏编程基础篇, 涵盖第 1~4 章; 第二部分为游戏引擎构建篇, 涵盖第 5~10 章; 第三部分为游戏引擎应用篇, 涵盖第 11、12 章。其中第 2~10 章采用理论加实践的方式编写, 除第 1 章为纯理论外, 其余章节都涉及代码编写。第 2~10 章的理论知识可以作为理论授课内容, 与其对应的实例可以作为实践课的实训内容。第 11、12 章的内容, 可以作为实践课的综合实训内容。全书教学内容完成后, 学生就掌握了 TinyEngine 游戏引擎各模块的应用, 通过模仿第 11、12 章的内容, 完成其他游戏编程。

本书最大特点是从游戏引擎构建入手阐述游戏程序设计的相关理论和实践。全书内容不是教会编写几个独立、无关联的简单游戏, 而是让学习者领会从游戏引擎构建到实际游戏开发的实用技术, 强调系统性、完整性和实用性。

本书实际上作为教学讲义已经在游戏开发课程中使用多年, 早在 2009 年就完成全书初稿, 而且历经多次修订、完善。所有理论材料和案例代码都在教学实践中实际运用, 教学效果良好。全部内容可在 64 个教学课时内完成, 其中理论 34 个课时, 上机实践 30 个课时, 可以根据实际教学需要细化各章节的具体教学时分分配。本书配套教学资源可在人民邮电出版社教学服务与资源网 (www.ptpedu.com.cn) 或作者网站 (www.wanlizhong.com) 上免费下载。

本书在编写过程中得到了笔者家人的大力支持, 其中的一些图片及游戏素材处理由万昕完成。整个游戏引擎架构与代码实现、调试由笔者亲自完成。

本书的结构

本书共分 12 章, 其中第 1~4 章介绍游戏编程基础, 第 5~10 章介绍 TinyEngine 游戏引擎各组成模块的构建, 第 11、12 章是引擎案例游戏编程。

第 1 章 游戏编程导论——介绍计算机游戏的概念、游戏基本构成要素、游戏类型、游戏设计常用术语、游戏制作基本流程、游戏引擎的含义及引擎结构、

著名游戏引擎简介以及游戏编程语言等。

第2章 了解 Windows 编程——介绍 Windows 编程基础,如事件驱动及消息机制、窗口生成、Windows 程序创建及程序框架分析等,最后通过实例介绍 Windows 程序图标和光标的修改。本部分知识是学习后续所有章节的基础。

第3章 构建游戏引擎框架——介绍 TinyEngine 引擎框架设计、T_Config 头文件定义、屏幕分辨率改变及恢复、引擎类定义和实现,最后通过实例应用所构建的引擎类。

第4章 Windows 图形编程——介绍位图相关概念、Windows 图形技术简介、GDI 图形操作及绘图函数、GDI+图形操作,最后通过实例对本章内容进行综合应用。

第5章 为引擎添加位图操作模块——介绍位图操作类的设计与实现、利用位图操作类设计游戏菜单类,最后通过实例介绍游戏菜单类的应用。

第6章 理解动画与游戏角色——介绍游戏动画的相关概念、帧动画绘制、文字动画绘制、游戏角色的含义、角色信息描述,最后通过编程实例实现游戏角色绘制与控制。

第7章 为引擎添加游戏角色模块——介绍游戏图层的概念、游戏图层类的构建、游戏角色类的创建、AI 类的创建等,最后通过实例应用游戏角色类及 AI 类。

第8章 理解游戏地图——介绍游戏地图的概念、游戏地图制作、游戏地图类设计与实现、为角色类增加地图处理、为 AI 类增加地图处理等,最后通过综合实例进行实际应用。

第9章 为引擎添加游戏场景模块——介绍游戏场景模块需求分析、设计与实现,最后是游戏场景模块应用实例。

第10章 为引擎添加声音模块——介绍资源交换文件格式及操作、波形文件读取及播放、DirectSound 操作类的创建等,最后是声音模块应用实例。

第11章 引擎案例——贪食鱼游戏——介绍如何使用 TinyEngine 游戏引擎制作贪食鱼游戏,具体涉及需求分析、游戏素材设计、数据结构及数据类型定义、游戏类设计及代码实现以及游戏运行与发布等。

第12章 引擎案例——新坦克大战游戏——介绍如何使用 TinyEngine 游戏引擎制作新坦克大战游戏,具体涉及需求分析、游戏素材设计、数据结构及数据类型定义、游戏类设计及代码实现等。

对读者的要求

本书读者要求具备 C++ 程序语言方面的知识。尽管全书是 Windows 游戏程序设计,但是并不需要 Windows 编程背景知识,全书并未使用太过深入、复杂的 C++ 编程概念,只要基本了解 C++ 语言的读者就可以阅读。由于全书采用 Microsoft Visual Studio 2008 为代码实现的集成开发环境,因此要求读者具备基本的 Microsoft Visual Studio 2008 的使用知识。

由于编者水平有限,书中难免存在错漏之处,敬请读者批评指正。

编者

2013 年 11 月

目 录

第 1 章 游戏编程导论 1

1.1 游戏设计概述	1
1.1.1 计算机游戏的概念	1
1.1.2 游戏基本构成要素	3
1.1.3 游戏类型	5
1.1.4 游戏设计常用术语	12
1.2 游戏制作基本流程	14
1.2.1 前期阶段	15
1.2.2 制作阶段	15
1.2.3 后期阶段	16
1.3 关于游戏引擎	17
1.3.1 游戏引擎的含义	17
1.3.2 游戏引擎的结构	18
1.3.3 关于游戏工具	19
1.3.4 著名游戏引擎简介	20
1.4 游戏编程语言	21
1.5 学习的先决条件	23
1.6 几点相关说明	24

第 2 章 了解 Windows 编程 25

2.1 Visual C++ 编程方式	25
2.1.1 直接调用 Windows API 方式	25
2.1.2 采用 MFC 方式	26
2.1.3 Windows 程序设计基本概念	26
2.2 事件驱动及消息机制	28
2.2.1 Windows 中的事件驱动	29
2.2.2 Windows 应用程序中的常见消息	29
2.3 Windows 窗口的生成	32
2.3.1 指定窗口特征	32
2.3.2 注册窗口类并创建窗口	35

2.4 Windows 程序创建及程序

框架分析	36
------------	----

2.4.1 Windows 程序的创建 36

2.4.2 Windows 程序的基本 结构分析	39
-----------------------------------	----

2.5 修改程序图标和光标 42

2.5.1 图标资源的创建和加载 43

2.5.2 光标资源的创建和加载 44

第 3 章 构建游戏引擎框架 46

3.1 游戏引擎框架的构建 46

3.1.1 引擎框架的设计 46

3.1.2 T_Config 头文件的定义 49

3.1.3 屏幕分辨率的改变及恢复 52

3.1.4 引擎类的定义和实现 55

3.2 引擎类应用实例 63

3.2.1 编写引擎框架测试代码 63

3.2.2 为框架测试代码增加绘图 处理任务	65
---------------------------------	----

第 4 章 Windows 图形编程 67

4.1 与位图相关的概念 67

4.1.1 色彩模式 67

4.1.2 色彩深度 68

4.1.3 Alpha 通道 69

4.1.4 RGB 颜色的描述 69

4.2 Windows 图形技术 70

4.2.1 图形技术简介 70

4.2.2 图形设备接口 70

4.3 GDI 图形操作 73

4.3.1 位图与设备环境 73

4.3.2 设备相关的位图	74
4.3.3 设备无关的位图	76
4.4 GDI 绘图函数	82
4.4.1 BitBlt()函数及其应用	83
4.4.2 StretchBlt()函数及其应用	83
4.4.3 AlphaBlend()函数及其应用	84
4.5 GDI+图形操作	85
4.5.1 GDI+简介	85
4.5.2 Graphics 类	86
4.5.3 Image 类与 Bitmap 类	86
4.5.4 图像显示	89
4.5.5 图像旋转与翻转	89
4.5.6 文字显示	94
4.6 综合应用实例	95

第 5 章 为引擎添加位图操作模块 101

5.1 位图操作类的设计与实现	101
5.1.1 位图操作类需求分析	101
5.1.2 位图操作类的定义	102
5.1.3 位图操作类的实现	103
5.2 利用位图操作类设计游戏菜单类	108
5.2.1 游戏菜单类的需求分析	109
5.2.2 游戏菜单类的定义	110
5.2.3 游戏菜单类的实现	111
5.2.4 测试游戏菜单类	118
5.3 游戏菜单类应用实例	118

第 6 章 理解动画与游戏角色 124

6.1 理解游戏动画	124
6.1.1 帧动画与帧图像	124
6.1.2 帧序列	125
6.1.3 游戏 FPS 的设置	126
6.1.4 帧动画绘制实例	126
6.1.5 文字动画绘制实例	130
6.2 游戏中的角色	132

6.2.1 角色的含义	132
6.2.2 游戏角色的共性	133
6.2.3 角色信息的描述	134
6.3 多角色绘制与控制实例	135

第 7 章 为引擎添加游戏角色模块 143

7.1 关于图层的概念	143
7.2 构建游戏图层类	144
7.3 创建游戏角色类	145
7.3.1 游戏角色类的设计	145
7.3.2 游戏角色类的实现	148
7.4 为游戏角色类添加 AI 控制	157
7.4.1 AI 类的基本需求与设计	158
7.4.2 AI 类的实现	160
7.5 游戏角色类及 AI 类应用实例	166

第 8 章 理解游戏地图 173

8.1 关于游戏地图	173
8.2 游戏地图的制作	174
8.2.1 基于图块的游戏地图原理	174
8.2.2 游戏地图的制作	176
8.3 为引擎增加游戏地图类	181
8.4 为角色类增加地图处理	187
8.5 为 AI 类增加地图处理	190
8.6 地图应用综合实例	191

第 9 章 为引擎添加游戏场景模块 203

9.1 游戏场景模块需求分析	203
9.2 游戏场景模块的设计	204
9.3 游戏场景模块的实现	207
9.3.1 场景操作函数	207
9.3.2 图层操作函数	211
9.3.3 地图文件解析函数	214
9.3.4 场景绘制函数	217
9.4 游戏场景模块应用实例	218

第 10 章 为引擎添加声音模块.... 226

10.1 资源交换文件格式	226
10.2 资源交换文件的操作	227
10.2.1 打开与关闭	227
10.2.2 文件的读写	228
10.2.3 文件的定位	229
10.3 波形文件读取	230
10.3.1 波形文件结构体	230
10.3.2 编写波形文件读取类	231
10.4 波形文件播放	233
10.4.1 DirectSound 概述	234
10.4.2 在开发环境中设置 DirectX SDK	234
10.4.3 DirectSound 对象创建及 初始化	235
10.4.4 声音缓冲区的创建	237
10.4.5 缓冲区声音的播放与控制	238
10.5 DirectSound 操作类的创建	239
10.5.1 AudioDXBuffer 类的定义与 实现	240
10.5.2 AudioDX 类的定义与实现	244
10.6 为菜单类添加音效处理	245
10.7 声音模块应用实例	247
10.8 游戏引擎小结	251

第 11 章 引擎案例——**贪食鱼游戏 253**

11.1 游戏需求分析	253
11.2 游戏总体设计	254

11.2.1 游戏素材设计	255
11.2.2 数据结构及数据类型定义	257
11.2.3 游戏类设计	258
11.3 游戏代码实现	262
11.3.1 资源加载类函数	262
11.3.2 游戏状态更新类函数	269
11.3.3 关卡处理函数	275
11.3.4 游戏细节处理类函数	276
11.3.5 引擎接口函数实现	281
11.4 游戏运行与发布	286
11.4.1 游戏运行测试	286
11.4.2 游戏项目发布	288

第 12 章 引擎案例——**新坦克大战游戏 291**

12.1 游戏需求分析	291
12.2 游戏总体设计	292
12.2.1 游戏素材设计	293
12.2.2 游戏地图设计	295
12.2.3 游戏类设计	296
12.3 游戏代码实现	299
12.3.1 资源加载类函数	299
12.3.2 游戏状态更新类函数	307
12.3.3 关卡处理函数	313
12.3.4 游戏细节处理类函数	315
12.3.5 引擎接口函数实现	321
12.4 游戏运行测试	325

参考文献 328**参考网站 328**

第1章

游戏编程导论

游戏和电影一样，是一种娱乐方式。电影通过故事情节、视觉特效、音响效果及背景音乐等元素吸引观众，娱乐元素以线性方式呈现，受众只能被动接受而无法参与其中。游戏除包含这些娱乐元素外，还具有交互性，这是它与电影最大的不同，也是其魅力所在。正是这种交互性，使得玩家不是被动地欣赏“英雄”，而是成为“英雄”。通过交互，玩家成为故事的主角，并沉浸在游戏创设的世界中，利用自己对游戏世界的掌控能力战胜怪物、战胜困难或解决难题，以此获得虚拟的物质奖励，并不断晋级使自己更加强大，获得游戏带来的成就感和满足感。

那么游戏如何实现交互功能呢？当然是通过编程实现。正是程序控制，才使得游戏中的怪物显得很“聪明”；也正是因为程序控制，让玩家体会到下一关的怪物总是比上一关更难对付，要用更多“法力”或更强大的武器才能战胜。事实上，游戏角色的一举一动无不是程序控制的结果。随着游戏的发展，那些与游戏内容无关的游戏控制主程序渐渐被抽象化，并独立出来成为“游戏引擎”，而且功能越来越强大。利用游戏引擎制作游戏成为当前游戏界的标准方式。

能够开发游戏引擎的程序员总是大受欢迎。但对于游戏编程初学者而言，要具备引擎开发能力或读懂商业引擎代码，他起码要理解游戏引擎原理，了解游戏开发过程，并有过利用游戏引擎开发游戏程序的经历。本书正是以此需求为出发点，首先阐述如何构建一个小型游戏引擎，然后利用该引擎制作完整游戏。全书内容编排符合游戏制作常规，是您学习游戏编程的良好开端。下面让我们从认识游戏设计开始吧。

1.1 游戏设计概述

1.1.1 计算机游戏的概念

在 Katie Salen 和 Eric Zimmerman 合著的《玩的规则：游戏设计基础》(Rules of Play: Game Design Fundamentals)一书中，他们把游戏定义为：“游戏是一个系统，玩家参与由规则定义的人工冲突，产生可量化的结果 (A game is a system in which players engage in an artificial conflict, defined by rules, that results in a quantifiable outcome)”。

简而言之，游戏就是有规则地玩，通常出于娱乐目的，有时也用作教育工具。游戏具有特定行为模式、规则条件以及输赢胜负的量化判定结果。游戏参与者通过脑力或体力活动参与人为设计的冲突或难题的解决，并在游戏的过程中，体力、脑力或两者都得到锻炼，同时也体验到了游戏带来的乐趣。

而计算机游戏则是以计算机为操作平台,通过人机交互实现的一种游戏方式,它在本质上是一种计算机软件。与传统游戏不同的是,计算机游戏中可以采用计算机的运算替代传统游戏中必须由人承担的角色,这些角色依赖计算机的算法进行控制。而计算机游戏的参与者往往被称为玩家(Player)。那些非玩家控制的角色,也就是受计算机操控的角色,一般简称为 NPC(Non-Player Character, 非玩家角色)。

计算机游戏的发展与计算机技术的发展是相互促进的。更新的图形处理技术让游戏设计者可以构建更加复杂、更加接近现实的虚拟世界;更佳的交互性让游戏具有更强的沉浸感;越来越强大的计算能力让游戏角色的仿真度越来越高,尤其伴随着人工智能和算法研究的不断深入,游戏中的电脑角色可以让游戏玩家得到接近真人角色的体验,而游戏中的场景也与真实环境一样栩栩如生。图 1-1 所示是《NBA》游戏中的逼真动作,图 1-2 所示是游戏《刺客信条》中的逼真场景。

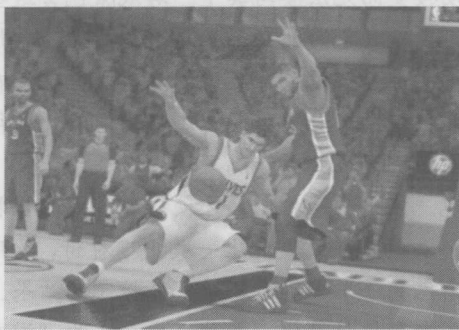


图 1-1 《NBA》游戏中的逼真动作



图 1-2 《刺客信条》游戏中的逼真场景

一些对脑力要求高的智力游戏,例如棋牌类游戏,得益于更强大的硬件和算法,电脑角色变得越来越难以战胜。1997 年 5 月 11 日,“深蓝”计算机打败俄罗斯国际象棋大师卡斯帕罗夫,让人们的人工智能有了新看法。同时,人们对更好游戏的期望也促使计算机技术不断向前发展。

计算机游戏不仅与计算机技术相关,同时它也是一种艺术形式。人们普遍认为绘画、雕刻、建筑、音乐、诗歌、舞蹈、戏剧、电影是人类历史上的 8 种艺术形式。而随着计算机游戏的普及和发展,有学者提出数字化游戏应该成为人类历史上的第 9 种艺术形式。因为计算机游戏和其他艺术形式一样,具有独特而丰富的表现力,可以创造鲜明生动的形象,甚至可以将多种艺术形式融为一体。例如运用动作捕捉技术就可以让游戏角色像真人一样翩翩起舞。图 1-3 所示是捕捉舞蹈动作的一个场景,这也是目前游戏设计中广泛使用的技术,利用这种技术可以创建更加逼真的游戏角色动画。图 1-4 所示是韩国一款利用虚幻 3 引擎设计的真人舞蹈网游 Mstar,游戏很好地将舞蹈、音乐以及电影和戏剧的手法有机地结合在一起。

计算机游戏最独特的魅力在于它能够让玩家成为虚拟世界的一员。玩家通过与计算机的交互,实现对虚拟游戏世界的掌控。玩家不同的控制行为和决策将直接影响到游戏结果。在游戏中,玩家会体验到胜利的成就感、失败的挫折感、虚拟物质带来的满足感,以及不断升级带来的希望体验,这是其他艺术形式无法同时满足的。

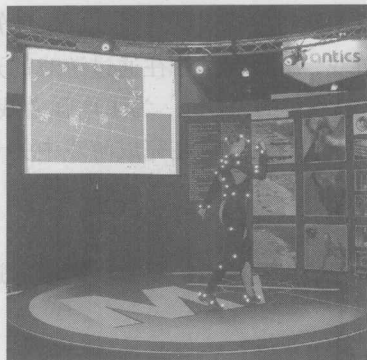


图 1-3 舞蹈动作捕捉



图 1-4 韩国虚幻3真人舞蹈网游 Mstar

1.1.2 游戏基本构成要素

综观各类游戏，无论简单或复杂，都具有既定的流程模式，游戏参与者必须遵守其中的约定来完成一个游戏任务，游戏最终都有输赢胜负的结果。由此可见，游戏具有共同的基本构成要素。

Marc Prensky 在其所著的《基于数字游戏的学习》(Digital Game-Based Learning)一书中，将计算机游戏总结为六个构成要素：规则(Rules)、目标与目的(Goals and Objectives)、结果和反馈(Outcomes and Feedback)、冲突/竞争/挑战/对抗(Conflict / Competition / Challenge / Opposition)、交互(Interaction)、表现或故事(Representation or Story)。

1. 规则(Rules)

正如前面所述，游戏就是有规则地玩。游戏中的全部活动都是建立在规则的基础之上，即便是简单的猜拳游戏或拔河游戏都是有规则的。游戏规则实际上是按照游戏逻辑设定的一系列限定条件，有时这些限定条件是保证游戏趣味性的的重要组成部分，例如用筷子夹乒乓球的趣味游戏，使用筷子就是最重要的限定条件，同时也是游戏的关键趣味性所在。

遵守游戏规则是游戏进行下去并确保游戏趣味性、公平性的基础。在传统游戏中，游戏规则必须事先规定好。例如玩跳房子的游戏，一个最重要的规则就是必须单脚跳，如果游戏的参与者不遵守这个事先的规定，游戏将失去乐趣。而在计算机游戏中，游戏规则已经被程序设计人员以代码的方式或脚本的方式整合到游戏中了，如果不遵守游戏规则，游戏将不能进行下去或无法进行操作。

2. 目标与目的(Goals and Objectives)

这两个词常常被混淆，两者都可以指一个人期望取得或实现的东西。目标一般指付出某种努力期望得到的最终结果；而目的则是指在总目标范围内要达到的具体的子目标，它往往是与时间相关的具体任务。因此，目标是导向性的，是努力的方向和最终想要达到的境地或标准。而目的则是要达到的具体指标或者特定结果，是清晰、可测量的。

对于游戏而言，游戏的目标就是游戏的最终结果，也是游戏全部活动的导向，同时还是激励游戏玩家参与游戏活动的动机。任何游戏都应该有明确的游戏目标，但不同的游戏对于实现目标的过程和采取的游戏策略各不相同，也就是游戏目的千差万别。游戏目的往往与游戏的竞争要素和游戏玩家需要采取的游戏行为和策略密切相关。为了达到游戏目标，游戏玩家必须遵守游戏规则，完成游戏设定的一系列从易到难的游戏任务，即游戏设定的关卡，并在此过程中采取恰当的游戏策略。

对于简单的游戏，如俄罗斯方块、连连看等，当玩家了解到游戏规则的时候，也就知道了游戏的目标和目的。而对于基于故事情节的 game，如 RPG 游戏，游戏玩家往往通过游戏的开头短片了解游戏背景和游戏目标，并通过游戏中的对话系统了解游戏目的。有时，为了增强游戏的悬念，激起玩家探索游戏世界的欲望，游戏的开头短片只给出部分信息，玩家在完成游戏中关卡任务的过程中逐渐了解到游戏目标。

3. 结果和反馈 (Outcomes and Feedback)

不管什么类型的 game 都要对玩家的游戏活动给出必须的结果和及时的反馈。玩家通过这些结果和反馈了解游戏进程和他们在游戏中的成长或进步情况。

典型的 game 结果往往是一个具体游戏子目标状态的获胜或失败。对于游戏玩家而言，无论是胜利或失败的结果都会带来强烈的情绪反应：胜利的结果往往给玩家带来自我满足的成功感，并激励游戏玩家继续玩下去的信心；而失败的结果往往给玩家一定的挫败感，并激起他们再次尝试和挑战的努力，以证明自己的能力。

游戏中的反馈必须是及时的，以便让游戏玩家能够马上得知自己的游戏行为是否正确，并指引他们朝着 game 设定的模式和目标行为。玩家通过反馈就可以知道如何在 game 中取得成功，如何完成游戏中的子目标，并最终取得胜利。

但 game 反馈的形式在不同的 game 中千差万别，许多 game 采用数字化的 game 成绩告知玩家 game 竞争中所处的当前状态，例如游戏中的积分、剩余的 game 生命数或已经消灭的敌人数等。也有的 game 采用图形、图像或音效来对玩家的行为做出反馈，例如遭到敌方的袭击，出现血点四溅的画面并伴随表示 game 生命血量的减少；当玩家得到不同的 game 装备，伴随不同的 game 音效等。可以说，game 反馈形式的设计直接关系到游戏玩家的直接体验。因此，反馈的形式要与 game 内容、game 类型及玩家习惯相适应。

4. 冲突/竞争/挑战/对抗 (Conflict / Competition / Challenge / Opposition)

game 的可玩性在于 game 中设计的竞争、挑战、冲突或对抗，它们在 game 设计中处于核心地位，也是 game 的灵魂所在。游戏中的规则、目标和目的以及结果和反馈都是围绕这些核心元素而设计的。

在不同的 game 中，竞争、挑战、冲突或对抗的表现是多种多样的，可以是难题的破解、障碍的逾越或与敌人面对面的厮杀等，它们构成了不同类型 game 的具体内容。对于益智类的 game 而言，游戏中的竞争、挑战主要以问题解决为主要内容；而 RPG 游戏中，冲突或对抗则是更常见的 game 内容。

游戏中竞争、挑战的对象可以是其他游戏玩家，也可以是通过 game 人工智能生成的对象。对于大型多人在线网络游戏而言，玩家挑战、竞争的对象主要是参与 game 的其他玩家；而对于单机版的 game，计算机控制的角色往往是玩家挑战、竞争的对象。

game 冲突和对抗设计的合理性可以作为衡量 game 好坏的标准。一款优秀的 game 往往在 game 冲突和对抗的强度、节奏、难易度等方面设计合理、恰当，符合心理学的原则。如果 game 冲突和对抗不能对玩家构成一定的挑战性，game 将失去趣味性和可玩性；而挑战难度太高的 game，玩家会由于屡受挫折，逐渐失去对 game 的兴趣和信心。游戏中的冲突和对抗必须做到张弛有度，既要让玩家能够保持适度的紧张感，又要让玩家体会到紧张情绪释放后的 game 愉悦感。

5. 交互 (Interaction)

交互性是 game 与其他艺术形式最大的区别。通过交互，玩家可以直接参与到 game 设计的冲突或对抗中，体验到驰骋沙场的感觉；通过交互，玩家可以在 game 中体现自己的意志和策略，体验

到自己的智力带来的成就感；通过交互，玩家可以掌控游戏世界，体会到一呼百应的领导力。

游戏中的交互主要表现在两个方面：一是玩家与计算机的交互；其次就是当前玩家与其他游戏玩家的交互。在单机游戏中，玩家与计算机的对抗是游戏的主要形式，所以游戏交互主要是玩家与计算机之间的。在这种环境下，游戏人工智能算法显得尤为重要，它将直接影响到玩家体验的真实感。而对于多人在线游戏，玩家与计算机的交互以及玩家与玩家之间的交互同时存在，游戏的交互设计也相对要复杂得多。

游戏交互设计是游戏操纵性的直接体现，它将直接影响到玩家对游戏的体验。随着计算机软硬件的发展，越来越多的新技术被用到游戏的交互设计中，例如多点触控交互技术、摄像头跟踪控制技术、手势交互技术等，让游戏交互设计的空间越来越大。

6. 表现或故事 (Representation or Story)

游戏中的竞争、挑战、冲突或对抗必须以某种方式表现出来，也就是说游戏的主题必须依托于一种表现形式。其实玩游戏就是从事游戏规定的某个具体的游戏活动，这个游戏活动可以很简单，也可以很复杂，可以一人参与，也可以多人同时参与。

对于简单的游戏，整个游戏活动可以只是完成一些简单的行为，例如完成某种模式的排列，像俄罗斯方块、五子棋或连连看这类益智游戏中的活动就是如此。但对于更大、更复杂的 RPG 游戏，游戏活动往往基于故事情节展开，整个游戏以一个交互式的故事脚本为线索，有着像电影一样的完整故事情节，包含故事发生的背景、故事情节的发展和故事的结局。游戏中的活动也更复杂多样，游戏中的角色行为往往以模拟真实的人物行为为主，例如跑、跳、行走、攀爬、武术动作等，这样的游戏可以给玩家带来更强的真实体验。

当然，游戏的表现形式毕竟是外在的，游戏的创意和内容决定了游戏的表现形式。但表现形式作为游戏的重要构成因素，对游戏的可玩性、趣味性和参与度有着直接的影响。同时，表现形式的复杂程度与游戏的趣味性和受欢迎程度不一定成正比。

1.1.3 游戏类型

计算机游戏可以从不同角度进行分类，例如从人机交互模式的角度，可以分为单机游戏和网络游戏；而从运行平台的角度，可以分为 PC 游戏、手机游戏、掌上游戏机游戏、街机游戏等。

最通用的分类方法是从游戏内容的性质进行分类的，这也是目前最广泛的分类方法，可以将计算机游戏大致分为：角色扮演游戏、策略游戏、动作游戏、格斗游戏、射击游戏、冒险游戏、体育游戏、竞速游戏、模拟游戏、牌类游戏、棋盘游戏、益智游戏、音乐游戏等。实际上，也有不少混合类型的游戏，它们不能单纯纳入某个类型。

1. 角色扮演游戏 (Role-playing Game, RPG)

这是最受欢迎的游戏类型之一。这类游戏会构建一个虚拟的社会环境并有相应的规则。游戏世界有各类角色、各种建筑，还有商店、迷宫及各种险峻的地形等。参与游戏的玩家可以在游戏中扮演一个角色，彼此可以互动，还可以在游戏世界中交谈、交易、打斗、旅行、探险及解谜等。每个玩家依靠自己的智慧、胆识、机敏获得一次又一次的成功，使自己扮演的主角在游戏中成长并变得越来越强大。这类游戏往往都有故事情节，玩家通过在游戏世界中不断探索，逐渐揭开游戏情节的线索，最终到达游戏的终点。

基于网络的 RPG 游戏是这类游戏中最受欢迎的，尤其是大型多人在线角色扮演游戏 (Massively Multiplayer Online Role Playing Game, MMORPG)，代表作有《传奇》、《奇迹》、《梦

幻西游》、《魔兽世界》(如图 1-5 所示)等。单机版 RPG 游戏的代表作有国内三剑——大宇的《仙剑奇侠传》系列、《轩辕剑》系列和西山居的《剑侠情缘》，以及欧美的《辐射》(如图 1-6 所示)系列、《上古卷轴》系列、《魔法门》系列等。



图 1-5 《魔兽世界》游戏截图

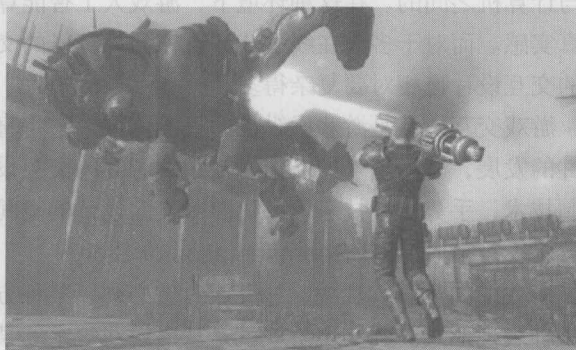


图 1-6 《辐射》游戏截图

2. 策略游戏(Strategy Games, SLG)

策略游戏也称为战略游戏。这类游戏中，决策对游戏结果将产生至关重要的影响，因此它要求游戏玩家要有一定的决策能力。根据决策的方式，可以分为即时战略游戏(Real-Time Strategy Game, RTS)、回合制战略游戏(Turn-based Strategy Game, TBS)以及塔防游戏(Tower Defend)。

在即时战略游戏中，游戏决策是即时的，也就是游戏在一个不间断的连续状态下，玩家可以在游戏进行中随时决策。这类游戏的代表作有《家园》、《罗马：全面战争》、《星际争霸》、《魔兽争霸 2》、《帝国时代》以及《红色警戒》等。

而回合制战略游戏则不同，游戏决策是采用轮流的方式。游戏中的各方(包括计算机在内)按照一定顺序分别部署战略。一次部署就称为一个回合。所以游戏是基于回合的，游戏中的各方要按照游戏规则轮流做出决策。只有当一方完成决策后，其他方才能进行决策。这类游戏的经典之作就是《文明》系列和《英雄无敌》(如图 1-7 所示)系列。

以上两种游戏的决策都是进攻型的，而塔防游戏的决策则是防守型的。游戏的内容基本都是如何防守玩家的家园或基地。玩家必须采取适当的决策来防御来犯的敌人，如在游戏地图上布置炮塔或其他可以消灭敌人的武器等。这类游戏的代表作有《植物大战僵尸》(如图 1-8 所示)、《王国保卫战》等。



图 1-7 《英雄无敌》游戏截图

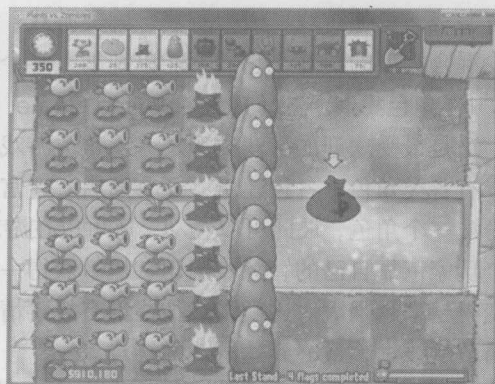


图 1-8 《植物大战僵尸》游戏截图

3. 动作游戏(Action Games, ACT)

动作游戏追求游戏角色形体动作的逼真性和良好的操纵手感,在游戏中往往会有火爆的打斗效果和复杂的攻击组合。这类游戏通常要求玩家能够根据游戏中的环境变化,灵活操纵游戏角色做出诸如攀爬、移动、跳跃、攻击、躲避、防守等不同的动作,以此达到游戏所要求的目标。这类游戏能够训练玩家手眼协调能力及对环境的反应力。最典型的动作游戏当属《古墓丽影(Tomb Raider)》系列,《波斯王子》(如图1-9所示)、《细胞分裂》以及《刺客信条》等也都是这类游戏的代表作。

当动作游戏与角色扮演游戏结合在一起,就形成了动作角色扮演类游戏(Action Role Playing Game, ARPG)。最著名的ARPG就是《暗黑破坏神》(如图1-10所示)系列,其他的还有《地牢围攻》系列、《传奇:神之手》,以及国内的《剑侠情缘》系列等。



图 1-9 《波斯王子》游戏截图



图 1-10 《暗黑破坏神》游戏截图

4. 格斗游戏(Fighting Games, FTG)

格斗游戏其实就是动作游戏,它是从动作类游戏分化出来的一类游戏。典型的格斗游戏采用两个游戏角色一对一决斗的形式。根据画面的维度,也可以分为2D格斗类游戏与3D格斗类游戏。2D格斗类游戏的经典代表作就是Capcom公司早期推出的《街霸》(如图1-11所示)系列,SNK公司推出的《拳皇》、《侍魂》等。

随着游戏业的发展,格斗游戏3D化成为大势所趋。SEGA公司于1993年推出第一款3D格斗游戏《VR战士》,后来Tecmo游戏公司推出了《生死格斗》(如图1-12所示)。3D格斗类游戏的最新代表作当属NAMCO BANDAI Games公司推出的《铁拳》系列。



图 1-11 《街霸》游戏截图



图 1-12 《生死格斗》游戏截图

5. 射击游戏(Shooting Games, STG)

射击类游戏是早期电脑游戏中最常见的游戏类型,也是许多玩家钟爱的游戏类型。射击游戏

多采用第一人称的模式,也称为第一人称视角射击游戏(First Personal Shooting Game, FPS)。根据画面的维度,可以分为平面射击游戏和三维射击游戏。

早期的射击游戏都是平面射击类,根据画面卷动的方式又可以分为横向卷轴与纵向卷轴两种。如《雷电》系列就是典型的纵向卷轴平面射击游戏;而早期的《魂斗罗》(如图 1-13 所示)、《合金弹头》就属于典型的横向卷轴平面射击游戏。

随着《毁灭战士(Doom)》、《雷神之锤(Quake)》的推出,射击游戏进入三维时代,三维射击游戏中最家喻户晓的要算《反恐精英(Counter-Strike)》(如图 1-14 所示)这款游戏了,人们常简称其为 CS。

也有射击游戏采用第三人称的模式,也称之为第三人称越肩视角的射击游戏(Third Person Shooter, TPS)。TPS 严格地说属于动作射击游戏,在第三人称模式下,游戏玩家可以做出比第一人称更多的动作,如依靠掩体、翻滚、近战攻击、攀爬跳跃等。这类游戏的代表作有《战争机器》、《死亡空间》等。

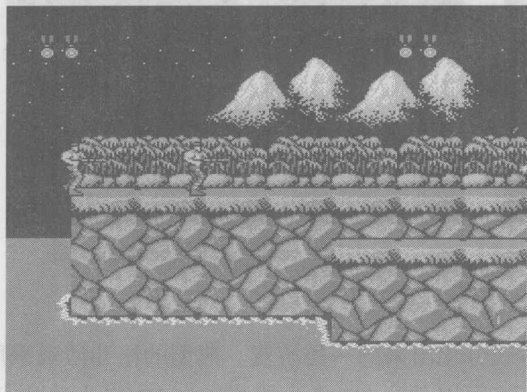


图 1-13 《魂斗罗》游戏截图



图 1-14 《反恐精英》游戏截图

6. 冒险游戏(Adventure Games, AVG)

冒险游戏更强调故事、情节以及解谜,包含探险、探索元素,通常被设计成侦探类型的解谜游戏。这类游戏往往以固定的剧情或故事为背景,为游戏玩家提供解谜的场景及环境,玩家通过在游戏中安排的故事中冒险来积累经验,解开游戏制作者所设定的谜题或疑点。

这类游戏的代表作有《神秘岛》、《魔域(Zork)》、《机械迷城(Machinarium)》(如图 1-15 所示)和《寂静岭》(如图 1-16 所示)系列等。

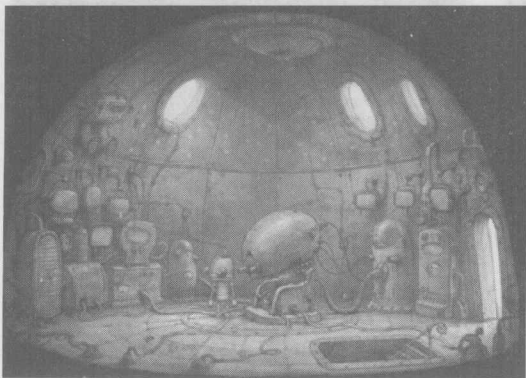


图 1-15 《机械迷城》游戏截图



图 1-16 《寂静岭》游戏截图

7. 体育游戏(Sports Games, SPG)

顾名思义,体育游戏就是在计算机上模拟各类竞技体育运动的游戏。这类游戏往往以现实中的某个运动项目为模拟对象,例如足球、篮球、拳击、钓鱼、网球、排球或高尔夫球等,玩家通过控制或管理游戏中的运动员或队伍进行比赛。这类游戏的代表作有《FIFA》(如图1-17所示)系列、《NBA》(如图1-18所示)系列、《实况》系列等。

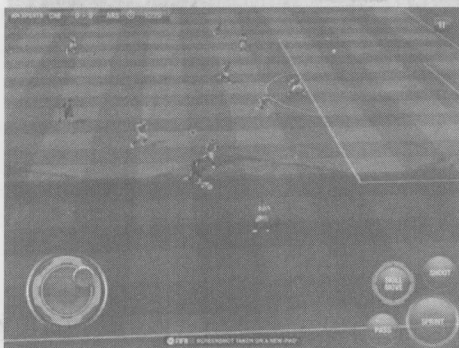


图 1-17 《FIFA》游戏截图

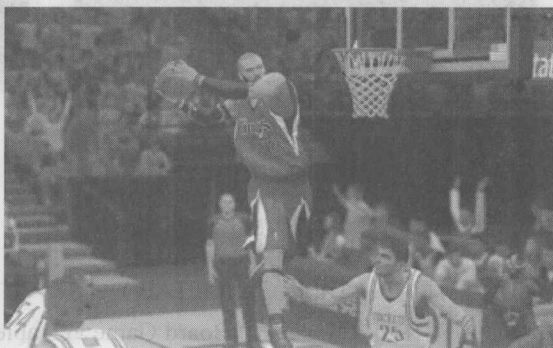


图 1-18 《NBA》游戏截图

8. 竞速游戏(Racing Games, RCG)

这类游戏往往没有故事情节,游戏的内容都是与操控某种交通工具进行高速运动为主,例如汽车、摩托车、赛艇等。但是这类游戏对图像效果和音响效果却非常讲究。为了能够流畅地运行这类游戏,往往需要强大计算机硬件的支持,因此常常被用于计算机性能的测试。这类游戏的代表作有《极品飞车》(如图1-19所示)、《山脊赛车》(如图1-20所示)、《摩托英豪》等。

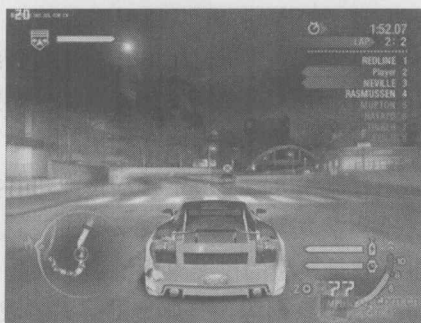


图 1-19 《极品飞车》游戏截图

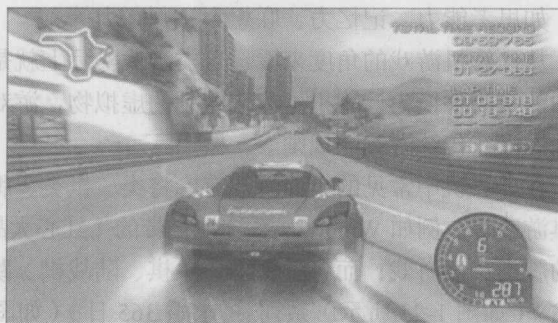


图 1-20 《山脊赛车》游戏截图

9. 模拟游戏(Simulation Games, SIM)

模拟游戏以模拟真实世界的或虚拟世界的活动为游戏内容。这类游戏会为玩家提供一个特定的虚拟世界,玩家操控游戏角色在这个世界从事各种符合逻辑的活动。从模拟的行为来划分,有两种类型的模拟游戏:经营模拟类游戏和养成模拟类游戏。

在经营模拟类游戏中,玩家需要通过对有限资源的管理和利用,去构建或扩大某种类型的社区、机构或一个帝国。这类模拟游戏的经典代表作有《模拟城市》、《主题医院》、《主题公园》及《铁路大亨》(如图1-21所示)等。

对于养成模拟类游戏而言,玩家要以养成为目的去模拟一个现实的情景。例如通过游戏中的模拟,让玩家掌握一种技术,如驾驶或飞行的技术等。这类游戏的代表作有《警察训练营》、《GT