

资深游戏设计师25年实践经验结晶，通过大量实例深入讲解了角色扮演游戏、实时战略游戏、第一人称射击游戏、模拟游戏等各种主流游戏的设计方法、技巧和最佳实践

以数据驱动的方式系统阐述了游戏机制设计、角色设计、场景设计、故事情节设计、动作设计、位置和物品设计、游戏对话设计、关卡和地图设计、界面设计、游戏数据收集与设置等游戏设计师必修技能

Basics of Game Design

游戏设计师修炼之道

数据驱动的游戏设计

(美) Michael E. Moore 著

傅鑫 陈征 戴锋 等译



机械工业出版社
China Machine Press

Basics of Game Design

游戏设计师修炼之道

数据驱动的游戏设计

(美) Michael E. Moore 著

傅鑫 陈征 戴锋 等译



机械工业出版社
China Machine Press

这是一本专供游戏设计师系统学习游戏设计方法与技巧的专著，是资深游戏设计师 25 年实践经验的结晶。本书以数据驱动的方式系统阐述了游戏机制设计、角色设计、场景设计、故事情节设计、动作设计、位置和物品设计、游戏对话设计、关卡和地图设计、界面设计、游戏数据收集与设置等游戏设计师必修技能，通过大量实例深入讲解了角色扮演游戏、实时战略游戏、第一人称射击游戏、模拟游戏等各种主流游戏的设计方法、技巧和最佳实践。

本书分 3 部分，共 12 章。第一部分包括第 1~3 章，简单介绍了游戏制作的周期、游戏设计师的职责、游戏玩法和游戏机制的分类，以及游戏制作中的数学和逻辑学，从基础层面阐释了游戏制作周期中各个要素。第二部分包括第 4~9 章，逐一详细地探讨了大多数重要的游戏机制（包括运动、战斗、角色和怪物、物品、魔法和技术、游戏中的谜题等），并且解释了设计师为查明每种机制将如何工作以及必须创建什么资产以使之工作所必须经历的过程。第三部分包括第 10~12 章，探讨了设计师必须知道的其他重要主题，比如故事和对话结构、关卡和地图设计，以及界面设计和游戏交互等。本书的附录部分收录了对领袖级游戏设计师之一 Chris Taylor 所做的一次深度采访，为即将崭露头角的游戏设计师们提供了宝贵的建议。

Basics of Game Design by Michael E. Moore. (ISBN 978-1-56881-433-9)

Copyright © 2011 by Taylor & Francis Group, LLC.

Authorized translation from the English language edition published by CRC Press, part of Taylor & Francis Group LLC; All rights reserved; 本书原版由 Taylor & Francis 出版集团旗下 CRC 出版公司出版，并授权翻译出版。版权所有，侵权必究。

China Machine Press is authorized to publish and distribute exclusively the Chinese (Simplified Characters) language edition. This edition is authorized for sale in the People's Republic of China only (excluding Hong Kong, Macao SAR and Taiwan). No part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher. 本书中文简体字翻译版授权由机械工业出版社独家出版并限在中国大陆地区销售。未经出版者书面许可，不得以任何方式复制或抄袭本书的任何内容。

Copies of this book sold without a Taylor & Francis sticker on the cover are unauthorized and illegal. 本书封面贴有 Taylor & Francis 公司防伪标签，无标签者不得销售。

封底无防伪标均为盗版

版权所有，侵权必究

本书法律顾问 北京市展达律师事务所

本书版权登记号：图字：01-2012-2099

图书在版编目 (CIP) 数据

游戏设计师修炼之道：数据驱动的游戏设计 / (美) 摩尔 (Moore, M. E.) 著；傅鑫等译. —北京：机械工业出版社，2012. 12

(游戏开发技术系列丛书)

书名原文：Basics of Game Design

ISBN 978-7-111-40087-5

I. 游… II. ①摩… ②傅… III. 游戏 - 软件设计 - 教材 IV. TP311.5

中国版本图书馆 CIP 数据核字 (2012) 第 245202 号

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑：秦 健

三河市杨庄长鸣印刷装订厂印刷

2013 年 1 月第 1 版第 1 次印刷

186mm × 240mm · 19.5 印张

标准书号：ISBN 978-7-111-40087-5

定价：69.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

客服热线：(010) 88378991；88361066

购书热线：(010) 68326294；88379649；68995259

投稿热线：(010) 88379604

读者信箱：hzjsj@hzbook.com

译者序

视频游戏行业在过去十年得到了快速增长，并且现在是家庭娱乐的主要提供者。许多分析师认为视频游戏行业是具有抗衰老能力的行业，这一点在近几年全球性的经济低迷期表现得尤为明显。尽管许多行业都不可避免地陷入了衰退期，但是视频游戏业仍然保持着稳步增长。

目前市面上有许多介绍游戏设计的图书，但是其中大多数图书谈论的都是制作周期以及设计师在这个周期的每个阶段中的责任，其他一些图书则着眼于故事构筑或关卡设计。还有一些图书详细说明了关于制作游戏的理论知识。但是几乎没有图书介绍游戏设计师应该如何着手构建驱动游戏玩法的图表和表格（数据库），本书填补了这个空白。本书解释了如何设计用于解决游戏玩法的数据集，提供了关于开发故事、构建地图和层次以及设计 GUI 的建议，还提供了许多真实的示例和实用的详细信息，包括游戏的截屏图。此外，本书还包括一个附录，是对领袖级游戏设计师之一（即 Gas Powered Game 的 Chris Taylor）所做的一次深度采访，他分享了在这个行业中的经验，并且对设计和构建游戏的过程给出了自己的见解。

本书是有抱负的专业游戏设计师的必备读物。它重点探讨了为数据驱动的游戏创建游戏机制，涵盖了角色扮演游戏、实时战略游戏、第一人称射击游戏、模拟游戏及其他游戏。本书致力于为几乎没有技术经验而又非常热爱游戏的新手提供实用指导，可以把它视作为任何有兴趣学习设计复杂视频游戏（或棋盘游戏）读者所要阅读的首本著作。

参加本书翻译的人员有：傅鑫、陈征、戴锋、许瑛琪、易小丽、陈婷、管学岗、王新彦、金惠敏、张海峰、徐晔、张德福、张士华、王涛、申川、孙玲、李振国、高德杰、官飞、俞群。

由于时间紧迫，加之译者水平有限，错误在所难免，恳请广大读者批评指正。

前 言

视频游戏行业在过去十年得到了爆炸性增长，并且现在是家庭娱乐的主要提供者。自从 NES（Nintendo Entertainment System，任天堂娱乐系统）于 1985 年发布了北美版本以来，根据市场研究公司 NDP Group 的调查研究，游戏行业的收入得到了显著增长，其中 2009 年在美国的总收入中差不多达到了 200 亿美元。世界上其他地方的销售额与美国的销售额大体相当，因此，2009 年游戏行业在全世界赚了大约 400 亿美元。甚至在经济衰退时期，游戏行业仍然保持着良好的财务状况。

游戏变得如此流行的原因有许多种，其中一个原因是：在许多不同的平台上都提供了它们，包括计算机、游戏控制台、手机、iPad 和各种手持设备，特别是 Sony PSP（PlayStation Portable）和 Nintendo DS。多年来，随着平台技术的不断改进，游戏也变得更先进，可以利用它们几乎逼真的图形和 3D 音效来创建使玩家沉迷于其中的世界，这些效果使玩家感觉到他们确实来到了战场上，或者正在与一大群僵尸较量。

当然，游戏不是电影。它们之间的主要区别是：游戏允许玩家积极地参与游戏世界里的的事件，而不是消极地袖手旁观事情的发生。游戏世界里的这种亲自参与的体验把游戏与传统的艺术和娱乐形式区分开，在游戏世界里，其中的实体似乎是独立行动的。游戏玩家将不会简单地观看英雄人物从一大群敌人当中穿过并且到达邪恶的反面角色的藏身地，而是可以选择武器、施展魔法或者使用非常高级的技术，甚至指挥军队向反面角色发起攻击。

当然，就像在其他的艺术和娱乐形式中一样，人们不得不创建环境和居住在游戏世界里的人物，把说辞强加给角色，以及定义事物将如何运转。艺术团队创建世界和实体。编程团队创建游戏引擎，用于协调使游戏出现在屏幕上以及处理世界物理学和敌方人工智能所涉及的所有过程。设计团队不仅要为游戏创建脚本，而且更重要的是，他们还有一项任务，即给游戏世界里的物体赋值，使得它们按照期望的方式交互。

在询问年轻人他们认为游戏设计师的角色是什么时，许多人回答说设计师只需提出游戏的故事情节，或许还要创建游戏中的对话。虽然故事情节和对话的确是游戏的主要成

分，但它们不是设计师的唯一职责。设计师创建驱动游戏玩法的图表和表格（数据库）的能力同样重要——例如，武器造成的伤害、再装填弹药的速度有多快、它的承载重量、它可以容纳多少发子弹以及其他的信息。除非把这些值赋予物体，否则游戏世界里将没有任何事情发生。当玩家对着猛冲过来的敌人开枪射击时，游戏代码将定义如何在屏幕上绘制武器，在开火以及在射弹命中其目标时将会出现什么特殊的视觉和音频效果。但是除非赋予了一些值，指定射弹造成的损伤和目标的健康状况，否则当射弹横穿目标时任何事情都不会发生。设计师必须提出这些值，并且解释战斗是如何解决的以及这些值相互之间有什么影响。

在简单的动作、街机和益智游戏中，出现在游戏中的物品数量是有限的，并且可以相对容易地基于测试确定和修改数据。在计算机和视频游戏的早期时代，一个人可以同时担任游戏的程序员、艺术家、设计师和录音师，因为代码比较短小，并且图形和音频很有限。当然，这些游戏就内容而言相对过于简单，要么可以随意改变启动条件，要么依靠玩家对控制的精确掌控来赢得游戏。

随着时间的推移，当游戏平台变得更快、更强大时，中央处理器（Central Processing Unit, CPU）可以处理的信息量也与日俱增。图形变得更详细并且具有更大的调色板，音频资产扩展到包括音乐、音效和解说，并且 AI 和物理学允许游戏世界里的物体执行更逼真的移动和动作。一个人不再能够处理游戏所需的所有数据，从而需要开发团队的力量。编程团队不仅需要编写包含游戏引擎的许多代码模块，还要创建供其他团队使用的工具。引荐艺术家是为了创建游戏场地以及占据它们的生物和物体的 2D 和 3D 视觉效果。聘请音频方面的专家谱写音乐、创建音效和录制声音艺术。最后，还要聘请设计师提出游戏设计理念、用文档记录团队的思想，以及生成可以保持游戏制作流程顺利进行的图表、表格及其他信息。

对游戏开发来讲，设计师现在的角色与制作电影的导演一样重要。设计师通过在文档中描述多种游戏机制来定义游戏规则。项目团队然后实现这些机制，并且设计师持续测试和修改值以使得可以按团队构思的那样玩游戏。此外，设计师还要为故事建立情节元素（假定有一个故事），以及描述在游戏世界里出现的角色、位置和物品，以便艺术团队可以构建游戏场地和模型。在必要时，如果事物在游戏世界里没有像期望的那样表现，设计师就可能被迫重新考虑游戏机制，并且必须与项目团队和艺术团队一起制定必要的修改，然后执行它们。设计师具有对游戏的构想，就像导演具有对电影的构想一样，它要求设计

师在漫长的制作周期中始终保持激情。

目前市面上有许多介绍游戏设计的图书，但是其中大多数图书谈论的都是制作周期以及设计师在这个周期的每个阶段中的责任，其他一些图书则着眼于故事构筑或关卡设计。还有一些图书详细说明了关于制作游戏的理论知识。几乎没有图书介绍游戏设计师应该如何着手构建驱动游戏玩法的图表和表格，本书填补了这个空白。本书逐一详细地探讨了大多数重要的游戏机制，并且解释了设计师为了查明每种机制将如何工作，以及必须创建什么资产以使之工作所必须经历的过程。还有一些关于设计师必须知道的其他重要主题的讨论，比如，故事和对话结构、设计关卡和地图，以及确定界面看起来将是什么样子的。但本书主要关注的是游戏机制——它们是什么以及它们将如何工作。作为一个额外的特点，本书的最后一章是对领袖级游戏设计师之一（即 Gas Powered Game 的 Chris Taylor）所做的一次深度采访，他分享了在这个行业中的经验，并且对设计和构建游戏的过程给出了自己的见解。

本书致力于为几乎没有技术经验而又非常热爱游戏的新手提供实用指导。读者无需学习地图编辑器或层次编辑器，也不用费力地通读复杂的代码示例。除了铅笔和纸张之外，机敏的读者应该无需借助任何工具即可创建游戏机制的纸质原型。可以把本书视作任何有兴趣学习设计复杂视频游戏（或棋盘游戏）的人所要阅读的首本著作。

术语**数据驱动**（data-driven）适用于本书中讨论的各类游戏，因为数据库中包括的所有数字和值都用于决定游戏动作。不过，**数据驱动的游戏架构**（data-driven game architecture）具有不同的含义，它涉及用于图形、AI、音频等的多个代码模块如何在游戏引擎中协同工作。为了防止任何混淆，本书中没有使用术语**数据驱动**。

致 谢

作者想要感谢以下人士在为本书收集图像和信息方面提供的帮助。首先并且最重要的是，要诚挚感谢 Thomas Skillman、Kendrick Fleck 和 Ben Colgan 玩过大量的游戏并且抓取了许多精彩的截屏图。感谢 Gas Powered Games 的 Chris Taylor 同意接受采访，还要感谢 Gas Powered Games 的 Bert Bringham、Kellyn Beeck 和 Dionne Roselli 提供的有价值的帮助。作者还想要感谢 Epic Games 的 Jay Wilbur、Bethesda Softworks 的 Todd Howard 和 Pete Hines、Zeni-Max Media 的 Amy Yeung、NIS America 的 Nao Zook 和 Jack Niida、TikGames 的 Mitzi McGilvray、CCP Games 的 Ned Cocker、PopCap Games 的 Eric Walter 和 Garth Chouteau、Paradox Interactive 的 Susana Meza Graham、Wiering Software 的 Mike Wiering、Albion Sword, Ltd. 的 Mike Sigman，以及 Ahmed Hakeen、Vidar Rapp 和 Steve Neeley，他们帮助我获得使用他们的游戏和产品中的图像的许可。

作者还要称赞艺术家 Ed Williamson，感谢他提供他们几个人一起开发的赛车游戏中的布局草图和故事板。另外也要向 Thomas Wilhite 表达诚挚感谢，他为本书中抓取的截屏图提供了技术帮助。

要特别感谢 Dave Eberly 充当了作者与 A. K. Peters 之间的联系人。最后并且最重要的是，要向 Sarah Culter 和 A. K. Peters 的所有优秀的人们表达深深的谢意，感谢他们帮助我把这个梦想转变成现实。

目 录

译者序
前言
致谢

第一部分 简介

第 1 章 制作游戏	2	1.4 小结	15
1.1 游戏玩法和游戏数据	2	1.5 练习	15
1.1.1 简单的设计和复杂的设计	3	第 2 章 游戏玩法和游戏机制	17
1.1.2 游戏数据	4	2.1 游戏玩法和“趣味因素”	17
1.2 设计师和开发流程	5	2.2 给游戏玩法分配百分比	18
1.2.1 游戏开发周期：预制作阶段	5	2.3 游戏题材的玩法元素	20
1.2.2 技术审查	6	2.4 游戏玩法不包括什么	21
1.2.3 游戏开发周期：制作阶段	7	2.5 游戏玩法的机制	22
1.2.4 游戏开发周期：后期制作 阶段	8	2.6 模拟现实	24
1.3 设计师在游戏开发中的角色	9	2.7 校正系数	25
1.3.1 思考和确定理念	9	2.8 小结	25
1.3.2 游戏提案	10	2.9 练习	26
1.3.3 游戏设计文档	11	第 3 章 游戏中的数学和逻辑学	27
1.3.4 原型化	12	3.1 概率和统计	27
1.3.5 资产创建	14	3.1.1 掷硬币	27
1.3.6 测试和调试	14	3.1.2 游戏中的“掷硬币”	28
		3.2 游戏中的随机选择	30
		3.2.1 随机数生成器	30
		3.2.2 把骰子用于随机选择	31
		3.2.3 看待掷骰子的另一种方式	32
		3.3 百分比	33
		3.4 视频游戏中的百分比	35
		3.5 使数学知识保持简单	36
		3.6 通过算法定义玩法	37

3.7 逻辑和脚本语言	37	5.1 石头、剪刀、布	71
3.7.1 条件语句	39	5.2 回合制战斗与实时战斗	73
3.7.2 布尔运算符	40	5.3 战斗属性（统计）	74
3.8 小结	41	5.4 角色扮演游戏中的战斗	75
3.9 练习	41	5.4.1 在 RPG 战斗期间可用的动作	76
第二部分 游戏机制揭秘		5.4.2 RPG 中的战斗属性	77
第 4 章 关于运动	46	5.5 RPG 战斗算法	78
4.1 比例	46	5.5.1 确定攻击的指令	78
4.2 图形界面的要求	47	5.5.2 确定攻击值	79
4.3 军事比例	48	5.5.3 确定防御值	80
4.3.1 战术（小型战斗）比例	48	5.5.4 战斗算法的示例	80
4.3.2 军事行动比例	49	5.6 战斗表格	82
4.3.3 战略（大战略）比例	49	5.7 关键击中表格	85
4.4 管制战争游戏中的运动	50	5.8 魔法/技术战斗	86
4.4.1 六边形地图	50	5.9 逃避战斗	87
4.4.2 样条线和导航点	52	5.10 战争游戏中的战斗	88
4.4.3 控制区域	53	5.11 战争游戏中的属性	91
4.5 运动的游戏统计	54	5.12 回合制战争游戏中的战斗	93
4.6 地形特征	55	5.13 战争游戏中的玩法序列	94
4.7 战略移动地图（世界地图）	57	5.14 实时战略游戏中的战斗	98
4.8 移动速率	59	5.15 战争游戏的 AI 设计	100
4.8.1 呈现移动路径	59	5.16 小结	101
4.8.2 勾勒出移动	61	5.17 练习	101
4.9 移动算法	61	第 6 章 关于角色和怪物	104
4.10 随机遭遇战	62	6.1 创建玩家角色	104
4.11 地图上的物品	67	6.1.1 基本的 RPG 角色属性	105
4.12 小结	68	6.1.2 RPG 种族	106
4.13 练习	68	6.1.3 RPG 阶层	110
第 5 章 关于战斗	71	6.1.4 RPG 技能	112
		6.1.5 额外的角色信息	114

6.1.6 初始装备	115	7.2 物品的游戏功能	141
6.1.7 初始技能和魔咒	115	7.3 物品图表	142
6.1.8 身体外貌	116	7.4 设计武器	149
6.2 经验点和升级	117	7.5 物品类型：近身武器	151
6.2.1 属性增加	117	7.5.1 刀	151
6.2.2 增强的魔法和技能	119	7.5.2 剑	152
6.2.3 故事发展	119	7.5.3 棍棒、斧头和其他近身武器	152
6.2.4 经验级别算法	120	7.5.4 扩展的近身武器	153
6.2.5 设定经验级别的上限	121	7.6 物品类型：远程武器	154
6.3 创建怪物、反面角色和盟友	122	7.6.1 投掷武器（非爆炸型）	154
6.3.1 实体类型	123	7.6.2 投掷武器（爆炸型）	154
6.3.2 实体属性	125	7.6.3 射弹武器（非火药型）	155
6.3.3 实体属性（怪物）图表	126	7.6.4 射弹武器（火药型）	156
6.3.4 实体战斗动作选择	128	7.6.5 射弹武器（炸弹型）	157
6.3.5 逃跑	130	7.6.6 其他便携式武器	157
6.4 财宝表格	130	7.7 物品类型：异种武器	158
6.5 体育游戏角色	132	7.7.1 激光枪	158
6.6 小结	134	7.7.2 不同寻常的射弹和动力源	158
6.7 练习	135	7.7.3 核武器	159
第7章 关于物品	137	7.8 物品类型：盔甲	159
7.1 物品类别	137	7.8.1 头部盔甲	160
7.1.1 健康状况和超自然力量	138	7.8.2 躯干盔甲	161
7.1.2 个人和环境修饰符	138	7.8.3 四肢盔甲	162
7.1.3 武器	138	7.8.4 盾牌	163
7.1.4 盔甲	139	7.9 物品类型：药剂和卷轴	164
7.1.5 弹药	139	7.9.1 健康药剂	164
7.1.6 资源	140	7.9.2 治疗药剂	164
7.1.7 任务物品	140	7.9.3 属性修饰符药剂	165
7.1.8 谜题物体	141	7.9.4 卷轴	165
7.1.9 体育器材	141	7.10 物品类型：辅助装备	166

10.3.2	“之”字形的结构	229	11.6	游戏场地设计中的问题	258
10.3.3	使命结构	230	11.7	在构建前规划	260
10.3.4	英雄的旅程结构	231	11.7.1	2D 地图原型	260
10.3.5	开放的世界结构	232	11.7.2	3D 层次原型	260
10.4	把情节链接到游戏玩法	233	11.8	小结	261
10.5	创建游戏故事	235	11.9	练习	262
10.5.1	纸面设计工具	236	第 12 章	界面设计	264
10.5.2	使用“章”组织故事结构	237	12.1	图形用户界面	264
10.5.3	测试故事的内聚性	238	12.1.1	游戏中的界面屏幕	265
10.6	游戏中的对话	239	12.1.2	外壳界面屏幕	265
10.6.1	编写对话的脚本	240	12.2	设计 GUI	266
10.6.2	减少对话选项	241	12.2.1	用于信息的屏幕位置	268
10.6.3	选通	242	12.2.2	信息表示	269
10.6.4	绘制 NPC 对话的流程图	243	12.2.3	菜单	271
10.7	小结	244	12.3	游戏控制	272
10.8	练习	244	12.3.1	定义控制交互性	273
第 11 章	设计游戏场地	246	12.3.2	非传统的游戏控制	274
11.1	2D 地图	246	12.3.3	游戏控制的问题	275
11.2	3D 层次	249	12.4	反馈	275
11.3	设计游戏场地	252	12.5	小结	277
11.3.1	多条通道	253	12.6	练习	278
11.3.2	把物体放在游戏场地中	253	附录 A	Chris Taylor 的采访	280
11.4	脚本语言和游戏场地设计	254	本书涉及游戏的版权声明	298	
11.5	游戏场地设计中的视点	256			

| 第一部分 |

Part 1 |

简 介

第 1 章 制作游戏

第 2 章 游戏玩法和游戏机制

第 3 章 游戏中的数学和逻辑学

制作游戏

视频游戏行业（多功能控制台游戏、计算机游戏和移动游戏都可以属于此类）已经从被少量技术爱好者享受的业余爱好成长为一种吸引了全世界玩家的娱乐形式，并且产值达到数十亿美元。游戏已经步入成熟期，它们现在正对诸如图书、电影、唱片甚至电视之类的传统娱乐形式发起挑战。在这之前，从未有一种娱乐形式可以让用户如此直接地与最终产品交互，并且这种交互性可以吸引如此多的玩家参与视频游戏。

一些视频游戏相对简单，只需几分钟的时间就能玩游戏。其中一些视频游戏是单机游戏，可以一个人玩；而另外一些视频游戏是在线游戏，可以有许多玩家参与其中。还有另外一些视频游戏要复杂得多，需要花费几天或几周的时间才能完成。视频游戏的种类如此繁多，足以使每个人都找到某款可以使他们自己感到愉快的游戏，而不管他们所希望玩游戏的时间有多长。

是什么使得视频游戏如此有吸引力呢？依赖于游戏的类型，可以给玩家提供许多不同的游戏情节。一位玩家可能享受使游戏中的角色执行一系列狂野的动作（跳到圆桶上，躲避落下的石块，拔出枪，轰炸坏蛋）所需的身体灵活性；另一位玩家可能享受求解谜题的智力练习；另有一位玩家梦想着成为一名著名的棒球运动员，利用大满贯的本垒打进行清垒。还有其他的玩家可能通过变成行业的百万富翁或企业界大亨而享受登上巅峰的挑战。玩家可以利用多种不同的方式享受视频游戏。不过，在所有情况下，玩家都是通过控制器或键盘/鼠标在屏幕上移动物体以及执行各种动作，来实现与游戏的交互。因此，与传统的被动娱乐形式不同，这种交互性有助于使视频游戏风靡全世界。

1.1 游戏玩法和游戏数据

玩家在玩游戏期间执行的动作构成了游戏玩法。每种游戏类型都有它自己的一组动作，尽管许多游戏共享公共的动作，比如，在屏幕上四处移动物体。简单的游戏要求玩家

执行的动作较少，而复杂的游戏可能具有许多动作。例如，在经典的街机游戏《Pong》中，玩家只需在屏幕上向上和向下移动控制手柄，以拦截一个移动的球，并把它弹回给另一位玩家（参见图 1-1）。在第一人称射击游戏中，主要关注的是通过游戏世界移动一个角色，并向 AI（人工智能）控制的敌人（在有的死亡模式中有时是其他玩家）射击。可能有多种不同类型的运动方式，比如奔跑、行走、跳跃、倚靠、蜷伏等。还有许多种不同的武器，玩家可以在玩游戏期间收集和使用它们。

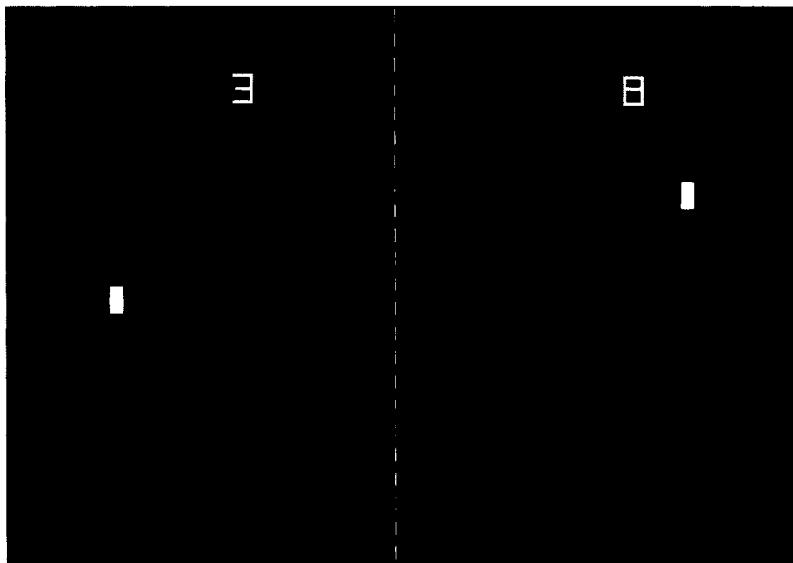


图 1-1 在《Pong》游戏中，与这里显示的图像类似，游戏玩法包括简单地上、下移动控制手柄，撞击球使其越过中间的网

随着游戏变得越来越复杂，涉及的动作也变得越来越复杂。例如，在角色扮演游戏中，玩家要执行许多活动——从探索世界到交战，到与非玩家控制的角色（Non-Player Character, NPC）交谈，到在商店中购买和销售物品，再到求解谜题。一些模拟游戏可以让玩家想象他们在驾驭飞行器或者开着赛车，而其他游戏则允许玩家建造金融帝国。在这样复杂的游戏里，玩家要做大量的工作，并且要做出许多决策。玩家可以反复地玩这些复杂的游戏，这是由于游戏世界中发生事件的多样性而导致每个结局都与上次玩游戏时不同。

1.1.1 简单的设计和复杂的设计

在所有情况下，无论是在简单的游戏中还是在复杂的游戏里，确定游戏玩法的工作原

理都是设计师的职责。在简单的游戏中，设计师经常直接与程序员打交道，以决定事件在游戏中是如何发生的——从乒乓球移过屏幕的速度，到球拍在与球接触时施加了多大的力等。简单的游戏通常依靠随机性防止游戏变得非常容易预测，除了指定事件发生的概率之外，设计师很少能够控制机会因素。

在更复杂的游戏（例如，第一人称射击游戏和实时战略游戏）中，设计师通常需要提供关于游戏世界以及其中包含什么内容的相当多的信息。例如，如果游戏中有武器，设计师就不得不考虑诸如射击速率、造成的伤害、每次装填的弹药量等之类的事情。在非常复杂的游戏（例如，角色扮演游戏和战争游戏）中，信息量非常庞大，并且可能要花很长的时间来进行平衡，以确保任何一支部队都不会过于强大，并且没有哪种玩法总会取得成功。

在简单的游戏中，通常没有太多的信息（或数据）需要设计师处理，因此把所有的一切都硬编码进游戏引擎中。更改代码以改进玩法通常不是太困难。不过，在更复杂的游戏，将会有多得多的数据，最好保持将数据与游戏引擎分隔开。这样做将允许设计师在测试期间根据需要调整值，而无需重新编译代码，因此可以在游戏中执行更改并对其进行测试。重新编译代码需要花费时间，并且会使程序员和设计师在这个过程中都无所事事。因此，对于更复杂的游戏，将把数据保存在单独的文件中，设计师无需重新编译代码即可访问和更改它们。一旦准备好交付游戏，就可以把最终的数据直接纳入游戏引擎代码中，使得阴险狡诈的玩家不能操纵它。

1.1.2 游戏数据

本书重点关注的是设计数据密集型游戏，并且讨论设计师在生产期间必须创建和维护的不同类型的数据。对于设计师来说，游戏数据可以分为4大类：图表和表格、关卡/地图布局、脚本以及对话。

- **图表和表格**（chart and table）是指关于游戏中的物体（例如，武器、怪物、财宝、制作、角色生成等）的值及其他信息的列表。表格特指在其中操纵图表数据以执行游戏动作（例如，战斗决议、魔法攻击、买卖物品等）的方法。
- **关卡/地图布局**（level/map layout）是指如何在二维或三维空间中创建游戏场地，以及如何在它们上面放置物体以使玩家与之交互。