

TP 311.41
B21

Flash 游戏设计宝典

Sham Bhangal

Igor Choromanski

等著

康

博

译

清华大学出版社

(京) 新登字 158 号

北京市版权局著作权合同登记号: 01-2001-0262

内 容 简 介

本书深入介绍了 Flash 在游戏设计和制作方面所具备的功能。主要内容包括电脑游戏的类型及特点、制作好游戏所要求的条件、游戏图形的优化、Flash 的内置对象、事件处理、理解人工智能、制作多人游戏、游戏中的声音和音乐、基于顺序的游戏与高级逻辑、3D 效果、结构化实时编程、联机数据及具体案例等。

本书内容深入浅出、形象生动, 各种知识技巧与具体案例紧密结合。非常适合于广大爱好 Flash 游戏制作的朋友及希望在此领域有所成就的专业人士阅读。

Sham Bhangal, Igor Choromanski et al: Flash Games Studio.

EISBN: 1-903450-67-5

Copyright© 2001 by friends of ED.

Authorized translation from the English language edition published by friends of ED.

All rights reserved. For sale in the People's Republic of China only.

Chinese simplified language edition published by Tsinghua University Press.

本书中文简体字版由清华大学出版社和英国乐思出版公司合作出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 翻印必究。

本书封面贴有清华大学出版社激光防伪标签, 无标签者不得销售。

书 名: Flash 游戏设计宝典

作 者: Sham Bhangal Igor Choromanski 等著 康博 译

出 版 者: 清华大学出版社(北京清华大学学研大厦, 邮编 100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑: 郭东青

封面设计: 康博

版式设计: 康博

印 刷 者: 北京昌平环球印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 35.75 字数: 915 千字

版 次: 2002 年 6 月第 1 版 2002 年 6 月第 1 次印刷

书 号: ISBN 7-302-05473-8/TP·3225

印 数: 0001~5000

定 价: 59.00 元

前言

如果你曾经考虑过如何开始设计电脑游戏这个问题，那么这本书会解答你的问题。Flash 为我们提供了一种简单而又功能强大的制作游戏的工具，它的界面简单，但是内容广泛而复杂。它使得我们能够制作出真正复杂精巧而且非常有趣的游戏，因此制作游戏就不再那么具有神秘感了，

Flash 中基于矢量的图形非常具有吸引力而且很容易创建，尤其是这些图形和电影剪辑结合在一起的时候就更容易在屏幕上显示。这些内容构成了一副非常漂亮的运动画面，再加上 ActionScript 就会组成一个完美的控件。这一切看上去都出奇地简单，当然这都是 Flash 的功劳。

我们在描述一个游戏的时候是通过每一步的有关内容，从简单（比如游戏的基本原则）到复杂（比如说人工智能）。期间将会说明使用 ActionScript 来控制动画的必要性，以及 Flash 内置的对象和方法如何帮助你制作出你想要的东西。此外，如果 Flash 中没有你想要的函数，那么还可以通过 Flash 自己动手来制作。

我们购买一个游戏软件的时候也许对一些问题不以为然，但是谁又想过，怎样使一个物体和另一个物体碰撞？怎样来确定碰撞的后果？如何使物体跟在另一个物体后面？如何使游戏在某种规则下进行，甚至如何知道游戏正在运行？如何将网络上玩家的高分保存下来，或者退一步讲，究竟分数是什么呢？这些问题只能想象得出来但都是很难解答的，不过有了 Flash 之后，这一切都可以实现。

当然，我们还要介绍一些有关背景，这个背景会使你的游戏鹤立鸡群。作为一个游戏，必须有趣，有气氛有动力。因为游戏要对众多玩家做出响应，因此面临的挑战还是很大的。但是这些要求并不意味着我们要去下载有关的背景，Flash 完全能够做到这点。本书中我们会向你介绍如何应用 Flash 做出最好的游戏效果。

这本书会讲到许多基础的内容，也会涉及一些高级的方面。如果你看到那一段代码比较复杂，这也正好验证了 Flash 可以达到的高度。当然，你还可能做到一些想都没有想过的事。

总之，你是制定游戏规则的人，Flash 只是规则的服务者，我们写这本书的时候，力求做到简单明了，因此用的格式也比较简捷。

代码下载

本书的所有源文件都可以从 www.friendsofed.com 这个网站下载。在这个网站中进入代码部分，然后根据 Flash Game Studio 的链接来找到合适的文件。如果源文件或者在下载时有什么问题，请你通过下面列出的方法和我们联系，我们将尽力给你帮助。

技术支持

如果对于本书有什么疑问，请到我们的网站获取 E-mail 和我们联系，或者直接使用下面这个 E-mail: feedback@friendsofed.com 与我们联系。



另外，在我们的网站上您还可以进行下面的活动：和知名的游戏设计者交流，获取我们编写的其他书目上的例子；你可以在留言板上发布自己的疑问，讨论和解答问题或者看看别人对问题的讨论。因此如果你有问题，请和我们联系，我们非常欢迎你的来信。

目 录

第 1 章	电脑游戏概述	1
1.1	认识计算机	1
1.2	认识 Flash	1
1.3	电脑游戏	2
1.3.1	从儿童游戏室到娱乐场	3
1.3.2	平台游戏	5
1.3.3	角色扮演游戏	5
1.3.4	第一人称射击游戏	6
1.3.5	体育游戏	7
1.3.6	格斗游戏	8
1.3.7	模拟游戏	9
1.4	分阶段编写游戏	10
1.5	好游戏的特征	11
1.5.1	在纸上设计游戏	12
1.5.2	缩小文件	13
1.5.3	压缩文件	13
1.5.4	游戏发送选项	14
1.5.5	性能	14
1.5.6	控制	16
1.5.7	游戏故事	17
1.5.8	挑战	18
1.5.9	奖励	19
1.5.10	用户统计	19
1.5.11	测试	20
1.5.12	标准	20
1.5.13	重复可玩性	20
1.6	本章小结	21
第 2 章	优化游戏图形	23
2.1	帧的概念	23
2.2	图形要素	24
2.2.1	像素	24
2.2.2	时钟	25



2.2.3	屏幕缓冲区	26
2.3	矢量	28
2.3.1	缩放与旋转	32
2.3.2	线条厚度	34
2.3.3	位图	34
2.3.4	透明度	35
2.3.5	图像质量	35
2.3.6	文字	38
2.3.7	渐变	38
2.3.8	性能统计	44
2.4	简化游戏	51
2.5	增强可下载性的文件优化	53
2.6	本章小结	58
第 3 章	Flash 的内置对象	59
3.1	定义对象	59
3.2	游戏对象	60
3.2.1	Object 对象	61
3.2.2	Color 对象	61
3.2.3	Sound 对象	62
3.2.4	Mouse 对象	63
3.2.5	Key 对象	63
3.2.6	Math 对象	64
3.2.7	String 对象	64
3.2.8	Array 对象	64
3.2.9	XML 对象	65
3.3	电影剪辑	65
3.3.1	特性	66
3.3.2	带有特性的动画	69
3.3.3	层次	76
3.3.4	方法	77
3.3.5	脚本库	79
3.3.6	剪辑事件与智能剪辑	79
3.4	本章小结	80
第 4 章	事件处理	81
4.1	onClipEvent 函数	81
4.2	用户交互	82
4.2.1	键盘	83

4.2.2 鼠标	83
4.3 按钮事件处理程序	83
4.3.1 用户输入 onClipEvent	87
4.3.2 输入特性和方法	87
4.4 使用的事件处理程序	88
4.5 自定义控件	90
4.6 派生与初始化	102
4.6.1 地雷	102
4.6.2 射击	103
4.7 碰撞检验	106
4.7.1 放置地雷时的碰撞检验	107
4.7.2 游戏运行时的碰撞检验	107
4.7.3 子弹与地雷的碰撞检验	108
4.8 计时器	110
4.9 销毁	110
4.9.1 超时销毁	110
4.9.2 碰撞销毁	111
4.10 本章小结	112
第 5 章 基于顺序的游戏与高级逻辑	113
5.1 定义游戏的原则	113
5.2 编写基于顺序的游戏代码	121
5.2.1 分段创建游戏代码	122
5.2.2 玩家的输入	129
5.2.3 利用事件脚本	130
5.2.4 最后的玩家脚本	132
5.2.5 简单智能的模拟	135
5.2.6 在游戏中使用函数	140
5.2.7 欢乐的时刻	144
5.2.8 警告	145
5.3 本章小结	145
第 6 章 结构化实时编程	146
6.1 结构化实时编程概述	146
6.2 做好实时编程的准备	146
6.3 对速度的要求	147
6.3.1 Flash 的瓶颈现象	147
6.3.2 简化编码	147
6.3.3 代码的优化	148



6.3.4 层次的功能	152
6.4 gameWorld 和 gameSprites	153
6.5 gameWorld 的几个例子	155
6.6 游戏的 Flash 时间线	162
6.7 例 2: 滚动射击游戏	175
6.7.1 游戏背景	177
6.7.2 游戏的图形	177
6.7.3 玩游戏	178
6.7.4 组件和工作流的标识	179
6.7.5 对象间的关系	181
6.8 对象	182
6.8.1 GameWorld 对象	182
6.8.2 Star 对象	184
6.8.3 Terrain 对象	186
6.8.4 PlayerShip 对象	187
6.8.5 Cursor 对象	190
6.8.6 Alien 对象	191
6.8.7 Radar 对象	194
6.8.8 LaserBeam(PlasmaControl)对象	196
6.8.9 AlienSpawner 对象	198
6.8.10 ScoreKeeper 对象	199
6.9 警告	200
6.10 本章小结	200
案例研究 1 Turn	202
C1.1 游戏如何运行	202
C1.1.1 游戏背景	204
C1.1.2 面向对象的代码	205
C1.1.3 游戏的设计	205
C1.1.4 游戏 Turn 的 FLA 结构	207
C1.1.5 创建游戏环境	207
C1.1.6 object 电影剪辑脚本——第一部分	209
C1.2 arms 电影剪辑	211
C1.2.1 object 电影剪辑脚本——第二部分	211
C1.2.2 计数器	213
C1.2.3 万事俱备	214
第 7 章 设计一个平台游戏构造软件包	215
7.1 平台游戏	215

7.2	游戏运行的原理	216
7.2.1	关图	216
7.2.2	游戏结构	217
7.2.3	滚动	218
7.2.4	拾取物和敌人	218
7.2.5	平台	220
7.2.6	玩家	220
7.2.7	玩家与平台的碰撞检测	223
7.2.8	重置	224
7.2.9	建立关图	224
7.3	建立关编辑器	226
7.3.1	编辑器窗口	226
7.3.2	函数	230
7.4	关的数据结构	236
7.5	显示关	238
7.6	扩展关编辑器	239
7.7	适应游戏引擎	240
7.7.1	新的图像, 新的游戏对象	240
7.7.2	超高跳跃、跳板以及可以消融的平台	241
7.7.3	固定玩家, 滚动游戏环境	242
7.8	本章小结	242
第 8 章	交互式声音	243
8.1	为什么游戏是不同的	243
8.1.1	声音文件导入选项	243
8.1.2	关于添加声音的一些内容	244
8.2	基本的声音 Scripting Actions	245
8.3	把声音应用到游戏	248
8.3.1	何时添加声音	249
8.3.2	声音创建过程	249
8.4	声音层次——目标	256
8.4.1	目标声音方法	256
8.4.2	setVolume	257
8.4.3	利用 setPan 添加立体声效果	261
8.4.4	setTransform	265
8.5	本章小结	267
第 9 章	游戏中的音乐	268
9.1	音乐	268



9.1.1	让音乐出现	268
9.1.2	看不见的音乐	268
9.2	音乐的几个要素	269
9.2.1	节奏	269
9.2.2	音调/旋律	269
9.2.3	和弦	270
9.2.4	风格	270
9.2.5	基本的事实	272
9.3	什么样的作品	272
9.3.1	Super Mario Brothers	272
9.3.2	Warcraft	272
9.3.3	You Don't Know Jack	273
9.3.4	Unreal Tournament	273
9.4	何时运行? 运行什么	273
9.5	软件	274
9.6	Flash 中的音乐	275
9.7	从 Flash 导出音乐	275
9.7.1	完整的声轨	278
9.7.2	简单循环	280
9.7.3	实时混合/分层	282
9.8	本章小结	289
第 10 章	理解人工智能	290
10.1	您自己的宠物	290
10.1.1	确定逻辑和模糊逻辑	291
10.1.2	人工智能	292
10.2	您的第一个虚拟宠物	292
10.2.1	一些游戏规则	293
10.2.2	代码	299
10.2.3	_root, 第 1 帧	299
10.2.4	_root, 第 2 帧	305
10.2.5	mrGreen, 第 2 帧	306
10.2.6	mrGreen, 第 8 帧	306
10.3	本章小结	306
第 11 章	第 3 维	307
11.1	仿 3D	308
11.1.1	层次化和视差滚动	309
11.1.2	预渲染	312

11.1.3 将 3D 图像加入 Flash 中	314
11.1.4 光强和阴影	316
11.2 疯狂弹跳	320
11.3 星际巡洋舰	330
11.3.1 能量球部分的代码	338
11.3.2 完整的飞船脚本	340
11.3.3 修改建议	347
11.4 等比例 3D	347
11.4.1 实现的方法	349
11.4.2 玩家的位置和运动	351
11.5 本章小结	354
第 12 章 真正的 3D	356
12.1 方块的制作	356
12.1.1 定义形体	357
12.1.2 物体的定义	358
12.2 物体的绘制	364
12.3 形体的渲染	367
12.3.1 基本概念	368
12.3.2 绘制立方体	379
12.4 子图	385
12.5 最后的思考	396
第 13 章 终极 3D	397
13.1 Flash 的局限性	398
13.2 Flash 多边形引擎	399
13.3 形成 3D 图像	407
13.4 游戏速度	412
案例研究 2 Mech Attack	414
C2.1 游戏介绍	414
C2.2 游戏的组成部分	416
C2.3 制作游戏	425
C2.3.1 机器人	426
C2.3.2 城市景象	429
C2.3.3 GUI 显示	430
C2.3.4 瞄准线	432
C2.3.5 激光	433
C2.3.6 路灯	435



C2.3.7	增强器	436
C2.3.8	飞行炸弹	439
C2.3.9	多边形	444
C2.3.10	加载程序	449
C2.3.11	Controller	454
C2.4	尾声	473
C2.5	本章小结	474
第 14 章	联机数据	475
14.1	PHP	475
14.2	MySQL	476
14.3	用 Flash 和 PHP 做游戏	476
14.4	保存分数	485
14.4.1	数据库	485
14.4.2	引入 PHP	486
14.4.3	Flash 和 PHP 配合使用	489
14.4.4	返回数据库	493
14.4.5	避免缓存	493
第 15 章	多人游戏应用程序	495
15.1	系统体系结构	496
15.1.1	点对点结构	496
15.1.2	客户机-服务器	497
15.1.3	Internet 协议地址	497
15.1.4	端口	498
15.1.5	防火墙	498
15.2	多人 Flash	499
15.2.1	HTTP	499
15.2.2	套接字	501
15.2.3	数据包	504
15.2.4	服务器	505
15.3	具体游戏	507
15.3.1	游戏屏幕	508
15.3.2	游戏的控制	509
15.3.3	碰撞	513
15.3.4	节点	517
15.3.5	多辆汽车	519
15.3.6	游戏中的数据包和通信	521
15.4	套接字对象	524

15.4.1	登录和连接	532
15.4.2	Connect 按钮代码	532
15.4.3	Play Local 代码	533
15.4.4	汽车控制的完全代码	534
15.5	同步	542
15.6	本章小结	544
第 16 章	后记——Director	546
16.1	针对 Flash 用户的 Director	546
16.1.1	文化冲击	547
16.1.2	界面	549
16.2	Director 中的脚本	554
16.3	本章小结	555

第1章 电脑游戏概述

娱乐自古以来就是一个备受人们关注的问题。不论是打水漂(即扔出石子使其在水面上蹦跳),捉迷藏,还是类似象棋这样的智力挑战,游戏在考验人们技巧的同时,也很好地释放了工作所带来的压力。

在每一个国王与公主、妖怪与魔法师的游戏中,人们的梦想都会尽情地释放出来,直至达到人类想象力的极致,而这正是游戏的娱乐本质所在。

在本章中,我们将要讨论的问题包括我们可以做什么,应该做什么,以及如何去做。其中具体包括以下主要内容:

- Flash 在游戏设计方面所具备的功能
- 目前常见的游戏种类,以及如何在 Flash 中对它们进行处理
- 制作好游戏所要求的条件,并提供几个经典游戏
- 如何开始进行游戏设计

1.1 认识计算机

现如今在比特、带宽、像素和 RAM 的时代,我们已经逐渐接受了计算机成为日常生活的一部分这一事实。计算机可以代替我们工作、写字、计算和通信。但有一样事情却是它无法做到的,那就是梦想。而这就需要现代的游戏设计人员把一堆冰冷、生硬且没有生命的电子器件和人类的想象力结合起来,从而创造一个全新的世界。

由于 Internet 越来越普及,它正在渗透到社会和文化的各个角落,因而人与人之间相互影响的广度大大超越了可以衡量的程度。如今一个游戏设计人员通过 Internet 在一天之内影响到的人数甚至超过了莫扎特毕其一生所做得到的结果。现在正有几百万人坐在计算机前等待着玩下一个让他们感到惊喜的游戏,而这正是您所需要做的。

那么,我们如何让自己成为一名杰出的游戏设计人员呢?其实要做的第一件事情就是坐下来好好看看 Flash 这个软件能做什么,不能做什么。

1.2 认识 Flash

请注意,如果您想让游戏具有以下特性:分辨率为 1024×768 的全纹理贴图 3D 效果,运行速度为 80 帧/秒,以及全 CD 的音响效果,那么很不幸,Flash 可不是您的最佳选择。因此您需要记住以下几个要点:

- 我们使用的是 Flash 的内置语言 ActionScript。通常用 C 和汇编语言编写的专业游戏可被



编译及优化为很好的可执行文件,而 ActionScript 在运行时是由 Flash 播放器或插件读取和解释的。这就降低了它的运行速度。

- Flash 无法对玩家的硬盘进行写操作。因此,它自己并不能大量保存游戏进度和玩家的个人信息,而这种功能是确保游戏可玩性的必要条件。不过这可以通过将 Flash 和 PHP 结合使用得以实现,这部分内容将在本书后面介绍。

- 最后一点也是关键的一点,Flash 不能访问单个像素。您不能对图像的像素颜色进行修改,或将位图作为纹理粘贴到多面体上。

总之,Flash 是在低带宽上传输矢量图形多媒体的一种手段。它最适合于显示简单、平滑及可准确生成的对象。不过这并不是说我们只能用 Flash 来制作飞行的文字和漂亮的图案。实际上,只要程序员很好地掌握了 ActionScript,同时充分发掘它的潜能,Flash 还是能够胜任非常多的领域的。

其实上面介绍的 Flash 的局限性并不是什么大问题。实际上 Flash 可以很强大。它在便于使用和功能强大之间保持了很好的平衡。另外,Flash 还具有优秀的声音效果,同时它充分代表了人们在轻松创建动画的努力中所取得的重要成果。

还记得早期的个人电脑吗?那时创建一个有声音的游戏可不是件容易的事情,而且当您声音进行管理时,它只有一个声道,因此每次只能发生一个事件。而现在好了,Flash 全面支持与事件相关的声音效果到全流音轨之间的任何音响效果,同时它最多可支持 8 声道的实时混音。

下面让我们再来看看 Flash 的图形功能。在早期的个人电脑上,在屏幕上显示一个简单的符号需要绘制出背景的所有像素,接着还要一个像素一个像素地检查子图形(即图形符号),以此来判断是否每个像素都被添色或不可见。这个过程对于计算机来说是一件很繁琐的事情,而现在 Flash 则避免了这个问题,原因就在于它绘制的是真正的矢量对象,因此它只需要处理子图形所在的各个区域就可以了。矢量的一个明显优势就是它们定义的是对象而不仅仅是像素,这就意味着它们可以直接绘制在任何给定的背景上。下一章中我们将详细介绍 Flash 的图形功能,同时还要专门针对游戏介绍这些图形功能是如何进行优化的。

Flash 还包含了完整的键盘处理例程,而且现在利用 Flash 5 您可以检测到非常重要的 Key UP 事件了。同时,在 Flash 中分辨率是独立的,也就是说不管您将监视器设置为何种分辨率,Flash 电影看起来总是清晰明快的。

因此,在具体涉及到 ActionScript 的速度问题之前,我们可以看到它是多么的优秀。它具有像 C 这样的传统语言的所有主要功能,包括全部数学计算、逻辑、数组和字符串方面的功能。另外利用 Flash 我们还有可能创建从空间和重力计算到图表及统计这样一些复杂的程序。而我们所面临的主要挑战是要了解如何让 Flash 完成我们需要的工作。相信完成了本书的学习之后,您就会掌握这些知识。

上面我们介绍了 Flash 的一些基本规则,下面将开始介绍本书的主题——游戏。

1.3 电脑游戏

目前市场上有许多不同类型的电脑游戏,而且还在不断产生新的游戏类型。现在最流行的

要算第一人称射击游戏(First Person Shooter)了, 不过这并不是说它会一直流行下去。现在已经有一些定义很明确的电脑游戏类别, 它们构成了目前电脑游戏的基础:

- 益智游戏
- 博弈/纸牌游戏
- 娱乐场游戏(Casino Games)
- 平台游戏
- 角色扮演游戏
- 第一人称射击游戏
- 冒险游戏
- 体育游戏
- 格斗游戏
- 模拟游戏
- 战略游戏(包括实时战略游戏)及其他游戏

1.3.1 从儿童游戏室到娱乐场

用 Flash 来制作像益智游戏这样最简单的游戏是非常有效的。尽管这些游戏制作起来简单明了, 但在画面的完美程度上要求却是很高的。举例来说, 通常益智游戏只包含一个画面, 而玩家要在其中解决一道难题。一些有条条块块的游戏, 如经典的 Tetris(Nintendo 1989), 要求玩家在一个画面中将下落的不同形状的条块放置在空白处, 目标是用条块填满并消除一行。这样的游戏简单且令人着迷, 如图 1-1 所示。

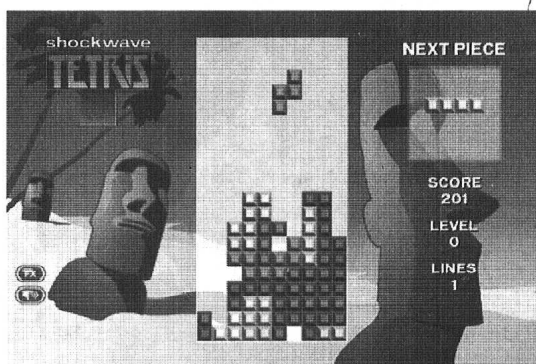


图 1-1

其他益智游戏都是在个人电脑出现以前的传统游戏基础上发展而来的, 如纵横字谜游戏、刽子手游戏和象棋游戏。虽然有时也会有设计人员突发奇想并通过动画和交互等手段让游戏中的对象更加生动, 但这仍然摆脱不了对原有游戏的模仿, 充其量只能算是一种改进而已。

除了让游戏画面更加漂亮和充实之外, 您还必须考虑游戏的逻辑问题, 这样才能在 Flash 中创建一个谜题。例如当您单击某个地方时会发生什么情况? 当某个符号移动到另一个地方时又会发生什么? 这也许才是制作益智游戏的过程中最重要并且也是最具挑战性的工作, 当然同时这也最能锻炼您使用 ActionScript 的技能。

纸牌游戏在电脑上开发起来同样相对容易。如今包括从 Solitaire 到 Poker 这样的经典纸牌游戏都已在个人电脑上重新开发过了。和益智游戏一样, 这些纸牌游戏的简易性正是其非常适



合 Flash 的原因。目前在这方面已经有一些很好的例子了,例如对古代中国的一种经典游戏——麻将所进行的改编,如图 1-2 所示。

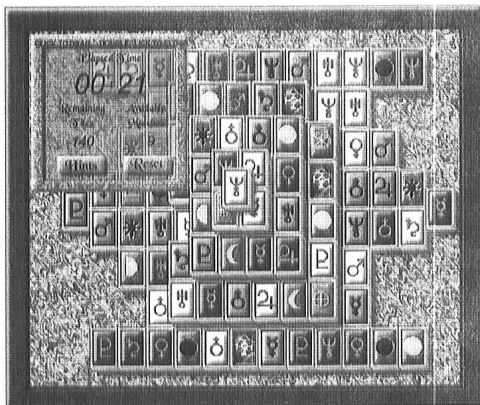


图 1-2

对于游戏设计的新手而言,在无须高深的声音和图形设计技巧的情况下,博弈游戏或纸牌游戏既为他们提供了很好的创作机会,同时又使其不必从头开始创建一个全新游戏。这样程序员就可以在已有游戏规则的基础上把精力集中在怎样触发游戏方面,而不用绞尽脑汁发明出一套游戏规则并对整个游戏进行设计。

这类游戏的另一个优点是它们为玩家提供了在 Internet 上与其他用户进行交互的机会。在 5 年前,玩娱乐场电脑游戏(如图 1-3 所示)虽然也挺有意思,但也只能在赢得积分点数的同时获得别人的喝彩而已。

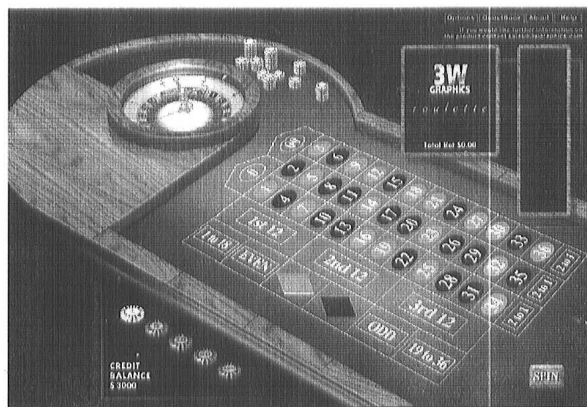


图 1-3

而如今娱乐场游戏正在逐渐允许人们赢取真正的钱,这就使这些游戏本身确实变成了一桩很大的买卖!联机娱乐场游戏从掷双骰子、轮盘赌到纸牌游戏和老虎机,其内容无所不包。而像 Bingo 这样的面向大众的游戏现在也越来越流行了,因而目前有许多公司都在致力于开发提供娱乐场游戏全攻略的网站。

但是必须要说明的是,要真正建立一个联机的娱乐场游戏,您必须同时有一台服务器或合适的后台设备,即您的 Flash 游戏能够与之通信的中央计算机,以此来确保玩家的输赢结果能得到处理,以及像 Bingo 这样的实时游戏保持同步。当然,这些并不应该成为您对编程开发游戏失去兴趣(哪怕是一丁点)的理由!