

TURING

图灵程序设计丛书

Apress®

Foundation Game Design with Flash

Flash游戏编程 基础教程



[加] Rex van der Spuy 著
李鑫 陈文登 付斌 译



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

Foundation Game Design with Flash

Flash游戏编程 基础教程



[加] Rex van der Spuy 著
李鑫 陈文登 付斌 译

人民邮电出版社

北京

TP391.41
5708

图书在版编目 (C I P) 数据

Flash游戏编程基础教程 / (加) 斯白 (Spuy, R.) 著;
李鑫, 陈文登, 付斌译. — 北京: 人民邮电出版社,
2011.7

(图灵程序设计丛书)

书名原文: Foundation Game Design with Flash

ISBN 978-7-115-25434-4

I. ①F… II. ①斯… ②李… ③陈… ④付… III. ①
网页—动画制作软件, Flash—教材 IV. ①TP391.41

中国版本图书馆CIP数据核字(2011)第087879号

内 容 提 要

本书是Flash游戏设计方面的入门级图书。全书共包括5个完整的游戏项目。作者循序渐进地讲解了游戏设计步骤、游戏设计技巧、游戏设计过程中遇到的问题及其解决方案,一步步引领你进入Flash游戏设计的殿堂。通过本书的学习,你将掌握Flash中的基础知识,如图层、帧、元件、实例、影片剪辑、库面板、属性面板、各种绘图工具及其使用技巧;你将学会AS3.0编程技术和技巧,例如如何利用Flash和AS3.0制作交互式游戏、平台类游戏,学会面向对象的游戏设计技术,了解AS3.0程序设计的所有相关内容。

本书适合Flash游戏设计初学者学习。

图灵程序设计丛书

Flash游戏编程基础教程

◆ 著 [加] Rex van der Spuy
译 李 鑫 陈文登 付 斌
责任编辑 王军花
执行编辑 李 静

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京艺辉印刷有限公司印刷

◆ 开本: 800×1000 1/16
印张: 30.25
字数: 718千字 2011年7月第1版
印数: 1-4 000册 2011年7月北京第1次印刷

著作权合同登记号 图字: 01-2010-2361号

ISBN 978-7-115-25434-4

定价: 75.00元

读者服务热线: (010)51095186转604 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Original English language edition, entitled *Foundation Game Design with Flash* by Rex van der Spuy, published by Apress, 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705 USA.

Copyright © 2009 by Rex van der Spuy . Simplified Chinese-language edition copyright © 2011 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由 Apress L.P.授权人民邮电出版社独家出版。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，侵权必究。

译者序

亲爱的读者，当你拿起这本书的那一刻，我们就已断定，你和我们一样深爱 Flash 游戏设计。因为我们也曾和你一样，怀着自己创作游戏的憧憬，开始了 Flash 游戏设计历程。有共同爱好的人，心灵的距离是最近的。虽然未曾谋面，但我们完全能想象出你内心对制作 Flash 游戏的那份向往。我们三个人就是在这共同理想的驱动下，翻译了此书。相信有了这本书的陪伴，你朝梦想迈出的第一步，一定会更加稳健。

本书的作者也是一名 Flash 游戏设计者，因此在阅读的过程中，你会发现他的每一处讲解、每一句提醒都出现得恰到好处，因为他也曾经历过这样的学习过程。这本书最吸引人的地方就是一个个生动的实例，作者为每一章的内容都精心设计了一款有趣的游戏。这些游戏本身玩起来就很有意思，从中你不但能够直观地看到各种技术的展示，还能真切地感受到，是作者天马行空的想象力为这些游戏赋予了生命。本书的另一大特点应该算是它简明通俗的讲解方式了。虽然随着讲解的逐步深入，作者也引入了不少术语，但几乎无处不在的比喻和拟人化文字，会使你感觉像在读一本故事书一样。因此，这本书不会让你有任何的压力。学习的过程中难免有困难与阻碍，但你会发现，每当讲解复杂的问题时，总会在后面看到一句鼓励的话语。虽然只是简短的一句，但它带给你的力量却足以去征服一个又一个困难。当你读完本书，成为一名合格的 Flash 游戏开发者后，你就会发现，这个伙伴带给你的不仅仅是设计 Flash 游戏的技术，还有处理难题时的那一份沉稳与自信。

在翻译本书的时候，各种游戏制作技术都没有停止它们飞速发展的脚步。如今，各种游戏运行平台、游戏编程语言层出不穷。虽然 Flash 游戏仍然牢牢占据着网页游戏的霸主地位，但其他技术也不甘示弱，纷纷奋起直追。现在你是不是也正在为学习哪一种技术而困惑呢？其实，只要了解了游戏的设计原理和规律，不管将来游戏制作技术怎样发展变化，对你来说都不是问题，你一样可以利用不同的平台、不同的编程语言来制作出精彩的游戏。

作为游戏制作的初学者，Flash 无疑是最佳选择。它不但拥有强大的编程语言——ActionScript 3.0，而且还有直观的图形绘制工具。这要比其他那些抽象的编程语言更适合初学者。而且 Flash 自身也在不断完善发展，充分挖掘内在的潜力。活跃的社区和广大的用户群给予它生生不息的活力。所以，你不必为选择哪种游戏制作技术而感到迷茫，先把 Flash 学好吧，它完全可以把你的游戏梦想一一实现。

本书的语言非常口语化，其中使用了很多俚语来帮助读者理解知识点。我们在翻译的过程中尽量还原书中一些非常有特色的语言，但因为水平有限，难免有不当之处，请读者不吝指正。

如果你在阅读过程中有任何疑问，欢迎与我们联系，让我们共同探讨、学习。

最后，我们要感谢这本书，因为它使我们三个人成为了很好的朋友。我们也真心地希望通过这本书，与每一位喜欢 Flash 游戏设计的人成为好朋友！

译者付斌（盐酸酸）已经在其博客中为这本书准备了专门的讨论页面。博客地址为 <http://www.asbinbin.com>。

（这里我还建议咱们赶个时髦，大家留下各自的微博 ID，比如@盐酸酸。电子邮件的局限性在于，如果反复问一个问题，我们就不得不反复回复或转发同样的内容。）

2011 年 2 月 7 日

引 言

想制作视频游戏的人该从何处入手？该学什么？该请教谁？

假如你曾想过这些问题，就会明白这些问题其实很难回答。许多年以前，当我还住在印度班加罗尔郊外一个小村庄的时候，我就问过自己同样的问题。当时，我在那里的一所国际学校里教授编程及交互式媒体制作。所有的学生都喜欢玩游戏并且也想自己制作，但当时他们却没有合适的书籍或网络资源，不知从何学起。

曾经有人做过一些调查，其结果令人吃惊：较为简单的游戏设计所需要的编程技术其实并不多，不仅如此，而且这些技术还可以不断用在以后的各种情景下，从而可以使我们开发出完全不同的游戏。游戏制作不仅很好玩，而且很快就能看到结果，此外你还可以在游戏制作过程中充分发挥创造力。秉承这个调查结果的精神，我们编写了中学阶段 3 门游戏设计课程的讲义，随后又专门编写了本书。

回想一下 Flash 4 和 Flash 5 时的情况吧。那时 ActionScript 编程语言还是 1.0 版，Flash 还自带了一些非常好的交互式教程，初学者可以根据这些教程逐步学习。当时大家似乎都是初学者，所以很容易就能找到一些针对新手的书籍及教程。

但是现在的 ActionScript 已经是 3.0 版了，跟过去已大相径庭。ActionScript 变得更为强大，但也更为复杂。你所搜集到的很多 ActionScript 教学资源都是针对高级内容的，从前没怎么编过程序的初学者越来越难以得到相对全面的启蒙教学。但颇具讽刺意味的是，用 AS3.0 开发游戏要比用 AS1.0 容易多了。所以，本书不会涉及 AS3.0 中那些明显很复杂的内容，你将学到的只是游戏制作所需的必备知识，因为我们想让游戏开发变成一件很好玩的轻松活，让任何人都可以开发游戏。

开发游戏是一件很奇妙的事情。要知道，现实中没有太多的魔力，而学会制作游戏却可以使我们的能力最接近于此。快抓住这头巨龙的缰绳，开始你的狂野之旅吧，马上你就会惊讶于自己的上手速度之快！

目 录

第 1 章 编程基础：如何制作视频游戏.....1	
1.1 所需基本软硬件配置.....1	
1.2 需要知道的内容.....2	
1.3 不必知道的内容.....2	
1.4 一切不过是编程而已.....2	
1.4.1 编程？但是我数学学得不好.....3	
1.4.2 我已经知道怎么编程了.....4	
1.4.3 我能做出什么样的游戏.....4	
1.4.4 学习一些新术语.....4	
1.5 打好地基.....5	
1.5.1 需要的文件.....5	
1.5.2 设置工作环境.....6	
1.5.3 设置 Flash 开发人员工作区.....7	
1.5.4 设置 ActionScript 代码格式首 选参数.....7	
1.6 编写你的第一个程序.....8	
1.6.1 开始编程.....8	
1.6.2 不要忽略类.....9	
1.6.3 使用构造函数.....11	
1.6.4 对齐代码.....12	
1.6.5 你的指令是什么.....13	
1.6.6 导入并扩展 MovieClip 类.....14	
1.6.7 为代码添加注释.....17	
1.7 发布 SWF 文件.....18	
1.8 项目面板.....21	
1.9 AS3.0 与 Flash Player 的更多内容.....22	
1.10 命名规范.....23	
1.11 小结.....24	
第 2 章 创建对象.....25	
2.1 理解交互对象.....25	
2.1.1 设置工作环境.....26	
2.1.2 创建故事书的第一页.....28	
2.2 绘制第一个页面.....30	
2.2.1 绘制背景.....30	
2.2.2 组织图层与时间轴.....33	
2.2.3 绘制前景对象.....35	
2.2.4 将对象组合到一起.....42	
2.2.5 加上几朵云.....43	
2.2.6 添上几朵花.....45	
2.2.7 再学习一些技术.....47	
2.3 创建角色.....49	
2.4 使用按钮.....52	
2.4.1 创建按钮元件.....53	
2.4.2 理解按钮状态.....56	
2.4.3 创建指针经过状态.....57	
2.4.4 创建按下状态.....58	
2.4.5 复制按钮.....59	
2.4.6 库的组织.....60	
2.4.7 在场景中加入按钮.....61	
2.5 小结.....61	
第 3 章 为对象编写程序.....63	
3.1 显示故事书的第一页.....63	
3.2 对按钮进行编程.....71	
3.2.1 使用点标记法.....71	
3.2.2 如何调用方法.....73	
3.3 事件与事件侦听器.....79	
3.4 对故事书按钮进行编程.....84	
3.4.1 onHillButtonClick 事件处 理函数.....87	
3.4.2 使用 onPondButtonClick	

事件处理函数	87	第 6 章 控制游戏角色	175
3.4.3 添加后退按钮	88	6.1 设置项目文件	175
3.4.4 要知道何时使用这种模式	90	6.2 用键盘控制游戏角色	176
3.5 小结	91	6.2.1 键盘控制对象的错误方法	176
第 4 章 控制影片剪辑对象	92	6.2.2 键盘控制对象的正确方法	182
4.1 影片剪辑属性	92	6.3 设置舞台边界	189
4.1.1 设置项目文件	94	6.3.1 在舞台边界阻止移动	190
4.1.2 上下移动小猫	94	6.3.2 创建更好的猪圈	191
4.1.3 对象的 x 坐标与 y 坐标	98	6.3.3 屏幕折返	195
4.1.4 递增移动	98	6.4 卷屏	196
4.1.5 调整影片剪辑对象的大小	105	6.4.1 创建游戏环境	197
4.1.6 隐藏影片剪辑对象	110	6.4.2 调优游戏角色	198
4.1.7 猫眼看世界	113	6.4.3 卷屏基础	199
4.2 控制影片剪辑的时间轴	117	6.4.4 优化卷屏	201
4.3 小结	128	6.4.5 完善卷动	206
第 5 章 游戏逻辑判断	129	6.5 知识拓展	209
5.1 设置项目文件	129	6.6 小结	210
5.2 设计 GUI	131	第 7 章 碰撞检测	211
5.2.1 输入/输出	133	7.1 创建项目文件	211
5.2.2 添加文本字段	133	7.2 哎呀, 撞上了	214
5.3 创建一个简单的猜谜游戏	137	7.3 改变动态文本字段	215
5.4 变量知识扩展	139	7.4 状态改变的触发	216
5.5 逻辑判断	144	7.5 减少生命槽的值	218
5.5.1 显示游戏状态	148	7.6 更新分数	223
5.5.2 游戏胜负	155	7.7 拾取及掉落物体	229
5.5.3 使用方法模块化程序	160	7.8 hitTestObject 的缺点	235
5.6 完善游戏	161	7.8.1 检测与对象边框的碰撞	235
5.6.1 解决随机数字	161	7.8.2 试着容忍它	237
5.6.2 禁用 Guess 按钮	164	7.8.3 建立一个子对象	237
5.6.3 游戏重玩	166	7.9 使用 hitTestPoint	239
5.6.4 最终程序代码	168	7.10 建立障碍物体	247
5.7 进一步分析	170	7.11 基于轴的碰撞检测	250
5.7.1 记录猜过的数字	171	7.12 小结	261
5.7.2 添加视觉显示	171	第 8 章 面向对象的游戏设计	262
5.7.3 使用回车键输入数字	171	8.1 面向对象编程简介	262
5.7.4 对调游戏逻辑	173	8.1.1 如何将类绑定到元件上	263
5.8 小结	174	8.1.2 使用属性和方法	263
		8.1.3 私有属性和方法	264

8.1.4 如何在类之间利用取值方法和赋值方法来进行通信	266	9.1.3 摩擦力	333
8.1.5 开始学习面向对象技术	270	9.1.4 反弹	335
8.2 实例练习: 地牢迷宫大冒险	270	9.1.5 重力	337
8.2.1 构建游戏	271	9.1.6 跳跃	339
8.2.2 准备项目文件及场景物体	272	9.2 实例学习: 捉虫者	342
8.2.3 进入地牢	272	9.2.1 建立项目文件	343
8.2.4 游戏关卡布局	274	9.2.2 使用 Player_Platform 类	344
8.2.5 游戏中的对象	274	9.2.3 添加平台	350
8.2.6 制作时间轴动画	276	9.2.4 平台上的碰撞检测	352
8.2.7 为对象制作动画	277	9.2.5 使用循环	355
8.2.8 在舞台上添加或移除对象	281	9.2.6 使用循环访问平台对象	358
8.2.9 地牢迷宫大冒险的工作原理	285	9.2.7 子对象的全局 x 和 y 坐标	359
8.3 DungeonOne_Manager 类	287	9.2.8 转向一个对象	363
8.3.1 移动游戏角色	290	9.2.9 让青蛙的眼睛转向游戏角色	365
8.3.2 捡起钥匙	293	9.2.10 改变堆叠层次	366
8.3.3 打开第一扇门	294	9.2.11 在代码中添加一些小虫——真正的小飞虫	369
8.3.4 添加声音到游戏中	298	9.2.12 使用数组	376
8.3.5 创建 Sound 和 SoundChannel 对象	299	9.2.13 收集小虫	381
8.3.6 与敌人发生碰撞	300	9.2.14 完整的 Main_BugCatcher 类	386
8.3.7 游戏失败	301	9.3 新的 Collision.playerAndPlatform 实用程序	390
8.3.8 捡起星形武器	302	9.4 小结	397
8.3.9 发射子弹	304	第 10 章 高级对象和角色控制	398
8.3.10 使用子弹对象	305	10.1 拖放对象	398
8.3.11 子弹与敌人的碰撞	307	10.1.1 面向过程方式的拖放	399
8.3.12 游戏角色与墙壁的碰撞	309	10.1.2 面向对象方式的拖曳	406
8.3.13 同步 ENTER_FRAME 事件	311	10.1.3 继承的替代品: 复合	411
8.3.14 游戏获胜	312	10.2 通过鼠标来移动对象	413
8.4 改进游戏	313	10.2.1 让一个对象移到鼠标所在位置	414
8.4.1 添加一个新的关卡	313	10.2.2 添加动态滤镜	416
8.4.2 创建一个游戏管理器	315	10.2.3 在对象运动中使用缓动	418
8.5 从舞台中移除对象	322	10.2.4 延迟鼠标跟随	420
8.6 小结	324	10.3 高级缓动	422
第 9 章 平台游戏: 游戏中的物理学及数据管理	325	10.3.1 Tween 类的属性和方法	423
9.1 模拟真实的运动	325	10.3.2 easing 包和方法	425
9.1.1 建立项目文件	326	10.3.3 Tween 事件	427
9.1.2 加速度	327		

10.3.4	缓动到随机位置并计算速度	428	10.6.1	跟随游戏角色	453
10.4	实例练习：复杂的鼠标角色控制	434	10.6.2	从游戏角色旁逃走	456
10.4.1	Player.as	434	10.6.3	瞄准并朝游戏角色开火	457
10.4.2	移动游戏角色	437	10.7	使用其他的角色控制系统	462
10.4.3	旋转魔杖	439	10.8	发送事件	463
10.4.4	360° 开火	439	10.9	实例：太空射击	465
10.4.5	Bullet.as	442	10.9.1	游戏结构	466
10.5	对象工厂	449	10.9.2	创建子弹	467
10.5.1	产品类	449	10.9.3	检测子弹与对象间的碰撞	468
10.5.2	工厂类	450	10.9.4	在舞台边界移除子弹	471
10.5.3	客户端类	451	10.9.5	类与事件	472
10.6	敌人的人工智能系统	453	10.10	小结	472

1.2 需要知道的内容

本书假定你没有使用过 Flash 软件，或者说没有任何编程经验。你会在循序渐进的学习旅途中见证游戏编程世界的奇妙所在。如果希望能从零开始学习游戏设计，本书将会满足你所有的入门需求。

即便如此，Flash 和 ActionScript 编程语言这两个巨大的主题也足以让你花费终身时间去学习，况且没有任何一本书能够解答学习时可能遇到的所有问题。如果以前从来没有使用过 Flash，那么我强烈建议你一定要花一点时间去仔细阅读一下该软件的帮助文档，而且还要完成一些 Adobe 在线帮助系统里的习题和示例项目（如要访问它们，只需从 Flash 的菜单中选择“帮助”（Help）▶“Flash 帮助”（Flash Help）即可）。你可能也会发现周末时读上一本介绍 Flash 的好书会非常有用，比如 friends of ED 出版社出版的那些专为 Flash 初学者准备的书就十分好。这些资料会为你可能遇到的一些更基础性的问题作出解答，比如如何使用 Flash 及其用途之类的问题，与此同时它们不会增加太多复杂的东西，因为你不必非得理解游戏设计与 ActionScript 编程中的一些概念性问题。

但是如果马上深入学习的话，我会帮助你！这本书将会是你绝好的起点，一册在手，万事无忧。

1.3 不必知道的内容

以下内容不是学习本书所必需的，了解这些没准会让你茅塞顿开。

- 数学知识（不管怎么说，用不着太多）。
- 计算机编程知识。
- 网站设计知识。
- 图形设计知识。
- 几乎其他一切知识。

事实上，我甚至允许你说“我讨厌计算机”，或者会任由你沉浸于在屏幕上将极为笨重的物体扔出去的幻想之中。不过请放心，我在某些时候、某些方面也会跟你有完全相同的感受。

1.4 一切不过是编程而已

本书内容中涉及最多的就是关于如何编写计算机程序的问题。**计算机程序**有点像影片脚本，它可以使游戏中的角色与对象都能知道他们该做什么以及在某些条件下他们该如何反应。比如你设计了一款视频游戏，玩家必须要使用键盘上的方向键来引导一只鸭子避开饥饿异常的乌龟，顺利通过池塘，当玩家按下方向键的时候鸭子如何能够知道它必须移动呢？这就需要编写一个程序来告诉鸭子这样做。

ActionScript 就是你要用来为游戏编写程序的计算机编程语言的名称。它是一门十分精巧且功能强大的语言，与 Java 语言密切相关。ActionScript 现在的版本是 3.0，一般简称为 AS3.0。就学习编程而言，它是一种极好的语言，具有以下优点。

- 它被完全集成到了 Flash 的图形设计环境中，用它开发具有丰富视觉元素的游戏要比用其他绝大多数编程语言快得多。
- 基于以上原因，你总可以在屏幕上立刻看到程序的执行结果。这就使得编程体验变得非常清晰且令人满意。与许多其他编程语言的学习过程相比，AS3.0 的学习要直观得多。
- Adobe 公司做了大量的工作，使得 AS3.0 的编写体验非常好。它简化了技术障碍，使得程序的启动与运行只需单击一次鼠标即可完成。
- AS3.0 同 Java 或 C++ 一样，是一种“真正的”编程语言。人们用了很多年来开发它，并且用了一种叫做 EMCAScript 的开源编程语言来对它进行编译。学习用 AS3.0 编写程序最大的好处是，所学技能都可以直接应用到其他编程语言的学习中。未来几年都可以将这些技能作为基础。你的编程能力会不断得以提升。
- 用 AS3.0 开发的游戏与程序都是跨平台的，这就意味着只要在系统中安装有 Adobe 的免费 Flash Player 软件，这些游戏与程序就可以运行在任何一个操作系统中（Windows、Mac OS X 或 Linux）。Flash Player 是有史以来安装量最大的软件之一，所以你不必为每一种操作系统重新编写程序代码，从而保证了你的游戏会有极多的潜在用户。
- 在因特网上有由大批热心的 AS3.0 开发者组成的社区，他们投入了大量精力来编写教程，并在讨论区与线上论坛内帮助别人。如果在编写程序的过程中遇到了困难，只需要在众多的 Flash 和 ActionScript 讨论区之一中提出你的问题，就一定会收到有建设性的答复。

1.4.1 编程？但是我数学学得不好

我的数学也不好！不会编程的人对编程有一个最大的误解，那就是认为编程就是一些数学计算。但情况不是这样的。虽然从表面上看可能如此。为了方便起见计算机编程借鉴了一些数学语法，但是从根本体系上来讲两者是完全不同的。

这并不是说在这些课程中完全用不到数学知识——肯定会用到。到什么程度呢？你会用到加减乘除和一些非常基本的代数知识（那些你从小学五年级就可能已经记住了的知识）。我们所用到的数学知识就只会达到这种复杂程度，而且实际上 AS3.0 会处理所有的数学计算。

尽管确实不必马上就使用很多数学知识来构建游戏大作，但对于游戏开发者来说，深入发掘用 AS3.0 进行编程的数学潜力却一定会带给你更多的选择机会。这里介绍两本非常值得一读的综合性图书，它们所介绍的数学知识要比本书详尽得多。一本是由 Keith Peters 编写的《ActionScript 动画基础：让物体动起来！》（*Foundation ActionScript Animation: Making Things Move!*），另外一本是由多人合著的《Flash 数学创意》（*Flash Math Creativity*）。这两本书都是由 friends of ED 出版社出版的，它们是本书的完美拍档。你可以直接把这些书里面所介绍的技术应用到本书的游戏项目中。

无论想要多么复杂的效果，利用 AS3.0 都能实现。我们会在后面章节里用一些几何知识来完成一些特殊的动画效果。不必一定要理解这些特效制作背后的几何学机制，只需知道如何在游戏

内容中使用它们就行了。本书内容在很大程度上是从非数学的角度来编写的，所以那些惧怕数学的人应该会比较喜欢！

1.4.2 我已经知道怎么编程了

本书是为了尽可能地让初学者接受而编写的，并没有假定读者会有任何编程基础。但读者中可能也有许多人是经验老到的程序员，他们阅读本书的目的可能是想知道如何使用自己现已掌握的 AS3.0 技术来开发游戏。别担心，尽管前面几章确实是专门面向 Flash 和 ActionScript 初学者的，但后面章节中却涉及了相当多的非常高深的面向对象的程序设计知识，它们应该能满足你的胃口。

我建议你翻到第 5 章，该章是本书中首次从头到尾地使用编程技术构建了一个完整游戏的章节。如果它的内容显得有些难懂或者其中有些术语和概念你不太理解，那你完全可以后退一到两章以便找到适合自己水平的内容。如果你能够很好地接受第 5 章内容，那就大胆地从这章开始读吧！从编程的角度来看，从这章往后内容才开始变得更有意思了。

游戏设计所涉及的许多技术都非常专业。即使你可能掌握了相当多的有关 ActionScript 或者编程的知识，但如何将这些技术应用到游戏开发当中却并不总会那么容易。本书的重点是游戏设计的架构而不是编程的具体细节。假如你具有丰富的编程经验，那么本书将会为你讲解如何在完整的游戏开发过程中使用这些技术。

有些读者可能是第一次接触 Flash CS4 和 AS3.0，但在此之前已经学过旧版的 Flash 并且用 AS1.0 或 AS2.0 编过程序。为使你们更容易转换到 AS3.0，这本书同样也会指出使用 AS3.0 与 AS 语言前续版本在问题处理方式上的差异。

1.4.3 我能做出什么样的游戏

本书所关注的主要是二维的动作游戏、冒险游戏以及街机游戏，也会涉及解谜与益智类游戏。Flash 非常适合创建这些类型的游戏。本书每章内容都会针对游戏中的每一步骤给予相应的指导，但书中的游戏项目都是开放且无限制的，我鼓励你在游戏中运用自己独创的一些方法。

尽管 Flash CS4 确实能够相当不错地实现 3D 效果，但对于 3D 效果这样一个庞大的主题来说，它值得单独去写一整本书来谈论。更重要的是，你在开发 2D 游戏时学到的绝大多数游戏设计技术（特别是关于如何架构游戏的知识）经过稍微修改或根本不用修改都可以直接应用于 3D 游戏的开发之中。为了简化学习进程以及尽可能地确保有针对性的教学目的和清晰的教学内容，我决定在本书中将内容锁定在 2D 游戏的开发上。

1.4.4 学习一些新术语

就像在任何大型专业领域一样，计算机编程领域也有许多新的专业术语需要学习。本书会尽量避免使用过多的行话，取而代之的是稍显冗长但意思比较明确的描述性语句。有些术语的应用范围很广，所以你应该掌握它们，而且本书也会在文中解释所有的新术语。

1.5 打好地基

作为一个游戏开发者，你可以将自己看做一名建筑师。所有不同大小或形状建筑物都有一些共同的基本要素：它们都需要地基、墙以及屋顶。不管你的房子是大是小，是简陋逼仄还是富丽堂皇，都需要打好地基，砌好几面墙，然后搭起一个屋顶。做好这些之后就可以开始干些真正有趣的事情了：设计室内布局、布置装修、购买家具，最后再举办一个乔迁派对。

本章下面的内容将会教你编写一个非常简单的程序，以此来奠定在本书余下部分中创建所有游戏及程序的基础。如果你以前没有编写过程序或者说打算利用 AS3.0 开始走上编程之路，那么本章就是为你准备的。

假如你之前有过编程的经历，那么你可能想要翻到本章的末尾，去看看究竟会有多少比较熟悉的技术与概念。你得保证已经熟悉了那些用来编写游戏和程序的代码架构，不过如果你觉得本章内容看起来相当简单，可以将它略过。

为了向计算机编程史致以敬意，本章将编写一个名叫 Hello World 的程序，因为它是人们所编写过的最简单的完整程序，所以通常它是初学者开始学习一门新编程语言时所编写的第一个程序。它的功能非常简单：只能在 Flash 的“输出”(Output)面板上显示出单词“Hello World!”。

这个程序似乎很小，但是通过编写它可以达到两个非常重要的目的。

(1) 创建一种健壮且灵活的编程规范，这种规范会成为本书中所有项目核心，并且有助于你日后创建其他项目。

(2) 这样一个 AS3.0 编程速成课可以为学习后续章节所涉及的一些很重要的概念与技术打好基础。

还对编程抱有恐惧心理吗？得了！不用理会它们了！在本章中，你会抓住编程这头野兽的双角，然后将它摔到地上！

1.5.1 需要的文件

用 Flash 和 AS3.0 创建游戏时至少需要下面两个文件。

- **AS 文件**：这个文件包含所有你写的 AS3.0 程序代码——它们是游戏的大脑。AS 文件只不过是一种文件后缀名为 .as 的文本文件（马上你就会知道得更详细）。如果你使用过前面几个版本的 Flash，而且是从 AS2.0 或 AS1.0 转到 AS3.0 上面来的话，可能会直接在时间轴上的帧中完成大部分或全部编码工作，并且只使用一个文件：一个 FLA 文件。确实，可以继续使用时间轴上的代码，不过从现在起你得努力改掉这一点并且永远不要再这样使用代码了。不用担心，你很快就会发现将所有程序代码放进一个单独的 AS 文件实际上要比 Flash 早先版本中的做法更有效率，事实上也更容易，因为后者总是会使用时间轴代码变得规模庞大且难于管理。
- **FLA 文件**：这就是一般的 Flash 文件，后缀名为 .fla，程序中的所有可视化对象都保存在这种文件中，比如游戏角色及环境。AS 文件实际上处理着所有的“思考性”工作，但是它得依靠 FLA 文件来做所有的管理性工作：对程序代码进行解释并运行代码（将程序代码

转变为可操控的游戏)，并将运行结果显示在屏幕上，还得使玩家能与其互动。FLA 文件也能包含游戏要用到的**嵌入式资源**（比如动画、声音、视频以及图像作品）。

游戏制作完成后，FLA 文件与 AS 文件会共同运作并生成一种被称为 **FLASH 影片文件** (Flash movie file) 的第三种文件。Flash 影片文件就是游戏的成品，可以将其上传到因特网与所有人共享。Flash 影片文件的后缀名为 .swf，所以在本书的余下部分我将它们简称为 SWF 文件。要想创建 SWF 文件，得从 Flash 中进行发布 (publish)，待会儿你就知道怎么做了。

1.5.2 设置工作环境

在开始编写 Hello World 程序之前，需要组织工作目录并创建空的 FLA 与 AS 文件，这是开始编程时所必需的东西。本书中的所有项目都会使用这种方式来创建。

(1) 在硬盘中合适的位置创建一个项目文件夹，名叫 Hello World Program。

(2) 打开 Flash。选择“文件”(File) ▶ “新建”(New)，然后从“新建文档”(New Document) 对话框中选择“ActionScript 文件”(如图 1-1 所示)。

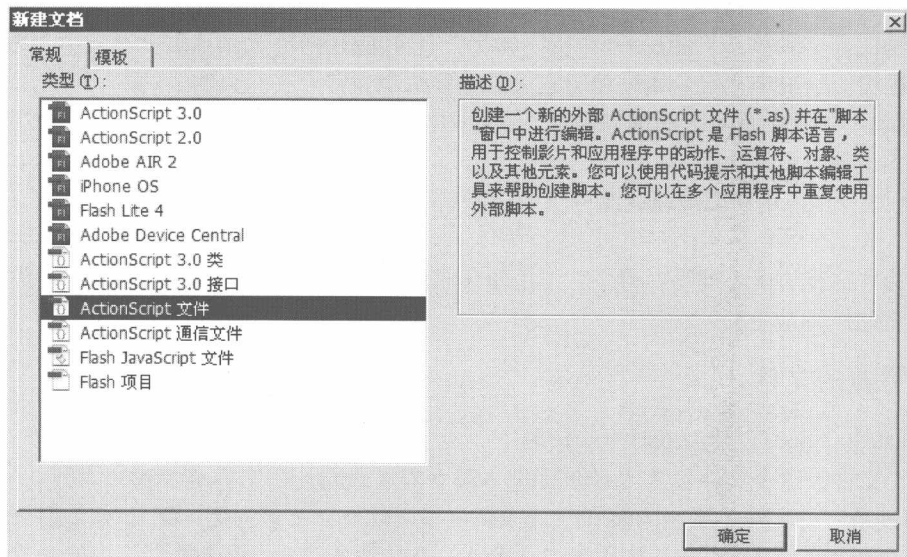


图 1-1 创建 ActionScript 文件

(3) 单击“确定”(OK)。

(4) 此时打开了一个新的名叫 Script-1 的 ActionScript 文件。选择“文件”▶ “另存为”(Save As)。输入新文件名 Main.as。将它保存在第 1 步所创建的 Hello World Program 文件夹中。

我们将在这个新文件 Main.as 中编写所有程序。(一定要大写该文件名的首字母 M，否则该程序将不能运行。)本章结束时，我会介绍编程中所用的一种习惯做法，它叫做命名规范，你在该项目中碰到的一些奇怪的字母大写方式都可以用它来解释。现在只需确保按照我所建议的名称来拼写即可。

(5) 现在创建 FLA 文件。选择“文件”>“新建”，然后从“新建文档”对话框中选择“Flash 文件 (ActionScript 3.0)”。

(6) 单击“确定”。

(7) 这就打开了一个新的叫做 Untitled-1 的 Flash 文件。选择“文件”>“另存为”。输入新文件名 helloWorld.fla。将它与 Main.as 文件保存在同一个文件夹 Hello World Program 中。

(8) 检查 Hello World Program 文件夹，确保其中的文件结构如图 1-2 所示。

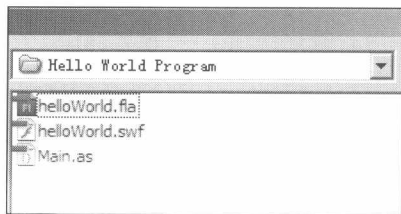


图 1-2 检查以确保 FLA 与 AS 文件在同一文件夹下

1.5.3 设置 Flash 开发人员工作区

Flash 允许自定义工作环境，这样就能安排好各种要用的窗口与面板布置。这些区域被叫做工作区 (workspace)。

请将工作区设置为开发人员工作区 (Develop workspace)，建议用该工作区进行编程。(程序员有时也被称为开发人员，因为他们能够开发软件应用，比如游戏。)

1.5.4 设置 ActionScript 代码格式首选参数

下面将设置 Flash 的首选参数，经过如此格式化后的程序代码可读性好。虽然目前还不一定能看到这样做的好处，但是本书余下的项目都要使用同样的格式，所以现在就该设置它。

(1) 如果你的操作系统是 Mac OS X，那么选择“Flash”>“首选参数” (Preference)。如果用的是 Windows 操作系统，选择“编辑” (Edit) > “首选参数”。

(2) 在“类别” (Category) 菜单中，选择“自动套用格式” (Auto Format)。

(3) 选中该项后，“首选参数”窗口应该如图 1-3 所示。

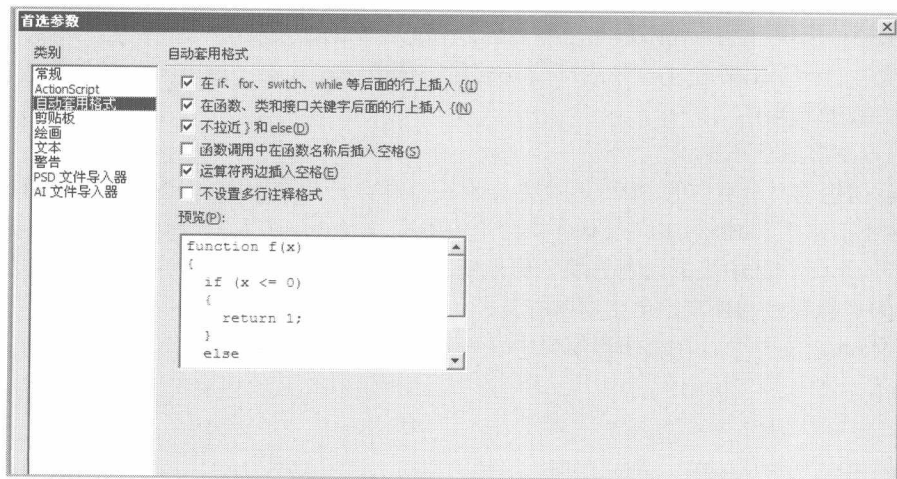


图 1-3 设定代码格式首选参数使代码容易阅读与调试