

使用HTML5和JavaScript开发属于你自己的游戏

HTML5

游戏开发进阶指南

Pro HTML5 Games

[印] Aditya Ravi Shankar 著

谢光磊 译



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

Apress®

HTML5

游戏开发进阶指南

Pro HTML5 Games

[印] Aditya Ravi Shankar 著

谢光磊 译

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书介绍了 HTML5 游戏开发的一般过程和技巧。全书共分 12 章,第 1 章介绍了本书相关的 HTML5 的诸多新特性,包括在 canvas 上绘图、播放声音等,另外还引入了子画面页的概念;第 2~4 章利用 Box2D 物理引擎开发了一款非常类似于《愤怒的小鸟》的游戏,全面介绍了物理引擎的概念,以及在游戏中使用物理引擎的方法,这一部分还引入了视差滚动技巧,以实现某种伪 3D 效果;第 5~10 章开发了一款简化版的《红色警戒》游戏,这一部分涉及了相当多的内容,包括地图的制作,建筑与单位的设定、绘制,玩家与单位的互动,触发器与剧情的编写,路径规划算法等;第 11、12 章利用 Node.js 和 WebSocket 开发了《红色警戒》游戏的多人对战模式,这一部分的主要问题是保持游戏在多个玩家的浏览器中的状态一致。

对于熟悉前端开发,想用前端技术做一些更酷的事情,却又不知如何下手的人,这是一本不可多得的好书。

Pro HTML5 Games By Aditya Ravi Shankar, ISBN: 978-1-4302-4710-4

Original English language edition published by Apress Media. Copyright © 2012 by Apress Media

Simplified Chinese-language edition copyright © 2013 by Publishing House of Electronics Industry All rights reserved.

本书中文简体版专有版权由 Apress Media 授予电子工业出版社。专有出版权受法律保护。

版权贸易合同登记号 图字: 01-2013-5971

图书在版编目(CIP)数据

HTML5 游戏开发进阶指南 / (印) 香卡 (Shankar, A.R.) 著; 谢光磊译. —北京: 电子工业出版社, 2013.9
书名原文: Pro HTML5 games

ISBN 978-7-121-21226-0

I. ①H… II. ①香… ②谢… III. ①超文本标记语言—游戏程序—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 186001 号

策划编辑: 张春雨 任晓露

责任编辑: 李利健

印 刷: 北京丰源印刷厂

装 订: 三河市鹏成印业有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×980 1/16 印张: 25.75 字数: 562.4 千字

印 次: 2013 年 9 月第 1 次印刷

定 价: 79.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。



译者序

香港回归那一年的某一天，我第一次在同学家的电脑上观摩了他玩《红色警戒 95》这款游戏的过程。至今我依然记得，当时他操纵着一辆吉普车和几名步兵，在地图上探索。队伍所到之处，原先覆盖着的黑色迷雾就会散去，显现出其下的地形来。说实话，我被这个效果深深地震撼了。不久，父母为家里添置了一台电脑，我也如愿以偿地玩上了这款游戏。事实上，作为 20 世纪 80 年代末出生的一批人，电子游戏伴随我度过了几乎整个少年时代。

一款优秀的游戏就好像一部引人入胜的小说（如《仙剑奇侠传》《空之轨迹》），或者一部伟大的百科全书（如《文明》《三国志》），令人击节称赞。不同的是，电子游戏带给玩家的沉浸感，比书籍带给读者的更加直接和浓烈。我无数次萌生出这样的念头：如果能制作出自己的游戏——制作游戏场景和背景音乐，设定角色与怪物的外形和技能，建立战斗系统与剧情——那多好。所以，作为读者的你，应当可以想象出，当得知有机会翻译一本关于游戏编程的书籍之时，我该有多兴奋。

这本书名为《HTML5 游戏开发进阶指南》，顾名思义，讲的是如何开发 HTML5 游戏。HTML 诞生之初仅仅是用来结构化文档信息，并创建可在浏览器上显示的网页。事实上，最初版本的 HTML 连图片都无法显示，页面仅由标题、文字段落和指向其他页面的链接组成。随着互联网的发展，HTML 逐渐变成了今天的样子。我们在花花绿绿的网页上浏览新闻、观看视频、听音乐、购物、更新 Facebook 状态、与同事聊天，甚至玩游戏——今天的 HTML，早已不是一页 A4 纸，而是能够与用户深度交互的平台。

HTML5 是最新的 HTML 标准，其诸多新特性中，最令人期待的一点莫过于新的标签 `<canvas>`。开发者可以在 `<canvas>` 中几乎不受限制地绘图，就像在 Windows 下使用 GDI 函数一样；在部分浏览器中，开发者甚至可以在 `<canvas>` 中绘制三维场景，就像使用 OpenGL 一样。结合 `requestAnimationFrame` 或 `setTimeInterval`，这个标签简直就是为游戏设计的。这本书主要介绍的就是如何使用 `<canvas>` 标签进行游戏开发。

本书手把手地带领读者实现了两款 HTML5 游戏：前 4 章实现了一个类似于《愤怒的小鸟》的游戏。后 8 章实现了一款简化版的《红色警戒》游戏。再简单的逻辑从零开始实

现，都不是一件容易的事情，对于这两款游戏，虽然看上去简单，但绝不是随随便便就能够写出来的。事实上，原作者力求这两款游戏臻于完善，并在书中详尽地阐述了游戏的细节：比如作者在绘制第一款游戏中的弹弓和橡胶带时，为了使橡胶带看上去很自然，而将弹弓分为两部分；又如，作者在第二款游戏中，为了防止车辆在地图上移动时发生“阻塞”而进行的努力。

对于熟悉前端开发，又想在前端有所作为的开发者，这是一本不可多得的好书。书中模块化的 JavaScript 代码，以及将琐碎的功能模块拼装成整体的方式，值得学习；本书介绍了很多游戏编程常用的技巧，如使用子画面页以优化性能，使用视差滚动以产生 3D 效果等。另外，还介绍了一些开源的游戏算法实现，如 JavaScript 版本的 Box2D 物理引擎、A* 路径规划算法等。最后涉及使用 Node.js 构建多人战役的方法，如果你对服务器端的 JavaScript 感兴趣，你也应该来读一读这本书。

总之，在读完本书之后，相信你一定知道如何着手去开发一款 HTML5 游戏。

谢光磊



关于作者



Aditya Ravi Shankar 是在 1993 年第一次接触计算机之后开始学习编程的。当时还没有互联网和在线教程，他在当地图书馆找到一本编程书籍，并通过艰苦地重写书中的代码，编写了他的第一个 GW-BASIC 游戏。

2001 年，他从印度理工学院毕业之后，做了近 10 年的软件顾问，为投资银行和大型企业开发交易与分析系统。最后，他离开了这个行业，这样他可以专注于从事自己所热爱的游戏开发。

从此，作为一个自信的极客，他使用各种新的语言和技术来进行自己的项目和实验，包括 HTML5。在这期间，他独立在 HTML5 平台上重建了著名的命令与征服 RTS 游戏，他也因此而著名。

除了编程，Aditya 还喜欢桌球、萨尔萨舞，以及学习发挥潜意识。他在其个人站点 (www.adityaravishankar.com) 中有一些关于游戏编程、个人发展和桌球的文章。

当他不忙于写作，或者不用专注于自己的项目时，他也会做一些咨询工作，以帮助企业来使用新的软件产品。



关于技术审核者



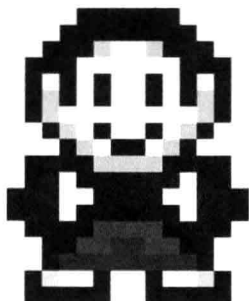
Shane Hudson 是一位自由的 Web 站点开发者，专注于最前沿的 Web 技术和 Web 标准。他曾经参与过大量交互式 Web 站点的工作并辅助项目的开发，具有丰富的使用 JavaScript 的经验。他对人工智能和计算机视觉具有浓厚的兴趣，他的 E-mail 地址是 shane@shanehudson.net。

Shane 说：“Shankar 撰写的这本书正是业界所需要的：关于如何有效并熟练地编写 JavaScript（和 Node.js）。这本书教你如何编写游戏，但它并不仅仅是一本关于游戏开发的书，还是一本教你如何编写高质量 JavaScript 代码的书。”



Josh Robinson 是一位专注于前沿技术的编程专家和自由开发者。他对编程的热爱最初来自一台二手的 Commodore 64 计算机屏幕上的蓝色荧光，后来他选择现代 Web 开发作为自己的职业。2006 年，他供职于一家网络电话供应商之时，发现了 Ruby 和 Ruby on Rails 的优雅之处，并开发了若干个 gem，其中包括流行的 countries gem。你可以通过 JoshRobinson.com 或者 Twitter: [@JoshRobinson](https://twitter.com/JoshRobinson) 联系到他。

Josh Robinson 说：“有了这本书，你就能像一个 Boss 一样写代码了。”



Syd Lawrence 经营并管理着 We Make Awesome Sh，他是 Twilio 的开发者和传播者。他获得过很多赞誉。Geek.com 曾经将他形容为现代的超级英雄，而 The Next Web 曾经说他是社交网络的领袖。



引言

欢迎阅读这本书。

我多么希望在我自己开始学习游戏开发的时候，能有这样一本书。现在，我就写了这本书。

与那些充斥着你永远用不到的抽象例程的书不同，这本书将向你提供第一手的关于 HTML5 如何被用来创建完整的可玩的游戏的资料。

我精心挑选了一款物理引擎游戏和一款即时战略游戏作为例子，这是因为，这两种类型的游戏中几乎包含了需要用来构建当下流行的各种类型游戏的所有的元素。

如果你完整地阅读本书，你将学到所有用来创建 HTML5 游戏的关键元素，并了解如何将这些元素组合在一起来实现看上去很专业的游戏。

如果你读完了本书，我希望你能够拥有自信，也拥有资源去开始创作你自己的令人惊叹的 HTML5 游戏。

本书读者

本书面向的读者将是那些已经有一些 HTML 和 JavaScript 开发经验，并且想要学习如何开发令人惊叹的 HTML5 游戏，却不知从哪里入手的人。

那些已经有一些其他语言（比如 Flash）的游戏开发经验，并想尝试 HTML5 的读者也会从本书中找到很多有价值的信息。

如果你对自己的游戏开发技能还没什么自信，也不用着急。这本书包含了用来构建游戏的所有关键元素，便于你跟随本书的章节来逐步学习如何设计大型、专业的 HTML5 游戏。如果你不能跟上，这本书也会告诉你到哪里能够找到补习的资料。

本书对 HTML5 基础、Box2D 引擎、路径规划和导航、战斗系统和敌人 AI，以及使用 Node.js 和 WebSockets 的多人对战模式，都用独立的章节进行介绍。所以，无论你编写程

序的水平如何，都能从本书中有所收获。

本书结构

本书将利用 12 章的内容向你展示创建两款完整游戏的过程。

前 4 章将创建游戏《水果战争》——一款基于 Box2D 物理引擎的游戏，它与游戏《愤怒的小鸟》极为类似。

第 1 章将讨论用来实现游戏的 HTML5 基础，如在 canvas 元素上绘图并产生动画，播放音乐，使用子画面页。

第 2 章将搭建一个基本的游戏框架，包括启动画面、主菜单、资源加载器和基于视差滚动的一个基本关卡。

第 3 章将详细介绍 Box2D 物理引擎，并展示如何使用 Box2D 来模拟游戏世界中的物理规律。

第 4 章将把游戏框架和 Box2D 引擎结合起来，添加音效和音乐，创建一个完整的可玩的物理类游戏。

接着，本书将创建一款即时战略游戏，包括单人战役模式和多人对战模式。接下来的 6 章将讨论单人战役模式。

第 5 章包含了建立基本游戏框架的内容，包括启动画面、主菜单、资源加载器和基于鼠标平移地图的基本关卡。

第 6 章向游戏中添加了不同类型的单位，如车辆、飞行器、建筑。

第 7 章使用路径规划算法和导航算法，展示如何使单位的移动更加智能。

第 8 章向游戏中添加经济系统和一个触发器系统。触发器将帮助我们编写游戏剧情。

第 9 章实现游戏的武器和战斗系统。

第 10 章在已有的框架之上，展示如何创建数个具有挑战性的单人战役关卡，并最终完成游戏单人战役模式。

最后两章将研究如何实现即时战略游戏的多人对战部分。

第 11 章将讨论使用 WebSocket API 和 Node.js 的基本方法，并创建一个多人对战大厅。

第 12 章将利用同步网络模型实现多人对战操作的框架，并在维持同步性的同时对网络延迟进行补偿。

下载源码

本书涉及的源代码可以在 Apreess 的 Web 站点 (www.apreess.com) 上下载。你可以在站点标题下方的 Source Code/Downloads 选项卡中找到本书信息页面的链接。

联系作者

如果你有任何问题或反馈,可以通过作者个人站点中本书的专页 www.adityaravishankar.com/pro-html5-games/ 与他联系。你也可以通过 E-mail: prohtml5games@adityaravishankar.com 联系他。



目 录

第 1 章 HTML5 与 JavaScript 概要	1
1.1 canvas 元素	2
1.1.1 绘制矩形	3
1.1.2 绘制复杂形状或路径	5
1.1.3 绘制文本	7
1.1.4 自定义画笔样式（颜色和纹理）	8
1.1.5 绘制图像	9
1.1.6 平移和旋转	10
1.2 audio 元素	11
1.3 image 元素	14
1.3.1 图像加载	15
1.3.2 子画面页（精灵图）	15
1.4 动画：计时器和游戏循环	17
1.4.1 requestAnimationFrame	18
1.5 小结	20
第 2 章 创建基本的游戏世界	21
2.1 基本 HTML 布局	21
2.2 创建启动画面和主菜单	22
2.3 关卡选择	26
2.4 加载图像	29
2.5 加载关卡	33
2.6 动画	34
2.7 处理鼠标输入	37
2.8 设置游戏阶段	39

2.9 小结	42
第 3 章 物理引擎基础	43
3.1 Box2D 基础	43
3.1.1 引入 Box2D	44
3.1.2 定义 World 变量	45
3.1.3 添加第一个物体：地面	46
3.1.4 绘制世界：调试绘图模式	48
3.1.5 动画	50
3.2 更多的 Box2D 元素	52
3.2.1 创建矩形物体	52
3.2.2 创建圆形物体	54
3.2.3 创建多边形物体	56
3.2.4 创建多种形状的复杂物体	58
3.2.5 连接物体的接合点	61
3.3 追踪碰撞与破坏	64
3.3.1 接触监听器	65
3.4 绘制角色	68
3.5 小结	71
第 4 章 物理引擎集成	72
4.1 定义物体	72
4.2 添加 Box2D	75
4.3 创建物体	78
4.4 向关卡加入物体	79
4.5 设置 Box2D 调试绘图	82
4.6 绘制物体	84
4.7 Box2D 动画	86
4.8 加载英雄	88
4.9 发射英雄	91
4.10 结束关卡	94
4.11 碰撞损坏	96
4.12 绘制弹弓橡胶带	99
4.13 切换关卡	101
4.14 添加声音	103

4.14.1 添加断裂和反弹的音效	104
4.14.2 添加背景音乐	107
4.15 小结	110
第 5 章 创建即时战略游戏世界	111
5.1 基本 HTML 布局	112
5.2 创建启动画面和主菜单	112
5.3 地图与关卡	118
5.4 加载任务简介画面	120
5.5 制作游戏界面	124
5.6 实现地图平移	131
5.7 小结	137
第 6 章 加入单位	138
6.1 定义单位	138
6.2 第一个物体：主基地	139
6.3 为关卡添加单位	143
6.4 绘制单位	147
6.5 添加星港	150
6.6 添加炼油厂	154
6.7 添加炮塔	157
6.8 添加车辆	160
6.9 添加飞行器	165
6.10 添加地形	170
6.11 选中游戏单位	175
6.12 强调选中的单位	178
6.13 小结	183
第 7 章 单位智能移动	184
7.1 命令单位	184
7.2 发送和接收命令	186
7.3 执行指令	188
7.4 实现飞行器移动	189
7.5 路径规划	195
7.6 定义寻径格网	196
7.7 实现车辆移动	203

7.8	碰撞检测和导航	207
7.9	将采油车展开为炼油厂	214
7.10	流畅移动	215
7.11	小结	219
第 8 章	添加更多的游戏元素	220
8.1	实现基本的经济系统	220
8.1.1	设置启动资金	220
8.1.2	实现侧边栏	223
8.1.3	获取资金	225
8.2	购买建筑和单位	226
8.2.1	添加侧边栏按钮	226
8.2.2	启用与禁用侧边栏按钮	229
8.2.3	在星港建造车辆和飞行器	232
8.2.4	从基地建造建筑	241
8.3	结束关卡	249
8.3.1	实现消息对话框	250
8.3.2	实现触发器	254
8.4	小结	259
第 9 章	添加武器和战斗	261
9.1	实现战斗系统	261
9.1.1	添加炮弹	261
9.1.2	炮塔的战斗指令	269
9.1.3	飞行器的战斗指令	274
9.1.4	车辆的战斗指令	279
9.2	创建智能的敌人	284
9.3	添加战争迷雾	287
9.3.1	定义迷雾对象	287
9.3.2	绘制迷雾	290
9.3.3	禁止在迷雾上建造建筑	293
9.4	小结	295
第 10 章	完成单人战役	296
10.1	添加音效	296
10.1.1	创建音效	296

10.1.2	命令确认音效	298
10.1.3	消息提示音	301
10.1.4	战斗音效	302
10.2	创建单人战役	303
10.2.1	救援	303
10.2.2	袭击	312
10.2.3	抵抗围攻	318
10.3	小结	329
第 11 章	WebSocket 与多人对战模式	331
11.1	使用 Node.js 操作 WebSocket API	331
11.1.1	浏览器端的 WebSocket	331
11.1.2	使用 Node.js 创建 HTTP 服务器	334
11.1.3	创建 WebSocket 服务器	336
11.2	创建多人对战游戏大厅	339
11.2.1	定义多人对战大厅界面	339
11.2.2	游戏房间列表	341
11.2.3	进入和离开房间	345
11.3	启动多人对战游戏	351
11.3.1	定义多人对战关卡	351
11.3.2	加载多人战役关卡	354
11.4	小结	359
第 12 章	多人对战游戏操作	360
12.1	同步网络模型	360
12.1.1	测量网络传输时间	361
12.1.2	发送命令	365
12.2	结束多人对战	370
12.2.1	玩家被击败时结束游戏	370
12.2.2	玩家断开连接时结束游戏	375
12.2.3	玩家丢失连接时结束游戏	376
12.3	实现玩家聊天	378
12.4	小结	382
索引		384

第 1 章



HTML5 与 JavaScript 概要

HTML5 是最新版的 HTML 标准，它提供了许多用以加强交互性和多媒体支持的新特性。这些新特性（如 canvas、audio 和 video）使得不依赖 Flash 等第三方插件而在浏览器端直接开发丰富与具有交互性的应用成为可能。

尽管 HTML5 规范仍然在制定中，但各大浏览器却已经着手实现其部分的最新特性，那些我们需要用来创建游戏的元素已经被绝大部分现代浏览器（Google Chrome、Mozilla Firefox、Internet Explorer 9 及以上版本、Safari、Opera）所支持。

若要开始创作 HTML5 游戏，你只需一款优秀的文本编辑器来写代码（我用的是 Mac 下的 TextMate——<http://macromates.com>），以及一个现代的兼容 HTML5 的浏览器（我使用的是 Google Chrome——<http://www.google.com/chrome>）即可。

HTML5 文件的结构和旧版本的 HTML 文件极为类似，不同的是，前者起始的 DOCTYPE 标签简单了许多。例 1-1 展示了一个最基本的 HTML5 文件结构，我们将在本章接下来的部分使用这个文件作为全书的起点。

例 1-1 基本的 HTML5 文件结构

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv = "Content-type" content = "text/html; charset = utf-8">
    <title > Sample HTML5 File</title>
    <script type = "text/javascript" charset = "utf-8">
      // 这个函数将在页面完全加载后调用
      function pageLoaded(){
        alert('Hello World!');
      }
    </script>
  </head>
  <body onload = "pageLoaded();">
```

```
</body>  
</html>
```

■注意 使用 body 的 onload 事件调用函数,可以确保函数运行前,页面就已经被完全加载了。这在操作 canvas 或 image 元素的时候非常重要,浏览器未完全加载页面之前就试图访问这些元素会导致 JavaScript 错误。

开始进行游戏开发之前,有必要好好复习一下用来搭建游戏的基本元素。最重要的几个元素如下。

- canvas 元素 (画布): 用于渲染游戏画面。
- audio 元素 (音频): 用于添加音效和背景音乐。
- image 元素 (图像): 用于加载游戏图像并在 canvas 中显示。
- 浏览器中的计时函数和循环函数: 用于实现动画。

1.1 canvas 元素

游戏中最重要的元素就是 HTML5 中新出现的 canvas 元素。按照 HTML5 标准的说明:“canvas 元素为脚本提供了像素级的画布,可以实时渲染图形、游戏画面或其他虚拟图像”。完整说明可在以下链接中找到: www.whatwg.org/specs/web-apps/current-work/multipage/the-canvas-element.html。

canvas 元素允许我们绘制直线、圆、矩形等基本形状,以及图像和文字,而且它已经为快速绘图做过优化了。各大浏览器都已经开始支持 GPU 加速的 2D canvas 渲染,因此,使用 canvas 绘制出的游戏动画运行速度会很快。

使用 canvas 元素十分简单,只需要把<canvas>标签加入 HTML5 文件的<body>标签中即可,如例 1-2 所示。

例 1-2 创建 canvas 元素

```
<canvas width = "640" height = "480" id = "testcanvas" style = "border:black 1px  
solid;">  
    Your browser does not support HTML5 Canvas. Please shift to another browser.  
</canvas>
```

例 1-2 创建了一个 640 像素宽、480 像素高的 canvas 元素。目前,这个 canvas 还是空的(我们指定它具有 1 像素宽的黑色边框)。我们可以使用 JavaScript 在这个矩形区域中画东西。