

移动开发经典丛书

构建适合在移动或桌面浏览器中运行的下一代多人网络游戏应用



Pro Android Web Game Apps: Using HTML5, CSS3, and JavaScript

Android Web Game App 高级编程

——使用HTML5、CSS3、JavaScript

[美] Juriy Bura 著
Paul Coates 译
明道洋 田钟晓



Apress®

清华大学出版社

014036152

TP317.6
07

移动开发经典丛书

Android Web Game App 高级编程

——使用 HTML5、CSS3、JavaScript

[美]

Juriy Bura
Paul Coates

著

明道洋 田钟晓

译



清华大学出版社



北航

C1715526

TP317.6
07

521380310

Pro Android Web Game Apps: Using HTML5, CSS3, and JavaScript

EISBN: 978-1-4302-3819-5

Original English language edition published by Apress Media. Copyright © 2012 by Juriy Bura and Paul Coates. Simplified Chinese-Language edition copyright © 2013 by Tsinghua University Press. All rights reserved.

本书中文简体字版由 Apress 出版公司授权清华大学出版社出版。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。

北京市版权局著作权合同登记号 图字: 01-2013-4604

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

Android Web Game App 高级编程——使用 HTML5、CSS3、JavaScript / (美) 杰瑞(Bura, J.), (美) 科茨(Coates, P.) 著; 明道洋, 田钟晓 译. —北京: 清华大学出版社, 2014
(移动开发经典丛书)

书名原文: Pro Android Web Game Apps: Using HTML5, CSS3, and JavaScript

ISBN 978-7-302-35103-0

I. A… II. ①杰… ②科… ③明… ④田… III. ①移动终端—应用程序—程序设计 ②超文本标记语言—程序设计 ③JAVA 语言—程序设计 IV. ①TN929.53 ②TP312

中国版本图书馆 CIP 数据核字(2014)第 009869 号

责任编辑: 王 军 于 平

装帧设计: 牛静敏

责任校对: 成凤进

责任印制: 杨 艳

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 清华大学印刷厂

装 订 者: 三河市漂源装订厂

经 销: 全国新华书店

开 本: 185mm×260mm

印 张: 33

字 数: 803 千字

版 次: 2014 年 3 月第 1 版

印 次: 2014 年 3 月第 1 次印刷

印 数: 1~4000

定 价: 69.00 元

产品编号: 052188-01

译者序

游戏就是在快乐中学会某种本领的活动

——拉夫·科斯特

当初网页游戏的发展，并不像今天一样充满机遇和一帆风顺。在客户端网络游戏鼎盛的年代，网页游戏似乎很少被人重视。玩法粗糙，画面简陋是那个年代网页游戏的体现。在客户端网游美轮美奂的画面和不同玩法面前，网页游戏显得苍白无力。直到 Web 2.0 的普及，才迎来了网页游戏的大发展，我们现在所熟知的网页游戏类型实际上是伴随着社交游戏的东风发展起来的。当“偷菜”一词成为 2009 年最为流行的网络词汇的时候，国内的网页游戏发展才如日中天。

HTML5 的到来让人兴奋不已，因为它可以将丰富多彩的桌面应用带入浏览器中。但时下到处“横行的”IE8 版本以下的浏览器，让人感觉 HTML5 技术普及之日还遥遥无期。但随着 Android、iOS 等平台的移动用户日益增多，基于 Webkit 内核的移动浏览器让 HTML5 大展宏图。即使 HTML5 和 Flash 之间的争论方兴未艾，但 HTML5 的出现必将对移动 Web 应用程序开发特别是网页游戏的开发产生重大影响。

我们望眼欲穿的 HTML5 “神器”真的来了！除了更丰富的标签，HTML5 还引入了大量新的 JavaScript API(比如 canvas API)，这些新的 API 无疑会引发新一轮的 JavaScript 框架大战，谁将成为下一个 jQuery？MooTools，Prototype 还是 YUI？此外，Node.js 的出现把客户端 JavaScript 逻辑带到了服务器端。当喧嚣过后，也许在未来的某个时间，HTML5 和 JavaScript 会像 C 一样成为一种永恒的经典。虽然这个过程中会有版本的变更、还会有喋喋不休的争议，但只要 Web 标准不变，HTML 和 JavaScript 的组合仍将是 Web 开发人员手中最强大的“武器”。

所有这些都为网页游戏的开发注入了新的活力，打开本书您就能了解这些方面的内容(当然这些只是本书的冰山一角)。本书的作者 Juriy Bura 是一位资深的游戏和实时网页应用开发方面的专家。相信通过阅读本书，读者朋友们一定能体会到他精湛的技术水平！

本书的英文原版 600 多页，几乎超过了一块板砖的厚度！但 Juriy Bura 和他的技术编辑 Paul Coates 让本书的表达深入浅出，更易于读者阅读和理解，在翻译过程中译者也试图保持这种风格。

本书的前言部分、第 1 章到第 10 章由明道洋翻译，第 11 章到 17 章以及附录 A 部分由田钟晓翻译，全书由明道洋统稿。

本书的翻译也得到了作者 Juriy Bura 热情无私的帮助，他曾多次于百忙中抽出宝贵的时间回答译者的邮件提问。

此外，感谢中科院深圳先进技术研究院的何晨光老师，以及清华大学出版社的编辑们，他们在本书的翻译过程中提出了很多宝贵的指导意见，他们的帮助确保了本书的顺利翻译和出版。

翻译是一项艰苦的工作(但也乐在其中)，虽然译者始终保持着一种审慎的态度，但囿于水平，译文中的不当之处还请广大读者批评指正。读者可通过 mingdaoyang@gmail.com 或通过此邮箱账号在 Google+ 上与译者保持联系。

译者

作者简介

Juriy Bura 是一位独立的咨询师，他经常游走于乌克兰基辅和瑞士苏黎世之间。他主要擅长桌面和移动平台上的游戏和实时网页应用开发。他是 Deadline Solutions(<http://deadline-solutions.com/about.html>)的合伙人。作为乌克兰 JavaScript User Group 的一位领导者，Juriy 经常在该组织的会议上发言。他是一位充满激情并一直谋求如何把浏览器功能发挥到极致的 Web 开发人员。Juriy 具有 7 年以上的 Java 和 JavaScript 的开发经验，他确信游戏开发领域的每一行代码都充满着最大的乐趣。

业余时间，Juriy 常和家人呆在一起，或是到一家全是“极客”的小俱乐部里与他人一起玩桌游。Juriy 的博客地址是 <http://juriy.com>，也可以通过@juriy 追随他。

Paul Coates 不仅是一位自由文字编辑，同时也是英语教师。他常工作的城市有英国特伦特河畔的柏顿和乌克兰基辅。他负责确保 Juriy 所写内容的英文表达正确，这样便于出版。Paul 教不同年龄段的学生说英语，也从事英文、俄文以及乌克兰文作品的文字编辑和校对工作。Paul 的爱好有电子游戏、电影以及旅行。

Paul 偶尔也写博客，博客地址是 <http://psykspopcornjungle.blogspot.com>，偶尔也可以通过@Psyklax 追随他。

致 谢

本书的问世得到了很多人士的大力支持和帮助。让我最感激的是我的妻子 Elena，她的爱给了我需要的灵感。当然还有我的父母——Vera Bura 和 Alexander。

十分感谢我的好友们！感谢 Vadim Voituk 的远见卓识和娴熟的专业技能！感谢 Alexey Kondratov，是他第一个告诉我一本书和 600 页博客的区别，还有 Artyom Volkhonskiy！让我欣慰的是，在我最需要支持和建议的时候，你们总在我的身边！你们的帮助对本书的日臻完善作用巨大！

特别要感谢一位卓有才华的艺术家 Sergey Lesiuk (<http://nitrokiller.deviantart.com>)。他不但对本书的封面进行了艺术设计，还为等距引擎创作了很多图像素材。特别感谢 Marcus 工作室的工作人员，他们设计了许多栩栩如生的卡通武士角色形象，他们的帮助给我们的项目带来了许多免费的艺术素材。

感谢 Apress 团队，为了本书的顺利付梓，他们做了大量的工作。感谢 Steve Anglin, Brigid Duffy、Charlie Cruz、Kimberly Burton、Anna Ishchenko、Stephen Moles、Jonathan Gennick、Jean Blackburn 以及为了本书顺利出版的幕后工作者。

感谢 Chris Nelson！他提出了很多意见和建议，还对本书的章节结构做了大量调整。正是在他的帮助之下，本书才初具雏形！和他一起工作真是一件快乐的事情！

感谢 Paul Coates 对本书做出的宝贵贡献！他爽快地抽出了周末时间对本书做了“最后一分钟”的编辑。

当然，也要谢谢你——我的小天使 Alysa！我保证以后不会说“爸爸今晚又要工作了”！

—Juriy Bura

我要感谢 Juriy，这是份很棒的工作。感谢 Apress 出版社的 Stephen 和 Chris，是他们让我们体验了写第一本书的乐趣，特别是 Sasha。没有你们，这一切都不可能！

—Paul Coates

前言

这是一本为当今最有前景的移动平台——Android 开发 JavaScript 网页游戏的书籍。游戏开发很富有挑战性，其目的是通过不同的方式来模拟生活。如果想让这种模拟更具有现实感、更令人信服，那么就需要应用很多知识和技巧。电子游戏是一个充满数学的领域，最明显的体现就是程序开发，同时也要涉及运动学、光学、声学、人工智能、艺术、音乐和讲故事等。

为什么要使用 JavaScript 和 HTML5？如果读者手里正捧着这本书，那么可能已经找到了这个问题的答案。我的这个疑问可能会让人感到不解，这是由于 JavaScript 是开发人员可以自由使用的最受欢迎的客户端跨平台解决方案。从台式计算机、智能手机到平板电脑、电视机顶盒，每一个接入互联网的设备都有浏览器，毫无疑问每个浏览器都有 JavaScript。一个用标准 HTML5 堆栈开发的应用程序可以运行在大多数平台上。想让您的游戏更快吗？想让台式计算机、移动设备以及平板电脑都能运行您的游戏吗？想让您的游戏运行在 Windows、Linux、iOS 以及 Android 平台上吗？想让用不同语言开发的游戏不需要重新编码就能运行在不同的设备上吗？所有这些问题，HTML5 都能解决！

本书的目的是让大家对最常见游戏类型背后的思想和开发方法有一个深入的理解。我不大喜欢技术手册。为了能够立竿见影，流线型的技术手册通常以牺牲重要细节为代价。然而，能快速达到目的“如何做”的方法看起来也许更快捷，但它通常会给读者带来知识缺口，并且这种缺口最终还得由读者自己去填补。当然除了最大限度地涵盖一切基本概念外，本书也有大量这种“如何做”的范例。

这就是为什么书中有很多数学的原因，但这不可避免——书中确实有那么几个数学公式。对于一个真正的游戏开发人员，不具备相当数量的数学知识是不可能的。但本书并不要求你具备非常专业的数学知识，只要掌握那些为应付各学科学习必备的数学知识就足够了。精通数学的读者，可能会觉得书中的一些解释未免太直白了，那就请随意跳过它们。

本书将刻意回避使用目前诸如 jQuery、prototype.js 或 Underscore.js 等“瑞士军刀式”的样式库。我实在不想让书中的示例和它们有什么硬性的联系，当然这里面也不乏很多优秀的库函数。每个程序开发者都有自己的个人偏好，我就发现与库无关的代码是最友好的。

本书主要内容

这是一本介绍如何使用 HTML5 和 JavaScript 进行 Android 游戏开发的书。它将指导您从一个空白 HTML 页面开始直到完成一个有动画、声音、没完没了的对话、支持多用户的

HTML5 游戏开发。

本书主要包括以下内容：

- 如何用 Canvas 元素进行一些游戏元素的绘制；如何使用“精灵”(sprites)和“精灵”表单；如何捕获用户输入。
- 如何开发激动人心的 3D 世界——包括 WebGL，一种最有前景的 Web 游戏开发 API。
- 如何借助 Node.js(一种把 JavaScript 的魅力带到服务器端的工具)开发多人游戏。
- 如何实现不同用户之间的实时通信；如何让用户在在线比赛中相互竞争。借助 JavaScript，所有这些问题都将迎刃而解，没有必要去了解其他服务器端语言就可以编写高效的服务器端代码。
- 如何让计算机控制的角色表现智能，在 AI 算法的帮助下自己做决策，找到通向游戏世界的路。
- 如何添加不错的声效。
- 如何在 Android Market 上发布我们的“杰作”。

该书将涉及游戏开发的一些优化算法，这些优化算法并不仅仅局限于 JavaScript。一旦精通了它们，将能快速地掌握其他平台上的游戏开发。理解 3D 渲染或路径查找机制，将有助于我们构建任何平台上的游戏，并不仅仅只局限于网页上的游戏开发。

总之，该书主要和游戏开发有关，同时也将告诉您如何编写世界上最激动人心的应用程序。如果这么做了，那么乐在其中！

与本书无关的内容

总的说来，本书和 Web 编程没有多大关系，书中不会涉及“HTML 是什么”或者“HTTP 如何工作”这样的问题。我假定本书的读者已经了解如何编写最基本的 JavaScript 代码并且能把代码嵌入到 HTML 页面。您不必是一位 Web 方面的开发专家，但起码要理解编程语言的一些核心概念。操作符、函数、对象以及变量这些术语对您来说应该是耳熟能详的。如果这些概念让你感觉不大舒服，也许你应该从 Terry McNavage 的 *JavaScript for Absolute Beginners* (Apress, 2010) 开始。

Terry 的这本书并不是关于如何开发游戏的，书中没有关卡创建、角色性格塑造或虚拟世界的经济学设计之类的内容，有关博弈、故事、情节、人物以及游戏设置诸如此类的任何内容同样也超出了本书的讨论范围。而这些方面的话题又是极其让人感兴趣，有一些书籍专门进行这些方面的探讨。这里我要给大家推荐这样一本书：由 Richard Rouse III 编写的 *Game Design: Theory and Practice*, Second Edition(Jones & Bartlett, 2004)。

本书读者对象

这是一本写给开发人员的书，将指导你掌握有关游戏开发方面的技术，这些技术主要包括 2D 和 3D 图形渲染、用户输入、联网、声音、人工智能以及如何应用市场发布程序。我努力让本书尽可能的实用，书中讲解的每一个概念都用示例代码进行了详细说明。可以

在 Android 智能手机和平板电脑上运行这些代码，自己动手运行代码是入门非常重要的方法。

本书适用于那些正打算为 Android 设备开发网页游戏的 Web 开发人员。本书的读者不需要有使用某个特定 JavaScript 库的任何经验——或者甚至是为移动平台搭建站点的经验。如果你知道如何从头开始制作个人主页，并且能够在里面使用一些 JavaScript。这差不多就足够了，可以开始了！

如果你是一个其他平台的游戏开发人员，并且想利用自己的经验进行 HTML5 和 Android 方面的开发，本书也是适用的。如果这样的话，本书的部分章节内容可能对于这些读者来说是非常熟悉甚至是平淡无奇的。例如，如果你有在 Java 应用开发中使用 OpenGL 的经验，可能知道什么是着色器或者如何把纹理映射到多边形，那么请随意跳过这些章节，只关注书中一些实用的方面——JavaScript 程序清单和示例。

有关艺术素材

本书有很多极好的艺术素材，特别是由 Sergey Lesiuk 创作的一些等角贴片(isometric tiles)和建筑，以及由 Marcus 工作室创作的卡通武士形象。由于这些素材完整的授权文本随文件一起分发，从而没有令人棘手的限制，无论免费项目还是商业项目都可以使用它们。

免费且没有使用限制的艺术素材对游戏开发的初期阶段是非常重要的。比起一团乱糟糟的图像，工作在一个看起来更像游戏的游戏上会让人的感觉更好。非商业化艺术素材免费共享出来的活动是由 Daniel Cook 最早发起的，Daniel Cook 通过自己的网站(www.lostgarden.com)为艺术家和艺术设计师提供免费使用的艺术素材。我支持读者朋友加入免费共享的行列，并把游戏开发的一些素材免费共享出来——开发者社区将非常感激你的善举。

本书组织结构

本书共分四部分。好笑的是：我们称每部分为一个“世界”。

2D 世界

书中这部分内容主要有关 2D 图像和 Canvas(画布)元素。我们将从第 1 章“入门”开始，配置开发所需要的工具：IDE(集成开发环境)、Web 服务器、Java SDK 以及 Android 模拟器。一旦这些配置完成，游戏开发的准备工作也就完成了。

第 2 章“浏览器中的图形：canvas 元素”将是见证奇迹开始的地方。本章将介绍如何通过 HTML5 canvas 进行图形渲染，如何使用路径和曲线、渐变和填充、变换以及 2D 上下文声明。

第 3 章“创建第一个游戏”将介绍如何创建一个“四球游戏”项目。这个小项目将使用第 2 章创建的一些元素，并向大家详细阐释一些重要的有关游戏开发的基础概念，例如：

游戏状态、机制，回合验证(turn validation)以及输/赢条件等。

现代游戏没有五颜六色的动画是不可能的。第 4 章“动画和‘精灵’”主要让我们对图像的加载过程有一个了解，指导大家一帧一帧地绘制一个跑动的动画角色。本章还将涉及更多的高级动画效果，例如，插值、加速、减速以及缓动函数等。

第 5 章“事件处理和用户输入”将介绍如何处理游戏中输入的方法，了解如何在复杂输入模式下捕获浏览器事件和创建一个高级 API(应用程序编程接口)。本章还将探讨拖放以及如何使用颜色蒙版进行完美像素的挑选。

到此，可以开发一个简单的游戏了，一些“入门工具”也被尽收囊中，所以是时候转向更高级的话题——游戏世界的渲染。

第 6 章“呈现虚拟世界”将介绍如何呈现游戏世界。将从最简单的瓦片地图(tile-map)技术开始，并逐步使之完善。将了解如何缓存地图片段，如何使用屏幕外缓冲区以及如何对诸如树木和岩石等外部物品进行呈现等内容。

第 7 章“创建等距引擎”主要介绍 2D 等距游戏引擎，是本书占用篇幅最多的一章。对于策略游戏、RPG(角色扮演游戏)、战术游戏以及其他流行的种种游戏，等距视图都是最受欢迎的游戏世界表征方法。我们将了解有关等距投射、瓦片模型以及如何对它们进行渲染的方法。除了第 6 章中讲述的技巧，本章将介绍更多的最优化渲染方法——脏矩形算法(dirty rectangles algorithm)以及世界对象的群集。本章我们将收获第二个大项目——一个用于下一个策略游戏或 RPG 开发的等距引擎。

3D 世界

这部分将介绍 3D 图形，从最基本的渲染概念到 WebGL。

第 8 章“浏览器中的 3D”将了解 3D 是什么、它如何工作以及它背后的数学原理。

第 9 章“使用 WebGL”将主要介绍 WebGL——一个正进入移动世界并且非常有前景的 Web 标准。本章可以了解如何初始化 WebGL、编写着色器、使用几何数据、载入纹理以及使用 3D 建模。

连接世界

“连接世界”部分全部与服务器通信和对话有关。从学习 Node.js 和 Express 框架开始，第 10 章“服务器端”覆盖了从 Node 安装直到一个简单游戏服务器构建的主要过程——包括如何使用合适的模板、会话处理、登录、错误处理以及通知。

第 11 章“与服务器对话”将带大家回到客户端，学习如何通过 Web 页面连接服务器以及如何同其他玩家进行数据交换。我们将了解不同的通信方式(通常也称为“传输”)，并了解它们的优缺点。

第 12 章“构建多人游戏”，本章我们将收获第三个大项目——一个多用户版本的四球游戏——使用 Node.js、Express 以及 Socket.IO。

提高世界

最后一部分主要介绍有关游戏开发的细枝末节。

第 13 章“游戏中的 AI”主要介绍人工智能，让计算机控制对手栩栩如生。你将了解有关路径发现和做决策的基本方法——一种让机器人看起来更智能的方法。

第 14 章“JavaScript 游戏引擎”主要探讨游戏引擎，同时介绍 Crafty.js，一个用 JavaScript 开发的小而精悍的游戏引擎。本章将完成第四个游戏——逃跑的武士。

第 15 章“构建原生应用”讲解了如何把一个 HTML5 游戏作为原生应用发布到 Android Market。将详细讲解这一过程的所有步骤——打包游戏，使用密钥进行签名，从做好发布准备直到发布，然后升级游戏到下一个版本。

第 16 章“添加声音”将使用 SoundManager2 加载和播放“逃跑的武士”游戏中的声音，了解 MP3 背景音乐循环播放以及如何把游戏中的事件通知给玩家等相关内容。

附录

附录 A “调试客户端 JavaScript”将探讨如何调试 JavaScript 游戏。我们努力编写优良的代码，但我们都是人，错误总是难以避免的。这个附录能让我们对如何发现错误并快速排除它们有一个好的理解，从而节省开发时间。

联系作者

如果有任何问题、意见、建议以及和本书或 HTML5 游戏开发相关的大致想法，我将非常高兴通过 juriy.bura@gmail.com 接收大家的反馈。我的网站地址是 <http://juriy.com>，或在 Twitter 上 @juriy。

目 录

第 1 章	入门	1
1.1	工具	2
1.1.1	所需要的组件	2
1.1.2	Java 开发工具包	5
1.1.3	集成开发环境	5
1.1.4	Web 服务器	10
1.1.5	Android SDK 和模拟器	12
1.2	技术	15
1.2.1	代码	16
1.2.2	面向对象编程	20
1.2.3	移动浏览器简介	28
1.3	本章小结	29
第 2 章	浏览器中的图形: canvas	
	元素	31
2.1	游戏剖析	31
2.2	在浏览器中绘图	32
2.3	HTML 基础设置	33
2.4	canvas 的概念	34
2.4.1	上下文	35
2.4.2	坐标系统	36
2.5	绘制图形	39
2.5.1	矩形	39
2.5.2	路径	41
2.5.3	子路径	48
2.6	描边和填充	50
2.6.1	纯色	50
2.6.2	渐变	50
2.6.3	图案	55
2.7	上下文状态及变换	57
2.7.1	平移	58

2.7.2	缩放	59
2.7.3	旋转	60
2.7.4	变换叠加	60
2.7.5	上下文状态	61
2.7.6	示例项目中的上下文转换	63
2.8	示例游戏项目的输出结果	64
2.9	本章小结	67
第 3 章	创建第一个游戏	69
3.1	HTML5 游戏框架	70
3.1.1	标准框架	70
3.1.2	强制方向	74
3.2	游戏架构	76
3.3	创建游戏	78
3.3.1	呈现游戏盘	78
3.3.2	游戏状态及逻辑	84
3.3.3	将组件结合在一起:Game 类	90
3.3.4	将 Game 类添加到 HTML 框架中	93
3.4	本章小结	95
第 4 章	动画和“精灵”	97
4.1	“精灵”	98
4.1.1	加载图像	99
4.1.2	绘制图像	110
4.1.3	“精灵”表单	114
4.2	动画基础	116
4.2.1	最简单的动画	117
4.2.2	JavaScript 线程模式	118
4.2.3	定时器	119
4.2.4	改善动画	123
4.3	本章小结	135

第 5 章 事件处理和用户输入..... 137

5.1 浏览器事件..... 138

5.1.1 桌面浏览器与 Android
浏览器输入..... 138

5.1.2 通过事件捕获获取用户输入..... 139

5.1.3 从事件获取更多..... 142

5.2 自定义事件..... 147

5.3 自定义事件侦听器和发射器..... 148

5.3.1 EventEmitter: 基类..... 149

5.3.2 事件与回调..... 151

5.4 自定义事件..... 152

5.4.1 实现 InputHandlerBase..... 155

5.4.2 创建 MouseInputHandler..... 158

5.4.3 创建 TouchInputHandler..... 161

5.5 高级输入..... 162

5.5.1 拖放..... 162

5.5.2 完美像素拾取和图像遮罩..... 164

5.5.3 复合操作..... 165

5.6 模拟操纵杆..... 168

5.7 本章小结..... 171

第 6 章 呈现虚拟世界..... 173

6.1 瓦片地图..... 173

6.1.1 瓦片地图背后的思想..... 174

6.1.2 实现瓦片地图..... 175

6.1.3 测量 FPS..... 180

6.2 优化渲染性能..... 182

6.2.1 仅绘制所需要的内容..... 182

6.2.2 屏幕外缓冲区..... 184

6.2.3 缓存视口周围的区域..... 187

6.3 世界对象..... 191

6.3.1 坐标系统..... 192

6.3.2 实现 WorldObjectRenderer..... 193

6.3.3 呈现顺序..... 196

6.3.4 优化..... 198

6.4 等距视图..... 200

6.5 本章小结..... 202

第 7 章 创建等距引擎..... 203

7.1 设置..... 204

7.1.1 计划..... 205

7.1.2 准备工作区..... 206

7.1.3 基本代码..... 206

7.1.4 实用函数..... 209

7.2 等距地形..... 216

7.2.1 坐标系统..... 216

7.2.2 呈现瓦片..... 217

7.2.3 实现 IsometricTileLayer..... 221

7.3 呈现对象..... 229

7.3.1 实现对象群集..... 232

7.3.2 对象缓存..... 235

7.3.3 处理移动..... 238

7.3.4 组合对象..... 240

7.3.5 对象层: 下一步骤..... 242

7.4 “脏”矩形..... 242

7.4.1 工作机制..... 243

7.4.2 实现..... 245

7.4.3 与层集成..... 249

7.4.4 创建“脏”矩形..... 252

7.5 UI 和层管理器..... 254

7.5.1 层管理器(LayerManager)..... 255

7.5.2 UI..... 257

7.6 交互..... 261

7.6.1 事件传播和处理..... 262

7.6.2 停止传播..... 265

7.7 本章小结..... 267

第 8 章 浏览器中的 3D..... 269

8.1 3D 渲染简介..... 269

8.2 3D 渲染的工作机制..... 271

8.2.1 数学..... 271

8.2.2 一个 3D 示例..... 271

8.3 “Hello World” 3D 引擎..... 273

8.3.1 模型和场景..... 273

8.3.2 渲染..... 275

8.4 本章小结..... 286

第 9 章 使用 WebGL..... 289

9.1 WebGL 的基础..... 289

9.1.1 初始化 WebGL..... 290

9.1.2 几何体	292	11.3 为基本的 HTTP 请求使用 XMLHttpRequest API	368
9.1.3 OpenGL ES 2.0 渲染管线	294	11.3.1 普通 XHR HTTP 请求	369
9.1.4 使用缓冲区	295	11.3.2 XHR 错误处理	370
9.1.5 着色器和 GLSL	297	11.3.3 XMLHttpRequest Level 2	371
9.2 基本示例: 渲染 3D 立方体	302	11.3.4 使用二进制数据	372
9.2.1 在 Web 页面中使用着色器	302	11.4 反向 Ajax	374
9.2.2 渲染 “Hello World”	304	11.4.1 存在的问题	375
9.3 探索 WebGL	309	11.4.2 解决方案	375
9.3.1 颜色	309	11.4.3 最佳解决方案	375
9.3.2 纹理	313	11.4.4 可接受的解决方案	377
9.4 本章小结	320	11.4.5 过时的解决方案	381
第 10 章 服务器端	323	11.5 现场测试 “传输”	382
10.1 Node.js 基础	324	11.6 本章小结	384
10.1.1 Node.js 概述	324	第 12 章 构建多人游戏	387
10.1.2 编程模式	325	12.1 网络游戏构成	387
10.1.3 安装 Node.js	327	12.1.1 游戏的体系结构: 从单机版 到多人版	388
10.1.4 调试 Node 脚本	328	12.1.2 项目结构	391
10.2 编写 Node.js 脚本	330	12.2 Socket.IO 游戏大厅	392
10.2.1 异常和堆栈跟踪	330	12.2.1 客户端-服务器通信	393
10.2.2 全局命名空间和 Node 模块	331	12.2.2 添加游戏大厅屏幕	396
10.2.3 编写第一个模块	335	12.3 添加游戏设置	401
10.2.4 发现模块	337	12.3.1 在客户端和服务器之间 共享逻辑	401
10.2.5 使用 NPM	338	12.3.2 服务器端	403
10.3 回归现实: 构建游戏服务器	341	12.3.3 客户端	408
10.3.1 Node 的 Web 开发框架	341	12.4 本章小结	415
10.3.2 基本输出	341	第 13 章 游戏中的 AI	417
10.3.3 呈现 Web 页面	345	13.1 AI 对于游戏的重要性	418
10.4 理解中间件	353	13.2 路径查找概述	418
10.5 内务处理	355	13.3 图论	420
10.5.1 错误报告	355	13.3.1 图的概念	420
10.5.2 日志	359	13.3.2 在 JavaScript 中实现图	422
10.5.3 服务器配置	361	13.4 构建路径查找 AI	426
10.6 本章小结	363	13.5 路径查找图的构造方法	432
第 11 章 与服务器对话	365	13.5.1 全手工图	432
11.1 浏览器中网络通信的演变	365	13.5.2 光线投射(视线法)	433
11.2 服务器设置	367		

13.5.3	导航网格	433	15.5.3	对该应用程序签名	478
13.6	决策	434	15.5.4	在 Google Play 上发布	479
13.7	本章小结	438	15.5.5	更新应用程序	483
第 14 章	JavaScript 游戏引擎	439	15.6	本章小结	484
14.1	图形 API、库以及游戏引擎	439	第 16 章	添加声音	485
14.1.1	图形 API	440	16.1	网页中的音频	486
14.1.2	图形库	440	16.1.1	audio 标签	486
14.1.3	游戏引擎	441	16.1.2	Web Audio API	487
14.2	Crafty	442	16.1.3	Android 浏览器中的 声音	488
14.2.1	实体组件系统	443	16.2	使用 SoundManager2	489
14.2.2	Crafty Hello World	446	16.2.1	初始设置	489
14.2.3	Crafty 游戏	449	16.2.2	循环	491
14.2.4	加载“精灵”	449	16.2.3	添加游戏声音	493
14.2.5	呈现地形和对象	451	16.3	在 Cordova 应用程序中 播放声音	495
14.2.6	动画	453	16.4	用户体验	496
14.2.7	与实体和事件系统进行 交互	454	16.5	本章小结	497
14.3	最终版本	455	16.6	进一步探索	497
14.4	本章小结	458	附录 A	调试客户端 JavaScript	499
第 15 章	构建原生应用	459			
15.1	原生应用	460			
15.2	设置 Apache Cordova (PhoneGap)	461			
15.2.1	安装 Cordova	462			
15.2.2	安装 Apache Ant	462			
15.3	构建原生应用	463			
15.3.1	创建一个空的 Android 项目	463			
15.3.2	对空 Android 项目进行 测试	464			
15.3.3	Cordova 项目的基础	465			
15.3.4	联网	469			
15.3.5	最后的“润色”：名称、 图标和全屏模式	470			
15.4	使用原生 API	473			
15.5	准备市场	476			
15.5.1	应用程序签名	476			
15.5.2	创建数字密钥	477			

入门

本章的目标是为开发工作准备一个适宜的工作环境。移动开发所需要的环境和工具总要比一般的桌面项目复杂一些。当提到 Android 和 JavaScript 应用程序开发时，选择一款能够高效开发的工具意义重大。但工作环境并不只局限于工具，对于开发过程而言，设置编码标准以及遵循最佳的编码实践同样非常重要。本章也将对有关的编码约定和基本架构决策进行探讨。

作为一名经验丰富的开发人员，可能我们已经对某种开发工具和编码方法有着自己个人的某种偏好，例如每个 Web 开发人员都有自己最喜欢的集成开发环境(IDE)以及用于基本测试的浏览器。对于如何编写可维护的 JavaScript 代码，可能我们也有自己的见解。如果这种偏好让你感觉很“舒服”，那么就使用它！但我还是鼓励大家尝试一下本章所介绍的一些工具和技术，其中的一些工具可能更方便。

本章分为开发工具和技术两个部分，其中的每一部分都展现了开发工作的一个重要方面。本章主要包括以下内容：

- 工具

- 安装 Java 开发工具包
- 对各种 IDE 的 JavaScript 支持度进行比较
- 安装 Web 服务器 nginx
- 安装 Android SDK 并对模拟器进行配置
- 创建一个基本的 Web 页面并确保能够通过桌面浏览器、移动设备以及模拟器对其进行加载

- 技术

- 回顾 JavaScript 最佳编码实践
- 实现一个简单的继承机制，并将其应用于书中的 OOP 代码