

译著双馨，资深游戏开发专家和HTML5技术工程师亲自执笔，HTML5布道者和资深游戏开发专家联袂翻译

经典HTML5游戏开发教程，通过引人入胜的游戏案例，全面展示使用HTML5和CSS3构建各种常见类型游戏的开发思路和设计方法

HTML5 Games Development by Example: Beginner's Guide

HTML5游戏 开发实战

(美) Makzan 著

吕定平 陈升想 译



机械工业出版社
China Machine Press

HTML5 Games Development by Example: Beginner's Guide

HTML5游戏 开发实战

(美) Makzan 著

吕定平 陈升想 译



机械工业出版社
China Machine Press

这是一本经典的 HTML5 游戏开发教程，它清晰而全面地展示了如何使用最新的 HTML5 和 CSS3 标准来构建各种常见类型的游戏。本书实战性非常强，围绕纸牌游戏、绘图游戏、物理游戏和多人游戏精心组织了 6 个富有趣味性和技术性的游戏案例。如果动手实践这些案例，不仅能掌握各种类型游戏的开发思路和设计方法，而且还能掌握 HTML5 和 CSS3 技术中与游戏开发相关的理论知识。本书共分 9 章，第 1 章介绍 HTML5、CSS3，以及相关的 JavaScript API 新功能；第 2 章通过使用 DOM 和 jQuery 创建传统游戏；第 3 章讨论如何用 DOM 和 CSS3 来创建游戏；第 4 章介绍在页面如何通过 Canvas 来绘制游戏并与之交互；第 5 章介绍如何在 Canvas 中绘制渐变效果和加入图像；第 6 章使用 Audio 元素给游戏添加声音效果和背景音乐；第 7 章使用新的本地存储 API 保存和恢复游戏进度，进一步增强游戏；第 8 章讨论如何实现多人联网玩游戏；第 9 章讲解如何把 Box2D 集成到 Canvas 游戏中。

本书以实例为导向，系统介绍网络游戏开发技术，结合具体示例的操作步骤讲解，浅显易懂，适合网络游戏开发人员、管理人员阅读。

Makzan: HTML5 Games Development by Example Beginner's Guide (ISBN: 978-1-849691-26-0)

Copyright © 2011 Packt Publishing. First published in the English language under the title "HTML5 Games Development by Example: Beginner's Guide".

All rights reserved.

Chinese simplified language edition published by China Machine Press.

Copyright © 2012 by China Machine Press.

本书中文简体字版由 Packt Publishing 授权机械工业出版社独家出版。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。

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：01-2011-7737

图书在版编目（CIP）数据

HTML5 游戏开发实战 / (美) 马克赞 (Makzan) 著；吕定平，陈升想译. —北京：机械工业出版社，2012.7
书名原文：HTML5 Games Development by Example: Beginner's Guide

ISBN 978-7-111-39176-0

I. H… II. ①马… ②吕… ③陈… III. 超文本—标记语言—游戏—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2012) 第 163956 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：谢晓芳

北京市荣盛彩色印刷有限公司印刷

2012 年 8 月第 1 版第 1 次印刷

186mm×240mm·17.25 印张

标准书号：ISBN 978-7-111-39176-0

定价：59.00 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

审校者简介

Matteo Ferrett 是一位软件工程师、一个业余漫画艺术家，在他设计自己的第一个视频游戏时还客串了一下平面设计师。

在 20 世纪末，他开始爱上 JavaScript，到现在仍热情不减。

他曾在 TomTom 担任两年多的高级软件工程师以及技术主管，目前，正在 Mozilla 任职。

非常感谢我的爱人：Elisa，感谢她的耐心和理解。我还要感谢我的父母，感谢他们一直以来对我的支持。

Henk Jurriens 是一位软件开发人员以及程序开发布道者，特别是 Java、Groovy 和 Grails 以及 HTML5 方面。Henk 对于新技术非常感兴趣，并且喜欢谈论这些新技术，经常给我们带来有关 HTML5、Linked Data(键连资料) 以及 Groovy 和 Grails 等不同技术的演讲。

另外，Henk 还创办了一个 HTML5 用户组来推动 HTML5 知识的分享。今年，还举办了几次不同的聚会。并与 Google 技术用户组一起组织了一次 HTML5 编程马拉松 (Hackathon) 活动。

去年，Henk 还帮助举办了一届 HTML5 游戏制作节，在那里他看到了许多有潜力的 HTML5 游戏，这也是他为本书作评审的一个原因。有了本书的帮助，相信精彩的 HTML5 游戏将会诞生！

William Malone 是一位专注于 ECMAScript 相关语言（ActionScript 和 JavaScript）的软件开发人员。他写了许多有关 Flash 和 HTML5 方面有关的论文，可以从下面的网址找到这些论文：<http://www.williammalone.com>。

译者序

在翻译本书之前，其实已经看完这本书了，觉得作者写得非常棒！当陈升想来找我一起合译本书时，很荣幸能参与到这本书的翻译工作中。

本书内容安排得很合理，各章节衔接得非常好，而且学习起来没有什么困难，非常适合入门。同时，作者所介绍的例子又是经过精心筛选的，看完本书后，会对各类型的游戏如何设计有一个粗略的了解。虽然，对于高级的开发人员来说，本书未免显得有些初级，但是，我个人认为，作者写书目的非常明确，考虑的受众就是那些没有太多开发经验的开发者。有明确的目的，才会让读者倍感贴心。

在翻译过程中，我曾遇到过很多的困难，感受最深的就是表达个人语言的困难，有的时候，明白作者所表述的意思，但是用中文来表达时，就非常吃力。为什么呢？这也许与我们程序员的职业有关。在前段时间参加的 Cocos2d-x 开发者大会，在大会提问环节中，两个程序员的程序式对话，让我体会到了这个原因。作为一个老的程序员，平时都在用代码与计算机进行沟通，久而久之，就养成了程序式的表述，与人沟通起来反而觉得有些生疏了。正是由于这个原因，在此也恳请读者的谅解，如果在阅读过程中发现文章表述有什么问题，也请多多包涵。

由于本人也是一个 HTML5 新手，翻译过程中难免会有一些错误和问题，还希望各位 HTML5 高手批评指正。如果你觉得翻译中有任何错误和问题，请你及时反馈给我，以便及时地进行修正。

翻译本书也是一份体力劳动，经常需要坐在电脑前忙到深夜。这里要感谢陈升想提供了这次合作机会，同时也感谢机械工业出版社华章分社各位编辑的细心指导与帮助。最后我的妻子和宝宝，感谢她们对我工作的支持与生活上的照顾！

最后，希望本书能对你的 HTML5 游戏开发有所帮助。

吕定平

前言

HTML5 有望成为网络游戏开发的热门新平台。HTML5 游戏能够运行于包括 iPhone 系列和 iPad 系列在内的计算机、智能手机以及平板电脑上。今天，我们加入 HTML5 游戏开发的急先锋，明日将成为时代的弄潮儿。

本书将展示如何使用最新的 HTML5 和 CSS3 Web 标准来构建纸牌游戏、绘图游戏、物理游戏，甚至是多人网络游戏。本书讲解清晰、系统全面，参照本书的讲解就能构建出 6 款示例游戏。

HTML5、CSS3 以及相关的 JavaScript API 是网络上最新的热点话题。这些标准给我们带来一个新的游戏市场——HTML5 游戏。通过新的功能，我们能用 HTML5 元素、CSS3 属性，以及 JavaScript 来设计运行于浏览器的游戏。

本书根据关注话题的侧重点而分成 9 章。我们将通过创建 6 款游戏来具体学习如何绘制游戏对象、绘制动画、加入音频、连接玩家，以及用 Box2D 物理引擎构建游戏。

本书内容：

第 1 章介绍 HTML5、CSS3，以及相关的 JavaScript API 新功能。该章还演示了利用这些功能和特性能创建什么样的游戏。

第 2 章通过使用 DOM 和 jQuery 创建传统《乒乓球》游戏来开始我们的游戏开发之旅。

第 3 章探讨 CSS3 新功能，讨论如何用 DOM 和 CSS3 来创建《纸牌记忆配对》游戏。

第 4 章介绍在页面中通过新 Canvas 元素来绘制游戏并与之进行交互的新方式，还演示如何 Canvas 来构建《解题》游戏。

第 5 章通过在 Canvas 中绘制渐变效果和加入图像进行美化，完善前面介绍的《解题》游戏。该章还讨论基于帧的动画精灵和多层管理方法。

第 6 章使用 Audio 元素给游戏添加声音效果和背景音乐。该章还讨论各浏览器对声音格式的兼容性，并在该章结尾完成一款由键盘驱动的音乐游戏。

第 7 章使用新的本地存储 API 保存和恢复游戏进度，记录最佳成绩功能，进一步增强第 3 章所创建的 CSS3《纸牌记忆配对》游戏。

第 8 章讨论能让浏览器持久连接到 socket 服务器的新 WebSocket API，用于实现多人联网玩游戏，并在该章完成《我画你猜》游戏。

第 9 章传授如何把 Box2D(著名物理引擎)集成到 Canvas 游戏中。该章讨论如何创建物

体、应用力、连接物体，以及如何把图形与物体进行关联，并最终创建一款平台汽车游戏。

阅读本书你需要做的准备

你需要最新的现代 Web 浏览器，一个优秀的文本编辑器，HTML、CSS 以及 JavaScript 的基本知识。

本书读者对象

本书适合于对 HTML、CSS 和 JavaScript 有基本理解，并希望设计网页类 Canvas 游戏、基于 DOM 游戏的设计者。

本书约定

在本书中，你会发现几个经常出现的标题。

为了明确指示如何完成一个过程或任务，本书使用如下标题：

实战时间

- 1) 动作 1
- 2) 动作 2
- 3) 动作 3

对游戏代码经常需要一些附加说明来解释其意图，有如下标题：

刚刚发生了什么

这个标题解释你刚刚完成的任务或代码的工作。

你还会在本书中发现其他的一些学习辅助内容，包括：

突击测验

包含一些简短的选择题，旨在测试你对问题的理解。

牛刀小试

在此设置一些挑战题目，并让你利用前面的知识来实验你的想法。

本书还用文本样式来区分信息类型。下面有一些不同样式的示例，以及它们所代表的意思。

文本中出现的代码格式如下：“我们的 HTML5 游戏开发之旅将从 `index.html` 开始。”

代码块设置格式如下：

```
// 开始游戏
var date = new Date();
audiogame.startingTime = date.getTime();

// 一段时间后
var date = new Date();
var elapsedTime = (date.getTime() - audiogame.startingTime)/1000;
```

当我们想提醒你注意代码块中的特定部分，会将相关的行或项设置为**粗体**：

```
function setupLevelData()
{
    var notes = audiogame.leveldata.split(";");

    // 保存点的数量
    audiogame.totalDotsCount = notes.length;

    for(var i in notes)
    {
        var note = notes[i].split(",");
        var time = parseFloat(note[0]);
        var line = parseInt(note[1]);
        var musicNote = new MusicNote(time,line);
        audiogame.musicNotes.push(musicNote);
    }
}
```

一些命令行的输入或输出写成这样：

```
$ ./configure
$ sudo make install
```

新术语和重要的语句也会用**粗体**显示。你在屏幕上的菜单或对话框中看到的单词，出现在文本中的样式将像这样：“你将获取到一个多用户画板的介绍页面。在 **Launch Experiment**（启动实验）选项上右击，并选择 **Open link in new window**（在新窗口中打开链接）”。



警告或重要注释出现于这样的框中。



提示与技巧这样呈现。

读者反馈

我们十分欢迎读者反馈的意见。我们想知道你对本书的看法：喜欢哪些部分，不喜欢哪些部分。这些反馈对于协助我们编写出真正对读者有所裨益的书至关重要。

给我们发送反馈意见，只需发送电子邮件到 feedback@packtpub.com，并在你的邮件标题中提及书名即可。

如果你确实需要并希望我们能够出版其他领域的书，请在 www.packtpub.com 中的 **SUGGEST A TITLE** 表格内留言，或者发送邮件至 suggest@packtpub.com。

如果你在某方面有所专长并且愿意参与图书编写，请参阅我们的作者指南，参见 www.packtpub.com/authors。

客户支持

既然你已经拥有了这本由 Packt 出版的图书，为了让你的付出得到最大的回报，我们还为你提供其他许多方面的服务。

下载本书示例代码

你可以在 <http://www.PacktPub.com> 上通过你的账号下载你购买的所有 Packt 图书的示例代码文件。如果你在其他地方购买，你可以访问 <http://www.PacktPub.com/support> 并进行注册，我们将把文件通过电子邮件形式直接发送给你。

勘误

尽管我们已经竭尽全力确保本书内容准确，但错误终难避免。如果你在我国的书中发现任何错误（无论是出现在正文还是代码中），我们都非常感谢你将错误提交给我们。通过这样做，你可以帮助其他读者解开谜团并提高本书后续版本的质量。如果需要提交勘误，请访问 <http://www.packtpub.com/support>，选择相应的书名，单击 **errata submission form** 链接，就可以开始输入你的勘误信息了。一旦通过验证，我们将接受你的提交，同时勘误内容也将上传到我们的网站，或者添加到对应书目勘误区的现有勘误表中。任何图书当前的勘误都可以通过 <http://www.packtpub.com/support> 来查看。

侵权行为

各种媒体在 Internet 上一直饱受版权侵害的困扰。Packt 坚持对版权和授权进行全力保护。如果你在 Internet 上发现我社图书的任何形式的盗版，请立即为我们提供地址或网站名称，以便我们采取进一步的措施。

请将涉嫌侵权的链接通过 copyright@packtpub.com 联系我们。

我们感谢您为保护我们作家所提供的帮助，以便我们有能力为您带来更多有价值的内容。

疑难解答

如果你对书中存在任何疑问，你可以通过 questions@packtpub.com 联系我们，我们会尽力解决问题。

目 录

审校者简介

译者序

前言

第 1 章 HTML5 游戏概述 1

1.1 探索 HTML5 新功能 1

1.1.1 Canvas 2

1.1.2 音频 2

1.1.3 GeoLocation 2

1.1.4 WebGL 2

1.1.5 WebSocket 3

1.1.6 本地存储 4

1.1.7 离线应用程序 4

1.2 探索 CSS3 新功能 5

1.2.1 CSS3 转换 5

1.2.2 CSS3 变换 7

1.2.3 CSS3 动画 7

1.3 HTML5 和 CSS3 新功能的更多 细节 8

1.4 创建 HTML5 游戏的好处 8

1.4.1 不需要第三方插件 9

1.4.2 不需要插件就能支持 iOS 设备 9

1.4.3 突破常规浏览器游戏限制 9

1.4.4 创建 HTML5 游戏 9

1.5 HTML5 还能做些什么 11

1.5.1 《记忆配对》游戏 11

1.5.2 Sinuous 11

1.5.3 小行星式书签小程序 12

1.5.4 Quake 2 12

1.5.5 蝌蚪聊天室 13

1.5.6 Scrabb.ly 13

1.5.7 Aves 引擎 14

1.6 浏览更多 HTML5 游戏 15

1.7 本书主要涉及哪些游戏 15

1.8 总结 16

第 2 章 DOM 游戏开发入门 17

2.1 准备开发工具 18

2.2 为 DOM 游戏准备 HTML 文档 18

2.2.1 新式 HTML5 doctype 声明 19

2.2.2 页眉和页脚 20

2.2.3 JavaScript 代码最佳位置 20

2.2.4 页面加载完后运行代码 20

2.3 设置《乒乓球》游戏元素 21

2.3.1 jQuery 简述 23

2.3.2 jQuery 选择器基础知识 23

2.3.3 jQuery css 函数 24

2.3.4 使用 jQuery 的好处 25

2.3.5 使用 jQuery 操纵 DOM 游戏元素 25

2.3.6 绝对位置的行为 26

2.4 获取玩家的键盘输入 27

2.4.1 了解按键代码	29	3.3.4 使用 jQuery 的子节点过滤器 选择首个子元素	61
2.4.2 让常数更具可读性	29	3.3.5 垂直对齐 DOM 元素	62
2.4.3 用 parseInt 函数将字符串 转换为数字	30	3.3.6 通过背景位置来使用 CSS 精灵	62
2.4.4 在控制台面板中直接执 行 JavaScript 表达式	31	3.4 给配对游戏添加游戏逻辑	63
2.4.5 检查控制台窗口	32	3.4.1 在 CSS 转换完后执行代码	66
2.5 支持玩家多键盘同时输入	32	3.4.2 翻牌后延迟代码的执行	67
2.5.1 更好地声明全局变量	34	3.4.3 在 JavaScript 中随机化数组	67
2.5.2 用 setInterval 函数创建 JavaScript 定时器	35	3.4.4 通过 HTML5 自定义数据 属性保存内部自定义数据	68
2.5.3 理解游戏主循环	35	3.4.5 用 JQuery 访问自定义数据属性	68
2.6 通过 JavaScript 间隔移动 DOM 对象	35	3.4.6 制作其他纸牌游戏	70
2.7 开始碰撞检测	37	3.5 在游戏中嵌入 Web 字体	70
2.8 动态显示 HTML 文本	41	3.6 总结	73
2.9 总结	43	第 4 章 用 Canvas 和绘图 API 构建 《解题》游戏	74
第 3 章 用 CSS3 构建《纸牌记忆配对》 游戏	44	4.1 介绍 HTML5 Canvas 元素	75
3.1 用 CSS3 转换和变换模块移动 游戏对象	44	4.2 在 Canvas 中绘制圆形	75
3.1.1 2D 变换函数	47	4.2.1 当 Web 浏览器不支持 Canvas 时的反馈信息	77
3.1.2 3D 变换函数	48	4.2.2 用 Canvas 的 arc 函数绘制 圆和图形	78
3.1.3 用 CSS3 转换实现样式间 过渡	49	4.2.3 把角度转换为弧度	78
3.2 创建翻牌效果	50	4.2.4 在 Canvas 中执行路径绘制 的操作	82
3.2.1 使用 jQuery toggleClass 函数切换类	52	4.2.5 在切换路径样式时先 调用 BeginPath	82
3.2.2 通过 z-index 控制重叠元素 的可见性	53	4.2.6 关闭路径	83
3.2.3 介绍 CSS perspective 属性	53	4.2.7 将画圆功能封装成函数	83
3.2.4 介绍背面可见性	54	4.2.8 在 JavaScript 中生成随机数	85
3.3 创建《纸牌记忆配对》游戏	55	4.2.9 保存圆的位置	85
3.3.1 下载纸牌精灵表图像	55	4.3 在 Canvas 中画线	87
3.3.2 设置游戏开发环境	56	4.4 通过 Canvas 中的鼠标事件与 绘制对象交互	90
3.3.3 使用 jQuery 复制 DOM 元素	61	4.4.1 在 Canvas 元素中获取鼠标位置	93

4.4.2 在 Canvas 中检测鼠标事件是否发生在圆上	93	6.2.3 为音乐游戏选择正确的歌曲	149
4.4.3 游戏主循环	95	6.2.4 存储和提取歌曲关卡数据	149
4.4.4 清除 Canvas	95	6.2.5 获取游戏的流逝时间	150
4.5 在 Canvas 中检测线的交点	96	6.2.6 创建音乐点	150
4.6 制作《解题》游戏	100	6.2.7 移动音乐点	151
4.6.1 定义关卡数据	105	6.3 将 PLAY 按钮与音乐游戏场景链接起来	152
4.6.2 判断是否过关	105	6.4 构建键盘驱动的迷你钢琴音乐游戏	155
4.6.3 显示当前关卡和完成进度	105	6.4.1 通过按键来击打三条音乐线	156
4.7 总结	106	6.4.2 判断按键是否击中音乐点	157
第 5 章 构建大师级 Canvas 游戏	107	6.4.3 通过给定索引将元素从数组中移除	158
5.1 用渐变色来填充图形	108	6.5 给迷你钢琴游戏添加额外的功能	159
5.1.1 给渐变添加色标	109	6.5.1 根据玩家的表现而调整音乐音量	159
5.1.2 填充径向渐变色	110	6.5.2 从游戏中删除音乐点	161
5.2 在 Canvas 中绘制文本	112	6.5.3 保存最近 5 个音乐点的成功率计数	162
5.3 在 Canvas 中绘制图像	116	6.5.4 记录音符来得到关卡数据	162
5.3.1 使用 drawImage 函数	120	6.6 音乐播放完后处理 audio 事件	164
5.3.2 装点 Canvas 游戏	121	6.7 总结	166
5.4 在 Canvas 中播放精灵表动画	123	第 7 章 利用本地存储技术保存游戏数据	167
5.5 创建多层 Canvas 游戏	127	7.1 使用 HTML5 本地存储技术保存数据	168
5.6 总结	132	7.1.1 创建游戏结束对话框	168
第 6 章 给游戏添加声音效果	134	7.1.2 在浏览器中保存成绩	171
6.1 给 PLAY 按钮添加声音效果	135	7.1.3 通过本地存储技术保存和加载数据	172
6.1.1 定义 audio 元素	137	7.1.4 本地存储只保存字符串值	173
6.1.2 播放声音	138	7.1.5 将本地存储对象看做关联数组	173
6.1.3 暂停声音	138	7.2 在本地存储中保存对象	174
6.1.4 调整音量大小	139	7.2.1 在 JavaScript 中获取当前日期和时间	177
6.1.5 使用 jQuery 的 hover 事件	139		
6.1.6 创建 Ogg 格式的音频以支持 Mozilla Firefox	140		
6.1.7 不同 Web 浏览器所支持的音频格式	141		
6.2 构建迷你钢琴音乐游戏	141		
6.2.1 在 HTML5 游戏中创建场景	144		
6.2.2 让音乐播放可视化	144		

7.2.2 使用原生 JSON 将对象 编码成字符串	178	8.4 通过在服务端广播接收到的消息 来创建聊天室	202
7.2.3 从 JSON 字符串中加载 所保存的对象	178	8.5 使用 Canvas 和 WebSocket 制作 共享绘图板	205
7.2.4 在控制台窗口中检测本地存储	179	8.5.1 构建本地绘图板	205
7.3 用一种漂亮的彩带效果来告诉 玩家破记录了	180	8.5.2 广播绘图数据给所有已连接的 浏览器	208
7.4 保存整个游戏的进度	183	8.5.3 定义用于在客户端与服务器之间 进行通信的数据对象	211
7.4.1 保存游戏进度	183	8.5.4 将画线数据打包成 JSON 以便 于广播	211
7.4.2 从本地存储中删除记录	186	8.5.5 再现从其他客户端接收到的 画线数据	212
7.4.3 在 JavaScript 中复制数组	186	8.6 构建多人游戏:《我画你猜》	212
7.4.4 恢复游戏进度	187	8.6.1 控制多人游戏的游戏流程	217
7.5 总结	189	8.6.2 在服务器端罗列出所有已 连接客户端	218
第 8 章 利用 WebSocket 构建多人 游戏——《我画你猜》	190	8.6.3 在服务器端发送消息给指定的 连接	218
8.1 初试 WebSocket Web 应用程序	190	8.6.4 改进现有游戏	218
8.2 安装 WebSocket 服务器	192	8.7 用 CSS 装点《我画你猜》游戏	219
8.2.1 安装 Node.JS WebSocket 服务器	193	8.8 总结	221
8.2.2 创建广播连接数的 WebSocket 服务器	194	第 9 章 用 Box2D 和 Canvas 构建 物理类汽车游戏	222
8.2.3 初始化 WebSocket 服务器	195	9.1 安装 Box2d JavaScript 库	223
8.2.4 在服务器端监听连接事件	195	9.1.1 用 b2World 创建新的物理 世界对象	226
8.2.5 在服务器端获取已连接的 客户端数	196	9.1.2 用 b2AABB 定义物理边界	226
8.2.6 向所有已连接的浏览器 广播消息	196	9.1.3 给物理世界设置重力	226
8.2.7 创建客户端来连接 WebSocket 服务器并获取总连接数	196	9.1.4 设置 Box2D 忽略休眠物体	227
8.2.8 建立 WebSocket 连接	198	9.2 在物理世界中创建静态地面	227
8.2.9 WebSocket 客户端事件	198	9.2.1 创建物体形状	228
8.3 使用 WebSocket 构建聊天室	199	9.2.2 创建物理物体	228
8.3.1 向服务器发送消息	199	9.3 在 Canvas 上绘制物理世界	229
8.3.2 从客户端发送消息到服务器	201	9.4 在物理世界里创建动态的长方形	232
8.3.3 在服务器端接收消息	201		

9.5 推进物理世界的时间.....	233	9.12.2 根据物理物体的状态来绘制 每一帧图像.....	250
9.6 给游戏安装车轮.....	234	9.12.3 在 Canvas 中旋转和翻转 图片.....	251
9.7 创建物理汽车.....	235	9.13 给游戏添加装饰, 让游戏更具 趣味性.....	251
9.8 通过键盘给汽车施加动力.....	237	9.13.1 用燃料的限制加速.....	256
9.8.1 向物体施加动力.....	238	9.13.2 用 CSS3 进度条显示剩余 燃料.....	256
9.8.2 理解 ApplyForce 与 ApplyImpulse 之间的不同点.....	239	9.14 总结.....	257
9.8.3 给游戏环境添加坡道.....	239	9.14.1 HTML5 游戏引擎.....	258
9.9 在 Box2D 世界检测碰撞.....	240	9.14.2 游戏精灵以及贴图.....	258
9.10 重启游戏.....	242	9.14.3 声音效果.....	258
9.11 让游戏支持关卡.....	244	附录 突击测验答案.....	259
9.12 为 Box2D 绘制图形轮廓.....	247		
9.12.1 使用形状和物体的 userData 属性.....	250		

第 1 章 HTML5 游戏概述

超文本标记语言（HTML）在过去几十年中塑造了 Internet。它定义了 Web 页面结构以及相关网页如何链接。HTML 经历了从 HTML 2.0 到 HTML 4.01，以及后来的 XHTML 1.1 的发展。感谢那些 Web 应用程序和社交网络应用，让 HTML5 隆重登场。

层叠样式表（CSS）定义了网页的视觉表现样式。它能定义所有 HTML 元素以及这些元素状态的样式，如鼠标指针悬停和激活状态。

JavaScript 是 Web 页面的逻辑控制器，它创建动态页面，并且让用户与客户端页面进行交互。它通过文档对象模型（DOM）访问 HTML 元素，并给 HTML 元素重新应用不同的 CSS 样式。

这三项技术给我们带来了新的游戏市场——HTML5 游戏。因为有了它们的新功能，就可以用 HTML5 元素、CSS3 属性以及 JavaScript 设计在浏览器上玩的游戏。

本章主要内容：

- 探索 HTML5 新功能。
- 探讨 HTML5 和 CSS3 让人兴奋的新特性。
- 用 HTML5 设计出的游戏。
- 预览一下后续章节中将要构建的游戏。

那么，让我们开始 HTML5 之旅吧！

1.1 探索 HTML5 新功能

本节介绍 HTML5 以及 CSS3 的很多新功能。在开始动手制作游戏前，需要先对这些新功能有一个总体的认识，并看看用它们能创建什么样的游戏。

1.1.1 Canvas

Canvas 是一个在底层提供绘制图形以及操纵位图功能的 HTML5 元素。可以假设 Canvas 元素是一个动态的图片标签。传统 `<img>` 标签显示静态图像，无论这幅图片是动态生成的还是从服务器上静态加载的，图片都是静态不变的。可以改变 `<img>` 标签为另一幅图像源或者给图片应用样式，但我们不能修改图片位图的上下文本身。

从另一方面来看，Canvas 就像一个客户端运行的动态 `<img>` 标签。我们能在 Canvas 里加载图片、绘制图形并且可以通过 JavaScript 与之交互。

Canvas 在 HTML5 游戏开发中扮演着一个重要角色。它也是本书主要关注的焦点之一。

1.1.2 音频

背景音乐和声音效果经常是游戏设计的一个基本元素。HTML5 将通过 `audio` 标签来实现原生音频（audio）支持。有了该功能，就可以让 HTML5 游戏不再需要用 Flash 播放器来播放声音效果。第 6 章将讨论如何使用 `audio` 标签。

1.1.3 GeoLocation

GeoLocation（地理位置定位）接口让 Web 页面能获取到用户计算机的经纬度。这个功能在用户使用台式机连接到 Internet 的时候用处不是非常大。有很多时候需要为用户提供精确到道路级的定位。只能通过分析 IP 地址来获得一个大概的位置。

但是，近来越来越多的用户使用他们功能强大的智能手机来连接 Internet。Webkit 以及其他现代手机浏览器进入每一个人的口袋里。GeoLocation 让我们可以根据用户所在的方位来设计手机应用以及手机游戏。

基于定位的服务已经应用于一些社交网络应用程序了，如 foursquare (<http://foursquare.com>) 和 Gowalla (<http://gowalla.com>)。这类基于定位的社交社区的成功开创了使用智能手机定位服务的趋势。

1.1.4 WebGL

WebGL 扩展了 Canvas 元素，为 Web 游戏器提供了一套 3D 图形 API，该套 API 基于 OpenGL ES 2.0 标准。WebGL 为 3D HTML5 游戏提供真正的 3D 渲染场所。然而，在写本书时，并不是所有浏览器都原生地支持 WebGL。当前，只有 Mozilla Firefox 4、Google Chrome 以及最新的 WebKit 浏览器原生地支持该功能。

使用 WebGL 构建游戏的技术不同于一般的 HTML5 游戏开发。用 WebGL 构建游戏需要处理 3D 模型并使用与 OpenGL 类似的 API。因此，我们并不打算在本书中讨论 WebGL 的游

戏开发。

图 1-1 是 Google Body (<http://bodybrowser.googlelabs.com>) 的截图, 演示了如何使用 WebGL 并根据用户输入为用户显示 3D 人体模型。

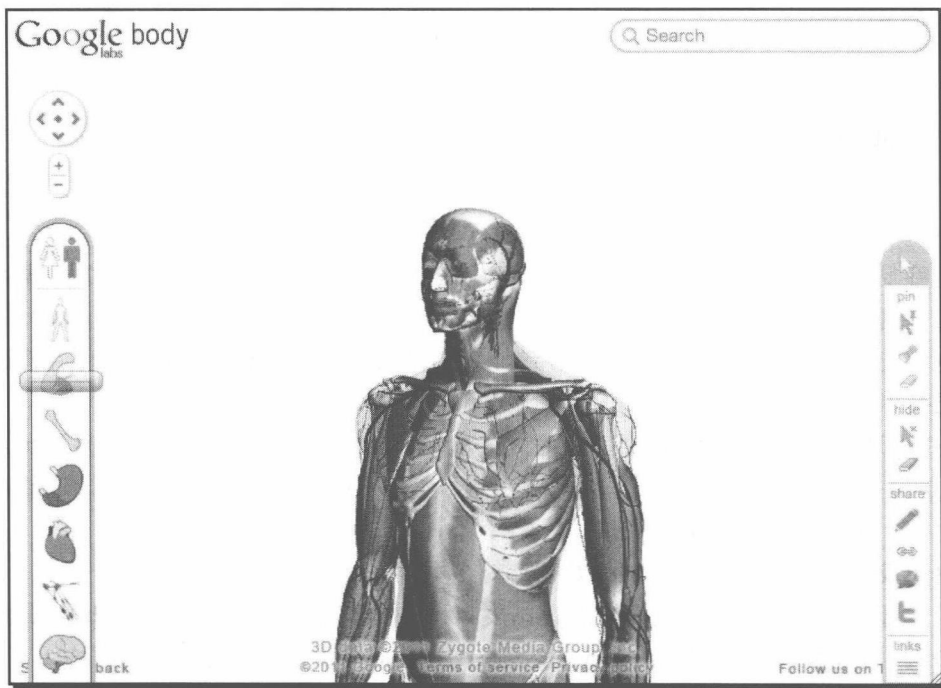


图 1-1



LearningWebGL (<http://learnwebgl.com>) 提供了一系列 WebGL 的入门教程。
如果你打算学习使用 WebGL, 那么该站点会是一个很好的出发点。

1.1.5 WebSocket

WebSocket 是 HTML5 标准的一部分, Web 页面可以用它来持久连接到 socket 服务器上。该接口提供了浏览器与服务器之间的事件驱动型连接, 这意味着客户端不必再每隔一个时间段就需要向服务器发送一次新的数据请求。当有数据需要更新时, 服务器就可以直接推送数据更新给浏览器。该功能的好处之一就是玩家之间可以实时进行交互。当一个玩家做了些事, 就会向服务器发送数据, 服务器将广播一个事件给其他已连接的所有浏览器, 让它们知道玩家做了什么。这样就使得制作 HTML5 网络游戏成为可能。