



首部全面讲解Three.js开发的实用指南，循序渐进讲解Three.js的各种功能
通过大量交互式示例，深入探索使用开源的Three.js库创建绚丽三维图形的实用方法和技巧



Learning Three.js

The JavaScript 3D Library for WebGL

[美] Jos Dirksen 著
李鹏程 译

Three.js

开发指南



机械工业出版社
China Machine Press

Three.js开发指南

[美] Jos Dirksen 著

李鹏程 译

Learning Three.js

The JavaScript 3D Library for WebGL



机械工业出版社

图书在版编目 (CIP) 数据

Three.js 开发指南 / (美) 德克森 (Dirksen, J.) 著; 李鹏程译. —北京: 机械工业出版社, 2015.1

(Web 开发技术丛书)

书名原文: Learning Three.js: The JavaScript 3D Library for WebGL

ISBN 978-7-111-48832-3

I. T… II. ① 德… ② 李… III. Java 语言—程序设计—指南 IV. TP312-62

中国版本图书馆 CIP 数据核字 (2014) 第 293435 号

本书版权登记号: 图字: 01-2014-2030

Learning Three.js: The JavaScript 3D Library for WebGL (ISBN: 978-1-78216-628-3).

Copyright © 2013 Packt Publishing. First published in the English language under the title “Learning Three.js: The JavaScript 3D Library for WebGL”.

All rights reserved.

Chinese simplified language edition published by China Machine Press.

Copyright © 2015 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。

Three.js 开发指南

出版发行: 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: 陈佳媛

责任校对: 董纪丽

印 刷: 三河市宏图印务有限公司

版 次: 2015 年 3 月第 1 版第 1 次印刷

开 本: 186mm × 240mm 1/16

印 张: 17

书 号: ISBN 978-7-111-48832-3

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

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzjsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

Preface 前言

在过去的两年里，浏览器变得更加强大了，而且能够作为平台来运行复杂的应用和图形。但大部分的图形都是二维的。现在大部分的浏览器已经实现了 WebGL，它们不仅可以创建二维图形和应用，还可以充分利用 GPU，创建漂亮的、高性能的三维应用。

直接使用 WebGL 编程非常复杂。你需要了解 WebGL 的内部细节，学习复杂的着色器语法，才能用好 WebGL。Three.js 库提供了一套基于 WebGL 的、非常易用的 JavaScript API，这样你不必学习 WebGL 的细节就可以创建漂亮的三维图形。

Three.js 库提供了大量功能和 API，你可以通过它们直接在浏览器中创建三维场景。本书提供了大量的交互式示例和代码，从中你可以学习到 Three.js 提供的所有 API。

本书内容

第 1 章讲述使用 Three.js 库的基本步骤。到这章结束的时候，你就可以创建出第一个 Three.js 场景。你可以直接在浏览器中创建三维场景，并制作动画。

第 2 章将会讲解 Three.js 库使用过程中你所需要理解的基础组件。你将会学习使用光源、网格、几何体、材质和相机。通过这章你会对 Three.js 库提供的各种光源及场景中使用的相机有一个大体的了解。

第 3 章进一步讲解场景中的各种光源。该章会展示几个例子，并解释如何使用各种光源，该章所介绍的光源包括：SpotLight、DirectionLight、AmbientLight、PointLight、HemisphereLight，以及 AreaLight。另外，该章还展示如何在光源中应用 LensFlare 效果。

第 4 章讨论 Three.js 库中可以用于网格的各种材质。这一章将会展示所有可以在材质中设置的属性，每种属性都有其特定的用途。并且这章还提供交互式的例子，通过它们你可以试验 Three.js 库中各种可用的材质。

第 5 章开始探讨 Three.js 库中的各种几何体。在这一章里你会学习如何创建、设置 Three.js 里的几何体，而且你还可以通过交互式的示例来试验，该章介绍的几何体包括：平

面、圆、自定义形状、方块、球体、圆柱体、圆环、环面纽结和多面体。

第 6 章继续第 5 章未讲完的内容，展示如何设置、使用 Three.js 库中的高级几何体，例如凸面体和扫描体。该章将会介绍如何从二维几何体中拉出三维几何体，以及如何使用二元运算将几何体组合起来。

第 7 章讲解如何使用 Three.js 库中的粒子系统。你将会学习如何从零开始创建粒子系统，以及如何基于已有的几何体创建粒子系统。你还会学习如何使用精灵和粒子材质来改变粒子的外观。

第 8 章展示如何从外部源导入网格和几何体。你将会学习如何使用 Three.js 库的 JSON 格式来存储几何体和场景。这一章还解释了如何从 OBJ、DAE、STL、CTM 和 PLY 格式的文件中加载模型。

第 9 章探讨各种动画效果，使场景活动起来。你将会学习如何结合 Tween.js 库使用 Three.js，以及基于变形和骨骼的动画效果。

第 10 章进一步展开讨论第 4 章引入的材质。这一章会深入讲解材质的细节。该章介绍了各种可用的材质，以及应用到网格上时如何控制材质。另外，在这一章你还会看到如何将 HTML5 视频的输出和 HTML5 画布中的元素作为纹理。

第 11 章探讨如何使用 Three.js 库来对已经渲染过的场景进行后期处理。通过后期处理，你可以对已经渲染过的场景应用某些特效，如模糊、移轴、陈旧。另外，你还会学习如何创建自定义的后期处理效果，以及创建自定义顶点和片段着色器。

第 12 章讲解如何在 Three.js 的场景中添加物理效果。有了物理效果后，你就可以检测物体之间的碰撞，使它们能够对重力产生反应，并产生摩擦。这一章将会展示如何使用 JavaScript 的物理库来达到上述效果。

阅读之前的准备

阅读本书你要准备的只是一个文本编辑器（例如 Sublime 文本编辑器），用来编辑示例，以及一个现代的浏览器，用来显示这些示例。有些示例需要一个本地的 Web 服务器，但是在第 1 章，你将会学习如何搭建一个非常轻量的 Web 服务器来使用本书中的示例。



在我们开始之前，先对书中示例可能出现的问题简单提一下。第 1 章大致介绍了一下支持 WebGL 的浏览器，这是运行 Three.js 所必需的。现在的浏览器，诸如 Chrome、Firefox 和 Internet Explorer 可以很好地支持 HTML5 标准。但是仍然有一些细节要注意。当浏览器有新的版本时，它们有可能会不支持 WebGL 的某些特性。例如，在本书完成的时候，Windows 7 上的 Chrome 和 Firefox 在运行第 11 章的例子时会有问题。所以在尝试这些示例时，请确保升级到最新版的 Chrome 和 Firefox。

读者对象

本书对所有已经了解 JavaScript 并开始想在浏览器里创建三维图形的人都会有很大的帮助。你不必去了解任何高级的算法或 WebGL，只要具有一般的 JavaScript 和 HTML 知识即可。所有示例代码可以登录华章网站 (www.hzbook.com) 免费下载，本书中用到的工具也都是开源的。所以，如果你想要学习如何创建漂亮的、交互的三维图形，并且可以在任何现代的浏览器上运行，那么这本书就是你想要的。

致 谢 *Acknowledgements*

写书是一件耗时且艰苦的事情。如果没有很多人的支持和帮助，我不可能完成。我想在这里感谢他们。

首先是我的项目协调人，Leena Purkait，要不是她帮助我协调所有的细节，从而让我可以专注于内容的创作，我根本不可能按时完成所有的章节。我还要感谢 Ritika Singh 和 Anita Nayak，她们花了很多时间和精力来完善本书。

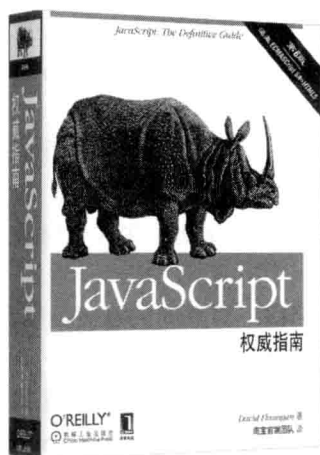
还有来自 Packt 出版社的好多人，他们在我写书、审校期间给了我很多帮助，保证整个过程顺利进行。干得不错，伙计们！

当然，我还要感谢 Ricardo Cabello，即 Mr.dò_ób，Three.js 正是他的杰作。

非常感谢本书的各位审校人。感谢你们那些非常好的反馈和建议，对本书的改进帮助很大。正是你们的积极评论才成就了本书。

我还没有提到最重要的人。我想感谢我的太太 Brigitte，她一再地容忍我在周末和晚上将大量的时间花费在笔记本电脑上。我要感谢我的女儿 Sophie，她经常把我从键盘上拉开，逗我开心。还有我的女儿 Amber，尽管只有几周大，但是她让我对生命中真正重要的事情和时刻充满感激。

推荐阅读



JavaScript权威指南（原书第6版）

从1996年以来，本书已经成为 JavaScript 程序员的《圣经》。

第6版涵盖HTML5和ECMAScript 5。很多章节完全重写，以便跟得上当今的最佳Web开发实践。该版本的新增章节描述了jQuery和服务端JavaScript。

对于那些希望学习Web编程语言的有经验的程序员和希望精通JavaScript的当前JavaScript程序员，本书最适合不过了。

深入理解PHP：高级技巧、面向对象与核心技术（原书第3版）

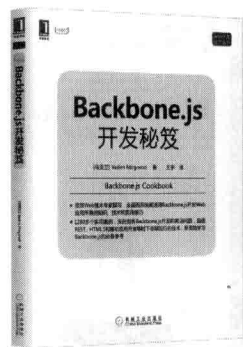
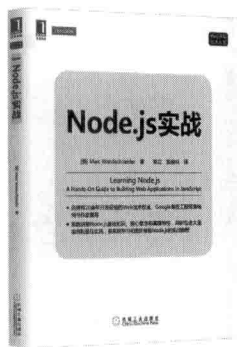
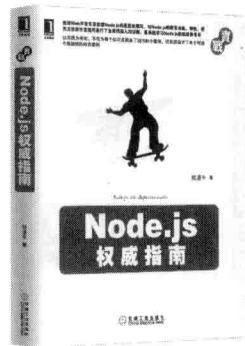
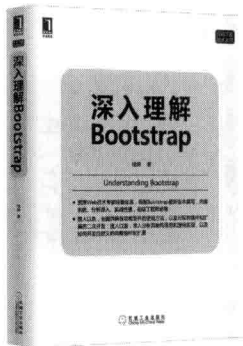
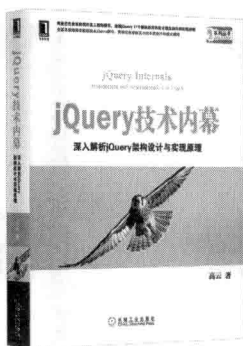
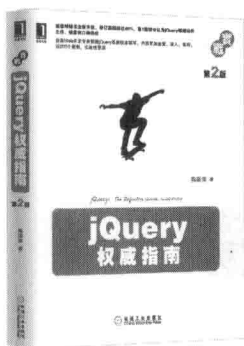
国际知名Web开发专家和技术畅销书作家最新力作，PHP领域经典著作。

从编程技巧、面向对象和扩展三个角度系统讲解和总结了成为中高级PHP程序员应该具备的技术和技能，包含大量实用案例，极具实践指导意义。

如果你已经具备一定的PHP编程基础，想使开发效率更高，想把应用做得更好，那么这本书应该是你需要阅读的。本书旨在为想修炼成为高级PHP程序员的中初级PHP程序员提供实用的方法和建议。

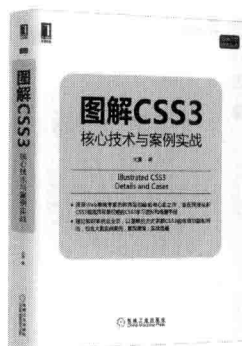
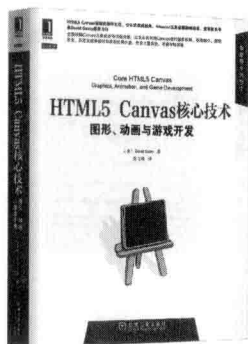
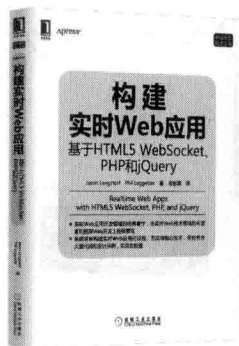
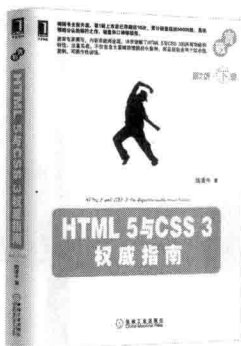
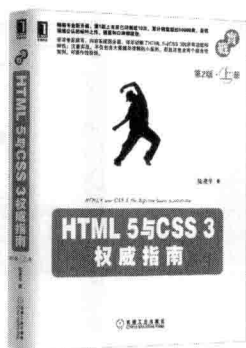
Web前端开发&设计经典

框架篇



Web前端开发&设计经典

HTML5&CSS3篇



前 言
致 谢

第 1 章 用 Three.js 创建你的第一个 三维场景.....1

- 1.1 使用 Three.js 的前提条件3
- 1.2 获取源代码4
 - 1.2.1 用 Git 克隆代码仓库4
 - 1.2.2 下载并解压压缩包5
 - 1.2.3 测试示例6
- 1.3 创建 HTML 页面框架7
- 1.4 渲染并展示三维对象9
- 1.5 添加材质、灯光和阴影13
- 1.6 用动画扩展你的首个场景15
 - 1.6.1 引入 requestAnimationFrame() 方法15
 - 1.6.2 转动方块17
 - 1.6.3 弹跳球17
- 1.7 使用 dat.GUI 库简化试验18
- 1.8 使用 ASCII 效果19
- 1.9 总结20

第 2 章 使用构建 Three.js 场景的 基本组件.....22

- 2.1 创建场景22

- 2.1.1 场景的基本功能23
- 2.1.2 在场景中添加雾化效果27
- 2.1.3 使用材质覆盖属性28
- 2.2 使用几何和网格对象29
 - 2.2.1 几何对象的属性和函数29
 - 2.2.2 网格对象的函数和属性33
- 2.3 选择合适的相机37
 - 2.3.1 正投影相机和透视相机37
 - 2.3.2 让相机在指定点上聚焦41
- 2.4 总结41

第 3 章 使用 Three.js 里的各种光源43

- 3.1 探索 Three.js 库提供的光源43
- 3.2 学习基础光源44
 - 3.2.1 AmbientLight——影响整个场景
的光源44
 - 3.2.2 PointLight——照射所有方向的
光源47
 - 3.2.3 SpotLight——具有锥形效果的
光源49
 - 3.2.4 DirectionalLight——模拟远处
类似太阳的光源53
 - 3.2.5 使用特殊光源生成高级
光照效果54
- 3.3 总结61

第 4 章 使用 Three.js 的材质62	6.2 LatheGeometry.....104
4.1 理解共有属性.....63	6.3 通过拉伸创建几何体.....105
4.1.1 基础属性.....63	6.3.1 ExtrudeGeometry.....106
4.1.2 融合属性.....64	6.3.2 TubeGeometry.....107
4.1.3 高级属性.....64	6.3.3 从 SVG 拉伸.....109
4.2 从简单的网格材质（基础、深度和面）开始.....65	6.3.4 ParametricGeometry.....110
4.2.1 简单表面的 MeshBasicMaterial.....65	6.4 创建三维文本.....112
4.2.2 基于深度着色的 MeshDepthMaterial.....67	6.4.1 渲染文本.....113
4.2.3 联合材质.....68	6.4.2 添加自定义字体.....114
4.2.4 计算法向颜色的 MeshNormalMaterial.....70	6.5 使用二元操作组合网格.....115
4.2.5 为每个面指定材质的 MeshFaceMaterial.....72	6.5.1 subtract 函数.....117
4.3 学习高级材质.....73	6.5.2 intersect 函数.....120
4.3.1 用于暗淡、不光亮表面的 MeshLambertMaterial.....74	6.5.3 union 函数.....121
4.3.2 用于光亮表面的 MeshPhongMaterial.....75	6.6 总结.....121
4.3.3 用 ShaderMaterial 创建自己的着色器.....76	第 7 章 粒子和粒子系统123
4.4 线段几何体的材质.....81	7.1 理解粒子.....123
4.4.1 LineBasicMaterial.....81	7.2 粒子、粒子系统和 BasicParticleMaterial.....125
4.4.2 LineDashedMaterial.....83	7.3 使用 HTML5 画布格式化粒子.....128
4.5 总结.....84	7.3.1 在 CanvasRenderer 类里使用 HTML5 画布.....128
第 5 章 学习使用几何体85	7.3.2 在 WebGLRenderer 中使用 HTML5 画布.....129
5.1 Three.js 提供的基础几何体.....86	7.4 使用纹理格式化粒子.....131
5.1.1 二维几何体.....86	7.5 从高级几何体中创建粒子系统.....139
5.1.2 三维几何体.....92	7.6 总结.....141
5.2 总结.....100	第 8 章 创建、加载高级网格和几何体143
第 6 章 使用高级几何体和二元操作102	8.1 几何体组合和合并.....143
6.1 ConvexGeometry.....102	8.1.1 对象组合.....143
	8.1.2 将多个网格合并成一个网格.....145
	8.2 从外部资源中加载几何体.....147
	8.3 以 Three.js 的 JSON 格式保存和加载.....147

8.3.1	保存和加载几何体	147
8.3.2	保存和加载场景	149
8.4	使用 Blender	151
8.4.1	在 Blender 中安装 Three.js 导出器	152
8.4.2	在 Blender 里加载和导出模型	153
8.5	导入三维格式文件	155
8.5.1	OBJ 和 MTL 格式	156
8.5.2	加载 Collada 模型	159
8.5.3	加载 STL、CTM 和 VTK 模型	160
8.5.4	展示蛋白质数据银行中的蛋白质	161
8.5.5	从 PLY 模型中创建粒子系统	163
8.6	总结	164
第 9 章 创建动画和移动相机 166		
9.1	基础动画	166
9.1.1	简单动画	167
9.1.2	选择对象	168
9.1.3	用 Tween.js 做动画	170
9.2	使用相机	172
9.2.1	轨迹球控件	172
9.2.2	飞行控件	174
9.2.3	翻滚控件	175
9.2.4	第一人称控件	175
9.2.5	轨道控件	177
9.2.6	路径控件	178
9.3	变形动画和骨骼动画	180
9.3.1	用变形目标制作动画	181
9.3.2	用骨骼和蒙皮制作动画	184
9.4	使用外部模型创建动画	186
9.4.1	用 Blender 创建骨骼动画	186
9.4.2	从 Collada 模型中加载动画	190
9.4.3	从雷神之锤模型中加载动画	191
9.5	总结	192
第 10 章 加载和使用纹理 194		
10.1	在材质中使用纹理	194
10.1.1	加载纹理并应用到网格	194
10.1.2	使用凹凸贴图创建皱纹	197
10.1.3	使用法向贴图创建更加细致的凹凸和皱纹	198
10.1.4	使用光照贴图创建假阴影	199
10.1.5	用环境贴图创建虚假的反光效果	201
10.1.6	高光贴图	204
10.2	纹理的高级用途	205
10.2.1	定制 UV 映射	205
10.2.2	重复映射	207
10.2.3	在画布上绘制图案并作为纹理	209
10.3	总结	213
第 11 章 定制着色器和渲染后期处理 215		
11.1	设置后期处理	215
11.1.1	创建 EffectComposer 对象	216
11.2	后期处理通道	218
11.2.1	简单后期处理通道	218
11.2.2	使用掩膜的高级效果组合器	223
11.2.3	用 ShaderPass 定制效果	226
11.3	创建自定义的后期处理着色器	231
11.3.1	定制灰度图着色器	232
11.3.2	定制位着色器	235
11.4	总结	237

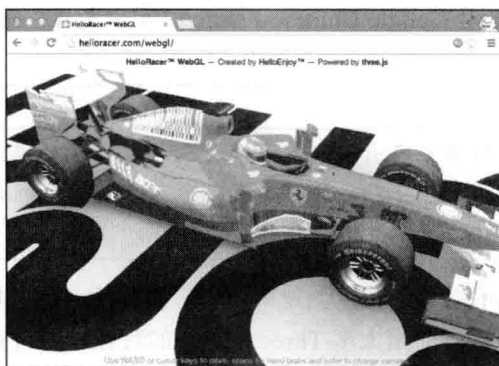
第 12 章 用 Physijs 在场景中添加物理效果	238
12.1 创建可用 Physijs 的基本 Three.js 场景.....	238
12.2 材质属性.....	243
12.3 基础图形.....	244
12.4 使用约束限制对象移动.....	248
12.4.1 用 PointConstraint 限制两点间的移动.....	249
12.4.2 用 HingeConstraint 创建类似门的约束.....	250
12.4.3 用 SliderConstraint 将移动限制到一个轴.....	252
12.4.4 用 ConeTwistConstraint 创建类似球销的约束.....	254
12.4.5 用 DOFConstraint 实现细节的控制.....	255
12.5 总结.....	259

用 Three.js 创建你的第一个三维场景

如今浏览器的功能越来越强大，而且这些功能可以通过 JavaScript 直接调用。你可以用 HTML5 标签轻松地添加视频和音频，而且可以在 HTML5 画布上创建各种交互组件。现在这个功能集合里又有了一个新成员，即支持 WebGL。通过 WebGL，你可以直接使用显卡的计算资源，创建高性能的二维和三维计算机图形，然后在 JavaScript 里直接使用 WebGL 编程，创建三维场景并生成动画，这个过程非常复杂，而且容易出错。Three.js 库可以简化这个过程。Three.js 可以帮助我们的地方如下：

- 创建简单的和复杂的三维图形
- 在三维场景中生成动画、移动物体
- 在物体上应用纹理和材质
- 从三维建模软件中加载图形
- 创建基于样条曲线的二维图形

只要几行代码，你就可以从简单的三维图形创建出逼真的、实时的场景来，如下图所示：



在第 1 章中我们会直接开始探究 Three.js，通过创建几个例子来展示 Three.js 是如何工作的。你可以自己亲手试验一下这几个例子。但我们并不会深入探究所有的技术细节，这些细节我们将会在今后的章节中讲解。本章会探讨下面几方面的内容：

- ❑ 使用 Three.js 时要用到的工具
- ❑ 下载本书附带的源码和示例
- ❑ 创建你的第一个 Three.js 场景
- ❑ 应用材质、光线和动画来改进你创建的场景
- ❑ 引入几个辅助库，用于统计和控制场景

我们首先会简单介绍一下 Three.js，然后立即开始讲解第一个示例及其代码。在开始之前我们先来快速浏览一下现在市面上几个最负盛名的浏览器，以及它们对 WebGL 的支持情况。

现在支持 Three.js 的浏览器如下表：

浏览器	支持情况
Mozilla Firefox	4.0 版以后开始支持
Google Chrome	第 9 版以后开始支持
Safari	5.1 版开始支持，以及那些新安装在 Mac OS X 的山狮版、狮子版和雪豹版操作系统上的版本。确保在 Safari 中打开了 WebGL 选项。为此你可以找到 Preferences Advanced 菜单，选中 Show develop menu in menu bar。然后选中菜单项 Develop Enable WebGL
Opera	12.00 版以后开始支持。要让 Opera 支持 WebGL，你还需要打开 opera:config 文件，设置 WebGL，并将 Enable Hardware Acceleration 置为 1。然后重启浏览器
Internet Explorer	Internet Explorer 是长期以来唯一一个不支持 WebGL 的主流浏览器。从 IE11 开始，微软才开始支持 WebGL

基本上 Three.js 可以在任何流行的浏览器上运行，除了 IE 的大部分版本。所以如果你想用旧版本的 IE，你有两个选择：你可以用 Google Chrome Frame 来支持 WebGL，可以从如下网址下载：<https://developers.google.com/chrome/chrome-frame/>。除了 Google Chrome Frame，你也可以使用 iewebgl 插件，下载网址是 <http://iewebgl.com/>。在 IE 上安装这个插件，使其支持 WebGL。



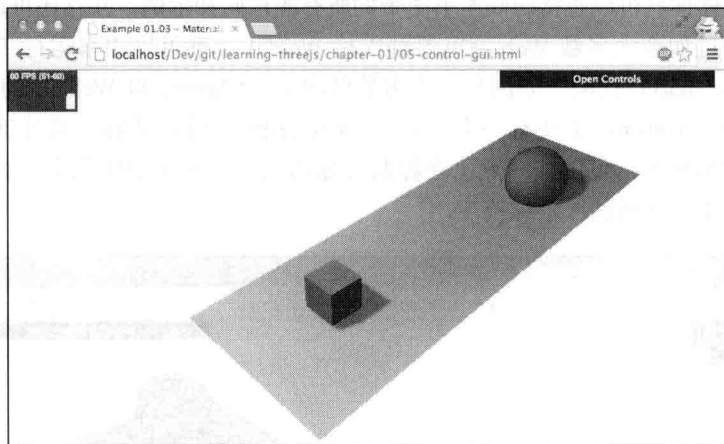
Google 已经停止了对 Google Chrome Frame 的官方支持，而且 Google Chrome Frame 也不支持 IE10。



现在 Three.js 的工程师正在开发一个遵从 CSS-3D 规范的新渲染器，而这个规范大多数浏览器都支持（包括 IE10）。除了桌面浏览器之外，手机和平板电脑上的浏览器也都支持 CSS-3D。

通过本章的学习，你可以直接开始创建第一个三维场景，这个场景应该可以在上述任意一个浏览器里运行。我们不会介绍太多 Three.js 复杂的特性，不过在本章结束的时候你应该

可以建立一个 Three.js 场景，如下图所示：



通过创建这个场景，你可以学习到 Three.js 的一些基础知识，还可以创建第一个动画。在开始学习这个示例之前，我们先介绍几个工具，这些工具可以帮助你更方便地使用 Three.js。我们还要学习如何下载本书中展示的那些示例。

1.1 使用 Three.js 的前提条件

Three.js 是一个 JavaScript 库，所以当你用 Three.js 创建一个 WebGL 应用时，你所需的仅仅是一个文本编辑器和一个支持 WebGL 的浏览器（用来渲染结果）。我非常诚恳地向你推荐三个 JavaScript 编辑器，在过去的几年里我一直在使用它们。

- ❑ **WebStorm**：这是一个由 JetBrains 公司开发的编辑器，对于编写 JavaScript 程序支持得非常好。它提供代码补全、自动部署等功能，还可以直接在编辑器里调试 JavaScript。除此之外，WebStorm 还能很好地支持 Git（以及其他的版本控制系统）。你可以从网址 <http://www.jetbrains.com/webstorm/> 获取一个试用版。
- ❑ **Notepad++**：这是一个通用型的编辑器，支持很多的编程语言。用它可以很方便地排版和格式化 JavaScript 代码。



Notepad++ 只有 Windows 版。

- ❑ **Sublime Text Editor**：这是一个非常好用的小型编辑器，对于编写 JavaScript 代码支持良好。除此之外，它还提供了很多很有用的选择和编辑功能，一旦你习惯了这些功能，那它就是一个真正不错的 JavaScript 编辑环境。Sublime Text Editor 也有免费试用版，你可以从网址 <http://www.sublimetext.com/> 下载。

即使你不用这三个编辑器，依然有很多编辑器可供选择，开源的、商业的都有。它们都可以用来编辑 JavaScript 代码，创建 Three.js 项目。网上有一个开源项目有点儿意思，你或