

Join the discussion @ p2p.wrox.com



Wrox Programmer to Programmer™

移动开发经典丛书



Game and Graphics Programming for iOS and Android with
OpenGL ES 2.0

OpenGL ES 2.0

游戏与图形编程

——适用于iOS和Android

[美] Romain Marucchi-Foino

王净

著
译



清华大学出版社

014032456

TN929. 53
577

移动开发经典丛书

OpenGL ES 2.0 游戏与 图形编程

——适用于 iOS 和 Android

[美] Romain Marucchi-Foino 著

王 净

译



清华大学出版社

北 京



北航

C1720821

TN 929.53
577

Romain Marucchi-Foino

Game and Graphics Programming for iOS and Android with OpenGL ES 2.0

EISBN: 978-1-119-97591-5

Copyright © 2012 by Romain Marucchi-Foino

All Rights Reserved. This translation published under license.

本书中文简体字版由 Wiley Publishing, Inc. 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: 01-2013-7449

Copies of this book sold without a Wiley sticker on the cover are unauthorized and illegal.

本书封面贴有 Wiley 公司防伪标签, 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

OpenGL ES 2.0 游戏与图形编程——适用于 iOS 和 Android / (美) 马鲁基-弗伊诺(Marucchi-Foino, R.) 著; 王净 译.
—北京: 清华大学出版社, 2014

(移动开发经典丛书)

书名原文: Game and Graphics Programming for iOS and Android with OpenGL ES 2.0

ISBN 978-7-302-35230-3

I. ①O… II. ①马… ②王… III. ①移动终端—游戏程序—程序设计 IV. ①TN929.53 ②TP311.5

中国版本图书馆 CIP 数据核字(2014)第 014856 号

责任编辑: 王 军 韩宏志

装帧设计: 牛静敏

责任校对: 曹 阳

责任印制: 沈 露

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 装 者: 三河市金元印装有限公司

经 销: 全国新华书店

开 本: 185mm×260mm

印 张: 18

字 数: 438 千字

版 次: 2014 年 2 月第 1 版

印 次: 2014 年 2 月第 1 次印刷

印 数: 1~3000

定 价: 48.00 元

产品编号: 054931-01

译者序

随着智能手机移动嵌入式平台硬件性能的不断提升,3D 游戏应用逐渐普及开来。目前,Android 和 iOS 平台占据了大部分市场份额。如何有效地在这两种平台上开发高质量的游戏已成为很多游戏开发人员追求的目标。OpenGL ES (OpenGL for Embedded Systems) 是 OpenGL 三维图形 API 的子集,专门针对手机、PDA 和游戏主机等嵌入式设备而设计,目前已有越来越多的开发人员选用 OpenGL ES 进行游戏和图形编程。但目前国内关于 OpenGL ES 开发的资料非常有限。相信本书的出版可为 OpenGL 的开发起到推进作用。

本书是一本系统的具备实战性的 OpenGL ES 图形开发指南。由资深的游戏引擎开发专家针对 OpenGL ES 最新版本撰写,不仅系统讲解了 OpenGL ES 的核心概念和技术,讨论 iOS 和 Android 的图形机制,还通过大量案例讲解在 iOS 和 Android 上进行 OpenGL ES 开发的方法和技巧。

全书共分 12 章,涵盖了创建完整游戏所需的所有知识。你将学习创建游戏所涉及的方方面面,比如加载 3D 几何体和纹理,如何处理材质、着色器、声音、相机、剪切、物理、AI、路径发现、骨骼动画等。本书所有章节都彼此相关。每章都以一个实例为基础逐步向读者演示如何掌握所需的技术。如果读者掌握了本书中的所有内容,就可以很快地开始编写实际的游戏和图形程序。

本书的讲解由浅入深,从 Android 和 iOS 平台上游戏和图形开发应用的基础知识到开发大型游戏程序,结构清晰、语言简洁,非常适合进阶开发者阅读参考。

本书第 1 章和第 3 章由范园芳、胡训强负责翻译,第 2 章和第 6 章由彭洁、刘晓昌负责翻译,第 4 章和第 9 章由姜茗瀚负责翻译,第 10 章由晏峰负责翻译,第 5 章、第 7 和第 8 章、第 11 和第 12 章由王净负责翻译。最后,全书由王净进行统稿。在此,我要对各位译者深表谢意。此外,还要感谢我的家人,她们总是无怨无悔地支持我的一切工作,生活在这样美满的家庭,我备感幸福。

译者在翻译过程中,尽量保持原书的特色,并对书中出现的术语和难词难句进行了仔细推敲和研究。但毕竟有少量技术是译者在自己的研究领域中所不曾遇到过的,所以疏漏和争议之处在所难免,望广大读者提出宝贵意见。

最后,希望广大读者能多花些时间细细品味这本凝聚作者和译者大量心血的书籍,为将来的职业生涯奠定良好基础。

王 净

作者简介

Romain Marucchi-Foino 是当今流行的移动游戏引擎 SIO2(<http://sio2interactive.com>)的原作者和创建者。Romain 以前是一名游戏引擎开发人员,自从 iPhone 诞生以来, Romain 就一直致力于使用 OpenGL ES 为移动设备创建先进的游戏引擎。目前,他是 sio2interactive.com 的一名高级 3D 程序员(SIO2 引擎的正式开发人员),该引擎通过 App Store 和 Android Market 驱动了数以千计的游戏和 3D 应用程序。凭借在移动游戏行业积累的丰富经验,他为许多网上社区、出版物以及博客作出重大贡献。

技术编辑简介

Effie C. Lee 在过去 4 年里一直从事游戏行业,并且是一名自营职业的游戏和图形设计师。在获得了计算机科学专业的学士学位后,他开始对电子游戏和计算机图形学产生了浓厚的兴趣,并参与了多个移动游戏产品的开发。凭借在游戏开发方面的广博知识,他逐渐成为一名专业的 2D 和 3D 图形设计师、游戏程序员(编写脚本)以及游戏网站的 Web 设计师,并且管理游戏质量和本地化。如果你想与他取得联系,可以发邮件到 effiecl@gmail.com。

前言

欢迎使用本书。本书并不是一本常见的“OpenGL Hello Triangle”书籍——也就是说并不会详细解释“为什么”(你可以从 Google 上找到相关的原因)，而重点介绍“如何做”。将会详细地介绍在编写游戏和图形程序时应该做什么，不应该做什么。

本书包括了 50 多个不同的教程(其中还包括了一些完整的游戏框架)，并且采用了一种简单实用的新颖方法重点介绍需要学习的内容，从而确保完成教程后可以创建一款游戏。

我们将学习使用丰富的 3D 图像创建完整游戏所需的所有元素。如果正在寻找一种好的教学方法来帮助你快速创建梦想中的游戏，那么本书正是你所需要的。

本书读者对象

首先，本书并不是一本入门级图书，而是一本中等层次的书籍，本书假设你已经熟悉线性代数(矩阵、向量和四元数)，具有深厚的 C/C++ 编程背景，至少接触过基础的 OpenGL 或者 OpenGL ES，以及知道计算机图形的基本工作方式。

如果已经掌握了这些必需的知识，并且想在游戏和图形编程方面取得闪电般的进步，那么本书正是你所需要的。本书主要是为那些想要学习核心知识以便创建功能完备的游戏并在 App Store 和 Android Market 上进行售卖的人所撰写的。

本书主要内容

从根本上讲，本书包括了创建一个完整的游戏所需的所有知识。你将学习创建游戏所涉及的各个方面，如加载 3D 几何体和纹理，如何处理材质、着色器、声音、摄像头、剪切、物理、AI、路径发现和骨骼动画等。

在学完本书后，你将能使用所学的知识并组合所完成的不同教程以创建自己的高级游戏。

本书组织结构

本书的组织结构是所有的章节都彼此关联。每一章都会逐步向读者演示如何掌握所需的技术，以便能够处理和理解下一章的内容。

接下来是所有章的列表，以及每一章所介绍的内容：

- 第 1 章“入门”——我们将学习如何构建开发环境，下载本书的 SDK，导入并重新编译教程以及如何处理全书将要使用的模板项目。
- 第 2 章“设置图形投影”——由于已经拥有了一个正在运行的模板，因此我们将学习如何构建所需的投影矩阵，以便能够处理 2D、2.5D 或 3D。此外还将学习如何在屏幕上绘制简单几何图形以及如何处理摄像头矩阵。
- 第 3 章“处理复杂几何图形”——我们将通过创建一个 Wavefront OBJ 查看器来学习如何从磁盘加载复杂几何图形。还将学习如何加载和创建纹理，如何处理基本光照以及响应触摸事件。
- 第 4 章“构建场景”——该章将对第 3 章学到的知识进行扩展，将解释如何处理更复杂的场景。我们将学习绘图序列以及如何创建可重用的着色器。
- 第 5 章“优化”——该章将介绍用来优化绘图性能的相关技术。我们将学习纹理压缩和着色器优化的基础知识，学习如何将三角形转换为三角形带以及其他用来获取更佳 FPS 的提示和技巧。
- 第 6 章“实时物理”——由于在该章之前我们已经学习了如何正确地处理场景，因此该章将介绍如何使用 Bullet 向场景添加实时物理行为。我们将首先学习如何创建物理世界和物理实体。然后学习如何在代码中使用不同的技术，以便在碰撞回调中或者根据两个或更多个物理实体之间的接触点添加相关逻辑。
- 第 7 章“摄像头”——该章将重点介绍摄像头。我们将首先学习构建视锥平面，并能从摄像头的角度确定场景中每个对象的可见性。然后学习如何实现多种不同类型的摄像头，包括带有碰撞的完整第一人称和第三人称摄像头(你可以在自己的应用程序中使用这些摄像头)。
- 第 8 章“路径发现”——人工智能(AI)和路径发现功能在游戏中通常扮演重要角色，而这恰好是本章将要介绍的内容。我们将学习如何使用 Recast 和 Detour 库构建导航网格以及让实体在场景中自动移动。该章还将演示如何使用 True Type Font 生成字体纹理以及如何在屏幕上绘制动态文本。
- 第 9 章“音频及其他极佳的游戏编程资料”——该章主要介绍如何使用 OpenAL 来播放音频。我们将学习如何加载 OGG Vorbis 声音文件以及如何实时地从内存中以流的方式读出这些文件或者静态地将它们存储到音频存储器中。此外，还将学习如何创建 3D 定位和环境声源，如何使用加速计以及如何实现纹理动画和创建其他多种效果。
- 第 10 章“高级光照”——该章将介绍如何应用动态光照，这可能是游戏和图形编程中最难掌握的内容。在该章，我们将创建多种不同类型的灯(从定向灯到聚光灯)，并学习如何实时处理这些灯。
- 第 11 章“高级 FX”——该章主要介绍一些特殊效果。我们将学习如何创建全屏的后处理效果、投影纹理和实时阴影，以及如何处理粒子。

- 第 12 章“骨骼动画”——最后的内容并不一定是最不重要的内容。在该章，我们将学习如何使用 MD5 文件格式处理骨骼动画。该章将首先介绍如何加载和绘制附加到骨骼的网格。然后介绍如何加载动作文件以及如何使用不同类型的混合方法来混合这些动作。

你将会发现，本书并不只是介绍相关理论，还会演示如何将每章所学到的知识应用到实际游戏方案中。

如你所见，本书介绍了在日常游戏或 3D 应用程序中需要使用的有用知识。如果掌握了本书中的所有内容，就可以很快地开始编写实际的游戏和图形程序。

使用本书所需条件

如果打算为 iOS 系统开发应用程序，那么需要一台可以支持 iOS SDK 最新版本的 Mac(如果想了解更多的信息，请访问 <http://developer.apple.com>)。由于 iOS SDK 提供了 iPhone/iPod Touch 和 iPad 模拟器(可用开发和测试应用程序)，因此 iDevice 是可选的。该模拟器与本书所包含的内容是完全兼容的。

如果打算为 Android 开发应用程序，那么需要一台带有 Android SDK 所支持的操作系统 Mac 或 PC(更多信息请访问 <http://developer.android.com>)。同时，还需要一台支持 OpenGL ES 2.0 的 Android 设备，因为 Android SDK 绑定的模拟器仅支持 OpenGL ES 1.0。

此外，本书使用 Blender 作为 3D 建模软件(因为该软件是免费和开源的)。所以，为了测试、调试和重新输出本书 SDK 中使用的所有测试场景，请从 <http://blender.org> 获取该软件的一份副本。

SDK 源代码

可从 www.wrox.com 下载本书中所使用的官方 SDK(被压缩为一个.zip 文件)。该 SDK 包含了本书中所介绍的所有教程的最终结果。同时，还包含了 SDK 的全部源代码以及教程中使用的所有原始资产，因此你可以访问书中的 2D/3D 场景，并且可以重新编译它们。也可以访问 <http://www.tupwk.com.cn/downpage>，输入本书中文书名或 ISBN，下载本书源代码。



注意：因为许多书都拥有相类似的标题，所以使用 ISBN 可以更容易地找到本书。本书英文版的 ISBN 为 978-1-119-97591-5。

此外，我个人还在 GFX 3D Engine(一款免费且开源的迷你 3D 游戏和图形引擎，我们

将在本书中使用该引擎,如图 0-1 所示)的官方网站上提供了本书的 SDK(使用了 GIT 版本的控制系统),可从 <http://gfx.sio2interactive.com> 下载该引擎。

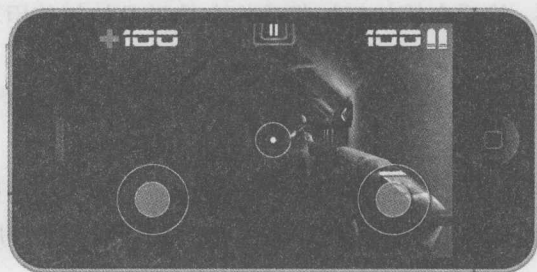


图 0-1 GFX 3D Engine

读者可以随时从 GFX 3D Engine 网站找到本书 SDK 的最新版本(针对上一个版本的 bug 进行了修复),因为对于我来说,在该网站上可以更加容易地使用版本控制来更新源代码。而对于 www.wrox.com 上的官方 SDK 来说,更新速度则相对较慢,因为官方 SDK 版本是由发布者提供的,如果愿意,可以耐心地等待官方 SDK 的发布。

此外值得一提的是,在 GFX 3D Engine 网站上(<http://gfx.sio2interactive.com>),你可以找到针对本书 SDK 的支持论坛以及 GFX 3D Engine 的最新版本。该网站还提供了其他的与 3D 游戏和图形相关联的演示、教程和其他材料,并且都与本书的 SDK 完全兼容。

勘误表

尽管我们已经尽了各种努力来保证文章或代码中不出现错误,但是错误总是难免的,如果你在本书中找到了错误,例如拼写错误或代码错误,请告诉我们,我们将非常感激。通过勘误表,可以让其他读者避免受挫,当然,这还有助于提供更高质量的信息。

要在网站上找到本书的勘误表,可以登录 <http://www.wrox.com>,通过 Search 工具或书名列表查找本书,然后在本书的细目页面上,单击 Book Errata 链接。在这个页面上可以查看 Wrox 编辑已提交和粘贴的所有勘误项。完整的图书列表还包括每本书的勘误表,网址是 www.wrox.com/misc-pages/booklist.shtml。

如果在 Book Errata 页面上没有看到你找出的错误,请进入 www.worx.com/contact/techsupport.shtml,填写表单,发电子邮件,我们就会检查你的信息,如果是正确的,就在本书的勘误表中粘贴一个消息,我们将在本书的后续版本中采用。

另外,如果有任何意见和建议,也可以给 wkservice@vip.163.com 发电子邮件。

p2p.wrox.com

P2P 邮件列表是为作者和读者之间的讨论而建立的。读者可以在 p2p.wrox.com 上加入 P2P 论坛。该论坛是一个基于 Web 的系统,用于传送与 Wrox 图书相关的信息和相关技术,

与其他读者和技术用户交流。该论坛提供了订阅功能，当论坛上有新帖子时，会给你发送你选择的主题。Wrox 作者、编辑和其他业界专家和读者都会在这个论坛上进行讨论。

在 <http://p2p.wrox.com> 上有许多不同的论坛，帮助读者阅读本书，在读者开发自己的应用程序时，也可以从这个论坛中获益。要加入这个论坛，必须执行下面的步骤：

- (1) 进入 p2p.wrox.com，单击 Register 链接。
- (2) 阅读其内容，单击 Agree 按钮。
- (3) 提供加入论坛所需的信息及愿意提供的可选信息，单击 Submit 按钮。
- (4) 然后就可以收到一封电子邮件，其中的信息描述了如何验证账户，完成加入过程。



提示：不加入 P2P 也可以阅读论坛上的信息，但只有加入论坛后，才能发送自己的信息。

加入论坛后，就可以发送新信息，回应其他用户的帖子。可以随时在 Web 上阅读信息。如果希望某个论坛给自己发送新信息，可以在论坛列表中单击该论坛对应的 **Subscribe to this Forum** 图标。

对于如何使用 Wrox P2P 的更多信息，可阅读 P2P FAQ，了解论坛软件的工作原理，以及许多针对 P2P 和 Wrox 图书的常见问题解答。要阅读 FAQ，可以单击任意 P2P 页面上的 FAQ 链接。

目 录

第 1 章 入门	1
1.1 软件需求	1
1.1.1 针对 iOS 开发人员	1
1.1.2 针对 Android 开发人员	2
1.2 下载本书的 SDK	3
1.3 导入项目	4
1.3.1 针对 iOS 开发人员	5
1.3.2 针对 Android 开发人员	5
1.4 模板	6
1.5 小结	7
第 2 章 设置图形投影	9
2.1 三种基本的投影类型	9
2.2 正射 2D 投影	11
2.2.1 程序和项目初始化	12
2.2.2 顶点和片段着色器	14
2.2.3 链接着色器程序	16
2.2.4 绘图代码	18
2.3 正交投影	23
2.4 透视投影	26
2.5 小结	27
第 3 章 处理复杂几何图形	29
3.1 Wavefront 文件格式	29
3.1.1 cube.obj	30
3.1.2 cube.mtl	31
3.2 准备 OBJ 浏览器代码	31
3.3 加载 OBJ	32
3.4 构建着色器	35
3.4.1 顶点着色器	35
3.4.2 片段着色器	36
3.4.3 顶点缓冲区对象	36

3.4.4 存储顶点数据	36
3.4.5 构建顶点数据数组(VBO)	38
3.4.6 构建元素数组 VBO	39
3.5 构建 VAO	40
3.6 渲染 Momo	42
3.7 处理 Touche 事件	44
3.8 逐顶点光照	45
3.8.1 顶点着色器的光照计算	46
3.8.2 修改片段着色器	47
3.8.3 更多 uniform 变量	48
3.9 对 Momo 进行美化	49
3.9.1 加载纹理	49
3.9.2 调整顶点数据	50
3.9.3 向顶点着色器添加 UV 支持	52
3.9.4 向片段着色器添加纹理 支持	52
3.9.5 绑定纹理	53
3.10 小结	54
第 4 章 构建场景	55
4.1 处理多个对象	55
4.2 代码结构	56
4.3 加载和绘制场景	57
4.4 着色器代码	61
4.5 不同的对象类型	62
4.6 绘制顺序	62
4.7 修复场景	63
4.7.1 Uber Shader	63
4.7.2 使用 Uber Shader	64
4.7.3 渲染循环对象分类	67
4.7.4 双面	69

4.8	逐像素光照	71	6.6.5	游戏逻辑	114
4.8.1	使顶点着色器更加丰富	71	6.7	3D 物理	118
4.8.2	获取使用了更多 Uber 的 片段着色器	72	6.7.1	Bullet 文件格式	118
4.8.3	封装实现代码	74	6.7.2	3D 弹珠游戏	119
4.9	小结	77	6.8	小结	125
第 5 章	优化	79	第 7 章	摄像头	127
5.1	基本应用程序	79	7.1	一触即发	128
5.2	从三角形到三角形带	80	7.2	摄像头视锥	130
5.3	构建三角形带	81	7.2.1	视锥构建方式	131
5.4	纹理优化	82	7.2.2	视锥剪切的实现	132
5.5	添加 16 位纹理转换	83	7.2.3	更多剪切函数	133
5.6	PVR 纹理压缩	84	7.3	摄像头飞行模式	134
5.7	仿造细节	85	7.4	带有碰撞检测的第一人称 摄像头	139
5.7.1	凸凹贴图的实现	85	7.5	3D 摄像头跟踪	141
5.7.2	精度限定符优化	86	7.6	带有碰撞的第三人称摄像头	143
5.7.3	法线贴图光照计算	88	7.7	小结	148
5.7.4	添加反射	90	第 8 章	路径发现	149
5.8	几何图形和着色器 LOD	91	8.1	Recast 和 Detour	149
5.9	纹理地图集	91	8.2	导航	150
5.10	在软件中管理状态	92	8.3	创建导航网格	151
5.11	自动着色器优化	93	8.4	3D 物理拾取	153
5.12	小结	94	8.5	玩家的自动驱动	157
第 6 章	实时物理	95	8.6	使路径点可见	159
6.1	物理对象类型	95	8.7	游戏“如果能就抓住我!”	161
6.2	物理形状	96	8.8	了解你的敌人	164
6.3	使用 Bullet	97	8.9	游戏状态逻辑	165
6.4	Hello Physics	97	8.10	小结	168
6.5	碰撞回调、触发器和接触点	102	第 9 章	音频及其他极佳的 游戏编程资料	171
6.5.1	Contact-Added 回调	103	9.1	OpenAL	172
6.5.2	Near 回调	105	9.2	OGG Vorbis	173
6.5.3	接触点	106	9.3	Hello World OpenAL 样式	173
6.6	2D 物理	107	9.4	初始化 OpenAL	174
6.6.1	更多形状!	108	9.5	播放静态内存声音	174
6.6.2	构建物理对象	111	9.6	定位声源	176
6.6.3	摄像头跟踪	112	9.7	钢琴游戏	177
6.6.4	用户交互	114			

9.7.1 加载静态的流式声音.....	178	第 11 章 高级 FX.....	237
9.7.2 颜色提取.....	182	11.1 渲染到纹理.....	238
9.7.3 钢琴游戏逻辑.....	185	11.2 后处理效果.....	238
9.7.4 最后的调整.....	188	11.2.1 第一渲染通道.....	241
9.8 滚球游戏.....	190	11.2.2 第二渲染通道.....	242
9.8.1 GFX 着色器.....	191	11.2.3 全屏通道和模糊着色器.....	243
9.8.2 链接定位声源.....	192	11.3 投影纹理.....	246
9.8.3 加速计驱动摄像头.....	196	11.4 投影着色器.....	249
9.8.4 廉价的 FX.....	199	11.5 投影实时阴影.....	250
9.8.5 游戏逻辑和调整.....	200	11.6 使用深度纹理投射阴影.....	254
9.9 小结.....	206	11.7 关于帧缓冲对象的 其他内容.....	255
第 10 章 高级光照.....	207	11.8 粒子.....	255
10.1 灯的类型.....	207	11.9 小结.....	257
10.2 使用光源.....	208	第 12 章 骨骼动画.....	259
10.2.1 定向灯着色器.....	211	12.1 传统的动画系统与最新的 动画系统.....	259
10.2.2 使用 Struct 作为 Uniform.....	214	12.2 MD5 文件格式.....	261
10.3 点灯.....	217	12.3 加载 MD5 网格.....	261
10.3.1 点光源着色器代码.....	218	12.4 对网格进行动画处理.....	264
10.3.2 光的衰减.....	221	12.4.1 LERP.....	266
10.3.3 带有衰减代码的点光源.....	222	12.4.2 SLERP.....	266
10.3.4 衰减 Uniform 变量.....	223	12.5 混合动画.....	267
10.3.5 球体点光源.....	224	12.6 相加混合.....	269
10.3.6 调整点光源代码.....	225	12.7 小结.....	271
10.3.7 聚光灯.....	227		
10.3.8 聚光灯着色器代码.....	229		
10.4 多个光源.....	231		
10.5 使着色器程序动态化.....	234		
10.6 小结.....	235		

第 1 章

入门

本章内容

- 了解本书使用的软件
- 下载本书的 SDK
- 理解 SDK 的体系结构
- 将相关项目导入到 IDE 中
- 理解本书介绍的模板应用程序
- 学习如何使用模板代码结构

本章将首先创建本书教程和示例使用的开发环境。

然后，快速介绍本书的 SDK 及其下载地址，以及 SDK 中包含的不同目录。随后学习如何将本书中现有的 SDK 项目和模板导入到你喜欢的 IDE 中，在后面学习本书的不同教程时将需要这么做。

在本章的最后一节，我们将学习跨平台的模板项目。最后，本章提供了一个快速教程，以帮助读者熟悉模板事件以及本书所有教程所使用的语言特点。

1.1 软件需求

本书所有内容都是基于 iOS 5.x+以及 Android 2.x+构建的；在撰写本书时，这是两种移动操作系统最新且最稳定的版本。

1.1.1 针对 iOS 开发人员

对于 iOS 操作系统来说，如果想使用本书，必须从 <http://developer.apple.com> 中获取最

新 iOS SDK 的副本，并在 Mac 上进行安装。

iOS SDK 提供了一个完全支持 GLES v2 的模拟器，所以即使没有 iOS 设备，或者没有从 Apple 处获取官方的 iOS 开发人员认证，也仍然可以使用本书。

1.1.2 针对 Android 开发人员

对于 Android 操作系统来说，构建环境并不像在 iOS 操作环境中那么简单。首先需要访问 <http://developer.android.com/sdk/installing.html>，并按照相关指示安装 Android SDK、Eclipse 和 ADT 插件。请注意，本书所使用的 Android SDK 版本是 v2.3.4，当然，也可以使用最新版本。

本书所有代码都使用了 C/C++，这意味着必须安装 Android 本机代码支持。为了完成开发环境的安装，请按照下面的步骤进行操作：

(1) 从 <http://developer.android.com/sdk/ndk/index.html> 处获取 Android SDK 的一份副本。虽然在撰写本书时使用的版本是 r5c，但本书的所有示例和教程在最新版本中也可以很好地运行。下载 Android NDK 压缩包，并在需要进行读写访问的计算机上解压缩。

(2) 为使用 Eclipse 编译和调试本机代码，需要安装 Sequoyah 插件。为此，首先从 Eclipse 主菜单中依次选择 Help | Install New Software | Available Software Sites | Sequoyah Metadata Repository，以便启动存储库。然后从 Work With 组合框中选择该条目，一旦加载完存储库数据，就可以选择并安装 Sequoyah Android Native Code Support，如图 1-1 所示。

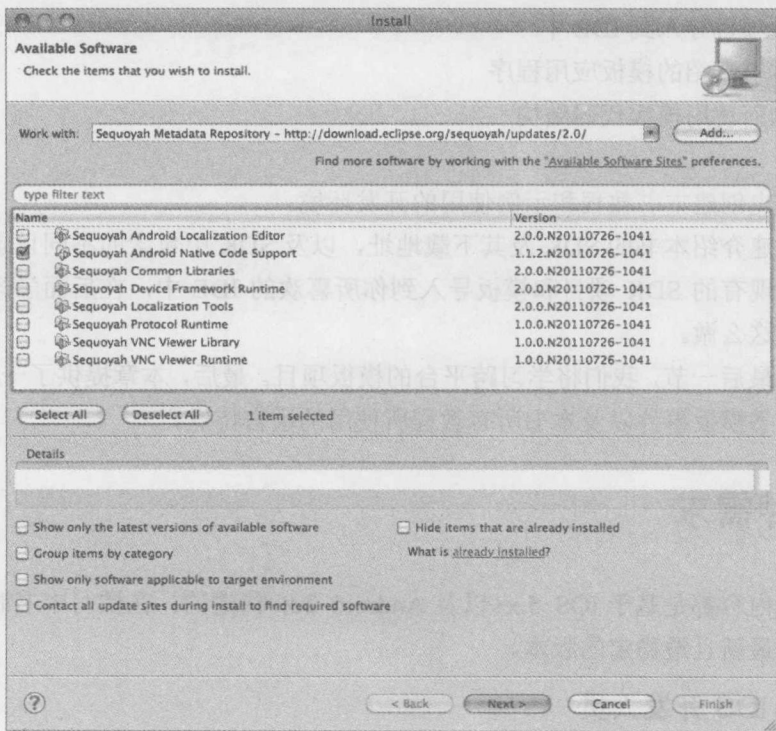


图 1-1 Sequoyah Native Code Support 插件

(3) 安装完 Sequoyah 后, 在主菜单中转到 Eclipse Preferences | Android | Native Development, 指定提取步骤(1)中解压缩的 Android NDK 的位置, 如图 1-2 所示。

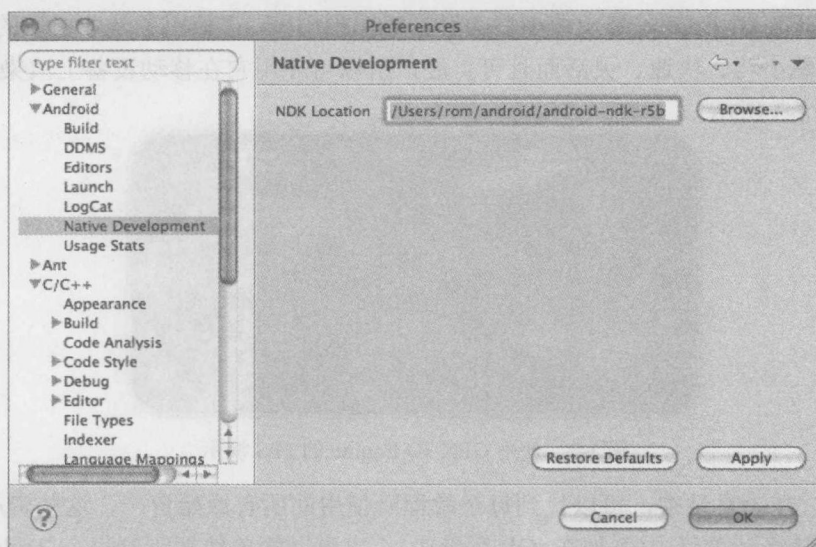


图 1-2 指定 Android NDK 的位置

恭喜你, 现在, 你的 Android 开发环境已经全部设置完毕! 然而请注意, 为在 Android 操作系统中使用本书, 还需要支持 OpenGL ES 2.0 的实际设备。当前的 Android SDK 所提供的模拟器仅支持 OpenGL ES 1.x, 而不支持 OpenGL ES 2.0。所以在 Android 系统上无法执行模拟器上的本地部署; 只有使用 GLES 2 时才支持设备部署。

1.2 下载本书的 SDK

一旦构建开发环境, 接下来就应该获取本书 SDK 的一份副本。可从 <http://www.wrox.com> 中下载官方的 SDK。或者如果想通过 GIT 下载 SDK, 也可以访问官方的 GFX 3D Engine 网站(<http://gfx.sio2interactive.com>), 并找到详细的说明。

如果下载了压缩文件, 只需在自己拥有读写访问权限的目录中进行解压缩就可以了。而如果是通过 GIT 下载压缩文件, 那么在你的设备上就已经可以使用所有的文件和 SDK 体系结构了。

本书 SDK 的体系结构非常简单。如果了解更多信息, 可参阅下面的目录列表。

- `_chapter#-#`: 该目录包含了在完成本书的教程后应该生成的最终结果。在阅读本书的过程中, 如果觉得相关的指示不够清楚, 或者不确定在什么地方插入代码, 又或者想要预先查看某一教程的最终结果, 都可以打开该目录。可在根目录中查找教程所使用的源文件(分别被命名为 `templateApp.cpp` 和 `templateApp.h`), 也可以访问根目录下的两个子目录, 它们分别包含 iOS 和 Android 项目文件。然后将项目加载到 IDE 中并从头开始重建。

- **common:** 该目录包含了创建本书中模板和教程所使用的 GFX 3D Engine 版本的免费和开源代码(在本书中,我们将使用迷你游戏和图形引擎),以及引擎所依赖的库源。GFX 3D Engine 是一种小型的轻量级图形引擎,它使用了专业引擎的很多内容。该引擎小巧、快速、灵活而且可扩展,所以可使用它在移动设备上渲染高级图形,如图 1-3 所示。

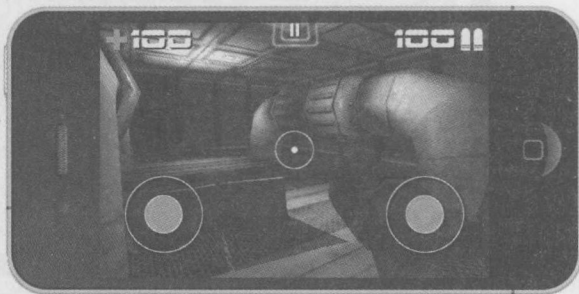


图 1-3 使用 GFX 3D Engine 的 FPS 演示

- **data:** 在该目录中,可以找到每个教程所使用的的所有原始资产。这些资产既可以动态地链接到项目中(例如在 iOS 系统中),也可以简单地复制到每个 Android 教程的 assets 目录中。请注意,所有原始项目 3D 场景都以.blend 形式(Blender 的默认文件扩展名)使用。虽然这并不是强制性的,但强烈建议为你的平台下载 Blender 的副本(可从 <http://blender.org> 处下载)。这样,就可以学习构建场景的方法以及如何连接资产和输出到 Wavefront OBJ(本书使用的官方 3D 模型交换格式)。
- **EULA:** 在该目录中,可找到本书 SDK 依赖的不同库的所有最终用户许可协议。如果想要发布使用了本书 SDK 的商业应用程序,务必确保该应用程序符合所有这些许可协议。
- **glsoptimizerCL:** 该目录包含了一个简单但功能强大的命令程序源,你可以使用该程序优化 GLSL 代码(如第 5 章所述)。
- **md5_exporter:** 该目录包含了 Blender 的 Python 脚本(v2.6x)。可使用该脚本将在 Blender 中创建的骨骼动画序列输出为 MD5 版本 10 文件格式(该脚本由 Paul Zirkle 慷慨提供)。
- **template:** 该目录包含了从头创建新项目时所使用的原始模板项目。
- **template_chapter#-#:** 为了加快速度并避免冗余,在阅读本书的教程时,可以复制这些目录,从而可以有一个良好的开始。通过使用默认的模板项目,可以避免从头构建所有内容。

1.3 导入项目

本书包含了 50 多个教程,其中有对简单技术的演示,也有成熟的游戏。为了能够在你的 IDE 中加载并重新生成这些项目,需要执行导入操作。为此,请遵循下面介绍的针对不同类型开发人员的相关指示。