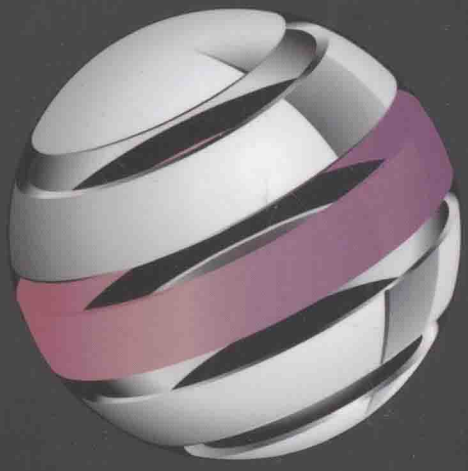
移动与嵌入式开发技术

开始使用Android 4开发Android移动游戏



# Android 4 游戏编程入门经典

[美] Mario Zechner
Robert Green
著

曾繁贰　于建业
王炜
译

Apress®

清华大学出版社

移动与嵌入式开发技术

# Android 4 游戏编程入门经典

[美] Mario Zechner 著
Robert Green
曾繁贰
于建业 译
王 炜

清华大学出版社
北 京

Mario Zechner, Robert Green
Beginning Android 4 Games Development
EISBN：978-1-4302-3987-1


北京市版权局著作权合同登记号　图字：01-2011-7430



**图书在版编目(CIP)数据**

Android 4 游戏编程入门经典 / (美) 策希纳(Zechner, M.)，(美) 格林(Green, R.) 著；曾繁贰，于建业，王炜 译. —北京：清华大学出版社，2012.11
书名原文：Beginning Android 4 Games Development
(移动与嵌入式开发技术)

ISBN 978-7-302-30105-9

Ⅰ.A…　Ⅱ.①策…②格…　③曾…　④于…　⑤王…　Ⅲ.①移动终端—应用程序—程序设计　Ⅳ.①TN929.53

中国版本图书馆 CIP 数据核字(2012)第 217838 号

**责任编辑**：王　军　于　平
**装帧设计**：牛艳敏
**责任校对**：邱晓玉
**责任印制**：张雪娇

**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**：北京富博印刷有限公司
**装 订 者**：北京市密云县京文制本装订厂
**经　　销**：全国新华书店
**开　　本**：185mm×260mm　　**印　　张**：34.25　　**字　　数**：876 千字
**版　　次**：2012 年 11 月第 1 版　　**印　　次**：2012 年 11 月第1 次印刷
**印　　数**：1～3000
**定　　价**：69.00 元

产品编号：040480-01

# 译 者 序

最近我们在移动互联网听到最多的词应该就是 Android，无疑它已经成为移动操作系统的一颗新星。越来越好的用户体验和应用程序已经博得了广大手机用户的喜爱。现在 Android 的全球市场占有率已经超越了 iPhone，在中国更是达到了 70%。这就给我们的应用开发者带来了无限的机会，也是广大应用开发者大展身手的绝佳平台。

现在，国内学习 Android 开发的人越来越多，而在 Android 平台上游戏应用占据了很大的一个比例，这就能看出市场对移动游戏有着巨大的需求。所以，很多开发者(特别是初学者)都希望能拥有一本 Android 游戏开发方面的书籍。本书就是专为这些游戏开发的初学者量身打造的。无论你是一个 Android 游戏开发的初学者，还是一个从其他平台转向 Android 平台，准备进行 Android 游戏开发的技术人员，本书都能够给你带来一些不错的启发。

当我第一次看到该书的英文原作的时候，就喜欢上了这本书，准确地说是喜欢上了该书的作者。我为他的认真和一种分享的精神所感动。因为它不再是一种简单的理论知识，而是一个又一个具体的实例。它让你在学习 Android 游戏开发的过程中不再感觉到枯燥乏味，而是获得一次又一次的惊喜。

本书的一个最大亮点是，它教你如何从零开始学习 Android 游戏开发。只要你具有一些 Java 的基础知识和对游戏开发的激情，那么开发出一款让自己满意的游戏(甚至是带来满意的收入)已不再是遥不可及的梦想。

如果你是一个没有 Android 基础的初学者，那么请从头开始阅读。如果你是一个具有 Android 基础的开发者，那么你可以跳过前面几章直接进入游戏开发部分。书中将会带领你完成游戏的策划、设计、编码和发布等过程。同时也以逐步深入的方式，从 Android 游戏的基本架构，再到 2D 和 3D 游戏，有针对性地帮你解决 Android 游戏开发中需要注意的问题。如果你是一个 Open GL 编程的爱好者，那本书更是一个不可多得的开发教程。

我非常庆幸自己能够成为该书的一名译者，同时，也希望我们的翻译能够为你的游戏开发带来一些帮助。翻译是一项辛苦的劳动，在此，我要感谢那些给予我们支持和帮助的家人和朋友们！

本书由曾繁贰、于建业和王炜翻译，同时得到了中科院李洪刚博士和清华大学出版社李阳老师的大力协助，在此表示由衷的感谢。由于时间比较紧张，译者水平有限，若译文有不当之处，敬请读者批评指正。

译者

# 作 者 简 介

**Mario Zechner** 白天是一名软件研发工程师，到了晚上就变身为热情高涨的游戏开发者，以 Badlogic Games 的名义发布游戏。他开发了基于 Android 平台的游戏 Newton，和基于 Windows、Linux 和 Mac OSX 的游戏 Quantum 以及其他众多游戏原型和小型游戏。目前，他正致力于一个名为 libgdx 的开源跨平台游戏开发解决方案。除了编写代码以外，他还积极撰写关于游戏开发的教程和文章，这些可以在网上特别是他的博客(http://badlogicgames.com)上免费查看。

**Robert Green** 是 Oregon 州 Portland 市的 Battery Powered Games 游戏工作室的创办人。他已经开发了 9 款 Android 游戏，包括 Deadly Chambers、Antigen、Wixel、Light Racer 和 Light Racer 3D。在全力开发和发布移动视频游戏以前，Robert 分别在 Minneapolis 和 Chicago 的软件公司中工作过，其中包括 IBM Interactive。Robert 目前的精力主要放在跨平台游戏开发和高性能移动游戏开发。Robert 经常在自己的博客 http://www.rbgrn.net 中更新关于游戏编程的知识。

# 前　言

大家好，欢迎来到 Android 游戏开发的世界。你阅读本书是为了学习 Android 游戏开发，希望我们能帮你实现这一目标。

我们将一起探讨许多游戏开发的思想和主题，包括 Android 基础知识、音频与图形编程、少量的数学或物理知识以及让人头疼的 OpenGL ES 编程。基于以上内容，我们将开发三款不同的游戏，并且其中一款为 3D 游戏。

如果你知道自己在做什么，游戏编程就会变得很容易。因此本书不仅提供了你可以重用的代码片段，同时也会让你知道游戏开发的方向。了解游戏开发的基本原理是进行日益复杂的游戏开发的关键。这样，你不仅可以写出类似于本书开发的各种游戏，而且还有足够的能力从网上或书店里吸取知识，甚至开创出你自己的游戏开发新天地。

## 本书读者对象

本书主要面向 Android 游戏编程的初学者。你不需要具备任何游戏开发的经验，因为本书将介绍 Android 游戏开发的所有基础知识。但是你必须具备一点 Java 基础知识。如果你对此觉得有些生疏，建议在线阅读 Bruce Eckel 撰写的 *Thinking in Java* 一书(Prentice Hall，2006)，该书是学习 Java 编程语言的优秀入门书籍。除此之外，你不需要具备其他知识，包括 Android 或 Eclipse。

该书同时也面向那些希望涉足 Android 的中级游戏开发人员。可能有些内容对他们来说已不是什么新知识，但仍有很多技巧和提示值得体会。如果说 Android 是一只奇怪的野兽，那么该书将是你向它吹响战斗号角的指南。

## 本书组织结构

本书将以循序渐进的方式逐步展示从基础到硬件加速等各种深度游戏开发的过程。在各章节中，我们创建了可重用的代码库，所以建议按顺序阅读本书各章。当然，有经验的读者可以跳过某些有把握的章节，但是要浏览一下跳过的章节中的程序清单，这样可以知道在后续章节中如何使用它们的类和接口。

# 本书的源代码

本书是完全独立的，包括运行本书的示例和游戏所需要的全部代码。但是，将本书代码复制到 Eclipse 开发环境时很容易出错。因为游戏应用程序不仅包含代码，它还需要其他一些无法从书中复制的资源。而且，复制本书的代码文本到 Eclipse 开发环境的过程中可能会引入一些错误。虽然 Robert(本书的技术编辑)和我做出了很大的努力来确保书中的程序清单少出错误，但我们知道仍有些地方还不尽如人意。

为了帮助你顺利学习，我们创建了一个 Google Code 项目，提供以下内容：

- 完整的源代码和资源遵循 GPL 3.0 版本，可通过该项目的 Subversion 存储库得到整个内容。
- 一个快速入门指南，让你知道如何以文本形式将项目导入到 Eclipse 开发环境，同时提供一个视频演示。
- 问题追踪系统，你可以提交任何发现的错误，包括书本身和书中附带的代码错误。一旦你向系统提交一个问题，我们就可以在 Subversion 存储库中修复。这样你就总能得到本书一份最新的、(希望)没有错误的源代码，当然其他读者也能从中受益。
- 一个讨论组，任何人都可以免费参加并讨论本书的内容，当然我也会参与讨论。

包含代码的每一章在 Subversion 存储库下都有一个对应的 Eclipse 项目。每个项目都是相对独立的，不依赖于其他项目，因为在本书进行的过程中，将逐渐改进其中的一些框架类。例如，第 5 章、第 6 章的代码就位于 ch06-mmom 项目中。

访问网址 http://code.google.com/p/beginning-android-games 可以找到 Google Code 项目。

# 目　录

# 第1章

# Android，后起之秀

对于我们这批20世纪90年代初出生的孩子们来说，成长过程中一直伴随着任天堂的Game Boy游戏机和Sega的Game Gear游戏机。我们花了无数个小时帮助Mario拯救公主，获得俄罗斯方块的最高分，通过连线和朋友们在Super RC Pro-Am中进行比赛等。对游戏的热情促使我们想要创建自己的世界，并与朋友们进行分享。我们开始在PC上编程，但很快就发现这些游戏的小杰作不能在游戏机上使用。我们仍然是充满热情的程序员，但随着时间的推移，我们对玩视频游戏的兴趣在慢慢减退。而且，我们的游戏机最终坏掉了……

转眼到了2011年，智能手机已经成为这个时代的新的移动游戏平台，相比于传统的专用的手持游戏设备(Nintendo DS或PlayStation PSP)。这又引起了我们的兴趣，我们开始调查哪种移动平台适合我们的开发需求。Apple公司的iOS系统看上去是个不错的游戏开发平台。不过，我们很快意识到该系统是封闭的，只有在经过苹果公司许可的情况下才能够和他人分享我们的作品，而且还需要一台Mac电脑来进行开发。所以，我们最终选择了Android平台。

我们马上爱上了Android平台，它的开发环境适用于所有的主流操作系统平台，没有任何限制。它还有一个很活跃的开发人员社区，在那里你可以寻求帮助以及获取全面的开发文档。任何人都可以免费使用和自由分享，同时如果你想盈利，可以在几分钟内把最新和最好的应用发布到一个全球性的电子市场上去，因为那里有数以百万计的用户。

接下来唯一要弄明白的事情是，如何利用PC游戏开发经验来开发Android平台的游戏。在后续章节中，我们将与你分享我们的开发经验，带你走进Android游戏开发世界。当然，这也许是个自私的计划，因为我们希望有更多的移动游戏出现。

让我们开始认识我们的新朋友吧——Android。

## 1.1 Android简介

2005年，谷歌收购了一家名叫Android的小型初创企业。那时Android首次被公众所关注，

同时也引发了大家对谷歌进军移动领域的猜测。直至 2008 年谷歌发布了 Android 1.0 版本，才使猜测烟消云散。Android 成为移动市场的一个新挑战者。从那时起，它就对已有平台，如 iOS(原来叫做 iPhone OS)和 BlackBerry 发起了挑战。Android 发展得相当好，每年的市场占有率都在提高。虽然移动技术的未来总有许多变化因素，但是有一个事实可以确定：Android 不会是昙花一现。

由于 Android 是开源的，因此手机制造商使用这一新平台的门槛很低。他们可以生产出各种价位的产品，通过修改 Android 系统的配置，以适应特定移动设备的需求。因此，Android 系统不仅适合于高端设备，也可以部署到低端设备上，正因为这样 Android 平台才有更广泛的受众面。

2007 年末开放手机联盟(OHA)的成立是 Android 取得成功的重要因素之一，该联盟包括宏达电(HTC)、高通(Qualcomm)、摩托罗拉(Motorola)和英伟达(NVIDIA)，他们共同致力于移动设备开放标准的制定。虽然 Android 的核心部分由谷歌负责开发，但其他的开放手机联盟成员也以不同的形式贡献了自己的一份力量。

Android 本身是一个移动操作系统和基于 Linux 内核 2.6 版本的平台，它可免费用于商业或者非商业用途。许多开放手机联盟的成员通过修改 Android 系统的用户界面来构建自定义的 Android 版本，以满足他们设备的需求，例如：宏达电的 Sense 和摩托罗拉的 MOTOBLUR。Android 的开源性也使得业余爱好者能够创建和发布他们自己的 Android 版本，这些常被称为 mod、固件或者 rom。截止本书创作时，一个最为大家所熟知的 rom 是由 Cyanogen 开发的，旨在为各种 Android 设备提供最新和最好的改进。

自 2008 年发布以来，Android 已经进行了 7 个版本的更新，所有版本代号均以甜品为名(Android 1.1 版本的除外，不过现在这个版本已经很少用了)。每个新版本都在原来 Android 平台的基础上增加了新功能，这些新功能或多或少给游戏开发人员带来一些启发。1.5 版本(Cupcake)开始支持在 Android 应用程序中包含本地库，而在以前的版本中只能使用纯 Java 编写应用程序。在我们更关注于程序性能的情况下，本地代码就能体现出它的优越性。1.6 版本(Donut)引入了对不同屏幕分辨率的支持，这对编写 Android 游戏有一些影响，所以本书中会有几次讨论该特性。2.0 版本(Eclair)增加了对多点触摸屏幕的支持，而 2.2 版本(Froyo)则向 Dalvik 虚拟机(VM)增加了即时编译(JIT)的功能，以提高 Java 程序在 Android 中的性能。JIT 能够大幅度提升 Android 平台下 Java 应用程序的性能，在最好的情况下，执行时间能够缩短到原来的 1/5。2.3 版本(Gingerbread)为 Dalvik 虚拟机增加了一个新的并发垃圾回收器。2011 年早期，Android 生产了一个针对平板电脑的版本，它的代号为 Honeycomb，版本号为 3.0。Honeycomb 中对应用程序接口(API)做了大量更改，比到目前为止的其他任何 Android 版本都多。到版本 3.1 时，Honeycomb 为拆分和管理高分辨率的大平板电脑屏幕提供了广泛的支持。它加入了更多与 PC 类似的功能，例如支持 USB 主设备和 USB 外设，包括键盘、鼠标和摇杆。这个版本唯一的问题是只面向平板电脑。Android 的小屏幕/智能手机版本仍然只能使用 2.3 版本。Android 4.0 发布了，情况有了改观。Android 4.0 的代号为 Ice Cream Sandwich(ICS)，它是将 Honeycomb(3.1)和 Gingerbread(2.3)合并后得到的一组在平板电脑和手机上都可以正常工作的公共功能。

ICS 对用户的热情是一种极大的激励，它对 Android 的用户界面做了很多改进，并且内置了浏览器、电子邮件客户端和照片服务等应用程序。ICS 为开发人员提供了许多便利，其中一

个就是合并了 Honeycomb UI API，使手机也支持大屏功能。ICS 还合并了 Honeycomb 的 USB 外设支持，使制造商也可以选择支持键盘和摇杆。ICS 添加了一些新的 API，例如 Social API，它为联系人、个人资料数据、状态更新和照片提供了一个集中的存储区。对于 Android 开发人员来说，值得庆幸的是 ICS 的核心仍然保持了良好的向后兼容性，这确保了巧妙构建的游戏可以在较老的版本(如 Cupcake 和 Eclair)上也可以很好地兼容。

## 1.2　版本分裂

Android 系统的快速更新也给那些制造商带来一些不便，因为制造商们在选择开发自有手机界面的时候，必须紧跟各种 Android 新版本的发布。这可能使一部手机没过几个月系统就已过时了，而运营商和制造商又没有提供新 Android 版本的更新，这就造成了各种 Android 版本的分裂。

版本的分裂有多方面的影响。对最终用户而言，因为使用的 Android 版本较老，这就意味着他们无法安装和使用一些新的应用程序的功能。而对于开发人员而言，这意味着当他们开发应用的时候，需要维护多个 Android 版本。早期 Android 版本的应用程序通常可以在新 Android 版本上正常运行，反之则不然。新版本中添加的一些功能(例如，多点触摸)当然无法在旧版本中使用，所以开发人员不得不针对不同的版本开发不同的代码。

在 2011 年，许多主要的 Android 设备制造商同意在 18 个月的设备生命期中支持最新的 Android OS。听起来时间好像不长，但是对于减轻分裂，这已经是迈出了一大步。这也意味着更多的手机会更快地支持 Android 的新功能(例如 Ice Cream Sandwich 中的新 API)。不过，市场上还是会有很多手机运行旧版本的 Android。如果游戏开发商想获得高市场占有率，就要使游戏至少能够在 6 个不同的 Android 版本，超过 400 种手机上运行。

听起来似乎很可怕，其实不必担心。通常情况下，需要采取的应对措施并不多。很多时候，我们完全可以忽视多种版本的存在，而假装只有一个 Android 版本。作为游戏开发人员，我们更多关注硬件的特性而忽视 API 的差异。这其实是分裂的另外一种形式，其他平台(如 iOS)也有这个问题，尽管没有 Android 那样明显。在本书中，我将对可能会给你开发 Android 游戏造成阻碍的分裂问题进行讨论。

## 1.3　谷歌的角色

虽然官方表示 Android 是开放手机联盟的心血结晶，但在实现 Android 和为 Android 提供必要的发展环境方面，谷歌无疑付出了最多的努力。

### 1.3.1　Android 开源项目

谷歌的贡献可归结为“Android 开源项目”。绝大多数的代码遵循 Apache License 2 协议，与其他开源许可协议(如 GNU General Public License，GPL)相比，这种协议更加开放，限制程

度更低。每个人都可以自由地使用这些源代码来构建自己的系统。然而，自称兼容 Android 的系统首先要通过 Android 的兼容性测试，这个过程可以确保应用程序能够与第三方的应用程序(像我们这样的开发人员开发的应用程序)基本兼容。兼容的系统可加入 Android 生态系统，其中包括 Android Market。

### 1.3.2 Android Market

Android Market 由 Google 于 2008 年 10 月对外公布，这是一个在线软件商店，用户可以从中搜索并安装第三方应用程序。该电子市场目前只能通过移动设备中的市场应用程序来访问，不过在不久的将来这种情况就会发生改变，因为谷歌已经承诺将发布可以使用桌面浏览器访问的网上商店。

电子市场允许第三方开发人员发布免费或付费的应用。付费应用可在许多国家购买，集成的购买系统使用 Google Checkout 处理汇率。Android 电子市场还提供了针对不同国家手动为应用程序单独定价的选项。

在注册了谷歌账户以后，用户可以访问电子市场。目前只能通过信用卡来购买电子市场中的应用。购买者自购买应用时起的 15 分钟内，可退还应用并获得全额退款，而以前的退款时间长达 24 小时。所以，这让用户有些难以接受。

开发人员为了能在电子市场上发布应用，必须注册一个谷歌开发人员账户并一次性支付 25 美元的费用。开发人员完成注册之后，几分钟内便可以开始上传应用程序。

Android 电子市场没有审批流程，而是依靠一个许可的制度。在安装一个应用之前，用户需要确认使用应用程序所需的一组权限。这些权限处理电话服务、网络接入、安全数字卡(SD)的访问等问题，确定它们之后方可安装该应用。系统正常运行的前提是用户执行了正确的操作。而在 PC 上，特别是 Windows 系统上，这样一种模式并不能很好地实现；但在 Android 上，至少直到现在它运行地很好，只有非常少的应用程序因为恶意用户行为被剔除出电子市场。

开发人员如果要出售应用程序，还需要注册一个谷歌 Checkout 账户来管理收入资金。注册是免费的，并且所有的商业金融交易都是通过该账号来进行的。谷歌还有一个包含在应用程序内的购买系统，它与 Android 电子市场和谷歌 Checkout 集成在一起。开发人员可以使用一个单独的 API 来处理应用程序内完成的购买交易。

### 1.3.3 挑战赛、设备播种计划和谷歌 I/O

为了吸引更多的开发人员加入 Android 平台，谷歌开始举行一些挑战赛。2008 年举行了第一次挑战赛——Android 开发人员挑战赛(ADC)，此次比赛提供了丰厚的奖金。同样谷歌在次年又举行了 Android 开发人员挑战赛，并吸引了大量的开发人员参与。也许有了前两次的基础，Android 已经吸引了大量的开发人员，感觉没有进一步进行推广的必要，谷歌在 2010 年停止了举办挑战赛。

谷歌于 2010 年初还推出了设备播种计划，给那些在电子市场上有一个或多个应用的下载量超过 5000 次，而且得到 3.5 星以上用户评价的开发人员提供一部 Motorola Droid、Motorola Milestone 或 Nexus One 手机。该计划在开发社区中十分受欢迎，不过刚开始的时候人们还有些

怀疑，因为许多人以为收到的电子邮件是一个精心设计的骗局。幸运的是，该计划得到了确切的落实，成千上万的设备被送往全球各个角落的开发人员手中。谷歌的这个举措给了第三方开发人员很大的甜头，让他们继续为该平台进行开发，并吸引更多的开发人员。

谷歌还为开发人员提供专门的 Android 开发机(ADP)。第一代 ADP 是 T-Mobile 的 G1 手机(也被称为 HTC Dream)，第二代 ADP 是 HTC 的 Magic 手机。同时，谷歌也发布了自己的手机——Nexus One。虽然很多人认为该机是 ADP 2 的后续，但是 Google 最终停止了向最终用户销售 Nexus One，现在仅对合作伙伴和开发人员出售。在 2010 年底，谷歌发布了第三代的 ADP，该机就是运行 Android 2.3 系统的三星 Nexus S 手机。所有的 ADP 都可通过 Android 电子市场进行购买，当然这需要你有一个开发人员账号。不过，Nexus S 手机可以通过谷歌的网站 www.google.com/phone 进行购买。

一年一度的谷歌 I/O 开发人员大会也是每个 Android 开发人员期待的盛会。在大会上，谷歌将会发布一些最新或最出色的技术和项目，而在近几年，Android 是其中最让人期待的部分。同时在大会期间，谷歌还会举行多场关于 Android 主题的会议，而这些内容将可以通过 YouTube 的 Google Developers 频道进行观看。在 2011 年度举办的谷歌 I/O 上，三星和谷歌向所有经常与会的人员免费赠送了 Galaxy Tab 10.1 设备。这标志着谷歌开始努力在平板电脑市场分一杯羹。

## 1.4　Android 的功能和体系结构

Android 并不是一个为移动设备而发布的 Linux 版本。因为当你进行 Android 开发时，并没有太大的可能会遇到 Linux 内核。对开发人员来说，Android 是一个平台，它抽象了底层的 Linux 内核，并使用 Java 语言进行应用开发。从上层应用方面来看，Android 有几个不错的功能：

- **应用框架**，提供丰富的 API 以供开发各种应用程序；同时它允许重用或替换系统或第三方应用程序。
- **Dalvik 虚拟机**，负责在 Android 平台上运行应用程序。
- **图形库**，可应用于 2D 和 3D 编程。
- **多媒体库**，支持常见的音频、视频和图片格式。如 Ogg Vorbis、MP3、MPEG-4、H.264 和 PNG 等。同时还提供专门的 API 用于回放音效，这在游戏开发中十分方便。
- **访问外设的 API**，如照相机、全球定位系统(GPS)、罗盘、加速度传感器、触摸屏、轨迹球和键盘等。请注意，并非所有的 Android 设备都有这些外设——这就是前面提到的“分裂”。

当然，Android 不只有上面所提到的这些功能，不过这些是我们进行游戏开发最为重要的部分。

Android 的体系结构是由一系列的组件所构成的，所有的组件都是基于它下面的组件，图 1-1 展示了 Android 主要组件的一个概览。