

<<Android攻略>>

图书基本信息

书名：<<Android攻略>>

13位ISBN编号：9787115284518

10位ISBN编号：7115284512

出版时间：2012-7

出版单位：人民邮电出版社

作者：[美]DaveSmith,[加]JeffFriesen

页数：360

字数：550000

译者：陈 钢

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<Android攻略>>

内容概要

《Android攻略》全方位讲解Google开放移动应用平台Android的各种开发技术。通过完整而真实的示例代码，介绍应用架构和多种特定于Android的API，讲述如何使用NetBeans、Eclipse、AppInventor等工具开发应用，如何通过AndroidNDK提高应用性能等。

编写Android应用程序的开发人员将是《Android攻略》的最大获益者。

作者简介

David Smith 资深软件开发工程师，重点关注移动开发领域，参与过从用SDK编写用户应用到构建和自定义Android源代码等各层次的开发工作。

博客地址为blog.wiresareobsolete.com），Twitter别名@devunwired。

Jeff Friesen 自由职业者，工作内容是培训和软件开发，主要是Java和Android。

除了本书之外，Jeff还为JavaWorld（www.javaworld.com）、informIT（www.informit.com）、java.net和DevSource（www.devsource.com）等网站撰写了很多技术文章。

可以到Jeff的网站上联系他，网址是tutortutor.ca。

<<Android攻略>>

书籍目录

目 录

- 第1章 Android入门 1
 - 1.1 Android简介 1
 - 1.2 Android演化史 2
 - 1.3 Android系统架构 3
 - 1.4 应用架构 6
 - 1.5 剖析Activity 11
 - 1.6 剖析Service 16
 - 1.7 剖析BroadcastReceiver 22
 - 1.8 剖析ContentProvider 23
 - 1.9 小结 58
- 第2章 用户界面攻略 60
 - 2.1 攻略2-1：自定义窗口 60
 - 2.2 攻略2-2：创建并显示视图 67
 - 2.3 攻略2-3：监控点击动作 69
 - 2.4 攻略2-4：适用于多种屏幕分辨率的图形资源 70
 - 2.5 攻略2-5：锁定活动方向 71
 - 2.6 攻略2-6：动态方向锁定 72
 - 2.7 攻略2-7：手动处理旋转 74
 - 2.8 攻略2-8：创建弹出菜单动作 76
 - 2.9 攻略2-9：自定义选项菜单 81
 - 2.10 攻略2-10：自定义返回按键 84
 - 2.11 攻略2-11：Home按键仿真 85
 - 2.12 攻略2-12：监控TextView的变动 86
 - 2.13 攻略2-13：自动滚动的TextView 89
 - 2.14 攻略2-14：动画视图 90
 - 2.15 攻略2-15：用可绘制资源做背景 97
 - 2.16 攻略2-16：创建自定义状态的可绘制资源 100
 - 2.17 攻略2-17：将遮罩应用到图片 102
 - 2.18 攻略2-18：创建持久的对话框 106
 - 2.19 攻略2-19：实现针对具体场景的布局 108
 - 2.20 攻略2-20：自定义键盘动作 112
 - 2.21 攻略2-21：隐藏软键盘 115
 - 2.22 攻略2-22：自定义AdapterView的空视图 116
 - 2.23 攻略2-23：自定义ListView行 117
 - 2.24 攻略2-24：制作ListView的节头部 122
 - 2.25 攻略2-25：创建组合部件 125
 - 2.26 好工具推荐：DroidDraw 128
 - 2.27 小结 132
- 第3章 通信和联网 133
 - 3.1 攻略3-1：显示Web信息 133
 - 3.2 攻略3-2：截获WebView事件 137
 - 3.3 攻略3-3：访问带JavaScript的WebView 138
 - 3.4 攻略3-4：下载图片文件 141
 - 3.5 攻略3-5：完全在后台下载 143

<<Android攻略>>

- 3.6 攻略3-6：访问RESTAPI 147
- 3.7 攻略3-7：解析JSON 153
- 3.8 攻略3-8：解析XML 156
- 3.9 攻略3-9：接收短信 160
- 3.10 攻略3-10：发送短信 162
- 3.11 攻略3-11：蓝牙通信 164
- 3.12 攻略3-12：查询网络连接状态 173
- 3.13 小结 174
- 第4章 实现设备硬件交互 175
 - 4.1 攻略4-1：整合设备位置 175
 - 4.2 攻略4-2：在地图上显示位置 178
 - 4.3 攻略4-3：在地图上标记位置 182
 - 4.4 攻略4-4：拍摄照片和录制视频 188
 - 4.5 攻略4-5：自定义摄像头覆盖层 192
 - 4.6 攻略4-6：录音 198
 - 4.7 攻略4-7：语音识别 201
 - 4.8 攻略4-8：播放音频视频 203
 - 4.9 攻略4-9：倾斜监控器 211
 - 4.10 攻略4-10：监控罗盘方向 214
 - 4.11 好工具推荐：SensorSimulator 218
 - 4.12 小结 223
- 第5章 数据持久化 224
 - 5.1 攻略5-1：制作设置界面 224
 - 5.2 攻略5-2：简单数据存储 228
 - 5.3 攻略5-3：读写文件 233
 - 5.4 攻略5-4：以资源的形式使用文件 238
 - 5.5 攻略5-5：管理数据库 240
 - 5.6 攻略5-6：查询数据库 245
 - 5.7 攻略5-7：备份数据 247
 - 5.8 攻略5-8：分享数据库 251
 - 5.9 攻略5-9：分享其他数据 258
 - 5.10 好工具推荐：SQLite3 264
 - 5.11 小结 272
- 第6章 与系统交互 273
 - 6.1 攻略6-1：从后台发送通知 273
 - 6.2 攻略6-2：创建定时和周期任务 276
 - 6.3 攻略6-3：规划周期任务 277
 - 6.4 攻略6-4：创建粘性操作 281
 - 6.5 攻略6-5：长时间运行的后台操作 286
 - 6.6 攻略6-6：启动其他应用 292
 - 6.7 攻略6-7：启动系统应用 294
 - 6.8 攻略6-8：让其他应用启动你的应用 298
 - 6.9 攻略6-9：与联系人交互 300
 - 6.10 攻略6-10：使用多媒体播放器 307
 - 6.11 攻略6-11：保存到MediaStore 309
 - 6.12 小结 311
- 第7章 使用库 312

<<Android攻略>>

7.1	攻略7-1：创建Java库JAR	312
7.2	攻略7-2：使用Java库JAR	314
7.3	攻略7-3：创建Android库项目	316
7.4	攻略7-4：使用Android库项目	319
7.5	攻略7-5：绘图	321
7.6	攻略7-6：消息推送实战	330
7.7	小结	338
附录A	Android的脚本层	339
附录B	AndroidNDK	345
附录C	App设计指南	355

<<Android攻略>>

章节摘录

版权页：插图：监听模式 点击Listen for Sharers按钮，启动应用监听收到的连接。

设备要接收其他未知设备的连接就必须设置可发现状态。

应用通过检测适配器的模式是否等于SCAN_MODE_CONNECTABLE_DISCOVERABLE来判断设备是否处于可发现状态。

如果适配器不满足要求，就会给系统发送一个Intent，告知用户应该让设备处于可发现状态，具体方法与要求用户启动蓝牙类似。

如果用户接受了该请求，Activity就会返回设备处于可发现状态的时长；如果用户拒绝了该请求，Activity就会返回Activity.RESULT_CANCELED。

这个例子会在onActivityResult () 方法中处理用户拒绝请求的行为，即终止应用。

如果用户启用了可发现状态，或是设备已经处于可发现状态，就会创建并执行Accept-Task。

这个任务会为我们定义的服务的UUID创建监听器端口，通过这个端口等待连接请求。

在收到有效的请求时，就会接受，然后应用会切换到Connected Mode（已连接模式）。

在设备处于监听状态的过程中，蓝牙的名称会被设为特定的值（SEARCH-NAME）来加速发现的过程（具体原因见后面的“搜索模式”一节）。

连接建立后，适配器就会恢复默认名称。

搜索模式 点击Connect and Share按钮会告诉应用开始搜索其他可连接的设备。

首先是启动蓝牙的发现过程，然后在Broadcast Receiver中处理结果。

用BluetoothAdapter.startDiscovery () 启动发现后，在两种情况下Android会异步地回调广播：一种情况是发现了其他设备，另一种情况是发现过程结束。

在Activity对用户可见时，私有的接收器mReceiver始终是注册的，每当发现一个新设备时就会接收相应的广播。

在介绍监听模式时，监听设备的设备名称会被设为一个特定的值。

每次进行发现时，接收器就会检查设备的名称是否匹配已知的值。

如果找到设备名称与已知的值相匹配，接收器就会尝试与其连接。

这对于提高发现过程的速度是很重要的，因为如果不这样做，验证设备的唯一方法就是尝试连接到特定的服务UUID，检测操作是否成功。

连接蓝牙的过程即耗时间又耗资源，只有在非常必要的时候才这么做。

这种匹配设备的方法还可以避免要求用户手动选择连接设备。

应用可以智能地找到另一个运行着相同应用的设备，然后在监听模式下完成传输，没有了用户的参与，也意味着这个值不能重复且非常少见，这样才能避免找到偶然有相同名称的其他设备。

找到匹配的设备后，就会停止发现过程（因为这是非常耗时、耗资源的，会拖慢连接的速度），连接到服务的UUID。

在连接成功后，应用就进入了已连接模式。

<<Android攻略>>

编辑推荐

《Android攻略》Android是Google开源移动开发平台，如今已经成为最主要的移动平台之一。对开发人员而言了解Android才能把握机会，跟上市场的变化。

《Android攻略》是一本资源丰富、可随时查询的实例手册，针对开发人员要解决的具体问题。提供有助于高效完成工作的实用建议。书中包括大量绝妙的代码示例。可以直接应用到开发人员的研发项目中。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>