# 各种资源

### Irritating Stick <http://www.ebay.com/ctg/Irritating-Stick-Sony-PlayStation-1-1999-/2274>

《Android应用开发全程实录电子书》《Android 2.0游戏开发实践宝典》源码Android游戏开发视频教程www.microdu.com》[压缩包] <http://www.verycd.com/topics/2864445/>

项目主页 <http://www.studentdeveloperchallenge.com/>

开发下载

开发工具<http://developer.android.com/sdk/index.html>

三星Kies <http://www.samsungapps.com/venus/about/onPc.as>

SPen驱动 <http://developer.samsung.com/s-pen-sdk/sdk/S-Pen-SDK-2-2-5>

开发资源

Android开发网站<http://developer.android.com/develop/index.html>

SPen网站 <http://developer.samsung.com/s-pen-sdk>

手机应用开发网站 <http://www.microdu.com>

Android游戏开发之旅总贴 <http://www.microdu.com/thread-359-1-1.html>

《Android 2.0游戏开发实践宝典》源码Android游戏开发视频教程www.microdu.com已更新到16章了》 <http://blog.csdn.net/rosonex/article/details/6077655>

雨松MOMO的程序世界专栏<http://blog.csdn.net/xys289187120>

### [Android游戏开发之检测游戏碰撞的原理实现（九）](http://blog.csdn.net/xys289187120/article/details/6630302)

[**android 检测碰撞汇总**](http://www.cnblogs.com/tijing/archive/2012/10/08/2715563.html) [Android UI开发专题之绘图基础](http://www.linuxidc.com/Linux/2011-08/41239.htm) [Android UI开发专题之各种Drawable](http://www.linuxidc.com/Linux/2011-08/41240.htm)

[**通过参考点写一个不规则碰撞**](http://bbs.9ria.com/thread-63072-1-1.html)[**基于像素的不规则对象的通用高效碰撞检测**](http://www.cc-space.com/?p=1511)**（flash）**

[Android 软件开发与游戏开发系列博文大合集](http://blog.csdn.net/xys289187120/article/details/6889408)

[原型程序](http://www.ebay.com/ctg/Irritating-Stick-Sony-PlayStation-1-1999-/2274)

[Android软件与游戏开发](http://blog.csdn.net/column/details/androidsw.html)专栏创建者：[雨松MOMO](http://blog.csdn.net/xys289187120) [雨松MOMO微盘](http://vdisk.weibo.com/u/2332920021?mode=list&type=&page=1)

[Android中View与SurfaceView的区别要点](http://m.poorren.com/android-view-surfaceview/) [**Android显示系统之View与SurfaceView更新屏幕的区别**](http://blog.csdn.net/conowen/article/details/7843487)

[Android SurfaceView 详解](http://www.55zm.com/a/20120625/37574.html)

### [Android笔记：invalidate()和postInvalidate() 的区别及使用](http://blog.csdn.net/mars2639/article/details/6650876)

### [Invalidate和postInvalidate的区别](http://blog.csdn.net/ssstudio/article/details/6232202) [Android invalidate & postInvalidate的区别与用法](http://hi.baidu.com/do_touch/item/cedb7fb84381c8412bebe34d)

[**【eoeAndroid社区索引】android用户界面之SurfaceView教程实例汇总**](http://www.eoeandroid.com/thread-168155-1-1.html)

[**Android之SurfaceView学习（一）**](http://www.cnblogs.com/xuling/archive/2011/06/06/android.html)

# 地图编辑器

## Mappy

[www.Tilemap.co.uk](http://www.Tilemap.co.uk)

Mappy <http://www.tilemap.co.uk/mappy.php> Java：<http://www.alienfactory.co.uk/javamappy/>

游戏演示：<http://www.tilemap.co.uk/platformone.html> <http://www.tilemap.co.uk/pyrastak.html>

## Tiled

<http://www.mapeditor.org/> 有QT和Java两个版本

## Tile Studio

<http://tilestudio.sourceforge.net/>

## open tume

<http://opentume.com/>

## Games Factory Pack

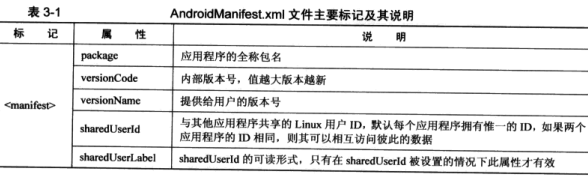
<http://www.arrakis.es/~esanchez/>

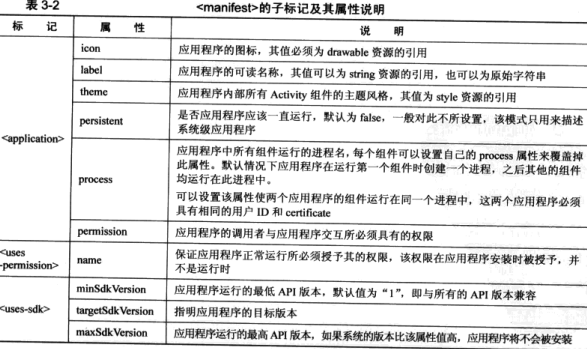
# 机制

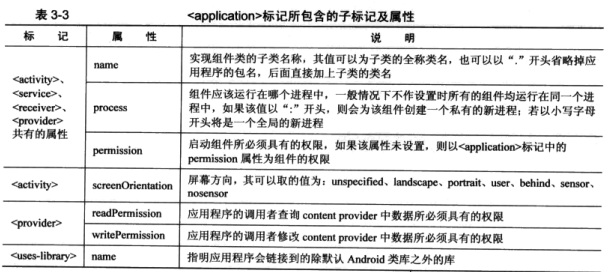
## AndroidManifest.xml

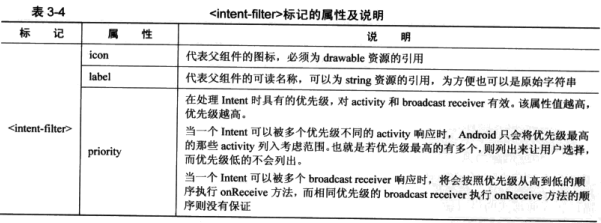


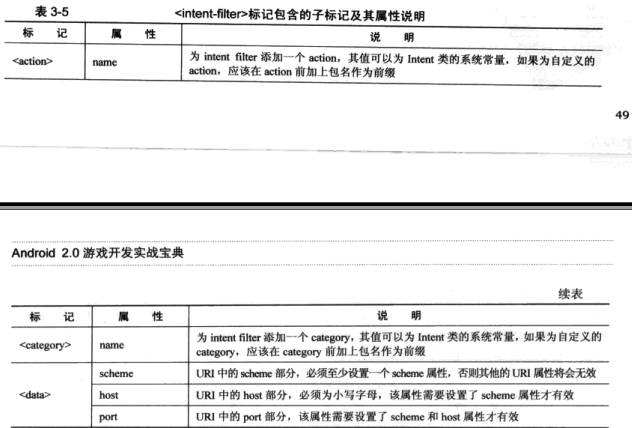
AndroidManifest.xml文件的结构如下：











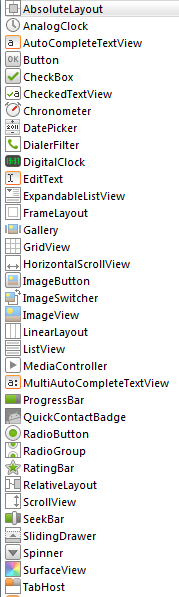
* [<activity>](http://developer.android.com/guide/topics/manifest/activity-element.html) elements for activities
* [<service>](http://developer.android.com/guide/topics/manifest/service-element.html) elements for services
* [<receiver>](http://developer.android.com/guide/topics/manifest/receiver-element.html) elements for broadcast receivers
* [<provider>](http://developer.android.com/guide/topics/manifest/provider-element.html) elements for content providers

## R.java

R.java文件是由开发工具自动生成的，用于建立Java代码和其他资源之间的联系。这个文件映射的资源有:icon图标，Layout布局，string字符串，图像变化的XML文件

每个App至少有一个Activity，每个Activity[是以Class方式从Activity类扩张]。一个App可以包含多个Activity，但是一般选一个Activity显示，其他的Activity通过第一个上面的startActivity方法调用来启动。

每个Activity在创建的时候至少对应一个Layout[是以XML文件定义(通过R.java文件映射)]

每个Layout里面定义布局类型，并且还含有多个View等UI组件

注意，不是所有View都需要在Layout里面定义，这个例子给出了一个用代码（java文件继承View类）来实现View的方法，例子：3.2

## 应用程序的内部通信

主线程，子线程。Handler类用于主线程和子线程进行通信。

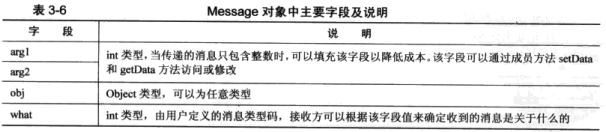
在主线程中维护一个消息队列。

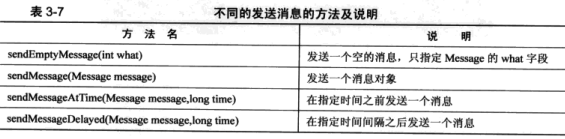
Handler使得线程间通过Message和Runnable对象传递和处理。

每个Handler对象与一个线程及其消息队列关联。

创建Handler对象时，将Handler对象于线程和消息队列绑定。然后，handler对象会向消息队列传递Message和Runnable并处理执行队列中的元素。

Handler最主要的用法是：安排Message和Runnable对象，使其在未来的某个时刻被处理或运行。





# 疑难问题

|  |
| --- |
| **Android 模拟器(emulator)无法启动AVD解决方案** |
| ANDROID\_SDK\_HOME |
| 新添加一个环境变量，名字为 "ANDROID\_SDK\_HOME”，然后把变量值改为你想将".android”文件夹(AVD信息)所放置的位置 |
| 一 般来说，创建虚拟设备AVD时，默认将创建好的AVD放到了“C:\Documents and Settings\Administrator\.android“ 下面，因而需要将"C:\Documents and Settings\Administrator\"添加到你的"ANDROID\_SDK\_HOME”环境变量中。 |
|  |
| 不会添加环境变量的看这 里：在桌面上，右键我的电脑→属性→“高级”选项卡→”环境变量”→再弹出的窗口中，选“系统变量”下方的新建→在“变量名”中填入 ANDROID\_SDK\_HOME ，在"变量值"中填入“C:\Documents and Settings\Administrator\” 。→OK，搞定。 |

# 例子

## 布局和按钮

2.1-2.6

## 画图

2.7

## 图片

2.8

## 图片旋转

2.9 动画的两种形式:渐变动画tweened animation和帧动画frame by frame， 渐变动画包括 透明度alpha，拉伸scale，位移translate和旋转rotate，动画使用的是PNG格式的图片，这种图片背景是透明色。

## 声音

2.10 SoundPool-短促，反应快 MediaPlayer较长但不太要求时间，长音乐用的是Mid，断音乐用的是ogg

## App内部线程通信

3.4 主线程，子线程和Handler的使用方法

## Intent通信

3.5 Intent对象运行在不同(也可以相同)的Activity，Service和broadcast Receiver组件之间。

对于Activity组件，调用Context.startActivity，Context.startActivityForResult， 等方法

对于Service组件，调用Context.startService和Context.bindService方法

对于Broadcast Receive组件，调用sendBroadcast发送广播

一个Intent对象包括：Component Name, Action, Data, Category, Extras and Flags

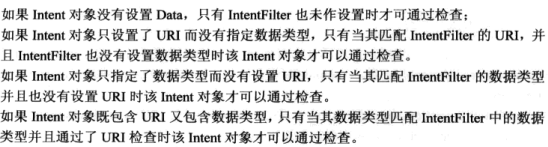
## IntentFilter

3.5 白名单机制。Activity，Service和Broadcast Receive可以有一个或多个IntentFilter。

所有IntentFilter必须在AndriodManifest.xml文件中声明，除了用于过滤Broadcast Receive的。

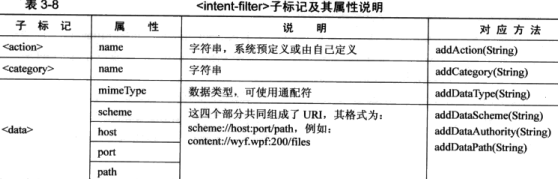
Action，Data和Category必须都符合才能被过滤到。

Action只能设置一个。IntentFilter的Action不能为空，否则无法匹配。反之，Intent的Action为空则符合所有IntentFilter的检查。

Data可以设置多个。Data部分而后Action相似。一个Data包括[URI 和 数据类型]。并且分别比较：

可设置多个Category。如果一个Intent能够通过IntentFilter检查，则意味着Intent的Category是IntentFilter的Category的子集。也就是说，如果Intent的Category个数为0（即空集）则可以通过任何IntentFilter。

IntentFilter可以在AndroidManifest.xml声明或在Code中创建。



## Service组件通信

3.6 代码中创建IntentFilter，创建Service线程，通过Broadcast使用Intent/IntentFilter进行通信。需要将一个DataReceiver注册到（DataReceiver的对象和IntentFilter对象绑定在一起）

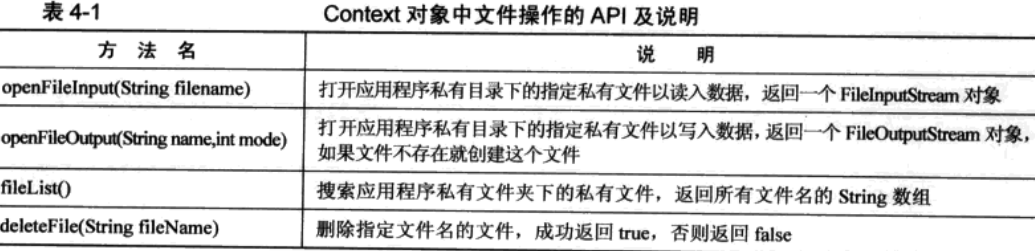
# 文件

## 私有文件

存在于/data/data/<appName>

## SD卡文件

64页



## 资源文件res/raw

4.2

## 资产文件assets

4.2

## SQLite

4.3 69页

## Content Provider

4.4 75页

## Preferences

以键值对的形式在应用程序的私有文件夹下存储简单数据类型[Boolean,int,float,long,String] 4.5 79页

# 传感器

加速度Accelerometer，姿态Orientation，磁场Magnetic Field，光Light

## 开发流程

82页

创建SensorManager对象

实现SensorListener接口（onAccuracyChanged精度发生变化时调用，onSensorChanged数据发生变化）

在程序中注册监听（registerListener）和取消监听（unregisterListener）

# 物理

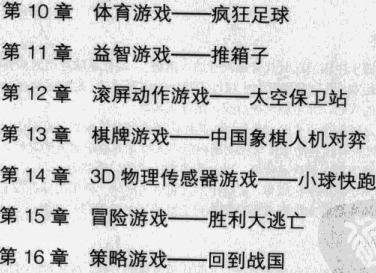
## 小球掉落

7.1 131页

## 粒子系统

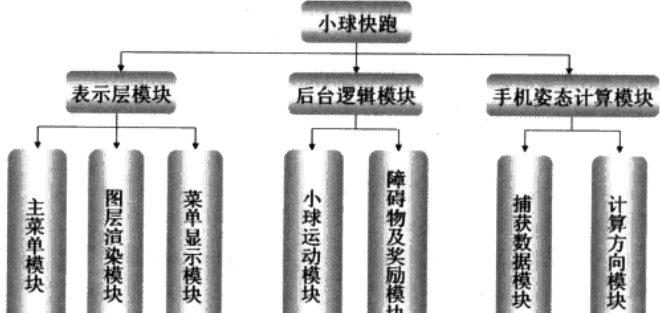
7.2 火焰粒子系统 7.3 瀑布粒子系统 142页

# 游戏例子



小球快跑358页

图层开发p378



WelcomeView继承于SurfaceView，播放欢迎动画。

GameView继承于SurfaceView，进行游戏。

WelcomeDrawThread和WelcomeThread都绘制WelcomeView

WelcomeDrawThread定时重绘，WelcomeThread修改内容

DrawThread负责重绘GameView。

GameMap提供地图信息。378页

碰撞，小球运动控制。383页

# 任务分解

## 碰撞

## 地图解析

## 游戏主程序

## 关卡设计（图）

# 待解决问题

技能－能量槽

血槽－三次机会

小怪

旋转

# 难度的地方

碰撞，旋转，地图

# 画图

## 画图函数

### [Android中Canvas的clipRect参数Region.OP的理解](http://blog.csdn.net/zealot1024/article/details/7640856)

|  |
| --- |
|  |

为了方便说明，把第一次clipRect的绘制范围设为A，第二次clipRect设定的范围设为B

Op.DIFFERENCE，实际上就是求得的A和B的差集范围，即A－B，只有在此范围内的绘制内容才会被显示；

Op.REVERSE\_DIFFERENCE，实际上就是求得的B和A的差集范围，即B－A，只有在此范围内的绘制内容才会被显示；；

Op.INTERSECT，即A和B的交集范围，只有在此范围内的绘制内容才会被显示；

Op.REPLACE，不论A和B的集合状况，B的范围将全部进行显示，如果和A有交集，则将覆盖A的交集范围；

Op.UNION，即A和B的并集范围，即两者所包括的范围的绘制内容都会被显示；

Op.XOR，A和B的补集范围，此例中即A除去B以外的范围，只有在此范围内的绘制内容才会被显示；

测试主要代码很简单如下：

  canvas.save();

  Rect rect = new Rect(20, 20, 200, 200);  
  canvas.clipRect(rect);  
  Paint paint = new Paint();  
  paint.setColor(Color.BLACK);  
  paint.setStyle(Style.FILL);  
  canvas.drawRect(rect, paint);  
    
  Rect clip = new Rect(100, 100, 300, 300);  
  canvas.clipRect(clip, Op.XOR);  
  paint.setColor(Color.BLUE);  
  canvas.drawRect(clip, paint);

  canvas.restore();