

[首页](#)[所有文章](#)[行业动态](#)[技术分享](#)[产品设计](#)[工具资源](#)[安卓小组](#)[更多频道 ▾](#)[- 导航条 - ▾](#)[伯乐在线](#) > [安卓 - 伯乐在线](#) > [所有文章](#) > [技术分享](#) > Android样式的开发：shape篇

Android样式的开发：shape篇

2015/11/25 · [技术分享](#) · [shape](#), [样式](#)

分享到：

5 原文出处：[Keegan/小钢](#)[Android项目重构之路：架构篇](#)[Android项目重构之路：界面篇](#)[Android项目重构之路：实现篇](#)[Android技术积累：开发规范](#)

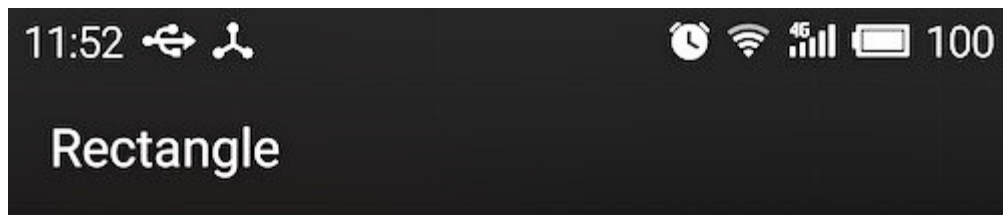
一个应用，应该保持一套统一的样式，包括Button、EditText、ProgressBar、Toast、Checkbox等各种控件的样式，还包括控件间隔、文字大小和颜色、阴影等等。web的样式用css来定义，而android的样式主要则是通过shape、selector、layer-list、level-list、style、theme等组合实现。我将用一系列文章，循序渐进地讲解样式的每个方面该如何实现。第一个要讲的就是shape，最基础的形状定义工具。

一般用shape定义的xml文件存放在drawable目录下，若项目没有该目录则新建一个，而不要将它放到drawable-hdpi等目录中。

使用shape可以自定义形状，可以定义下面四种类型的形状，通过android:shape属性指定：

- **rectangle**: 矩形，默认的形状，可以画出直角矩形、圆角矩形、弧形等
- **oval**: 椭圆形，用得比较多的是画正圆
- **line**: 线形，可以画实线和虚线
- **ring**: 环形，可以画环形进度条

rectangle是默认的形状，也是用得最多的形状，一些文字背景、按钮背景、控件或布局背景等，以下是一些简单的例子：



默认矩形

加了内间距的矩形

加了渐变的矩形

圆角矩形

只有上半部分圆角的矩形

弧形边的矩形

加了实线描边的矩形

加了虚线描边的矩形

镂空描边的矩形

实现上面的那些效果，都用到了以下这些特性：

- **solid:** 设置形状填充的颜色，只有android:color一个属性

- **padding:** 设置内容与形状边界的内间距, 可分别设置左右上下的距离
 - *android:left* 左内间距
 - *android:right* 右内间距
 - *android:top* 上内间距
 - *android:bottom* 下内间距
- **gradient:** 设置形状的渐变颜色, 可以是线性渐变、辐射渐变、扫描性渐变
 - *android:type* 渐变的类型
 - *linear* 线性渐变, 默认的渐变类型
 - *radial* 放射渐变, 设置该项时, *android:gradientRadius*也必须设置
 - *sweep* 扫描性渐变
 - *android:startColor* 渐变开始的顏色
 - *android:endColor* 渐变结束的顏色
 - *android:centerColor* 渐变中间的顏色
 - *android:angle* 渐变的角速度, 线性渐变时才有效, 必须是45的倍数, 0表示从左到右, 90表示从下到上
 - *android:centerX* 渐变中心的相对X坐标, 放射渐变时才有效, 在0.0到1.0之间, 默认为0.5, 表示在正中间
 - *android:centerY* 渐变中心的相对Y坐标, 放射渐变时才有效, 在0.0到1.0之间, 默认为0.5, 表示在正中间
 - *android:gradientRadius* 渐变的半径, 只有渐变类型为radial时才使用
 - *android:useLevel* 如果为true, 则可在LevelListDrawable中使用
- **corners:** 设置圆角, 只适用于rectangle类型, 可分别设置四个角不同半径的圆角, 当设置的圆角半径很大时, 比如200dp, 就可变成弧形边了
 - *android:radius* 圆角半径, 会被下面每个特定的圆角属性重写
 - *android:topLeftRadius* 左上角的半径
 - *android:topRightRadius* 右上角的半径
 - *android:bottomLeftRadius* 左下角的半径
 - *android:bottomRightRadius* 右下角的半径
- **stroke:** 设置描边, 可描成实线或虚线。
 - *android:color* 描边的颜色
 - *android:width* 描边的宽度
 - *android:dashWidth* 设置虚线时的横线长度
 - *android:dashGap* 设置虚线时的横线之间的距离

接下来说下实际怎么使用。以下是加了虚线描边的矩形的代码, 文件命名为bg_rectangle_with_stroke_dash.xml, 放在drawable目录下:

XHTML

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <!-- android:shape指定形状类型, 默认为rectangle -->
3 <shape xmlns:android="http://schemas.android.com/apk/res/android"
4     android:shape="rectangle">
5     <!-- solid指定形状的填充色, 只有android:color一个属性 -->
6     <solid android:color="#2F90BD" />
7     <!-- padding设置内容区域离边界的间距 -->
8     <padding
9         android:bottom="12dp"
10        android:left="12dp"
11        android:right="12dp"
12        android:top="12dp" />
13     <!-- corners设置圆角, 只适用于rectangle -->
14     <corners android:radius="200dp" />
15     <!-- stroke设置描边 -->

```

[首页](#)[资讯](#)[文章](#) ▾[资源](#)[小组](#)[❤ 相亲](#)[频道](#) ▾[➔ 登录](#)[👤 注册](#)

```
18 android:color="@android:color/darker_gray"
19 android:strokeWidth="4dp"
20 android:strokeDash="4dp" />
21 </shape>
```

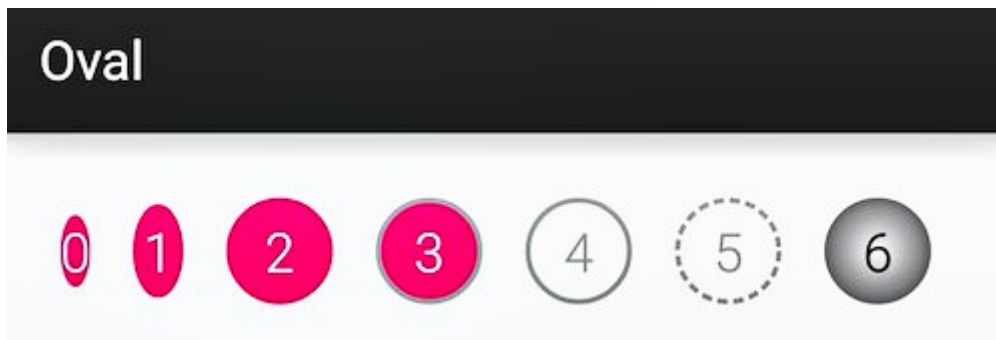
接着在要使用的view里引用就可以了，例如本例中用做TextView的背景：

XHTML

```
1 <TextView
2     android:layout_width="wrap_content"
3     android:layout_height="wrap_content"
4     android:layout_margin="8dp"
5     android:text="加了虚线描边的矩形"
6     android:textSize="16sp"
7     android:textColor="@android:color/white"
8     android:background="@drawable/bg_rectangle_with_stroke_dash" />
```

oval

oval用来画椭圆，而在实际应用中，更多是画正圆，比如消息提示，圆形按钮等，下图是一些例子：



上面的效果图应用了solid、padding、stroke、gradient、size几个特性。size是用来设置形状大小的，如下：

- **size:** 设置形状默认的大小，可设置宽度和高度
 - `android:width` 宽度
 - `android:height` 高度

数字0是默认的椭圆，只加了solid填充颜色，数字1则加了上下左右4dp的padding，后面的数字都是正圆，是通过设置size的同样大小的宽高实现的，也可以通过设置控件的宽高一致大小来实现。数字3加了描边，数字4是镂空描边，数字5是虚线描边，数字6用了radial渐变。注意，使用radial渐变时，必须指定渐变的半径，即`android:gradientRadius`属性。

以下是渐变的代码实现，文件为`bg_oval_with_gradient.xml`：

XHTML

```
1 <?xml version="1.0" encoding="utf-8"?><!-- android:shape指定形状类型，默认为rectangle -->
2 <shape xmlns:android="http://schemas.android.com/apk/res/android"
3     android:shape="oval">
4     <!-- padding设置内间距 -->
5     <padding
6         android:bottom="4dp"
```

[首页](#)[资讯](#)[文章 ▾](#)[资源](#)[小组](#)[❤ 相亲](#)[频道 ▾](#)[🔑 登录](#)[👤 注册](#)[?](#)

```
9      android:top="4dp" />
10     <!-- size设置形状的大小 -->
11     <size
12         android:width="40dp"
13         android:height="40dp" />
14     <!-- gradient设置渐变 -->
15     <gradient
16         android:endColor="#000000"
17         android:gradientRadius="40dp"
18         android:startColor="#FFFFFF"
19         android:type="radial" />
20 </shape>
```

引用的代码：

```
1 <TextView
2     android:layout_width="wrap_content"
3     android:layout_height="wrap_content"
4     android:gravity="center"
5     android:layout_margin="8dp"
6     android:text="6"
7     android:textSize="20sp"
8     android:textColor="@android:color/black"
9     android:background="@drawable/bg_oval_with_gradient" />
```

XHTML

line



line主要用于画分割线，是通过stroke和size特性组合来实现的，先看虚线的代码：

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <shape xmlns:android="http://schemas.android.com/apk/res/android"
3     android:shape="line">
4     <!-- 实际显示的线 -->
5     <stroke
```

XHTML

[首页](#) | [资讯](#) | [文章](#) ▾ | [资源](#) | [小组](#) | [❤ 相亲](#)
[频道](#) ▾[🔑 登录](#)[👤 注册](#)

```

8      android:dashGap="2dp"
9      android:dashWidth="4dp" />
10     <!-- 形状的高度 -->
11     <size android:height="4dp" />
12 </shape>

```

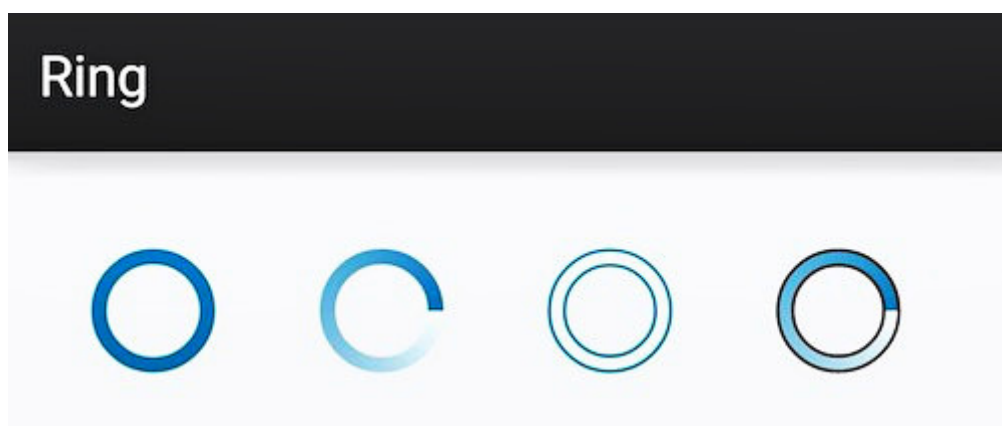
画线时，有几点特性必须要知道的：

1. 只能画水平线，画不了竖线；
2. 线的高度是通过stroke的android:width属性设置的；
3. size的android:height属性定义的是整个形状区域的高度；
4. size的height必须大于stroke的width，否则，线无法显示；
5. 线在整个形状区域中是居中显示的；
6. 线左右两边会留有空白间距，线越粗，空白越大；
7. 引用虚线的view需要添加属性android:layerType，值设为" software"，否则显示不了虚线。

ring

首先，shape根元素有些属性只适用于ring类型，先过目下这些属性吧：

- *android:innerRadius* 内环的半径
- *android:innerRadiusRatio* 浮点型，以环的宽度比率来表示内环的半径，默认为3，表示内环半径为环的宽度除以3，该值会被android:innerRadius覆盖
- *android:thickness* 环的厚度
- *android:thicknessRatio* 浮点型，以环的宽度比率来表示环的厚度，默认为9，表示环的厚度为环的宽度除以9，该值会被android:thickness覆盖
- *android:useLevel* 一般为false，否则可能环形无法显示，只有作为LevelListDrawable使用时才设为true



第一个图只添加了solid；第二个图只添加了gradient，类型为sweep；第三个图只添加了stroke；第四个图添加了gradient和stroke两项特性。

以下为第四个图的代码：

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <shape xmlns:android="http://schemas.android.com/apk/res/android"
3     android:innerRadiusRatio="3"
4     android:shape="ring"

```

XHTML

[首页](#)[资讯](#)[文章](#) ▾[资源](#)[小组](#)[❤ 相亲](#)[频道](#) ▾[➔ 登录](#)[👤 注册](#)[?](#)

```
7      <gradient
8          android:endColor="#2F90BD"
9          android:startColor="#FFFFFF"
10         android:type="sweep" />
11      <stroke
12          android:width="1dp"
13          android:color="@android:color/black" />
14  </shape>
```

如果想让这个环形旋转起来，变成可用的进度条，则只要在shape外层包多一个rotate元素就可以了。

XHTML

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <rotate xmlns:android="http://schemas.android.com/apk/res/android"
3      android:fromDegrees="0"
4      android:pivotX="50%"
5      android:pivotY="50%"
6      android:toDegrees="1080.0">
7      <shape
8          android:innerRadiusRatio="3"
9          android:shape="ring"
10         android:thicknessRatio="8"
11         android:useLevel="false">
12          <gradient
13              android:endColor="#2F90BD"
14              android:startColor="#FFFFFF"
15              android:type="sweep" />
16      </shape>
17  </rotate>
```

结尾

shape篇就讲到这里了，代码也已经放上github，地址：

<https://github.com/keeganlee/kstyle.git>

[👍 1 赞](#)[🔖 11 收藏](#)[💬 评论](#)

相关文章

- [图文详解 Andorid 中用 Shape 定义 GradientDrawable](#)
- [Style 在 Android 中的继承关系](#)
- [Android样式的开发：Style篇](#)
- [Android样式的开发：Property Animation篇](#)
- [Android样式的开发：View Animation篇](#)

可能感兴趣的话题

- [女生应该放弃安逸的国企工作，去每天工作10... · 3](#)
- [大众点评2016校园招聘：二叉树遍历](#)
- [宜信2017 数据挖掘工程师：最小堆](#)
- [宜信2017 数据挖掘工程师：广度优先遍历 · 2](#)
- [怎么在业余时间继续保持一定量的前端开发？ · 1](#)
- [宜信2017数据挖掘工程师：快排 · 3](#)

登录后评论

新用户注册

直接登录



安卓小组话题

我有新话题 💬



[教你破解apk签名验证](#)
[起始页](#) 发起 • 5 回复



[请教 Android 牛, ContentProvider ...](#)
[lixyz](#) 发起



[ColorPicker一款android取色器, 高仿...](#)
[钉某人](#) 发起



[基于Tigase的android即时通讯开发](#)
[娇fightingforever](#) 发起 • 2 回复



[求解rem在安卓设备上的兼容问题](#)
[style](#) 发起 • 7 回复



[okhttp和xutils哪个好](#)
[蜗牛和玫瑰树会在一起吗](#) 发起 • 6 回复

- [本周热门安卓文章](#)
- [本月热门](#)
- [热门标签](#)



业界热点资讯

更多 »



[打赏抽成 30%，直播平台早有对策](#)
2 天前 · 👍 29 · 💬 1



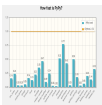
[物联网开发的主要编程语言有哪些？（IoT 开发的趋势调...](#)
1 天前 · 👍 8



[Firefox 54 为何默认开启四进程？Mozilla：不牺牲内存...](#)
15 小时前 · 👍 3



[美国国家安全局：朝鲜是 WannaCry 勒索病毒幕后黑手](#)
15 小时前 · 👍 3



[PyPy v5.8 发布，Python 解释器](#)
2 天前 · 👍 7



安卓工具资源

更多资源 »



IT及生活



MaterialProgressBar : 和Material Design效果外观一... Android, GUI · 1



Cordova : 跨平台前端编译库 Android, HTML、CSS与Javascript



android-remote-notifications : 从远程JSON文件拉取... Android, 通知



WilliamChart : 优美和直观的图表库 Android, 图表

关于安卓频道

安卓频道分享Android开发文章，精选工具和安卓相关的行业动态。

快速链接

- 网站使用指南 »
- 加入我们 »
- 问题反馈与求助 »
- 网站积分规则 »
- 网站声望规则 »

关注我们

新浪微博：[@安卓开发频道](#)
RSS：[订阅地址](#)
推荐微信号



安卓应用频道



ImportNew



UI设计达人

合作联系

Email：bd@jobbole.com

更多频道

- [小组](#) - 好的话题、有启发的回复、值得信赖的圈子
- [头条](#) - 分享和发现有价值的内容与观点
- [相亲](#) - 为IT单身男女服务的征婚传播平台
- [资源](#) - 优秀的工具资源导航
- [翻译](#) - 翻译传播优秀的外文文章
- [文章](#) - 国内外的精选文章
- [设计](#) - UI,网页,交互和用户体验
- [iOS](#) - 专注iOS技术分享
- [安卓](#) - 专注Android技术分享
- [前端](#) - JavaScript, HTML5, CSS
- [Java](#) - 专注Java技术分享
- [Python](#) - 专注Python技术分享

