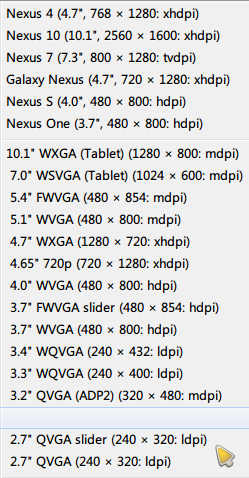
# 多设备分辨率的适配

## 前言

Android的设备分辨率千差万别，如下图：



多分辨率所引发的问题：

1. 显示异常

如果我们严格要求：针对于480\*800屏幕，view的显示原点为 距离屏幕左上角(x,y)，而在其他屏幕尺寸上，显示原点 要按该比例 类推

为了完成上述需求，我们想到了dip。但是dip最终也要转换为pixel，那么，你从一个孤岛，跳入了另外一个孤岛

也许我们想到了layout\_weight：由LinearLayout来替我们平分，但是，并非所有的布局都要使用LinearLayout

或者，我们可以使用居中，再一次，并非所有的view都适合居中，又或者，在实现的过程中，你用到了layout\_margin，那么，无可避免，你将使用dip，进而使用px

1. 和屏幕坐标相关的其他东东
2. 用户在屏幕上的touch坐标

如果该坐标落在某个区间内，将触发一个动画的播放

1. …

## 解决

我们放弃dip，采用了一种“原始”的方式：算得view在基准分辨率reso\_base下的dimen（左上角顶点坐标，长宽），然后，将该dimen映射到另外一个 分辨率reso\_x下

这种做法有什么缺点：

你必须知道你的layout中的每一个view，那么言外之意是：

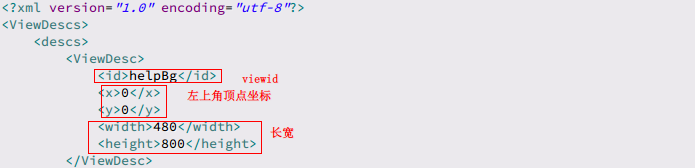
1. 你必须要知道你要映射哪些view的dimen（从reso\_base到reso\_x）
2. 如果你遇到了GridView, ListView, ViewPager等，你要停下来，注意一下：请设置这些ViewGroup的dimen，而不是其孩子的（该孩子的dimen通过代码控制）

### 基准分辨率reso\_base下的dimen

我们的基准分辨率，目前为：480\*800

你要做的是：

1. 使用任何你喜欢的layout，来布局你的view
2. 明确哪些view的dimen需要被映射
3. 算得那些view的dimen
4. 将该dimen放入到一个配置文件xxx\_base\_dimen.xml中，格式如下：



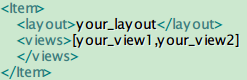
工具工程：talkingTomMuiltiScreen

它协助你来完成：在基准分辨率上，一个view的dimen

使用步骤：

1. 使用任何你想要用的layout，来进行布局。注意：请命名该layout为xxx\_base.xml
2. 在该layout中 额外加入 @layout/tool
3. 编写assets/configs/layout\_config.xml

在其中指明，哪个layout中的哪些view需要计算dimen，如：



设置：com.ql.talkingTomMultiScreen.DimenProvideActivity.TYPE 为 xxx

设置：com.ql.talkingTomMultiScreen.DimenProvideActivity.ROOT为 xxx\_base.xml中的根view的id

1. 运行程序，点击caculate按钮，注意观察logcat



1. 如上图所示，你也许会好奇moveTo是干嘛的：它是为了检验 你的RelativeLayout布局 是否构建正确

步骤如下：

1. 请将你的dimen配置放入到：assets/configs/layout/中
2. 构建以RelativeLayout为ViewGroup的布局，具体参见：章节：将dimen映射到另外一个分辨率reso\_x上
3. 点击moveTo，如果你看到的下一个页面的展示 和 之前的 相同，那么证明你的RelativeLayout的布局是正确的

### 将dimen映射到另外一个分辨率reso\_x上

你得到了基准分辨率reso\_base下的view的dimen，并将它们放入到了配置文件 xxx\_base\_dimen.xml

那么剩下的事情，就要明白的多：

1. 构建完全基于RelativeLayout的新的布局

在该布局中，你所用到的ViewGroup的类型，均为RelativeLayout

所有的view的layout\_width和layout\_height都是wrap\_content

如果你愿意，所有的layout\_gravity都是left|top

1. 通过映射计算，将dimen(left,top,width,height) 变为 dimen’(left’,top’,width’,height’)
2. 代码重设：layout\_marginLeft和layout\_marginTop，以指定view的左上角顶点位置
3. 代码重设：layout\_width和layout\_height，以指定view的长宽

以上就是大体思路，并且，我们的代码已经帮你做了：

API: xxx.context.MyContext.setLayout(Activity activity, String type)

如果你愿意，你可以查看setLayout的工作流，你也许会感兴趣：

xxx.context.MyContext.adjustViewDimen(Activity, String)

xxx.context.MyContext.adjustViewDimen(View, String)

一个例子来描述该API的内部执行：

如：我提供的type为study

1. 我将读取layout/study.xml布局文件
2. 我将读取assets/configs/layout/study\_base\_dimin.xml

以确定：有哪些view的dimen要重新映射

它们的基准dimen是什么

1. 我将会根据当前分辨率，和view的基准dimen，来计算新的dimen，并且，修改view的layout\_xxx