### [android中的dp,px深度解析](http://zhangkun716717-126-com.iteye.com/blog/1772696)

**博客分类：**

* [android](http://zhangkun716717-126-com.iteye.com/category/126650)

[android](http://www.iteye.com/blogs/tag/android)

　　dip: device independent pixels(设备独立像素)。不同设备有不同的显示效果,这个和设备硬件有关，一般我们为了支持WVGA、HVGA和QVGA 推荐使用这个，不依赖像素。   
　　与密度无关的像素，这是一个基于屏幕物理密度的抽象单位。密度可以理解为每英寸包含的像素个数（单位是dpi）,1dp实际上相当于密度为160dpi的屏上的一个点(可否理解为物理尺寸？)。也就是说，如果屏幕物理密度是160dpi时，dp和px是等效的。现在用实际的手机屏幕说一下。一块拥有320\*480分辨率的手机屏幕，如果宽度是2英寸，高度是3英寸，这块屏幕的密度就是160dpi。如果屏幕大小未变，而分辨率发生了变化，例如，分辨率由320\*480变成了480\*800，这时屏幕的物理密度就变大了（大于160dpi）。这就意味着屏幕每英寸可以显示更多的像素点，屏幕的显示效果就更细腻了。假设一个按钮的宽度使用dp作为单位，在160dpi时设为160，而在更高的dpi下（如320dpi)，按钮的宽度看上去和160dpi时的屏幕一样。这是由于系统在发现屏幕的密度不是160dpi时，会计算一个转换比例，然后用这个比例与实际尺寸相乘就得出新的尺寸。计算比例的方法是目标屏幕的密度除以160.如果目标屏幕的密度是320dpi,那么这个比例就是2。如果按钮的宽度是160dp，那么在320dpi的屏幕上的宽度就是320个像素点（dp是抽象单位，在实际的屏幕上应转换成像素点）。从这一点可以看出，dp可以自适应屏幕的密度。不管屏幕密度怎样变化，只要屏幕的物理尺寸不变，实际显示的尺寸就不会变化。如果将按钮的宽度设成160px，那么在320dpi的屏幕上仍然会是160个像素点，看上去按钮的宽度只是160dpi屏幕的一半。Android官方建议弃置表示宽度，高度，位置等属性时应尽量使用dp作为尺寸单位。   
　　据px = dip \* density / 160，则当屏幕密度为160时，px = dip。根据 google 的建议，TextView 的字号最好使用 sp 做单位，而且查看TextView的源码可知Android默认使用sp作为字号单位。将dip作为其他元素的单位。   
　　备注: 根据google的推荐，像素统一使用dip，字体统一使用sp    
　　举个例子区别px和dip：   
　　px就是像素，如果用px,就会用实际像素画，比个如吧，用画一条长度为240px的横线，在480宽的模拟器上看就是一半的屏宽，而在320宽的模拟器上看就是2／3的屏宽了。   
而dip，比如你做一条160dip的横线，无论你在320还480的模拟器上，都是一半屏的长度（当屏幕尺寸不变时）。

**Java代码  IMG_256**

1. **public** **static** **int** dip2px(Context context, **float** dipValue) {
2. **final** **float** scale = context.getResources().getDisplayMetrics().density;
3. **return** (**int**) (dipValue \* scale + 0.5f);
4. }
6. **public** **static** **int** px2dip(Context context, **float** pxValue) {
7. **final** **float** scale = context.getResources().getDisplayMetrics().density;
8. **return** (**int**) (pxValue / scale + 0.5f);
9. }

  下面这篇文章可以做深入理解之参考，[url＝http://blog.csdn.net/eggcalm/article/details/7006378]查看原文[/url]：   
   今天偶然间问了同事一个关于dp单位的问题，然后由这个问题引发的一连串的问题彻底颠覆了我关于dp的理论体系。   
  
   我那个问题是这样的：既然dp的本质是物理尺寸，为什么不用cm或者mm等传统长度单位替代？   
  
   然后他回答我dp是和像素密度无关的。。。我对这个回答不屑一顾，不过他接下来的一句话把我彻底震惊了，那句话是这样的：在你的手机上320dp就刚好满屏了，310dp就差一点点满屏。   
  
   我的手机是HTC Desire，这个理论我闻所未闻，然后马上做了个小实验，事实确实是这样，把一个TextView背景设成红色，宽度设成320dp，能看到满屏，310dp就差那么一点点。   
  
   看到这个测试结果的时候，我再一次崩溃了，我希望同事第二句话是一个美丽的错误，我无法接受这么久以来我理解的东西是错误的，可是事实是残酷的。   
  
   Android Developers关于dp的文档我看过N次，那个px和dp的转换公式我记得很清楚： px = dp \* (dpi / 160)，可是今天翻了源码了才发现，原来这里的dpi是归一化后的dpi，可能值只有120(low)、160(medium)、240(high)、 320(xhigh)四种，而我之前理解的竟然是实际设备真实的dpi！   
  
   G7的真实dpi是252，根据我以前的理解，310dp换算成px应该是：310 \* (252 / 160) = 488像素，而G7水平方向是480px，310dp在这上面绝对满屏都不止了，事实上Android系统并没有拿252作为dpi来计算，而是将G7视 作hdpi设备，然后使用240dpi来计算最终像素，所以在G7上320dp刚好是：320 \* (240 / 160) = 480像素，刚好满屏了，310dp确实要差一点点。   
  
  搞清楚这个问题后，我心里稍微好受点了，可是另一个问题接踵而来：   
dp的本质还是物理尺寸，难道不是吗？尽管Android系统对待dp这事上，将所有设备视为四种：120(low)、160(medium)、 240(high)、320(xhigh)，在160dpi上，160dp就是1英寸，在240dpi上，160dp还是1英寸，120dpi和 320dpi也还是1英寸，虽然他们占用的像素数不一样，但是最终显示出来的效果都是占用了屏幕上1英寸的范围。这套体系其实是非常合理的，一个宽为 160dp的按钮，它在所有设备上占用的物理尺寸应该是一样大才合理，这样他们对人眼所形成的张角才一样大，观看或者阅读的感觉才一致（姑且不考虑按钮的 背景是否一样细致）。这应该是Android系统引入dp概念的原因，因为手机屏幕的像素密度相差实在太大了，web那套东西已经完全不适用，你想电脑屏 幕的像素密度能相差多大？   
  
   终极问题来了，现实生活中真的只有以上四种不同像素密度的设备吗？不可能。虽然所有这些设备都可以粗略地划分为low、medium、high、 xhigh四种密度，可是对于划在同一范围内但具有不同像素密度的两个设备来说，同样的dp最终所占用的物理尺寸是不一样的。举个例子，G7(HTC Desire)的屏幕尺寸是3.7英寸，分辨率是480\*800，像素密度是252dpi，G10(HTC Desire HD)的屏幕尺寸是4.3英寸，分辨率同为480\*800，像素密度是217dpi。假设现在有一个按钮，它的宽度设为100dp，由于G7和G10同被 划分为hdpi，那么在这两个设备上，这个按钮的实际宽度是：100 \* (240 / 160) = 150像素，可由于这两个设备的实际像素密度是不一样的，所以真实的显示效果是：这个按钮在两个设备上的实际物理尺寸是不一样大的。而这，和 Android进入dp的概念是相悖的。   
  
   你可以说对于同属于hdpi的设备而言，这个差别很小，但是在ldpi和hdpi之间这个差别就很明显了。我非常认同，可是如果在将dp转化为px的时 候，不是使用归一化dpi（也就是120(low)、160(medium)、240(high)、320(xhigh)这四种），而是使用设备真实的像 素密度，那么得出的像素数目虽然各不一样，但是最终显示出来的物理尺寸确实一样大的，而这种计算方法，我认为是忠于像素密度无关的理论的。   
  
   最后我还是想说，如果Android希望一个宽度为160dp的按钮在任何设备上都是1英寸大，那为什么不直接使用英寸作为度量单位呢？如果你有好的想法，欢迎留言。   
  
UPDATE:   
  
   今天下午在回答factar网友的问题的时候，我上面那个“终极问题”终于找到了一个合理的答案。在factar贴的网址里，我发现一句重要的话：   
  
“However, bitmap scaling can result in blurry or pixelated bitmaps, which you might notice in the above screenshots.”   
  
   这句话的意思是说，图片资源在缩放的时候会造成图像模糊。按照我以上的分析，如果为了保证相同的图片资源在不同像素密度的设备上保持完全一样的尺寸 大小（这完全可以做到，在dp转化成px的时候使用设备的物理像素密度参数），那图片在不同设备上的缩放因子必然不一样，而这会导致图像模糊！所以我猜想 Google为了保证了图像不会模糊退了一步，让相同dp在不同设备上“差不多一样大”。   
  
   还有，这个答案也纠正了我的一个误区，现在有很多应用程序开发商为了降低安装包的大小，只使用一套hdpi资源或者一套xhdpi资源，而不提供 mdpi资源或ldpi资源，希望在mdpi和ldpi设备上有系统完成缩放适应，虽然可行，但是我们不应忽视因为缩放带来的图像模糊、显示效果不佳的现象。   
  
  
   网友问答参考：   
   不知eggclam现在是否对dp有了更深入的理解，我现在对dp这里也陷入到了这步，我现在有三个pad，一个10寸，2个7寸，   
   参数如下：   
A：7寸pad 1，   
denstiy：1.0 分辨率 1024X600   
B: 7寸pad 2，   
denstiy： 1.33 分辨率 1280X800   
C: 7寸pad 3，   
denstiy：1.0 分辨率 1280X800   
  
   按照google 官方文档称（http://developer.android.com/guide/practices/screens\_support.html）   
Density independence 段落，用dp在不同的denstiy 下，大小看起来应该是一样的。   
我在三个pad 都设置了同样dp长度的按钮，但是在3个pad上的长度看着都不一样，不知道是因为什么呢？能不能加你好友讨论下呢？   
  
   引用“factar”的评论：不知eggclam现在是否对dp有了更深入的理解，我现在对dp这里也陷入到了这步，我现在有三个pad...   
  
C 设备应该是10寸吧？   
  
如果C 设备是10寸，那么这三款设备的物理dpi大致如下：   
A：169.5   
B：215.6   
C：150.9   
  
根据这里的划分（http://developer.android.com/guide/practices/screens\_support.html#range），A和C被评价为mdpi（density=1.0），B本来应该被评价为hdpi（density=1.5），但是自从API LEVEL 13开始Android引入了DENSITY\_TV（dpi=213），而且你贴的数据也的确证明这一点，所以B设备的density沿用你的数据1.33   
  
接下来我们算算一根50dp的线条在这三个设备上显示成多少像素：   
A：50 \* 1.0 = 50(px)   
B：50 \* 1.33 = 67(px)   
C：50 \* 1.0 = 50(px)   
  
接下来的问题就简单了，问题直接转化成这三个像素值在ABC上占用的物理尺寸一致吗？我们可以算算：   
A：50 / 169.5 = 0.2950(英寸) = 0.7493(厘米)   
B：67 / 215.6 = 0.3108(英寸) = 0.7894(厘米)   
C：50 / 150.9 = 0.3313(英寸) = 0.8418(厘米)   
  
根据以上结果，你看到的不完全一样大是合乎情理的，因为在Android中，相同dp在所有mdpi设备上虽然像素数量是一样的，但是因为各个设备物理dpi不一样，所以在最终的显示尺寸上是有微弱差别的。