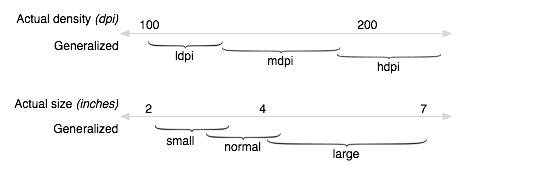
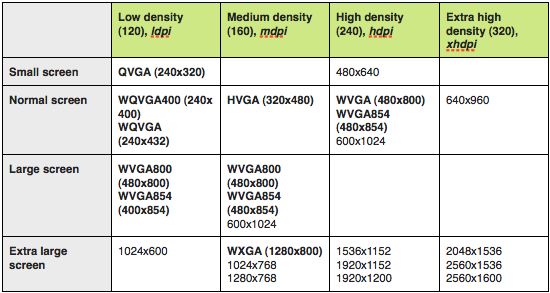
[drawable文件夹详解](http://blog.csdn.net/persuit/article/details/7663574)

2012-06-14 16:25 8895人阅读 [评论](http://blog.csdn.net/persuit/article/details/7663574#comments)(1) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/persuit/article/details/7663574#report)

[android](http://www.csdn.net/tag/android)[verizon](http://www.csdn.net/tag/verizon)[keyboard](http://www.csdn.net/tag/keyboard)[手机](http://www.csdn.net/tag/%e6%89%8b%e6%9c%ba)[input](http://www.csdn.net/tag/input)[bt](http://www.csdn.net/tag/bt)

QVGA使用ldpi，虽然有不同尺寸，但都是120dpi左右；HVGA同理；如下图；





-finger    用于触摸屏的设备

-hdpi    近似于240dpi的高级显示密度的屏幕

-mdpi    近似于160dpi的中级显示密度的屏幕

-ldpi    近似于120dpi的低级显示密度的屏幕

-land    横屏显示

-port    竖屏显示

-long    比较长的屏幕，例如WQVGA(432×240), WVGA(800×480), FWVGA(854×480)

图片缩放比列为 3 ：4：6：8   ldpi，mdpi，hdip，xhdip

具体计算方法为：

  ( (size \* tdensity) + (sdensity >> 1) ) / sdensity;

size：长/宽  
sdensity：原图密度；  
tdensity：终端密度；

drawable-en:英文设置下的图片资料，drawable-zh中文设置下的图片资料。  
若是要为不合像素的手机筹办的话，须要（以drawable-en为例申明）  
:drawable-en-hdpi  
:drawable-en-ldpi  
:drawable-en-mdpi  
同理：为美式英语，  
:drawable-en-rUS-hdpi  
:drawable-en-rUS-ldpi  
:drawable-en-rUS-mdpi  
再错杂一点，为不合标的目标的筹办图片资料  
//竖直  
:drawable-en-rUS-port-hdpi  
:drawable-en-rUS-port-ldpi  
:drawable-en-rUS-port-mdpi  
//横屏  
:drawable-en-rUS-land-hdpi  
:drawable-en-rUS-land-ldpi  
:drawable-en-rUS-land-mdpi

建树这些文件夹是有次序的。即按优先级别分列的，见下表。所以建树文件夹时一般从左到右的分列其优先级别如：drawable-en-rUS-land-mdpi  
如：drawable-en-rUS-port-160dpi-finger-qwerty-dpad-480 x320/

|  |  |
| --- | --- |
| **限制符Qualifier** | **&＃20540;Values** |
| 移动国度码MCC和移动收集码MNC | 手机设备SIM卡上的移动国度码和移动收集码。比如mcc310-mnc004 （美国，Verizon品牌）; mcc208-mnc00 （法国，Orange品牌）; mcc234-mnc00 （英国，BT品牌）.  若是这个设备应用一个无线连接（GSM德律风），则MCC来自SIM卡，而MNC来自该设备将要附着的收集。你有时会仅应用MCC，例如包含特定国度合法资料在您的应用法度中。若是您的应用法度指定了MCC/MNC组合的资料，这些资料仅在MCC和MNC都匹配的时辰才干应用。 |
| 说话和区域Languageand region | 两个字母的ISO639-1说话码和ISO3166-1-alpha-2区域码 （以"r"为前缀）。比如en-rUS，fr-rFR，es-rES.这个代码是大小写敏感的：说话码是小写字母，国度码是大写字母。你不克不及零丁指定一个区域，然则你可以零丁指定一个说话，比如en，fr， es， zh. |
| 屏幕标的目标Screenorientation | 纵向，横向，正方形（port，land， square） |
| 屏幕像素密度Screenpixel density | 92dpi，108dpi等. 当Android选择应用哪个资料时，它对屏幕像素密度的处理惩罚和其它限制符不合。在文章后面描述的步调1Android如何查找最匹配的目次中，屏幕密度总被认为是匹配的。在步调4中，若是被推敲的限制符是屏幕密度，Android将选择在那个地位的最佳匹配，而无需持续步调5。 |
| 触摸屏类型Touchscreentype | 非触摸式，触摸笔，手指（notouch，stylus， finger） |
| 键盘可用体式格式Whetherthe keyboard is available to the user | 外在键盘，隐躲键盘，软键盘（keysexposed，keyshidden，keyssoft） 若是你的应用法度有一个特定的资料只能经由过程软件盘应用，则应用keyssoft &＃20540;，若是没有keyssoft 资料可用（只有keysexposed 和 keyshidden）并且该设备显示了一个软键盘，那么体系将应用keysexposed 资料。 |
| 首选文本输进办法Primarytext input method | 不支撑按键，标准键盘，12键（nokeys，qwerty， 12key） |
| 首选非触摸式导航办法Primarynon-touchscreen navigation method | 不支撑导航，滑板，跟踪球，滚轮（nonav，dpad， trackball， wheel） |
| 屏幕辨别率Screendimensions | 320 x240，640 x480， 等. 更大的辨别率必须先被指定。 |
| SDK版本SDKversion | 设备支撑的SDK版本，比如v3。Android1.0SDK是v1，1.1SDK是v2，1.5SDK是v3。 |
| 小版本（Minorversion） | 你今朝还不克不及指定小版本，它老是被设置为0。 |

http://www.2cto.com/kf/201204/127339.html

表2.配置限定符名称

|  |  |  |
| --- | --- | --- |
| 配置 | 限定符值 | 说明 |
| MCC和MNC | 例如： mcc310 mcc310-mnc004 mcc208-mnc00 等 | MCC是移动国家代码的英文首字母缩写（The mobile country code），它的后面可选择性的跟随来自设备内的SIM卡的移动网络代码（MNC：mobile network code）。如在任何载体上，mcc310代表美国，mcc310-mnc004代表美国的Venizon公司，mcc208-mnc00代表法国的Orange公司。 如果设备使用音频连接（GSM 电话），那么MCC和MNC的值来自SIM卡。 也可以单独使用MCC（例如，在应用程序中包含特殊国家合法的资源）。如果仅需要指定语言环境，那么可以使用language和region限定符来替代（稍后讨论）。如果决定要使用MCC和MNC限定符，就要仔细测试，使它能够满足你所期望的工作。 还可以查看配置域mcc和mnc，它们分别指示了当前的移动国家代码和移动网络代码。 mcc：<http://developer.android.com/reference/android/content/res/Configuration.html#mcc> mnc：<http://developer.android.com/reference/android/content/res/Configuration.html#mnc> |
| 语言和地区 | 例如： en fr en-rUS fr-rFR fr-rCA | 语言是用两个字母的ISO 639-1语言代码定义的，紧跟其后的是可选的两个ISO-3166-1-appha-2地区代码字母（前面是小写的“r”）。 这个编码不区分大小写，r前缀被用于区分地区部分，不能够单独指定地区。 如果用户改变了系统中的语言设置，那么在应用程序的运行期间也能够改变为对应的语言。 |
| 最小宽度 | sw<N>dp 例如： sw320dp sw600dp sw720dp 等 | 屏幕的基本尺寸，是指最短的可用屏幕区域。具体的说，设备的最小宽度是屏幕可用的宽度和高度中最短的那个（也可以把它看做是屏幕的最小可能的宽度）。这样就可以使用这个限定符来确保应用程序至少有<N>dp的宽度可用于UI界面，而不管屏幕的当前方向。 例如，如果布局在任何时候都需要至少600dp的最小屏幕尺寸，那么就能够使用这个限定符，在res/layout-sw600dp/目录中创建布局资源。系统只会在可用屏幕的尺寸至少是600dp的时候才会使用这些资源，而不管600dp是否是被用户认知的高度或宽度。最小宽度是设备的固定屏幕尺寸特征，当屏幕的方向发生改变时，设备的最小宽度不改变。 设备的最小宽度需要考虑屏幕的装饰和系统UI的占用。例如，如果设备有一些固定的UI元素要沿着最小宽度的轴向，占用一定的屏幕空间，那么系统声明的最小宽度要比实际的屏幕尺寸要小，因为被系统占用的像素部分对用户应用程序的UI无效。因此，这个值应该是应用程序布局所需要的最小的实际尺寸（通常，这个值是布局支持的最小宽度，而不管屏幕的当前方向）。 以下是可以使用的通用屏幕尺寸的一些值： 1.320，针对以下屏幕配置的设备：   240x320ldpi（QVGA手持设备）   320x480mdpi（手持设备）   480x800hdpi（高分辨率手持设备） 2.480，针对480x800mdpi的屏幕（平板或手持设备） 3.600，针对600x1024mdip的屏幕（7英寸平板） 4.720，针对720x1280mdip的屏幕（10英寸平板） 当应用程序提供了多个带有不同值的最小宽度限定符资源目录时，系统会使用最接近（不超出）设备最小宽度的那个资源。 这个限定符被添加在API级别13中。 还要看android:requiresSmallestWidthDp属性，它声明了与你的应用程序兼容的最小的最小宽度，并且smallestScreenWidthDp配置字段会持有这个设备最小宽度的值。 |
| 可用宽度 | w<N>dp 例如： w720dp w1024dp 等 | 指定最小的可用屏幕宽度，在资源中应该以dp为单位来定义<N>的值。当方向在横向和纵向之间改变时，这个配置值会跟当前的实际的宽度相匹配。 当应用程序给这个配置提供了多个不同值的资源目录时，系统会使用最接近（不超过）设备当前屏幕宽度的那个配置。这个值需要考虑屏幕装饰占据的空间，因此，如果设备在显示的左边或右边有一些固定的UI元素，那么使用的宽度值就要比实际的屏幕尺寸小，因为这些固定UI元素的占用，使得应用程序的可用空间减少。 这个特性被添加在API级别13中 还要看screenWidthDp配置字段，它持有当前的屏幕宽度。 |
| 可用高度 | h<N>dp 例如： h720dp h1024dp 等 | 指定最小的可用屏幕高度，在资源中应该以dp为单位来定义<N>的值，当方向在横向和纵向直接改变时，这个配置值应该跟当前的实际高度匹配。 当应用程序给这个配置提供了不同值的多个资源目录时，系统会使用最接近（不超过）设备当前屏幕高度的那个配置。这个要考虑屏幕装饰的占用情况，因此，如果设备在显示的上方或底部有一些固定的UI元素，那么要使用的高度值要比实际的屏幕尺寸小，因为这些固定UI元素的占用，使得应用程序的可用空间减少。不固定的屏幕装饰（如电话的状态栏能够在全屏时被隐藏）是不考虑的，像标题栏或操作栏这样的窗口装饰也不考虑，因此应用必须准备处理比它们指定的空间要小的情况。 这个限定符被添加在API级别13中。 还要看screenHeightDp配置字段，它持有当前屏幕的高度。 |
| 屏幕尺寸 | small normal large xlarge | small：这种屏类似低分辨率的QVGA屏幕。对于小屏的最小布局尺寸大约是320x426dp。例如QVGA低分辨率和VGA高分辨率。 normal：这种屏类似中等分辨率的HVGA屏幕。对于普通屏幕的最小布局尺寸大约是320x470dp。如，WQVGA低分辨率屏、HVGA中等分辨率屏、WVGA高分辨率屏。 large：这种屏类似中等分辨率的VGA屏幕，对于大屏幕的最小布局尺寸大约是480x640dp。例如VGA和WVGA的中等分辨率屏。 xlarge：这种屏被认为比传统的中等分辨率的HVGA屏幕大。针对xlarge屏的最小布局尺寸大约是720x960dp。在大多数情况下，这种超大屏幕的设备因为太大而要放到背包中来携带，而且最有可能的是平板样式的设备。 注意：使用尺寸限定符不意味着资源仅用于这个尺寸的屏幕。如果没有用限定符提供与当前设备配置相匹配的可选资源，那么系统会使用与配置最接近的资源。 警告：如果所有使用尺寸限定符的资源都比当前屏幕大，那么系统将不会使用它们，并且应用程序会在运行时崩溃（例如，如果所有的布局都被标记了xlarge限定符，而设备却是一个普通尺寸的屏幕）。 这个限定符被添加在API级别4以后的版本中。 |
| 屏幕外观 | long notlong | long：长屏幕，如WQVGA、WVGA、FWVGA notlong：非长屏幕，如QVGA、HVGA、VGA 这个限定符被添加在API级别4以后的版本中 这个限定符完全是基于屏幕的外观比率，不相对屏幕的方向。 还要看screenLayout配置字段，它指示了屏幕是否是长屏。 |
| 屏幕方向 | port land | port：纵向设备（垂直） land：横向设备（水平） 如果用户旋转屏幕，这个限定能够在应用程序运行期间改变。 orientation配置字段指示当前设备的方向。 |
| 泊位模式 | car desk | car：设备停靠在汽车中 desk：设备停靠在书桌中 这个限定符被添加在API级别8以后的版本中 如果用户改变了设备的停靠地点，那么能够在应用程序的运行期间改变这个限定。可以使用UiModeManager对象来启用或禁止这种模式。 |
| 夜间模式 | night notnight | night：夜间 notnight：白天 被添加在API级别8以后的版本中 如果夜间模式被保留在自动模式中（默认），那么在应用程序运行期间，会基于白天的时间来进行模式的改变。可以使用UiModeManager对象来启用或禁止这种模式。 |
| 屏幕像素密度（dpi） | ldpi mdpi hdpi xhdpi nodpi tvdpi | ldpi：针对大约120dpi的低分辨率屏幕； mdpi：针对大约160dpi的中等分辨率屏幕（在传统的HVGA上）； hdpi：针对大约240dpi的高分辨率屏幕； xhdpi：针对大约320dpi的超高分辨率屏幕，被添加在API基本8以后的版本中； nodpi：这个限定被用于不想根据匹配的设备分辨率进行缩放的位图资源。 tvdpi：在mdpi和hdpi之间的屏幕，大约是213dpi。这种分组不是主要的分辨率，大多数是为电视来考虑的，并且大多数应用不需要它---提供mdpi和hdpi资源就可以满足大多数应用程序需要了，并且系统会适当的缩放它们。这个限定符在API级别13以后被引入。 四种主要的分辨率之间的缩放比例是：3:4:6:8（忽略tvdpi分辨率），因此一个9x9的ldpi位图，在mdpi中是12x12、在hdpi中是18x18、在xhdpi中是24x24。 如果感觉在电视或其他某些设备上的图片资源不好看，并且想要试用tvdpi资源，那么缩放因子是1.33\*mdpi。例如，一个100px x 100px的mdpi图片的图片应该被放大成133px x 133px的tvdpi图片。 注意：使用分辨率限定符不意味着资源仅适用与对应分辨率的屏幕。如果没有提供与当前设备配置匹配的可选资源，那么系统会使用最接近的资源。 |
| 触屏类型 | notouch stylus finger | notouch：非触屏设备 stylus：有适用手写笔的电阻屏设备 finger：触屏设备 touchscreen配置字段，指示到了设备上的触屏类型。 |
| 键盘可用性 | keysexposed keyshidden keyssoft | keysexposed：设备有可用的键盘。如果设备启用了软键盘，那么即使在硬键盘没有暴露给用户时也可以使用这个限定符。如果没有提供软键盘或者软键盘被禁用，那么只有在硬键盘被暴露给用户时才能够使用这个限定符。 keyshidden：设备有可用的硬键盘，但是被隐藏了，并且设备没有可用的软键盘。 keyssoft：设备有可用的软键盘，不管它是否可见。 如果提供了keysexposed资源，但没有keyssoft资源，那么只要系统有可用的软键盘，系统就会使用keysexposed资源而不管键盘是否可见。 如果用户打开了硬键盘，就可以在应用程序运行期间改变这个限定。 hardKeyboardHidden和keyboardHidden配置字段分别指明硬键盘的可见性以及可见的键盘类型（包括软键盘）。 |
| 主要文本输入法 | nokeys qwerty l2key | nokeys：设备没有用于文本输入的硬键盘； qwerty：设备有标准的硬键盘，不管用户是否可见； 12key：设备有12个键的硬键盘，不管用户是否可见。 keyboard配置字段指明可用的主要文本输入方法。 |
| 导航键的有效性 | navexposed navhidden | navexposed：导航键对用户可用； navhidden：导航键不可用。 如果用户能够看到导航键，那么在应用程序运行时就能够改变这个限定。 navigationHidden配置字段，指示导航键是否隐藏。 |
| 主要的非触屏导航方法 | nonav dpad trackball wheel | nonav：除了使用触屏以外，设备没有其他导航设施。 dpad：设备有用于导航的定向板（d-pad）。 trackball：设备有用于导航的轨迹球。 wheel：设备有用于导航的定向滚轮（不常见）。 navigation配置字段指明可用的导航方法类型。 |
| 平台版本（API 级别） | 例如： v3 v4 v7 等 | 设备支持的API级别。如v1代表API级别1（带有Android1.0或更高版的设备），v4代表API级别4（带有Android1.6或更高版本的设备） 警告：Android1.5和1.6只有在限定符跟平台版本完全匹配时，才能匹配资源 |