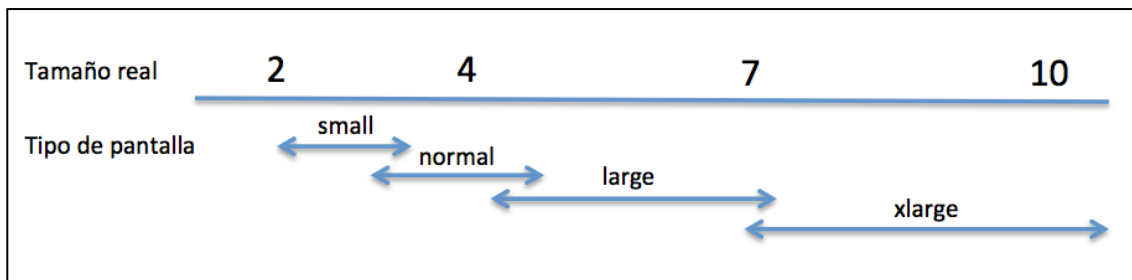


## 15. Tamaños de pantalla y unidades de medida

Una pantalla se caracteriza por su tamaño y su densidad. Android reconoce cuatro **tamaños**, medidos como el número de pulgadas de la diagonal:

- small: de 2 a 3 pulgadas
- normal: de 3 a 4,5 pulgadas
- large: de 4,5 a 7 pulgadas
- xlarge: de 7 a 10 pulgadas



La **densidad** de la pantalla es el número de píxeles por unidad de área (dpi o dots per inch, es decir, puntos por pulgada). Android reconoce cuatro densidades de pantalla (siguen aproximadamente las proporciones 3:4:6:8):

- baja (low density - ldpi): 120 dpi.
- media (medium density - mdpi): 160 dpi.
- alta (high density - hdpi): 240 dpi.
- extra alta (extra high density - xhdpi): 320 dpi.

Por ejemplo, el LG Optimus L5 E610 tiene una pantalla de 4 pulgadas y 320 x 480 píxeles. Podemos calcular su densidad como el número de píxeles por unidad de longitud de su diagonal:

$$\frac{\sqrt{320^2 + 480^2}}{4} = 144$$

Se trata, por lo tanto, de un dispositivo de densidad media.

Las siguientes unidades de medida se pueden utilizar para especificar el tamaño de los elementos de una interfaz Android (los dos primeros son los recomendados):

- **dp** o **dip** (*density independent pixel*). 1 **dip** es equivalente a 1 pixel en una pantalla de densidad 160 dpi (dots per inch). Es la unidad recomendada para dimensiones de elementos en ficheros de diseño como, por ejemplo, `layout_margin`, `padding`, etc.
- **sp** (*scale independent pixel*). Son píxeles independientes de densidad que tienen en cuenta el tamaño del font. Es la unidad recomendada para tamaños de texto.
- **pt** (*point*). Un punto (*point*) es 1/72 de una pulgada (*inch*) de una pantalla física.

- $px$  (*pixel*). Esta unidad se corresponde con los píxeles de una pantalla.

Como los  $dips$  son independientes de densidad y 1  $dip$  equivale a 1 píxel en una pantalla de densidad media (160 dpi), 1  $dip$  debería equivaler a 2 píxeles en una pantalla con el doble de densidad (320 dpi). En general, esta es la fórmula que convierte  $dips$  en píxeles:

$$px = dp \frac{dpi}{160}$$

donde  $dpi$  es la densidad de pantalla del dispositivo.

Para crear interfaces de usuario en Android debemos especificar distintos ficheros de diseño para cada tamaño de pantalla (utilizando recursos alternativos) y bitmaps específicos para cada densidad.

Los calificadores para especificar recursos con tamaños específicos son: `small`, `normal`, `large` y `xlarge`. Por ejemplo, el fichero de recurso de nombre `/res/layout-small/activity_main.xml` será seleccionado por Android en dispositivos con pantallas pequeñas. A partir del API 13, en lugar de estos calificadores, se recomienda utilizar el calificador `sw<N>dp` (anchura mínima). Por ejemplo, el recurso `/res/layout-sw600dp/activity_main.xml` se utilizará en dispositivos con una anchura de pantalla, cualquiera que sea la orientación del dispositivo, de al menos 600  $dps$ . Esta es la forma más sencilla de especificar recursos alternativos independientemente de la orientación del dispositivo. Los siguientes ejemplos pueden servir de orientación:

- 320  $dp$ : pantalla de un teléfono típico.
- 600  $dp$ : pantalla de una tableta de 7".
- 720  $dp$ : pantalla de una tableta de 10".

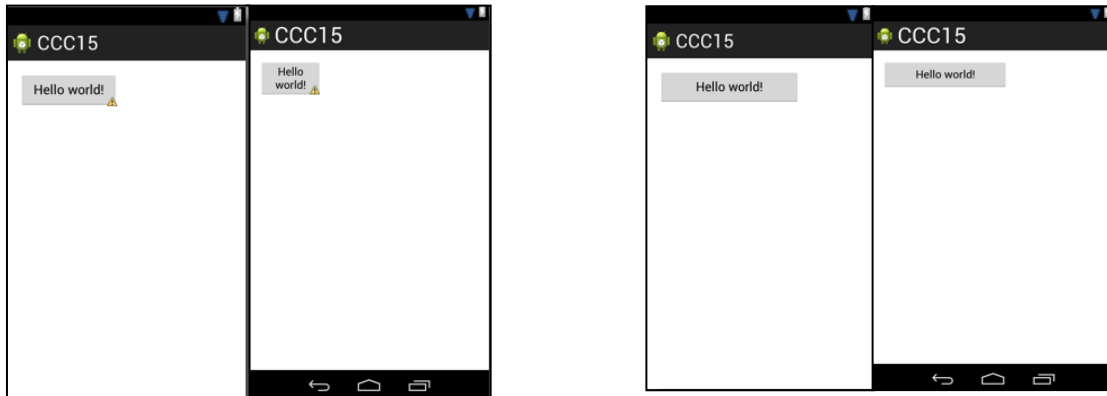
Es fundamental expresar las dimensiones de los elementos de la interfaz en  $dips$  en lugar de  $px$ . Para entender mejor esta cuestión, vamos a simular dos dispositivos de tamaño normal con densidades diferentes: Nexus 4 ( $xhdpi$ ) y Nexus S ( $hdpi$ ). El fichero de diseño contiene simplemente un botón cuya anchura especificamos primero en píxeles ( $px$ ) y luego en píxeles independientes de densidad ( $dip$ ):

`/res/layout/activity_main.xml`

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity">
    <Button
        android:layout_width="200px"
        android:layout_height="wrap_content"
        android:text="@string/hello_world"/>
</RelativeLayout>
```

En el grupo de la izquierda, con la anchura del botón expresada en píxeles, el tamaño del botón disminuye al aumentar la densidad (de izquierda a derecha).

En el grupo de la derecha, este efecto se compensa al especificar la dimensión en píxeles independientes de densidad:



En el caso de bitmaps no es suficiente con dar su tamaño en dips, ya que un dip se traduce en un número de píxeles distinto para cada densidad de pantalla. Como las imágenes se definen en píxeles, esto puede dar lugar a imágenes borrosas al contraer o ampliar. Para evitarlo deberíamos añadir una versión de cada imagen en distintas densidades.

Por ejemplo, si tenemos una imagen de 100x100 píxeles que se ve correctamente para una densidad media (mdpi), teniendo en cuenta que las densidades siguen aproximadamente las proporciones 3:4:6:8, deberíamos añadir las siguientes versiones del bitmap:

- bitmap de 75 x 75 píxeles en la carpeta `/res/drawable-ldpi`
- bitmap de 150 x 150 píxeles en la carpeta `/res/drawable-hdpi`
- bitamp de 200 x 200 píxeles en la carpeta `/res/drawable-xhdpi`