



Animaciones Android

Descripción de animaciones

La descripción de una animación se puede hacer en Java o también en un fichero XML, como lo he hecho. Estos ficheros XML deben estar en la carpeta *res/anim*. Cada animación debe estar en un fichero diferente.

Las cuatro animaciones básicas que existen son:

- <translate />* para mover un objeto
- <rotate />* para hacer rotaciones
- <scale />* para cambiar el tamaño
- <alpha />* para jugar con su transparencia

Cada uno tiene propiedades (que empiezan con “*android:*”) que permiten describir la animación:

- *fromX* y *toX* para definir los valores de la propiedad al principio y a la fin de la animación
- *duration* para la duración y *startOffset* para empezar la animación con este valor de milisegundos de retraso.
- *pivotX* y *pivotY* para definir el centro de transformaciones (útil para rotaciones y cambio de tamaño)

Un ejemplo simple de animación se puede ver en *res/anim/move.xml*

Si quieres hacer la animación varias veces, utiliza la propiedad *repeatCount* con el número de veces que quieres repetirla. Hay dos maneras de hacer la repetición que puedes cambiar con la propiedad *repeatMode* que puede estar *reverse* o *restart* (por defecto)

Para hacer que la animación no esté lineal, puedes utilizar la propiedad *interpolator* con una de las siguientes valores:

- *accelerate* : lento después rápido
- *decelerate* : rápido después lento
- *accelerate_decelerate* : rápido a la mitad
- *anticipate* : hace la animación en inverso antes de hacerla normalmente
- *bounce* : rebota del objeto
- *overshoot* : va un poco más lejos que el valor de fin y vuelve
- y combinaciones...

Un ejemplo de utilización de interpolator se puede ver en el fichero *res/anim/rotate.xml*

Para cambiar varias propiedades al mismo tiempo, puedes utilizar un *set* dentro del cual puedes poner las animaciones. Este elemento permite también hacer varias animaciones de la misma propiedad una

después de la otra, jugando con las propiedades *startOffset* de cada animación. Por ejemplo el fichero *res/anim/scale.xml* hace animaciones de tamaño: para cada animación, hay que poner su *startOffset* al valor del *startOffset* de la animación de antes más su valor de *duration*. Cuidado con los valores de *fromPropiedad* y *toPropiedad* que están siempre relativos : por ejemplo la segunda animación “*scale*” tiene valor de principio *1.0*, pero en realidad es *1.0* por la valor de fin de la animación de antes. Entonces si quieres una animación fluida debe siempre poner una valor de *1.0* al principio de las siguientes animaciones.

Lanzar animaciones

El lanzamiento de animaciones se hace en Java; en mi ejemplo lo he hecho en “*MainActivity.java*”. Hay simplemente que cargar la animación y lanzarla en un “widget” después.

```
Animation A = AnimationUtils.LoadAnimation(context, id);  
miWidget.startAnimation(A);
```

Donde *context* e *id* son *getContext()* y *R.anim.nombreDeMiAnimacio*

En mi ejemplo, he puesto *TouchListener* en botones que lanzan la función *miAnimacion(numeroAnim)* que lanza una animación sobre este mismo botón.

Animaciones de Layout

Lo que hemos visto antes es para hacer animaciones de widget. Si queréis hacer animación sobre un Layout entero, no se puede hacer de la misma manera. Este otro tipo de animación permite hacer animación sobre cada uno de los elementos contenido en el Layout.

Primero, debes crear dos ficheros XML: un exactamente como hemos visto antes, con el contenido real de la animación a hacer sobre cada elemento, y un otro para definir como la animación se difundir entre los elementos. Se define en un elemento XML “*layoutAnimation*” con esas propiedades:

```
android:delay="10%" : tiempo de retraso entre cada elemento del Layout  
android:animationOrder="normal" : orden de difundo de la animación entre los elementos  
android:animation="@anim/layoutanimcontenido": enlace sobre el contenido de la animación
```

Los ficheros de ejemplo son *layoutanimcontenido.xml* y *layoutanim.xml*.

Para lanzar la animación, es también un poco diferente de lo que debemos hacer para los widgets:

```
LayoutAnimationControlller A = AnimationUtils.LoadLayoutAnimation(context, id);  
miLayout.setLayoutAnimation(A);
```

La función *setLayoutAnimation()* permite normalmente lanzar la animación de layout, sin embargo no funciona cuando yo lo hago en mi ordenador. La única manera que he encontrado para lanzarla es de añadir las líneas:

```
L.setVisibility(0);  
L.setVisibility(1);
```

Código

Puedes encontrar mi código de ejemplo en este enlace github :
<http://github.com/PaulDesiles/AnimacionesAndroid/>

Bibliografía

OpenClassrooms.com

/informatique/cours/creez-des-applications-pour-android

(Tutorial de Frederic Espiau)

developer.android.com

/guide/topics/graphics/prop-animation.html

/training/animation/crossfade.html