

Animatable Panel

Julian Loscombe and Ben Milne

April 27, 2015

1 Overview

AnimatablePanel is a class to assist with animation when dealing with panels. It adds a number of animatable properties to the standard JPanel class and provides a simpler mechanism for animating more arbitrary values, for example, when drawing in `paintComponent()`.

2 Panel animation

Currently AnimatablePanel allows you to animate the preferred size of a panel and its background color including the alpha component. The appropriate methods are respectfully:

```
setPreferredSize(Dimension size, Double duration)
setBackground(Color background, Double duration)
```

3 Animator

AnimatablePanel uses a helper class, Animator to produce the desired animation. This acts on a single Double value and increments it between an initial value and a target value over a duration. You can create extra Animator objects from outside the AnimatablePanel class to animate arbitrary values by calling `createAnimator()`. For example, if we wanted to make a circle which is drawn in `paintComponent()` expand when we call a method we would do something like:

```
private class CustomPanel extends AnimatablePanel {
    AnimatablePanel.Animator radiusAnimator;

    public void growCircle() {
        //Create a new Animator with specified values
        radiusAnimator = createAnimator(0.0, 20.0, 0.5);
    }

    ...

    public void paintComponent(Graphics g0) {
        Double r;
        ...
        //Draw using the animated value
        if (radiusAnimator != null) r = radiusAnimator.value();
        g.fillOval(x - r, y - r, r * 2, r * 2);
    }
}
```

4 Easing

You can change the easing of an animation by specifying an argument of type `AnimationEase` in the animation call.

```
setPreferredSize(Dimension size, Double duration, AnimationEase ease)
setBackground(Color background, Double duration, AnimationEase ease)
```

This is an enumeration with values:

```
AnimationEase.LINEAR
AnimationEase.EASE_IN
AnimationEase.EASE_OUT
AnimationEase.EASE_IN_OUT
```

If no easing is set then the default is `LINEAR`. All easings are cubic and you can find nice examples of what each means at www.easings.net.

5 Notifications

It can be useful to know when an animation has begun or finished and so `Animatable` panel contains two methods that you can override to be notified when such events occur. These are:

```
animationsBegun()
animationsCompleted()
```