

Forum:[Hack Forums](#)

Thread: [Animation Library - Easily animate your application](#)

BlackCap - October 10, 2014, 8:18 pm

Source: <http://pastebin.com/WBqqJJqd>

Usage

First of all you need a method that takes at least one double between 0 and 1, and does something with it; move a control for instance:

PHP Code

```
<?php
```

```
Sub MoveCtrl(ctrl As Control, x As Double, y As Double)
```

```
    ctrl.Location = New Point(x * ctrl.Parent.Width / 2, y * ctrl.Parent.Height / 2)
```

```
End Sub
```

you could also use lambda

PHP Code

```
<?php
```

```
Dim MoveCtrl = sub(ctrl As Control, x
```

```
#, y#) ctrl.Location = New Point(x * ctrl.Parent.Width / 2, y * ctrl.Parent.Height / 2)
```

You then have to make a animation

PHP Code

```
<?php
```

```
Dim a As New Animation
```

```
a.Add(1000, AddressOf MoveCtrl).Apply(MyButton, Transitions.Linear, Transitions.Linear)
```

1000 is the number of milliseconds that your animation should last

You don't have to write "AddressOf" if you made a lambda

Everything in the second parantece is used as parameters for your method, and you can use a transition instead of any kind of number (doesn't have to be double)

To run your animation you simply call:

PHP Code

```
<?php
```

```
a.Start()
```

The animation class works pretty much like a timer. It have both a Start and a Stop method, and a Interval property too. The interval is how often it should call whatever method you gave it, and it defaults to 40 times a second. It also have a Reset method, which stops the animation and resets everything.

You can also add another animation to the first with the Add method.

Calling Add with a method of your choise actually returns a instance of the "Builder" class, which is a class that let you partially apply parameters to your method:

PHP Code

```
<?php
```

```
a.Add(1000, AddressOf MoveCtrl)(MyButton)(Transitions.Linear).Apply(Transitions.Linear)
```

So if you wanted to move a specific control a lot you could partially apply a control to the builder, and then apply a shitton of transitions to it afterwards:

PHP Code

```
<?php
```

```
Dim whatever = a.Add(1000, AddressOf MoveCtrl)(MyButton)
```

The animation class is actually a list of "Motions" which is a class which have a single transition, a single method (e.g. our MoveCtrl function), and also have StartAt, Length and Loop properties. StartAt and Length are integers representing where to start and the length of the motion in milliseconds, and Loop is a boolean representing wether or not to loop the motion.

If you only need a single motion there is a Transistor class, which is basically a single Transition and a timer in one package.

A Transition is defined like this:

PHP Code

```
<?php
```

```
Delegate Function Transition(x As Double) As Double
```

Which means that any function that both takes and returns a double can be used as a Transition.

In the Transitions class you'll find the following transitions:Linear

Circular

Bounce

Quadrantic

JumpBack1

Elastic1

The reason that the last two transitions ends with "1" is that the Transitions class also have correspondending Jumpback and Elastic functions aswell as a Pow function. These three functions take a double indicating the "Strength" of the transition, and returns a Transition.

You also have a few extension methods for transitions:Negate - Runs the Transition backwards

EaseOut

EaseInOut

Usage:

PHP Code

```
<?php
```

Transitions.Linear.Negate

And a few handy ones for constructing transitions:To - Gives the result of this transition to another

From - Gives the result of another transition to this

ToTransL - Creates a transition from a function taking two parameters, where the left one is the one you'd like to use

ToTransR - Creates a transition from a function taking two parameters, where the right one is the one you'd like to use

I will support this thing for at least a couple of months, and I'd be sweet if you guys cared to hand me your two cents as well as a bucket or two of bug reports.

Corios - October 10, 2014, 8:42 pm

Just curious, why would you want to use this over say WPF? Not attacking you, just curious as to why use this when there are other extremely developed options

BlackCap - October 10, 2014, 8:46 pm

(Oct 10, 2014 08:42 PM)Corios Wrote: [\[CITE\]](#) Just curious, why would you want to use this over say WPF? Not attacking you, just curious as to why use this when there are other extremely developed options

Partly for backwards compability, but mostly because it's a lot more fun to make stuff yourself. Also, this isn't windows forms exclusive; it would work just as good with WPF.

Corios - October 10, 2014, 9:33 pm

(Oct 10, 2014 08:46 PM)BlackCap Wrote: [\[CITE\]](#) Partly for backwards compability, but mostly because it's a lot more fun to make stuff yourself. Also, this isn't windows forms exclusive; it would work just as good with WPF.

Fair enough :P I'll check it out once I get back, looks pretty good!

SaketSaket - October 11, 2014, 2:38 am

Would love to see how it actually works.... Wonderful work... ;)

Only3 - October 11, 2014, 11:37 am

back to net section looks good if i remember u have another thread about it

SIM-1456245 - October 11, 2014, 11:57 am

Very impressive man, I was really looking forward to learn about animations in VB.Net, but hmm, this moves around controls only, what about GDI animation class ?