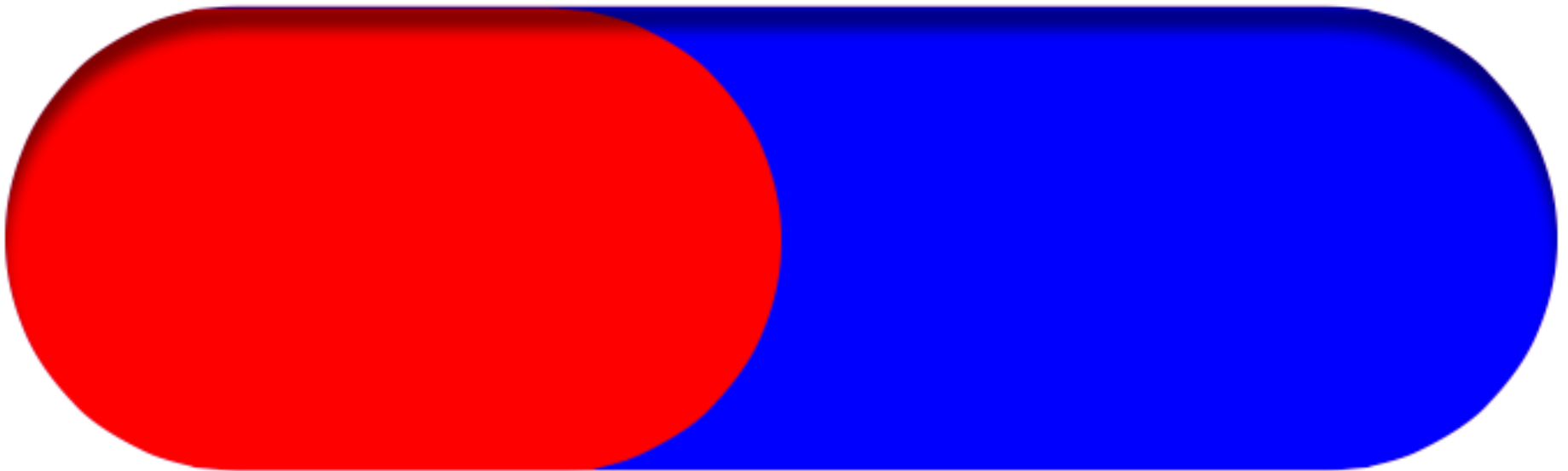


Custom UIViews from CALayers

Heath Borders (@heathborders)
Senior Software Engineer
Asynchrony Solutions (@asynchrony)
St. Louis, MO
<http://asolutions.com>

Create a Progress View



Simplest Way

- Get 3 Images
 - Background, Completion, Shadow
- Fixed size, colors :(
 - Can mitigate with `-[UIImage resizableImageWithCapInsets:]`
- Booooooring :(

More Fun Way

- Use 2 CAShapeLayers
 - For the Background, Completion
- Set a shadow on one and mask it out

CAShapeLayer

- Draws using a CGPath
- Use UIBezierPath as an easy way to create your CGPath

Shadows

- Just some properties on CALayer
 - @property CGColorRef shadowColor
 - @property float shadowOpacity
 - I don't know why this isn't a CGFloat
 - @property CGSize shadowOffset
 - @property CGFloat shadowRadius

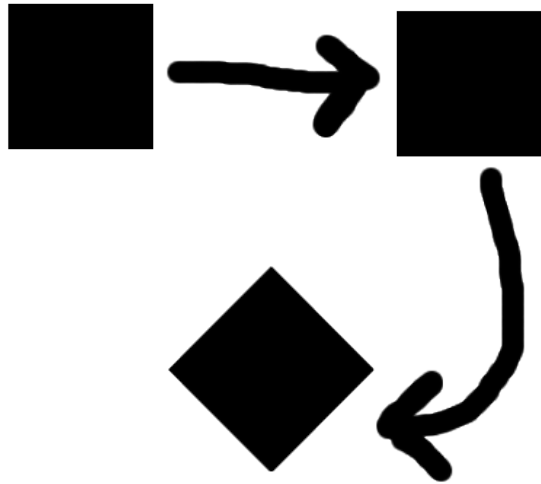
Live Coding

- Coding from scratch
- Don't worry. I practiced, and I have a series of granular git commits to back me up.
- Follow along:
 - `git clone https://github.com/hborders/HBProgressView.git`

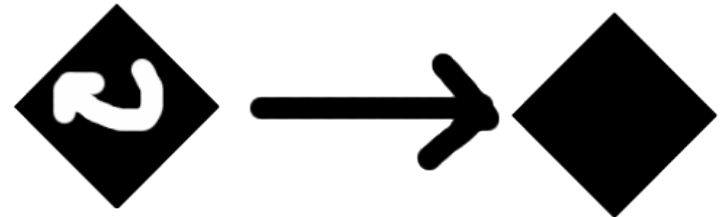
Affine Transform

- Move, Scale, Rotate
- Order Matters!

Translate then rotate
(Assuming we start at the origin)



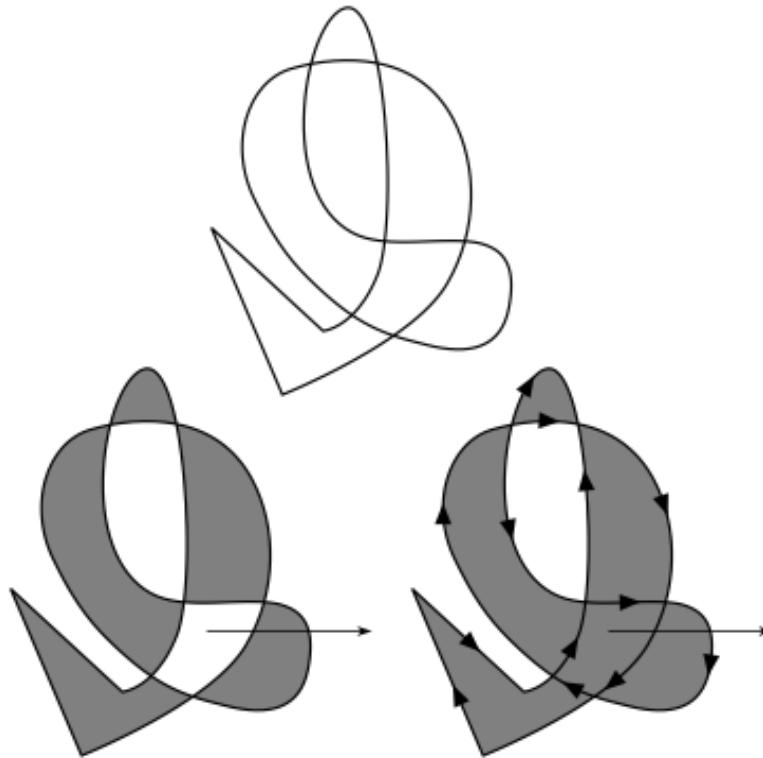
Rotate then translate
(Assuming we start at the origin)



More Live Coding

Inside or Outside?

- Even-Odd vs. Non-Zero Winding



Next Steps

- Make completionFactor animatable
 - <https://github.com/briancoyner/Core-Animation-Pie-Chart>
- Try Other CoreAnimation Demos
 - <https://github.com/bobmccune/Core-Animation-Demos>
- Try Some Quartz Demos
 - (not directly CoreAnimation related, but this is cool)
 - <https://github.com/tapharmonic/QuartzDemos>