Easy

COMPLEX UICONTROLS

SPEC

SPEC (DIAL)

- Track

SPEC (DIAL)

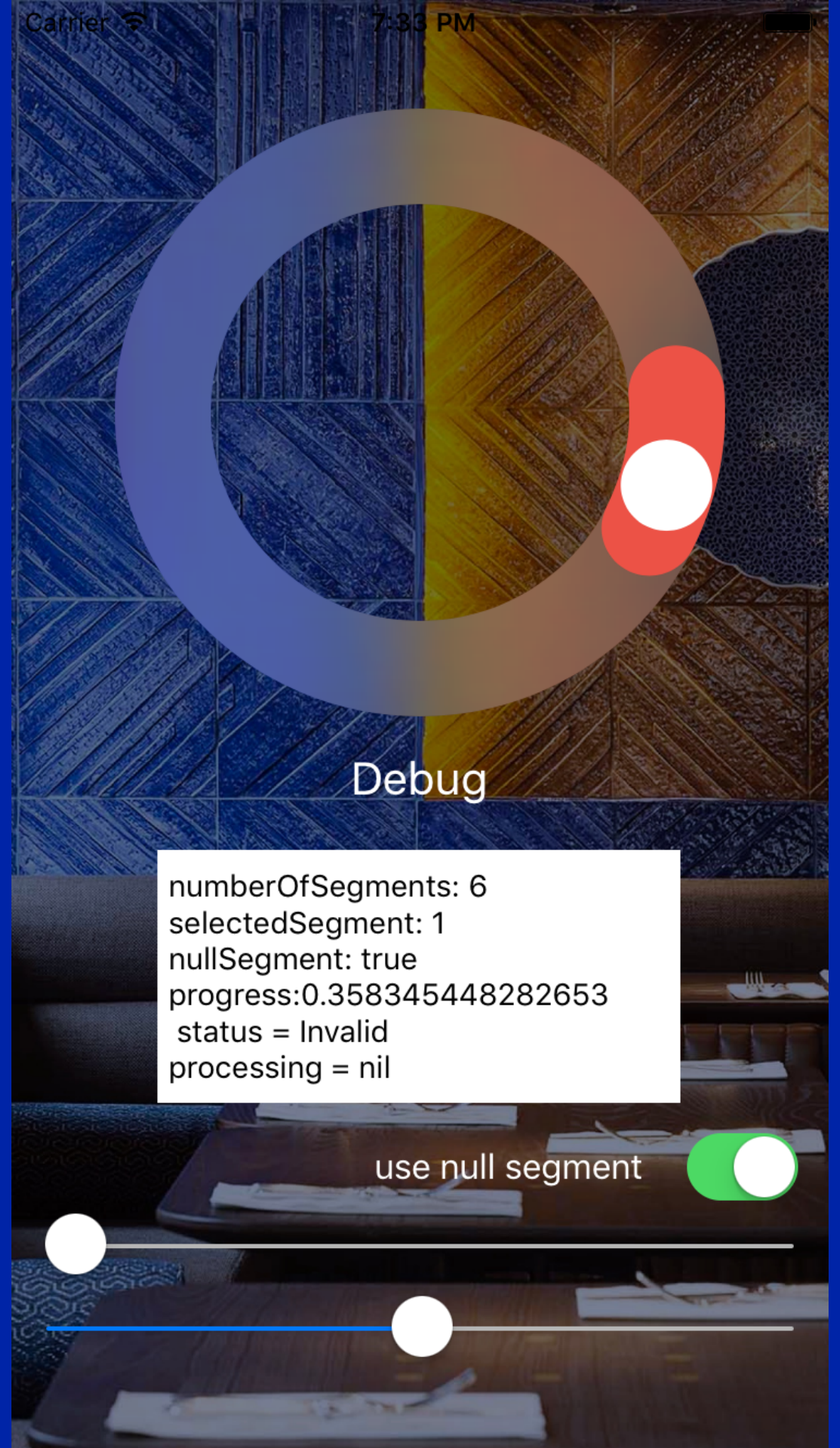
- Track
- Handle

SPEC (DIAL)

- Track
- Handle
- segment selection

SPEC (DIAL / SEGMENTS)

- arbitrary number of segments
- `null` or '*unselected*' segment
  - '*unavailable*' segments



SPEC (DIAL / SEGMENTS)

- arbitrary number of segments
- `null` or '*unselected*' segment
  - '*unavailable*' segments

OUT OF SCOPE

- trailing color

^ though we can have unavailable

- inside button

TRACK

```
let layer = CAShapeLayer()
layer.path = UIBezierPath(ovalInRect:self.bounds).CGPath
layer.fillColor = .clearColor()
layer.lineWidth = 44
layer.strokeColor = UIColor(white:1, alpha:0.5).CGColor
```

TRACK

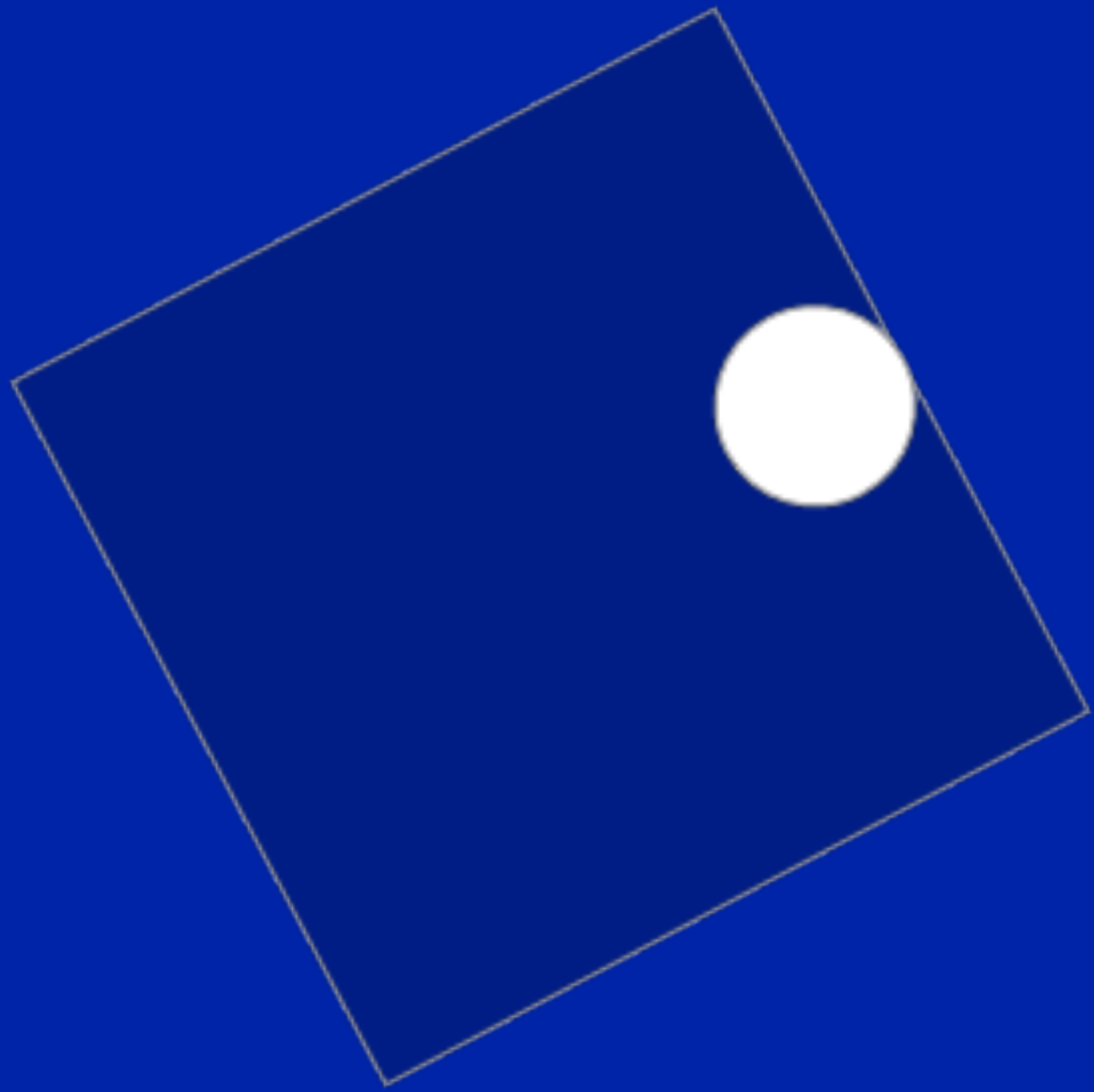
```
let layer = CAShapeLayer()
layer.path = UIBezierPath(ovalInRect:self.bounds).CGPath
layer.fillColor = .clearColor()
layer.lineWidth = 44
layer.strokeColor = UIColor(white:1, alpha:0.5).CGColor
```

HANDLE

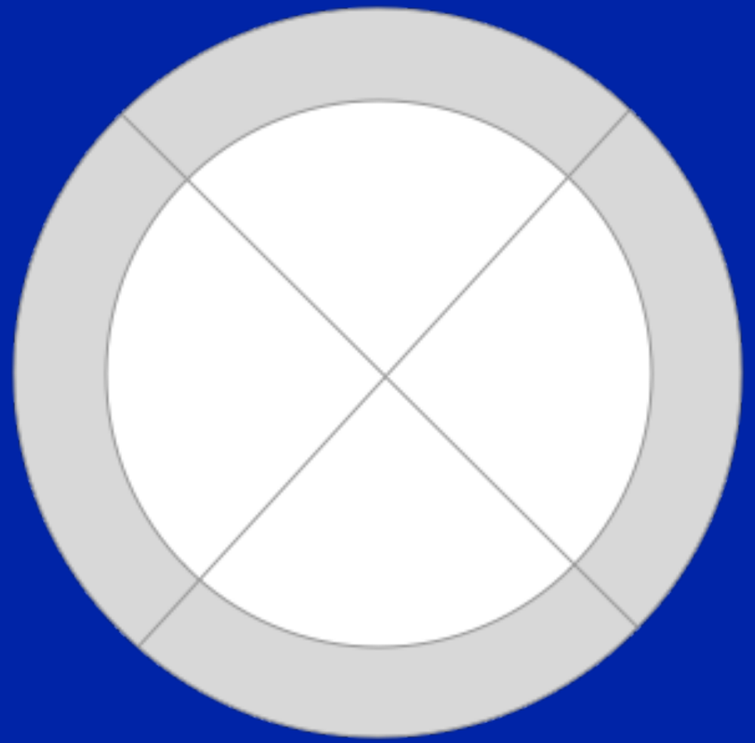
```
let hadleContainerLayer = CALayer()
let handleLayer = CAShapeLayer()

hadleContainerLayer.frame = self.bounds
hadleContainerLayer.path = UIBezierPath(ovalInRect:CGRect(x:0,y:0,width:42,height:42)).CGPath
hadleContainerLayer.addSublayer(handleLayer)
handleLayer.fillColor = .whiteColor().CGColor
handleLayer.position = CGPoint(x:y:hadleContainerLayer.frame.height - 44/2 ,y:hadleContainerLayer.frame.height/2)

//display
hadleContainerLayer.backgroundColor = UIColor.blackColor().CGColor
```

HANDLE

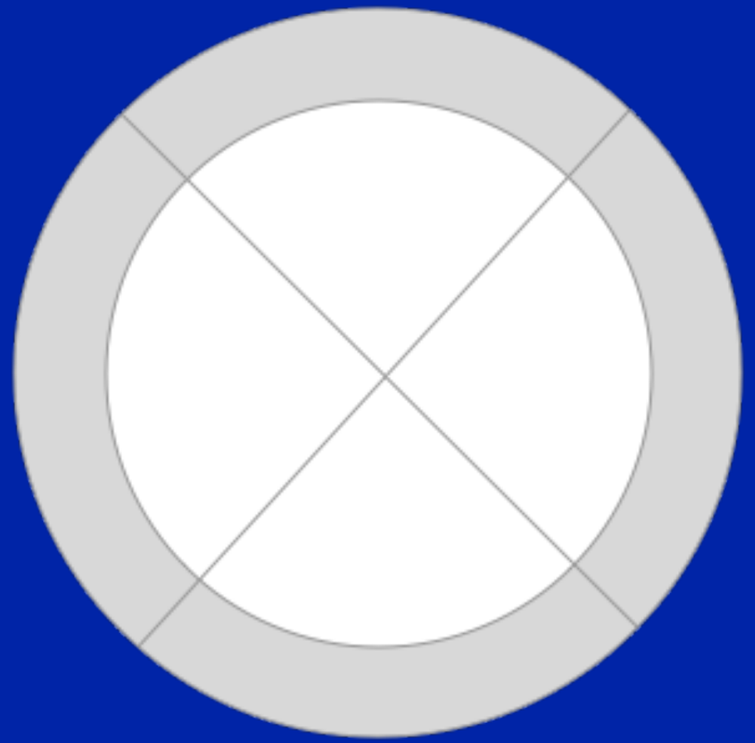
```
CATransform3DMakeRotation(angle, 0, 0, 1)
```

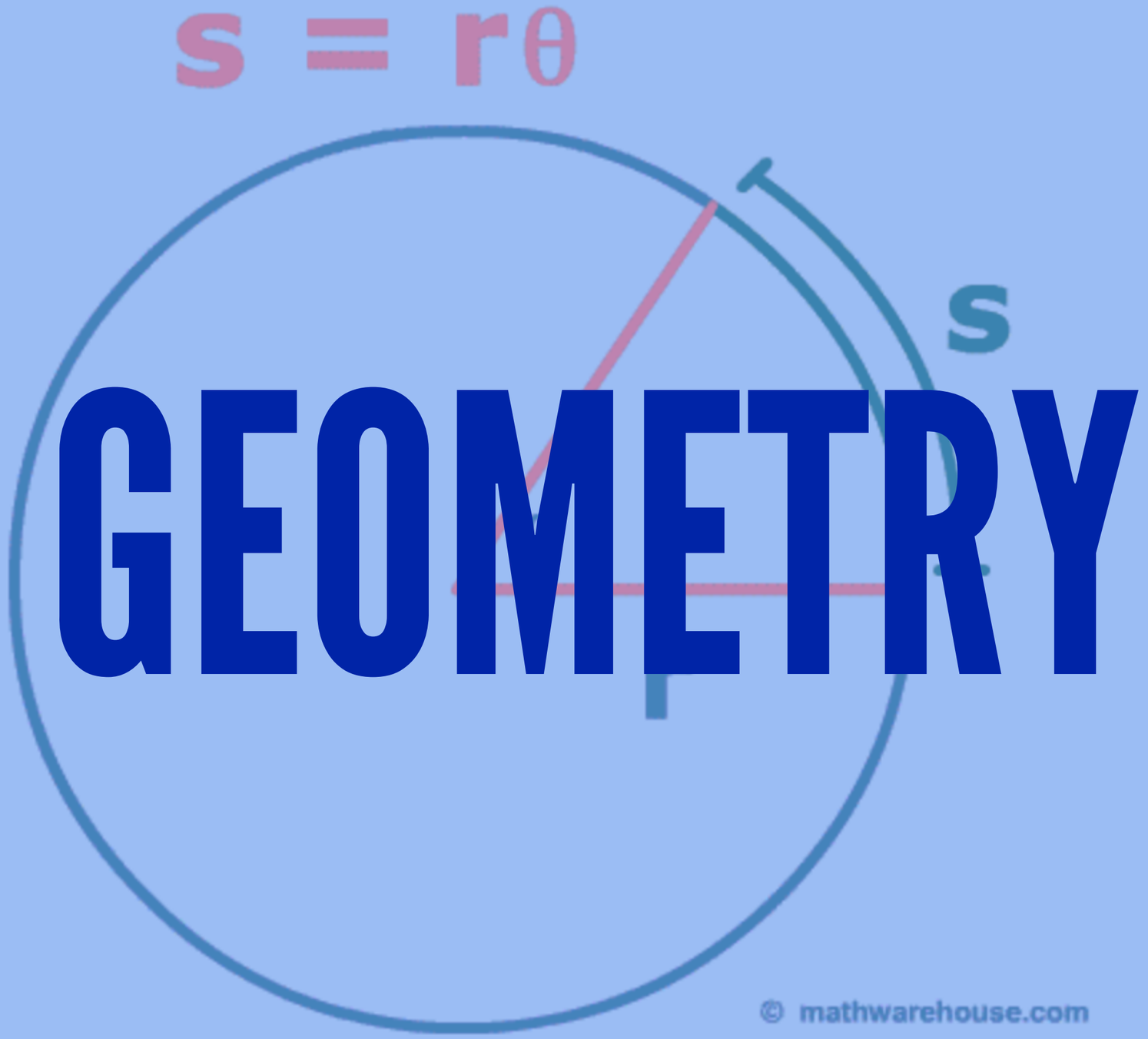


SEGMENT HIGHLIGHT



SEGMENT HIGHLIGHT





GEOMETRY

SEGMENT HIGHLIGHT

```
let layer = CAShapeLayer()
let path = UIBezierPath(
arcCenter: self.position,
 radius: radius,
  startAngle: startAngle,
   endAngle:endAngle,
    clockwise: true
    ).CGPath

layer.path = path
```

SEGMENT HIGHLIGHT

Easy

COMPLEX UICONTROLS

Easy COMPLEX UICONTROLS once you have already implemented them

MISSED:

- Performance
- null segment
- user interaction
- segment highlighting
- sending events
- state / rendering

PERFORMANCE

UIViews + CGAffineTransform

NULL SEGMENT

SEGMENT HIGHLIGHTING

SEGMENT HIGHLIGHTING

Simulator

SEGMENT HIGHLIGHTING

USER INTERACTION

SENDING EVENTS

- ChangeValue

STATE / RENDERING

```
self.state = self.state.newStateWithAngle(angle)
```

Then render...

```
var state: DialState {
    didSet {
        self.render(state)
    }
}
```

ANIMATIONS

Easy

- growFromStart
- growFromEnd
- shrinkMiddle
- expandMiddle

ANIMATIONS

Easy

```
let pathAnimation = CABasicAnimation(keyPath: "strokeEnd")
pathAnimation.duration = 0.15
pathAnimation.fillMode = kCAFillModeBoth
pathAnimation.fromValue = 0
pathAnimation.toValue = 1
pathAnimation.removedOnCompletion = true
pathAnimation.timingFunction =
 CAMediaTimingFunction(name: kCAMediaTimingFunctionEaseOut)
```

ANIMATIONS

Hard

SEMI FINAL PRODUCT