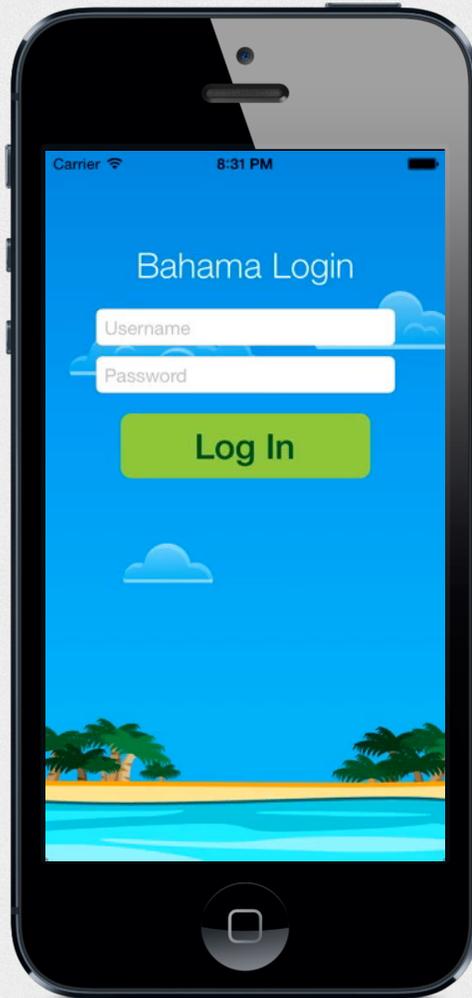
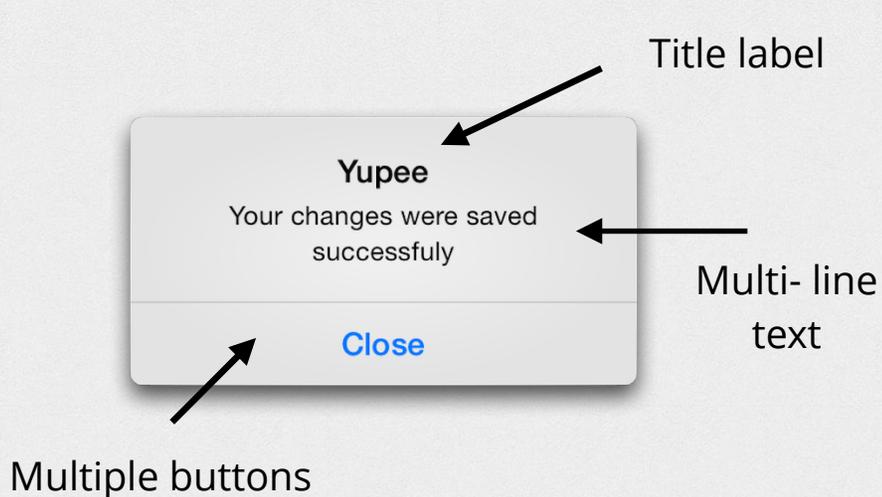


iOS Animation with Swift

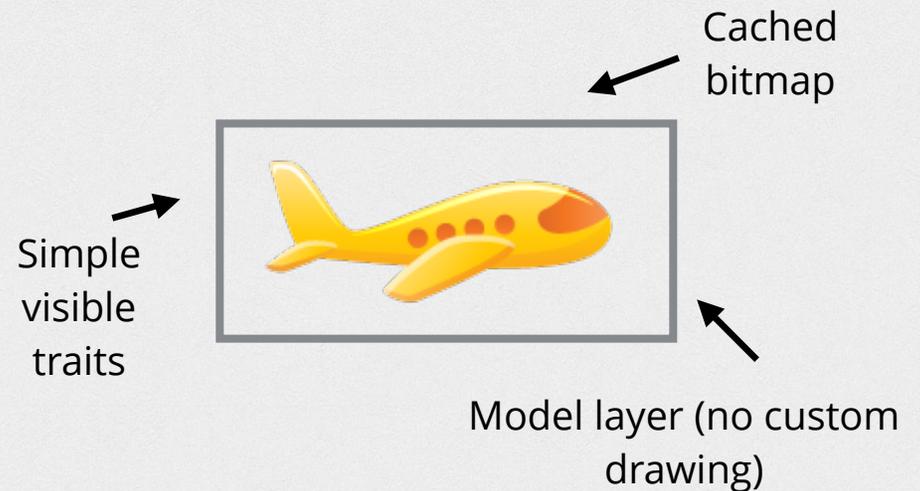
Part 6: Basic Layer Animations



Views vs. layers



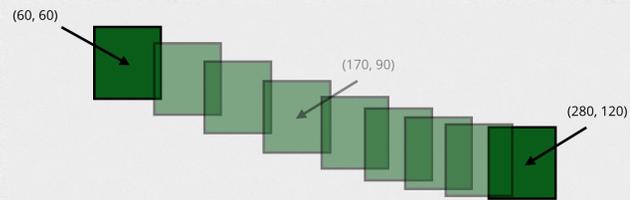
- ⚙️ complex, view hierarchy
- ⚙️ constrains, resize masks, etc.
- ⚙️ drawRect: runs on the CPU



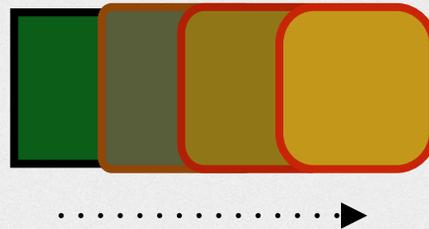
- ⚙️ simpler, layer hierarchy
- ⚙️ drawn directly on the GPU

Layer animations

- ⚙ bounds
- ⚙ position
- ⚙ transform



- ⚙ borderColor
- ⚙ borderWidth
- ⚙ cornerRadius



Animatable Properties

- ⚙ bounds
- ⚙ position
- ⚙ anchorPoint
- ⚙ backgroundColor
- ⚙ borderColor
- ⚙ borderWidth
- ⚙ cornerRadius
- ⚙ shadowOffset
- ⚙ shadowOpacity
- ⚙ shadowPath
- ⚙ shadowRadius
- ⚙ transform
- ⚙ zPosition
- ⚙ contents
- ⚙ mask
- ⚙ masksToBounds
- ⚙ opacity
- ⚙ sublayerTransform



CABasicAnimation

```
import QuartzCore
```

```
let flyRight = CABasicAnimation(keyPath: "position.x")
flyRight.fromValue = -view.bounds.width/2
flyRight.toValue = view.bounds.width/2
flyRight.duration = 0.5
heading.layer.addAnimation(flyRight, forKey: nil)

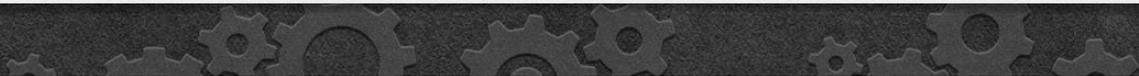
heading.layer.position.x = view.bounds.width/2
```

┆ Create animation

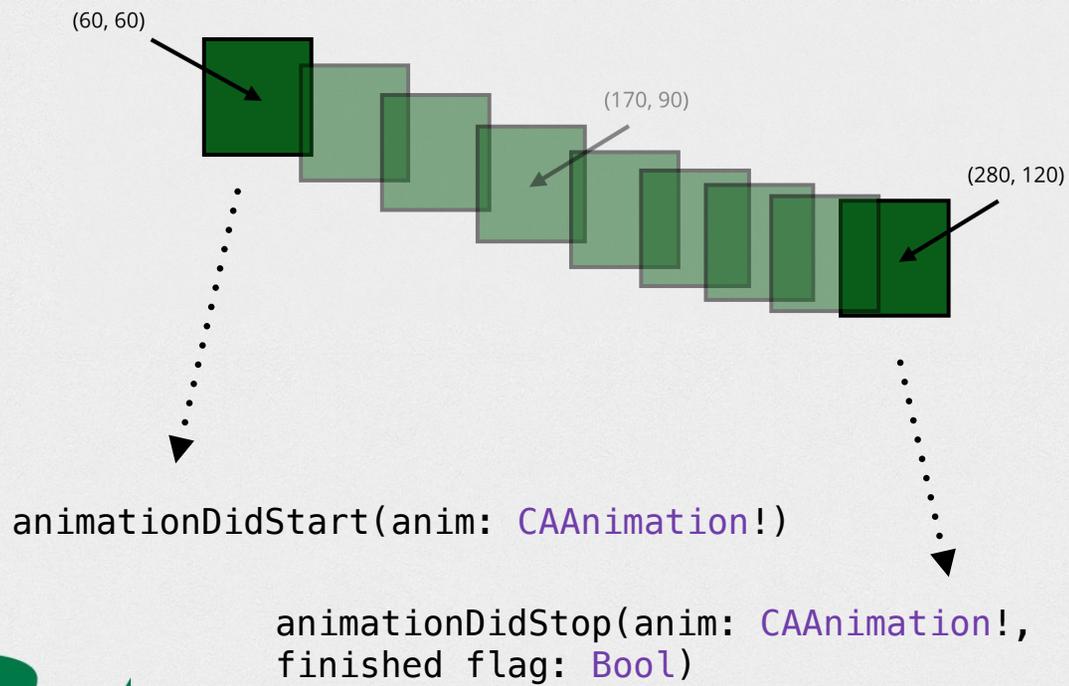
┆ Send animation to GPU

┆ Update layer model

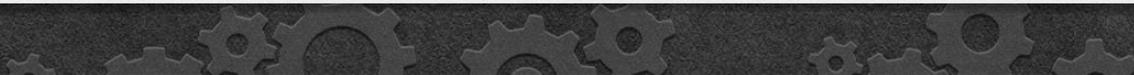
Demo: CABasicAnimation demo



CAAnimation delegate



Demo: CABasicAnimationDelegate demo



Challenge Time!

