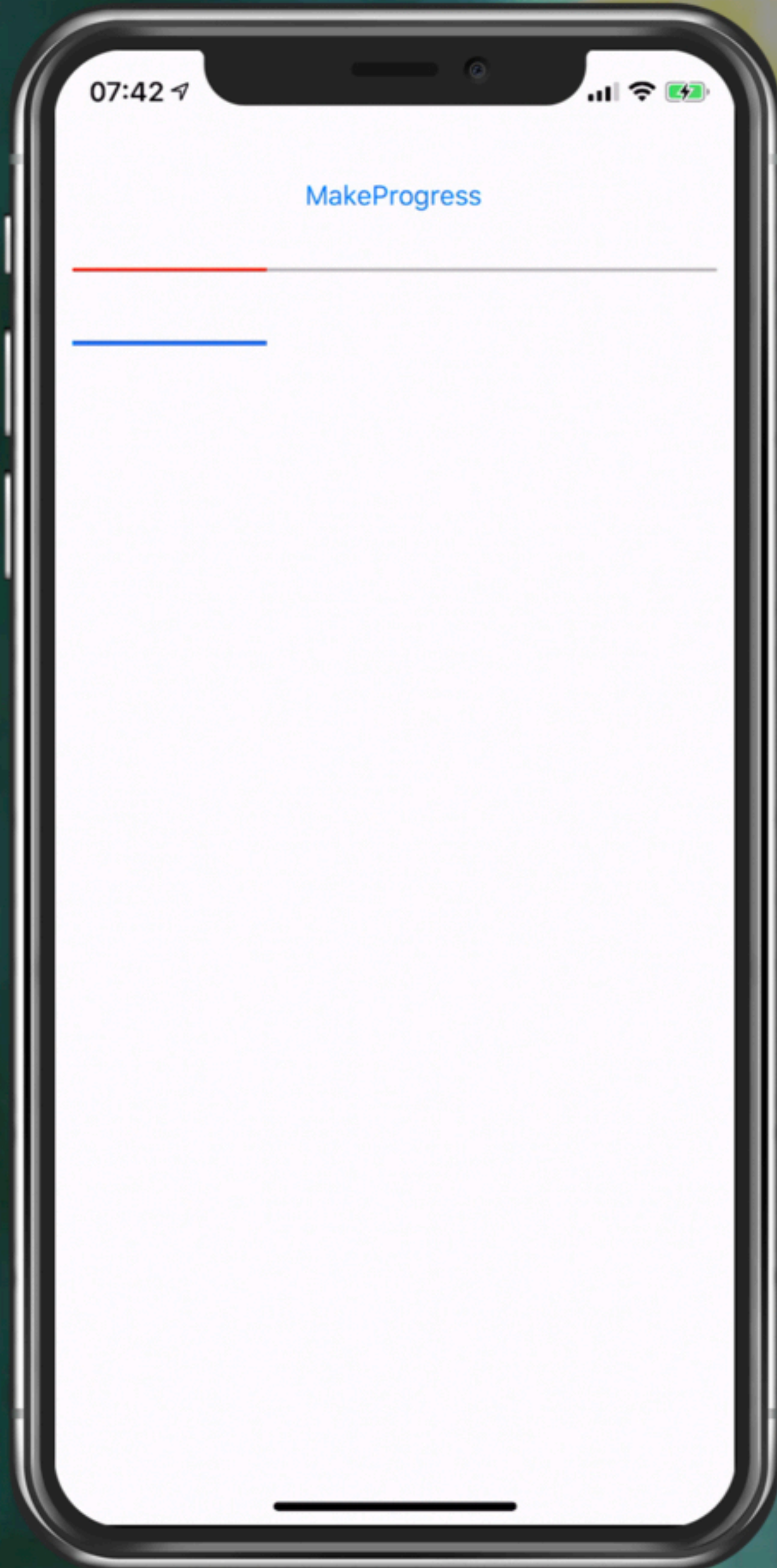


UIProgressView

Fabrice.Kordon@lip6.fr



As an introduction...



Still when the user wait



Alternative to



But with some completion estimation



Evolves with computation



Progression, a value $\in [0,1]$

▶ Property progress



Two styles

▶ default, bar

▶ `UIProgressViewStyleDefault`,
`UIProgressViewStyleBar`

Example



ViewController



Sake of simplicity...

Code located in a
ViewController

ViewController

```
import UIKit

class ViewController: UIViewController {

    private let b = UIButton(type: .system)
    private let p1 = UIProgressView(progressViewStyle: .default)
    private let p2 = UIProgressView(progressViewStyle: .bar)

    override func viewDidLoad() {
        super.viewDidLoad()
        self.view = UIView()
        self.view.backgroundColor = UIColor.white
        b.setTitle("MakeProgress", for: .normal)
        b.addTarget(self, action: #selector(makeProgress), for: .touchDown)
        p1.tintColor = UIColor.red
        self.view.addSubview(b)
        self.view.addSubview(p1)
        self.view.addSubview(p2)
        self.displayInSize(size: UIScreen.main.bounds.size)
    }

    func displayInSize(size : CGSize) {
        b.frame = CGRect(x: Int(size.width)/2 - 50, y: 70, width: 100, height: 20)
        p1.frame = CGRect(x: 10, y: 120, width: Int(size.width) - 20, height: 20)
        p2.frame = CGRect(x: 10, y: 160, width: Int(size.width) - 20, height: 20)
    }
}
```

ViewController

```
@objc func makeProgress() {
    if p1.progress < 1.0 {
        p1.progress += 0.1
        p2.progress += 0.1
    } else {
        p1.isHidden = true
        p2.isHidden = true
    }
}

override func viewWillTransition(to size: CGSize,
    with coordinator: UIViewControllerTransitionCoordinator) {
    super.viewWillTransition(to: size, with: coordinator)
    self.displayInSize(size:size)
}
}
```

As a conclusion...

When some computation lasts

- You need to have «something going on the screen»



The «golden rule»

- When you cannot estimate the progression
 - ▶ `UIActivityIndicatorView`
- When you can estimate the progression
 - ▶ `UIProgressView`