

ШАБЛОН MODEL-VIEW ПРОЕКТИРОВАНИЯ ПРИЛОЖЕНИЙ НА ЯЗЫКЕ SWIFT

ГРУНЕВ ДЕНИС АЛЕКСАНДРОВИЧ,
АСПИРАНТ КАФЕДРЫ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ
НОВГУ

ДНИ НАУКИ И ИННОВАЦИЙ НОВГУ 2021

ЧТО ТАКОЕ ШАБЛОН

ПОД ШАБЛОНОМ ПРОЕКТИРОВАНИЯ (DESIGN PATTERN) ПОНИМАЕТСЯ СПОСОБ СОЗДАНИЯ ПРИЛОЖЕНИЯ, КОТОРЫЙ ОБЛЕГЧАЕТ ЕГО РАЗРАБОТКУ, ПОДДЕРЖКУ И РАЗВИТИЕ

ЛИТЕРАТУРА

- Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. ПРИЕМЫ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОЕКТИРОВАНИЯ. ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ
- ФРИМЕН Э., ФРИМЕН. Э. ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ

ШАБЛОН MVC (MODEL-VIEW- CONTROLLER)

VIEW

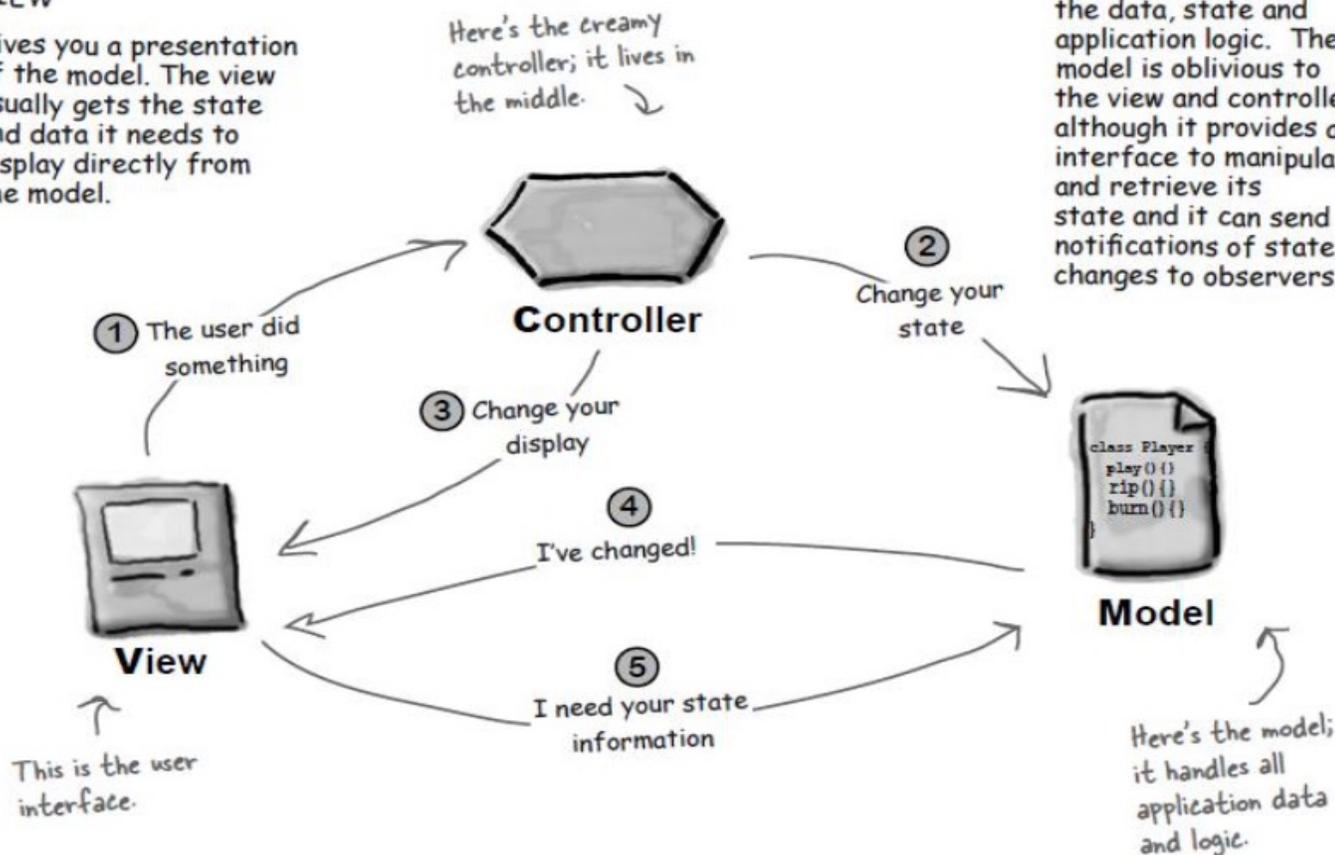
Gives you a presentation of the model. The view usually gets the state and data it needs to display directly from the model.

CONTROLLER

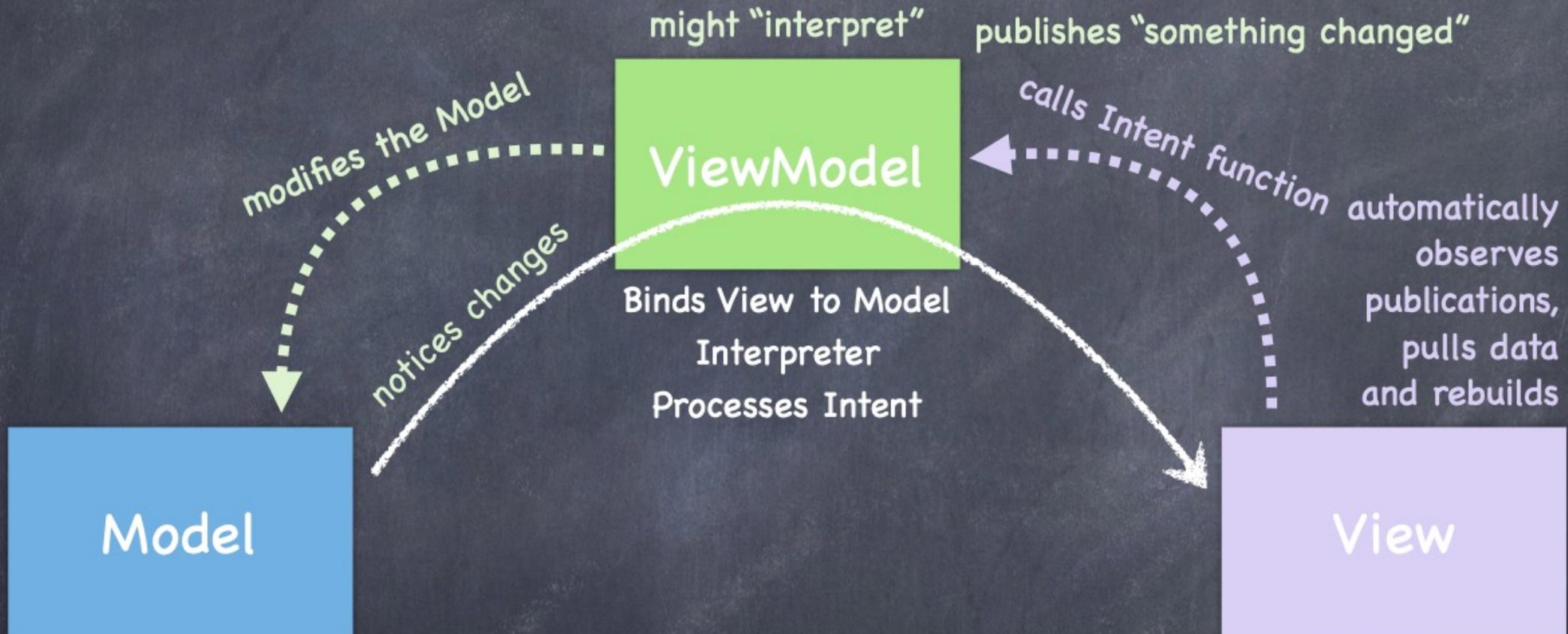
Takes user input and figures out what it means to the model.

MODEL

The model holds all the data, state and application logic. The model is oblivious to the view and controller, although it provides an interface to manipulate and retrieve its state and it can send notifications of state changes to observers.



ШАБЛОН MODEL-VIEW



VIEW

```
        Button(action: {
            viewModel.deleteLastPoint()
        }) {
            Text("Удалить")
        }
    }
    Spacer()
    Picker("Выберите число шагов N", selection: $selectedInt){
        ForEach(ints, id: \.self){
            Text("\($0)")
        }
    }
    Text("Выбранное число шагов N: \($selectedInt)")
    Spacer()
    HStack{
        Button(action: {
            viewModel.calcRisk(numSteps: selectedInt)
        }) {
            Text("Вычислить риск")
        }
        Text(viewModel.risk)
    }
}
```

Добавить

Удалить

10

15

20

25

30

35

Выбранное число шагов N: 15

Вычислить риск не подсчитан

```
class ModelView: ObservableObject {
    @Published private var model: BernulliModel = BernulliModel()

    // Access
    var points: [Point]{
        model.points
    }

    var risk: String{
        model.risk
    }

    // Intent(s)
    func deleteLastPoint(){
        model.deletePoint()
    }

    func calcRisk(numSteps: Int){
        model.calcRisk(numSteps: numSteps)
    }
}
```

VIEW- MODEL

```
struct BernulliModel {
  var points = [
    Point(id: 0, p: 0.214, dp: 0.154, lambda: 0.115),
    Point(id: 1, p: 0.312, dp: 0.149, lambda: 0.165),
    Point(id: 2, p: 0.373, dp: 0.147, lambda: 0.164),
    Point(id: 3, p: 0.436, dp: 0.145, lambda: 0.147),
    Point(id: 4, p: 0.503, dp: 0.144, lambda: 0.134),
    Point(id: 5, p: 0.560, dp: 0.143, lambda: 0.125),
    Point(id: 6, p: 0.640, dp: 0.143, lambda: 0.150)
  ]

  var risk: String = "не подсчитан"

  mutating func calcRisk(numSteps: Int){
    let bernulli = Bernulli(numSteps: numSteps, points: points)
    risk = bernulli.go()
  }

  mutating func deletePoint(){
    points.removeLast()
  }

  mutating func addPoint(point: Point){
    points.append(point)
  }
}
```

MODEL

ПОЛЕЗНЫЕ ИСТОЧНИКИ

- THE SWIFT PROGRAMMING LANGUAGE. SWIFT 5.3 EDITION (В APPLE BOOKS)
- КУРС DEVELOPING APPS FOR IOS ([HTTPS://CS193P.SITES.STANFORD.EDU](https://cs193p.sites.stanford.edu))
- YOUTUBE-КАНАЛ IAN SOLOMEIN
- [HTTPS://DEVELOPER.APPLE.COM](https://developer.apple.com)

СПАСИБО ЗА ВНИМАНИЕ!