MVC & MVP &MVVM

李湛

2020/5/16

目录

① 发展历程

② 实际案例

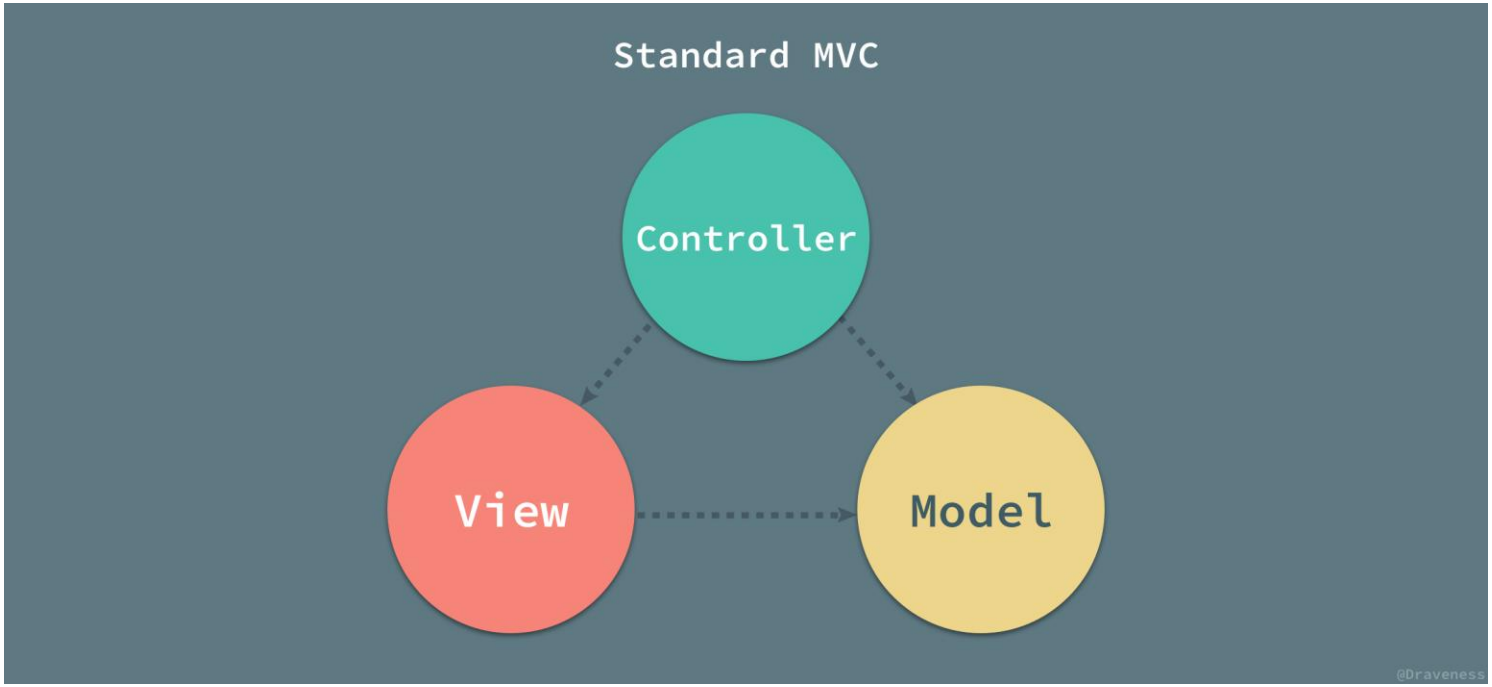
发展历程

1. 解耦角度
2. 应用领域角度

从解耦的角度

MVC基础流程

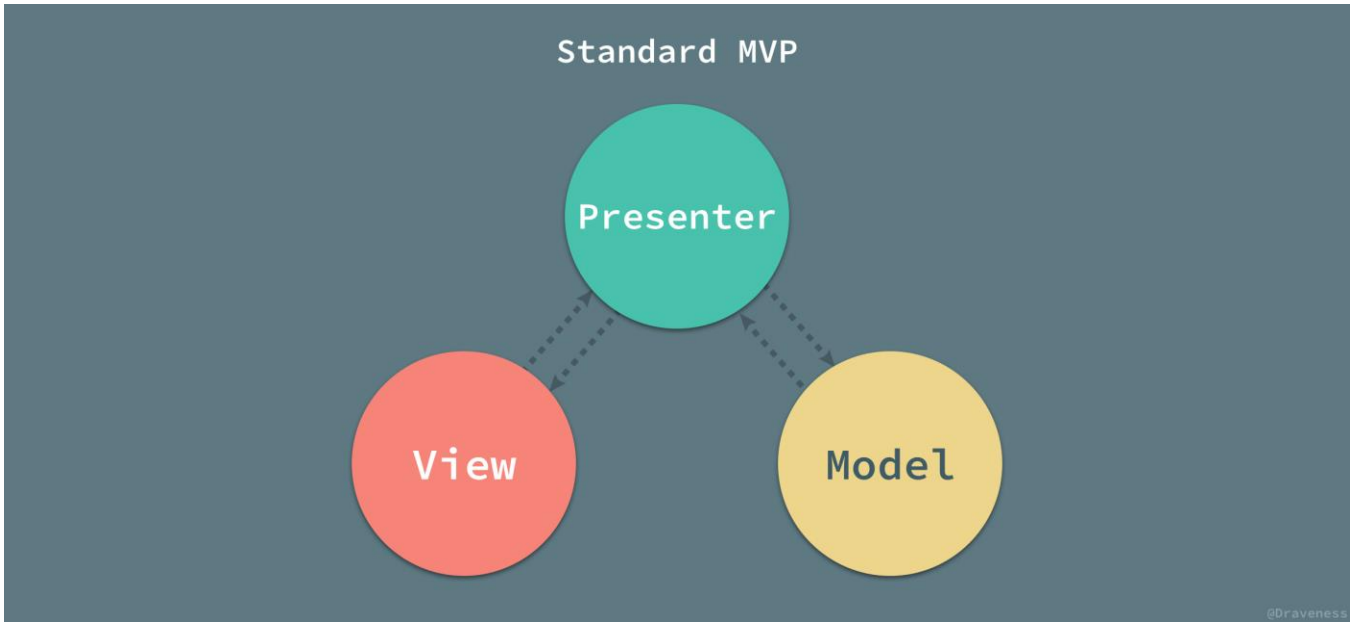
1. Controller管理View和Model
2. View通知Controller处理
3. Controller转发给Model处理
4. Model通知View更新界面



MVP基础流程

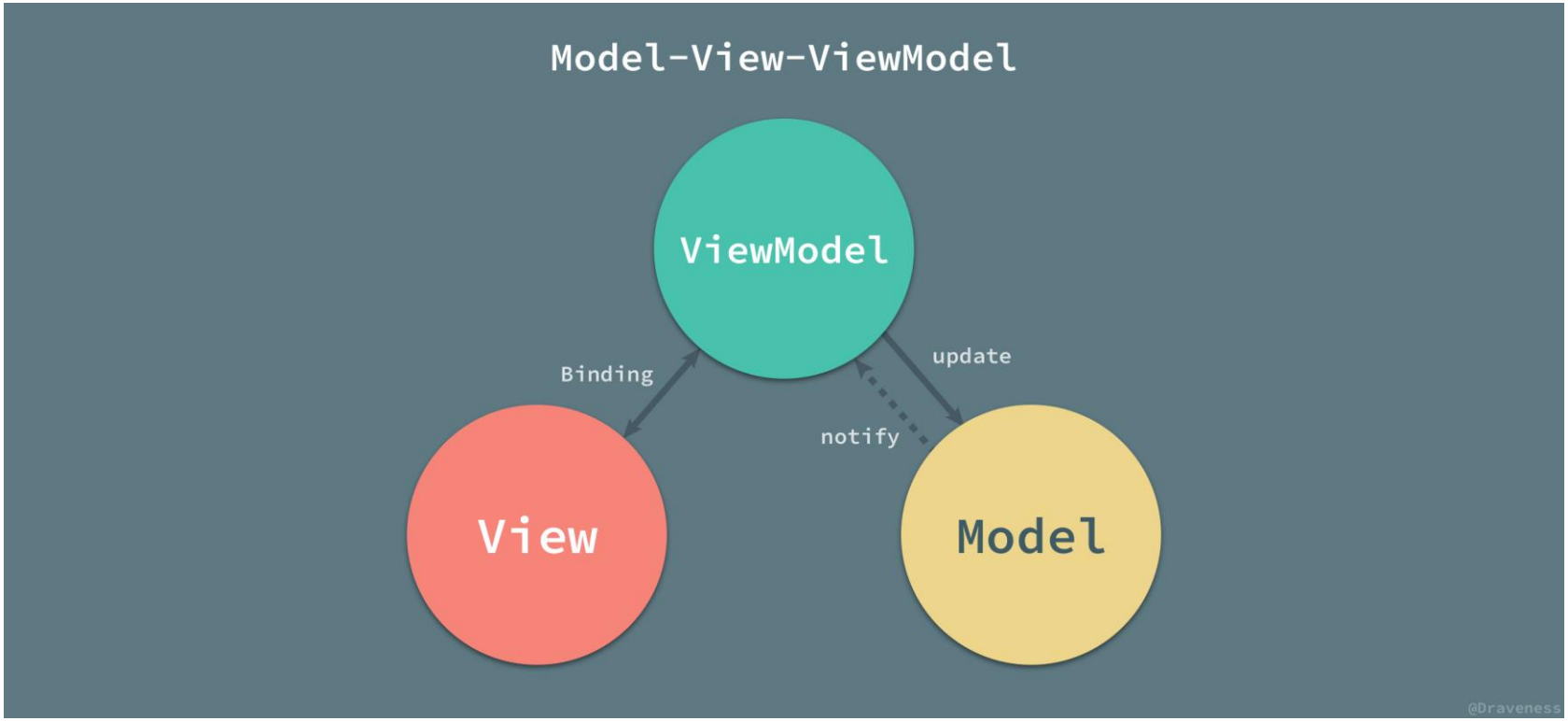
MVC里面耦合太紧密，MVP将Model和View彻底分开。

1. 用户输入
2. View通知Presenter处理
3. Presenter拿到Model处理之后的数据
4. Model通知View更新界面



MVVM基础流程

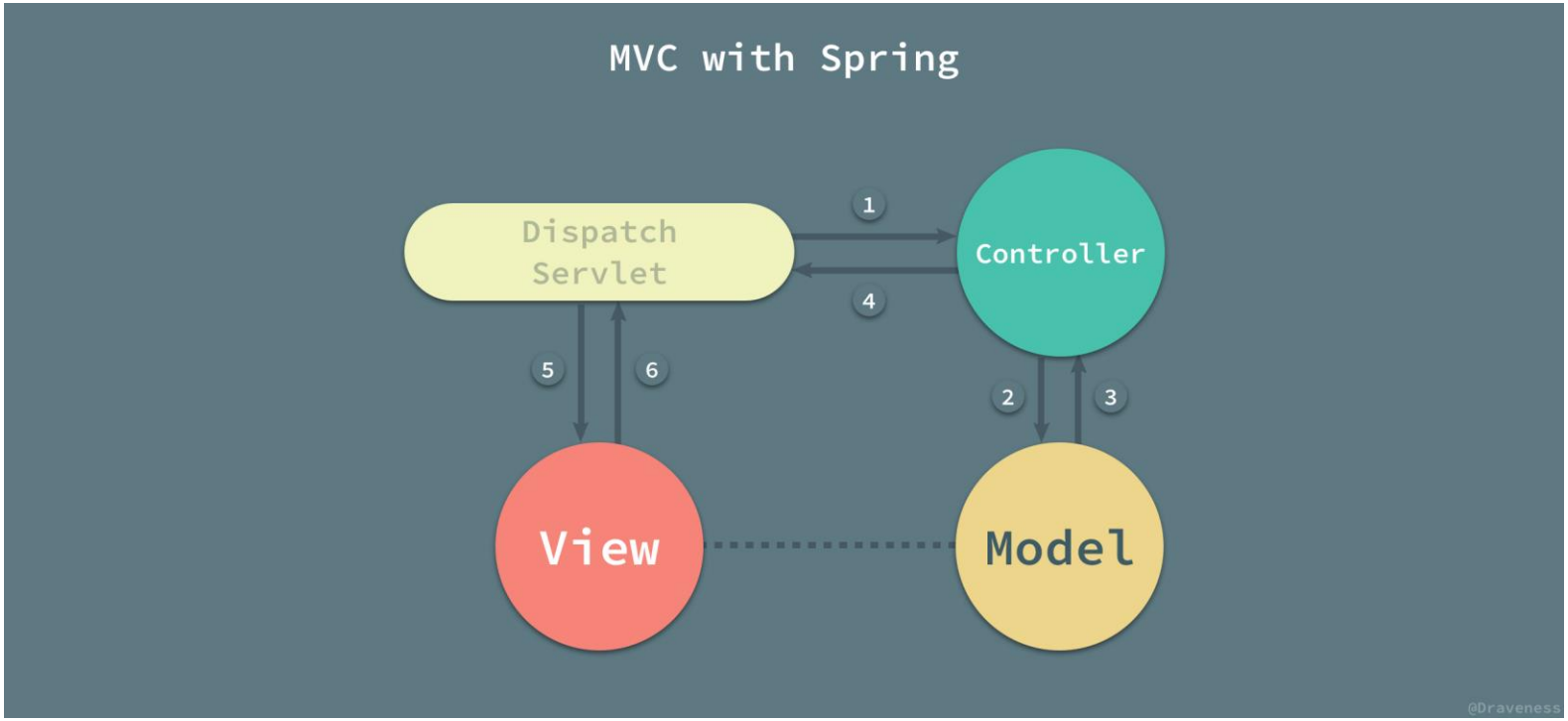
MVP里面Presenter做的事情太繁琐，MVVM能把界面和数据双向绑定起来，View更新，自动更新Model，Model更新，View自动刷新界面。



从应用领域的角度

MVC with Spring

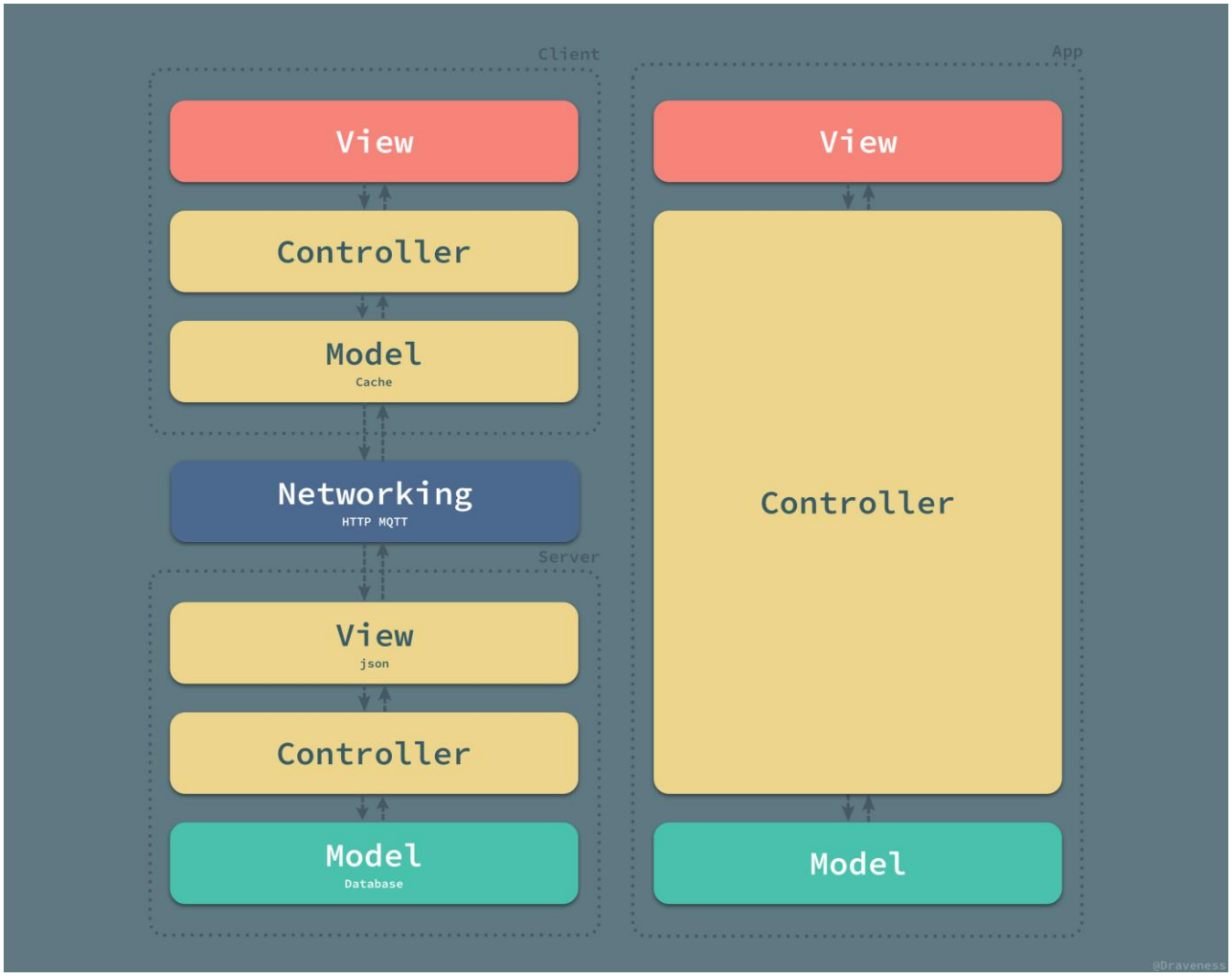
- 通过 DispatchServlet 将控制器层和视图层完全解耦;
- 视图层和模型层之间没有直接关系，只有间接关系，通过控制器对模型进行查询、返回给 DispatchServlet 后再传递至视图层。



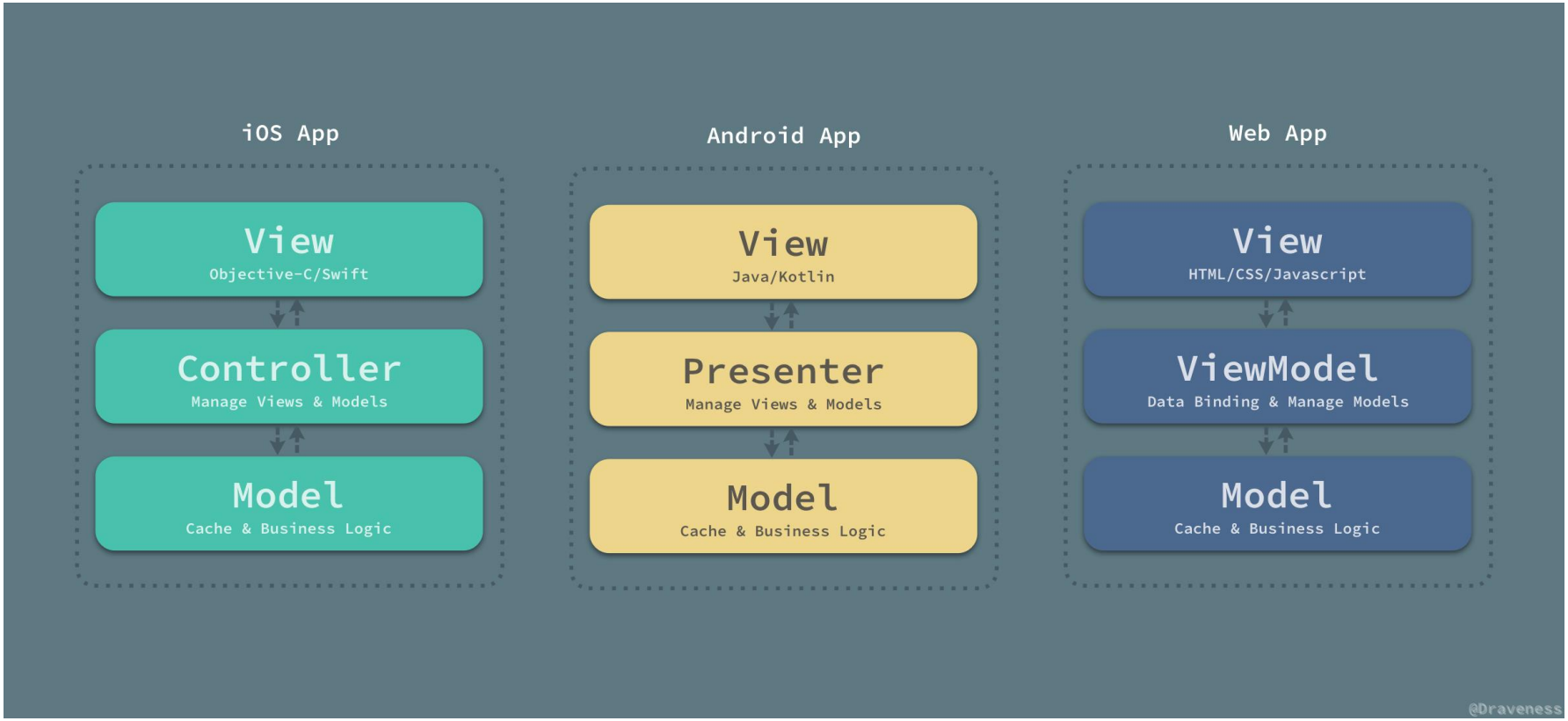
前后端分离

- 前端越来越臃肿
- 前端单独作为View显示，难以满足需求

前后端分离


Client
View
Controller
Model
Cache
Networking
HTTP MQTT
Server
View
json
Controller
Model
Database
App
View
Controller
Model
@Draveness

多平台多模式


iOS App
View
Objective-C/Swift
Controller
Manage Views & Models
Model
Cache & Business Logic
Android App
View
Java/Kotlin
Presenter
Manage Views & Models
Model
Cache & Business Logic
Web App
View
HTML/CSS/Javascript
ViewModel
Data Binding & Manage Models
Model
Cache & Business Logic
@Draveness

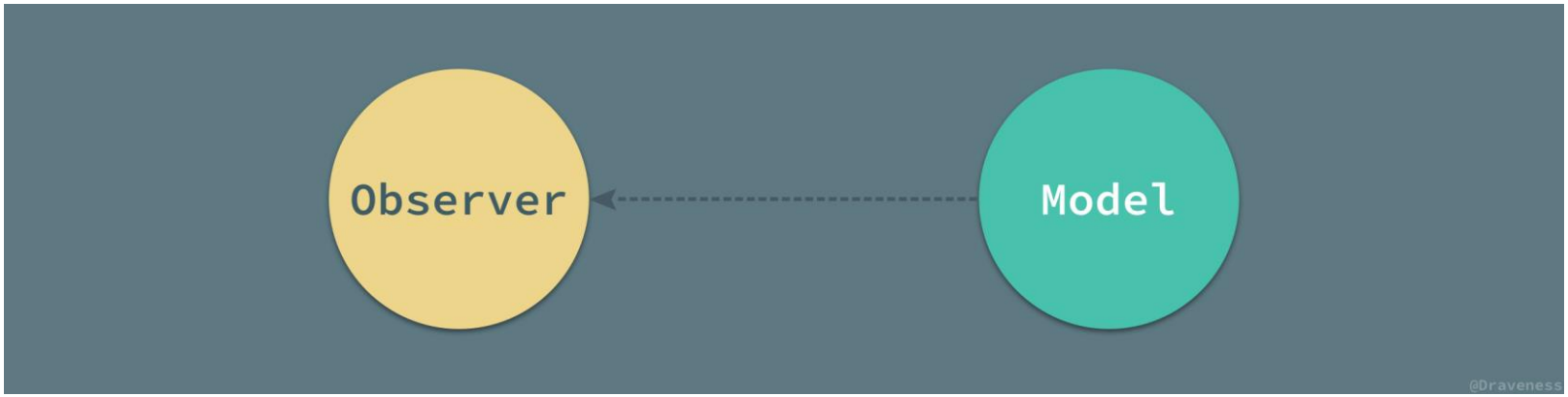
实际案例

1. 我们的产品
2. 其他界面开发技术

我们的产品

- 数据和界面分离
- VM和C分离
- Model一般用来操作json/ini文件等，用的地方不多

- 准确来说，我们更像



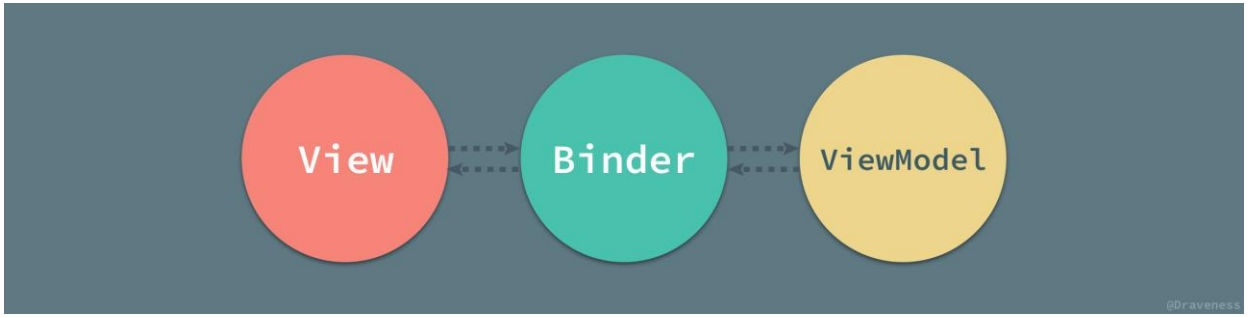
Qt里面的Model和View

- 例如QListView和QListModel

- 在滚动时，view会通知model计算位置，view负责展示当前的列表；
- 修改model的数据时，model会通知view更新界面。

WPF里的双向绑定

- WPF隐含了一个Binder层，用于数据的双向绑定。




```
<Window x:Class ="WPFDataBinding.MainWindow" Title="MainWindow" Height="350" Width="604">
   <Grid>
      <Label Name="nameLabel" Margin="2">_Name:</Label>
      <TextBox Name="nameText" Grid.Column="1" Margin="2"
         Text="{Binding Name}"/>
      <Label Name="ageLabel" Margin="2" Grid.Row ="1">_Age:</Label>
      <TextBox Name="ageText" Grid.Column="1" Grid.Row ="1" Margin ="2"
         Text="{Binding Age}"/>
   </Grid>
</Window>
```

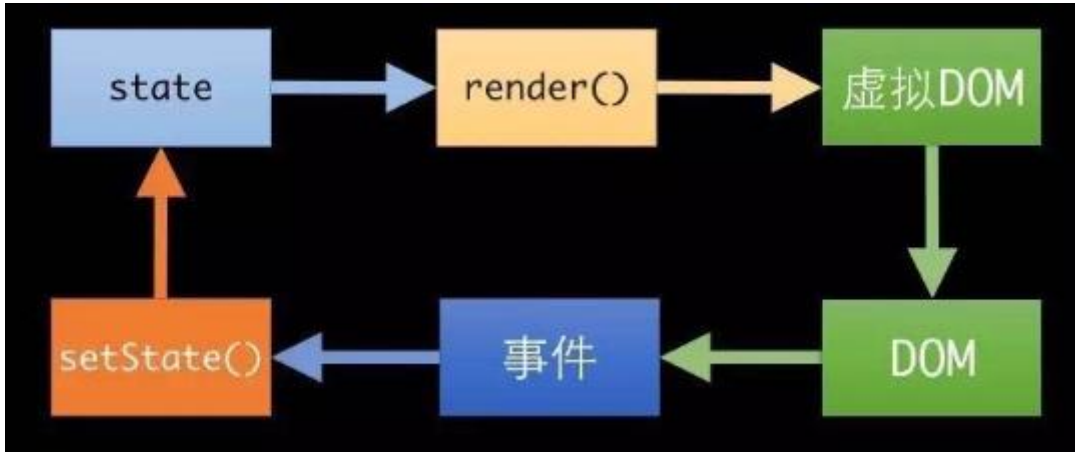
Vue的双向绑定

- Object.defineProperty() 可以为对象定义一个新的属性，或者修改原有属性的值。

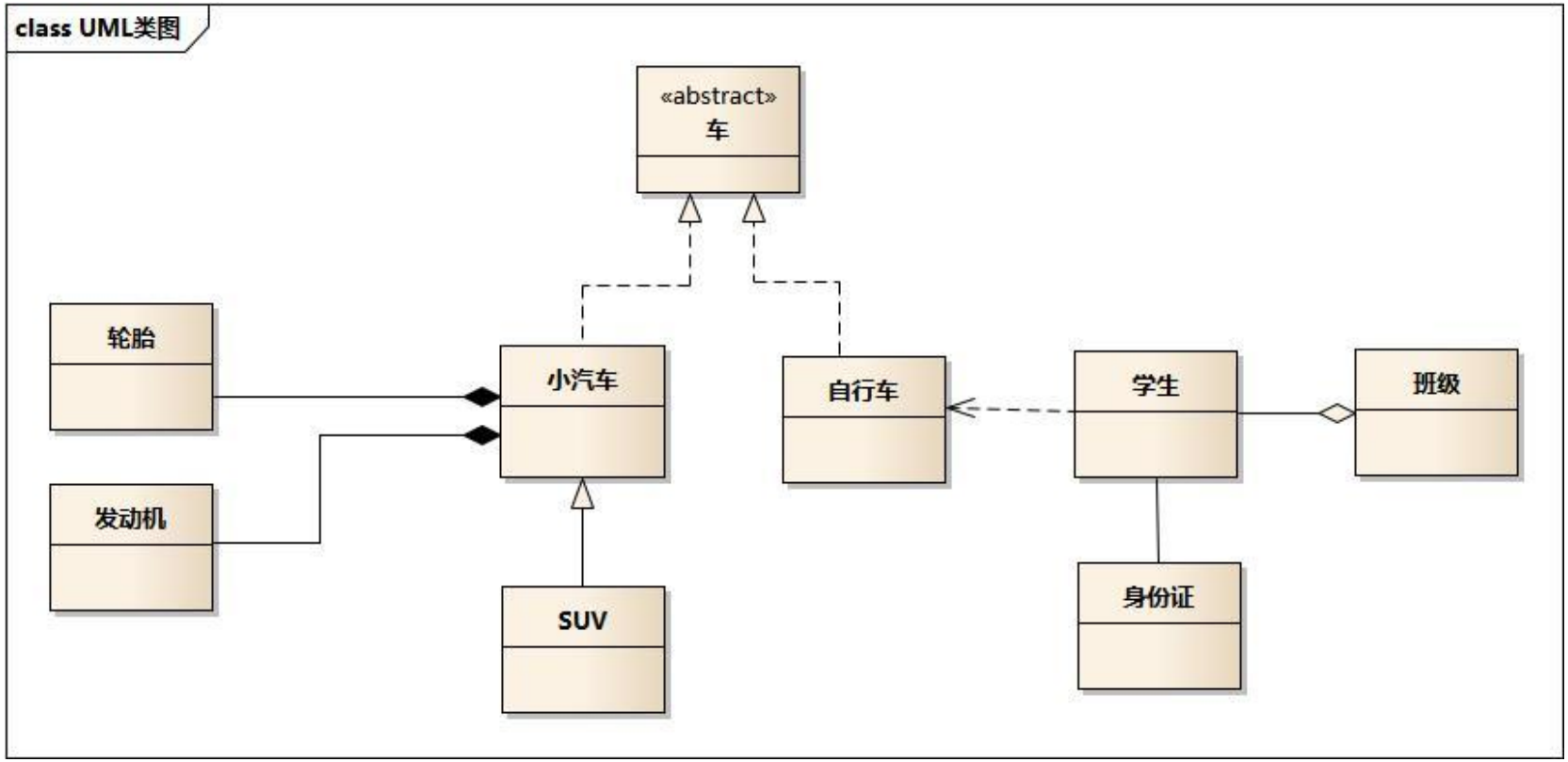
1. Vue对特定格式的数据，劫持，并生成了getter/setter方法；
2. 每个组件实例都有相应的 watcher 实例对象，它会在组件渲染的过程中把属性记录为依赖；
3. 当依赖项的 setter 被调用时，会通知 watcher 重新计算，从而致使它关联的组件得以更新。

React的单向绑定

- React一般情况下只负责渲染界面



附录：UML图解



欢迎批评指正!