

疯狂的猫：微信小程序入门系列《五》：数据绑定

作者：疯狂的猫

原文地址：<http://www.cnblogs.com/nosqlcoco/p/5987864.html>

前几篇讲述微信小程序开发工具使用、生命周期和事件。

本次讲述微信小程序**数据和视图绑定**

>>>数据视图绑定

做前端开发的同学，尤其是 WEB 前端，每天都要跟视图打交道，假如你是用过 jQuery，你就能体会到 jQuery 的代码冗余和操作不便性，需要手动管理视图和对象的数据一致性。

以下数据和对象等同。

传统的视图和数据绑定



那么微信小程序是通过什么方法来管理视图和对象绑定的呢？状态模式-单向数据流。

状态模式定义一个对象，这个对象可以通过管理其状态从而使得应用程序作出相应的变化。



简单的讲，对象状态化，只要对象状态发送变化，就通知页面更新视图元素。

三个节奏：

1. 识别哪个 UI 元素被绑定了相应的对象。
2. 监视对象状态的变化。
3. 将所有变化传播到绑定的视图上。

注意数据流向是单向的，即视图变化不会影响对象状态。

```
<view> {{ message }} </view>
```

```
Page({  
  
  data: {  
  
    message: 'Hello MINA!'  
  
  }  
  
})
```

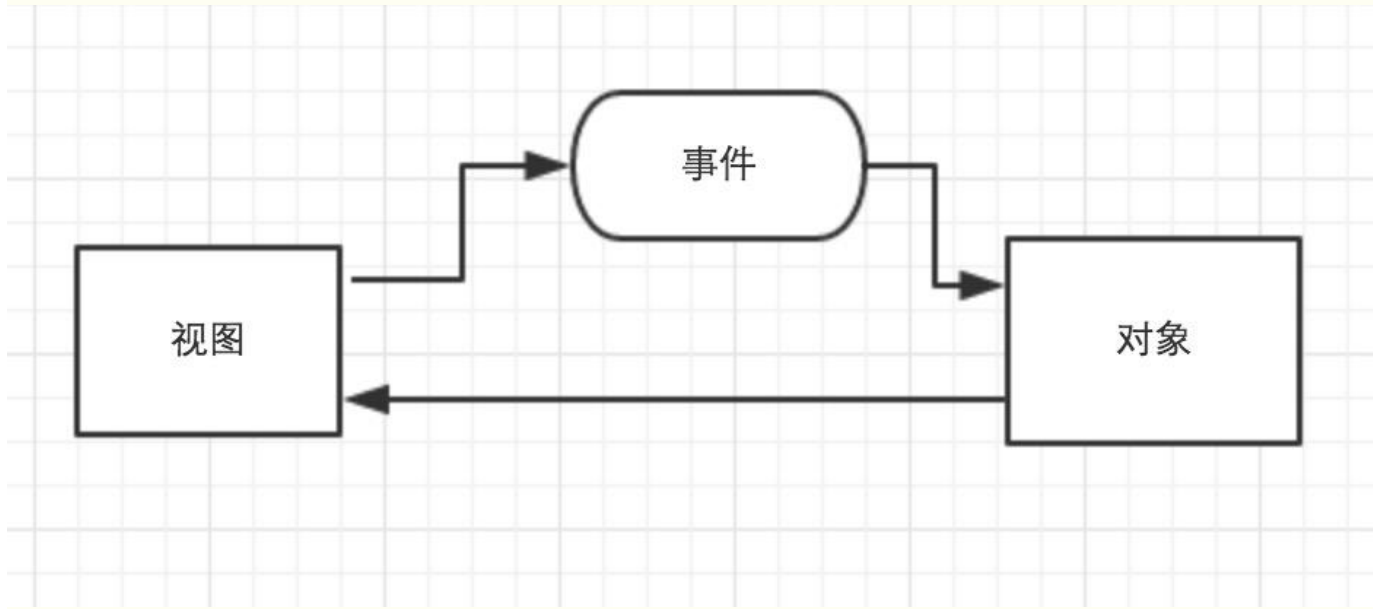
就这么简单完成视图跟数据的绑定。

仅仅通过数据更新视图是不够的，用户操作引起视图更新，数据怎么同步呢？

这里要区分的是，用户触发事件不仅要考虑当前 UI 元素更新，还会通过当前元素更新其他视图。

所以视图上的数据都必须用过事件传递给对象，只有用户操作视图，才能获取到数据，并更新对象状态。

如下图：

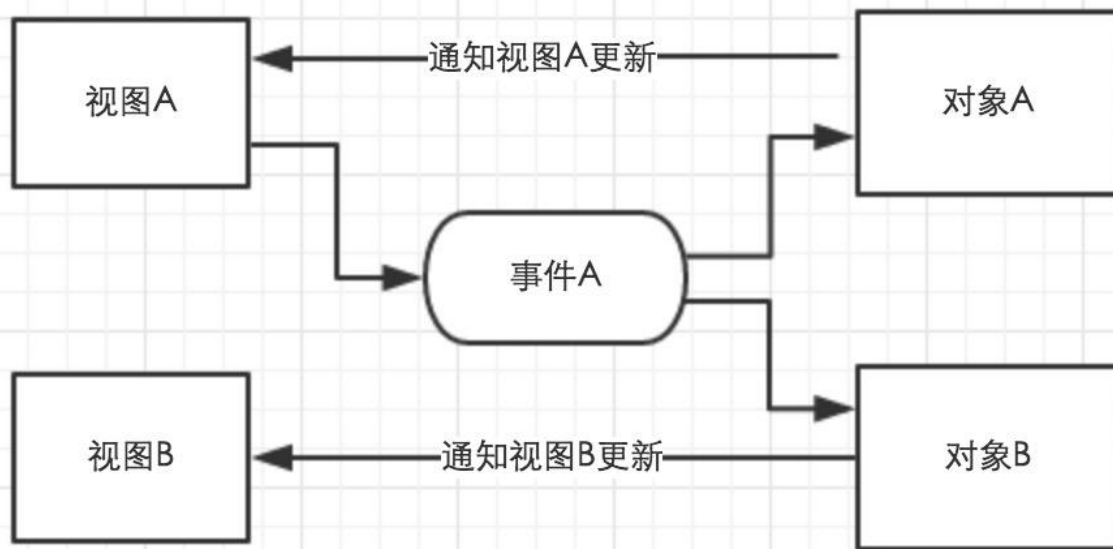


什么是『事件』：

事件是视图层到逻辑层的通讯方式。

想知道为什么的童鞋可以了解数据单双向流，这里不介绍了。

再来看视图与视图之间是如何影响？



流程说明：

1. 视图 A 由于用户操作，触发事件 A
2. 事件 A 处理函数中，更新对象 A 和对象 B 的状态
3. 由于对象 A 和 B 状态变化，通知视图 A 和 B 更新

我们以用户登录为例，用户点击(事件 A)登录按钮后，把按钮变成禁用不可点(视图 A)，同时弹出等待框(视图 B)。

部分代码如下：

1	<view>
---	--------

```
2      <loading hidden="{{loadingHidden}}">正在登录...</loading>
3      <button type="primary" size="default" disabled="{{disabled}}" bindtap="loginBtn">数据请求</button>
4  </view>
```

```
1  Page({
2
3      data: {
4          disabled: false,
5
6          loadingHidden: true
7      },
8
9      //按钮事件
10     loginBtn: function(event) {
```

```
11
12    //禁用按钮
13
14    this.setData({disabled:  true});
15
16    //弹出正在登录框
17
18    this.setData({loadingHidden:  false});
19
20    }
21  })
22
23
24
25
```

总结：

现在流行数据单双向绑定，小程序使用了单向数据流，如果采用传统的 jQuery 方式操作数据和视图，开发效率低，开发者不买账。如果采用双向数据流，程序执行效率偏低，而且是逻辑层对象状态不可控。

总体来说，小程序数据视图单向绑定开发模式，让开发者专注于事件处理上，改变对象状态，实现视图更新。