### 一、Vuejs介绍

1、Vue.js 是什么

vue 是一套用于构建用户界面的**渐进式框架，避免全家桶式的安装；**

vue被设计为**自底向上逐层应用**，先构建基础组件，逐层向上调用；

vue的核心库**只关注视图层**，便于与**第三方库**或**已有项目整合**；

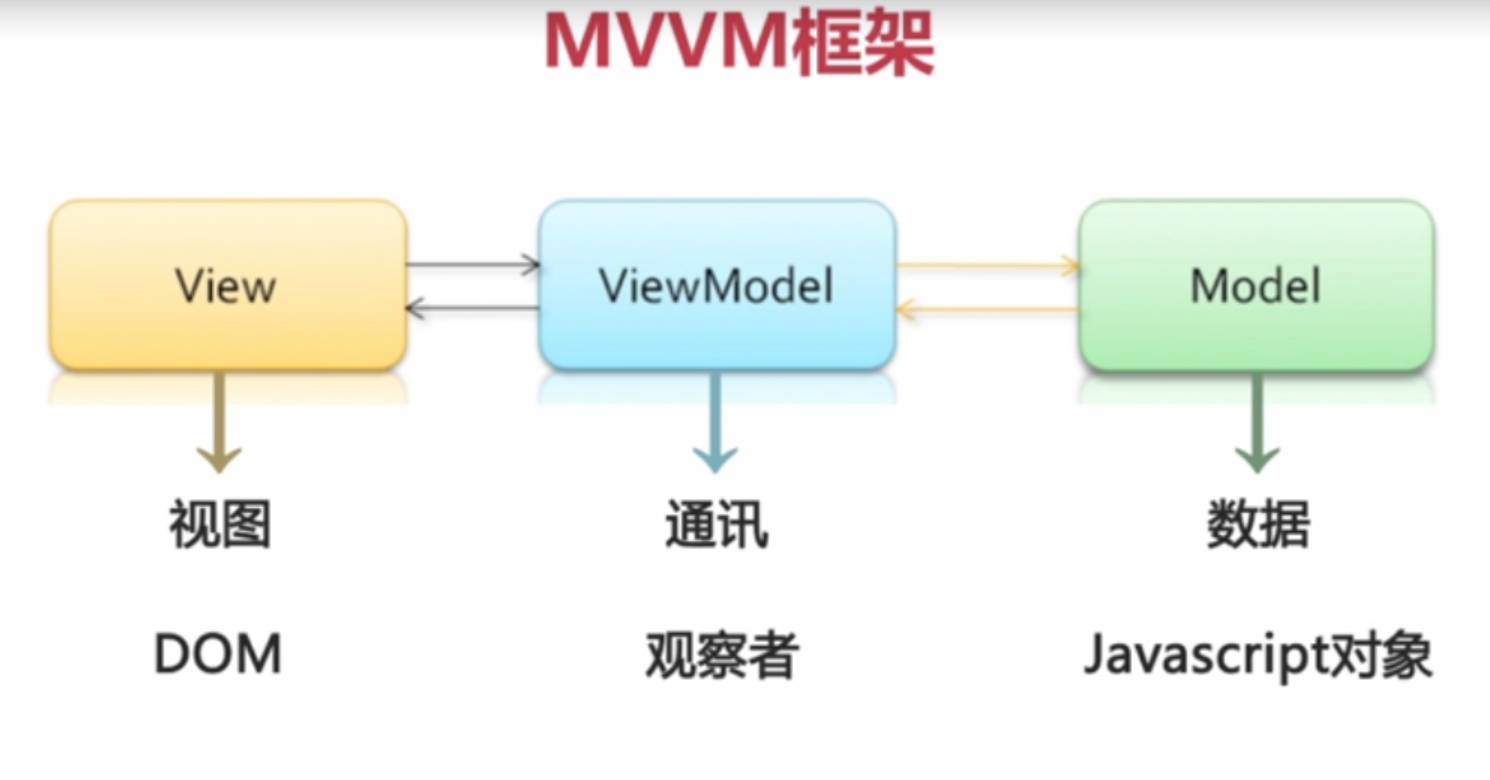
vue与其他类库结合使用时，完全能够为复杂的单页面应用提供驱动；

2、vue的本质以及核心思想

2.1本质：是一个轻量级的MVVM框架(gzip只有20k+)；

MVVM是Model-View-ViewModel的简写，它本质上就是**MVC的改进版**。MVVM 就是将View的状态和行为抽象化，将视图UI和业务逻辑分开，ViewModel担任view和model直接的通讯交互，起到监听控制作用。View绑定到ViewModel，然后执行一些命令再向model发起请求动作，而反过来，ViewModel跟Model通讯，告诉它更新来响应UI。

如下图所示：View(dom)和model(js),不能直接通讯，需要ViewModel(监听)进行交互；



2.2核心思想**：**数据驱动+组件化的前端开发，实现了数据的双向绑定；

**2.2.1数据驱动：**

数据驱动的本质，dom是数据的一种自然映射，后台返回的数据最终都要在界面上用dom元素展示出来，而数据的变化其实就是对前端界面dom元素产生变动；

**传统做法：**

使用ajax异步获取数据，然后手动触发dom的改变，前端交互改变数据时，对应着改变dom，同时也需要手动来触发dom的改变；

**Vue的做法：**

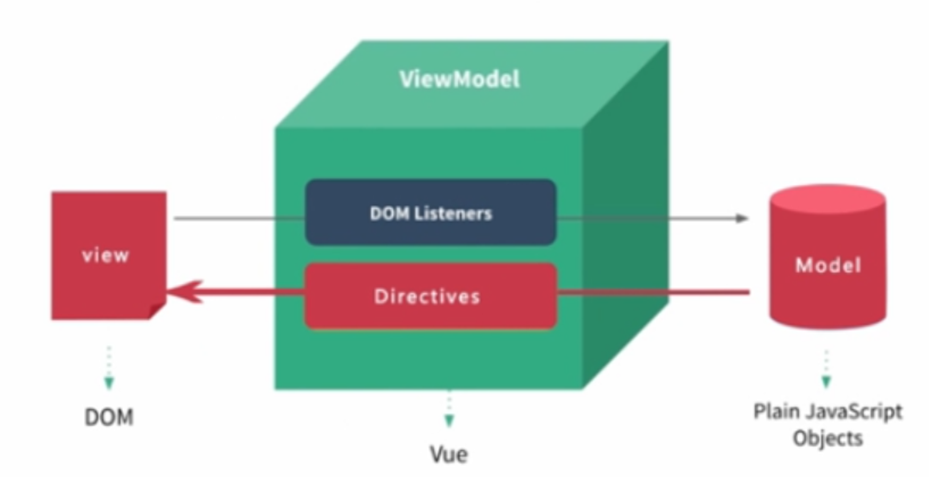
**不用手动改变dom对象，用户只需要改变数据model（js），Vue通过directives指令去修改dom；**

详细Vue指令的文档介绍地址：<https://cn.vuejs.org/v2/guide/>；

如下图所示，以v-if为例简单说明，vue如何通过指令去自动修改dom元素；



如下图所示：Model通过directive指令改变dom的原理；

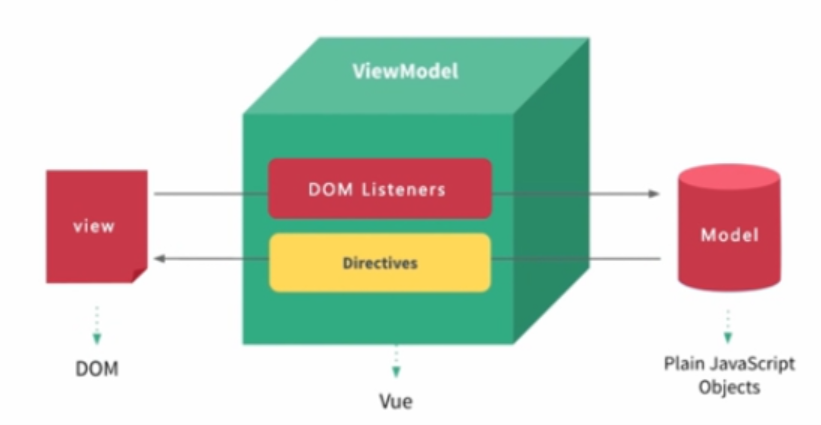


Vue对界面的操作进行监听，当操作视图view时，DOM Listeners监听到变更后，也会对应的改变model；

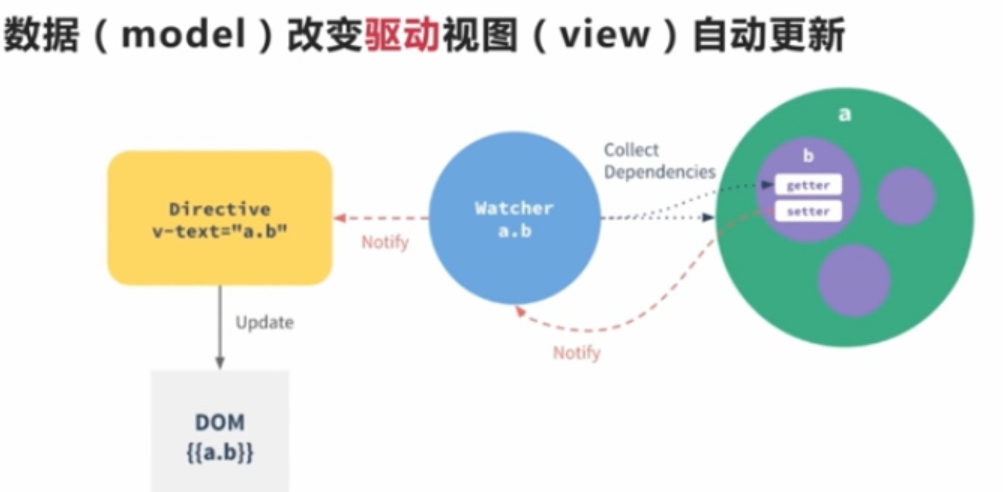
如下图所示：用 v-on 指令添加一个事件监听器，通过它调用在 Vue 实例中定义的方法：



如下图所示：view通过DOM Listeners监听改变dom的原理；



**数据响应原理**：数据被变动时，会从新计算求值，wather 监听到值变化了，就会发出指令给directive，然后directive去更新dom操作；



**2.2.2组件化：**

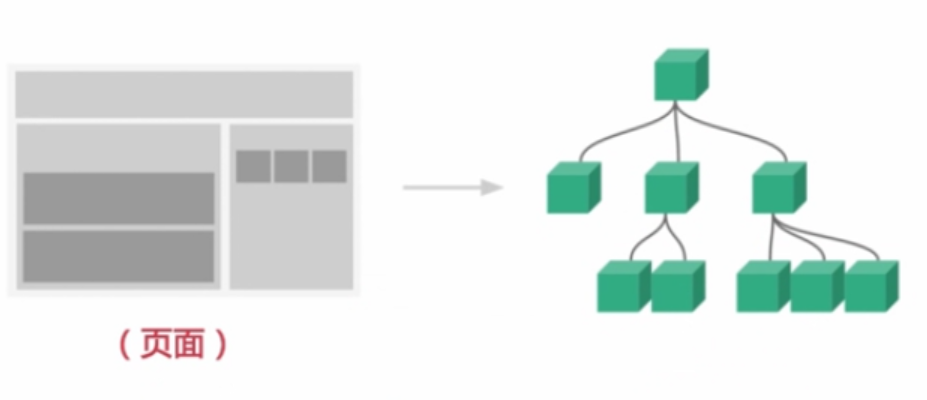
**组件设计原则：**体现设计思想是“**自底向上逐层应用**”；

页面上每个独立可视、可交互区域，都是一个组件；

页面是组件的容器，组件可以嵌套自由组合形成完整的页面；

每个组件对应一个工程目录，组件所需要的各种资源在这个目录下就近维护；

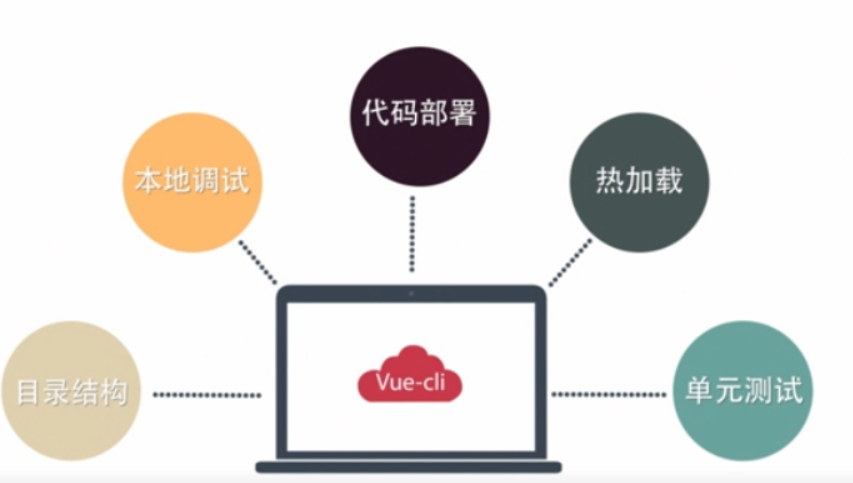
如图所示：view界面组件化拆分：



### 二、vue-cli介绍：

1、**Vue-cli是vue的脚手架工具**，帮助我们生成vue的基础代码框架，规范vue代码编写；

下图所示，vue-cli可以帮我们做这些事情：



1. **vue-cli的安装步骤；**

2.1首先我们安装需要如下工具：

1. 首先安装node.js：因为vue-cli是一个node包，需要使用node的npm包管理器。官网下载node安装包，下载地址为：<http://nodejs.cn/download/>；
2. 安装webpack：构建项目的架构模板，命令行为：npm install webpack –g；
3. 安装vue-cli脚手架：命令行：npm install vue-cli –g；

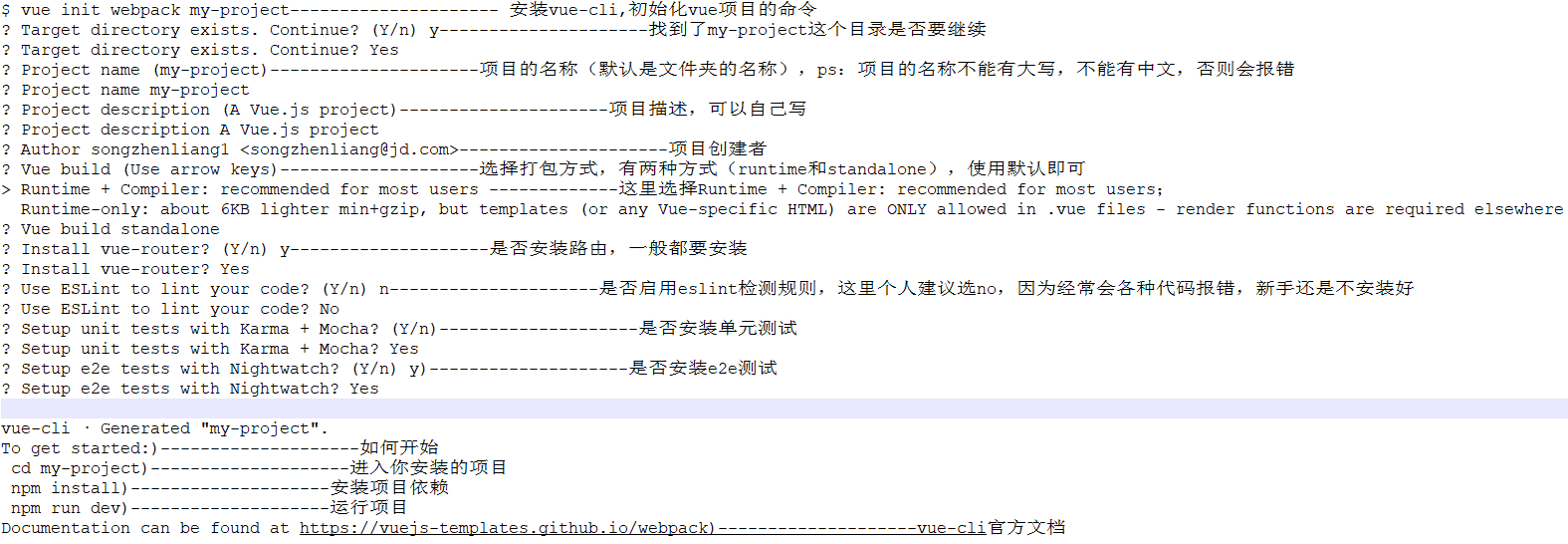
2.2使用vue-cli搭建vue项目：

1、新建一个my-project(项目名)文件夹来存放项目：

命令行：vue init webpack my-project (项目名)；

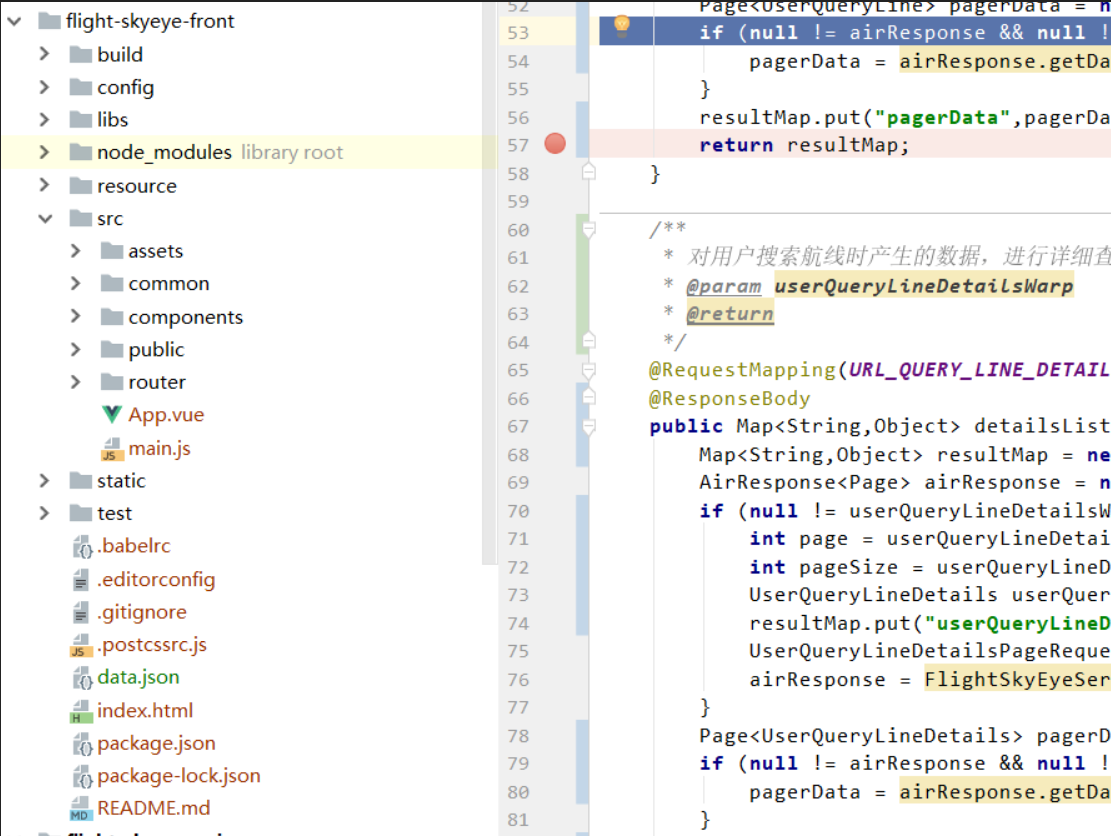
**注：**项目名称不能大写，不能使用中文 ；这个命令的意思是**初始化一个vue项目**，其中webpack是构建项目的架构模板，也就是整个项目是**基于webpack**的。其中my-project是整个项目文件夹的名称，这个**文件夹会自动生成在你指定的目录**中；

2、vue-cli初始化项目选项配置详细解析：



项目已经初步初始化完成了，里面也有一些文件，但是现在还不能成功运行。需要安装项目依赖包，执行命令行：npm install；项目初始化的时候，已经存在了package.json文件，直接使用npm install 安装项目所需要的依赖，否则项目不能正确运行。

3、使用vue-cli脚手架工具，打包生成的前端项目架构（天眼的mis前端代码结构）



一级目录简介：

Build和config 目录是webpack的配置相关文件；

node\_modules 目录存放依赖的代码库，执行npm install命令自动添加依赖；

src 目录存放编写的源码，我们开发的编写的代码都在这里；

static 目录存放第三方静态资源,.gitkeep保证文件夹为空时也能将文件夹打包的项目中；

二级目录简介：属于配置文件

.babelrc 因部分浏览器不支持ES6语法，将ES6转换为ES5的语法；

editorconfig 编译器的配制，配置代码的一些缩进风格，编码格式等；

eslintignore 忽略某些文件下的语法检查，里面配置需要忽略检查的文件目录；

eslintrc.js 是eslint的配制文件，extends和rules属性继承已定义或自定义的一些编码开发的规则；

gitignore 打包是忽略的文件配置；

index.html 访问的入口文件，webpack编译的时候回自动插入到此文件中；

package.json 用来描述一个项目，script属性配置命令行脚本，dependencies配置版本依赖，

devdependenciespeizhi 编译过程中的依赖；

README.md 是对项目描述的文件；

4、启动项目：

我们已经成功使用vue-cli初始化了一个vue项目，在my-project目录运行，命令行：npm run dev，启动服务成功后浏览器默认打开一个欢迎页面，如下图所示：

到此使用vue-cli脚手架搭建vue项目完毕，后续可以开始进行我们的vue开发之旅了；

