# 面试题精选 第五篇 MVC框架和NodeJS

1. 什么是Node.js？它有什么特性？适合做什么？

Node.js是用Chrome浏览器的V8引擎移植出来做的平台，允许我们使用JavaScript开发服务器。Node.js有强大的读取文件、生成文件的能力，所以现在很多工程化工具比如webpack、gulp都是NodeJS程序。特点就是单线程（single-tread）、事件驱动（event-driven）、非阻塞I/O（none-blocking I/O），使用事件环来调度事件。性能非常好。

NodeJS适合做IO多的业务，比如表单处理程序，本地的gulp、webpack都是I/O多的程序。

1. 怎么用Node.js写一个最简单的HTTP服务？

Node.js中提供一个http的模块，有一个createServer()的方法创建一个server对象，方法里面有res、req实参对象。可以根据res.url的不同来res.write()不同的内容。可以用server的listen来监听某一个端口。

1. 你用过的Node.js模块有哪些？

http、fs、url、path、querystring、crypto、util

1. Node.js中Express是什么？为什么要用Express？它有哪些基本用法？

Express是简化Node.js开发web应用的。提供非常清晰的路由中间件功能，与后台模板jade（现在改名了叫做pug）或者ejs结合很好，还有很多插件比如express-session等，最重要的这个框架非常方便我们做MVC架构。路由表就是控制器，数据库的对象就是Model，视图就是View。非常方便制作RESTful风格路由。

使用方法，首先罗列路由表，有get、post、use、all方法。此时路由到一个控制器函数，控制器函数会读取数据库，比如根据id读取这个新闻的内容等等，通过回调函数返回控制器，然后呈递res.render()一个视图。

1. 什么叫RESTful API？

就是说老掉牙的URL设计是这样的：

比如要读取新闻，就是readNews.php?id=10001

比如要增加新闻，就是addNews.php?id=10001

比如要删除新闻，就是removeNews?id=10001

比如要修改新闻，就是 updateNews?id=10001

在RESTful路由下，一切对10001号的新闻的操作都是：

http://127.0.0.1/news/10001

根据请求的类型不同来决定做什么事情。

比如POST请求，那么就是修改、增加新闻。如果是DEL请求，此时就是删除新闻。如果是GET请求，就是得到新闻信息。

知乎、豆瓣都是RESTful 路由的。

1. NoSQL是什么？和典型的PHP+MySQL有什么不同？

非关系型数据库，和传统SQL（瑟儿kiu）相比不需要设置行、列，每个条目可以有不同的字段。非常方便记录大数据，但是不利于检索，功能少，不能做主从查找。我一般用MongoDB，如果公司redis，我也能学。

Node.js和MongoDB配合开发的时候，一般都是异步的，数据库通过回调函数传给控制器。但是PHP+MySQL一般都是同步的。

1. npm是什么？

node package managent，就是NodeJS包管理器，是一个nodejs软件平台。所有的开发者都可以开发自己的项目，然后发布到的npm平台上，其他的开发者就能够通过npm install空格包名字命令加载这个项目，每个项目都有自己的依赖，写在package.json中，只需要通过npm install就可以安装所有的依赖。这样的话开发的时候，就可以大量使用他人的成熟项目，比如express、gm、excel等等第三方模块，我们可以把精力放到自己的业务上。

npm还有淘宝镜像cnpm。 可以安装工程化的软件，gulp、webpack都可以从npm上下载。

1. Angular是什么？通常用来做什么？

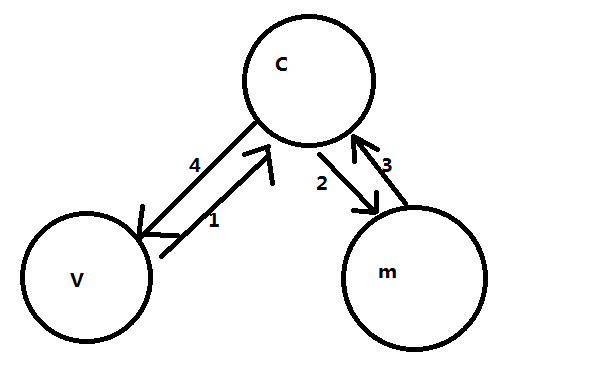
我理解的Google的Angular是一个HTML大补丁，通过给HTML标签添加指令让HTML更智能，我们也能够自己定义指令。同时Angular自己的丰富的MVC架构和双向数据绑定，彻底颠覆了我们的开发思维，不是DOM思维了，而是数据思维了。数据一变化，DOM就能变化。Angular中天生有MVC架构，controller就是控制器C，service就是M，视图就是HTML页面里面可以写双大括号自己的表达式。

Angular通常用来做单页面应用，SPA。

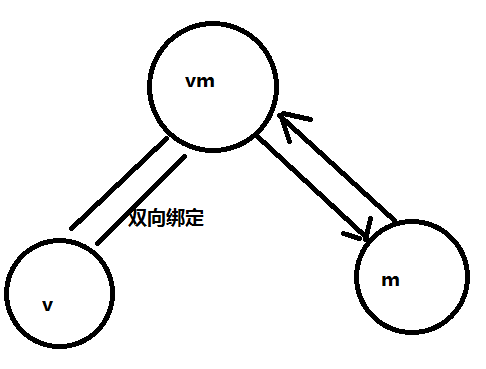
1. 请说说MVC？MVVM？MV\*？

MVC是一种设计架构，减少程序耦合性的。M就是model模型，V是view视图，C是controller控制器。Model（模型）是应用程序中用于处理应用程序数据逻辑的部分。通常模型对象负责在数据库中存取数据。View（视图）是应用程序中处理数据显示的部分。通常视图是依据模型数据创建的。Controller（控制器）是应用程序中处理用户交互的部分。通常控制器负责从视图读取数据，控制用户输入，并向模型发送数据。

主要流程是：控制器负责接收用户的响应，去找M要数据，M经过数据库查询或者Ajax调用接口之后，返回控制器，控制器呈递视图。



MVC的不好的地方就是控制器和视图有明显的发送命令的感觉，控制器必须显著的发送命令让视图改变。但是Angular中，控制器和视图感觉“粘在一起”的。控制器中的变量一变化，视图中的{{a}}就会立即改变，而不需要有明显的发送命令的感觉。所以叫做MVVM，VM就是view-model是view和model的粘合剂替代控制器。表述方法很多，称为MV\*。



1. Angular中什么叫做双向数据绑定？使用什么完成双向数据绑定？

Angualr的哲学是不需要操作DOM，也就是说不需要操作视图。只要数据变化了，此时视图就会变化。返回来视图上面比如有表单元素值被用户改变了，此时控制器中的数据也变化了，与这个数据有关系的其他视图元素也就变化了。这就是双向绑定。

双向数据绑定只能在表单元素上，只用ng-model指令即可。比如：<input type="text" ng-model="indexctrl.a" />

1. 说说Angular中的 Scope

Angular1.3之前使用$scope为ng-controller指令标签里面设置作用域，但是1.3之后就是用Controller as 的语法，用控制器的实例来表示作用域了。

标签：

<div ng-controller="MainCtrl as mc1">

{{ a }}

</div>

仅仅在这个div中有定义，其他div中没有这个控制器的实例的。

JS程序：

myapp.controller("MainCtrl",function($scope){

$scope.a = 100;

});

$scope就表示HTML中那个携带ng-controller的那个div实例。

1. Angular中的控制器和服务有什么关系？

脑子中一定要有形象，有代码的印象。

控制器和服务以及指令、路由叫做Angular中的四大金刚，都是app直接打点调用。

app.controller()

app.router()

app.factory()

app.directive()

控制器负责数据的整合、逻辑分析，通过双向绑定更改视图的程序，服务通常写与服务器进行数据交换的代码。服务是单例模式，不管在任何控制器中使用同一个服务，使用的是同一个对象。所以控制器之间传输数据，通常使用服务。

控制器一般是将服务注入，在内部使用服务暴露的方法。控制器生命周期切换频繁，所以不方便做数据的持久，而通常使用服务。比如选项卡，第一个选项卡改变了数据，切换到第二个再切换回来，控制器实例被销毁又创建了，之前的数据不能保存，所以必须借助服务来传递数据。

1. Angular指令是什么？可以用来做什么？

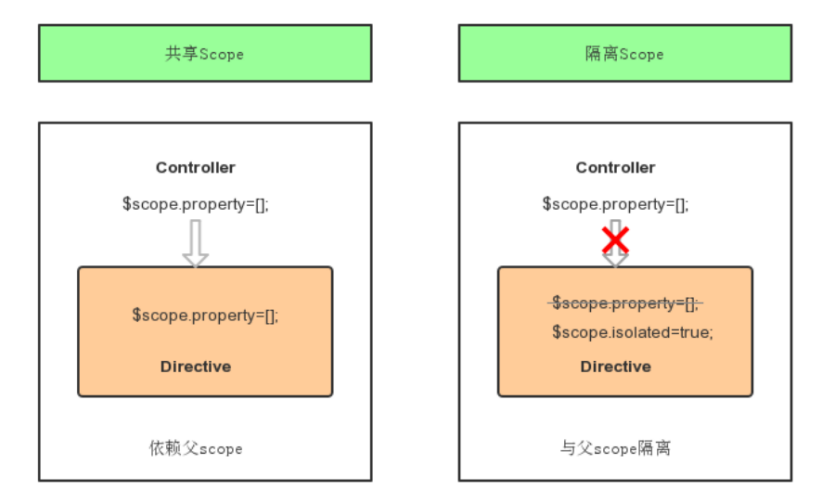
Angular中的指令就是HTML大补丁，让HTML标签有灵性、变得聪明。比如ng-repeat、ng-style、ng-show、ng-disabled等等。可以让HTML标签变得智能。

我们也可以定义自己的指令，可以定义ACME四个级别的指令，让指令可以是属性、类名、注释、标签名，此时E级别的指令就是我们在React、Vue中的模块。

指令用app.directive()来定义，常用的属性有restrict、template、templateURL、scope。scope中用=、@符号来将父元素的值传入组件内部。

1. Angular组件如何实现？子组件的作用域中有父组件的作用域，这种说法对么？

Angular中的组件就是指令，是E级别的指令，将指令的restrict属性设置为E即可。scope值如果是对象，就表示自己的内部是一个隔离作用域，此时里面无关。此时用scope对象的=、@符号，就可以将父组件的值传入子组件，甚至可以双向绑定。如果不写scope此时就是开放作用域。



1. 依赖注入是什么？

Dependencie Injection。有一些控制器需要使用某个服务，此时就需要依赖注入，表示这个控制器需要使用你。Angular的依赖注入是惰性的，只有在第一次使用的时候会加载这个文件。依赖注入是一个在组件中给出的替代了硬的组件内的编码它们的依赖关系的软件设计模式。这减轻一个组成部分，从定位的依赖，依赖配置。这有助于使组件可重用，维护和测试。

1. Angular可以使用webpack构建打包么？

**构建不需要使用webpack，但是可以使用webpack合并文件、压缩png格式图片为base64、混淆css等等，webpack有一些gulp的功能。但是，如果需要用webpack构建Angular是不行的，因为webpack是CMD规范的，而Angular天生有依赖注入的模式，自己带有依赖管理系统，自己是自己的AMD。**

**Angular诞生于2009年，webpack是2012年。Angular1.x代还没有那种webpack构建的思维。我们在Angular中如果要把服务写在外面的js文件中，不是通过import进来的，而是通过angular-async-loader来加载**

**但是Angular2取消了依赖注入系统，此时就可以用webpack编译了。**

**也就是说，依赖注入这个模式，和webpack的CMD是相悖的。**

1. Angular中路由是什么意思？如何实现？

单页面应用使用/# 叫做hash路由，有ngRoute内置的，还有插件ui-router。

ui-router需要配置

myapp.config(["$stateProvider","$urlRouterProvider",function($stateProvider,$urlRouterProvider){

//如果输入了路由表中不存在的URL，那么就跳转到根

$urlRouterProvider.otherwise("/");

//状态表

$stateProvider.state("index",{

"url" : "/",

"views" : {

"leftView" : {

"template" : "我是广告"

},

"rightView" : {

"template" : "<h1>欢迎使用考拉娱乐咨询APP</h1>",

"controller" : "IndexCtrl as indexctrl"

}

},

});

$stateProvider.state("movie",{

"url" : "/movie",

"views" : {

"leftView" : {

"template" : "我是广告"

},

"rightView" : {

"templateUrl" : "views/movie.html",

"controller" : "MovieCtrl as moviectrl"

}

}

});

$stateProvider.state("movie.detail",{

"url" : "/:id",

"views" : {

"leftView" : {

"template" : "我是广告"

},

"detail\_view" : {

"template" :

'{{ moviedetailctrl.detail.name }}剧情简介：' +

'<p>{{ moviedetailctrl.detail.jianjie }}</p>' ,

"controller" : "MovieDetailCtrl as moviedetailctrl"

}

}

});

$stateProvider.state("music",{

"url" : "/music",

"views" : {

"leftView" : {

"template" : "我是广告"

},

"rightView" : {

"template" : "<h1>音乐频道</h1>",

"controller" : "MusicCtrl as musicctrl"

}

}

})

}]);

1. Angular2最大的改变是什么？

Angualr2中使用typescript进行开发，用webpack + babel的方式编译为普通JavaScript。Angular2中去掉了Angular1中的依赖注入，变为CMD模式。创建组件更简单了。

@Component({

templateUrl : ""

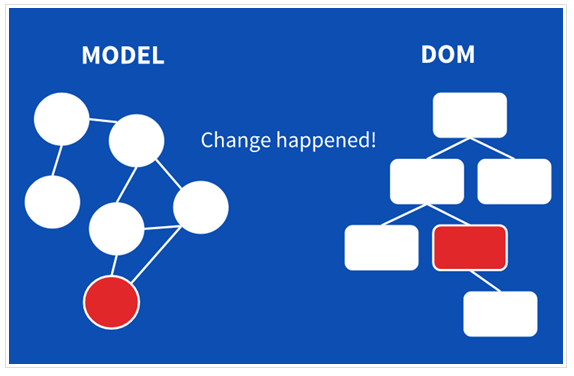
})

class Yinyue{

}

1. Angular2和Angular1哪个渲染效率更高？

Angular2中有一个机制，叫做“脏检查”。片面的说脏检查是对比当前的数据和曾经的数据是否发生改变。angular2智能从发现数据的变化到找到变化的点到更新DOM的整个过程。也就是说这里所说的脏值检查是Viewmodel与view层的那座桥梁。也就是说，脏检查可以尽可能小的缩小DOM更改的范围。



1. React是什么？

React来自Facebook，一个做用户界面组件（UI-Component）的框架。它自己不是MVC框架，可以结合Redux来形成一个系统架构。

React中非常简单可以创建一个组件，然后组件可以相互嵌套，数据可以传输。使用jsx语法实现模板，让HTML和JS不分家。

基本API说说：

ReactDOM.render(

<h1>Hello, world!</h1>,

document.getElementById('example')

);

1. React和Angular有什么相同点和不同点？（每个说三点）

相同点：

Angular和React+Redux都可以做分层开发，就是模块化开发。

Angular 和 React 相同的是 Model Driven View 的套路。

Angular和React都有双向数据绑定，哲学都是只需要操作数据，不需要操作DOM。

都有组件的生命周期、组件可以嵌套，组件通过属性来接收数据。

使用 Virtual DOM

不同点：

Angular重一些，React很轻。React效率更高。

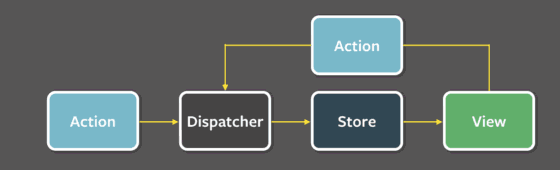
React不是框架，angular 是MV\* 框架

angular 是最适合CRUD的SPA 单页面的应用程序，而React并不局限在SPA上，任何页面都可以用React开发组件。

Angular创建组件更麻烦，React简单一些。

1. 你知道Flux架构么？它的核心思想是什么？React技术栈中的实现是？

　Flux是Facebook用户建立客户端Web应用的前端架构， 它通过利用一个单向的数据流补充了React的组合视图组件，这更是一种模式而非正式框架。



用户的操作反应在View上，并不是直接更改数据，而是让Dispatcher去更改Store，Store改变之后影响view。

React中的Flux架构，叫做Redux。

1. 请介绍React组件的生命周期

每个组件从诞生到消亡有很多具体的过程。

**getInitialState**作用于组件的实例，在实例创建时调用一次，用于初始化每个实例的state，此时可以访问this.props。

**componentWillMount** 在完成首次渲染之前调用

**componentDidMount**  真实的DOM被渲染出来后调用

**componentWillUpdate** 接收到新的props或者state后，进行渲染之前调用

**componentDidUpdate** 完成渲染新的props或者state后调用

**componentWillUnmount** 组件被移除之前被调用，可以用于做一些清理工作

1. JSX是什么？如何将JSX变为js的？你觉得JSX有什么优点和缺点？

JSX是用一个语法糖，让我们在JS中可以不书写引号直接写HTML结构。比如：

ReactDOM.render(

<div><h1></h1></div>

,

document.getElementById("box")

)

通过babel-loader让jsx变 为js的。babel就两个功能：翻译ES2016、翻译JSX。

JSX基本语法规则，遇到HTML标签（以<开头），就用HTML规则解析；遇到代码块（以{开头），就用JS规则解析。

优点：

语境统一，不用来回切换文件。

JSX中使用js语法，使用{}形式

可以内嵌一些JS逻辑，比如this.arr.map((item)=>{return <h1>{item}</h1>})

缺点：

标签必须严格封闭，<img>就是错的。

类名必须写className，for属性写htmlFor

1. Vue是什么？最新版本是什么？

Vue.js是一套构建用户界面的渐进式框架。Vue 的目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件。

Vue最新版本是2代。

1. Vue和Angular有什么异同点？

Vue 的一些语法和 Angular 的很相似（例如 v-if vs ng-if）。据Vue开发者说因为 Angular 是 Vue 早期开发的灵感来源。然而，Augular 中存在的许多问题，在 Vue 中已经得到解决

**复杂性**

在 API 与设计两方面上 Vue.js 都比 Angular 1 简单得多，因此你可以快速地掌握它的全部特性并投入开发。

**灵活性和模块化**

Vue.js 是一个更加灵活开放的解决方案。它允许你以希望的方式组织应用程序，而不是在任何时候都必须遵循 Angular 1 制定的规则，这让 Vue 能适用于各种项目。我们知道把决定权交给你是非常必要的。

这也就是为什么我们提供 Webpack template，让你可以用几分钟，去选择是否启用高级特性，比如热模块加载、linting、CSS 提取等等。

**数据绑定**

Angular 1 使用双向绑定，Vue 在不同组件间强制使用单向数据流。这使应用中的数据流更加清晰易懂。

**指令与组件**

在 Vue 中指令和组件分得更清晰。指令只封装 DOM 操作，而组件代表一个自给自足的独立单元 —— 有自己的视图和数据逻辑。在 Angular 中两者有不少相混的地方。

**性能**

Vue 有更好的性能，并且非常非常容易优化，因为它不使用脏检查。

在 Angular 1 中，当 watcher 越来越多时会变得越来越慢，因为作用域内的每一次变化，所有 watcher 都要重新计算。并且，如果一些 watcher 触发另一个更新，脏检查循环（digest cycle）可能要运行多次。Angular 用户常常要使用深奥的技术，以解决脏检查循环的问题。有时没有简单的办法来优化有大量 watcher 的作用域。

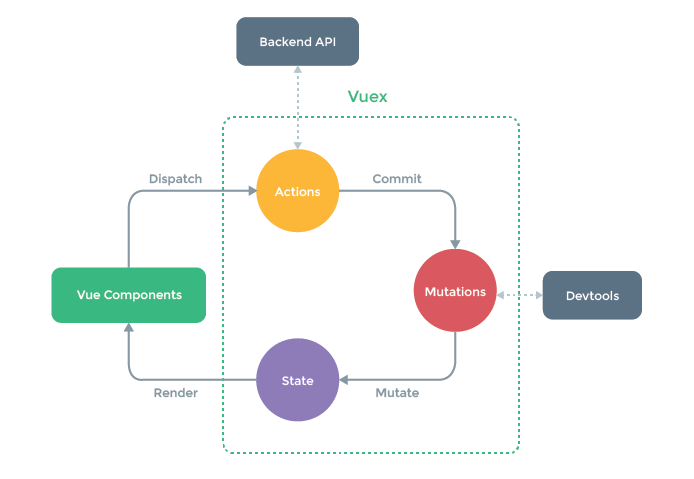
1. Vue中的模板如何实现？模板中的表达式用什么表示？

.vue文件就是模板，然后在webpack中用vue-loader进行编译。模板中有<style></style> 、<template></template>、<script></script>。模板中用{{}}来当做表达式。

1. Vuex是什么？使用Vuex可以解决什么问题？

**官方：Vuex 是一个专为 Vue.js 应用程序开发的状态管理模式。它采用集中式存储管理应用的所有组件的状态，并以相应的规则保证状态以一种可预测的方式发生变化。**

Vuex类似React中的Redux架构，数据的单向流动，有一个全局共享的state。



Vuex 可以帮助我们管理共享状态，如果要构建的是一个中大型单页应用，会考虑如何更好地在组件外部管理状态，Vuex 将会成为自然而然的选择。

**最好的回答就是我曾经用这个东西做过项目。**