

**毕业论文（设计）文献综述**

**[](http://bysj.zjgsu.edu.cn/)**

**题目：前端整合框架在商品交易系统中的应用**

**学 院：电子商务与工程管理学院**

**专 业：电子商务**

**班 级：商务16A**

**学 　 号：1612200132**

**学生姓名：王勇**

**指导教师：吴云标**

**二○ 一七 年 12月**

目录

[1.课题分析 4](#_Toc497157806)

[1. 文献范围以及检索工具 4](#_Toc497157807)

[2.检索词及检索式获取 5](#_Toc497157808)

[3.检索的过程以及结果 6](#_Toc497157809)

[4.1 维普数据库 6](#_Toc497157810)

[4.2 中国学术期刊 8](#_Toc497157811)

[4.3 EBSCO 11](#_Toc497157812)

[4.5 ScienceDirect 12](#_Toc497157813)

[4. 检索体会 13](#_Toc497157814)

[5. 文献综述案例 14](#_Toc497157815)

[一、引言 16](#_Toc497157816)

[二、传统web前端开发 16](#_Toc497157817)

[（一）开发流程 16](#_Toc497157818)

[（二）存在的问题 16](#_Toc497157819)

[（三）MVC框架 17](#_Toc497157820)

[三、React.js 17](#_Toc497157821)

[（一） 数据驱动渲染 18](#_Toc497157822)

[（二）Virtual DOM 18](#_Toc497157823)

[（三）与传统开发相比React的优势 18](#_Toc497157824)

[四、Vue.js 20](#_Toc497157825)

[（一）渐进式框架 20](#_Toc497157826)

[（二） 双向数据绑定 20](#_Toc497157827)

[五、AngularJS 21](#_Toc497157828)

[（一）MVC 21](#_Toc497157829)

[（二）双向数据绑定 21](#_Toc497157830)

[六、前端模块化和工程化 22](#_Toc497157831)

[七、参考文献 24](#_Toc497157832)

# 1.课题分析

随着互联网的发展，前端交互的场景变得越来越复杂，为了解决这些在前端开发中的痛点，在最近几年的前端领域中，优秀的框架层出不穷，比如Vue.js, AngularJS使用web前端框架搭建的单页应用（SPA）在不向服务器发起HTTP请求的情况下就可以实现路由的转发，并且按需加载所需要的文件，极大的提升了用户的体验。其中FaceBook开源React.js引起了业界极大的关注。由于FaceBook前端业务需求越来越复杂，MVC的前端框架已经满足不了其复杂的业务场景的需要，FaceBook推出了React + Flux 来解决前端复杂的交互场景。React.js采用了分而治之的方法，把一个大应用拆分几个小组件，每个组件都会完成自己需要完成的特定的功能。React.js还采用了JSX将JavaScript代码、CSS代码和HTML代码混合在一个组件中，实现了真正的组件封装。Flux实现了更加严格的单向数据流，页面上的UI只能通过派发Action改变状态从而导致组件重新渲染，阻止了View层直接和Model直接交互。正是因为React技术栈实现了前端架构的高内聚，低耦合，国内很多大型互联网公司包括阿里巴巴部分的业务也使用React进行了前端的重构。在实现后台系统和传统JQuery构建前端应用的前提下，如何通过前端框架的整合，将传统方式构建的web电子商务前端页面进行重构，实现前端的状态管理并且兼容移动端的单页应用是本次研究需要解决的问题。

# 文献范围以及检索工具

这是一个web开发相关的课题，固然需要检索下列数据库

1. 中文数据库

维普中文期刊数据库

中国学术期刊全文数据库

中国科技论文在线

1. “读秀知识库”
2. 学校图书馆实体书籍
3. 外文数据库

EBSCO

# 2.检索词及检索式获取

核心检索词：web前端、框架、MVVM、MVC、React、Vue、 Angular

其他检索词：webpack、工程化

通用的中文检索式：

S1 web 前端\*框架\*MVVM ---全面了解MVVC框架在前端的作用

SU = "web前端" and TI="MVVM"

命中数：8

S2 web 前端\*工程化\* ---全面了解工程化在前端的作用

SU = "web前端" and TI="工程化"

命中数：1

S3 web\*React.js -全面了解React.js应用的发展

SU = "web" and TI="react"

命中数：2

S5 web\*Vue.js -全面了解Vue.js应用的发展

SU = "web" and TI="Vue"

命中数：4

S6 web\*Angular.js -全面了解Angular.js应用发展

SU = "web" and TI="Angular"

命中数：1

英文检索式

检索式：("web") and("AngularJS") 命中数 5

检索式：(web)and (react js) 命中数 1

检索式：(web) and (SPA) 命中 18

检索式：（web front-end）and(framework)命中 5

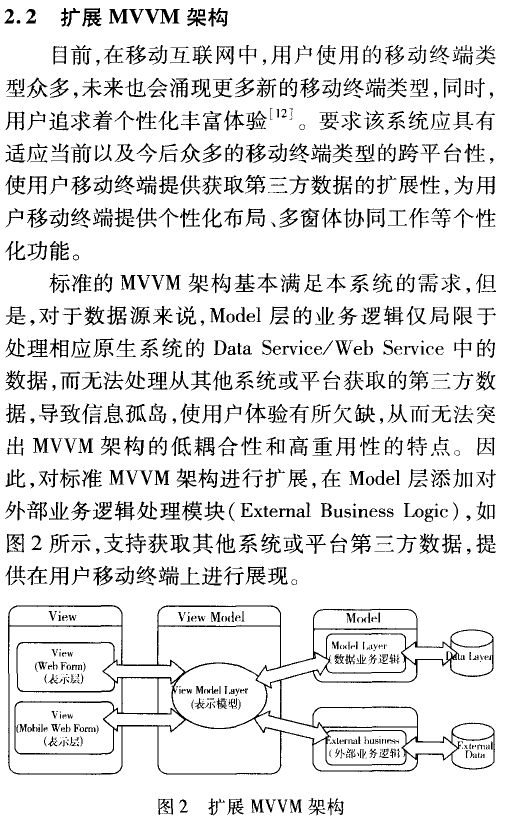
# 3.检索的过程以及结果

## 4.1 维普数据库

在检索框中检索 M= web \* (MVVM) 时间限制在2005年到2011年得到5条结果，选择了相关性比较大的2篇文章。

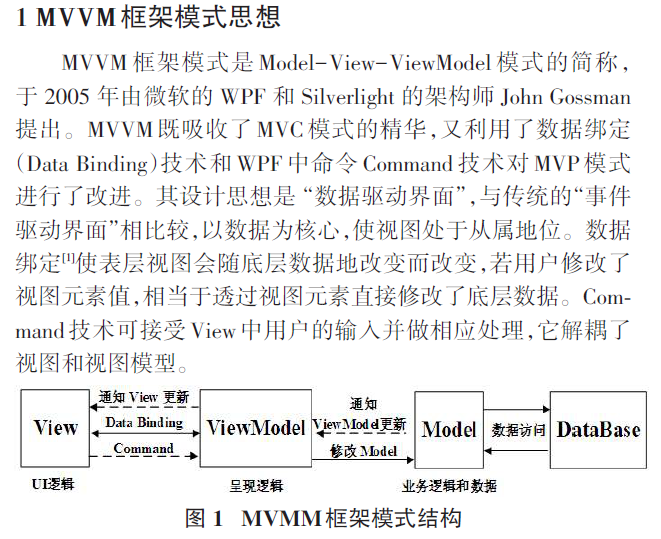
1. 封宇[1,2],陈宁江[1].[基于MVVM架构的移动Web前端展示方案](http://qikan.cqvip.com/article/detail.aspx?id=663114992&from=article_detail)[J].计算机与现代化,2014,0(11):1-4.

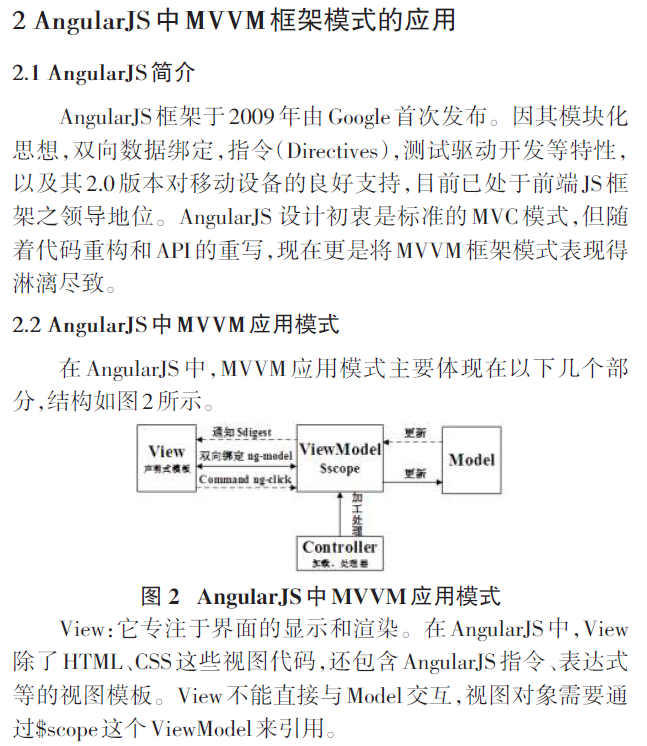
【文摘】



孙连山,李云倩.MVVM框架在Web前端的应用研究[J].电脑知识与技术：学术交流,2016,0(2X):45-46.

【文摘】





## 4.2 中国学术期刊

在检索框中检索

SU=”web” and TI =”Vue”;

命中数4，选取2个最相关的2篇文章

[1]麦冬,陈涛,梁宗湾. 轻量级响应式框架Vue.js应用分析[J]. 信息与电脑(理论版),2017,(07):58-59.

[文摘]

Vue.js 是一个非常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，着重关注MVVM 前端View 和

Model 部分，允许嵌入一个现有页面，而不一定要做成一个

庞大的单页应用。

Vue.js 并不是一个入侵式前端框架库，在配合其他库方

面它给了较大的空间。例如，Vue.js 核心默认不包含路由和

Ajax 功能，并且通常假定在应用中使用了一个模块构建系统。

而AngularJS 和ReactJS 都过于庞大和全面，两者涉及了一

个页面应用的所有阶段和所有功能，一旦使用，就需要抛弃

现有的成熟框架，否则基础功能均导致重复冗余，降低客户

的体验效果。而更换基础框架的学习和维护成本巨大。

Vue.js 具有以下优点：响应式编程；组件化；模块化。

使用Vue.js 可以去除繁琐的DOM 操作，只需要关注数

据的源头，而不用担心DOM 元素变化之后的绑定变化。这

就是数据驱动与DOM 驱动的差别。

[2]朱二华. 基于Vue.js的Web前端应用研究[J]. 科技与新,2017,(20):119-121.

[文摘]

Vue 的核心是响应式原理，把一个普通JavaScript 对象

传给Vue 实例的data 选项，同时每个Vue 实例都有相应的

watcher 实例对象。如果data 的属性发生变化，会通知watcher

重新计算，从而致使它关联的组件得以更新。Vue 异步执行

DOM 更新。只要观察到数据变化，Vue 将开启一个队列，

并缓冲在同一事件循环中发生的所有数据改变。如果同一个

watcher 被多次触发，只会一次推入到队列中，在缓冲去除

了重复数据，避免了不必要的计算和DOM 操作。Vue 的响

应为双向绑定数据，实时反映数据的真实变化，并映射到数

据源上，避免了前端页面开发中DOM 选择器繁杂的操作，简化了Web 前端开发流程，降低了开发难度，提高了前端

开发效率，缩短了开发成本和周期。

Vue 具有以下特点：①灵活开放；②易学易用，有更多

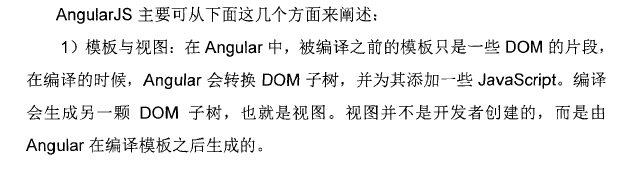
成熟工具的支持；③性能好，易优化；④有功能强大的路由

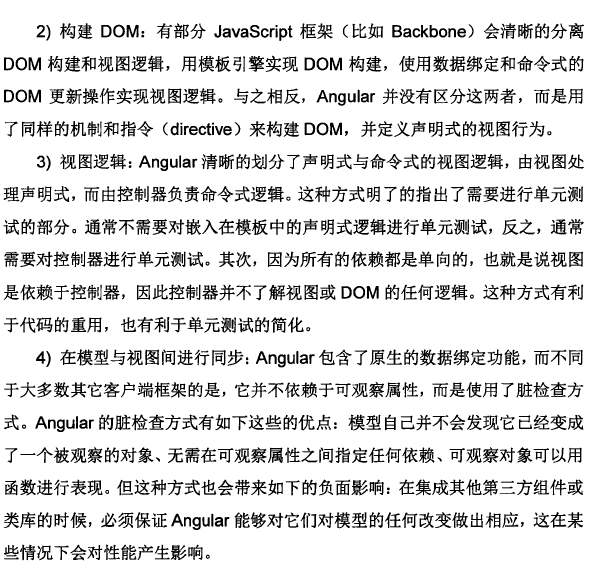
机制；⑤有多种方式实现过渡效果。

SU = "web" and TI="Angular"

[1]冯晨超. 基于AngularJS的物品和用户查询模块的设计与实现[D].南京大学,2014.

【文摘】





### 4.3 EBSCO

检索式 ：（web front-end）and(framework)

相关的文档有

[Global Journal of Enterprise Information System](http://web.a.ebscohost.com/ehost/LinkTarget/journalbSearch?sid=7f14f70e-d98a-46ea-9891-ae874362aeae@sessionmgr4006&vid=10&term=JN%2b%2522Global%2bJournal%2bof%2bEnterprise%2bInformation%2bSystem%2522). Jan-Mar2017, Vol. 9 Issue 1, p75-84. 10p.

【文摘】

Optimization of Request pipeline processing Systems: In this category we will examine all the systems involved in the web request processing pipeline. This involves browser software, CDN, proxy server, network, load balancer, web server, application server, integration middleware, backend services, database server and such

请求管道处理系统的优化：在这个类别中，我们将检查Web请求处理流程中涉及的所有系统。 这涉及浏览器软件，CDN，代理服务器，网络，负载平衡器，Web服务器，应用服务器，集成中间件，后端服务，数据库服务器等

• Client-side and server side optimization: Client-side performance optimization includes all optimizations performed on client-side presentation components such as HTML pages, images, assets and such. Server side optimization includes performance tuning of server –side components such as fine-tuning business components, setting optimal server configuration, right infrastructure sizing and such

客户端和服务器端优化：客户端性能优化包括在客户端演示组件（如HTML页面，图像，资产等）上执行的所有优化。 服务器端优化包括服务器端组件的性能调整，例如微调业务组件，设置最佳服务器配置，正确的基础架构尺寸等

• Design time and run time optimization: Design time optimization includes the static and offline performance optimizations activities such as performance code reviews, performance testing, and offline performance tuning and such. Run time and dynamic performance optimization activities include real-time performance

设计时间和运行时间优化：设计时间优化包括静态和离线性能优化活动，如性能代码评估，性能测试和离线性能调优等。 运行时间和动态性能优化活动包括实时性能

### 4.5 ScienceDirect

【文摘】

• jQuery is a popular JavaScript library3. jQuery helps in finding and manipulating the Document Object Model (DOM) elements, processing browser events, and dealing with browser incompatibilities. jQuery is an extensible library, and thousands of plugins have been created by developers from around the world.

jQuery是一个流行的JavaScript库3。 jQuery有助于查找和操作文档对象模型（DOM）元素，处理浏览器事件以及处理浏览器不兼容性。 jQuery是一个可扩展的库，并且来自世界各地的开发人员创建了数千个插件。

• AngularJS is a toolset for building the framework most suited to application development. It is fully extensible and works well with other libraries. Every feature can be modified or replaced to suit unique development workflow and feature needs 4. AngularJS is often used for creating single-page applications, where only certain portions of the page (sub-views) are updated as a result of the user‘s actions or data being sent from the server. Other features include tw-way data-binding, reusable components, deep linking, built-in services for backend communication and localization support. Mobile Angular is another User Interface (UI) framework that is built on AngularJS and Bootstrap for mobile-friendly application development.

AngularJS是一种用于构建最适合应用程序开发的框架的工具集。 它是完全可扩展的，并且与其他库一起使用。 每个功能都可以修改或更换，以适应独特的开发工作流程和功能需求。 AngularJS通常用于创建单页应用程序，其中仅由于用户的操作或数据从服务器发送而更新页面的某些部分（子视图）。 其他功能包括双向数据绑定，可重复使用的组件，深层链接，后端通信和本地化支持的内置服务。 Mobile Angular是另一个基于AngularJS和Bootstrap的用户界面（UI）框架，适用于移动设备的应用程序开发。

• Angular Angular 2 (or just Angular) is an open source JavaScript framework maintained by Google. It is an evolution of its popular predecessor, AngularJS. Apart from JavaScript, Angular applications can be developed in Dart, or TypeScript 5. The framework makes it simpler to create custom components that can be added to HTML documents and to implement application logic. Angular uses data binding extensively, includes a dependency injection module, supports modularization, and offers a routing mechanism. Whereas AngularJS was MVC-based, Angular is not. This framework doesnt include UI components.

• Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile first projects on the web6. It is an open source library of UI components developed by Twitter. The components are built using the responsive web design principles, which makes this library extremely valuable for web applications that needs to automatically adjust its layout depending on the screen resolution.

Bootstrap是最流行的HTML，CSS和JS框架，用于在web6上开发响应式，移动的第一个项目。 它是由Twitter开发的UI组件的开源库。 这些组件使用响应式Web设计原则构建，这使得该库对于需要根据屏幕分辨率自动调整其布局的Web应用程序非常有价值。

• React is an open source library by Facebook for building user interfaces. It‘s non-intrusive and can be used with any other library or a framework. React creates its own virtual DOM object, minimizing access to browser‘s DOM, which results in better performance. For content rendering, React introduces the JSX format, which is a JavaScript syntax extension that looks like XML. Using JSX is recommended but optional.

React是Facebook的开源库，用于构建用户界面8.它是非侵入性的，可以与任何其他库或框架一起使用。 React创建自己的虚拟DOM对象，最大限度地减少对浏览器DOM的访问，从而实现更好的性能。 对于内容呈现，React引入了JSX格式，这是一个JavaScript语法扩展，看起来像XML。 建议使用JSX，但可选。

• Node.js (or Node) is a framework or a library, as well as is a runtime environment .This framework can be used to develop JavaScript programs that run outside the browser. The Node.js framework includes an API to work with the filesystem, access databases, listen to HTTP requests, and more.

Node.js（或Node）是一个框架或库，也是一个运行时环境。 该框架可用于开发在浏览器之外运行的JavaScript程序。 Node.js框架包括一个可与文件系统一起使用的API，访问数据库，收听HTTP请求等。

# 4. 检索体会

由于课题定义的比较小，在检索过程中某些具体的关键词并不能搜到到与课题相关的内容。特别的如React.js 这种解决问题的具体的框架并不能直接搜索到相关的文献，但是通过web前端开发存在的问题出发，想到了用MVVM数据双向绑定的机制，解决web开发的痛点，从这个关键字进行搜索得到了很多相关的文献资料。在外文搜索的过程中，大多外文的概念都比较大没没有

# 5. 文献综述案例

**摘要：**在现代Web系统的开发的过程中，MVC模式中的View层交互功能越来越强大，业务逻辑也越来越复杂。传统的前端开发模式已经满足不了复杂的电子商务系统的业务需求的扩展与维护。为了使得前端应用的开发更加有效率，近些年出现了许多前端框架和自动化构建工具。本文在使用传统的技术方法实现电子商务系统的前端页面的提前下，使用前端整合框架React.js+ Redux + Reat-Router对该系统前端页面进行模块化重构，由此比较两种开发模式的主要区别。

**关键词:**Web前端,，React.js，Redux, 单页应用，JSX，单向数据流，电子商务系统前端，前端模块化开发

**“The Application of Front - end Integration Framework in E - commerce System”**

**Abstract:** In the development of modern Web system, interaction in view layer becomes more and more powerful and business logic is also more and more complex. The traditional front-end development model has been unable to meet the complex e-commerce system’s business needs expansion and maintenance. In order to make the development of front-end applications more efficient, in recent years there have been many front-end framework and automated building tools. In this paper, the traditional method is used to build the e-commerce system in advance, using the front-end integration framework React.js to modulate the front-end applications of the system, thus comparing the main differences between the two development models.

**Keywords:** Web front end; React.js; Redux; single-page-application; Single data stream; front-end reconstruction; front-end modular development

# 一、引言

随着互联网的发展，前端交互的场景变得越来越复杂，为了解决这些在前端开发中的痛点，在最近几年的前端领域中，优秀的框架层出不穷，比如Vue.js, AngularJS使用web前端框架搭建的单页应用（SPA）在不向服务器发起HTTP请求的情况下就可以实现路由的转发，并且按需加载所需要的文件，极大的提升了用户的体验。正是因为React技术栈实现了前端架构的高内聚，低耦合，国内很多大型互联网公司包括阿里巴巴和蚂蚁金服的部分的业务也使用React进行了前端的重构。在实现后台系统和传统JQuery构建前端应用的前提下，如何通过前端框架的整合，将传统方式构建的web电子商务前端页面进行重构，实现前端的状态管理并且兼容移动端的单页应用是本次研究需要解决的问题。

# 二、传统web前端开发

## （一）开发流程

在早期的web时代，人们使用浏览器大多都是用浏览内容，充其量填写一些表单传送到服务器，传统web应用的开发，将不同的种类的技术分门别类。放到不同的文件夹中，先编写HTML页面，确定页面的语义，再编写CSS代码实现页面布局，最后编写JavaScript实现 页面的交互。这样的分工并没有做到逻辑上的分治。只是单纯把不同类型的技术分开了而已。

## （二）存在的问题

在传统的web前端运用的开发中,开发人员常常会使用jQuery在DOM节点上绑定事件，一旦用户与界面的交互变得复杂之后，事件和元素的关系会变得纠缠不堪。导致代码难以维护，功能难以扩展，状态难以管理。而且在DOM节点绑定大量的事件很容易造成内存的泄露。

为了加强代码的复用性，在开发的过程中常常会封装大量的组件，但是组件的样式又部分在CSS文件夹中，导致了组件复用的复杂程度。而且在使用jQuery编写组件的过程中会多次操作DOM节点，严重影响性能。

View层直接和处理数据的业务逻辑交互，导致处理相同逻辑业务代码在重构代码的时候才能抽离出来，降低了代码的复用，严重影响开发的效率。在后期需求发生变化的时候，可能修改一个部分就会导致其他的bug.

但是随着前端页面的交互逻辑变得越来越复杂的时候，更多的数据处理和业务的逻辑交给了客户端去处理，可能一个网站需要维护的javaScript代码可能到达上万行，在保持用户体验，减少http请求的情况下，一个页面的所有的逻辑都会在一个javaScript文件中。极其不利于代码的扩展与复用。

## （三）MVC框架

为了克服这些问题，出现了前端的MVC框架，在MVC框架中我们把应用分成了三个部分：Model来管理数据，View层来负责渲染用户的界面，Controller来负责接收用户的响应，并根据用户的行为来改变Model，再把产生的数据结果交给View层，刷新用户界面。

但是在前端MVC框架实现中，并不一定有Controller, 客户端根据用户的行为修改客户端Model,客户端更新和该Model相关的View, 客户端更新和该Model送sync请求到服务器，服务器返回修改是否成功。如果不成功，返回上一个view的显示结果。所以前端的MVC框架并不是严格意义上遵循MVC的，常常为了开发的便捷，View层和Model层会直接进行通信。这就是导致了，View层和Model的逻辑会交错复杂。

# 三、React.js

FaceBook开源React.js引起了业界极大的关注。由于FaceBook前端业务需求越来越复杂，MVC的前端框架已经满足不了其复杂的业务场景的需要，FaceBook推出了React + Flux 来解决前端复杂的交互场景。

## 数据驱动渲染

在React的理念可以归结为一个简单的公式： [7] UI = render（data）

这里render可以说是一个纯函数，那么它的返回结果只会受参数data的影响，所以只要data不发生变化，那么UI就不会发生变化。所以在React中我们所能看到的UI是完全取决于data的状态的，只有data发生改变的时候，UI才会更新，所以react也实践了响应式编程的思想。那么开发者只要思考，需要在组件中显示什么样的数据就可以了，不需要思考具体要怎么样去显示。

那么我们编写交互的方式也从操纵DOM元素转变成了操纵数据，在react中分别有props和state保存着组件的数据也就是上文所提到的data参数，props是来自外界的数据，state是组件自身维持的数据。每当props和state发生变化时就会引起组件的重新渲染。

## （二）Virtual DOM

显而易见的是React如果发现组件的数据进行了变化，通过重复渲染组件来改变UI。但是通常情况下，一个组件中只有一部分的数据发生了变化，如果重新渲染整个组件无疑是浪费时间的。

React采用了Virtual DOM来解决这个问题，Virtual DOM 采用了React Diff算法，在组件状态发生了变化之后。React会在先在JavaScript树形结构中构建Virtual DOM, 每次渲染组件之前都会和之前的渲染结果对比，然后发现不同的处，在真正修改DOM时候，就只要修改有差别的部分就足够了。

## （三）与传统开发相比React的优势

传统的web开发通过直接操控DOM节点，这种方式很容易上手，开发者只要改变DOM节点的属性就能够改变用户界面的UI 。当项目变得庞大的时候，如图1所示，用JQuery写出的代码往往和DOM节点互相纠缠，当节点的事件和引用没有清理的情况下是很容易引起内存泄漏的。

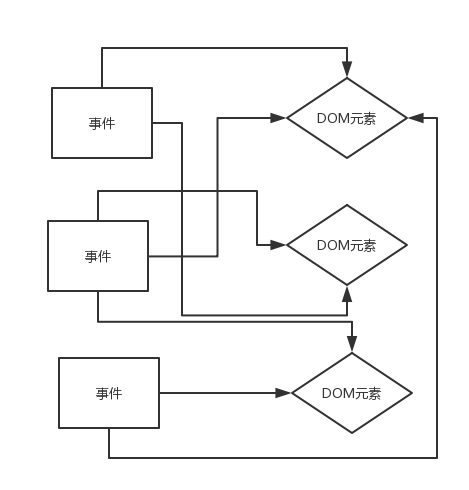


图1

React通过函数式编程的思维，通过了数据来驱动UI的改变，强制要求每一个组件，必须通过数据来驱动渲染，所以不管应用多么庞大复杂，都能让程序处于一个可控的范围。

React让开发者专注于界面应该是显示成什么样子的，而不是怎么样去显示这个界面。大大提高了开发组织代码的效率。

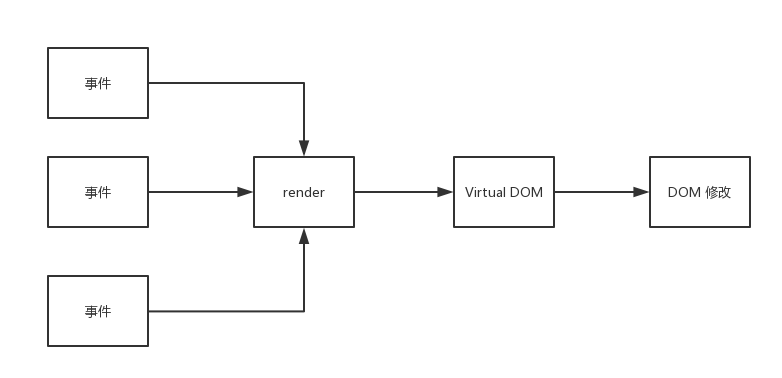


图2

# 四、Vue.js

## （一）渐进式框架

在框架使用的过程中，不免会被框架框架自身所限制。每个框架都有自身的主张，对比React和AngularJS我们就会发现，React主张的是函数式编程。而AngularJS更具有排他性，一旦使用AangularJS，就必须使用它的模块机制、依赖注入和特定的组件。而Vue采用渐进式的设计原则，框架的使用者可以按照需求来进行配置，从简单到复杂的场景都可以覆盖到。

## 双向数据绑定

在Vue中通过使用 ES5 提供的 Object.defineProperty() 方法，监控对数据的操作，从而可以自动触发数据同步，实现了数据双向绑定。并且，由于是在不同的数据上触发同步，可以精确的将变更发送给绑定的视图，而不是对所有的数据都执行一次检测，避免了繁琐的DOM操作。[10]每个Vue实例都会有相应的watcher实例对象。如果data数据源发生了变化，就会通知到watcher从而比较Virtual DOM 异步更新组件中需要更新的部分。

# 五、AngularJS

## （一）MVC

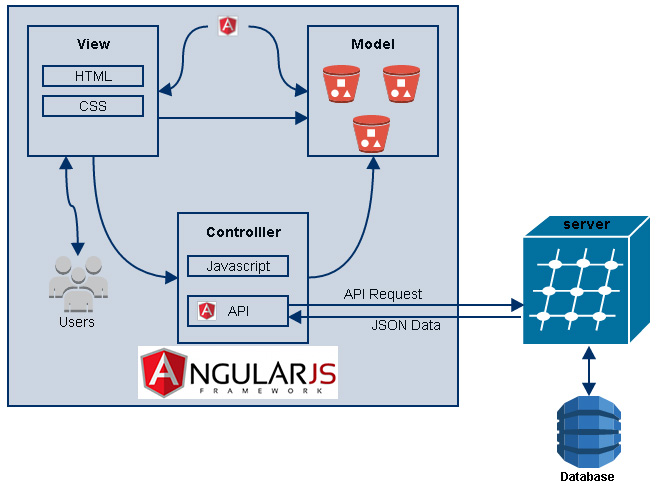


图3

如图3所示与React和Vue相比，Angular会显得十分的笨重。也只有Angular集成了MVC的每一个环节。

## （二）双向数据绑定

AngularJS 采用“脏值检测”的方式，数据发生变更后，对于所有的数据和视图的绑定关系进行一次检测，识别是否有数据发生了改变，有变化进行处理，可能进一步引发其他数据的改变，所以这个过程可能会循环几次，一直到不再有数据变化发生后，将变更的数据发送到视图，更新页面展现。如果是手动对 ViewModel 的数据进行变更，为确保变更同步到视图，需要手动触发一次“脏值检测”。

# 六、前端模块化和工程化

有了React，应对大型前端应用的开发似乎已经很完美了，但是考虑到团队的协作，似乎还是存在着许多的问题。例如：如何统一使用的框架的版本，如何制定开发的代码规范，如何处理各个模块之间的依赖关系。前端工程化就是为了解决如上所说的这些问题。

工程化的要素，会有以下几个方面

* 依赖管理与统一的组件发布与仓库

npm依赖于Node.js平台，是现代前端应用的开发的依赖管理工具。通过package.json配置文件的定义就可以统一开发环境对不同框架依赖的版本。开发者也可以通过npm平台管理和维护自己构建的工具库和插件，使得代码能够重复利用。

* 统一的开发规范

对于代码的规范需要人为制定规则和统一。可以通过ESLint制定代码规范，ESlint中定义了200多条JavaScript的编码规范。我们可以通过配置文件，决定代码缩进的格式，避免变量隐式转换，禁止扩展原生对象的原型，禁止使用with语句。

* 模块化/组件化开发

1. 模块化

随着前端代码的复杂程度不断地提高，JavaScript模块化的概念被提了出来，在没有模块化的前，编写JavaScript代码时，往往要在一个html页面中引入多个<script>标签。这样的方式组织代码会导致依赖关系不明显，也不利于代码后期的维护。

随之提出的AMD和CMD规范就是为了解决在浏览器端模块化开发的问题，依托了加载器实现的javascript模块化开发和加载，极大程度地提升了前端开发的体验。

1. Webpack

但是模块化开发的体验对开发人员来说是好，但是在加载上就有一些问题， 在前端优化上，我们一般会通过减少请求来优化页面的性能， 模块化加载显然是背道而驰的。这时候解决的方案就是把模块进行打包，发布前把所有的依赖打包在一起，这样就可以把页面资源请求的降到了最低。这时候打包工具就出现了，当前Webpack成为了前端应用最佳的依赖打包的解决方案，成为了当前最流行的依赖打包工具。在Webpack中，通过loader处理过的一切文件都是模块。Webpack还能够处理JavaScript和CSS扩展语言，将这些文件转换成在浏览器端能够使用的文件。Webpack还提供了插件的形式，支持拓展的功能。只要通过npm进行下载，在配置文件进行插件的配置。

# 七、参考文献

[1]CORY GACKENHEIMER著；张铮铮译.REACT导学[M].北京：人民邮电出版社.2016.

[2]（英）Artemij Fedosejev著.React 精髓[M].北京：电子工业出版社.2016.

[3]张轩著.React全栈 Redux+Flux+webpack+Babel整合开发[M].北京：电子工业出版社.2016.

[4]刘一奇.React与Redux开发实例精解[M].北京：电子工业出版社.2016.

[5]程墨著.深入浅出React和Redux[M].北京：机械工业出版社.2017.

[6]封宇[1,2],陈宁江[1].基于MVVM架构的移动Web前端展案[J].计算机与现代化,2014,0(11):1-4.

[7] 孙连山,李云倩.MVVM框架在Web前端的应用研究[J].电脑知识与技术：学术交流,2016,0(2X):45-46.

[8] [1]冯晨超. 基于AngularJS的物品和用户查询模块的设计与实现[D].南京大学,2014.