

分享议题： 重新认识FE

By hfuuss

分享目录

- Web发展史
- 为什么FE的工资越来越高
- 从零开始搭建nbugs前端项目

分享目录

- Web发展史
- 为什么FE的工资越来越高
- 从零开始搭建nbugs前端项目

Web发展史

开发效率越来越快，几个月缩减到几分钟

- Web的诞生
- 动态内容的出现：CGI
- Web编程语言：PHP/ASP/JSP
- 分布式企业计算平台：J2EE/.Net
- 框架横飞的年代：MVC，ORM
- 浏览器端的魔术：AJAX
- 回归Web本质：REST
- 前端MVC：Angular/React/Backbone/Vue
- JavaScript在服务器端的逆袭：Node
- 前端的自动化、服务化和实时化
- 前端组件化

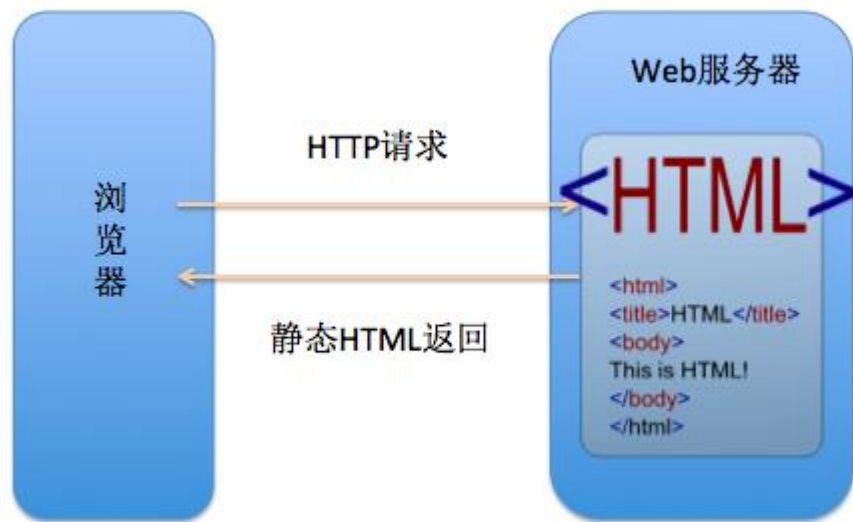


后端为主的时代



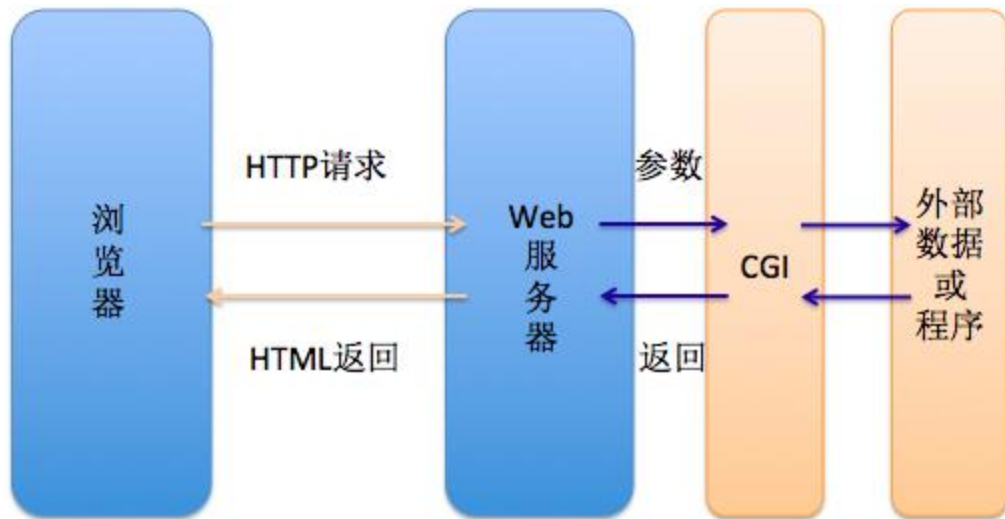
前端为主的时代

Web的诞生



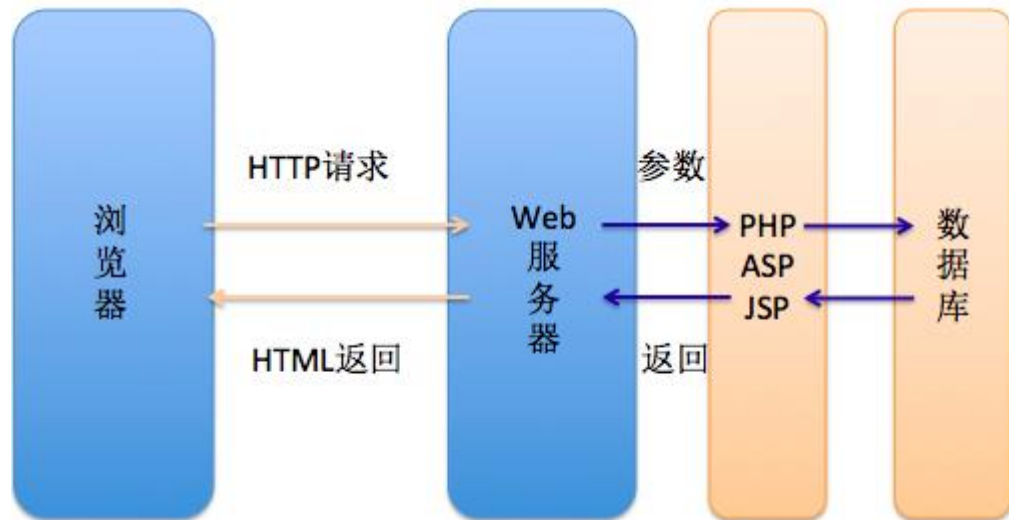
动态内容的出现：CGI

- CGI可以用任何支持标准输入输出和环境变量的语言编写，比如Shell脚本,C/C++语言，只要符合接口标准即可。比如你用C语言编写CGI程序，你把希望返回的HTML内容通过printf输出就可以发送给Web服务器，进而返回给用户。大一的时候，我在学习《c语言也能干大事》的时候，将一个用c语言写的exe可执行文件放到web服务上面。发现真的可以实现，并没有安装我想象的那样会下载exe。



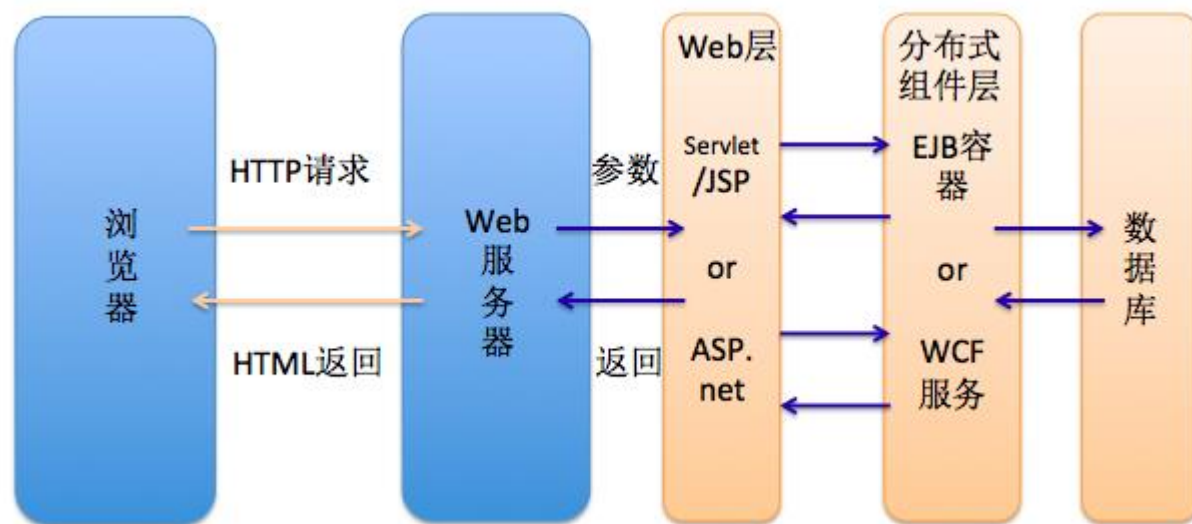
Web的诞生

- 模板功能



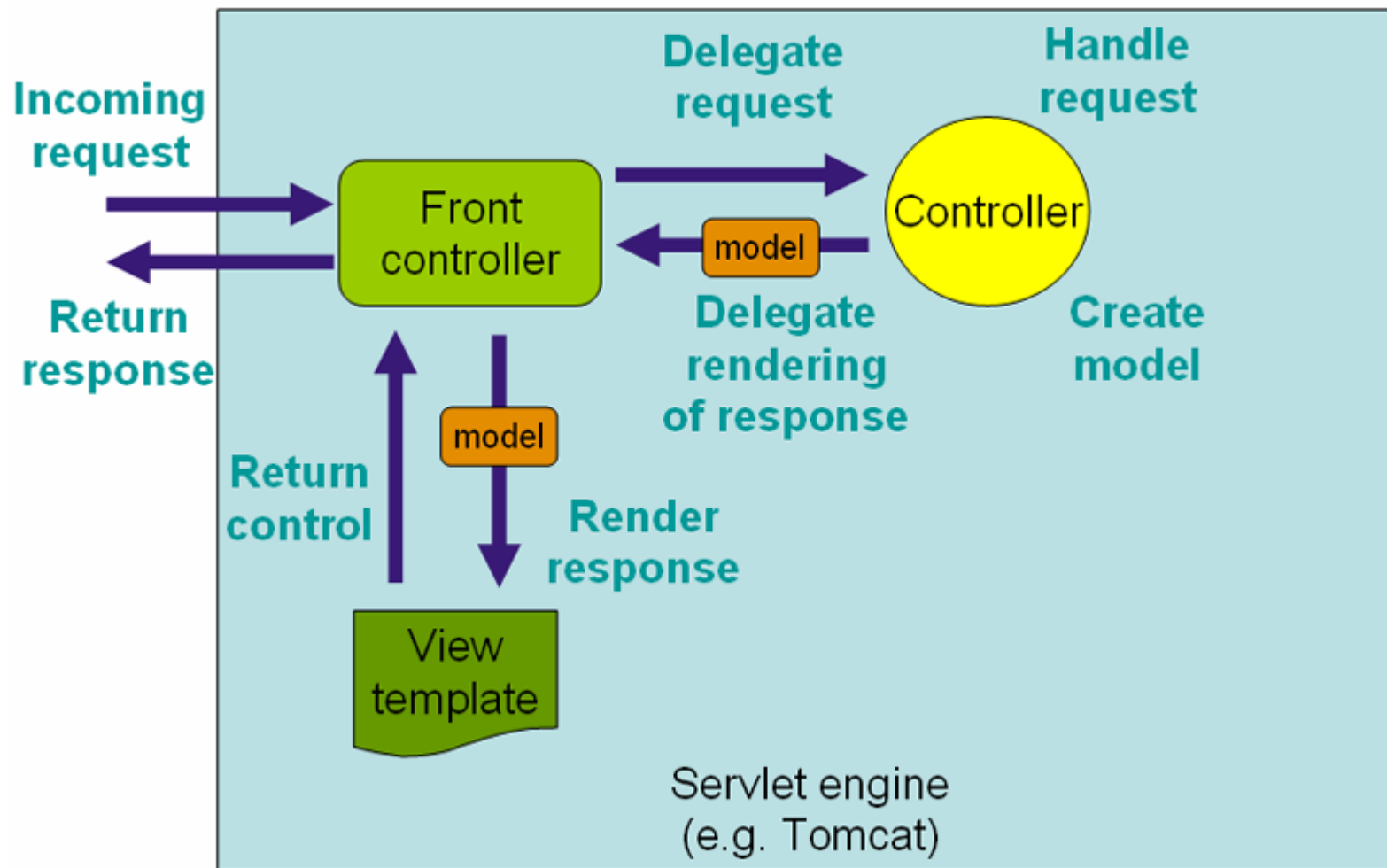
分布式企业计算平台：J2EE/.Net

- 分布式、安全性、事务性等方面需要升级

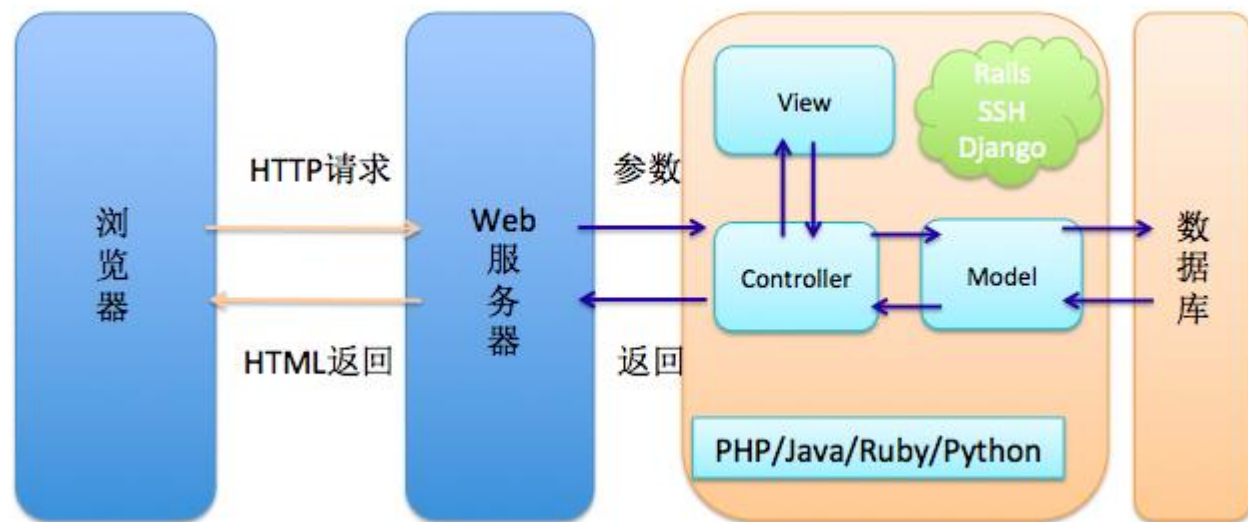


框架横飞的年代：MVC，ORM

Spring MVC的示意图



框架横飞的年代：MVC，ORM

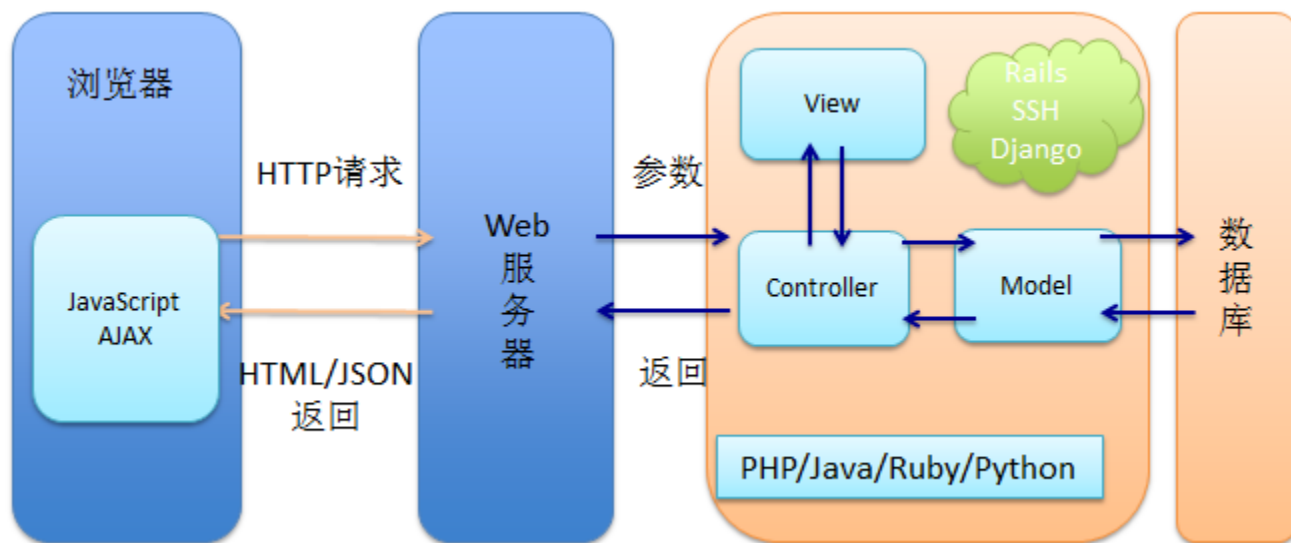


回归Web本质： REST

- REST鼓励基于URL来组织系统功能，充分利用HTTP本身的语义，而不是仅仅将HTTP作为一种远程数据传输协议。

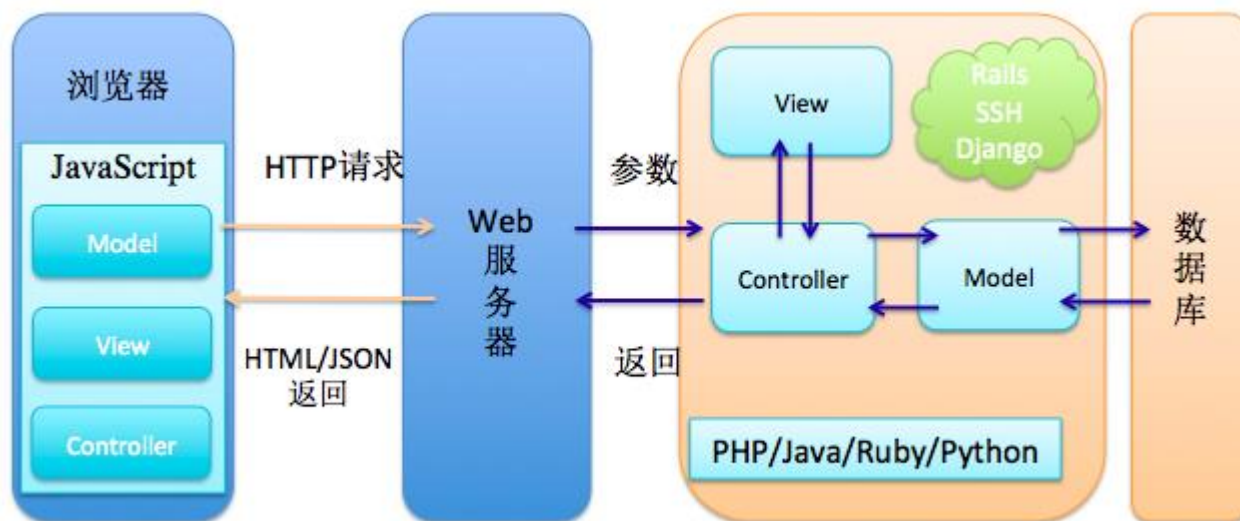
浏览器端的魔术：AJAX

- AJAX的出现，以及诸如EXTJS、DOJO等一些前端开发框架的出现，也使得单页应用（Single Page Application）在这个时候流行起来。
- 这种模式下，前后端的分工非常清晰，前后端的关键协作点是 Ajax 接口，规定好交互接口后，前后端工程师就可以根据约定，分头开工，开发环境中通过Mock等方式进行测试，同时在特定时间节点进行前后端集成测试。



前端MVC: Angular/Backbone/Vue

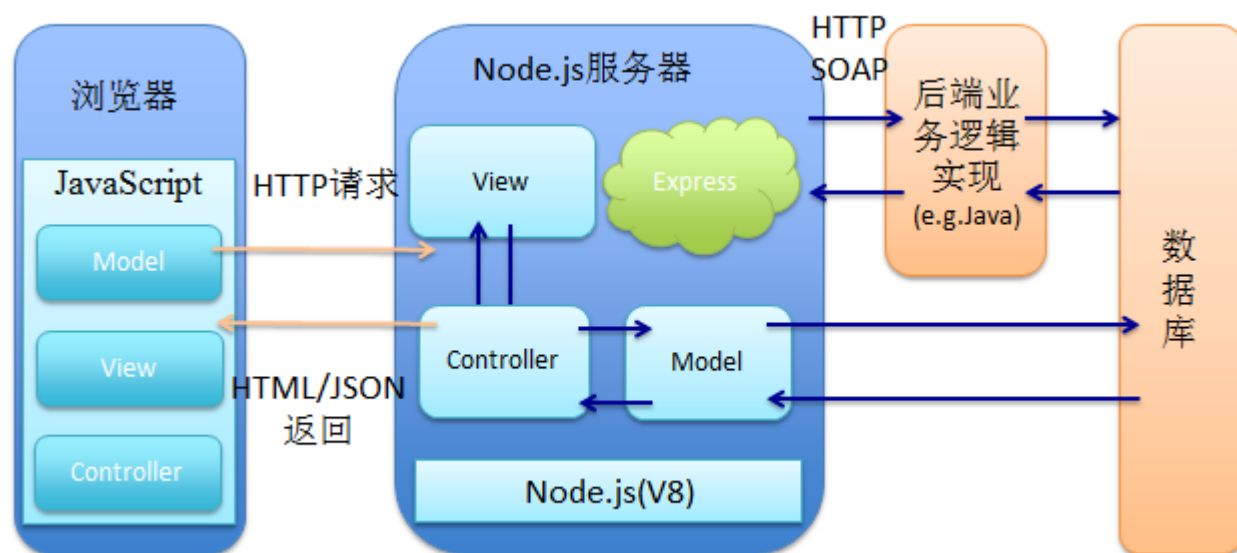
- 现在, 对于一个前端功能、交互复杂的SPA, JavaScript代码很容易膨胀(超过10万行)。很自然地, 像服务端从JSP向MVC框架转换的过程一样, 前端开发也出现了大量的MVC框架



JavaScript在服务器端的逆袭：Node

- 各大浏览器的竞争，使其引擎的性能不断提升，至今Google V8引擎的性能已经足以运行大型Javascript程序。在V8之上加以网络、文件系统等内置模块，形成了如今的Node.js。

[npm](#) 是 Node 的模块管理器，功能极其强大。它是 Node 获得成功的重要原因之一。



Node.js能干什么？

- 网站（如express/koa等）
- im即时聊天(socket.io)
- api（移动端，pc，h5）
- http proxy（淘宝首页）
- http proxy延伸，组装rpc服务，作为微服务的一部分
- 前端构建工具(grunt/gulp/bower/webpack/fis3...)
- 写操作系统（NodeOS）
- 跨平台打包工具（nw.js、electron、cordova/phonegap）
- 命令行工具（比如cordova）
- 编辑器（atom，vscode）
- 总的来说，啥都能干。

前端的自动化、服务化和实时化

- 自动化: **grunt**、**gulp**已帮助前端很好地解决了代码压缩、生成md5、合图等功能。结合七牛云等平台。(目前nbugs前端架构采用**gulp**和七牛云)
- 服务化: 作为接入层, 调用**java**、**c++**等基础功能, 处理数据库, 将数据发给浏览器处理。(java等语言可以专注数据处理, 业务交给**nodejs**)
- 实时化: **GraphQL**(前端操作数据库) **Meteor**(DDP协议, js全栈框架)

前端组件化

- React Facebook出品 小而轻 学习成本：middle
- Angular2 Google出品 大而全 学习成本：hard
- Vue尤雨溪出品 吸收了ng1和react的优点 学习成本：easy
- 尤雨溪曾在google工作，觉得angular1不好用，自己写了一个Vue，凭借Vue，进入了meteor公司。后来vue在国内大火。辞职全职做Vue。Vue的缺点感觉就是生态圈还没有react生态圈完善。因为react早出来的，很多牛人们都在搞React。
- 2016年9月3日，在南京的JSConf上，Vue作者尤雨溪正式宣布加盟阿里巴巴Weex团队
- 个人观点：前端新人可以加入Vue阵营，一起完善Vue生态圈。

Web发展史

开发效率越来越快，几个月缩减到几分钟

- Web的诞生
- 动态内容的出现：CGI
- Web编程语言：PHP/ASP/JSP
- 分布式企业计算平台：J2EE/.Net
- 框架横飞的年代：MVC，ORM
- 浏览器端的魔术：AJAX
- 回归Web本质：REST
- 前端MVC：Angular/React/Backbone/Vue
- JavaScript在服务器端的逆袭：Node
- 前端的自动化、服务化和实时化
- 前端组件化



后端为主的时代



前端为主的时代

总结：

- 1、技术发展越来越快
- 2、前端和后端的角色转变
- 3、web开发效率越来越快

分享目录

- Web发展史
- 为什么FE的工资越来越高
- 从零开始搭建nbugs前端项目

为什么FE的工资越来越高

- 因为FE的门槛越来越高了，负责的工作也越来越多了。
- FB的工作重心大部分交给了FE。FB的重心在想数据处理方向走。

分享目录

- Web发展史
- 为什么FE的工资比一般的FB高
- 从零开始搭建nbugs前端项目

从零开始搭建nbugs前端项目

- 1、安装nodejs (类似安装jvm)
- <https://nodejs.org/zh-cn/>
- 配置npm库为nbugs-inc: `npm config set registry http://registry.npm.nbugs-inc.com/`
- 2、`git clone xxxxxxxx` (克隆项目)
- 3、`cd xxxx` (进入项目)
- 4、`npm install`
- 5、`npm start`
- 注意：要根据readme文件，进行不同的代理文件配置。即可进行调试。

谢谢观看