

2023-2024学年秋季学期

Web安全技术  
*Web Security*

授课团队：刘奇旭、刘潮歌

学生助教：曹婉莹、孙承一

# Web安全技术

Web Security

## 1.2 Web的简明历史

刘潮歌

liuchaoge@iie.ac.cn

2023年09月19日



中国科学院大学  
University of Chinese Academy of Sciences

# 一章一问

- Web服务器和浏览器曾经存在哪些安全问题、现在又面临着哪些安全问题，如何解决？



# 本章大纲

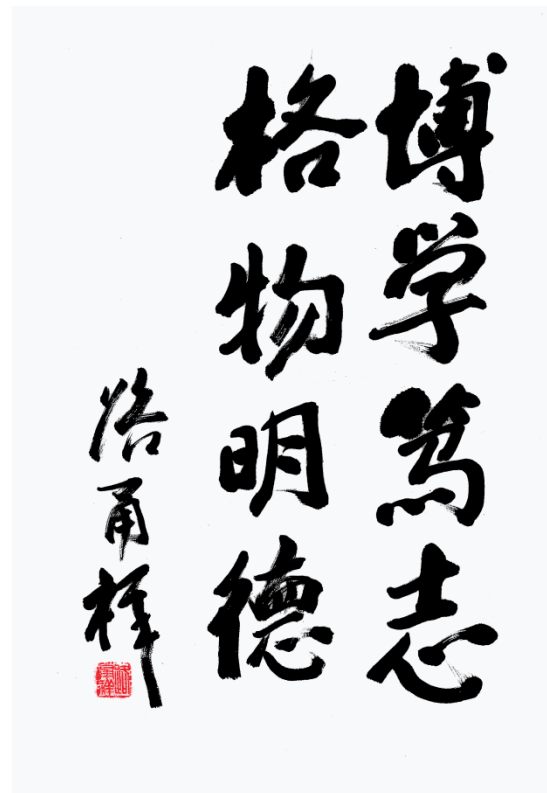
## □ Web发展简史

- Web技术发展
- Web 1.0/2.0/3.0

## □ 浏览器发展简史

- 浏览器内核
- 浏览器发展
- 浏览器安全机制

## □ Web安全简史



# WEB发展简史

## □ Web描述

- “Web是一个抽象的（假想的）信息空间”
- Web的首要任务就是向人们提供信息和信息服务
- 就技术而言，互联网是指通过TCP/IP协议族互相连接在一起的计算机网络。而Web是运行在互联网上的一个将计算机的信息资源连接在一起的超大规模的分布式系统。
- 大家开发的Web应用本质上就是可以提供信息或者功能的Web资源，成为Web这个全球超大规模分布式系统中的一部分。



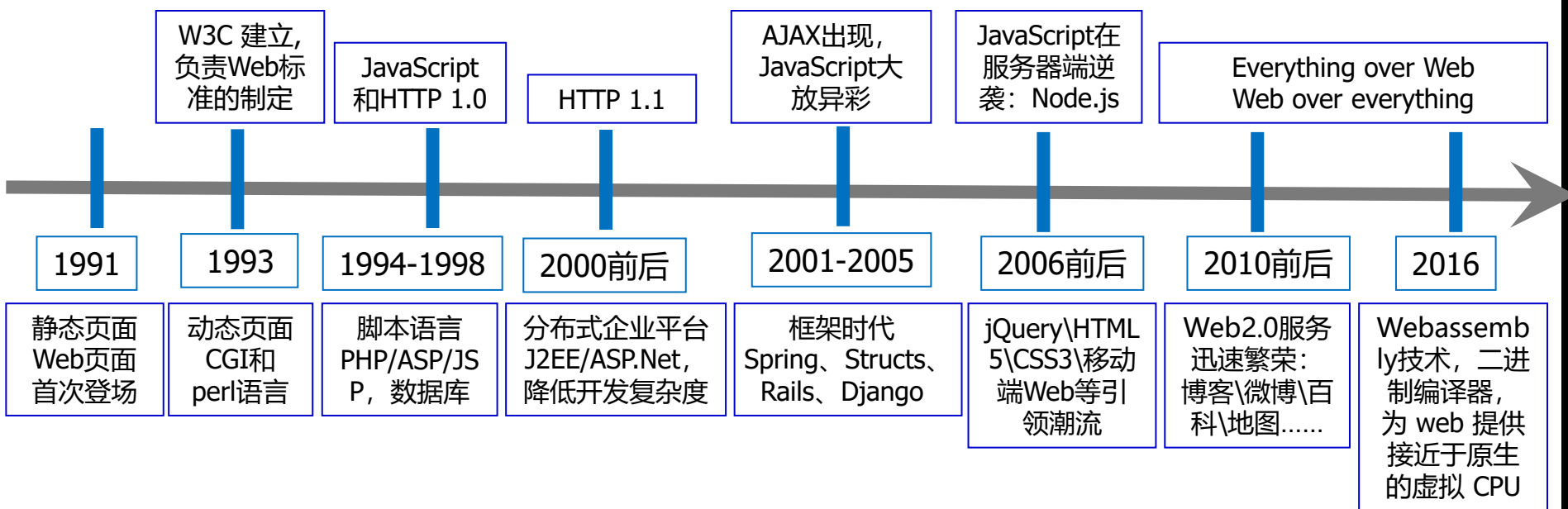
# WEB发展简史

## □ 技术基础

- 通过超文本标记语言（HTML）描述信息资源
- 通过统一资源定位技术（URI）定位信息资源
- 通过应用层协议（HTTP）请求信息资源，实现分布式信息共享
- HTML、URI和HTTP三个规范构成了Web的核心体系结构，是支撑着Web运行的基石



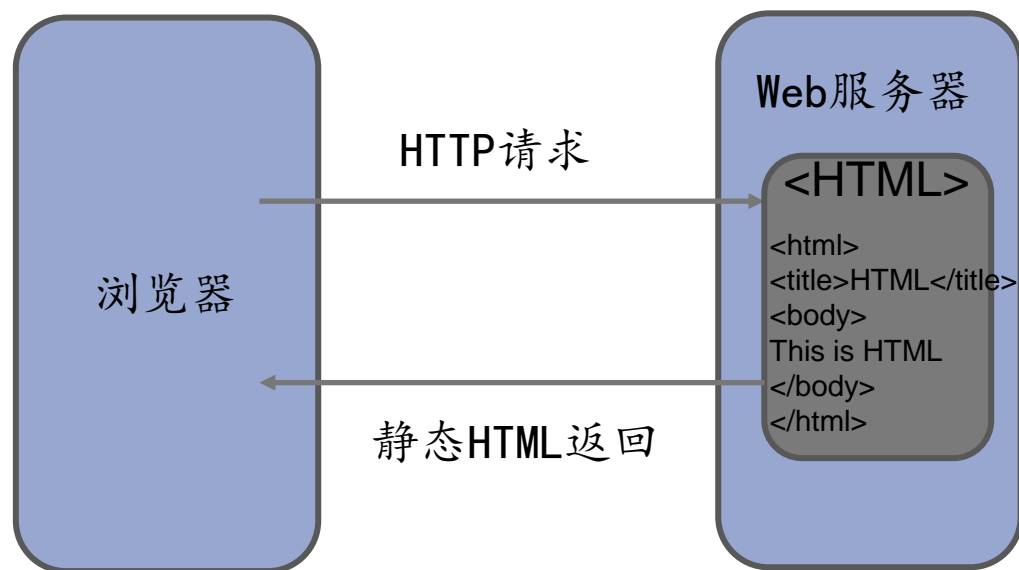
# WEB发展简史



# 简单WEB静态页面

1991年

- 客户端：通过浏览器访问Web站点，浏览器展现静态文本或图像信息
- 服务器端：每一个Web站点由Web服务器及许多Web页所组成
- 典型技术：HTML（超文本标记语言），HTTP（超文本传输协议）



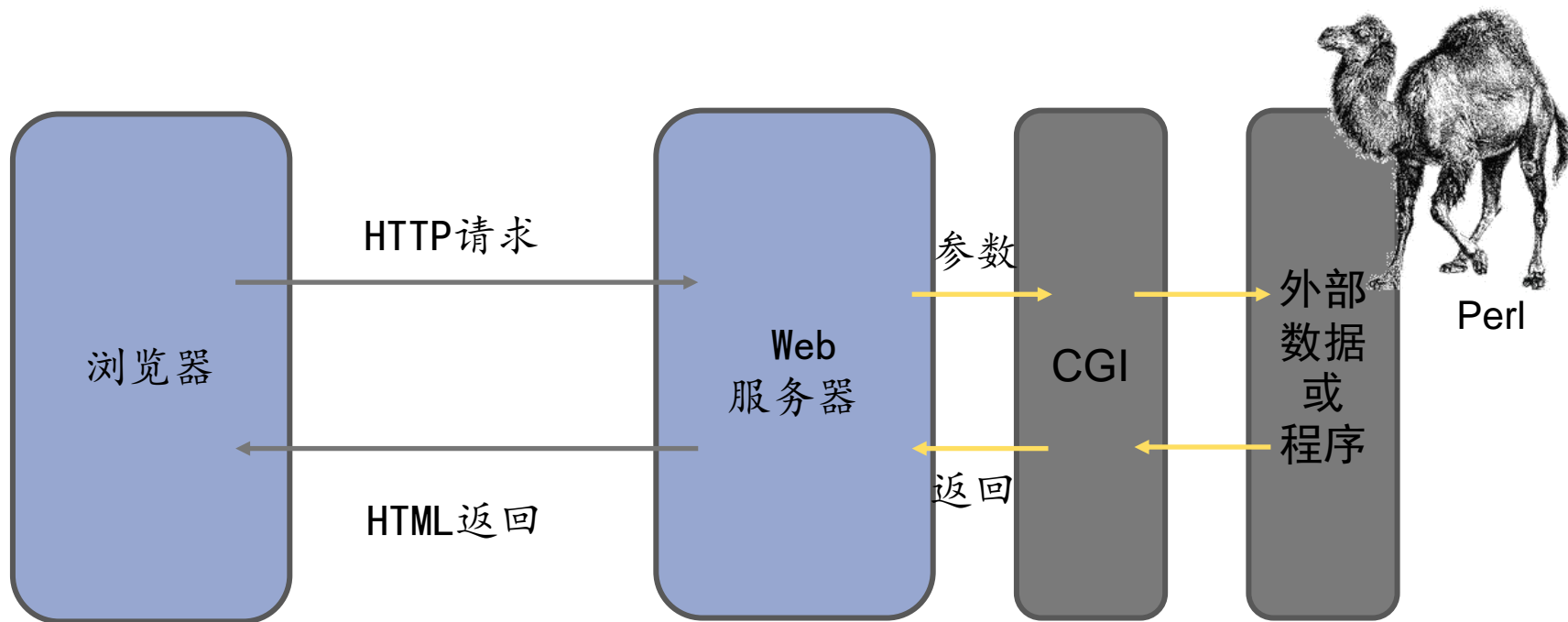
那个年代精美页面的典范（Apple）



# 动态内容出现: CGI

1993年

- 1993年, CGI(Common Gateway Interface)出现
- CGI定义了Web服务器与外部应用程序之间的接口, Web服务器可以通过CGI执行外部程序, 让外部程序根据Web请求生成动态的内容



# 动态内容出现：CGI，缺点

1993年

- CGI对每一个请求都会启动一个进程来处理，性能上扩展性不高
- 组织CGI/Perl这样的脚本代码太混乱，可读性和维护性都是大问题
- 直接使用文件系统或者环境变量，不安全



- 静态部分：用HTML固定起来，形成“模板”
  - 动态部分：Web请求处理的时候，用程序动态生成
  - 返回信息：动态内容嵌入到静态模板
- 1994年，PHP可以把程序（动态内容）嵌入到HTML（模版）中去执行，不仅能更好的组织Web应用的内容，而且执行效率比CGI还更高
  - 1996年出现的ASP和1998年出现的JSP本质上也都可以看成是一种支持某种脚本语言编程（分别是VB和Java）的模版引擎

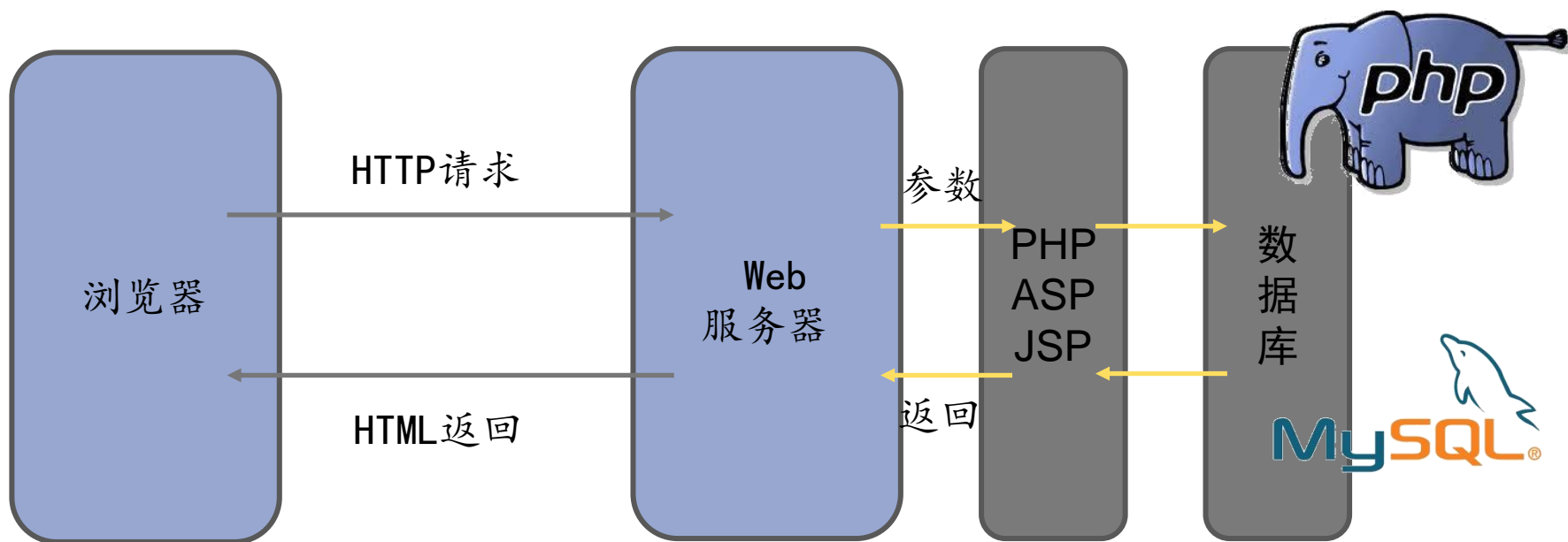
1994年5月15日，中国科学院高能物理研究所设立了国内第一个WEB服务器，推出中国第一套网页，内容除介绍中国高科技发展外，还有一个栏目叫"Tour in China"。此后，该栏目开始提供包括新闻、经济、文化、商贸等更为广泛的图文并茂的信息，并改名为《中国之窗》。



# WEB编程脚本语言：PHP/ASP/JSP

1994-1998年

- 1996年W3C发布了CSS1.0规范，允许开发者用外联的样式表来取代难以维护的内嵌样式，让HTML页面更加容易创建和维护
- Web大杀四方：脚本语言 + 数据库技术



# WEB发展简史

## □ 安全问题

- SQL注入
- 文件包含
- 文件上传
- webshell



# 分布式企业计算平台:J2EE/.NET 1999-2000年

- J2EE (Java EE) : 1999年, Web开始广泛用于构建大型应用, 在分布式、安全性、事务性等方面的要求催生了J2EE
- ASP.Net: 2000微软ASP升级为ASP.Net, 其ASP.net构件化的Web开发方式以及Visual Studio.net开发环境的强大支持, 大大降低了开发企业应用的复杂度

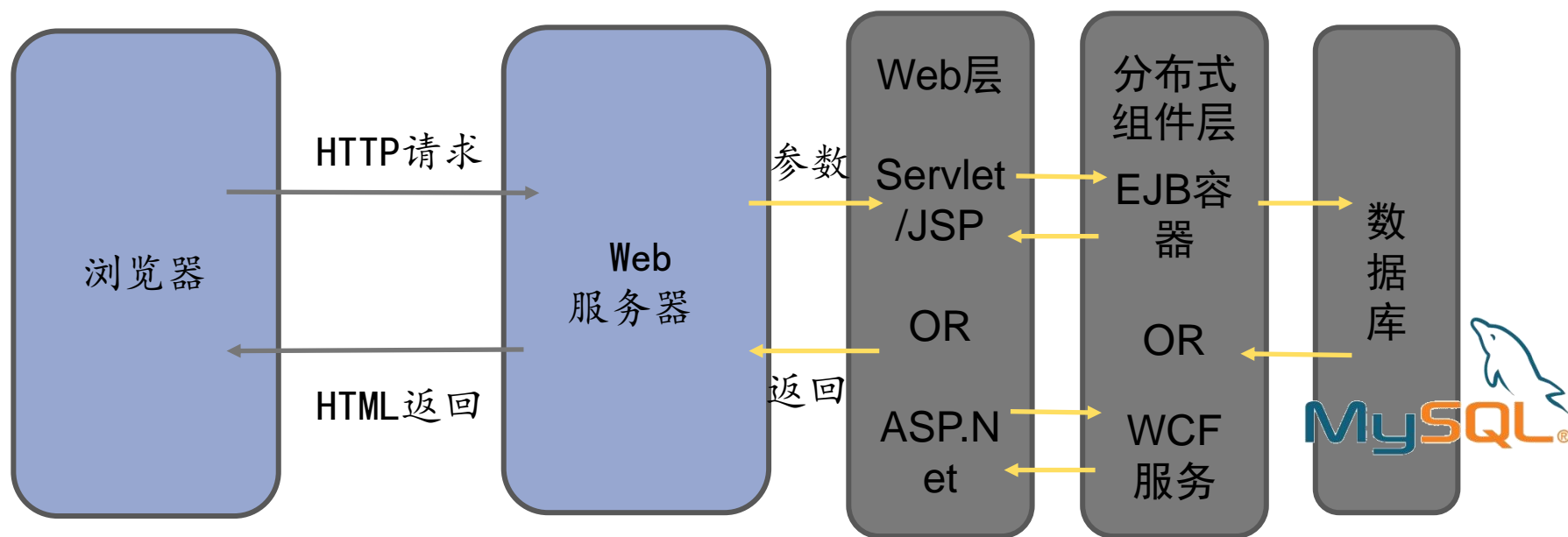


语言跨平台、易开发



# 分布式企业计算平台:J2EE/.NET

1999-2000年



- ❑ 脚本语言大大提高了应用开发效率，但对于复杂的大型Web应用，URL地址纷繁复杂，Web页面多种多样，管理大量的后台数据
- ❑ 在架构层面上解决维护性和扩展性等问题
  - MVC（Model View Controller），Model用于封装与业务逻辑相关的数据和数据处理方法，视图View是数据的HTML展现，控制器Controller负责响应请求
  - ORM（Object Relation Mapping），用于实现面向对象编程语言里不同类型系统的数据之间的转换，解决面向对象编程和关系数据库不匹配问题



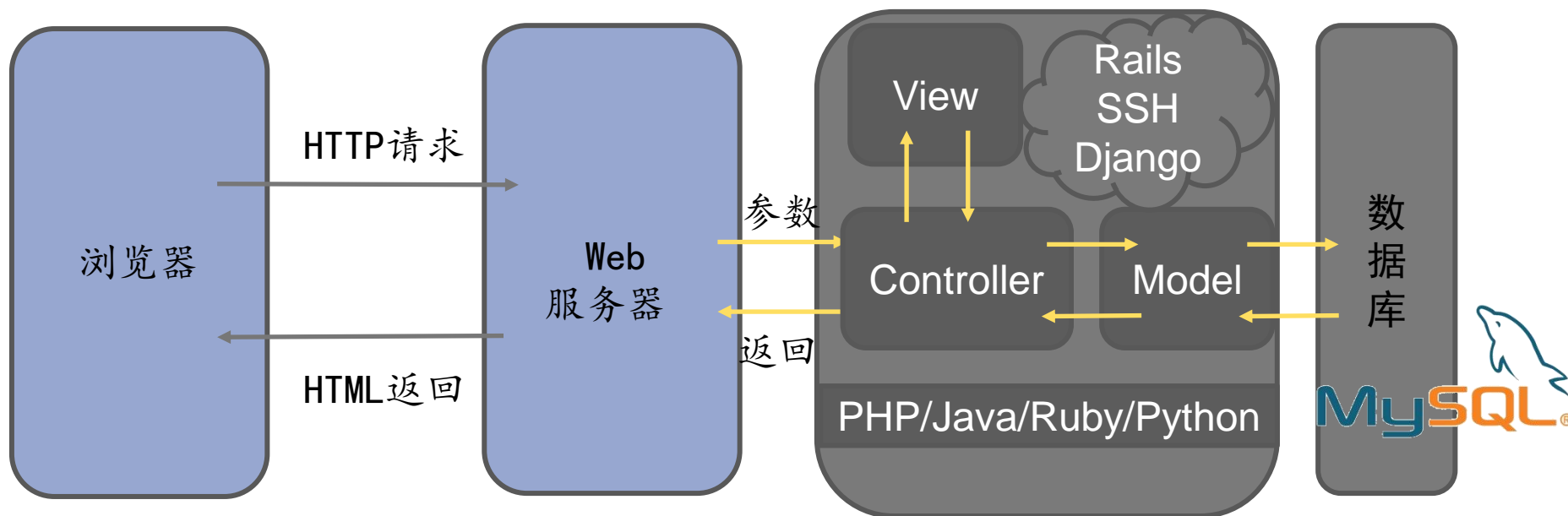


- 2003年出现的Java开发框架Spring，同时更多的动态语言也被加入到Web编程语言的阵营中
- 2004年出现的Ruby开发框架Rails
- 2005出现的Python开发框架Django，都提供了全栈开发框架，或者自身提供Web开发的各种组件，或者可以方便的集成各种组件

系统维护和扩展

# 框架时代

2001-2005年



# WEB框架

2001-2005年

 python™



 Microsoft  
ASP.NET



Tornado

django

Spring



STRUTS



HIBERNATE



 Microsoft

ASP.NET MVC 4



# WEB发展简史

## □ 安全问题

### □ 框架漏洞，影响广泛，批量中招

- Struts2漏洞，远程执行代码

- S2-001, S2-007, S2-008, S2-012, S2-013, S2-015, S2-016, S2-029, S2-32, S2-033, S2-036, S2-037

### □ 血洗！血洗！血洗！

今天晚上中国互联网被Struts2漏洞血洗-Apache, Struts2, 漏洞, 安全 ...

[news.mydrivers.com](http://news.mydrivers.com) > IT业界 > 业内动向 ▼

2016年4月26日 - Apache官方今天晚上发布安全公告(官方编号S2-032/CVE编号CVE-2016-3081),

Apache Struts2服务在开启动态方法调用(DMI)的情况下, 可以被 ...



# 浏览器端的魔术: **AJAX**

2005年

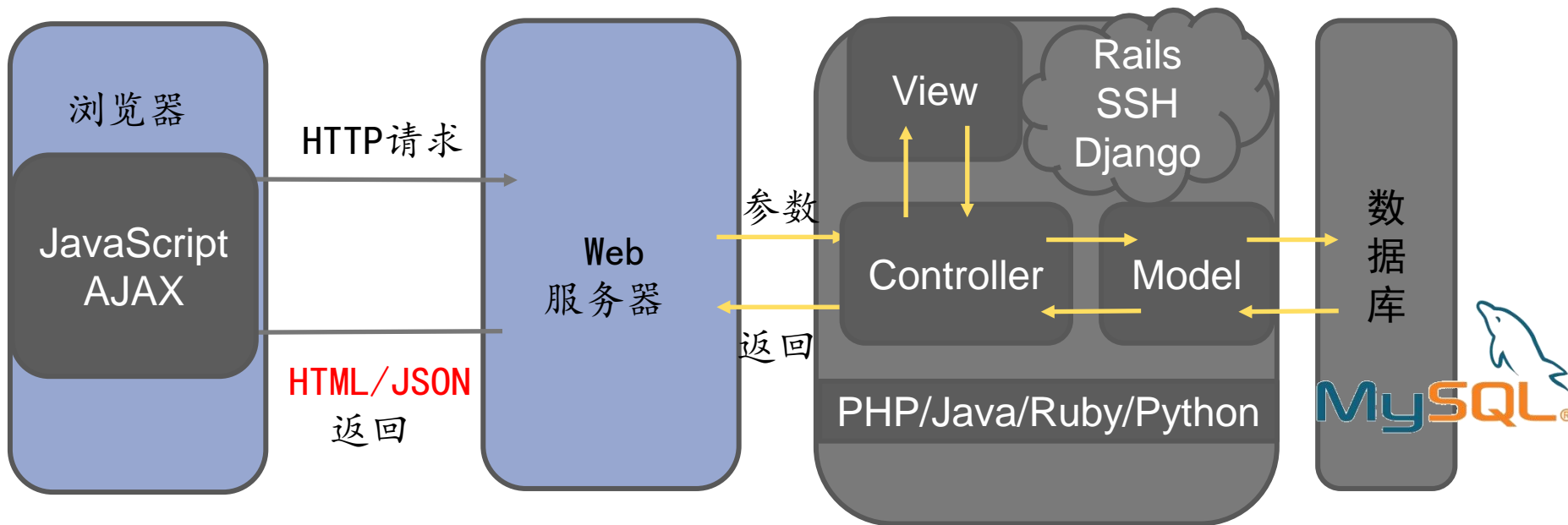
- 1995年NetScape推出JavaScript脚本语言，用于在浏览器上运行，增加网页动态性
- 微软推出JScript，CEnvi推出ScriptEase，但是缺乏统一规范，使得浏览器兼容性成为程序员的梦魇
- 欧洲计算机制造商协会，创建ECMA-262标准（ECMAScript）
- JavaScript：响应浏览器用户事件，检测表单的正确性，动态修改HTML页面，减少与服务器端通信，可以做出很酷的页面动态效果



# 浏览器端的魔术: **AJAX**

2005年

- 2005年出现的AJAX（Asynchronous JavaScript and XML），基于JavaScript的XmlHttpRequest，用于创建交互性更强的Web应用
- 异步地与服务器通信，局部地修改页面



# 浏览器端的魔术: **AJAX**

2005年

- AJAX向服务器发送并取回必须的数据，并在客户端采用JavaScript处理服务器响应，更新页面局部信息
- 浏览器和服务器的数据交换大大减少，客户端也可以更快速地响应用户操作
- Ajax可以认为是Web2.0的基石性的技术，为互联网的腾飞起到了重要作用

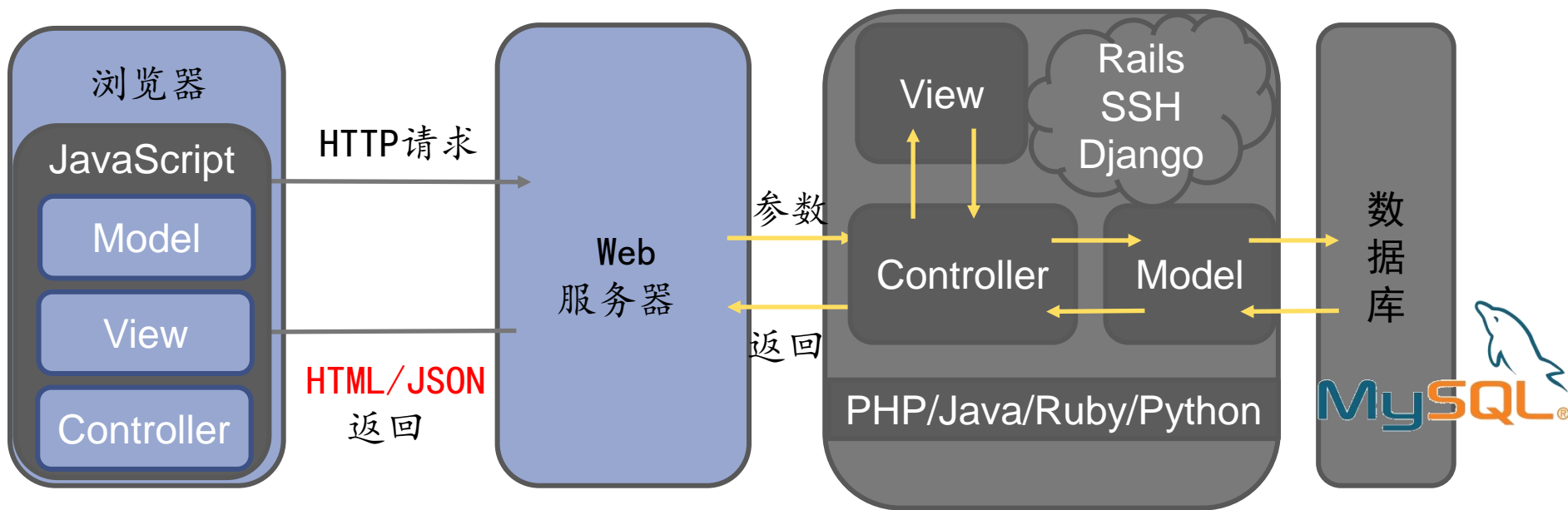
**JavaScript和AJAX  
是前端技术的革命！**



# 前端MVC: ANGULAR、BACKBONE等

2006~年

- 对于前端功能、交互复杂的系统，JS代码很容易膨胀
- 与服务端向MVC框架转换类似，前端开发也出现了大量的MVC框架





# WEB发展简史

## □ 安全问题

## □ XSS、CSRF

## □ JS木马！

### 网站代码中暗藏JS挖矿机脚本

 2017-09-26 共1262328人围观，发现 11 个不明物体 WEB安全

挖矿机这个名字相信大家已经越来越熟悉，但网页版的挖矿机大家有没有见过呢？近日360互联网安全中心就发现了这样一款以JavaScript脚本形式存在于网页中的挖矿机。

扩展阅读：

[挖矿>>](#)

[区块链专区>>](#)

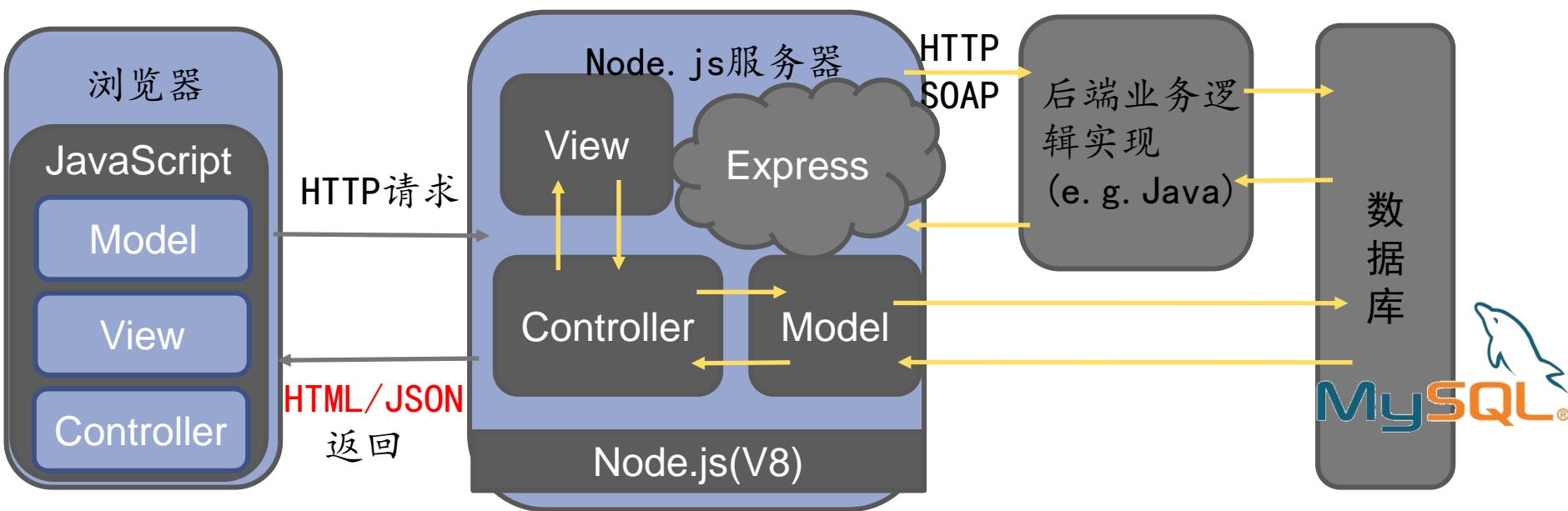
该脚本代码被嵌入了一个叫做“万方科技学院内网代理系统”的网站中，一旦用户打开该网站，浏览器便会按照脚本的指令变成一个门罗币挖矿机。这一行为，完全没有告知用户，更没有经过用户的同意，而这一段附加的挖矿代码通常因为大量占用CPU，使用户的计算机变得异常卡顿甚至无法正常使用。



# JAVASCRIPT在服务器端的逆袭：NODE

2006~年

- JavaScript用于服务端：Node.js
- 异步本质：Node.js在处理I/O密集型业务中优势凸显，而大多Web业务中I/O性能都是瓶颈



# NODE.JS

## 定义

Node.js是能够在服务器端运行的，跨平台的JavaScript运行时。

## 核心特点

Node.js是JavaScript运行环境，而不是一种语言，其内置Web服务器，可以与操作系统及文件系统直接交互。

## 重度用户

IBM、Microsoft、Yahoo、Walmart、Groupon、SAP、LinkedIn、Rakuten、PayPal、Voxer、GoDaddy

## PC端程序

将Node.js环境嵌入到PC应用程序中，诞生了很多优秀的PC端应用程序，如Visual Studio Code、Whatsapp桌面版、Skype等



# 小结

- 抽象层次不断提高，更高的抽象层次屏蔽更低层的复杂性，提高开发效率
- 计算机技术发展的一个普遍规律：需求的扩张，技术的进步，技术瓶颈，技术突破。如此往复
- 从语言到框架的演进，前后端都越来越强大



# 本章大纲

## □ Web发展简史

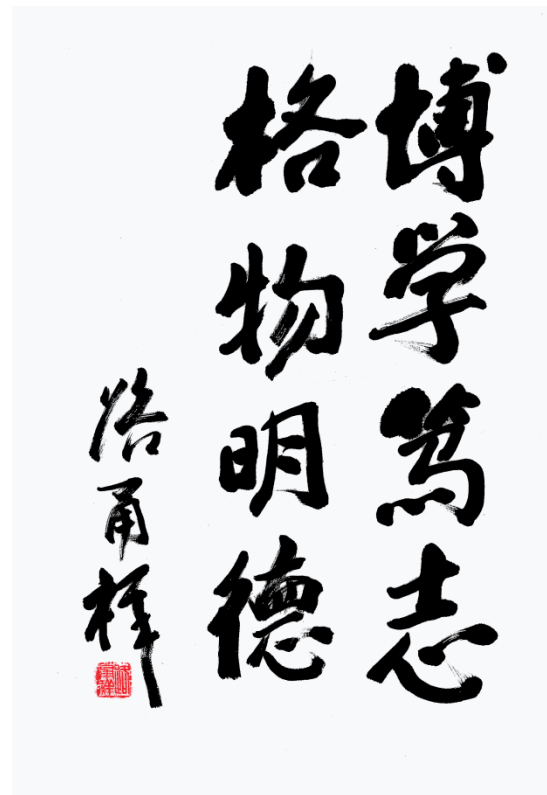
- Web技术发展
- Web 1.0/2.0/3.0

## □ 浏览器发展简史

- 浏览器内核
- 浏览器发展
- 浏览器安全机制

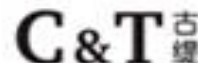
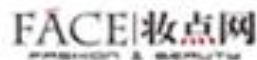
## □ Web安全简史

- 框架实例



# 过去时：WEB 1.0

- 特点：信息共享，内容为王，单向性
- 具有媒体性质的互联网页面表现方式，高度聚集而不产生用户交互



E  
m

# 现在时：WEB 2.0

- 特点：信息共建，关系为王，交互性
- 在模式上由单纯的“读”向“写”和“共同建设”发展；由被动地接收互联网信息向主动创造互联网信息发展
- Web2.0不同于Web1.0的最大之处在于它的互动性，用户既是网站内容的浏览者，也是网站内容的制造者
- 一种以用户为中心的网络技术与服务
- 以用户参与、用户互动为典型特征的万维网



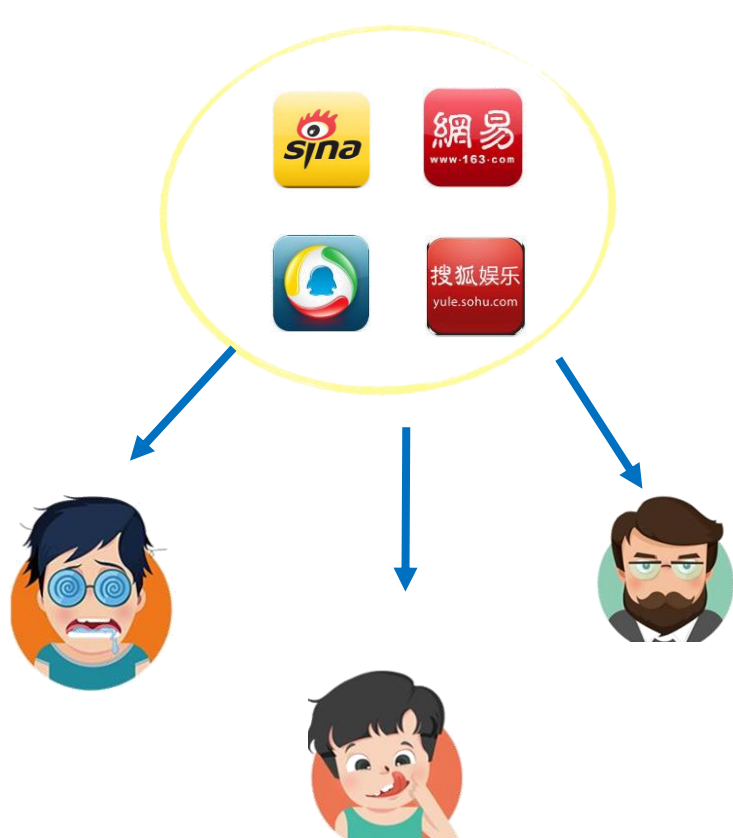
# WEB 2.0 典型的应用和技术

- BLOG（谷歌博客等）
- SNS（人人网，开心网，猫扑网等）
- 微博
- 维基百科Wiki
- 内容聚合RSS
- Mash up(混聚)
- Ajax

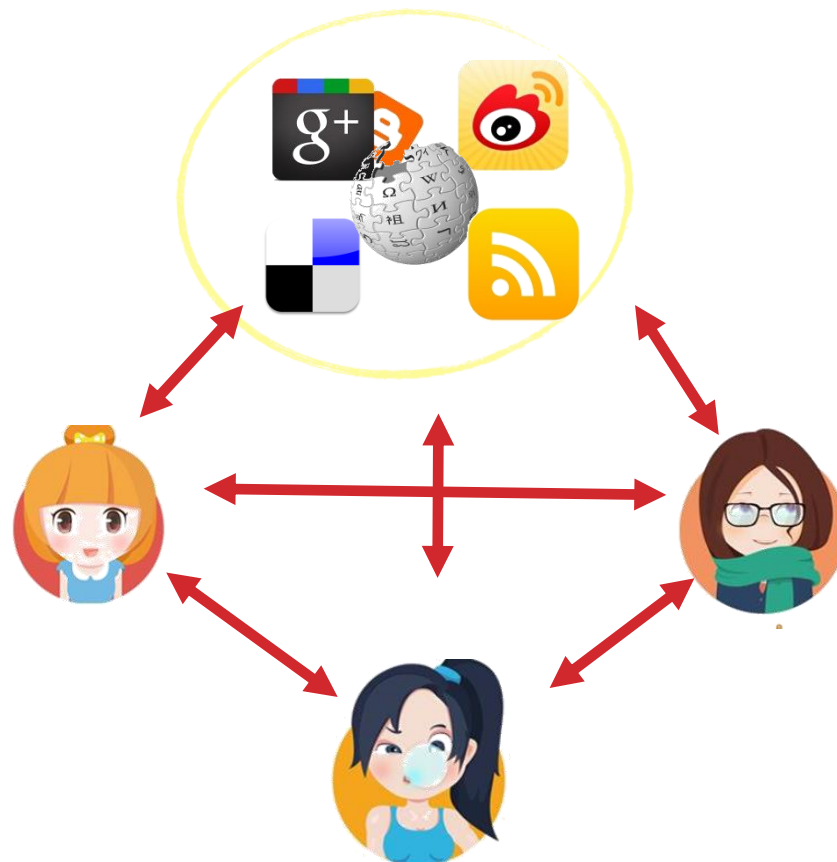




# WEB 1.0和WEB 2.0对比



Web1.0时代



Web2.0时代

# WEB 3.0

## □ 特点：数据关联，人工智能

- 未来的Web综合利用云计算，**语义网**等技术，让人可以从时间和资源上解放，使得人们可以更加专注于知识的获取，知识的分享和更加方便进行创新，Web本身将更加体现用户体验，更加专注于文化层面和“虚拟社会”
- **3D页游？ WebOS？ 虚拟现实？**

### Semantic Web - Wikipedia

[https://en.wikipedia.org/wiki/Web\\_3.0](https://en.wikipedia.org/wiki/Web_3.0) ▼

"Semantic **Web**" is sometimes used as a synonym for "**Web 3.0**", though the definition of each term varies. **Web 3.0** has started to emerge as a movement away from the centralisation of services like search, social media and chat applications that are dependent on a single organisation to function.



# WEB3

## Web3的提出：

2014, Gavin Wood coined the term **Web3**, he believes decentralized technologies are the **only hope of preserving liberal democracy**.

At the most basic level, Web3 refers to a **decentralized online ecosystem based on the blockchain**.

- Web3几乎与“Web”无关，如同WX几乎与“5G/Mobile”无关
- Web3是一个“广义”概念，代表一种去中心化理念及各种应用



# WEB3能力

- ✓ Web 1.0: read-only
- ✓ Web 2.0: read-write
- ✓ Web3: read-write-OWN



# WEB3登陆

## Web 1.0 Login Form

**Web 1.0**

Username

e.g unclebigbay

Password

your secret key

LOGIN

## Web 2.0 Login Form

**Web 2.0**

Login with Facebook

Login with Twitter

Login with GitHub

## Connect Wallet

**Web3**



Coinbase Wallet



MetaMask



WalletConnect



Slope



# WEB发展简史

## □ 安全问题

- 人人都可以成为“媒体”
- 互联网的放大效应
- 恶俗炒作，人肉搜索，暴恐音视频，有害言论.....



# WEB发展简史

## □ 小结

- web1.0: 网络-人
- web2.0: 人-人
- web3.0: 人-网络-人
- 区块链领域的Web3和Web没有关系
- 网络的主体是人，释放人的智慧



# 本章大纲

## □ Web发展简史

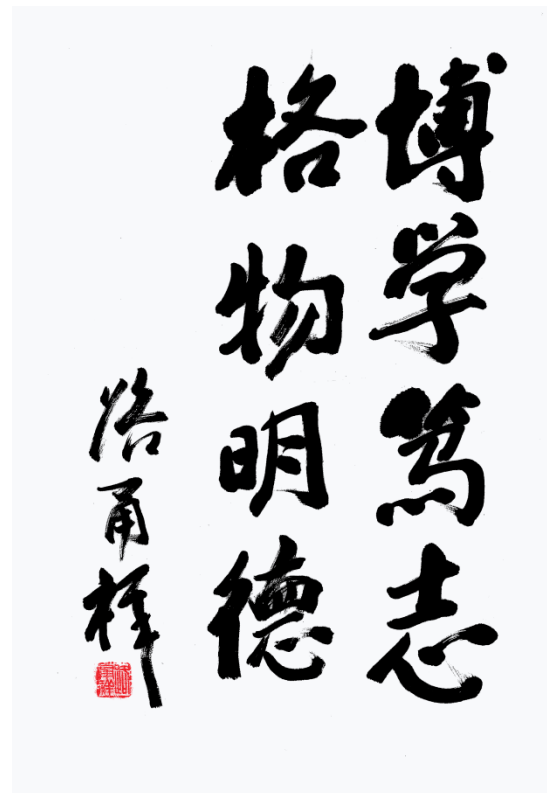
- Web技术发展
- Web 1.0/2.0/3.0

## □ 浏览器发展简史

- 浏览器内核
- 浏览器发展
- 浏览器安全机制

## □ Web安全简史

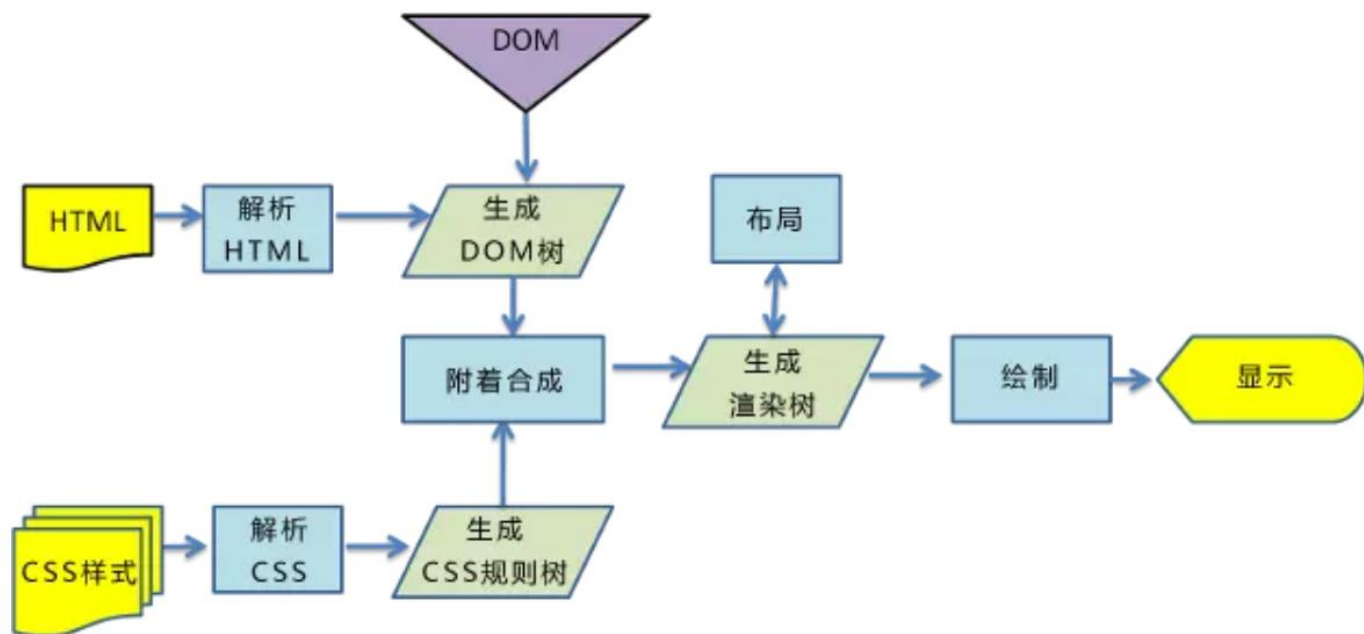
- 框架实例





# 浏览器如何渲染网页

1. 处理 HTML 标记并构建 DOM 树。
2. 处理 CSS 标记并构建 CSSOM (CSS Object Model)树。
3. 将 DOM 与 CSSOM 合并成一个渲染树。
4. 根据渲染树来布局，以计算每个节点的几何信息。
5. 将各个节点绘制到屏幕上。



(webkit渲染引擎流程)

# 浏览器内核

- “Rendering Engine”，渲染引擎
- 对网页语法的解释并渲染（显示）网页
- 不同的浏览器的内核，对于网页的语法解释会有不同，所以渲染的效果也不相同
- 内核的种类很多，可能有 10 多种，常见的浏览器内核：















Trident、Gecko、Blink、Webkit



# 浏览器内核

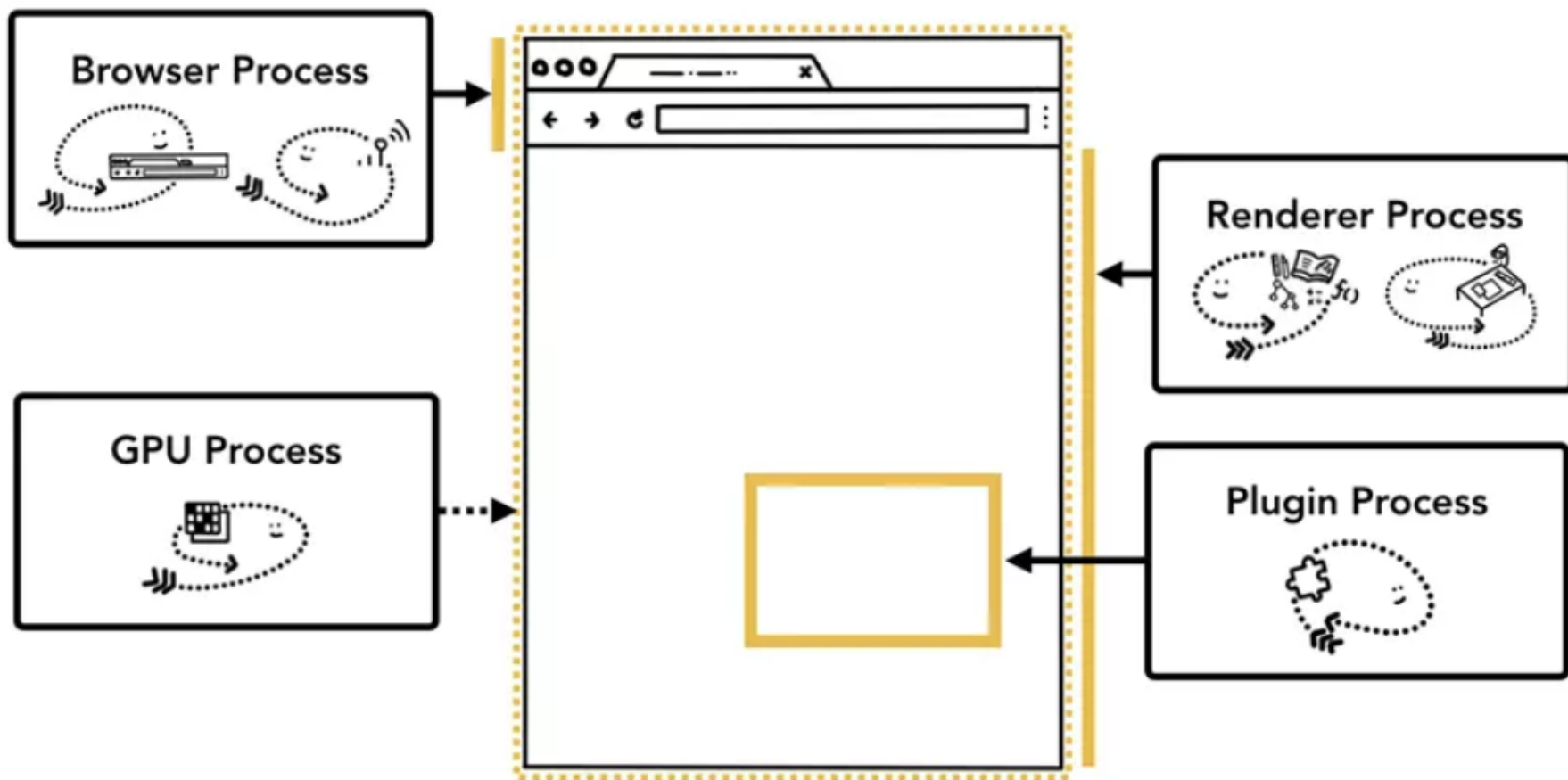


# CHROME 的多进程架构示意图

名称	PID	状态	用户名	CPU	内存(活动的专用工...	UAC 虚拟
 acrotray.exe	5072	正在运行	admin	00	156 K	已禁用
 aesm_service.exe	2624	正在运行	SYSTEM	00	36 K	不允许
 ApplicationFrameH...	12884	正在运行	admin	00	5,692 K	已禁用
 armsvc.exe	5680	正在运行	SYSTEM	00	24 K	不允许
 Calculator.exe	8620	已挂起	admin	00	0 K	已禁用
 chrome.exe	15784	正在运行	admin	00	25,864 K	已禁用
 chrome.exe	12492	正在运行	admin	00	1,784 K	已禁用
 chrome.exe	14764	正在运行	admin	00	83,036 K	已禁用
 chrome.exe	17180	正在运行	admin	00	5,560 K	已禁用
 chrome.exe	2408	正在运行	admin	00	5,332 K	已禁用
 chrome.exe	10128	正在运行	admin	00	8,404 K	已禁用
 chrome.exe	10456	正在运行	admin	00	17,592 K	已禁用
 chrome.exe	6544	正在运行	admin	00	12,608 K	已禁用
 chrome.exe	13464	正在运行	admin	00	7,292 K	已禁用



# CHROME 的多进程架构示意图



《Inside look at modern web browser》

[译] 现代浏览器内部揭秘（第一部分）

<https://juejin.im/post/5b9b0932e51d450e9059c16a>



# 本章大纲

## □ Web发展简史

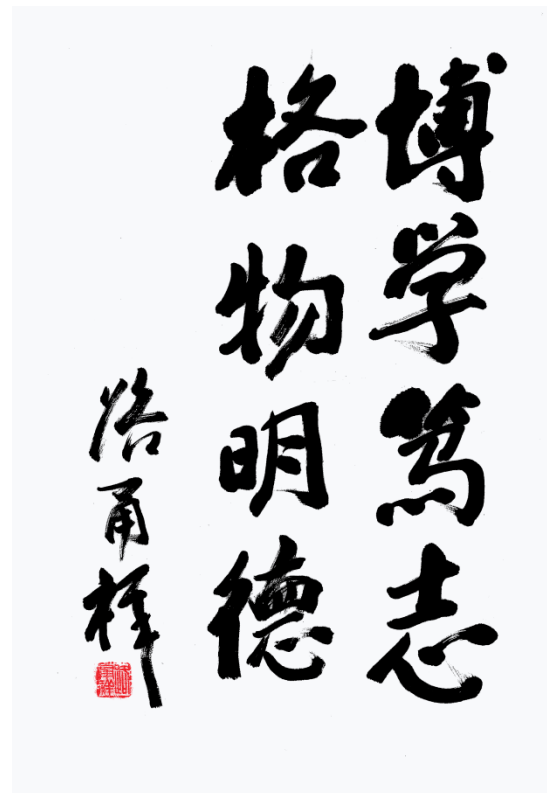
- Web技术发展
- Web 1.0/2.0/3.0

## □ 浏览器发展简史

- 浏览器内核
- 浏览器发展
- 浏览器安全机制

## □ Web安全简史

- 框架实例



# 首款商业化浏览器MOSAIC



- ❑ Mosaic使网页浏览变得美观，让使用者更容易接受
- ❑ Mosaic是另外两个浏览器的基础
  - Mosaic Netscape，后来改名为Netscape Navigator
  - Spyglass Mosaic，后来被微软收购并改名为Internet Explorer
- ❑ 1994年11月14日，Mosaic通讯更名为网景通信
- ❑ 1994年12月15日，网景导航者（Netscape Navigator）推出（1.0版本），迅速占据浏览器市场，并在微软进入浏览器市场之前始终处于统治地位



# 微软进军浏览器市场

- 1995年8月16日，微软在Windows 95 Plus Pack中发布了浏览器1.0，微软将浏览器与操作系统捆绑，并免费为用户提供
- 两年后，微软超越网景成为第一大浏览器厂商





# 第一次浏览器大战

- ❑ Internet Explorer和Netscape Navigator大打出手
  - ❑ 1995年前，网景是互联网浏览器的绝对标准
  - ❑ 遵循W3C标准的IE 4.0赢得了市场，微软投入巨大财力人力
  - ❑ 1998年11月24日，AOL收购网景，但未能帮助Navigator重生。网景被微软打垮
- 各竞争产品快速迭代，疯狂加入新功能，不顾及产品规范标准，擅自调整核心HTML的特性，
  - 牺牲安全性：如果浏览器A可以打开一个有问题的页面，浏览器B却拒绝解析，用户会认为浏览器B有问题，而选择貌似强大的浏览器A
  - 浏览器开发商为了避免掉队，亦步亦趋的跟进



# 平淡期

- 微软一家独大，占据绝大部分市场份额
- 垄断下的自满：IE5之前，每年一个新版本；IE6两年；IE7用了五年！
- 推出了XMLHttpRequest，改变了世界
- 浏览器牺牲了安全性！



# 第二次浏览器大战

□ 2002年前后，蠕虫和浏览器漏洞成为严重安全威胁。**安全问题**引发了第二次浏览器大战

□ IE的竞争者：

- 对标准支持更好，浏览更安全，效率更高
- 2004年，Mozilla Firefox出现，优化了安全性和不兼容性
- 2008年，Chrome出现，简洁易用，速度快
- Safari和Opera在智能手机领域也步步领先



# 小结

- ❑ 回顾浏览器发展历程，由于缺乏统一的远景目标和完整的安全规范，整个发展过程竞争激烈、变换莫测，结果漏洞百出
- ❑ 浏览器安全，以非同寻常的方式演进变化，可能是浏览器安全问题高居不下的原因
- ❑ 安全问题成为打破IE统治地位的关键因素



# 小结

## □ 用户是安全风险的一个环节

- 易用性：满足大量的小白用户
- 用户没有安全上网的概念

## □ Web运行环境难以隔离

- 文档和代码交织在一起

## □ 浏览器安全缺乏统一的格局

- 零零碎碎地小修补



# 本章大纲

## □ Web发展简史

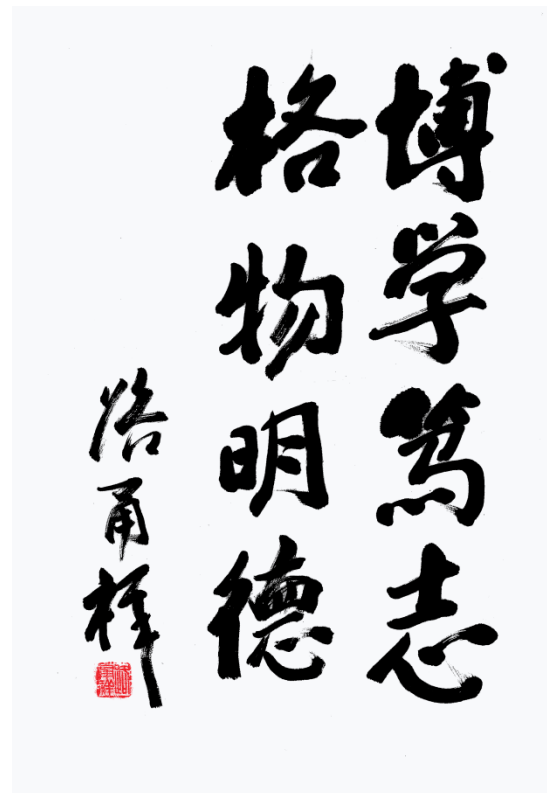
- Web技术发展
- Web 1.0/2.0/3.0

## □ 浏览器发展简史

- 浏览器内核
- 浏览器发展
- 浏览器安全机制

## □ Web安全简史

- 框架实例



# 同源策略

## □ 同源策略(Same-Origin Policy)

## □ 浏览器核心安全功能

{protocol, host, port}

Compared URL	Outcome	Reason
http://www.example.com/dir/page2.html	Success	Same protocol, host and port
http://www.example.com/dir2/other.html	Success	Same protocol, host and port
http://username:password@www.example.com/dir2/other.html	Success	Same protocol, host and port
http://www.example.com:81/dir/other.html	Failure	Same protocol and host but different port
https://www.example.com/dir/other.html	Failure	Different protocol
http://en.example.com/dir/other.html	Failure	Different host
http://example.com/dir/other.html	Failure	Different host (exact match required)
http://v2.www.example.com/dir/other.html	Failure	Different host (exact match required)
http://www.example.com:80/dir/other.html	Depends	Port explicit. Depends on implementation in browser.



# 沙箱 (SANDBOXING)

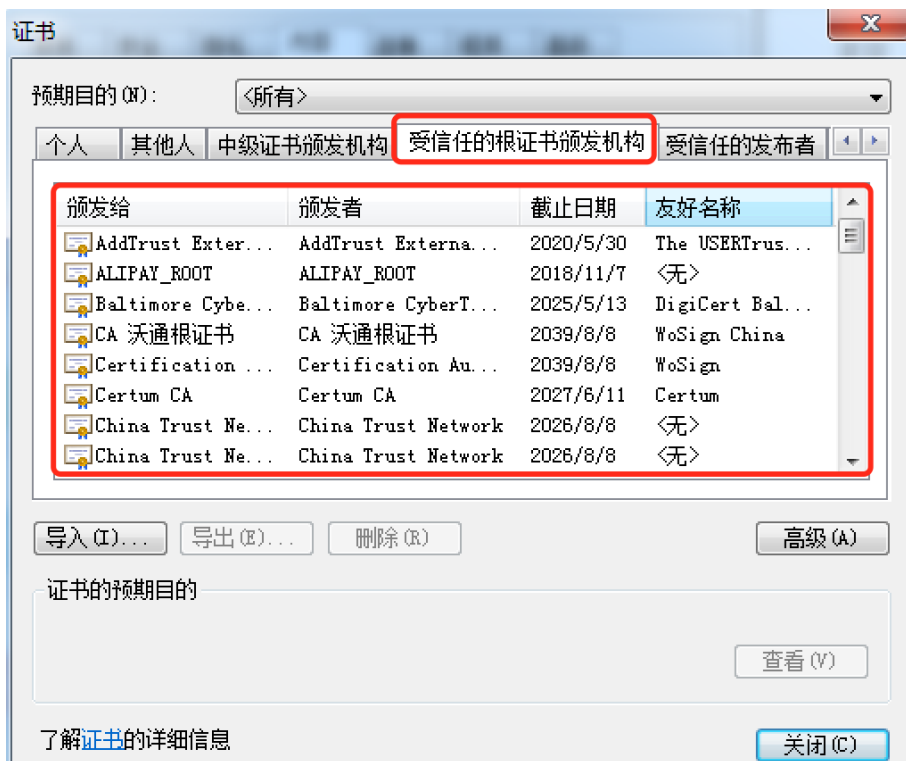
- 安全的虚拟运行环境，与真实环境隔离
- 隔离对象/线程/进程，控制浏览器访问系统资源的权限
- 保护用户的系统不被网页上的恶意软件侵入，保护用户系统的输入事件不被监视、保护用户系统中的文件不被偷取





# 根证书

- ❑ 数字证书不可篡改，标识网站身份
- ❑ 证书的信任链可传递，浏览器默认携带若干“值得信任”的根证书



证书路径(P)

GlobalSign Root CA - R1  
GlobalSign Organization Validation CA - SHA256 - G2  
[baidu.com](http://baidu.com)

查看证书(V)



# WEB发展简史

## □ 安全问题

- 沙箱逃逸
- 恶意浏览器扩展
- 数字证书的窃取和伪造



# 本章大纲

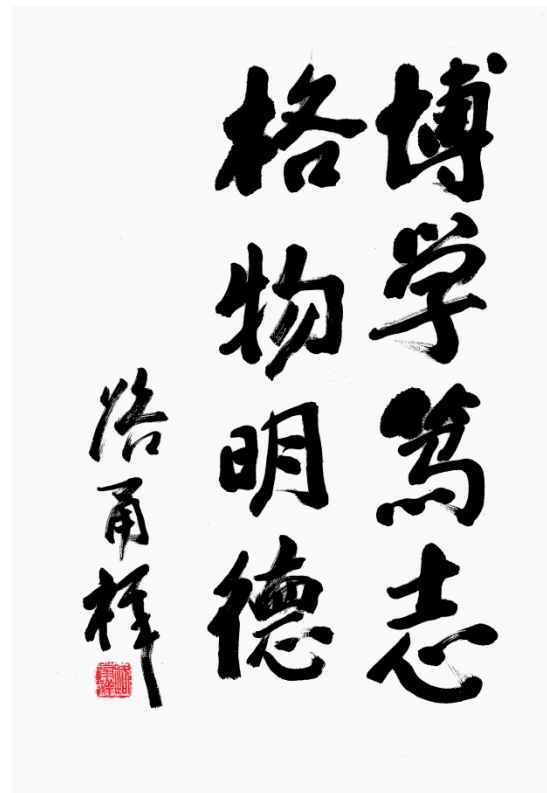
## □ Web发展简史

- Web技术发展
- Web 1.0/2.0/3.0

## □ 浏览器发展简史

- 浏览器内核
- 浏览器发展
- 浏览器安全机制

## □ Web安全简史



# WEB安全简史

## □ Hacker

- 起初，研究计算机系统和网络的人，被称为“Hacker”
- 他们对计算机系统有着深入的理解，因此往往能发现其中的问题
- 黑客的本质：精通系统和网络的人，最优秀的程序员



# WEB安全简史

- 黑客技术的发展历程（1）
- 攻击操作系统、应用软件、网络服务
- Web攻防技术处于非常原始的阶段
  - Web并非主流应用
  - 攻击系统软件可以直接获得root权限



# WEB安全简史

## □ 黑客技术的发展历程（2）

- 直接暴露在互联网上的系统得到保护
- 网络防护增强：防火墙，访问控制列表（Access Control List）
- 直接攻击系统软件并不容易了，非Web服务越来越少

### 2003年冲击波蠕虫

- 针对Windows的RPC服务（445端口）
- 短时间内席卷全球，造成数百万台机器被感染
- 运营商坚决地在骨干网上屏蔽了135、445等端口
- 互联网对于安全的重视空前



# WEB安全简史

## □ 黑客技术的发展历程（3）

### □ Web 1.0时代：关注服务器端动态脚本安全

- SQL注入、文件上传、文件包含

### □ Web 2.0时代：关注客户端安全

- XSS、CSRF、ClickJacking



# WEB安全简史

## □ 黑客的帽子

□ 黑客世界中，以“帽子”颜色比喻黑客的“好坏”??

□ 白帽子：精通安全技术，工作在反黑客领域的安全专家们。

□ 让网络变得更安全

□ 黑帽子：利用黑客技术造成破坏，甚至造成网络犯罪的群体。

□ 让网络变得更不安全

□ 灰帽子：业余爱好者或者是义务工作者，警告或示警，没有恶意

□ 红帽子：以正义、道德、进步、强大为宗旨，以热爱祖国、坚持正义、开拓进取为精神支柱





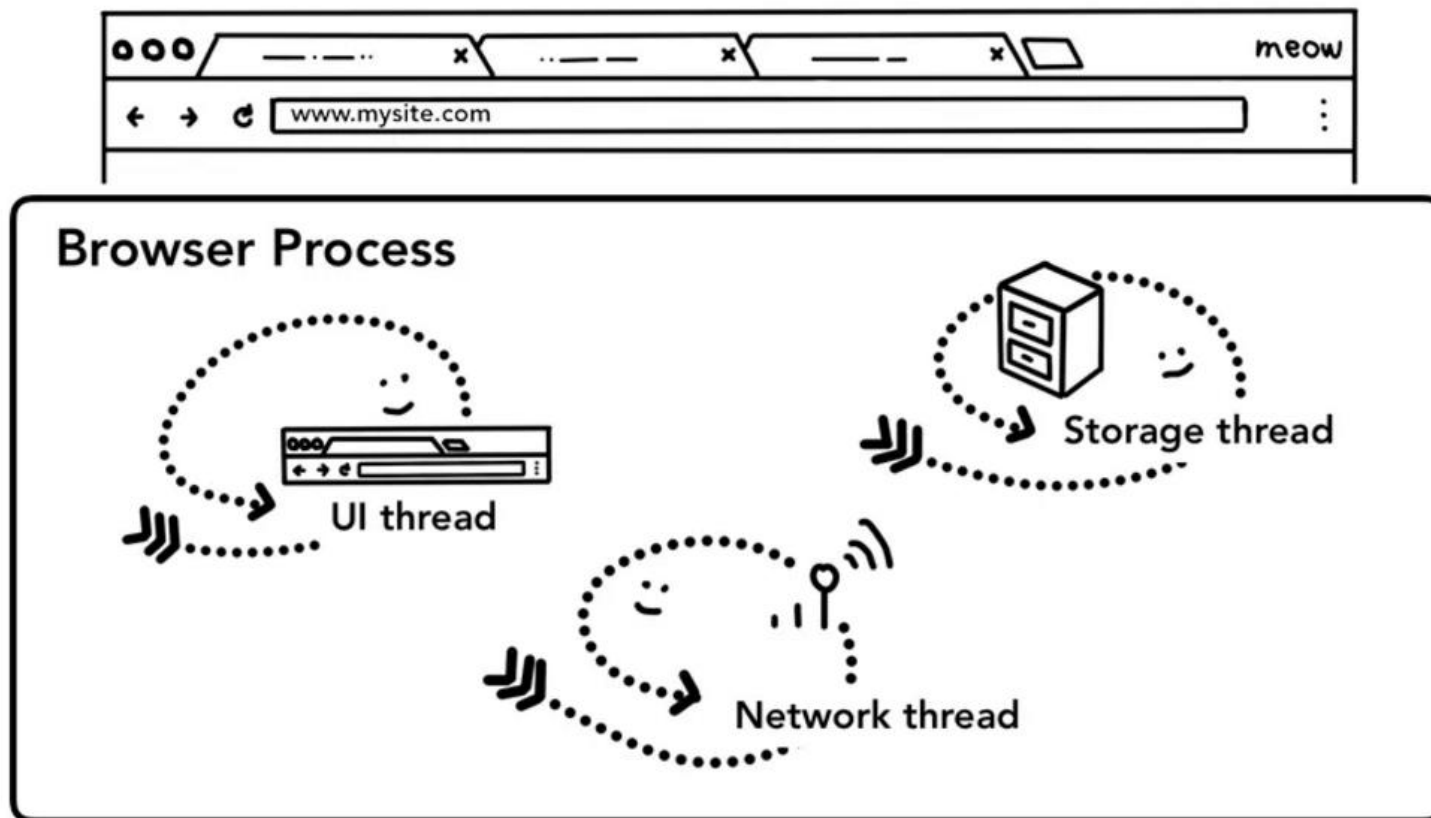
# 小结

- Web已经成为了互联网上最为重要的应用
- Web的技术一直在快速地发展：编程语言、开发框架
- 从Web 1.0到2.0，以及未来的3.0，人创造了丰富多彩的Web
- 客户端的安全，浏览器的安全，越来越受到重视
- 白帽与黑帽，一念之间



# 延伸阅读

## 《Inside look at modern web browser》



图片引自Mariko Kosaka的《Inside look at modern web browser》

# 后续课程内容

## □ 第一部分：基础知识

□ 介绍Web安全定义与内涵，国内外现状与趋势、近年来重大网络安全事件等，以及本课程可参考的书籍和网络资源；介绍本课程所需掌握的基础知识，包括HTTP/HTTPS协议、Web前后端编程语言、浏览器安全特性等。

### □ 1.1 绪论

### □ 1.2 Web的简明历史

### □ 1.3 HTTP与Cookie

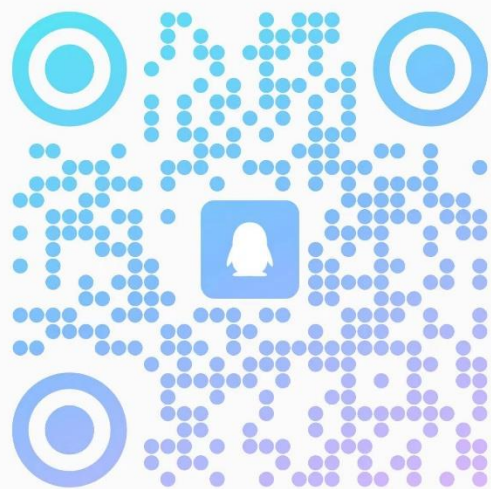
### □ 1.4 同源策略





[2023秋]Web Security

群号: 920050957



扫一扫二维码，入群聊



# 谢谢大家

刘潮歌

liuchaoge@iie.ac.cn

中科院信工所 第六研究室



**中国科学院大学**  
University of Chinese Academy of Sciences