

河北师大软件学院 @Software College

---

# 前端开发与HTML5 程序设计基础

---

王岩

## 2.6 Web核心技术（一）

### HTTP协议基础

什么是协议？



---

# 网络协议 (networking protocol)

---

- ❖ 通信计算机双方必须共同遵从的一组约定。如怎么样建立连接、怎么样互相识别等。只有遵守这个约定，计算机之间才能相互通信交流。

---

# 常见应用层协议

---

- ❖ HTTP
- ❖ DNS
- ❖ FTP
- ❖ POP3、SMTP、IMAP
- ❖ SSH
- ❖ DHCP
- ❖ ...

# HTTP协议



---

# HTTP协议

---

- ❖ HTTP：超文本传送协议(HyperText Transfer Protocol)
- ❖ 万维网(Web)的基础
- ❖ 位于TCP/IP协议栈的应用层

---

# HTTP协议版本

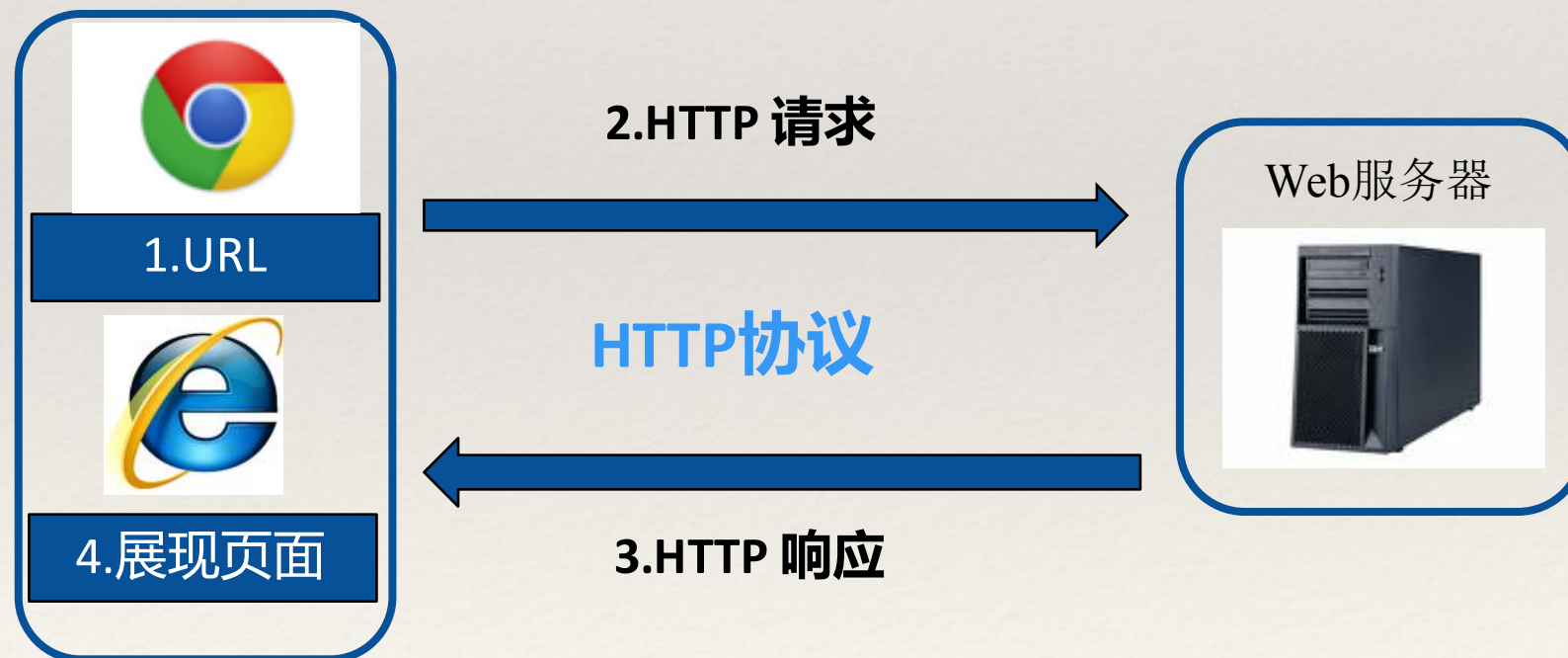
---

- ❖ 0.9
- ❖ 1.0
- ❖ 1.1（目前通用）
- ❖ 2.0（2015年5月发布）



# Web访问过程

- ❖ HTTP协议管理：
  - ❖ Web浏览器如何向web服务器请求内容
  - ❖ Web服务器如何将请求内容响应给Web浏览器



访问之前

---

# 指定访问URL

---

- ❖ URL (Uniform Resource Locator)

- ❖ 示例: `http://name:pass@www.aa.com:80/dir/file.php?var=value#frag`

- ❖ `http` 协议名称

- ❖ `name` 用户名 (可选)、`pass` 密码 (可选)

- ❖ `www.vimer.cn` 主机网络地址

- ❖ `80` 端口号 (可选)

- ❖ `/dir/file.php` 资源路径

- ❖ `var=value` 查询字符串 (可选)

- ❖ `frag` 锚点 (可选)



---

# 访问方式

---

- ❖ HTTP协议特征：请求 / 响应模式
  - ❖ 依据请求返回响应
  - ❖ 无请求时，不会返回响应

发送请求

---

# 请求的发起

---

- ❖ 用户
  - ❖ 手工输入URL
  - ❖ 点击超链接（超链接指向URL）
  - ❖ 表单提交（表单指向URL）
- ❖ JS程序
  - ❖ 自动控制跳转（程序指定URL）
  - ❖ Ajax请求（程序指定URL）



---

# HTTP请求

---

- ❖ 请求方式：浏览器发送请求消息
- ❖ 消息结构
  - ❖ 请求行
  - ❖ 请求头
  - ❖ 空行
  - ❖ 请求体

---

# 请求行

---

- ❖ 请求消息的第一行
- ❖ 形式如：GET /user/login.php HTTP/1.1
  - ❖ GET：请求方法
  - ❖ /user/login.php：请求资源路径
  - ❖ HTTP/1.1：协议名称及版本

---

# 请求头

---

## ❖ 请求头：一些客户端属性信息

```
accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
accept-encoding: gzip, deflate, sdch
accept-language: zh-CN,zh;q=0.8,en;q=0.6,zh-TW;q=0.4
cache-control: max-age=0
cookie: NID=77=tJH-mStLBPxT6ovgtP0r0e3693XAi4U3rr5QNcUy9H1brQ3GJR5UXmpFETPN-kgmeJ1Qwf0h5uwK
10G0xv7Ue73ByrNIvmzfqWkHM_3rKUBc2ThFbf6Uj3G_uNv3WE7xqf-ToAE0pW0
user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_3) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/43.0.2357.130 Safari/537.36
x-client-data: CKK2yQEIXLbJAQjhlMoBCP2VygE=
```



---

# 请求头

---

- ❖ 常用属性信息
  - ❖ User-Agent: 客户端环境信息
  - ❖ Cookie
  - ❖ Host: 主机名
- ❖ 其他属性信息: Accept、Accept-Language、Accept-Encoding...

---

# 请求体

---

- ❖ GET请求
  - ❖ 请求体为空
- ❖ POST请求
  - ❖ 请求体为向服务器发送的数据
    - ❖ 例如：表单数据、文件...

---

# HTTP请求方法

---

- ❖ **GET**: 向指定的资源发出获取请求。使用GET方法应该只用在读取数据。
- ❖ **POST**: 向指定资源提交数据，请求服务器进行处理（例如提交表单或者上传文件）。
- ❖ 其他请求方法
  - ❖ HEAD
  - ❖ OPTIONS
  - ❖ PUT
  - ❖ DELETE
  - ❖ TRACE
  - ❖ CONNECT



返回响应

---

# HTTP响应

---

- ❖ Web服务器处理完成请求后返回响应消息
- ❖ 消息结构：
  - ❖ 状态行
  - ❖ 响应头
  - ❖ 空行
  - ❖ 响应主体

---

# 状态行

---

- ❖ 响应消息的第一行
- ❖ 形式如：HTTP/1.1 200 OK
  - ❖ HTTP/1.1：协议版本
  - ❖ 200：状态码
  - ❖ OK：状态消息



---

# 状态码

---

- ❖ 1XX 提示信息 - 表示请求已被成功接收，继续处理
- ❖ 2XX 成功 - 表示请求已被成功接收，理解，接受
- ❖ 3XX 重定向 - 要完成请求必须进行更进一步的处理
- ❖ 4XX 客户端错误 - 请求有语法错误或请求无法实现
- ❖ 5XX 服务器端错误 - 服务器未能实现合法的请求

---

# 状态码

---

- ❖ 200 OK : 请求被成功地完成, 所请求的资源发送回客户端
- ❖ 302 Found : 重定向, 新的URL会在response中的Location中返回, 浏览器将会使用新的URL发出新的Request
- ❖ 304 Not Modified : 文档已经被缓存, 直接从缓存调用
- ❖ 400 Bad Request : 客户端请求与语法错误, 不能被服务器所理解
- ❖ 403 Forbidden : 服务器收到请求, 但是拒绝提供服务
- ❖ 404 Not Found : 请求资源不存在
- ❖ 500 Internal Server Error : 服务器发生了不可预期的错误
- ❖ 503 Server Unavailable : 服务器当前不能处理客户端的请求, 一段时间后可能恢复正常

---

# 响应头

---

- ❖ 响应头：返回资源的相关属性信息

```
cache-control: private, max-age=0  
content-encoding: gzip  
content-type: text/html; charset=UTF-8  
date: Tue, 15 Mar 2016 12:56:27 GMT  
expires: -1  
server: gws  
status: 200  
x-frame-options: SAMEORIGIN  
x-xss-protection: 1; mode=block
```



---

# 响应头

---

- ❖ 常用属性信息
  - ❖ Content-Type: 资源类型（字符集）
    - ❖ 例如: `text/html; charset=UTF-8`
  - ❖ Location: 重定向新URL
  - ❖ Set-Cookie: 写入Cookie
  - ❖ Cache-Control: 缓存方式
  - ❖ Expires: 缓存过期时间
- ❖ 其他属性信息: Date、Content-Encoding、Server...

---

# 响应主体






















---

- ❖ 请求资源的数据
  - ❖ 如：页面源代码

# 浏览器解析



1. 浏览器得到源文件进行解析
2. 源文件如果连接了其他资源，继续依据资源URL进行
  - ❖ 如：外部CSS文件、外部JS、图片、音视频文件

Name	Method	Status	Type	Initiator	Size	Time	Timeline	1.00 s
 ?gws_rd=cr,ssl	GET	200	document	?gws_rd=cr,ssl:263	57.6 KB	648 ms		
 nav_logo242.png	GET	200	png	/?gws_rd=cr,ssl:53	(from cac...)	0 ms		
 googlelogo_color_272x92dp.png	GET	200	png	/?gws_rd=cr,ssl:53	(from cac...)	0 ms		
 l1_1967ca6a.png	GET	200	png	/?gws_rd=cr,ssl:53	(from cac...)	0 ms		
 rs=ACT90oERhi8s6rrjq4pD8z4li0jtsUd1Tg	GET	200	script	?gws_rd=cr,ssl:53	(from cac...)	0 ms		
 tia.png	GET	200	png	rs=ACT90oERhi8...	(from cac...)	0 ms		
 data:image/png;base...	GET	(data)	png	rs=ACT90oERhi8...	(from cac...)	0 ms		
 data:image/gif;base...	GET	(data)	gif	rs=ACT90oERhi8...	(from cac...)	0 ms		
 rs=ACT90oERhi8s6rrjq4pD8z4li0jtsUd1Tg	GET	200	script	rs=ACT90oERhi8...	(from cac...)	0 ms		
 gen_204?v=3&s=webhp&atyp=csi&ei=BwzoVv...	GET	204	text/html	Other	48 B	343 ms		
 rs=AA2YrTtNM9DxjujqugaNyuU27_lOrKt1Pg	GET	200	script	?gws_rd=cr,ssl:219	(from cac...)	0 ms		
 cb=gapi.loaded_0	GET	200	script	rs=AA2YrTtNM9D...	(from cac...)	0 ms		

---

# 扩展知识点

---

- ❖ Web速度优化方式之一
  - ❖ 减少HTTP请求数
- ❖ CDN
  - ❖ Content Delivery Network（内容分发网络）
  - ❖ 部署节点服务器，使用户就近获取资源
  - ❖ 降低应用的主Web服务器压力

# PHP控制响应头



---

# 响应头信息的来源

---

- ❖ Web服务器设定
- ❖ 程序控制
  - ❖ header( ) 函数

---

# 常用响应头设置

---

- ❖ 设置文档类型及字符集

- ❖ 例：header('Content-Type:text/html;charset=utf-8');

- ❖ Demo2-6-1

- ❖ Demo2-6-2

- ❖ 设置页面重定向

- ❖ 例：header('Location:http://www.sina.com');

- ❖ Demo2-6-3

---

# 相关配置

---

- ❖ “header already sent...”的错误提示
  - ❖ 原因：响应头返回早于响应体返回，故设置响应头的代码之前不能有内容输出（HTML代码、文本信息、程序输出）
  - ❖ 解决方法：设置php.ini，开启输出缓存
    - ❖ `output_buffering = 4096`



---

# 示例：生成Excel文件

---

❖ Demo2-6-4

谢谢！