

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 (三)

状态维持

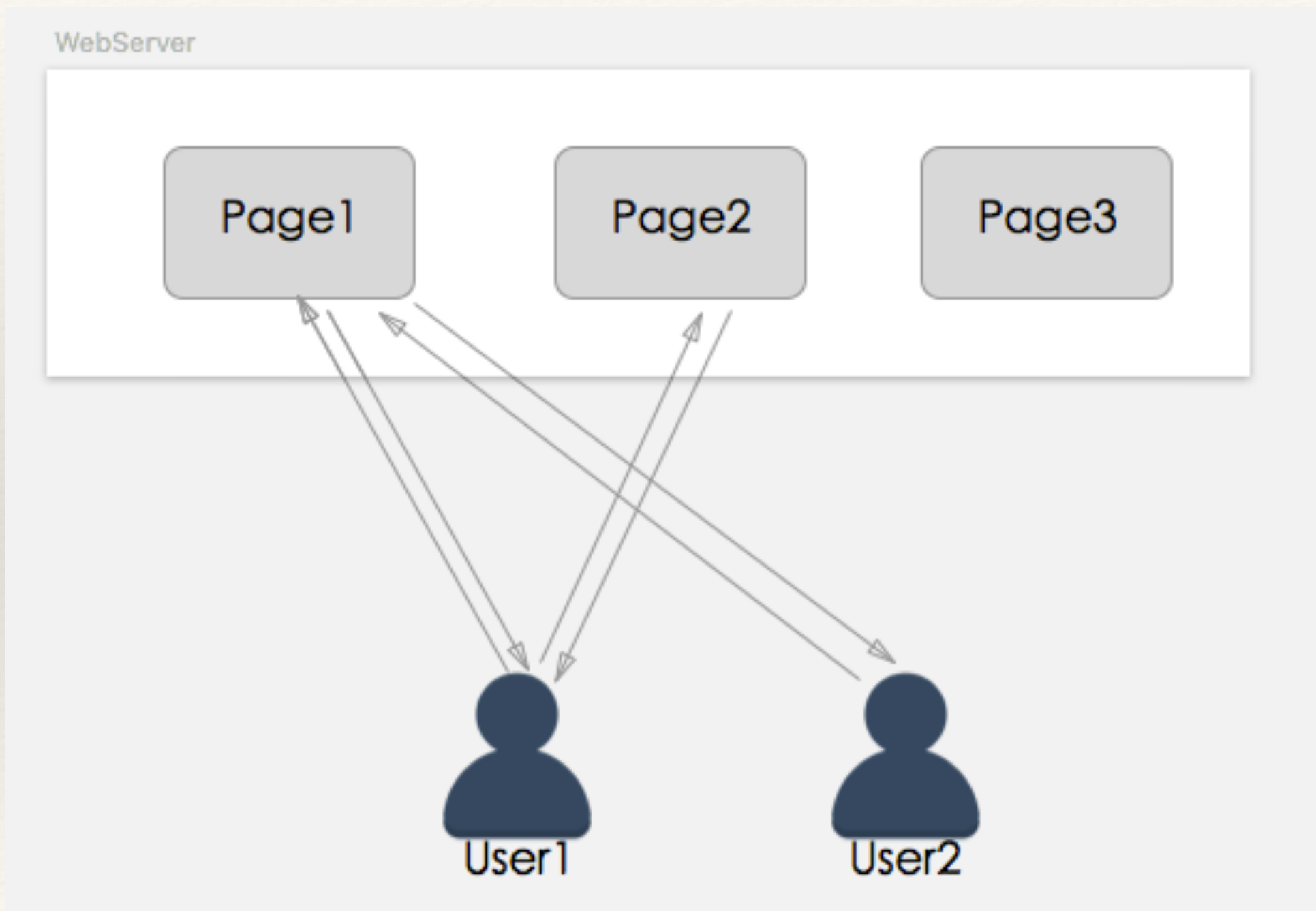
什么是状态?

- ❖ 访问网页的用户是否已登录?
- ❖ 当前用户购买了哪些商品?
- ❖ 用户访问过哪些页面?
- ❖

HTTP协议特征

- ❖ 无状态
 - ❖ 每次请求都与服务器建立新连接
 - ❖ 响应结束后与服务器断开连接
 - ❖ 多次请求间无关系

HTTP协议特征



无状态的影响

- ❖ 如何实现用户登录?
 - ❖ 依靠协议? (不可行)
 - ❖ 依靠变量? (临时存储数据, 以页面为单位)
- ❖ 问题关键
 - ❖ 如何识别每一个访问者?
 - ❖ 同一访问者在不同页面之间如何传递数据 (状态信息)?

Cookie

Cookie简介

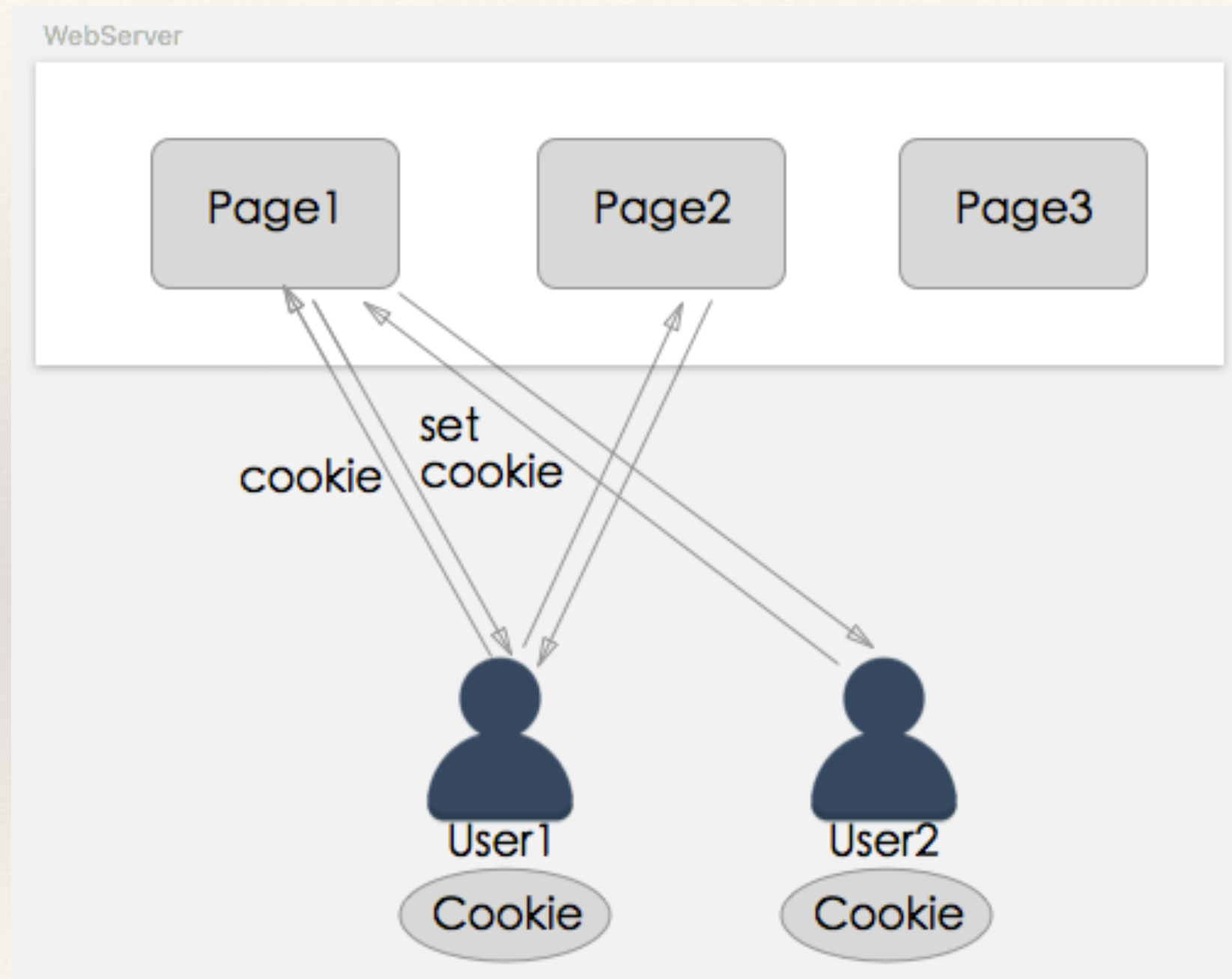
❖ Cookie

- ❖ 网站程序存储在客户端的数据
- ❖ 当用户访问该网站时，自动携带Cookie数据
- ❖ 服务器可以读取Cookie的值以获取所需的信息

❖ 演示

- ❖ 查看本地Cookie文件
- ❖ 查看HTTP请求头信息中的Cookie

Cookie图示



PHP中使用Cookie

- ❖ 设置Cookie（存储数据）
- ❖ 获取Cookie数据（获取已存储数据）
- ❖ 删除Cookie

设置Cookie

- ❖ `setcookie()`
- ❖ Demo2-6-14
- ❖ 注意：
 - ❖ `setcookie()`会修改响应头，可能引发“Can not modify header, header already sent at...”警告。需要开启输出缓存（类似`Header()`函数）。

获取Cookie数据

- ❖ 使用超全局数组：\$_COOKIE[]
 - ❖ 数组键名为Cookie名称
- ❖ Demo2-6-15
- ❖ Demo2-6-16
- ❖ 注意
 - ❖ 服务器只能读取本服务器设置的Cookie

删除Cookie

- ❖ 使用setcookie()函数，将要删除的Cookie设置为NULL即删除了该Cookie
- ❖ Demo2-6-17

练习

- ❖ Demo2-6-18: 使用Cookie完成用户登录、注销功能

Cookie特征

- ❖ 设置Cookie值，总是在下次请求中才能能在\$_COOKIE中取到。

COOKIE实际应用

- ❖ 用户登录
- ❖ 用户未登录状态下，数据记录
 - ❖ 商品浏览历史
 - ❖ 页面浏览历史
- ❖ 其他与客户端相关状态存储



COOKIE在客户端

- ❖ JS读写cookie
 - ❖ document.cookie
- ❖ Demo2-6-19
- ❖ 插件：jQuery.cookie

练习

- ❖ Demo2-6-20: 实现“记住用户名”功能

Cookie缺点

- ❖ 浏览器限制Cookie容量（通常为4K）
- ❖ COOKIE数据安全性
 - ❖ 客户端存储数据暴露
 - ❖ 网络传输过程中数据暴露
- ❖ 用户可以在浏览器端禁用COOKIE

Session

Session简介

- ❖ Session（会话）：实现状态维持的一种技术
- ❖ Session把数据存储在服务器端
 - ❖ 相对于Cookie来说Session更安全
 - ❖ 能存储更多的数据

PHP中使用Session

- ❖ 如何使用Session实现状态维持?
 - ❖ 开启Session （开启 / 恢复会话）
 - ❖ 设置会话数据 （向Session里存储数据）
 - ❖ 访问会话数据 （获取设置的Session数据）
 - ❖ 删除Session数据

开启会话

- ❖ 开启会话

- ❖ 要使用Session的每个页面都必须先调用session_start()方法，来创建新会话或者恢复已有会话

- ❖ session_start()

- ❖ 注意：

- ❖ 和header()函数以及setCookie()函数一样，session_start()函数也必须放所有输出代码之前，同样，如果设置了php.ini的output_buffering则可以取消这个限制

存储/访问会话变量

- ❖ 使用超全局数据\$_SESSION[]
- ❖ 设置会话变量
 - ❖ 直接向\$_SESSION中添加元素
- ❖ 访问会话变量
 - ❖ 直接访问\$_SESSION中的元素
- ❖ Demo2-6-21

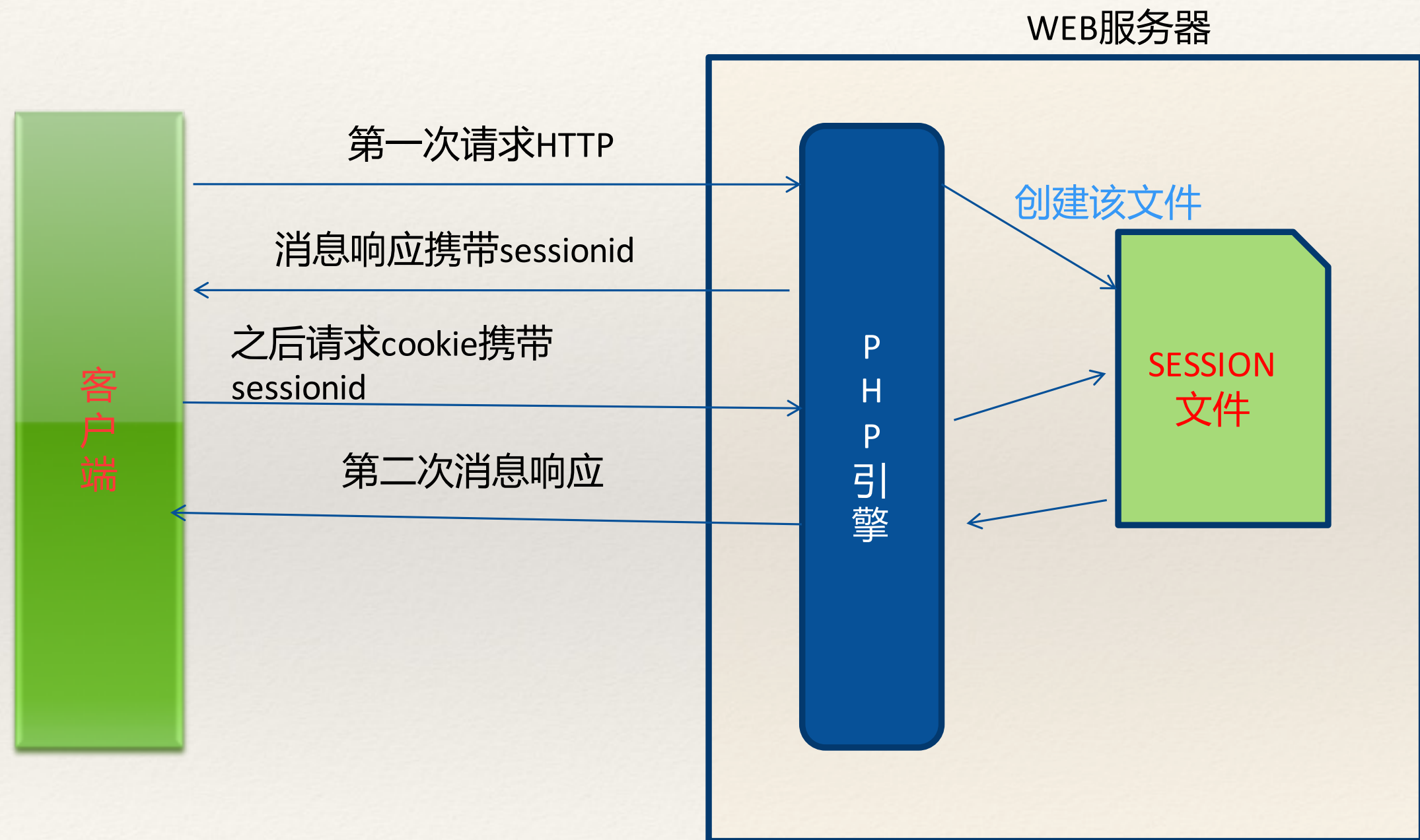
删除Session

- ❖ 删除单个会话变量
 - ❖ 使用`unset()`删除某个单独的会话变量
- ❖ 删除本站点全部会话变量
 - ❖ 通过重置`$_SESSION`数组为空数组
 - ❖ 使用`session_destroy()`
- ❖ Demo2-6-22

练习

- ❖ Demo2-6-23: 使用Session完成用户登录、注销功能

Session工作原理



禁用Cookie时，Session处理

- ❖ 禁用Cookie不可以使用Session
- ❖ 解决方案：
 - ❖ 设置PHP.ini，更改session.use_trans_sid=on参数
 - ❖ 无法更改php.ini：URL重写，
output_add_rewrite_var('PHPSESSID',session_id ());

Cookie、Session总结

❖ 共同点：通过存储特定的用户信息来实现状态维持

	COOKIE	SESSION
存放位置	客户端	服务器端
大小	有限制	没有限制
安全性	SESSION的安全性更好些	
❖ 不同点：	过期时间	程序设定

练习

- ❖ 实现基于Session的用户登录有效期控制
 - ❖ 完善Demo2-6-23，实现用户登录时，通过表单选项设置登录有效期（1小时、1天、1周）。
 - ❖ 提示：使用Session时，客户端Cookie存储SessionID，可以手工设置该Cookie的有效期来控制会话有效期。
 - ❖ session_name()函数可以取到session名称
 - ❖ session_id()函数可以取到session ID

谢谢！