

# HTML5网页设计初窥（8）

苏鹏  
ASP.NET MVP

# 独立数据存储

# 内容

- 离线数据存储
- 离线应用的开发

# 为何要有独立存储

- 带宽日渐增长，但是应用日趋复杂
- Cookies不给力
  - 8kb
  - 明文
- 需要新的独立存储方案



# 存储发展

- Flash LSO 默认100k最大10MB
- Silverlight 独立存储 默认1MB
- <http://dev.w3.org/html5/webstorage/>

# Storage接口

- 无论是sessionStorage还是localStorage

属性/方法	返回值	描述
length	<i>integer</i>	包含对象数目，此属性只读
key(n)	<i>DOMString</i>	数组对象返回值
getItem(key)	<i>data</i>	获得某对象值
setItem(key,data)	<i>void</i>	设置某对象
removeItem(key)	<i>void</i>	移除某对象
clear()	<i>void</i>	清空

- 存储结构本质上仍然是个哈希表

# 补充细节

- 变量使用UTF-16，即使空字符也可以当变量名
- 访问变量方式
  - `setItem()`
  - `getItem()`
  - `localStorage.currentTemp = 18`

# 遍历所有存储变量

- 遍历数组的办法

```
for (var i=0;i<localStorage.length;i++) {  
    var item =  
    localStorage.getItem(localStorage.key(i));  
    alert("Found item "+item);  
}
```

- 目前只支持字符类型，要转换

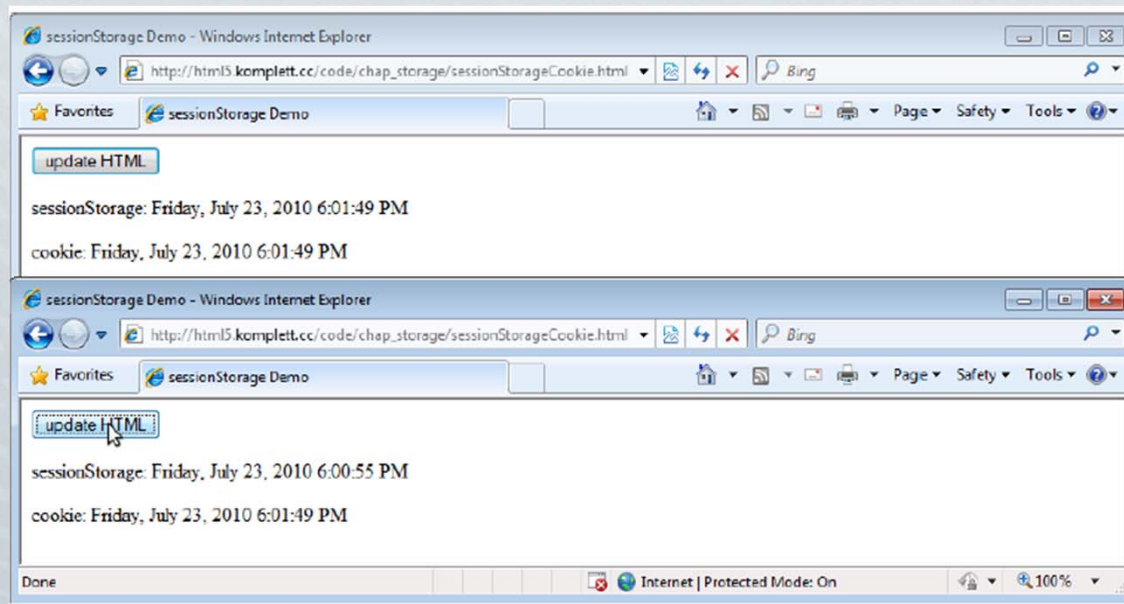
JSON.stringify(itemsObject)

- 建议空间为5MB



# sessionStorage

- Cookies对站点绑定，多个浏览器实例共享一个
- sessionStorage对浏览器绑定（无论新窗口或者新tab）



# 代码解说

## – 读cookies的会变

```
window.onload = function() {  
    var currDate = new Date();  
    sessionStorage.setItem("currenttime",  
        currDate.toLocaleString());  
    document.cookie =  
        "currenttime="+currDate.toLocaleString();  
    updateHTML();  
}  
function updateHTML() {  
    document.getElementById("currenttime").innerHTML =  
        sessionStorage.getItem("currenttime");  
    document.getElementById("currtimeCookie").innerHTML  
        = getCookie("currenttime");  
}
```

# localStorage

- 不同于SessionStorage，localStorage是针对协议、域名、端口的，类似于全局应用

```
<script type="text/javascript">  
localStorage.lastname="Smith";  
document.write(localStorage.lastname);  
</script>
```

# 存储事件

— 尚未形成标准



# 调试

- Alert
- 调试工具（opera演示）

# 离线应用

- Offline Web applications
- 使用.appcache保存

# 缓存文件

- 使用UTF-8编码
- 以CACHE MANIFEST开头
- 三个基本部分

CACHE MANIFEST

menu.html

menu.js

# login requires network  
connection

NETWORK:

login.php

FALLBACK:

/ /menu.html

CACHE:

style/innbar.css

# 缓存文件

- NETWORK:此类文件需要从网络访问
- FALLBACK:缓存找不到的时候显示它
- CACHE:真正要缓存的内容



# 离线状态

## – window.applicationCache

取值	名称	含义
0	UNCACHED	页面完全没有被缓存，可能是没被设定或者尚未被下载
1	IDLE	已经被下载并可访问
2	CHECKING	正在检查缓存是否需要更新
3	DOWNLOADING	发现新版本正在更新
4	UPDATEREADY	已经下载完成，下次离线访问就可用
5	OBSOLETE	缓存清单文件访问失败，此页面会被从缓存中清除

`window.applicationCache.status`

# 离线事件

```
window.applicationCache.addEventListener("progress",  
function(e) {  
    alert("New file downloaded");  
}, false);
```

名称	描述
checking	检查在manifest里面的文件是否有更新
noupdate	没有找到新文件
downloading	下载中
progress	任何文件被下载这一事件都会被唤起
cached	所有缓存都被下载完
updateready	由于manifest，所有文件被从新下载完毕
obsolete	由于manifest清空，文件找不到
error	各种错误，比如manifest文件格式错

# 更新缓存方法

- JS中提供update() 和swapCache()直接更新缓存

```
<button  
onclick="window.applicationCache.update();">  
update applicationCache</button>  
  
window.applicationCache.addEventListener("updateready",  
function(e) {window.applicationCache.swapCache();  
alert("New Cache in action");  
}, false);
```

- 可以使用SetInterval()定期唤醒更新

# 检测是否在线

- window.navigator.onLine

```
window.addEventListener("online", function() {  
    alert("You are now online");  
}, false);  
window.addEventListener("offline", function()  
{  
    alert("You are now OFFLINE");  
}, false);
```

- 不测路由，只测连接，所以可能联网但不能上网



# 调试

- 一定小心缓存

# 总结

- 离线数据存储
- 离线应用的开发
- 苏鹏 supper3000@126.com

# 获取更多MSDN资源

- **MSDN 中文网站**  
<http://msdn.microsoft.com/zh-cn/>
- **MSDN Webcast 中文网络广播**  
<http://msdn.microsoft.com/zh-cn/dd796167>
- **MSDN 免费中文速递邮件**  
<http://msdn.microsoft.com/zh-cn/aa570311>
- **MSDN 开发中心**  
<http://msdn.microsoft.com/zh-cn/aa937802.aspx>
- **Internet Explorer 9 开发中心**  
<http://msdn.microsoft.com/zh-cn/ie/>
- **HTML5 学习资源**  
<http://msdn.microsoft.com/zh-cn/aa740476.aspx>



**Microsoft®**

Be what's next.™