

# 浅析浏览器缓存机制

最近学习了浏览器缓存机制的相关内容，在此归纳总结。

浏览器缓存主要指基于 HTTP 协议定义的缓存机制，通过 HTTP 响应头在服务端设置，Expires、Cache-Control、Last-Modified、ETag 是 RFC 2616 (HTTP/1.1) 协议中和网页缓存相关的几个字段。其中：

- (1) Expires、Cache-Control 用来控制缓存的失效日期，浏览器可通过它来判定，需不需要发出 HTTP 请求；
- (2) Last-Modified、ETag 用来验证网页的有效性，服务端利用它来验证这个文件是否需要重新返回，应与 Expires/Cache-Control 结合使用。

字段	作用
Expires	指定缓存的过期时间
Cache-Control	指定请求和响应应遵循的缓存机制 <ul style="list-style-type: none"><li>• 常用于响应头 <b>Cache-Control: max-age=xxx</b></li></ul>
Last-Modified/If-Modified-Since	一对验证文件修改时间的响应/请求头 <ul style="list-style-type: none"><li>• 响应头 <b>last-Modified</b>：响应资源的最后修改时间</li><li>• 请求头 <b>If-Modified-Since</b>：值等于 <b>last-Modified</b></li></ul>
ETag/If-None-Match	一对验证文件实体标记的响应/请求头 <ul style="list-style-type: none"><li>• 响应头 <b>Etag</b>：响应资源的唯一标识</li><li>• 请求头 <b>If-None-Match</b>：值等于 <b>Etag</b></li></ul>

## 1. Expires 和 Cache-Control

Expires 规定了缓存失效时间，而 Cache-Control 的 max-age 规定了缓存有效时间（单位：s），理论上这两个值计算出的有效时间是相同的（实际有误差）。如果 Cache-Control 的 max-age 和 Expires 同时

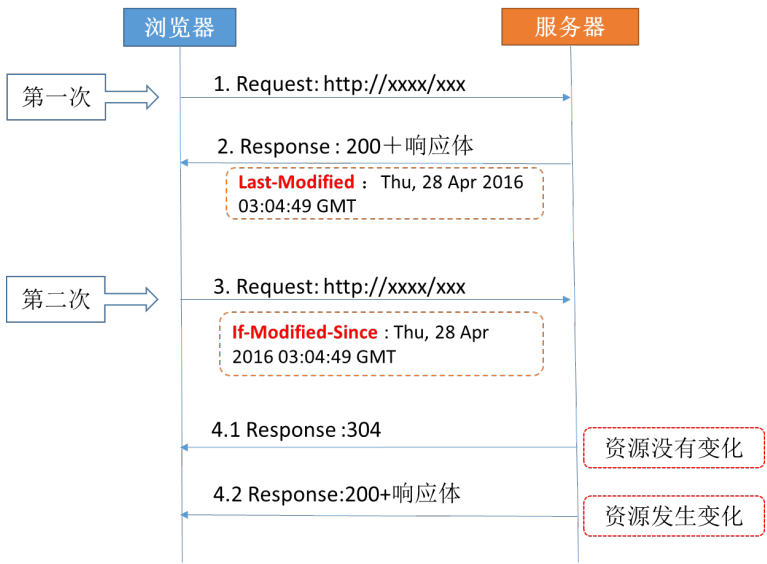
存在，前者优先级高于后者。

```
▼ Response Headers    view source
Age: 1
Cache-Control: max-age=315360000
Connection: keep-alive
Content-Type: text/css
Date: Tue, 31 May 2016 16:14:39 GMT
Expires: Fri, 15 May 2026 07:56:56 GMT
Last-Modified: Wed, 20 Apr 2016 07:36:58 GMT
X-Via: 1.1 dwt76:1 (Cdn Cache Server V2.0)
```

如图所示，两者同时存在，而 Cache-Control 比 Expires 优先级高，Expires 的作用被忽略。

2. Last-Modified/If-Modified-Since

Last-Modified/If-Modified-Since 是确认 web 服务器和浏览器缓存有效性的一种机制，用来确认服务器端的文件是否更新，需要与 Expires/Cache-Control 结合使用。



Last-Modified 用于标记请求文件在服务器端的最后被修改时间，浏览器再请求此 URL 时，根据 HTTP 协议的规定，浏览器会向服务

器传送 If-Modified-Since 报头，其值等于 Last-Modified。

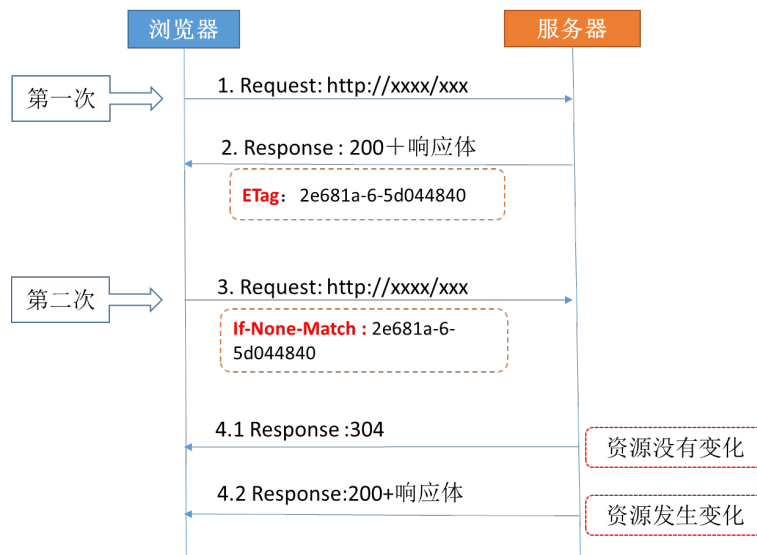
### 3. ETag/If-None-Match

ETag/If-None-Match 同 Last-Modified/If-Modified-Since 一样，用于确认服务器端的文件是否更新。提出 ETag 的原因，主要为了解决 Last-Modified 无法解决的一些问题。如：

(1) 一些文件也许会周期性的更改，但是它的内容并不改变(仅改变文件的修改时间)，这个时候并不希望客户端认为这个文件被修改了而重新请求；

(2) 某些文件修改非常频繁，比如在秒以下的时间内进行修改(1s 内修改了 N 次)，而 If-Modified-Since 能检查到的粒度是 s 级的，对这种修改无法判断；

(3) 某些服务器不能精确的得到文件的最后修改时间；



ETag 由服务器端生成，是一个和文件相关的标记，默认是对文件的索引节 (INode)，大小 (Size) 和最后修改时间 (MTime) 进行

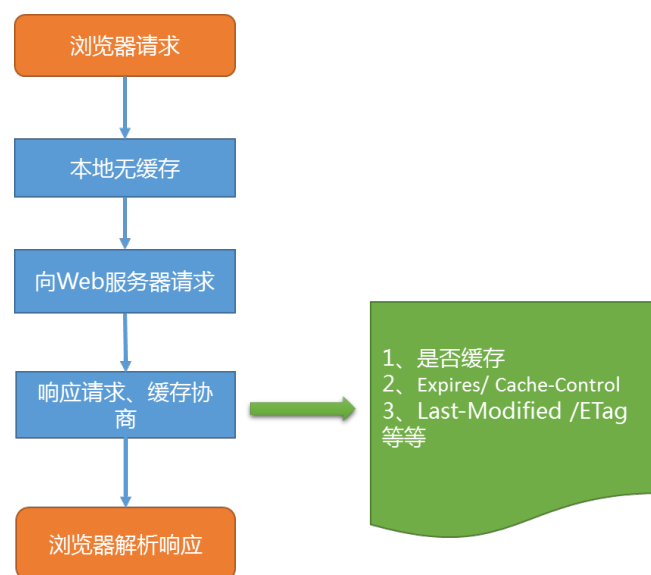
---

Hash 后得到的。

如果服务器端同时设置了 ETag 和 Last-Modified, 则只有 If-None-Match 和 If-Modified-Since 完全匹配, 服务器才返回 304。

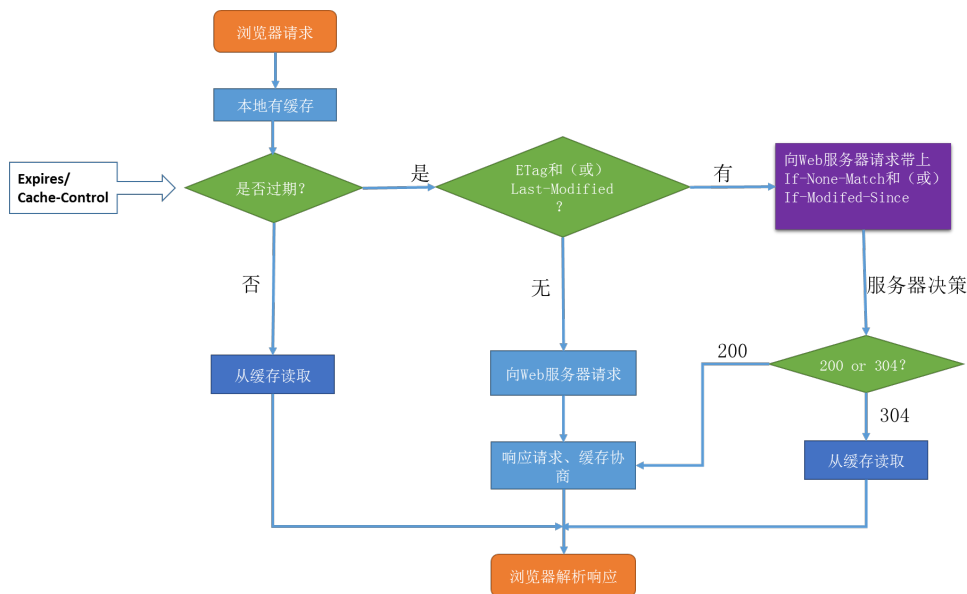
#### 4. 整体流程

(1) 浏览器第一次请求时与服务器交互过程:



如果服务器希望浏览器对该页面做缓存, 则会设置与缓存机制有关的响应头: Expires/Cache-Control 指定缓存时间; Last-Modified/ETag 指定文件相关属性。

(2) 浏览器再次请求且有缓存时与服务器交互过程:

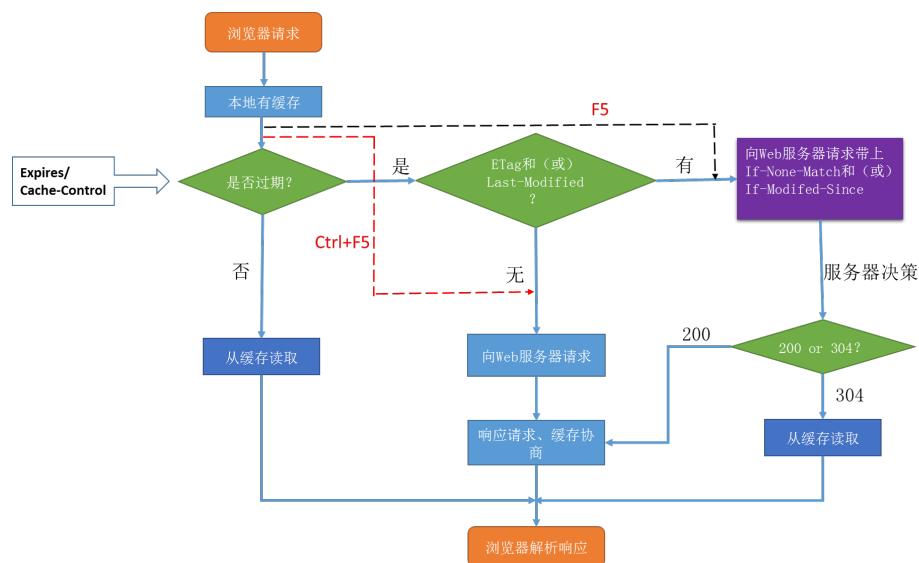


判断缓存是否过期，如果未过期，直接读取 http 缓存文件，不发送 http 请求；否则判断是否含有 Etag 和（或）Last-Modified，有则带上 If-None-Match 和（或）If-Modified-Since 发送请求，服务器判断后，如果缓存有效返回 304，无效返回 200。

## 5. 浏览器缓存行为与用户行为的关系

当浏览器要加载的资源存在于缓存且未过期的情况下，浏览器是否直接使用缓存跟用户的行为是有关系的，并不一定有缓存就立即使用。具体如下：

用户操作	Expires/Cache-Control	Last-Modified/Etag
地址栏回车	有效	有效
页面链接跳转	有效	有效
新开窗口	有效	有效
前进、后退	有效	有效
F5刷新	无效	有效
Ctrl+F5刷新	无效	无效



## 6、小结

作为 Web 前端的一种优化机制，通过使用有效的缓存可避免重复的网络请求，从而在整体上加速网页的渲染速度，提升用户体验。Chrome 浏览器可通过在地址栏输入 `chrome://cache/` 查看缓存；Firefox 浏览器下输入 `about:cache?storage=disk&context=查看缓存`。

## 7、参考

- (1) <http://www.alloyteam.com/2012/03/web-cache-2-browser-cache/>
- (2) <http://www.zhihu.com/question/22883627>
- (3) <http://www.cnblogs.com/skynet/archive/2012/11/28/2792503.html>
- (4) <http://www.dreamdu.com/xhtml/no-cache/>
- (5) <http://blog.csdn.net/kikikind/article/details/6266101>
- (6) <http://blog.csdn.net/21aspnet/article/details/6602933>
- (7) <http://www.w3ctech.com/topic/866>
- (8) <http://www.techweb.com.cn/network/system/2016-01-05/2252395.shtml>