

# http 缓存

---

缓存是什么？

通过复用以前获取的资源，可以显著提高网站和应用程序的性能。Web 缓存减少了等待时间和网络流量，因此减少了显示资源表示形式所需的时间。通过使用 HTTP 缓存，变得更加响应性。

哪些资源可以缓存？

CSS, JS, 图片, 音频, 视频这些不常修改的资源可以用来做缓存。html 一般不能用来缓存，因为 html 会常修改，每次都不一样。

## 1. 强缓存

直接使用客户端缓存，不从服务器拉取新资源，返回的状态码为 200。

expires 为服务端返回的到期时间，是一个 GMT（格林尼治标准时间）绝对时间，如：Tue, 17 Jan 2023 08:00:00 GMT。下一次请求时，客户端判断当前系统 GMT 时间是否小于缓存携带的 GMT 时间。若小于，直接使用缓存数据，否则从服务器请求新的文件。比较的是浏览器时间，浏览器时间和服务器时间可能不一样，导致判断不准确。HTTP 1.1 中使用 Cache-Control 的 max-age 来替代 expires。

**HTTP 中的时间均用国际标准时间表示，从来不使用当地时间。**

Cache-Control 缓存控制，他的取值有 max-age, no-cache, no-store, private, public。

- max-age: 设置缓存的时间，单位 s。
- no-cache: 表示缓存由服务端控制，把请求提交给原始服务器进行验证（协商缓存验证）。
- no-store: 不使用任何缓存。
- public: 表明响应可以被任何对象（包括：发送请求的客户端，代理服务器，等等）缓存。
- private: 表明响应只能被单个用户缓存，不能作为共享缓存

如果服务器同时下发了 expires 和 max-age，应该以 max-age 为准，这是规定。

## 2. 协商缓存（对比缓存）

**客户端缓存失效后会向服务器进行缓存有效性验证，这个缓存有效性验证的过程就是协商缓存。若资源有效，则返回 304 (Not Modified)。客户端拿到 304 状态码后会再从本地缓存中获取资源。**

- 协商缓存是一种服务端缓存策略。
- 服务端判断资源是不是之前一样，一样则返回 304 状态码，使用之前的本地缓存资源，否则返回 200 状态码和新的资源。

## 3. 资源标识

在 response header 中有两种协商缓存相关的资源标识，Last-Modified 和 ETag。

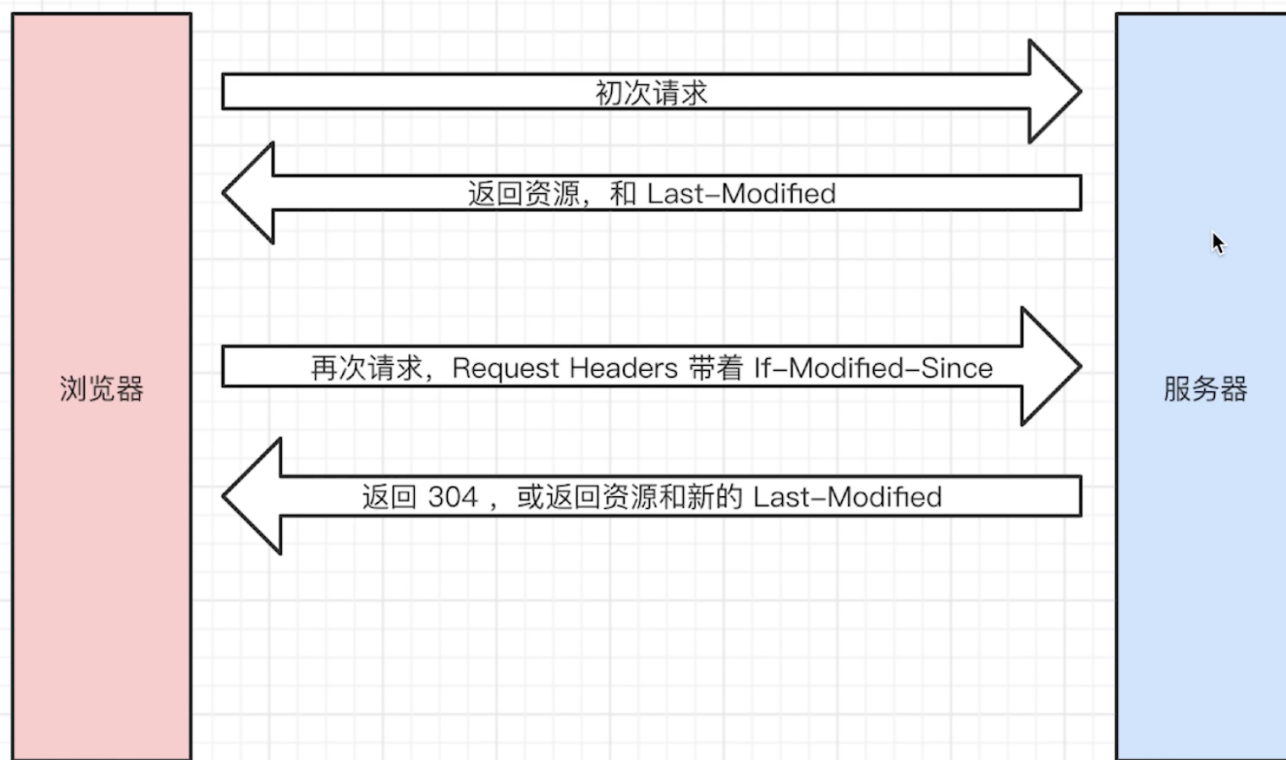
- Last-Modified 资源的最后修改时间，单位秒级。浏览器下一次请求时携带 If-Modified-since（只可以用在 GET 或 HEAD 请求中）请求头，或者 If-Unmodified-Since（被用于 POST 或其他非简单请求）请求头。

- ETag 资源的唯一标识（一个字符串，类似于人的指纹）。浏览器下一次请求时携带 If-None-Match 请求头。

服务器将客户端的 ETag（与 If-None-Match 一起发送）与其当前版本的资源的 ETag 进行比较，如果两个值匹配（即资源未更改），服务器将返回不带任何内容的 304 Not Modified 状态，告诉客户端缓存版本可用（fresh）。

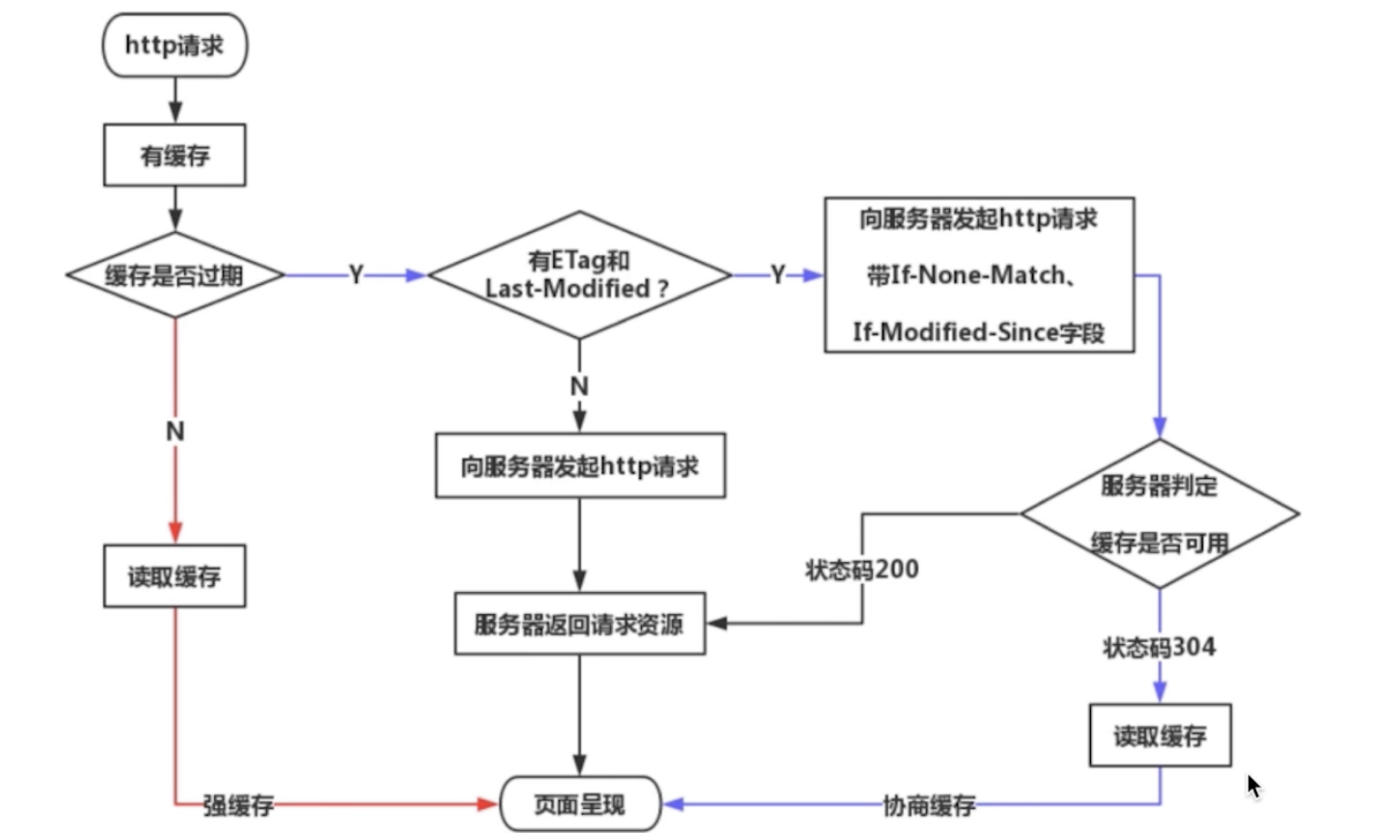
```
// 响应头:  
Last-Modified: Wed, 21 Oct 2015 07:28:00 GMT  
ETag: "33a64df551425fcc55e4d42a148795d9f25f89d4"
```

## Last-Modified



如果响应头同时存在两种协商缓存的资源标识，会优先使用 Etag。因为 Last-Modified 只能精确到秒级（秒对于计算机来说是很长的时间单位，计算机一般以毫秒计算）。如果资源被重复生成，而内容不变，则 ETag 更精确。

# http 缓存 - 综述



## 4. Expires

- 相对文件的最后访问时间(Atime)：此时和max-age的值相等
- 绝对修改时间(MTime)：此时Expires的初始值为文件创建时间；而Max-age的初始值为客户端请求数据的时间

	强缓存	协商缓存
HTTP 1.0	Expires	Last-Modified
HTTP 1.1	Cache-Control	ETag

## 5. 推送缓存

HTTP/2.0 中设计了新的缓存方式，服务器推送（Push Server）。有别于强制缓存和协商缓存，属于推送缓存。这种新的缓存方式主要是为了解决客户端缓存时效性的问题，即还没有收到客户端的请求，服务器就把各种资源推送给客户端。比如，客户端只请求了a.html，但是服务器把a.html、a.css、a.png全部发送给客户端。这样的话，只需要一次请求，客户端就更新了所有文件的缓存，提高了缓存的时效性。