流程

当资源第一次被访问的时候，HTTP头部如下   
(Request-Line) GET /a.html HTTP/1.1   
Host 127.0.0.1   
User-Agent Mozilla/5.0 (X11; U; Linux i686; zh-CN; rv:1.9.0.15) Gecko/2009102815 Ubuntu/9.04 (jaunty) Firefox/3.0.15   
Accept text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8   
Accept-Language zh-cn,zh;q=0.5   
Accept-Encoding gzip,deflate   
Accept-Charset gb2312,utf-8;q=0.7,;q=0.7   
Keep-Alive 300   
Connection keep-alive   
HTTP返回头部如下   
(Status-Line) HTTP/1.1 200 OK   
Date Thu, 26 Nov 2009 13:50:54 GMT   
Server Apache/2.2.11 (Unix) PHP/5.2.9   
Last-Modified Thu, 26 Nov 2009 13:50:19 GMT   
Etag "8fb8b-14-4794674acdcc0"   
Accept-Ranges bytes   
Content-Length 20   
Keep-Alive timeout=5, max=100   
Connection Keep-Alive   
Content-Type text/html   
  
当资源第一次被访问的时候，http返回200的状态码，并在头部携带上当前资源的一些描述信息，如   
Last-Modified // 指示最后修改的时间   
Etag // 指示资源的状态唯一标识   
Expires // 指示资源在浏览器缓存中的过期时间   
　　接着浏览器会将文件缓存到Cache目录下，并同时保存文件的上述信息   
　　当第二次请求该文件时，浏览器会先检查Cache目录下是否含有该文件，如果有，并且还没到Expires设置的时间，即文件还没有过期，那么此时浏览器将直接从Cache目录中读取文件，而不再发送请求   
  
　　如果文件此时已经过期，则浏览器会发送一次HTTP请求到WebServer，并在头部携带上当前文件的如下信息   
If-Modified-Since Thu, 26 Nov 2009 13:50:19 GMT   
If-None-Match "8fb8b-14-4794674acdcc0"   
　　即把上一次修改的时间，以及上一次请求返回的Etag值一起发送给服务器。服务器在接收到这个请求的时候，先解析Header里头的信息，然后校验该头部信息。   
　　如果该文件从上次时间到现在都没有过修改或者Etag信息没有变化，则服务端将直接返回一个304的状态，而不再返回文件资源，状态头部如下   
(Status-Line) HTTP/1.1 304 Not Modified   
Date Thu, 26 Nov 2009 14:09:07 GMT   
Server Apache/2.2.11 (Unix) PHP/5.2.9   
Connection Keep-Alive   
Keep-Alive timeout=5, max=100   
Etag "8fb8b-14-4794674acdcc0"   
  
　　这样，就能够很大程度上减少网络带宽以及提升用户的浏览器体验。   
　　当然，如果服务器经过匹配发现文件修改过了，就会将文件资源返回，并带上新文件状态信息。