

应首部作为响应首部的起点。然后缓存对这些基础首部进行了修改和扩充。

缓存负责对这些首部进行改造，以便与客户端的要求相匹配。比如，服务器返回的可能是一条 HTTP/1.0 响应（甚至是 HTTP/0.9 响应），而客户端期待的是一条 HTTP/1.1 响应，在这种情况下，缓存必须对首部进行相应的转换。缓存还会向其中插入新鲜度信息（Cache-Control、Age 以及 Expires 首部），而且通常会包含一个 via 首部来说明请求是由一个代理缓存提供的。

注意，缓存不应该调整 Date 首部。Date 首部表示的是原始服务器最初产生这个对象的日期。

7.7.6 第六步——发送

一旦响应首部准备好了，缓存就将响应回送给客户端。和所有代理服务器一样，代理缓存要管理与客户端之间的连接。高性能的缓存会尽力高效地发送数据，通常可以避免在本地缓存和网络 I/O 缓冲区之间进行文档内容的复制。

7.7.7 第七步——日志

大多数缓存都会保存日志文件以及与缓存的使用有关的一些统计数据。每个缓存事务结束之后，缓存都会更新缓存命中和未命中数目的统计数据（以及其他相关的度量值），并将条目插入一个用来显示请求类型、URL 和所发生事件的日志文件。

最常见的缓存日志格式为 Squid 日志格式和网景的可扩展通用日志格式，但很多缓存产品都允许用户创建自定义的日志文件。第 21 章探讨了日志文件格式。

174

7.7.8 缓存处理流程图

图 7-12 以简化形式显示了缓存是如何处理请求以获取一个方法为 GET 的 URL 的。¹²

7.8 保持副本的新鲜

可能不是所有的已缓存副本都与服务器上的文档一致。毕竟，这些文档会随着时间发生变化。报告可能每个月都会变化。在线报纸每天都会发生变化。财经数据可能每过几秒钟就会发生变化。如果缓存提供的总是老的数据，就会变得毫无用处。已缓存数据要与服务器数据保持一致。

HTTP 有一些简单的机制可以在不要求服务器记住有哪些缓存拥有其文档副本的情

注 12：可以通过条件请求在一步里完成图 7-12 列出的资源再验证及获取（参见 7.8.4 节）。