

## 7.12 缓存和广告

读到这里，你一定已经意识到缓存可以提高性能并减少流量。知道缓存可以帮助用户，并为用户提供更好的使用体验，而且缓存也可以帮助网络运营商减少流量。

### 7.12.1 发布广告者的两难处境

你可能认为内容提供商会喜欢缓存。毕竟，如果到处都是缓存的话，内容提供商就不需要购买大型的多处理器 Web 服务器来满足用户需求了——他们不需要付过高的网络服务费，一遍一遍地向用户发送同样的数据。更好的一点是，缓存可以将那些漂亮的文章和广告以更快，甚至更好看的方式显示在用户的显示器上，鼓励他们去浏览更多的内容，看更多的广告。这就是内容提供商所希望的！吸引更多的眼球和更多的广告！

但这就是困难所在。很多内容提供商的收益都是通过广告实现的——具体来说，每向用户显示一次广告内容，内容提供商就会得到相应的收益。（可能还不到一两便士，但如果一天显示数百万条广告的话，这些钱就会叠加起来！）这就是缓存的问题——它们会向原始服务器隐藏实际的访问次数。如果缓存工作得很好，原始服务器可能根本收不到任何 HTTP 访问，因为这些访问都被因特网缓存吸收了。但如果你的收益是基于访问次数的话，你就高兴不起来了。

### 7.12.2 发布者的响应

现在，广告商会使用各种类型的“缓存清除”技术来确保缓存不会窃取他们的命中流量。他们会在内容上加上 no-cache 首部。他们会通过 CGI 网关提供广告。还会在每次访问时重写广告 URL。

这些缓存清除技术并不仅用于代理缓存。实际上，现在主要将其用于每个 Web 浏览器中都启用了的缓存。但是，如果某些内容提供商维护其命中率的行为太过火了，就会降低缓存为其站点带来的积极作用。

理想情况下，内容提供商会让缓存吸收其流量，而缓存会告诉内容提供商它们拦截了多少次命中。现在，缓存有好几种方式可以做到这一点。

一种解决方案就是配置缓存，每次访问时都与原始服务器进行再验证。这样，每次访问时都会将命中推向原始服务器，但通常不会传送任何主体数据。当然，这样会降低事务处理的速度。<sup>23</sup>

注 23：有些缓存支持这种再验证的变体形式，在这种方式中，它们可以在后台发起条件 GET 或 HEAD 请求。用户不会感觉到时延，但这个请求会触发对原始服务器的离线访问。这是一种改进方式，但这种方式加重了缓存的负荷，极大地增加了流经网络的流量。