

网站缓存的实现方式主要有以下几种：

(1) 动态页面的静态化。应用程序把动态生成的页面缓存到文件服务器，以后用户请求该动态页面，直接从文件服务器加载对应的静态缓存的 html 文件返回给用户，这里面主要节省了应用程序的执行时间和数据库访问时间，但是会增加缓存框架的加载和缓存查找的时间。

(2) 把解释执行的开发语言编译成为目标代码。将解释执行的高级语言（如 Java，Php），直接编译成为平台相关的目标代码，在 Java 里面，比较著名的就是即时编译器 (JIT)。该方法主要节省了解释执行代码的时间，但会增加即时编译的时间。

(3) 利用反向代理服务器的缓存。利用类似 nginx 的反向代理服务器，对请求的 url 对应的输出进行缓存。一般情况下，当用户向 web 服务器发出请求时，首先是经过反向代理服务器，由反向代理服务器根据当前服务器的资源使用情况，将不同用户的请求转发给不同的服务器，当服务器处理完毕，把结果通过反向代理服务器返回给用户。如果把用户最经常访问的数据，缓存在反向代理服务器上，那么当用户发出请求后，直接就可以从反向代理服务器的缓存中获取到数据，这样就加快了网站的响应速度。

(4) 客户端浏览器缓存。当客户端发出的 HTTP 请求中包含 If-Modified-Since，如果服务器端代码没有修改，则返回 302 响应代码的请求响应头（内容不返回）；当客户端发现如果上次请求的页还未过期，通过 Expires 或者 Cache-Control 进行辨别，直接显示本机缓存的内容，不与服务器进行通信，节省了服务器执行代码时间以及数据传输时间。