

- 定义一个独立的散列函数,这个函数会根据所请求 Web 对象的 URL 返回一个数字。
- 将 URL 散列函数的结果代入代理服务器的散列函数,得到一个数字阵列。这些数字中的最大值决定了要为这个 URL 使用的代理服务器。由于算出来的值是确定的,所以对同一个 Web 对象的后继请求会被转发给同一台代理服务器。

以上 4 项任务可以由浏览器、插件执行,也可以在一个中间服务器上计算。

为每个代理服务器集群创建一个表,表中列出了集群中的所有服务器。表中的每个条目都应该包含全局参数的相关的信息,比如,负载因子、生存时间(TTL)、倒数数值和应该以何频率查询成员之类的全局参数。负载因子说明机器可以处理多少负载,这取决于那台机器的 CPU 速度和硬盘容量。可以通过 RPC 接口对此表进行远程维护。只要表中的字段被 RPC 修改了,就可以使其对下游的客户端和代理可见,或将其发布给它们。这项发布工作是在 HTTP 中进行的,这样,所有的客户端或代理服务器就都可以在不引入另一种代理间协议的基础上消化表格信息了。客户端和代理服务器只用了一个知名 URL 来获取这张表。

所使用的散列函数必须能够确保 Web 对象在参与的代理服务器间是统计分布的。应该用代理服务器的负载因子来确定分配给那台代理的 Web 对象的统计概率。

总之, CARP 协议允许将一组代理服务器看成单个的集群缓存,而不是(像 ICP 中那样的)一组相互合作但又相互独立的缓存服务器。确定的请求解析路径会在一跳内找到某个特定的 Web 对象的家。这样会降低 ICP 在一组代理服务器中查找 Web 对象时常会产生的代理间流量。CARP 还可以避免在不同的代理服务器上存储 Web 对象的多个副本的问题,这样做的优点是缓存系统集群的 Web 对象存储容量较大,缺点是任意一个代理的故障都要改写现存代理的部分缓存内容。

477

20.9 超文本缓存协议

之前我们讨论了 ICP, 这个协议允许代理缓存向兄弟缓存查询文件是否存在。但设计 ICP 时考虑的是 HTTP/0.9 协议, 因此, 向兄弟缓存查询资源是否存在时, 只允许缓存发送 URL。HTTP 版本 1.0 和 1.1 引入了很多新的请求首部, 这些首部可以和 URL 一起用来确定文件是否匹配。因此, 只在请求中发送 URL 可能无法得到精确的响应。

HTCP(超文本缓存协议)允许兄弟缓存之间通过 URL 和所有的请求及响应首部来相互查询文档是否存在, 以降低错误命中的可能。而且 HTCP 允许兄弟缓存监视或请求在对方的缓存中添加或删除所选中的文档, 并修改对方已缓存文档的缓存策略。