

文章编号:1000-582X(2003)05-0121-04

基于内容的 Web 集群系统负载均衡策略*

李双庆, 游 莲, 程代杰

(重庆大学 计算机科学与工程学院, 重庆 400044)

摘 要: Web 集群技术解决了 Web 服务器系统容量问题, 其核心思想是负载均衡策略和算法。传统的策略对某些站点类型有效, 而不能满足各种类型站点的要求。笔者分析了各种类型站点的负载特点, 讨论了负载均衡需要解决的主要问题, 提出了一种基于内容的 Web 集群负载均衡策略和相应的负载均衡算法, 并通过仿真实验验证了该策略和算法的有效性。

关键词: 负载均衡; Web 集群; 负载均衡算法

中图分类号: TP393.11

文献标识码: A

Web 站点往往面临迅速增长的访问量和系统有限处理能力之间的矛盾。如果采用更高处理性能的服务器替代原有系统无法保护投资, 并且缺乏伸缩能力。一些站点采用镜像 (mirroring) 的方式, 在多个服务器上复制相同信息, 以不同的 URL 供用户选择。但这种方式对用户不透明, 系统无法控制用户访问, 不能达到系统负载均衡分配的目的。

更有效的办法是在一个基于 LAN 的分布式体系结构下实现负载均衡, 所有来自客户端的请求被透明地分配到若干服务器上。对用户而言, 整个分布式系统仿佛是一台单一的逻辑服务器, 称之为 Web 服务器集群 (web cluster)。这样的集群系统能够提供较强的可扩展性和较好的吞吐性能。从商业角度而言, 不仅可以保护原来的投资, 而且也可以通过廉价的集群系统获得高性能计算机所能达到的处理能力。因此该领域的研究近几年来十分活跃, 并取得了一定的进展^[1-3]。但是, 目前这些负载分配策略往往不能适应不同的 Web 负载情况, 因而缺乏普适性。笔者将讨论一种基于内容识别的负载均衡策略。该策略能够较好地解决不同负载状况下的负载均衡问题。

1 基于分配器的请求分配机制

Web 服务器集群的负载均衡策略有两大类: 一类是基于 DNS 的分配, 如 RR - DNS 和 WRR - DNS, 它们在标准 DNS 的基础上扩充动态映射的功能^[4-5], 其负载均衡能力有限, 且对客户端不透明, 在此不对其进

行讨论; 另一类是基于 IP/TCP/HTTP 重定向的分配^[1], 该类策略一般需要一个特殊的前端节点, 也称分配器 (dispatcher), 所有的客户端请求都经过分配器并由它分配到后端服务器处理。笔者将讨论该类负载均衡策略。

基于分配器的请求分配机制一般要求对客户端透明, 目前主要有两种机制^[2]。一种称为中继机制 (relaying), 如图 1(a) 所示, 客户端请求到达分配器后, 由分配器按一定的负载分配算法, 将请求传递给后端被选中的服务器。服务器处理后的结果传回至分配器, 再由分配器转发给客户端, 分配器的工作通常在操作系统的用户层完成。也有修改操作系统核心直接支持中继机制的系统, 其性能会好一些, 这种优化方法称为 TCP 衔接 (TCP splicing); 另外一种机制称为 TCP 传递 (TCP handoff), 如图 1(b) 所示, 客户端的请求经过分配器分配到达后端服务器, 服务器处理后, 将结果不经过分配器而直接发送给客户端。

中继或 TCP 衔接机制要求所有的通信均要经过分配器, 特别是处理结果信息量往往大于请求信息量, 因此容易形成通信瓶颈; TCP 传递机制只有请求信息经过分配器, 因此性能更好。但需要修改前端和后端节点的网络部件, 以支持 TCP handoff 协议。

2 负载分配策略

在进行负载分配时需要注意两点。一是内存

* 收稿日期: 2003-01-21

基金项目: 重庆市科技攻关项目 (2001-6715)

作者简介: 李双庆 (1969-), 男, 重庆人, 重庆大学博士研究生, 主要研究方向: 电子商务, 多代理技术, 分布式处理。

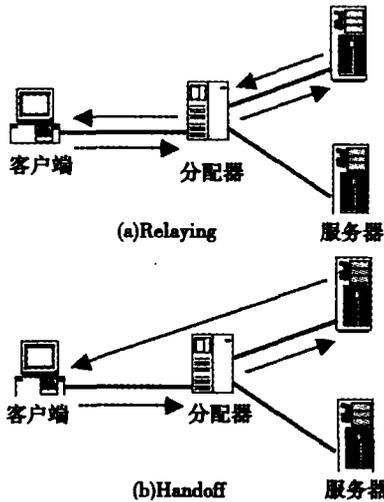


图1 基本分配器的请求分配机制

cache 的访问速度明显高于磁盘文件访问的速度,故提高 cache 的命中率对集群系统的响应性能有很大的帮助^[6];二是随着动态页面的出现,不同页面所消耗的系统资源差异可能很大^[7]。笔者将 Web 服务分成 4 种类型:

1) 发布型:主要提供静态信息(如 HTML 页面及嵌入的对象),该类服务的页面一般可以在内存 cache 中。

2) 事务型:结果主要来自动态数据库查询,且查询条件一般由用户通过一个 HTML 页面动态提供。由于它需要进行密集的磁盘访问,因此也称为磁盘密集型服务。该类服务不允许内存 cache。

3) 电子商务型:提供静态、动态和安全信息传输,面向电子商务应用。在进行安全信息传输时,加解密操作需要消耗大量的 CPU 资源,数据库访问时需要密集访问磁盘,因此也称为磁盘/CPU 密集型服务。该类服务不允许内存 cache。

4) 多媒体型:提供实时的音频和视频服务。此类服务在 Web 站点中一般通过特殊的服务器和网络协议连接,这里不作讨论。

根据上述服务的特征,提出一种负载分配策略。其核心思想是:通过分配器识别客户端请求的内容,确定请求的服务类型。如果是发布型请求,也称静态类型,则应当保证请求的局部性(locality),即尽可能将相同请求发送给同一台服务器,以获得较高的 cache 命中率。如果请求的服务类型是事务型或电子商务型,也称动态类型,则将同类请求均匀地分配给后端服务器。这样,每个服务器得到的各类请求量大致相同,负载保持均衡。图 2 给出了一个简单的负载分配策略示意图。

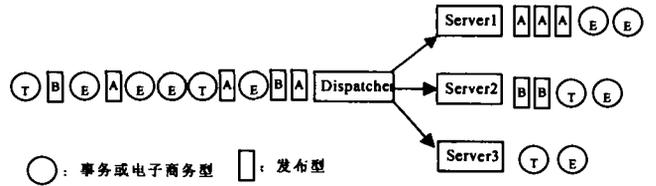


图2 负载分配策略示意图

3 负载均衡算法 CLARD

根据上述负载分配策略,提出了一种基于内容的负载均衡算法 CLARD(Combined Locality Aware Request Distribution)。算法描述如下:

```

fetch next request r;
t = Request_type( r );
if ( t = PUB ) { // 发布型服务
  if serverSet( h ) = null then // 首次分配
    n, serverSet( h ) = { server with least Q };
  else {
    n = { server with least Q in serverSet( h ) };
    m = { server with highest Q in serverSet( h ) };
    if ( n.load > Thigh && ∃ si ∈ S, Qi < Tlow ) || n.load ≥ 2Thigh then { // 负载再分配
      p = { server with least Q };
      add p to serverSet( h );
      n = p;
    }
    if |serverSet( h )| > 1 && time() - serverSet( h ).lastMod > K then
      remove n from serverSet( h );
  }
  send r to n;
  Modify Q of n;
  if serverSet( h ) changed in this iteration then
    serverSet( h ).lastModi = time();
}
else { // 事务或电子商务型服务
  s1 = { server with lowest load of r };
  if ( |s1| > 1 ) {
    s2 = { server in s1 with least dynamic load };
    if ( |s2| > 1 )

```

```

        n = { server in s2 with least con-
        nection };
    else
        n = s2;
}
else
    n = s1;
send r to n;
ModifyLoad(n, r);
}

```

算法说明:

CLARD 对发布型服务采用了度量服务器负载状态的两个门限参数 $Thigh$ 和 $Tlow$ 。如果服务器活动连接数高于 $Thigh$ 表示负载较重,需要负载再分配。如果服务器活动连接数低于 $Tlow$ 表示负载过轻。如果分配器不限制集群中的连接总数,当所有服务器的负载量均超过 $2Thigh$,系统相当于采用 WRR 算法,响应性能将明显下降。集群的最大负载量为 $S = (n - 1) * Thigh + Tlow - 1$,其中 n 为服务器数量。这保证了当没有服务器负载量 $\leq Tlow$ 时,最多有 $n - 2$ 台服务器负载 $\geq Thigh$ 。当负载再分配时,目标服务器的负载差至少为 $Thigh - Tlow$,最大为 $2Thigh - Tlow$ 。

显然设置 $Thigh$ 和 $Tlow$ 的值对算法十分重要。 $Tlow$ 根据服务器的处理能力确定,处理能力越强,取值越大,反之则越小。 $Thigh - Tlow$ 的取值与响应的局部性和延迟差相关。取值过大,则局部性好而延迟差大,取值过小,则局部性差而延迟差小。因此, $Thigh - Tlow$ 的取值不能过高或过低。在给定 $Tlow$ 后,如果期望的最大延迟差为 D ,平均请求服务时间为 R ,则 $Thigh$ 的取值为 $(Tlow + D/R)/2$,且满足 $Thigh > Tlow$ 。在仿真实验中,典型的取值是 $Thigh = 25$ 和 $Tlow = 5$ 。

此外,算法还考虑了同一种发布型 URL 请求在一台服务器处理不能满足响应要求的情况,将一类请求映射给一个服务器集(server set)。服务器集中的服务器数量有 1 到 n 台,并且根据负载情况动态变化。当某请求所对应的服务器集中所有服务器都过载时,分配器给该服务器集增加一台服务器;如果经过时间 K 服务器集中没有发生服务器的增减,则减少其中一台服务器。为防止系统出现抖动现象, K 值设置不应过小。在仿真实验中, $K = 15$ s。服务器集是一种逻辑分配,同一台服务器同时被分配到一个或多个服务器集中。

对事务型或电子商务型服务,CLARD 算法尽可能均匀地将同种请求分配给后端服务器,同种请求在后端服务器上的数量相差最大为 1;如果某一请求在多台服务器上均有最小负载,则算法选择其中具有最少动态负载的服务器;如果满足条件的服务器数量仍然

大于 1,则选择其中连接数最少的服务器;假设其中连接数最少的服务器数量也大于 1,则任意选择其中一台服务器。

当一个请求服务完成后,服务器的活动连接数和此类服务的数量将会相应减 1。

4 仿真实验

为了研究不同负载均衡策略在不同集群规模、不同 CPU 处理速度和内存容量情况下的系统特性,设计开发了 Web 服务器集群仿真器。仿真器对网络本身的传输带宽和特性不作限制,且以 450 MHz Pentium II 微机运行 FreeBSD 2. 2. 5 获得的测试数据^[8]为依据,建立服务器集群处理和响应模型。当文件大小为 17 kB 时,全部命中内存 cache 的吞吐率为 1 000 req/sec。cache 替换算法采用 LRU 算法,cache 大小为 16 MB。对动态页面文件,不允许内存 cache,并假设磁盘文件的访问速度为内存访问速度的 10%。对磁盘密集访问,例如数据库查询操作,其速度根据结果数据大小确定,设定为同等大小内存文件访问速度的 1%。对系统加密处理的开销,不同的加密算法开销差异较大,假设采用 RSA 和 Triple-DES 算法,内存文件访问速度下降为原来的 1%。

同时可以建立 3 种典型 Web 站点的负载模型:1) 发布型站点,所有页面请求均为静态请求;2) 事务型站点,包含 75% 的静态请求和 25% 的动态请求;3) 电子商务型站点,包含 60% 的静态请求、20% 磁盘密集的动态请求、15% CPU 密集型请求以及 5% CPU/磁盘密集型请求。此外,整个站点的规模为 20 000 个 URL。整个负载模型中 90% 的请求会访问集中在整个站点 10% 的 URL 上。请求结果页面的大小满足正态分布,且平均大小为 17.3 kB。在以上模型的基础上,进行仿真实验,可获得以下结果,如图 3~图 5 所示:

从仿真结果可以看出 CLARD 在发布型、事务型和电子商务型站点等不同环境下,与 LARD 和 CAP 算法相比^[6-7],均具有较好的响应性能,避免了其它算法只能针对某类站点服务的局限性。其中对发布型站点,CLARD 算法与 LARD 算法的性能相似,明显优于 CAP 算法。对事务型和电子商务型站点,CLARD 算法与 CAP 算法相似,性能优于 LARD 算法。

5 总结

对不同类型的站点,负载均衡策略和算法需要解决不同的问题。对发布型静态页面,需要通过保证负载分配的局部性,提高 cache 命中率,来获得较好的响应性能。对事务处理和电子商务型页面由于所消耗的系统资源较大,而且不能进行内存 cache,因此需要考

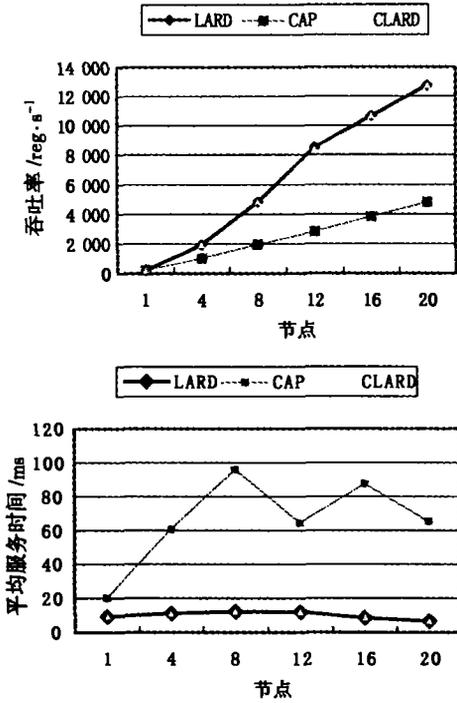


图3 发布型站点仿真结果

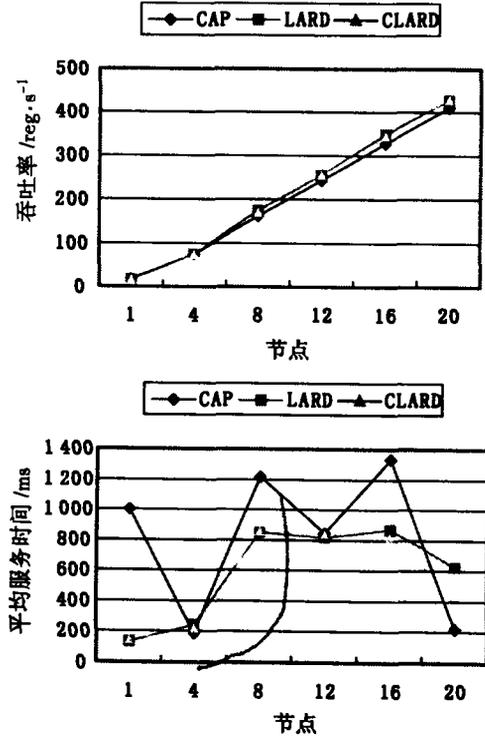


图5 电子商务型站点仿真结果

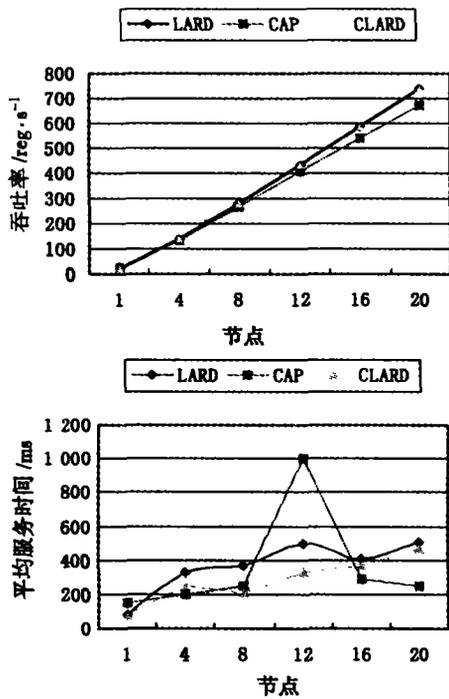


图4 事务型站点仿真结果

虑在不同服务器上对同一种请求的均匀分配。结合上述思想,提出了一种基于内容的复合型负载分配策略和相应的负载分配算法 CLARD。通过仿真实验验证了该策略和算法较好解决了静态负载的局部性和动态

负载的均匀分配问题,与其它传统算法相比具有较好的性能。

参考文献:

- [1] TREVOR S, STEVE G, BYRAV R. Scalable Web Server Clustering Technologies[J]. IEEE Network, 2000,14(3): 38~45.
- [2] VALERIA C, MICHELE C, PHILIP S Y. Dynamic Load Balancing on Web - Server Systems [J]. IEEE Internet Computing, 1999,3(3):28-39.
- [3] MOHIT A, DARREN S, PETER D, et al. Scalable Content-aware Request Distribution in Cluster-based Network Servers[A]. Proc. Of the 2000 Annual Usenix Technical Conference[C]. San Diego, CA,2000.
- [4] RFC2391. SRISURESH, GAN. Load Sharing using IP Network Address Translation (LSNAT) [S], August 1998.
- [5] RFC1794, BRISCO, T. DNS Support for Load Balancing [S], April 1995.
- [6] VIVEK S, MOHIT A. Locality-Aware Request Distribution in Cluster-based Network Servers[A]. In Proceedings of ASPLOS-VIII[C]. ACM SIG-PLAN, 1998: 205-216.
- [7] EMILIANO C, MICHELE C. A Client-Aware Dispatching Algorithm for Web Clusters Providing Multiple Services [A]. Proc. of 10th Int'l World Wide Web Conference [C]. Hong Kong, 2001.
- [8] The Workload for the SPECweb96 Benchmark [EB/OL]. <http://www.specbench.org/osg/web96/workload.html>.

frequency distribution [J]. Proc IEEE ICASSP - 94, 1994,4:301 - 304.

fractional Fourier transform [J]. IEEE Trans Signal Processing, 1996,44:2 141 - 2 150.

[12] OZAKTAS H M, ARIKAN O. Digital computation of the

A Numerical Algorithms of Fractional Fourier Transform

CHEN Ming-jie

(College of Computer and Information Engineering, Chongqing Technology and Business University, Chongqing 400067, China)

Abstract: Signal and its Fourier transform can reflect the info of time-domain and frequency-domain independently. As a usually math tool, Fourier transform is using in math, physical and engineering field widely. A new tool for signal analysis, which is named Fractional Fourier transform (FRFT), is introduced. It is interpreted in view of the classic Fourier transform. It is important to study the numerical algorithms of FRFT, because usually the FRFT can not be given to a analytical expression. A numerical algorithms of FRFT is given. It is possible to separate the distortion from the simple Gaussian signal on which a chirp distortion has been superimposed by using fractional transform.

Key words: Fractional Fourier transform; Fourier transform; filter; disturb

(责任编辑 李胜春)

~~~~~  
(上接第 124 页)

## Content-aware Load Balancing Strategy in Web Cluster System

*LI Shuang-qing, YOU Lian, CHENG Dai-jie*

(College of Computer, Chongqing University, Chongqing 400044, China)

**Abstract:** The capacity problem of web server can be solved by web cluster technology. The key point is load balancing strategy and algorithm. Traditional strategies are effective to some kinds of web sites, but not suitable for all. The paper analyses the load characters of different kind of web sites. A content-aware web cluster load balancing strategy and algorithm is proposed. The efficiency will be evaluated through emulation experiments.

**Key words:** load balancing; web cluster; load balancing algorithm

(责任编辑 吕赛英)