文章编号: 1008-1542(2010)03-0252-04

基于集群的 Web 服务负载均衡算法研究

张立岩, 杨国霞, 郑 琨

(河北科技大学信息科学与工程学院, 河北石家庄 050018)

摘 要: 使用服务器集群系统解决网络服务节点的性能瓶颈问题是目前普遍采用的方法, 其中负载均衡技术又是决定整个集群的性能和可伸缩性的关键因素。在此提出了一种结合动态任务分配表和基于多 IP 虚拟路由器集群系统的负载均衡技术。

关键词: 负载均衡; 任务分配; 虚拟路由器

中图分类号: TP393 文献标志码: A

Research of load balancing technology for Web server cluster systems

ZHANG Li-yan, YANG Guo-xia, ZHENG Kun

(College of Information Science and Engineerning, Hebei University of Science and Technology, Shijiazhuang Hebei 050018, China)

Abstract: A server cluster system is always adopted to solve the performance bottleneck problem of the network service node, and the load balancing technology is the key factor to decide the whole cluster's efficiency and scalability. In this paper, the authors present a new kind of load balancing technology which is based on mobile task allocation tables and virtual router clusters.

Key words: load balancing; task allocation; virtual router

Web 应用服务器集群系统是由多台服务器组成。其将客户的大量、突发的请求以及服务器端处理任务均衡地分配给集群中的服务器, 从而实现更高的并发处理能力和可伸缩性, 达到极高的可靠性和更好的性能价格比。

1 应用系统对集群要求

高可靠性可以作为系统的一种冗余设定。对于一个特定的请求, 如果所申请的服务器不能进行处理, 对于一个高效的系统, 如果该 Web 服务器失败, 其他的服务器可以马上取代它的位置, 对所申请的请求进行处理, 而这一过程对用户来说, 要尽可能透明, 使用户察觉不到。

稳定性决定了应用程序能否支持不断增长的用户请求数量, 它是应用程序自身的一种能力。稳定性是影响系统性能的众多因素的一种有效的测量手段, 包括集群系统所能支持的同时访问系统的最大用户数目以及处理一个请求所需要的时间[1-3]。

2 算法的提出

一般而言, Web 服务对计算结果不作苛刻的时间限制, 建立一个能满足系统及时要求的负载平衡机制

收稿日期: 2009-12-03; 修回日期: 2010-03-10; 责任编辑: 李 穆
基金项目: 河北省教育厅自然科学基金资助项目(Z2008431)
作者简介: 张立岩(1970-), 女, 河北藁城人, 讲师, 硕士, 主要从事分布式应用开发方面的研究。

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

就能满足一般的用户需求。针对 Web 应用的负载均衡主要有 2 大类: 一是专用硬件; 二是软件。硬件负载均衡通过独立 IP、支持 TCP/IP 4 层应用的路由器或网桥, 来自网络上对某个 Web 的 TCP/IP 请求, 首先与这台设备建立连接, 然后, 负载均衡设备通过某种均衡策略, 将数据包解包再重新打包改变目的 IP, 将连接请求及数据包转发给具体的服务器进行处理; 最后, 再由负载均衡设备将处理结果转发给客户端。软件负载均衡方法主要通过 1 台高性能计算机作为专用的负载均衡服务器, 通过对 TCP/IP 协议数据报文的解包、再打包、转发, 将负载均衡分配到具体的 Web 服务器上。

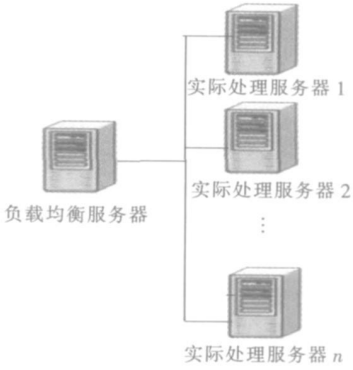

图 1　系统工作原理

Fig. 1　Principle of the system

在研究了一些动态负载平衡算法后, 提出了一种采用硬件与软件相结合的多 IP 虚拟路由器集群技术, 通过中心集中控制方式, 建立、动态维护、修改任务配置表, 实现 Web 集群系统动态负载平衡。该算法思想是: 利用应用协议分析、URL 重新定向的工作机理来进行 Web 负载均衡, 因此, URL 重新定向与基于数据包转发负载均衡相比, 具有更快的处理速度、更可靠的连接服务。系统工作原理如图 1 所示。

3　算法描述

此算法主要包括任务配置表初始化和负载平衡调度策略。

3.1　初始化任务配置表

以实际处理服务器数量划分任务, 每个实际处理服务器分配 1 个数据通道, 任务配置表为每一路通道设置标识信息, 主要内容包括计算实际处理服务器号、节点号、进程号、换算系数等。表的长度为 n 倍于任务配置表单元结构, n 为处理服务器数量, 单元结构如图 2 所示。其中换算系数是根据服务器的配置, 对于服务器配置高、处理能力强的服务器, 给予一个较小的大于 0 小于 1 的系数(小数); 对于配置低、处理能力弱的服务器给予较大的系数, 大于 1 的数。

实际处理服务器号	任务描述	节点号	进程号	换算系数	实际连接数

图 2　任务配置表单元结构

Fig. 2　Unit structure of task allocation table

任务分配原则: 对每个实际处理服务器以循环轮转的方式进行任务分配, 但分配前先检查标识, 若该节点可以进行任务分配, 则根据换算系数、实际连接数, 计算该节点的处理能力系数和当前负载情况, 若在所有可分配节点中, 该节点处理能力系数最小且连接数较小, 则将该节点任务配置表项填充, 以保证将任务分配到目前可用的处理能力最强的处理节点上, 减少服务延迟和请求的阻塞。

1) 首次产生任务配置表时, 根据在线实际处理服务器分配处理能力系数, 首先分配处理能力小的服务器。这样能使所有处理能力相同的实际处理服务器尽可能地均匀分配;

2) 当访问量增大、请求增多时, 动态增加就绪的实际处理服务器时, 增加任务配置表;

3) 当有实际处理服务器减少时, 记录所对应的实际处理服务器号, 实际处理服务器恢复时首先被分配;

4) 当有实际处理服务器失效时, 将它的任务配置表项分配给其他在线实际处理服务器。

3.2　负载均衡调度策略

负载均衡是并行处理和集群计算首先要解决的重要问题, 负载均衡算法的优劣直接影响系统的性能。负载均衡的本质是将任务合理高效地进行调度, 将各个任务比较均衡地分配到不同的处理节点进行并行处理, 同时使各有效计算节点的利用率趋于最大[4-5]。笔者采用负载当量轮转法的负载分配策略, 实现负载均衡。

通过动态任务配置表的任务描述结构, 说明任务权值、任务处理精度、处理服务器号、任务结果服务器号、任务处理进程号。任务配置表的主要数据结构见图 3。

实际处理服务器的负载当量定义为各个计算节点实际连接数与对应任务权值的加权和与实际处理服务器换算连接数乘积。任务权值表示任务处理计算量大小。

换算连接数为　$\omega = L_n \times \lambda$　L_n 为连接数, λ 为处理能力换算系数。

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

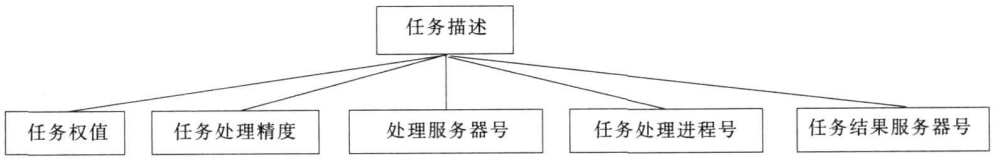

图 3 任务描述结构

Fig. 3 Data structure of task description

对任务请求进行分类。各有效实际处理服务器的负载当量向量为 $\boldsymbol{L}_W = \{l_{w_1}, l_{w_2}, l_{w_3}, \cdots, l_{w_n}\} \times \overline{\omega}$。

处理服务器最大负载当量为 $l_{w_{max}} = \max\{l_w\}$。

处理服务器最小负载当量为 $l_{w_{min}} = \min\{l_w\}$。

负载当量差为负载当量最大值与负载当量最小值之差, 即 $l_{w_{diff}} = l_{w_{max}} - l_{w_{min}}$。

平均负载当量是 $l_{w_{ave}} = \sum_{i=1}^{n} l_{w_i} \backslash n$, n 为在线实际处理服务器数量。负载当量阈值取决于当前连接数与处理服务器处理能力、系统网络延迟以及各个任务负载计算量的大小。就本系统的具体应用而言, 取负载当量阈值为 $l_{w_{th}} = l_{w_{ave}} \times \overline{\omega}$。

当各处理服务器的负载当量差别不大时, 将任务请求分配给连接个数少的在线处理服务器; 当负载当量差别较大时($l_{w_{diff}} > l_{w_{ave}}$), 将任务请求分配给在线实际处理服务器换算系数与连接数乘积最小的处理服务器处理。

负载当量轮转分配任务算法步骤如下:

1) 负载平衡服务器收集系统内所有在线处理服务器总数, 计算有效在线处理服务器数量和在线处理服务器进程号;

2) 计算系统内在线处理服务器的实时负载当量和负载当量差 $l_{w_{diff}}$ 等, 统计并更新在线处理服务器的实际连接数;

3) 根据请求计算任务, 计算任务类型权值;

4) 判断并分配任务请求, 找到并计算请求任务配置表中负载当量最小的在线处理服务器, 填充它的动态任务配置表。否则将计算请求任务顺序地分配到下一个有效在线处理服务器, 并填充相应的动态任务配置表;

5) 重新循环转到 1)。

3.3 多 IP 虚拟路由器集群技术

服务器集群系统对外表现为多个 IP 地址(VIP), 集群内每台在线处理服务器对外都表现为 1 台路由器, 它们有各自独立的 IP 地址, 并且向与之相连的负载平衡服务器通告其到达 VIP 地址的路由度量值, 该路由度量值与其当前的换算连接数成正比。不利用 TCP/IP 协议分析、解包、打包、转发, 而应用协议分析、URL 重新定向的工作机理进行 Web 负载均衡, 对于客户与 Web 服务器建立直接 TCP 连接, 发送服务请求, 具有更快的处理速度、更可靠的连接服务。负载平衡服务器自动选择路由度量值最短的路径, 根据任务配置表计算当前在线处理服务器换算连接数最小的服务器对任务进行转发, 负载平衡服务器使用如下算法计算到达 VIP 的路由度量值 R, 对于 Web 服务, R_{max} 为可能的最大会话连接数(session), 当 R 达到最大会话连接数时, 表示处理服务器已经不能再接收新的连接服务请求; 用 x 表示服务器当前剩余负载能力与总负载能力的比值。

路由度量值修正函数为 $f(x) = \sigma \times \delta^{\frac{1}{x}} - \sigma \times \delta$。

$f(x)$ 具有以下特点: $f(1) = 0$, $\lim_{x \to 0} f(x) = +\infty$,

$1 \geqslant f^1(x) > 0$, $x_1 = 1/(\log_{\delta}^{\delta + \sigma^{-1}})$, $f(x_1) = 1$ 解得: $\{0 < \sigma < 1, \delta > 1, \ln^{\delta} \times \delta < \frac{1}{\sigma}\}$。路由度量值为

$$R = \begin{cases} \lceil R_{max} \times f(x) \rceil, x_1 \leqslant x \leqslant 1; \\ R_{max}, 0 < x < x_1。 \end{cases}$$

路由度量值修正函数的物理意义是: 当没有连接会话请求时, 负载平衡服务器通告的路由度量值最小(为 1), 路由度量值修正函数曲线的斜率最大(大于 − 1, 小于 0); 路由度量值修正函数曲线的斜率随会话连接数量的增加而逐渐减小, 当 x 接近 0 时, 函数曲线斜率急剧减小, 最终接近负无穷大; 度量值在 x_1 处达

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

到 R_{max}, x_1 表示服务器的设定的负载能力, 即当负载平衡服务器的当前负载能力小于 x_1 时, 就不再接收任何会话请求[6-9]。系统中 x_1 的取值是根据系统实际情况设定。算法的物理意义符合 Web 系统服务中任何实例访问中活动会话的数目、应答的次数、高峰负载次数以及在高峰期和低谷期的会话的数目等的变化情况, 因而可以很好地反映服务器的实时工作状况, 进而提高系统的利用率和稳定性。

3.4　δ 和 σ 的定量性分析

路由度量值修正函数的走势由参数 δ和 σ控制, δ和 σ的取值应符合上面推导的结论并结合 Web 服务具体的环境要求。有 $\sigma = (\delta^{-1} - \delta)^{-1}$, 这样问题就转化为 x_1 与 δ的选取问题。x_1 是与具体环境相关的量, 在系统中可以作为常量处理, 令 $\alpha = x_1^{-1} - 1$, 则有　$\ln^{\delta} + 1 < \delta$。

3.5　算法测试与评估

对当量轮转负载分配算法的实时性能分 3 种情形作了大量测试, 结果如表 1 所示。

表 1　负载分配算法测试结果

Tab. 1　Test result of load balancing algorithm

测试情形	负载平衡延时最小值(均值)/μs	负载平衡延时最大值(均值)/μs
任务配置表初始化	423	676
处理服务器异常	423	558
超过设定连接数	723	877

运行平台为 Fedora8, 网络为千兆以太网, 取 $x_1 = 0.15$, $\delta = e$, 则 $\sigma \cong 0.007$。通过 RIPA 模块, 对应用协议进行分析, 利用模块 LMD 对处理服务器状态信息进行统计分析, 模块 OSPFD 负责与负载平衡服务器进行通信; 模块 LOAAD 负责负载的分配以及系统的任务配置表的实时更新与维护。

系统平均延时包括算法本身的执行时间, 包括处理能力换算系数计算、任务配置表的更新、网络执行的延迟等。对于硬件配置较高的系统, 网络延迟占的比例较大。延迟最小值取决于网络最小延迟, 延迟最大值在网络性能确定时取决于负载平衡服务器的处理能力。测试结果显示, 当量轮转负载分配算法的最大延迟小于 1 ms, 故本算法能达到负载均衡和灾难恢复的效果, 具有较高的实时性。

4　结　语

采用多路虚拟路由器集群技术与动态任务分配策略, 利用当量轮转负载分配算法, 利用易购设备构建扩展性好、性价比高的集群系统, 可以解决单一的 Web 服务器负载过大、访问速度过慢的问题, 提高用户访问 Web 站点性能, 提供更高的有效性和稳定性服务。尤其是并行集群系统开发具有周期短、投资少、编程方便等优点, 为学校, 机关等场所提供可靠服务。

参考文献:

[1]　BUYYA R. 高性能集群计算: 结构与系统(第一卷)[M]. 郑伟民, 汪东升, 石　威, 等译. 北京: 电子工业出版社, 2001.

[2]　晏容杰, 张玉明. 多处理机系统的自适应动态负载平衡算法研究[J]. 计算机应用 (Computer Applications), 2001, 21(7): 34-36.

[3]　胡成军, 王　戟, 陈火旺, 等. 区间逻辑的一个辅助证明工具[J]. 软件学报(Journal of Software), 2000, 11(1): 116-121.

[4]　章文嵩. 可伸缩网络服务的研究与实现[D]. 北京: 中国人民解放军国防科学技术大学, 2000.

[5]　丁　原, 刘玉树, 朱天焕, 等. 利用 IP 转发技术在集群服务器中实现负载均衡[J]. 北京理工大学学报 (Journal of Beijing Institute of Technology), 2001, 21(3): 326-329.

[6]　席　静, 褚兴军, 徐志伟, 等. 一种提高 Web 服务器性能的方法[J]. 计算机研究与发展(Journal of Computer Research and Development), 2002, 39(4): 415-422.

[7]　骆宗阳, 董玮文, 杨宇航, 等. 一种具有高可用性的通用负载均衡技术[J]. 计算机工程(Computer Engineering), 2003, 29(2): 134-136.

[8]　张永弟, 岳彦芳, 陈曹维, 等. 基于 Web 的 CAPP 与 MES 信息集成化管理系统[J]. 河北科技大学学报(Journal of Hebei University of Science and Technology), 2008, 29(2): 152-157.

[9]　池　越, 刘剑飞, 陈国鹰, 等. 基于 QPSO 的自适应均衡算法[J]. 河北科技大学学报(Journal of Hebei University of Science and Technology), 2009, 30(2): 116-119.

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net