

# 校园网络的负载均衡算法研究

舒万能<sup>1</sup>, 郑世珏<sup>1</sup>, 陈广东<sup>1</sup>, 曹晓静<sup>2</sup>

(1. 华中师范大学 计算机科学系, 湖北 武汉 430079;

2. 暨南大学 信息科学技术学院, 广东 广州 510632)

**摘 要:** ChinaGrid 支撑平台的开发成功, 将消除信息孤岛, 实现中国高校的计算资源和信息资源的有效共享。一个亟待解决的问题是在 ChinaGrid 环境下, 基于校园网络的服务器集群响应能力低下。目前已提出多种技术与方案以解决并提高校园网络的服务器集群的响应能力, 负载均衡技术就是一种全新的技术。文中根据校园网络的特点和影响负载均衡的因素, 对基于校园网络的负载均衡技术进行了分析和探讨, 并提出一种网格自调度负载均衡算法。

**关键词:** 校园网络; 开放网格服务结构; 负载均衡

中图分类号: TP301.6

文献标识码: A

文章编号: 1005-3751(2006)01-0126-03

## Research on Load Balanced Algorithm for Campus Grid

SHU Wan-neng<sup>1</sup>, ZHENG Shi-jue<sup>1</sup>, CHEN Guang-dong<sup>1</sup>, CAO Xiao-jing<sup>2</sup>

(1. Department of Computer Science, Huazhong Normal University, Wuhan 430079, China;

2. College of Information Science & Technology, Jinan University, Guangzhou 510632, China)

**Abstract:** With the successful development of ChinaGrid platform, the shared tremendous calculation and information resources in higher learning will come true in China recently, thus, the information isolated island on campuses will be disappeared. The urgent problem is how to solve the low performance responding ability of server cluster in campus grid within ChinaGrid. At present, several kinds of schemes of technology have been put forward so as to improve the response performance of each node, while load balanced technology is a new one. This paper tries to analyze the load balanced technology in ChinaGrid, based on the features in ChinaGrid and the disturbed factors of load balanced, and a self-adjusted load balanced algorithms in ChinaGrid has been put forward too.

**Key words:** campus grid; OGSA; load balance

## 0 引言

网络被誉为继传统 Internet, Web 技术之后的第三次互联网革命, 它是 Internet 发展的高级形式, 其目标是将跨地域的多台高性能计算机、大型数据库、通信设备和各种传感器重整合成一台巨大的超级计算机系统, 从而可以不必考虑地点、硬件类型和操作系统之间的差异, 使总体性能真正超越各部分性能之和, 来实现计算资源、存储资源、数据资源、信息资源、知识资源、专家资源、设备资源全面共享和协同工作<sup>[1]</sup>。

开放网格服务结构(OGSA, Open Grid Services Architecture)<sup>[2]</sup>把 Globus 标准与面向商业应用的万维网服务结合起来, 把网络计算机从科学与工程计算应用扩展到更广泛的以分布式系统服务集成为主要特征的商业应用领域, 建立网格服务的基本概念。OGSA 将整个网格看做是网格服务的集合, 这个集合的动态性很强, 是可以扩展的, 体

现了网格的动态特性。网格的异构和动态性特点, 使得网格上的资源可以随时加入或退出网格。

ChinaGrid(中国教育科研网格)支撑平台利用中国教育科研网和高校的大量计算资源和信息资源, 实现资源的有效共享, 消除信息孤岛, 提供有效的服务器, 形成高水平、低成本的计算服务平台。到网格平台全面搭建成功时, 全国 100 所重点高校的校园网格将依托 ChinaGrid 建立可聚合与共享资源的公共服务平台, 逐步实现计算资源、存储资源、数据资源、信息资源、专家资源的全面共享。图 1 是湖北省某高校计算网格平台示意图。

随着校园网格不断地深入研究和不断发展, 网络带宽瓶颈的限制将逐渐被排除, 一个亟待解决的就是提高网格环境下各个服务器站点的响应能力。影响服务器站点响应速度的因素有很多, 其中最主要的两点就是站点的处理能力和吞吐能力。单纯考虑靠提高服务器的性能来解决问题是不切合实际的, 因为客户请求的增长速度总是远远超过服务器性价比提高的速度。目前较常见的解决方案是在一个站点中用大量的服务器来同时实现 Web, FTP, DNS, E-MAIL 和 SSL 等应用。如何让这些服务器能够平滑地提供服务? 这就是负载均衡技术出现的主要原因。

收稿日期: 2005-04-06

**作者简介:** 舒万能(1981--), 男, 湖北武汉人, 硕士研究生, 研究方向为集群、网络计算; 郑世珏, 博士, 硕士生导师, 研究方向为集群、网络计算。

网络环境下负载均衡可以避免用户请求长时间等待得不到响应和资源使用不均衡的情况发生,负载均衡也是获得全局高性能的一种途径,它避免了网格中的一部分资源负载超重,并且有可能不堪重负而崩溃,而另一部分资源却没有什么事情可做。

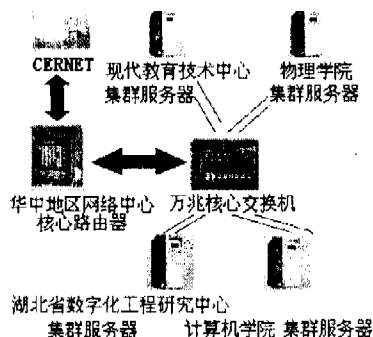


图1 湖北省某高校计算网格平台

目前学术界与工业界已经提出并实现了诸多的负载均衡机制<sup>[3]</sup>,比如轮询调度(Round Robin)、加权算法、最小连接数策略和哈希表等。在文献<sup>[4]</sup>中采用不同的原则对负载均衡方法进行了分类,文献<sup>[5]</sup>中详细论述并分析了诸类方法及其优缺点。

在对校园网络和 OGSA 分析基础上,结合影响负载均衡的因素,文中特提出一种基于校园网络的自适应负载均衡算法。

## 1 负载均衡的基本工作原理

负载均衡是由多台服务器以对称的方式组成一个服务器集合,每台服务器都具有等价的地位,都可以单独对外提供服务而无须其他服务器的辅助。通过某种负载分担技术,将外部发送来的请求均匀分配到对称结构中的某一台服务器上,而接收到请求的服务器独立地回应客户的请求。

图2显示了负载均衡的基本工作原理:给负载均衡设备设一个虚IP(VIP, Virtual IP),把它放在服务器群的前面。这个VIP就是向外界公布的地址,或者说是域名解析的地址。

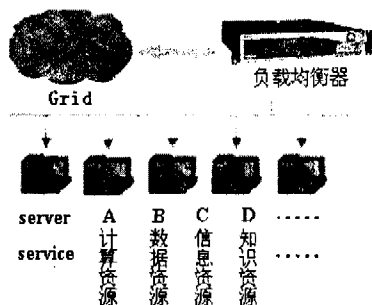


图2 负载均衡的基本工作原理

当请求到达负载均衡器时,它会重写该请求的头文件,并将之指定到服务器群中的机器上。如果某台机器被从集群中移除了,请求不会发往已经不存在的服务上,因为所有的机器表面上都具有同一个IP地址,即使集群

中的某个节点被移除了,该地址也不会发生变化。当返回一个应答时,客户端看到的只是从负载均衡器上所返回的结果。也就是说,客户端操作的对象是负载均衡器,对于其更后端的操作,对客户端来讲,是完全透明的。

一般情况下,在网格环境中信息通过高效的消息机制传给平衡器,平衡器监视所有处理节点的状态,并根据各个节点当前的资源使用状态动态调整负载均衡的任务分布,主动决定下个任务传给谁。平衡器可以是单个设备,也可以是一组平行或树状分布的设备。

## 2 自调度负载均衡算法

当客户访问集群资源时,提交的任务所需的时间和所要消耗的计算资源是千差万别的,它依赖于很多因素。例如:任务请求的服务类型、当前网络带宽的情况、以及当前服务器资源利用的情况等等。一些负载比较重的任务需要进行计算密集的查询、数据库访问、响应数据流;而负载比较轻的任务请求往往只需要读一个小文件或者进行很简单的计算。

对任务请求处理时间的不同可能会导致处理节点利用率的倾斜(Skew),即处理节点的负载不平衡。有可能存在这样情况,有些节点已经超负荷运行,而其他节点基本是闲置着。同时,有些节点已经忙不过来,有很长的请求队列,还不断地收到新的请求。反过来说,这会导致客户长时间的等待,而集群整体的服务质量下降。因此,有必要采用一种机制,使得平衡器能够实时地了解各个节点的负载状况,并能根据负载的变化做出调整。

具体的做法上采用动态反馈机制的负载均衡算法。该算法主要包括状态监控和负载分配。状态监控负责周期性地监视和收集集群内各个节点的负载信息;而负载分配根据收到的全部节点的负载信息来进行同步操作,即对要分配的负载按照权值的比例重新进行分布。

### 2.1 加权轮循调度

加权轮循调度(Weighted Round - Robin Scheduling)算法用相应的权值表示节点的处理性能。该算法根据权值的高低顺序并按照轮循的方式将任务请求分配到各节点。权值高的节点比权值低的节点处理更多的任务请求,相同权值的节点处理相同份额的请求。加权轮循的基本原理可描述为:

假设某集群内有一组节点  $N = \{N_1, N_2, \dots, N_n\}$ ,  $W(N_i)$  表示节点  $N_i$  的处理能力(权值),  $S_i(\Delta t)$  为节点  $N_i$  在  $\Delta t$  时间内的负载量。理想的系统应在任何时间  $\Delta t$  内满足:  $S_i(\Delta t)/S_j(\Delta t) = W(N_i)/W(N_j)$ , 即流量负载与节点的处理能力成正比。上式表明任务的分配是按照各个节点处理能力(权值)大小成比例分配的。

### 2.2 权值计算

在集群的节点初次投入系统中使用时,系统管理员根据节点的硬件配置情况对每个节点都设定一个初始权值  $IW(N_i)$  (通常性能越高的节点其初始权值越高),然后,

随着节点负载的变化,均衡器对节点权值不断进行动态调整。

在自适应负载均衡算法的设计中,笔者周期性地从各个节点中收集以下参数:

- CPU 利用率  $Cpu(N_i)\%$
- 内存利用率  $Memory(N_i)\%$
- 当前网络流量  $T(N_i)$
- 磁盘 I/O 访问率  $Io(N_i)\%$
- 响应时间  $Rt(N_i)$
- 进程总数  $Pr(N_i)$

将这些参数作为计算公式的因子。结合每个节点当前的权值,可以计算出新权值的大小。动态权值目的是要正确反映节点负载的状况,以预测节点将来可能的负载变化。对于不同类型的系统应用,各个参数的重要程度也有所不同。为了方便在系统运行过程中针对不同的应用对各个参数的比例进行适当调整,为每一个参数设定一个常量系数  $\pi_i$ ,用来表示各个负载参数的重要程度,其中  $\sum \pi_i = 1$ 。因此,任何一个节点  $N_i$  的权值公式就可以描述为:

$$\begin{aligned} Load(N_i) = & \pi_1 * Cpu(N_i)\% + \pi_2 * Memory(N_i)\% + \\ & \pi_3 * T(N_i) + \pi_4 * Io(N_i)\% + \\ & \pi_5 * Rt(N_i) + \pi_6 * Pr(N_i) \end{aligned}$$

另外,虽然很短的周期可以更确切地反映各个节点的负载,但是很频繁地采集(如 1 秒 1 次或者多次)会给均衡器和节点带来负担,也可能增加不必要的网络负荷。另外,由于采集器是在采集时刻进行负载计算的,经实验证明,均衡器反映出来各个节点的负载信息会出现剧烈的抖动,均衡器无法准确捕捉节点真实的负载变化趋势。因此解决这些问题,一方面要适当地调整采集负载信息的周期,一般在 5 ~ 10 秒;另一方面,可以使用移动平均线或者是滑动窗口来避免抖动,使得均衡器收集到的负载信息表现为平滑曲线,这样在动态反馈机制的调整效果上就会比较好。

均衡器的动态权值采集程序周期性地运行,若缺省权值不为零,则可查询该节点的各负载参数,并计算出动态权值  $Load(N_i)$ 。引入以下权值计算公式,结合节点的初始权值和采集的动态权值来计算最终的权值结果。

$$W(N_i) = A * IW(N_i) + B * (Load(N_i) - IW(N_i))^{1/4}$$

在公式中,如果动态权值恰好等于初始权值,最终权值不变,则说明系统的负载状况刚好达到理想状况,等于初始权值  $W(N_i)$ 。如果动态权值计算结果高于初始权值,最终权值变高,则说明系统负载很轻,均衡器将会增加分配给该节点的任务比率。如果动态权值低于初始权值,最终权值变低,说明系统开始处于重载状况,均衡器将会减少对该节点分配的任务。在实际使用中,若发现所有节点的权值都小于它们的  $W(N_i)$ ,则说明当前集群处于超载状态,这时需要加入新的节点到集群中来处理部分负载;反之,若所有节点的权值大大高于  $W(N_i)$ ,则说明当前系统的负载都比较轻。

### 3 结 论

校园网格在高校教学科研中的重要性已经初见端倪,负载均衡技术的研究在计算机领域也是方兴未艾,一种优良的负载均衡算法的设计往往与系统的体系结构、网络流量特性、流量模型有关,它一般只在某些特殊的应用环境下才能发挥最大效用。因此在应用时可根据实际系统的访问特性结合系统中各服务器站点的响应能力,把不同的算法策略和技术结合起来使用,使整个校园网格的服务器集群系统的性能得以充分发挥。

### 参考文献:

- [1] Foster I, Kesselman C, Tuecke S. The anatomy of the Grid: Enabling scalable virtual organizations[J]. International Journal Supercomputing Application, 2001, 15(3): 200 - 222.
- [2] Foster I, Kesselman C. 开放网格服务体系结构. 网格计算(第 2 版)[M]. 金 海, 等译. 北京: 电子工业出版社, 2004. 125 - 139.
- [3] Dias D M, Kish W, Mukherjee R, et al. A Scalable and Highly Available Web Servers[A]. Proc of 41st IEEE Computer Society Intl. Conf, Computer Conference, IEEE International (COMPCON)[C]. [s.l.]: [s.n.], 1997. 85 - 92.
- [4] Schroeder T, Goddard S, Ramamurthy B. Scalable Web Server Clustering Technologies[J]. IEEE Network, 2000, 14(3): 38 - 45.
- [5] Cardellini V, Colajanni M. Dynamic load balancing on Web-server systems[J]. IEEE Internet Computing, 1999, 3(3): 28 - 39.

(上接第 125 页)

- [4] 马忠梅, 李善平, 康 慨, 等. ARM 嵌入式处理器结构与应用基础[M]. 北京: 北京航空航天大学出版社, 2002.
- [5] Furber S. ARM Soc 体系结构[M]. 田 泽, 于敦山, 盛世铭译. 北京: 北京航空航天大学出版社, 2002.
- [6] 毛德操, 胡希明. 嵌入式系统—采用公开源代码和 StrongARM/Xscale 处理器[M]. 杭州: 浙江大学出版社, 2003.
- [7] UMSUNG 公司. S344BOX 手册[DB/OL]. <http://www.hyesco.com/download/ARM/s3c44box.rar>, 2004.
- [8] ARM 公司. ARM Software Development Toolkit Version 2.0 - Programming Techniques[DB/OL]. <http://infoeng.ee.ic.ac.uk/gac1/Architecture/Progtech.pdf>, 2004.