一种新主动式自调度集群系统

A New Auto Self－Allocating Cluster System

谢向文　（中南大学信息科学与工程学院 湖南长沙 410075）
费耀平　（中南大学信息科学与工程学院 湖南长沙 410075）
（中南大学网络中心 湖南长沙 410075）

摘要：针对一种与传统不同的主动式自调度集群系统(ASACS)，分析了其优点和不足。通过改进它的体系结构和调度框架，并提出了相应的实现方式，很好地解决了由于在 ASACS 中所有从客户端发向服务器的报文都要经过集中器转发而造成了集中器成为了整个集群系统的瓶颈问题，以及由于服务器单网卡的配置而造成的传输效率低的问题。

关键词：自调度 集群 网络瓶颈 瓶颈问题

1 引言

随着网络的快速发展，作为网络应用中最重要的组成部分——服务器的服务性能问题受到越来越多的重视。提高服务器的性能，要么提高服务器的硬件性能，要么以目前研究的热点——集群技术[1]来解决。由于硬件技术的提高速度跟不上人们的需求，并且升级的成本需要成倍于能得到的性能的提高。通过集群技术可以使多台低配置的服务器协调工作，大大提高了服务能力，并且不用付出太多的成本。集群技术的好坏决定了能否充分发挥其最大能量，本文从集群的体系结构和调度机制来讨论其性能问题。

2 传统的集群体系结构

传统的集群系统主要可分为无前端分配器和有前端分配器两种类型。它们的明显区别是前者对于用户表现为多个 IP 地址，而后者通常只表现为单一的 IP 地址。前者在结构上不存在瓶颈，通常没有整个系统的一致性维护工作，因而实现简单，但在可靠性和均衡效果方面存在问题；后者的特点是用户只能看到前端分配器的 IP 地址，分配器不参与实际的服务工作，只进行后端服务器集群的状态收集、系统一致性维护和负载分配的工作。

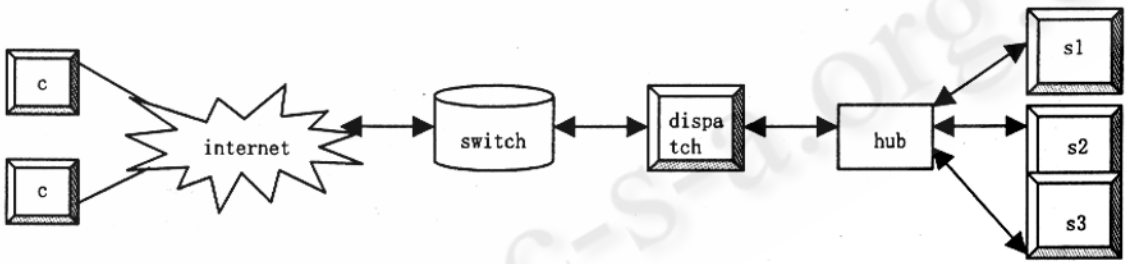


图 1　传统服务器集群体系结构

目前使用最广泛的还是上面所说的第二种类型，一般由前端的分配器和后端的服务器群组成集群系统。体系结构如图 1 所示。

目前关于传统集群系统的研究，一般集中在前端分配器上。研究最多的是分配器上的调度与分发策略，已达到提高集群性能的目的。吞吐率，任务响应时间以及各服务器间的负载均衡程度是集群性能好坏的主要指标。

传统服务器集群系统中一般都采用基于分发器的调度机制，如图 2 所示。客户端发出请求，请求到达集群的分配器，分配器通过一定的分配策略选择一台后端的服务器，然后将请求发送到被选择的后端服务器。服务器处理完请求后将处理结果发送到前端分配器由其转发给客户。

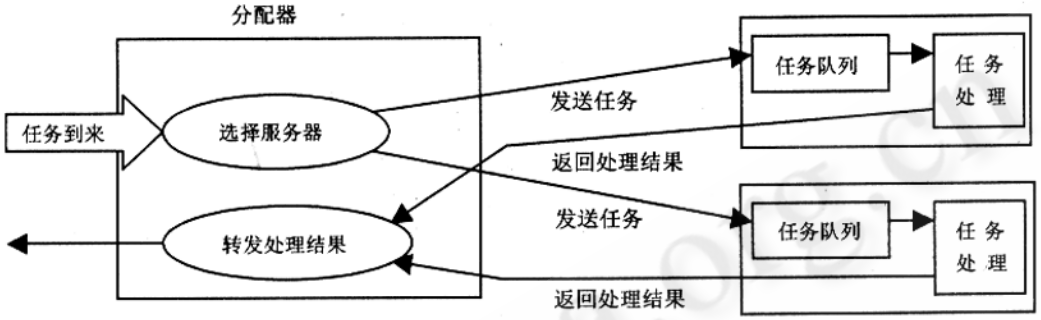
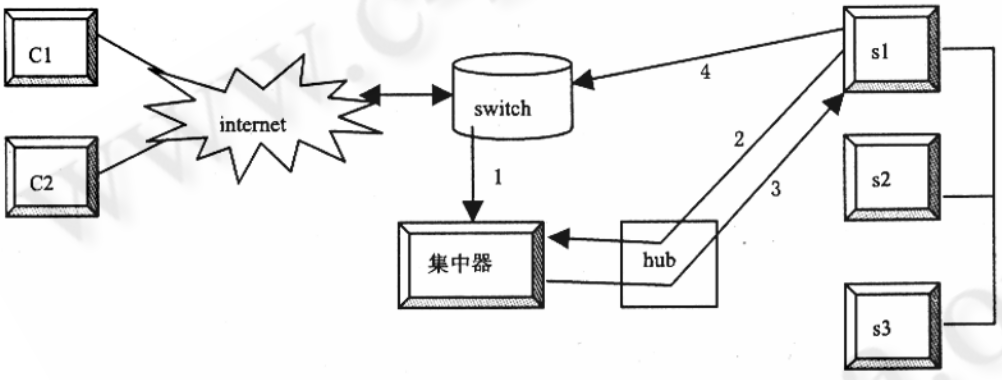


图 2　传统服务器集群调度框架



1 用户提交任务请求报文

2 后端执行服务器(Server)向集中器索取任务

3 集中器从任务池中取出请求报文分发给后端执行服务器

4 执行服务器处理用户请求并向客户返回结果

图 3　ASACS 体系结构

传统的调度机制一般采用基于分配器的负载均衡技术，而实现负载均衡需要实时地收集各服务器的负载信息，从而将新到的任务发送到负载较轻的服务器。然而以上方法存在以下不足：

第一，由于服务器的异构性以及任务队列的异构性(任务类型的多样性)，服务器的实际负载很难准确地获得，容易造成负载倾斜，不能有效地利用资源；

第二，大规模用户对服务器集群的访问往往具有突发性，大量请求在短时间内到达，由于分配器会立即分发每一个连接请求，使得后端服务器重载，后端服务器上大量的来到请求中断会消耗大量的系统资源，影响其有效利用率。严重的情况下还会造成活锁(即高优先级的网络中断频繁打断低优先级的服务进程，使得服务进程没有足够的 CPU 时间处理服务请求)，使用户请求长时间得不到响应，整个集群性能得不到应有的发挥，服务质量严重下降。

第三，这种基于前端分配器的调度机制不能较好地进行 QOS 控制，容易造成服务不公平。

由上可知，传统的集群调度机制远远不能满足实际的需要，需要进行改进以提高服务器集群的服务质量和性能。

3　ASACS 的提出

国防科大的金士尧教授提出了一种与传统完全不同的集群服务器系统体系结构[2]，叫主动自调度集群系统(ASACS)。在这种体系结构中，连接请求的分发受各后端执行服务器的控制，执行服务器在一定的容量规划下工作，从而保证了服务器资源的有效利用和合理的服务质量。在这里，传统意义上的负载平衡工作变得没有必要。集群体系结构如图 3 所示。

和传统的基于分配器调度机制不同的是，请求报文不是由分配器经过负载平衡策略计算后，转发给某一个后端服务器去应答，而是集中器将请求报文分级缓存在缓冲区内，由后端服务器根据自身的忙闲情况主动地到缓冲区去申请。这种后端服务器主动自调度的集群技术具有分布控制的特点，不需要集中器进行集中的负载平衡和调度。ASACS 调度框架如图 4 所示。

4 ASACS 的优点与不足

4.1 优点

(1) 从排队论的观点来看,基于分布式调度机制的主动自调度集群服务系统体系结构与传统的基于分

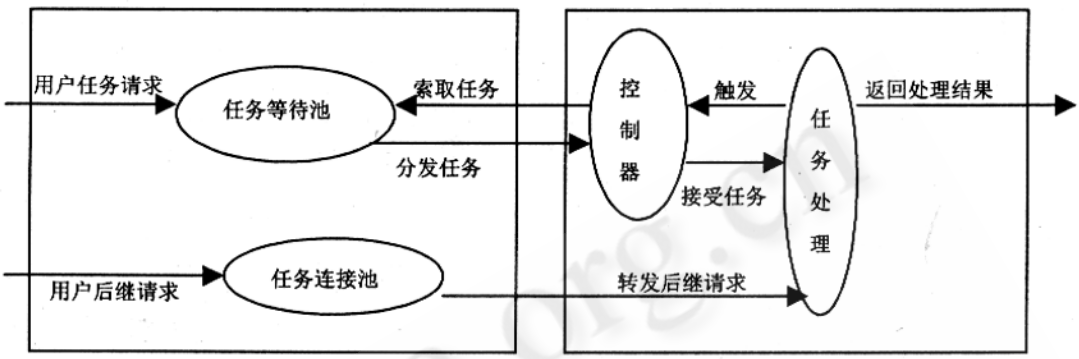


图 4　ASACS 调度框架

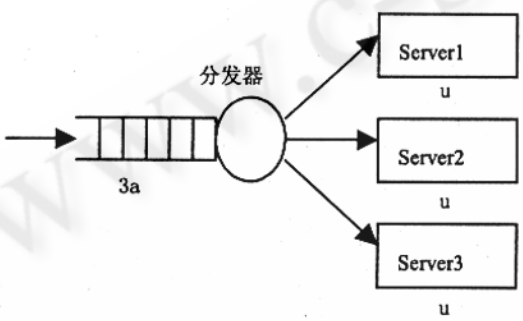


图 5

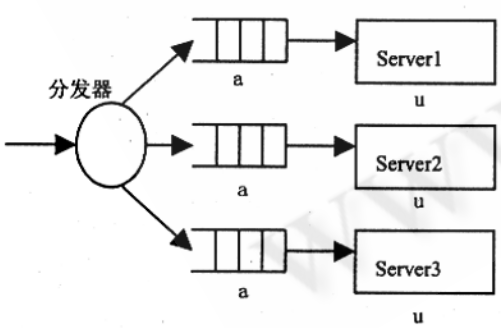


图 6

发器集中调度的体系结构体现了不同的排队论模型。在基于分发器集中调度的体系结构中,采用分发器集中式的调度机制,用户请求在分发器处直接转发,在执行服务器处形成等待队列,实际上是由多个单队列、单服务台的简单排队系统组合而成的复合排队系统。而在主动式集群体系结构中,用户请求都在集中器处缓存,执行服务器根据实际的处理能力从集中器获取请求,实际上是单队列(集中器处),多服务台(各执行服务器)的排队系统。假设用户请求到达速率服从泊松分布,服务器处理单个请求时间服从指数分布,采用分布式调度机制的集群系统可用 M/M/c 的排队系统近似表示,采用传统调度机制的集群系统可用多个 M/M/1 的排队系统近似表示,图 5 和图 6 显示了以三台服务器为例的两种调度机制的调度模型。

其中,3a 表示用户请求平均到达速率,u 表示服务器平均处理速率。假设每台服务器处理能力相同,并且在两种不同的调度机制下,用户请求都能均匀地分发到各个服务器。由排队论的理论分析可知[3],在上面两图所示的条件下,单队列服务模型在响应时间上优于多队列服务模型,即基于执行服务器主动调度策略的集群体系优于传统集群系统。

(2) ASACS 能更好地支持异构性,并且克服了由于异构导致的负载估计不准确问题,只要还有空闲能力就去申请任务,多能者多服务,充分发挥各服务器的资源,在不超过集群负载的前提下保证执行服务器得到最大的利用率。

(3) 集中器在调度方面减少了大量的工作,可以把更多的资源用于其他方面,比如拥塞控制等 QOS 控制。

(4) 摒弃了传统的负载平衡工作,消除了高负载下服务器的"活锁"现象,提高了服务器的有效利用率,保证服务器可预测的响应时间,并且为实现区分服务提供了基础。

4.2 不足

相对于传统的集群系统，ASACS 具有无比的优势，但是它也有不足的地方。

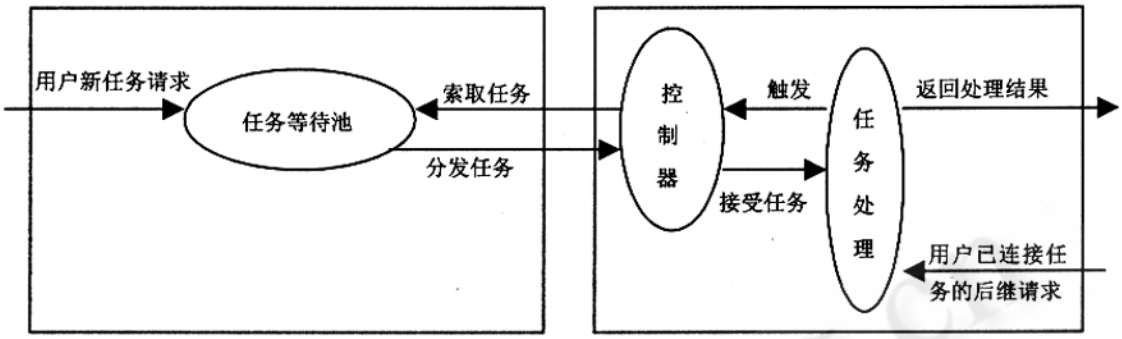
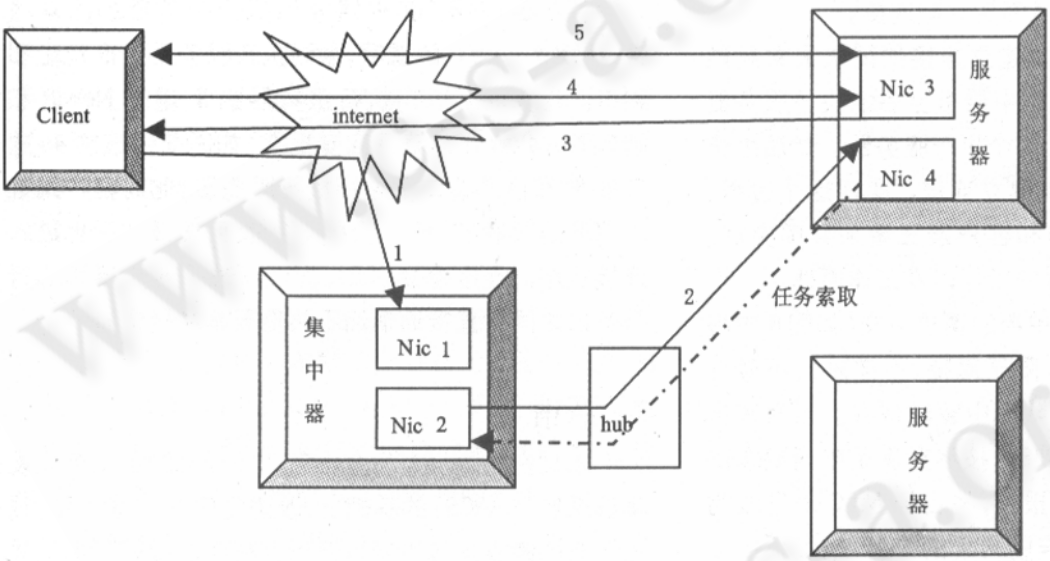


图 7　NASACS 调度框架



（其中为 Nic1，Nic2，Nic3，Nic4 为网卡）

1 客户提交新连接请求报文

2 集中器响应执行服务器的请求从任务池中取出请求报文分发给该执行服务器

3 执行服务器响应用户连接请求并向客户返回回复报文

4 客户将确认回复报文直接发给执行服务器，此后连接建立

5 客户和执行服务器直接通信

图 8　NASACS 建立 TCP 连接的过程

(1) 在 ASACS 上，集中器需要保存各后端执行服务器的连接信息，已连接的任务请求的后继请求也会发往集中器，然后由集中器转发给相应的执行服务器，这样会加重集中器的负担。而这部分后继请求本应该属于相应的执行服务器的工作。

(2) ASACS 的集中器只有一个网卡，而这个网卡既要不断地接收从客户端发来的报文，又要接收从后端执行服务器发来的任务申请报文，还要分发新任务以及已连接任务的后继报文。集中服务器的网卡会是一个传输瓶颈，严重阻碍集群的服务性能。

下面一节，我们改进了这些缺点，提出了新的体系结构和调度框架，称它为 NASACS (New ASACS)。

5　NASACS (New ASACS—新主动自调度集群系统)

在 ASACS 中，每一个已连接请求的后继请求都需要经过前端集中器转发，这不仅给集中器带来了负担，而且由于拥塞控制等 QOS 控制功能从后端执行服务器转移到了前端的集中器上，后继请求的干扰会影响到集中器的拥塞控制等 QOS 控制的实现。

为了解决上述问题，下面提出 NASACS，它改进了 ASACS 关于建立连接的机制，使集中器只负责新任务请求的接受，存储和响应分发请求的工作。由执行服务器和客户之间进行连接的建立过程以及连接建立后的相互直接通信。这样已建立连接用户的后继请求就会直接发给执行服务器而不会再需要经过集中器转发。NASACS 调度框架如图 7 所示。

在体系结构上，为了提高传输性能，给集群中每台服务器配置双网卡，一个用于集群内服务器之间的通

信,另一个用于与集群外的网络设备通信。如图 8 所示,集中器的网卡 Nic1 只接收客户的新任务请求报文,并且是单向传输的;Nic2 用于向执行服务器分发任务和接收任务申请报文;Nic3 用于和客户端进行直接的数据传输;Nic4 用于向集中器发申请任务报文和接收申请到的任务报文。这种传输专用化的配置有利于缓解传输拥塞。

6 NASACS 的实现

目前的网络服务模式基本上都是基于客户/服务器模式。一般地,由客户端向服务器发起一条 TCP 连接请求,服务器回复连接请求,然后客户端确认服务器的回复。通过三次握手,从而建立起连接。

在 NASACS 中,客户端建立连接的请求报文只能发往前端集中器,然后由集中器根据后端执行服务器的申请将请求报文分发给一个执行服务器。然后由该执行服务器和客户端直接建立连接。实现这个过程,需要集中器,执行服务器和客户端三者来共同完成。下面分别介绍它们需要做的工作以及工作原理。

(1) 集中器。到达集中器的请求报文(如图 8 中报文 1)如果满足接纳条件(限于篇幅,不详述),不做任何修改,将报文插入请求队列中暂时保存。当有来自后端的任务申请报文达到时,按尽量满足申请要求的原则,选择足够多的请求报文,将目标 MAC 地址改为与相应的执行服务器相连的网卡的 MAC 地址后的新报文(如图 8 中报文 2)发给该执行服务器。

(2) 执行服务器。任务被分发到执行服务器后,在储备队列中等待,根据容量控制算法(限于篇幅,不详述)进入并发服务队列。首先是建立连接的过程,服务器向客户端发送响应报文 SYN ACK,源 IP 地址和源端口分别为请求报文(如图 8 中报文 3)的目的 IP 地址和目的端口(即集中器的 IP 地址 C_IP 和端口 C_Port),这样客户端就会认为得到了正确的回复。为了以后可以直接和客户端通信,需要把执行服务器自己的 IP 地址和端口作为 SYN ACK 的附带信息发送给客户端。

让 SYN ACK 携带执行服务器的地址信息,我们通过 TCP 协议头选项来实现。TCP 数据段的起始部分是一个固定的 20 字节的头。固定头之后可能跟着头选项。选项之后可以有数据,也可以没有数据。

选项域提供了一种增加额外设施的办法,可以是 0 到多个 32 位字。所有选项以 1 字节的 kind 字段开头,指定选项类型[4]。我们增加一个选项,选项类型为 0,长度字段后是四个字节的 NewIP 和两个字节的 NewPort 两个字段,分别记录执行服务器自己的 IP 地址和端口,如图 9 所示。这两个字段加上一字节的类型字段和一字节的长度字段刚好是 8 字节,所以不需要填充字节。

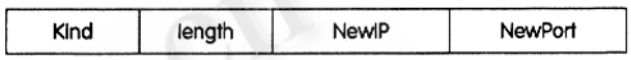


图 9　TCP 头中增加的一个选项

(3) 客户端。客户端收到执行服务器的回复报文,读取类型为 0 的选项,得到执行服务器的 IP 地址和端口。这样就可以用执行服务器的 IP 地址 NewIP 和端口 NewPort 作为确认回复报文(如图 8 中报文 4)的目的 IP 和目的端口发送给执行服务器,此时客户端进入 TCP 连接状态,而执行服务器收到该报文后也进入连接状态。连接建立以后,就可以进行客户端和执行服务器之间的直接通信而不再需要集中器中转。

7 总结

本文在国防科大的金士尧教授提出的主动自调度集群系统(ASACS)的基础上,改进了它的一些不足,提出了新的建立连接机制以及调度框架。减轻了前端集中器的负担,也为以后在集中器中的拥塞控制等 QOS 控制提供了优越的条件。通过服务器的双网卡配置,使传输链路专用化,提高了传输性能,缓解了传输的瓶颈。

参考文献

1 陈智勇、杨辉华、蔡国永,集群计算中的负载共享策略[J],桂林电子工业学院学报,2001,21(4):23226.
2 王晓川、叶超群、金士尧,一种基于分布式调度机制的集群体系结构[J],计算机工程,2002,28(8):2322234.
3 逯昭义、王思明,计算机通信网信息量理论[M],北京:电子工业出版社,1997.
4 Andow S. Tanebaum 著,潘爱民 译,计算机网络(第 4 版)[M],清华大学出版社,2004.