第 35 卷 第 4 期 计 算 机 工 程 2009 年 2 月
Vol.35 No.4 Computer Engineering February 2009
·软件技术与数据库· 文章编号：1000—3428(2009)04—0072—03 文献标识码：A 中图分类号：TP311

分布资源管理信息服务的研究与实现

曾 琼
(南京政治学院上海分院军事信息管理系，上海 200433)

摘 要：针对分布资源管理中的信息服务问题，通过建立统一的资源信息模型，设计并实现能够集中、统一提供信息服务的资源信息服务器，测试和分析该资源信息服务器对整个系统性能的影响。结果证明能够提高整个分布式系统的性能。
关键词：分布资源管理；资源信息服务；资源信息模型

Study and Implementation of Distributed Resource Management Information Service

ZENG Qiong
(Military Information Management Department, Nanjing Political Institute Shanghai Branch, Shanghai 200433)

【Abstract】 Aiming at the information service in distributed resource management, this paper proposes a uniform resource information model, a resource information server which provides centralized, uniform information service. After testing and analyzing the impact to the performance of the whole system, it is proved the method can improve the whole system.
【Key words】 distributed resource management; resource information service; resource information model

1 概述

在分布计算环境中，高效、可靠的分布资源管理方案为高效、可靠的分布高性能计算提供了有力的保证。目前典型的几种资源管理系统在目标、结构、功能和实现上各有差异，从不同侧面反映了资源管理系统应具备的特性。PBS, CONDOR, LSF, LOADLEVELER 是当今颇具代表性和影响力的几种资源管理系统。其中，PBS, CONDOR 是研究产品，LSF, LOADLEVELER 是商业软件。它们基本都采用单一集中式全权管理，使管理的资源规模和种类受限；各个系统都有自己的一套资源信息的采集和表示模式，各个系统之间很难进行互操作；资源信息的获取基本都是直接与资源实体打交道，当系统扩大到一定规模时，对资源信息的获取会在很大程度上影响整个系统的效率。用标准的资源信息模型表示各资源的相关信息，采用像目录这样可扩展的资源信息组织方式，实现在多域之间共享资源信息并提供信息服务，可以为整个系统提供统一、高效的资源信息服务，并提高整个系统的性能。本文以 PBS 作为资源管理研究和改造的对象，单独的资源信息服务模块提供前和提供后的 PBS 系统结构如图 1、图 2 所示。

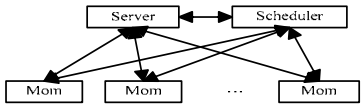

图 1 PBS 结构

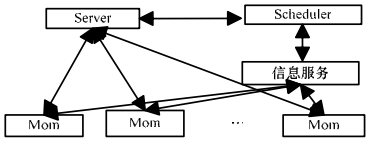

图 2 改进后的 PBS 结构

2 资源管理模型

统一、标准的资源管理模型是实现集中式资源信息服务的基础。分布、多域环境中的资源通常包括计算资源、存储资源、网络资源、数据资源等。通用信息模型(Common Information Model, CIM)是 DMTF(Distributed Management Task Force)提出的一种用于管理信息和系统的面向对象的方法。它将信息模型的经典概念与对象模型结合起来，吸收了两者的长处，从而建立了一个分层的信息模型。

CIM 模型中定义了核心模型、通用模型和扩展模型 3 层结构[1]，能够很好地表示分布环境中的各种资源信息。但是，CIM 模型相当复杂，分布环境中的资源管理并不需要实现 CIM 模型的全部模块，选择 CIM 模型中的 Cluster(机群系统)、Queue(队列)、Job(作业)、Host(节点)、Resource(资源)等几类元素进行进一步扩展和实现。扩展后的资源信息模型如图 3 所示。

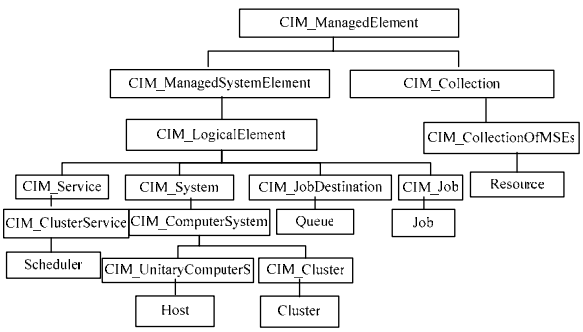

图 3 扩展的机群系统资源信息模型

作者简介：曾 琼(1977－)，男，博士研究生，主研方向：信息集成，知识工程
收稿日期：2008-08-05 **E-mail：**q_zeng@163.com

为了保证资源信息服务的扩展性，采用基于目录的 LDAP 协议实现资源信息模型。具体的映射和实现机制在文献[2]中进行了研究。实现后的命名模型如图 4 所示。顶端 Computing Environment 是一个全局的环境，本文针对这个计算环境进行讨论。在该环境中可能有一个或多个同构或者异构的 Cluster 以及作业集合 Job。一个 Cluster 可以对应本文讨论的一个机群系统资源信息模型。在 Cluster 下面分别是 Scheduler, Queue, Host, Resource 和 Relation。其中，Scheduler 是这个机群的局部调度器；Queue 是局部的队列集合；Host 是这个机群中的全部节点集合，而且其中的某个 Host 可以是其他的 Cluster；Resource 是该机群系统的全部资源集合；Relation 则是一个关系集合，通过关系集合中的关系可以把上述实体联系起来并描述这些实体的某些行为。根据命名模型，在该环境中，机群系统 Cluster1 的节点 node1 的 DN 为：cn=node1, cn=host, ou=cluster1, dc=Computing Environment；而作业 testjob 的 DN 则为：cn=testjob, ou=Job, dc=computing。

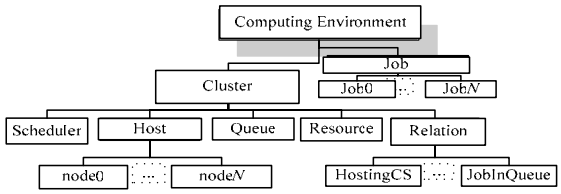

图 4　资源信息模型命名模型

3　资源信息服务

在 PBS 中，各个节点的信息由 Scheduler 在需要时从各个 Mom 和 Server 处直接获取。这样当系统规模扩大到一定程度时，Scheduler 与每个 Mom 通信获取资源信息的消耗就会很大，影响整个系统的性能。为了改善系统性能、提高资源管理的效率，引入一个存储各种资源信息并提供统一信息服务的资源信息服务器。引入资源信息服务器前后的结构如图 1 和图 2 所示。在改进的 PBS 系统中，资源信息服务器需要完成的功能主要包括以下 4 个部分：

(1)收集各种资源信息

资源信息服务的基础是各种资源信息。因此，资源信息服务器首先要能够收集到系统关心的各种资源信息。在系统实现中，分别在创建各类资源和更新信息服务器中的资源信息时收集各种资源信息。

(2)存储资源信息

资源信息通过资源信息服务器的相关接口收集起来后，就要选择合适的存储方式保存起来，供 Scheduler 和其他模块访问。在系统实现中，采用 OpenLDAP 以基于目录的形式存储各种资源信息，以供 Scheduler 和其他模块访问。

(3)更新资源信息

因为资源信息不仅包括节点的 CPU 数量、操作系统类型、物理内存大小等相对静态的资源信息，还包括系统负载状况、作业和队列信息等动态资源信息。其中动态资源信息又可以分为系统负载状况这样随时变化、频繁更新的资源信息和作业、队列信息这样更新不是特别频繁的资源信息。所以，要保证资源信息服务的可靠性就需要及时更新资源信息并选择合适的更新方式和时机。在系统实现中，对几类资源信息的更新采用不同的更新时机和方法。对于相对静态的资源信息，只在相关资源主体退出和加入时更新资源信息。对于作业、队列信息这样更新不是特别频繁的资源信息，分别在新作业达到、作业结束和调度周期达到时更新资源信息。对于节点负载状况这样变化频繁的资源信息，采用定期更新的机制，并且使这个更新周期可以根据系统和应用状况进行调整。这样，就可以在系统效率和资源信息的可靠性之间达到一个最佳的平衡。

(4)为 Scheduler 模块提供访问节点信息的接口

资源信息服务器的目标是为系统的调度或者其他应用提供可靠的资源信息服务。所以它应该对外提供统一、方便的访问接口，以便系统其他模块访问资源信息。这里分别针对几类不同的资源对象提供了几种常用的功能接口。

因为 PBS 本身就有一个集中管理的 Server 模块，而 LDAP 协议又是 Client/Server 结构的，所以资源的相关信息以目录形式存储在 LDAP Server 中，Mom 和 Scheduler 通过 LDAP 协议使用 LDAP 协议提供的存储和访问接口，以 LDAP Client 的形式读取或者修改 LDAP Server 中存储的资源信息。而且从系统的整体效率来考虑，资源信息服务器无须在系统形成一个单独的模块，运行时也没有必要再使用一个单独的 daemon，而是将 LDAP 的 Client 端和 Server 端分别嵌入 PBS 的 Server, MOM 和 Scheduler 之中，在运行时分别利用 LDAP 协议进行通信。本系统采用 LDAP 协议实现资源信息服务器。

增加了资源信息服务器后的 PBS 系统结构如图 5 所示。

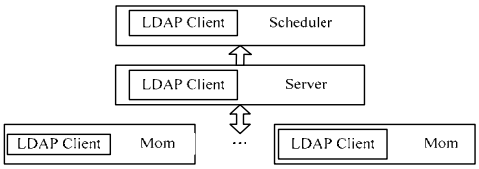

图 5　资源信息服务器结构

4　系统性能测试及结论

影响分布资源管理系统性能的主要因素通常不是该系统中的调度和管理节点的物理计算能力，而更多地取决于该系统的结构、组织调度算法等。如果一个分布资源管理系统的结构不好，组织和调度算法效率较低，那么调度和管理节点的性能再强大，也不能对提高整个资源管理系统的性能有很大的帮助。资源管理性能的提高意味着获取资源信息的时间缩短，减少调度开销在整个作业运行周期中的比例。

系统的利用率是指作业在系统中的实际运行时间与理想运行时间的比例[3]。影响系统利用率的因素主要包括获取资源信息的时间、作业调度时间和系统的负载状况。如果每个作业的运行时间是固定的，则工作负载在系统中运行所需要的时间只依赖于调度策略和资源管理的性能。因此，可以在固定的调度策略和相对稳定的系统负载状况下，依靠测试系统的利用率衡量资源管理系统的性能，分析资源信息服务对资源管理系统性能的影响。

4.1　测试程序

常见的测试基准程序(如 LINPACK 和 NAS)常用于测试系统的计算性能，但是很少涉及系统的效率问题，而本文最关心是系统的效率。需要获取系统的效率衡量统一的资源信息服务对资源管理性能的影响。因此，选择 NERSC(National Energy Research Scientific Computing Centre)设计的 ESP (Effective System Performance)测试基准程序进行系统利用率评测。

ESP 测试基准程序是 NERSC 设计的测试套件，它包括适用于不同规模系统的 3 个应用程序 qcd, tlbe, superlu；一组

14 类共 230 个作业包[4]。它提供着重于测试并行系统属性的一种度量方法。这些系统属性包括并行启动、作业调度、强制优先作业和系统重启时间。这种度量方法与处理器的速度无关，与文件系统这样的共享资源的相关性也很小，能够用于在不同的平台和系统之间进行比较。

在 ESP 测试包中使用的测试负载由 14 类 230 个作业组成。为了获得准确的运行时间，一旦测试开始就不允许再有其他的干预。这里，测试被用于 4 个节点 8 个 CPU 的系统中。而且，测试目标是衡量统一的资源信息服务对资源管理性能的影响，只要能够得出加入统一资源信息服务后对资源管理系统性能的影响，即在有和没有统一资源信息服务的资源管理系统中的系统利用率的差值，就达到了测试目标。所以，选择了 ESP 测试套件中提供的 3 个应用程序适用于小规模的 superlu 进行测试，并根据系统的实际情况对 ESP 测试套件中的作业规模和作业运行时间进行了调整，所使用的测试作业如表 1 所示。

表 1　测试作业列表

作业类型	单作业运行时间/s
A	52
B	104
C	263
D	528

通过调整上述几类作业在实际运行中的数目，达到了调整激活调度事件次数的目的。而随着调度事件的发生，需要获取资源信息，这样就可以衡量通过统一的资源信息服务获取资源信息与直接从计算节点获取资源信息的差别。

4.2　测试环境和结果

本文所使用的测试环境是一个 4 节点 8 个 CPU 的机群系统。具体测试环境如下：

机群系统为 4 个双 CPU 节点。其中 4 个节点配置为：双 CPU，Pentium 4 2.4 GHz 至强，内存 1 GB，千兆网卡，操作系统为 RedHat Linux 9.0，使用千兆网连接。

这里的时间都使用墙钟时间，以秒为单位。分别提交了 6 组测试作业。在没有加入资源信息服务器的 PBS 系统中，将测试作业提交后的测试结果见表 2。

表 2　测试结果 1

作业类型	单作业运行时间/s	作业数量					
		第 1 组	第 2 组	第 3 组	第 4 组	第 5 组	第 6 组
A	52	20	60	140	400	680	830
B	104	10	20	40	80	100	150
C	263	10	10	10	10	10	10
D	528	10	10	10	10	10	10
总作业数量/个		50	100	200	500	800	1 000
运行时间/s		1 789	2 241	3 230	6 751	9 017	11 668
系统利用率		0.698 0	0.731 3	0.671 4	0.685 6	0.744 0	0.714 2

在同样的测试环境中加入了资源信息服务器的 PBS 系统中提交相同的 6 组测试作业，相关数据如表 3 所示。

表 3　测试结果 2

作业类型	单作业运行时间/s	作业数量					
		第 1 组	第 2 组	第 3 组	第 4 组	第 5 组	第 6 组
A	52	20	60	140	400	680	830
B	104	10	20	40	80	100	150
C	263	10	10	10	10	10	10
D	528	10	10	10	10	10	10
总作业数量/个		50	100	200	500	800	1 000
运行时间/s		1 818	2 263	3 277	6 742	8 975	11 534
系统利用率		0.686 7	0.724 0	0.661 7	0.686 5	0.747 5	0.722 5

每组作业在加入资源信息服务器和没有加入资源信息服务器的系统中的利用率，以及在加入资源信息服务器的新系统和原系统中的利用率差值如表 4 所示。

表 4　测试结果 3

作业	原系统利用率	新系统利用率	系统利用率差值
第 1 组	0.698 0	0.686 7	−0.011 3
第 2 组	0.731 3	0.724 0	−0.007 3
第 3 组	0.671 4	0.669 7	−0.001 7
第 4 组	0.685 6	0.686 5	+0.000 9
第 5 组	0.744 0	0.747 5	+0.003 5
第 6 组	0.714 2	0.722 5	+0.008 3

4.3　测试结果分析

在该测试环境中，由于调度策略没有发生变化，并且在加入资源信息服务器之后和没有加入资源信息服务器的系统中执行的是一组相同的作业集合，因此系统负载可以近似认为没有变化。这样，在这 2 个系统中系统利用率的差值就可以反映资源信息服务器给系统性能带来的影响。本文以系统利用率在加入资源信息服务器的系统中和没有加入资源信息器的系统中的差值为横轴，以运行的每组作业中的作业数量为纵轴，见图 6。

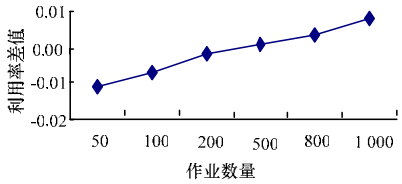

图 6　作业数量与系统利用率差值关系

由图 6 可以看出，在 4 节点 8 个 CPU 的系统中，当作业数量达到一定程度(500 个)后，系统利用率的差值开始大于 0。这就意味着在调度事件发生足够多次的情况下，资源信息服务器开始给资源管理系统的性能带来好处。从图 6 还可以看出，随着作业数量的增加，系统利用率的差值也在增加。这也意味着随着调度次数的增加，资源信息服务器能够逐渐改善资源管理系统的性能。

随着作业数量和调度次数的增加，有资源信息服务器的系统和没有资源信息服务器的系统的利用率差值从小于 0 逐渐增加到大于 0。这是因为在这样一个小规模的系统中，每次直接从计算节点获取资源信息的消耗并不是很大，而资源信息服务器本身也有一定的消耗。因此，在作业数量不大、调度次数不够多的情况下，资源信息服务器并不能给整个资源管理系统性能带来好处。

随着作业数量和调度次数的增多，直接从计算节点获取资源信息的消耗开始积累，但是资源信息服务器的消耗并不会随作业数量的增加和调度次数的增多而增加，因此，当作业数量增加、调度次数增多到一定程度之后，资源信息服务器给系统性能带来的好处开始显现，系统利用率的差值也由小于 0 逐渐增大为大于 0。如果系统规模足够大(如 128 个节点)，因为每次直接从各个节点获取资源信息的消耗比较大，而资源信息服务器本身的开销并没有很大的变化，所以资源信息服务器给整个资源管理系统性能所带来的好处将能够更加容易、充分地体现。

因此，在一定规模和发生足够多次调度事件的情况下，统一的资源信息服务机制能够改善资源管理系统的性能。

(下转第 77 页)