

# 计算网络的资源分发和发现机制<sup>\*</sup>)

武秀川<sup>1,2</sup> 鞠九滨<sup>1</sup>

(吉林大学计算机学院 长春 130012)<sup>1</sup> (烟台大学计算机学院 烟台 264005)<sup>2</sup>

## The Mechanism of Resource Dissemination and Resource Discovery for Computational Grid

WU Xiu-Chuan<sup>1</sup> JU Jiu-Bin<sup>2</sup>

(College of Computer Science and Technology, Jilin University, Changchun 130012)<sup>1</sup> (College of Computer, Yantai University, Yantai 264005)<sup>2</sup>

**Abstract** Computational Grid is a large-scale distributed computing environment. The resource management of computational Grid discovers and locates and allocates resources for users within the filed of grid environment as they have a request to these resources. The other case for that is co-operating in order to finish a large computing. These tasks are accomplished by the mechanism of resource dissemination and resource discovery of the resource management for the grid system. In this paper, some problems about resource dissemination and resource discovery are discussed and analyzed, further more future work about that is proposed.

**Keywords** Computational grid, Resource management, Resource dissemination and resource discovery

## 1 引言

计算网络的资源管理系统是为实现计算网络系统资源共享所应提供的最主要的服务之一。计算网络资源管理系统的基本功能是接受来自计算网络范围内的机器的资源请求,并且把特定的资源分配给资源请求者,并且合理地调度相应的资源,使请求资源的作业得以运行。资源分发、资源发现和资源的调度构成了计算网络资源管理系统的最主要的内容。资源分发和资源发现提供方法,通过该方法,在计算网络内部的机器能够形成一个可用的资源和其状态的一个视图<sup>[1]</sup>。资源发现和资源分发提供相互补充的功能。资源分发提供有关机器资源的信息或一个源信息资源的指针。而资源发现是一个机制,通过该机制应用能够发现资源信息。资源发现由网络应用启动并在网络范围内发现合适的资源,即资源发现从其他结点把资源的信息拉(pull)到一个终端系统上。而资源分发由资源启动并试图去发现能够利用该资源的合适的应用。换句话说,资源分发把来自于具有资源的终端系统的有关资源的信息推(push)到其它结点。通常通过使用一个资源信息协议来分布计算网络环境中的资源信息,而资源信息协议由资源分发功能所实现。资源调度被使用来把资源分配到一个请求上去,包括通过不同结点的资源的协作分配。

本文对所采用的两类最主要的资源分发和资源发现方法进行了深入的比较分析讨论和研究,并提出进一步的工作。

## 2 资源分发和资源发现的方法

支持计算网络系统的位置独立的计算的最基本操作之一是资源发现。通常,资源发现模式维持和查询一个资源状态数据库,而资源状态信息分发是保持资源状态数据库一致性的关键操作之一。

就计算网络所采用的资源分发和资源发现策略大致可被划分为两类:基于数据库查询的方法和基于代理(agent)的方

法。

### 2.1 基于数据库查询的资源分发和资源发现方法

资源发现依赖于资源信息分发,而资源信息分发依赖于更新资源信息的方法。使用来实现资源分发的机制也决定在计算网络中使用的机器之间传送的数据量<sup>[5,6]</sup>。资源信息在计算网络中传送的数据延迟也由分发机制所决定。

一般来说,资源发现模式能够通过维持一个全局的数据库和对它进行查询来实现,因此涉及到状态数据库的组织方式、数据库如何被更新和何时更新等因素。

状态数据库的组织形式可以是:集中式的、分布式的和层次式的等三种<sup>[4]</sup>。与分布式和层次式的组织方法相比较,集中式的组织方式容易产生单点失效问题,并且降低可扩展性,而分布式组织方式为了维持状态信息的一致性增加了消息的传递量而导致了通讯代价的增加,该方式是容错的并且具有比较低的查询时间,因为查询是在本地和相同的网络段的服务器内被处理。层次结构组织把集中化方法的低更新代价的优点与分布式的低查询代价和容错的优点结合起来。

状态信息数据库可以基于某些事件周期性的或非周期性的进行更新。这些事件可能包含所管理的对象的状态有效的变化,或是新资源请求的到达<sup>[1]</sup>。通常采用三种更新状态信息数据库的方法。第一种是从管理的对象中把更新信息推到状态数据库。第二种是从所管理的对象中的传感器把状态信息拉到状态数据库。第三种方法以一种对称的方式结合上述两种模式的特点:在所管理的对象中把更新信息推到一个中间数据库并且在需要的时候把这些信息从这些中间数据库中拉出来。

计算网络环境的动态特性意味着程序包组件、程序设计组件以及应用必须能够适应它们的行为以响应系统结构和状态的变化。Globus的MDS(Globus Metacomputing Directory Services)支持这种类型的适应性,方法是通过提供一个信息丰富的环境,在此环境中有关系统组件的信息总是可用的。

<sup>\*</sup>)本文受到国家自然科学基金项目(69873018)的资助。武秀川 博士研究生,副教授,研究领域为分布式系统。鞠九滨 教授,博士生导师,研究领域为分布式系统和计算机网络。

MDS 存储和生成可访问的信息如体系结构类型,操作系统版本,计算机的存储容量,网络带宽和延迟,可用的通讯协议,以及 IP 地址和网络技术之间的映射。MDS 提供了一组工具和 APIs 用于发现、公布和访问有关计算网络的结构和状态的信息。在这种情况下,正在被使用的标准是由 LDAP 所定义的数据表示和 API。LDAP 定义了一个层次的树状结构的名字空间,该名字空间被称之为目录信息树并且被设计来作为一个分布式的服务:任意的子树能够与不同的服务器相连接,因此用来支持 MDS 的本地服务其实是一个 LDAP 服务器(如果多个网站共享一个服务器的话,则是到其他的 LDAP 服务器的一个网关)。全局 MDS 服务简单的就是所有这些服务器的总和。

在批处理/周期性的(batch/periodic)<sup>[5]</sup>方法中,资源信息被成批地发送到每个计算网格机器上去然后通过计算网格周期性地分发出去。信息能够被原始机器发送到其他的机器,在这种情况下,它是正在进行信息的推操作,或者计算网格中的其他机器能够从原始机器上请求信息,在这种情况下,它正在进行信息的拉操作。在线或者基于要求的方法中信息从原始机器立即被分发,在这种情况下信息被推到计算网格中的其他机器上。信息被传送到机器的集合依赖于计算网格的组织 and 资源数据库的实现。

有关资源和广域分布式系统中节点的信息有时候需要被用来以协调一组节点的活动。为了提供有关资源利用率的信息,一般所采取的方法是在每个节点安装一个监控器去搜集本地资源的信息,本地监控器可能周期性地更新所存储的数据库,或提供有关要求的信息,有时信息可能被存储在一个层次的服务器中。

## 2.2 基于代理(agent)的资源分发和资源发现方法

代理是一个智能化的程序,它是自治的,更进一步它是可移动的。自治性隐含着代理是主动的对象,具有自己的控制处理。代理也能够显示智能的行为。移动性确保代理能够在快速变化的异构式环境中操作运行。代理可以是一个逐步的发现进程,也能够实现对一组要求改变的响应。这些代理能够适应本地节点的结构和操作环境。基于推断进程的结果,代理也可以改变它的行为,并且赋予代理的任务也可能是相当复杂的。一般来说基于代理的分布式资源发现体系结构中的代理也可被在远程所形成并且当需要的时候可以被修改。代理聚集需要被用来进行资源管理的信息。

基于代理的方法发送主动式代码段跨越计算网格中的机器,这些代码被每台机器在本地解释。代理也能够被动地监控并且也可以或者周期性地发布资源信息或者响应另外的代理。这样代理能够模拟一个基于资源发现模式的查询。基于查询和基于代理的方法之间的主要的不同是基于代理的系统允许一个代理基于它自己的内部逻辑而不是依赖于一个固定的功能查询引擎去控制查询进程并且进行资源的发现。大多数基于代理的系统是强调灵活性和性能。基于代理的资源发现系统本质上是分布式的。

文[7]所介绍的 Bond 系统是一个基于 Java 的分布式对象系统和代理的结构框架。Bond 系统的核心部件是一个对象模型的核心容器和面向消息的中间件的移动式代码和一个服务层,包含目录服务和存储服务等的分布式服务层。最主要的部件是一个代理结构框架,它提供基本的工具用于生成自治的网络代理和一个数据库,通过该代理和数据库用户可以用很少量的程序设计进行代理的综合。在省缺的情况下,Bond

对象是网络对象,Bond 对象可以是消息的发送者和接收者。Flooding 算法假设每个结点仅仅与连接到它的一组最初的邻居相通讯,并且需要时对它们进行更新。Swamping 算法不仅与连接到它的一组最初的邻居相通讯,而且还可以与目前的邻居结点进行通讯。在随机指针跳越算法(random pointer jump)中,可以连接到任意一个邻居结点进行通讯。与这三种方法比较<sup>[7]</sup>,Bond 系统完全独立于消息传递机制,并且可以交替地使用几个通讯方法,目前提供了基于 TCP、UDP 和 Infosphere 的通讯方法以及多播方法。所有的 Bond 对象通讯使用一个代理通讯语言 KQML。Bond 定义了子协议的概念,它是高度专用的和封闭的一组命令。如属性访问子协议、代理子协议和安全子协议。实际上 Bond 代理体系结构是一些工具的应用,这些工具由 Bond 核心层提供以实现协作网络代理。代理被动态的由一个多平面状态机在一个结构中的部件进行综合。这个结构由一个特殊化的称之为蓝图(blueprint)的语言所描述。主动式的组件就是策略在本地或远程被装载,或被嵌入在 blueprint 脚本程序中的一个解释性的语言所定义。代理的状态信息和知识被聚集在一个单个的称之为“世界模型”的对象中,该对象使得可以非常容易地实现代理的检查点和迁移。代理运行在单个的结点上,使用一个称之为“关注分布”(distributed awareness)的机制并且知道其他结点的存在性,每个代理都维持有有关其他代理的信息,而这些代理与该代理进行周期性的通讯并且周期性地进行这些信息的改变。只要一个代理需要有关系统的独立部件的相关信息,则使用由关注分布机制所搜集的信息然后动态的聚集能够报告远程资源的状态的代理去协商这些资源的使用。

## 3 进一步的工作

一般来说,资源发现服务使用由网络范围信息服务所维持的“状态”数据库实现客户机的请求。对于可扩充的实现来说,以一种分布式的方式组织状态数据库是重要的。因为对于状态数据库来说应该以分布式方式组织,请求能够被很有效地执行,但是对数据库的更新代价比较大。而大部分的更新代价是由在计算网格范围内去分发状态信息所完成的通讯操作所引起的。如何有效地减少资源分发的通讯代价是计算网格资源管理系统中的一个十分重要的问题。对此很多系统进行了很有意义的工作。文[5]把资源发现定义为一个进程,通过该进程在一个分布式系统中的一个结点或机器注意到系统中的其他结点的属性和能力。在计算网格系统中,资源发现的进程主要由一个结点所使用,用以发现一组最好的资源以便能够运行一个作业或提供特定的服务。该问题被考虑作为一个特殊的资源发现情况。计算网格是由各种异构式的机器和网络所组成,并且从性能方面来考虑某些机器可能比其他机器更有效。因此,通过根据机器的有效性控制机器的状态分发的程度就可以减少数据分发的代价<sup>[5]</sup>。文[5]通过引入计算网格潜能(potential)的概念封装了组成计算网格的不同机器和网络的相对处理能力来实现这一目标。

资源发现或者是分布式的查询或者是基于代理的。集中化的查询不适合于工作在大规模的计算网格系统中。为使得计算网格的资源管理系统提供一个综合的环境,可使用一种综合的方法实现资源的发现,在这种情况下分布式的查询给予有关资源信息的部分结果,而通过使用代理的方法获得所有资源的信息。如用代理从管理的对象中把更新信息推到状

(下转第 24 页)

- Wireless Link Sharing via enhanced Class-Based Queuing with Channel-State-Dependent Packet Scheduling. In: Proc. IEEE INFOCOM '98, April 1998
- 13 Mogul J, Deering S. Path MTU Discovery. Request For Comment 1191, Nov. 1990
  - 14 Degermark M, Nordgren B, Pink S. IP Header Compression. Request For Comment 2507, Feb. 1999
  - 15 Casner S, Jacobson V. Compressing IP/UDP/RTP Headers for Low-Speed Serial Links. Request For Comment 2508, Feb. 1999
  - 16 Bormann C, Burmeister C, et al. ROBust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed. Request For Comment 3095, July 2001
  - 17 Shacham A, Monsour R, et al. IP Payload Compression Protocol (IPComp). Request For Comment 2393, Dec. 1998
  - 18 Braden B, Clark D, et al. Recommendations on Queue Management and Congestion Avoidance in the Internet. Request For Comment 2309, April 1998
  - 19 Allman M, Paxson V, Stevens W. TCP Congestion Control. Request For Comment 2581, April 1999
  - 20 Caceres R, Iftode L. Improving the Performance of Reliable Transport Protocol in Mobile Computing Environment. IEEE Journal of Selected Areas in Communication, 1995, 13(5): 850~857
  - 21 Mathis M, Mahdavi J, Floyd S, Romanow A. TCP Selective Acknowledgement Options. Request For Comment 2018, Oct. 1996
  - 22 Wang S Y, Kung H T. Use of TCP Decoupling in Improving TCP Performance over Wireless Networks. ACM/Baltzer Wireless Networks Journal, 2001, 7(3): 221~236
  - 23 Bakre A, Badrinath B R. I-TCP: Indirect TCP for Mobile Hosts. In: Proc. of 15<sup>th</sup> Intl. Conf. on Distributed Computing Systems (ICDCS), Vancouver, 1995. 136~143
  - 24 Balakrishnan H, Seshan S, Amir E, Katz R. Improving TCP/IP Performance over Wireless Networks. In: Proc. of 1st ACM Conf. on Mobile Computing and Networking (Mobicom'95), Berkeley, CA, Nov. 1995
  - 25 Vaidya N H, Mehta M, Perkins C, Montenegro G. Delayed Duplicate Acknowledgements: A TCP-Unaware Approach to Improve Performance of TCP over Wireless. [Technical Report 99-003]. Computer Science Dept., Texas A&M University, Feb. 1999
  - 26 Karunaharan R, Ibrahim M. WTCP: An Efficient Mechanism for Improving TCP Performance over Wireless Links. In: Proc. of the 3rd IEEE Symposium on Computer and Communications (ISCC '98), Athens, Greece, June 1998
  - 27 Ramakrishnan K, Floyd S. A Proposal to add Explicit Congestion Notification (ECN) to IP. Request for comment 2481, Jan. 1999
  - 28 Bakshi B S, Krishna P, Vaidya N H, Pradhan D K. Improving Performance of TCP over Wireless Networks. In: 1st ACM Conf. on Mobile Computing and Networking (MobiCom'95), Berkeley, CA, Nov. 1995
  - 29 Biaz S, Vaidya N. Discriminating Congestion Losses from Wireless Losses using Inter-Arrival Times at the Receiver. [Technical Report 98-014]. CS Dept., Texas A&M University, Aug. 1998
  - 30 Parsa C, Garcia-Luna-Aceves J J. Improving TCP Performance over Wireless Networks at the Link Layer. ACM Mobile Networks and Applications Journal, 2000, 5(4): 57~71
  - 31 Brown K, Singh S. M-TCP: TCP for Mobile Cellular Networks. ACM Computer Communications Review, 1997, 27(5)
  - 32 Brown K, Singh S. M-UDP: UDP for Mobile Cellular Networks. ACM SIGCOMM Computer Communication Review, 1996, 26(5)
  - 33 Larzon L, Degermark M, Pink S. UDP Lite for Real Time Multimedia Applications. In: Proc. of the QoS mini-conference of IEEE Intl. Conf. of Communications (ICC'99), Vancouver, Canada, June 1999

(上接第13页)

态数据库。也可由在目标结点的资源发现代理从所管理的对象中的传感器把状态信息拉到状态数据库。可以一种对称的方式结合上述两种模式的特点:在所管理的对象中的代理把更新信息推到一个中间数据库并且由在目标结点上的代理把这些信息从这些中间数据库中拉出来。这样采用综合的方法能够构造一个可扩展的和容错的方法。如一个计算网格能够使用一个层次的调度程序跨越自治的管理域并且在每个域中使用发现技术。但是在此综合方法中结合不同的模式所需要的代价问题还不很清楚<sup>[2]</sup>。在文[6]中,计算网格空间被维持作为一个弱一致性的分布式数据库,因为计算网格空间集成所管理对象的属性的方法没有详细的信息用于一个远程的对象。一旦一个资源发现查询通过一个集成的计算网格空间,则把该计算网格的表目缓存在计算网格空间的缓存中,这实际上是基于代理的方法。然而这种方法比基于代理的方法的优越性是随后的对相同对象感兴趣的查询不会导致全盘的开销,一直到缓存的表目超时。

移动式代理(Mobil Agent)作为一个可移动的程序可以在计算网格范围内的主机间移动,可以自主和灵活的发现计算所需要的资源。但由于其复杂性,目前尚未发现它在计算网格系统应用中十分成功的范例。

## 参考文献

- 1 Krauter K, Buyya R, Maheswaran M. A Taxonomy and Survey of Grid Resource Management Systems. Software Practice and Experience, 2002, 32(2): 135~164
- 2 Krauter L, Maheswaran M. Architecture for a Grid Operating System. <http://www.cs.umanitoba.ca/~anrl/PUBS/grid2000>
- 3 Gehring J, Streit A. Robust resource management for metacomputers. In: 9<sup>th</sup> IEEE Int'l Symposium on High Performance Distributed Computing, 2000
- 4 Buyya R, Chapin S, DiNucci D. Architectural Models for Resource Management in the Grid. First IEEE/ACM International Workshop on Grid Computing (GRID 2000), Springer Verlag LNCS Series, Germany, Dec. 17, 2000, Bangalore, India
- 5 Maheswaran M. Data Dissemination Approaches for Performance Discovery in Grid Computing Systems. 10<sup>th</sup> IEEE Heterogeneous Computing Workshop (HCW), Apr. 2001
- 6 Maheswaran M, Krauter K. A Parameter-based Approach to Resource Discovery in Grid Computing Systems. In: 1st IEEE/ACM Int'l Workshop on Grid Computing (Grid'00), Dec. 2000
- 7 Jun K, Boloni L, Palacz K, Marinescu D C. Agent-based resource discovery. In: 19<sup>th</sup> IEEE Heterogeneous Computing Workshop (HCW'00), Oct. 1999