

# 网格计算技术及应用综述<sup>\*</sup>

洪学海<sup>1,2</sup> 许卓群<sup>1</sup> 丁文魁<sup>1</sup>

(北京大学计算机科学技术系 北京 100871<sup>1</sup>) (成都理工大学计算机工程系 成都 610059)<sup>2</sup>

## A Survey on The Technology & Application of Grid Computing

HONG Xue-Hai<sup>1,2</sup> XU Zhuo-Qun<sup>1</sup> DING Wen-Kui<sup>1</sup>

(Department of Computer Science & Technology, Peking University, Beijing 100871)<sup>1</sup>

(Department of Computer Engineering, Chengdu University of Technology, Chengdu 610059)<sup>2</sup>

**Abstract** Grid computing is a new kind of distributed computing technology and computing environment, and also an attentive hot point of information technology in the world. It launches a space for new generation internal application, due to its supporting the complicated service efficiently and useable resource in the Internet. This paper sums up the results and applications instance in the grid computing in the world during recent years, and then analyzes emphatically the system architecture, components, working principle in the grid computing and some typical grid systems, and then discusses the problems of the large scale science computing and network service in grid computing in China, and also points out the future trends.

**Keywords** Grid computing, Grid, Meta computing, Data grid, Distributed computing, Middleware, OGSA (Open Grid Service Architecture)

## 1 引言

网格计算(Grid Computing)是近年来国际上兴起的一种重要信息技术,称之为21世纪的IT技术基础设施。其目的是把因特网整合为一台巨大的超级计算机,将网络上的各种资源,如价钱昂贵的超级计算机、服务器、大规模存储系统、各种工作站、个人PC、可视化设备和贵重的精密仪器设备、各种软件资源和数据库等组织在一个统一的大框架下,为解决大型复杂科学计算、数据服务和网络信息服务提供一个方便用户使用的虚拟平台,实现计算资源、存储资源、信息资源、知识资源等的全面共享,消除信息孤岛和资源孤岛。网格技术也可用于企业内部网、局域网。目前对网格还没有一个非常精确的定义,国外媒体常用:“下一代因特网”、“Internet2”、“下一代Web”来称呼网格相关技术。中国工程院院士李国杰认为<sup>[1]</sup>：“网格是继传统因特网、Web之后的第三个大浪潮,可称为第三代因特网应用。简单地讲,传统因特网实现了计算机硬件的连通;Web实现了网页的连通;而网格试图实现互联网上所有资源的全面连通”。

目前,世界上许多国家对网格技术非常重视。美国、欧洲、日本、印度等国都启动了大型网格研究计划,并得到了产业界的大力支持。专家们认为,计算机工业下一波大浪潮的主线,就是网格技术,它将带来因特网的新生,极大地改变我们的工作和生活。美国《福布斯》杂志预测,网格技术将在2004~2005年出现一个高峰,在2020年产生一个年产值为20万亿美元的大工业。目前国外许多政府部门、研究机构、跨国公司和著名大学的许多科研人员从事网格计算系统的研究,已经开展了许多研究论坛、实验环境和研究项目。较有代表性的网格计算项目包括<sup>[2]</sup>: Globus、Legion、Globe、NetSolve、Javalin

等。本文是对近几年网格计算技术发展所做的一个概述,并分析了网格计算的基础设施及体系结构、工作原理等,探讨了中国网格计算的应用前景。

## 2 网格计算研究与应用现状

网格计算的研究主要在美国和欧洲。网格的理想早在1960年就提出来了,但对网格的大规模研究只是近十年的事。最“正统”的网格研究起源于美国政府过去十年来资助的高性能计算科研项目。20世纪90年代初,根据网上主机大量增加,但其利用率却并不高的情况,美国国家科学基金会(NFS)将其4个超级计算中心构筑成一个能够进行元计算(meta-computing)的整体<sup>[3]</sup>。元计算的含义是通过网络,将计算资源连接起来,形成对用户透明的超级计算环境。元计算的术语已被网格计算所代替。网格方面的代表性研究工作有美国的“国家技术网格(NTG)”、“分布万亿次级计算设施(DTF)”、美国宇航局的IDG、美国能源部的ASCI Grid、美国国防部网格计划——“全球信息网格”(Global Information Grid)以及欧盟的Data Grid。随着网格研究在学术界的加速,信息产业界的大公司也相继公布了与网格目标一致的研究开发计划。惠普、IBM、微软、Sun等公司最近取得共识,支持XML、SOAP、UDDI等万维网标准,从而更有利于开发新一代的网络应用,即万维网服务。惠普推出了eSpeak万维网服务平台;IBM用它的WebSphere平台和一系列中间件实现万维网服务;微软的路线是通过其.Net计划和C#语言实现万维网服务;Sun则通过Sun ONE(Open Network Environment)计划和Java平台来实现它<sup>[4]</sup>。Sun公司在2000年就启动了以网格引擎(Grid Engine)分布式资源管理软件为基础的开放源代码战略。为此,Sun公司发布了“网格引擎”企业版

<sup>\*</sup> 国家自然科学基金项目(60173004)资助。洪学海 博士后,副教授,主要研究方向为高性能计算与环境、数据库系统、人工智能等。许卓群 教授,博士导师,主要研究方向为空间信息与大规模并行处理、人工智能等。丁文魁 教授,主要研究方向为高性能编译器。

5.3 的测试版。IBM 宣布在网格计算领域投资 40 亿美元,以在全球建设 40 家数据中心,从而正式进入网格计算领域。同时,IBM 被英国政府选中,负责提供英国国家网格项目的关键技术。此外,IBM 还宣布了一项名为北卡罗莱纳生物信息科学网格的项目,涉及 60 家企业、大学和生物医学研究公司。Microsoft 决定支持网格组织 Globus 项目。该组织的 Globus Toolkit 软件使企业可以建造和管理网格。目前 Globus 已经成为 Grid 技术的事实标准。

英国政府已决定投资 1 亿英镑,用来建设“英国国家网格(UK National Grid)”,这类研究的目标是将跨地域的多台高性能计算机、大型数据库、贵重科研设备、通信设备、可视化设备和各种传感器等整合成一个巨大的超级计算机系统,支持科学计算和科学研究。日本 NTT 宣布将于 2002 年中期开展为期 6 个月的网格计算试验,参与者包括了 Intel、SGI 等。试验将连接日本家庭、企业和学术机构的 100 万台 PC,预计试验的处理能力将达到每秒 65 万亿次浮点运算,是现有的最快超级计算机的 6 倍。

为了不再落后于世界,我国也在加强网格方面的研究。已有由中国科学院牵头的“国家高性能计算环境(HNPCE)”项目和由清华大学牵头的“先进计算基础设施北京上海试点工程”两个项目开展对网格计算的研究。HNPCE 使用 GridWare 系统软件进行全网格的统一的资源信息管理、用户管理、作业管理和安全、认证。使用 LDAP(Lightweight Directory Access Protocol)协议软件 OpenLDAP 作为网格资源信息管理的支撑系统。“试点工程”目前已经初步实现了基于“网页超级计算”而不是传统的“远程登录”的计算方式<sup>[3]</sup>。中科院计算所的“织女星网格”(Vega Grid)计划以元数据、构件框架、智能体、网格公共信息协议和网格计算协议为主要突破点对网格计算进行研究<sup>[4]</sup>。此外在国家科技部 2002 年“863”计划中也启动了一个专项开展网格技术研究和网格基础软硬件设施建设<sup>[1]</sup>。

### 3 相关技术与问题

网格技术的目标是基于因特网技术、Web 技术和高性能计算等技术,采用开放标准,实现网络虚拟环境上的资源共享和协同工作,消除信息孤岛和资源孤岛。与它相关的技术包括:实时企业信息系统、网络化虚拟设计环境、因特网技术、知识管理、XML 技术、ASP 技术、高性能计算技术等。也出现了很多与网格相关的技术名词:元计算(Metacomputing)、对等计算(Peer-to-Peer Computing)、分布计算(Distributed Computing)、计算网格(Computing Grid)、信息网格(Information Grid)、知识网格(Knowledge Grid)、数据网格(Data Grid)、访问网格(Access Grid)、万维网服务(Web Service)、语义网(Semantic Web)等。基于网格概念的共同技术问题是“如何动态调整资源共享和问题求解,实现多个研究机构的虚拟组织(一组被界定可以共享规则的个体和研究机构,我们称之为虚拟组织,即 VO——Virtual Organization)<sup>[5]</sup>”,网格所关心的问题主要不是文件交换,而是直接访问计算机、软件、数据和其它资源。它具有分布式系统和并行系统的特征,但是又同二者有着非常重要的区别<sup>[6]</sup>。与分布式系统类似,位于多个管理域下的超级计算机通过不可靠的网络进行连接,并且需要对广域分布的动态资源进行集成。但是网格系统对高性能的要求使其编程模型及接口与分布式系统有极大的差别。同时,网格系统作为并行系统还需要进行超级计算机之间的通信调度

以满足应用对性能的要求。然而,由于网格系统的异构性以及动态性使得现有的并行计算技术的应用受到很大的限制。网格系统中各类资源的共享都是在工业、科学和工程等领域的合作问题求解和资源需求突破情况下所需要的。这个共享是必须的、高可控制的、由资源的提供者和资源的用户清晰、仔细地界定什么是共享的、谁被允许共享以及共享发生的条件。由于分布式计算技术强调为用户提供一个逻辑或虚拟的集中环境,目前它还不能描述不同资源之间的互操作性、资源共享以及它们的需求。比如,目前的 Internet 技术描述在计算机中的通讯和信息交换,但是不能为计算提供在多个站点调整资源使用的一体化访问的途径。因此对于网格来说,它需要解决的问题是如何在动态、异构虚拟组织间实现协同的资源共享以及协同地解决某一问题?这一问题无论从范围、程度还是本质上都已经与 Web 所关心的互连问题有了明显的不同。

## 4 网格计算系统结构与演化

### 4.1 网格计算的系统结构

网格主要由网格结点、宽带骨干网、贵重仪器、可视化设备、网格软件和数据库六大部分组成。粗略地说,网格系统结构可以分为三个基本层次:资源层、中间件层/网格操作系统(Grid Operating System)和应用层。网格资源层:构成网格系统的硬件基础,它包括各种计算资源,如超级计算机、贵重仪器、可视化设备以及其它同构或异构、不同计算能力的计算资源结点。网格资源层仅仅实现了计算资源在物理上的连通,但从逻辑上看,这些资源仍然是孤立的,资源共享问题仍然没有得到解决。因此,必须在网格资源层的基础上通过网格中间件层来完成广域计算资源的有效共享。网格中间件层/网格操作系统:一系列工具和协议软件,其功能是屏蔽网格资源层中计算资源的分布、异构特性,向网格应用层提供透明、一致的使用接口,使用户感觉不到计算系统在地理位置上的不同、在能力上的差异。同时需要提供用户编程接口和相应的环境,以支持网格应用的开发。网格应用层:用户需求的具体体现。在网格操作系统的支持下,网格用户可以使用其提供的工具或环境开发各种应用系统。能否在网格系统上开发应用系统以解决各种大型计算问题是衡量网格系统优劣的关键。

由于现在的互联网结构并不是针对网格计算设计的,为了使网格计算和现有的结构兼容,一个可扩展的中间件层是必须的。上述以美国政府研发机构为主的推动网格计算的项目,都采用了一种网络协议“Globus”。Globus 对资源管理、安全、信息服务及数据管理等网格计算的关键技术进行研究,开发能在各种平台上运行的网格计算工具软件(Toolkit),帮助规划和组建大型的网格试验平台,开发适合大型网格系统运行的大型应用程序。Globus 认为<sup>[5]</sup>:在网络环境下的互操作,意味着需要开发一套通用协议,用它来描述信息的格式和信息交换的规则。Globus 的网格计算协议是建立在互联网协议之上的,以互联网协议中的通信、路由、名字解析等功能为基础。Globus 的协议分为 5 层(如图 1):构造层、连接层、资源层、汇集层和应用层<sup>[5]</sup>。上层协议可调下层协议的服务。网格内的全局应用都通过协议提供的服务来调用操作系统。图 1 左边是 Grid Architecture 的结构。右边是对应的目前的 Internet 的网络层次。该结构的设计采用了“时间瓶”模型(hour glass model)。在瓶子的中间细的部分对应于一些关键的和新的协议,如目前 Internet 中的 TCP 以及 HTTP 协议。上面和下面两个大的部分分别表示顶层在这些协议基础上所提供的

共同协议以及底层其他一些协议。首先是最底层的 Fabric,它向顶层提供了一个节点最基本的资源管理、任务创建等服务,如, fork, malloc<sup>[5]</sup>。就象是一个安装了 Globus 的节点,有节点本身的操作系统所提供的服务或者说是协议及对应于这一层。接下来的一层是连接层,它在 Fabric 的基础上定义了一些核心的通信以及安全认证方面的协议,这有对应于 Globus 的 GSI(Grid Security Infrastructure)模块。资源层则定义了这样一些协议,它负责描述本地的资源管理方案。主要是向上一层协同的资源分配提供信息。例如它提供了资源信息协议,负责提供本地的资源使用状况。资源管理协议,负责进行有条件的资源分配。在 Globus 中 GRIP(Grid Resource Information Protocol)、GRAM(Grid Resource Access and Management)以及 GridFTP 都属于这一层协议。Collective 是在时间瓶模型核心部分的最上一层,在它之上就是应用层,对应于具体的网格应用。它所提供的最主要的就是系统的资源管理协议,这对一个具体的请求如何在多个分散的资源中进行选择,与每个资源自己本身的 Resource 层协议进行合作,完成系统的资源分配。例如, Globus 中的 MDS(Monitoring and Diagnostics Services)、DRS(Data Replication Services)、DUROC(Dynamically Updated Request Online Coallocator)都是这样一些协议。最上层的应用层提供的就是一些编程语言以及问题解决环境。不难发现的一点就是,每一层都对应了 Globus 的一个具体模块。可以认为所谓 Grid Architecture 就是在现有的 Globus 基础上提炼得到的抽象概念。Foster 等人<sup>[5]</sup>分析现状认为 Globus 已经成为网格计算中一个事实上的标准,但仍然不完善。这就是后来的 OGSA 的概念内容。

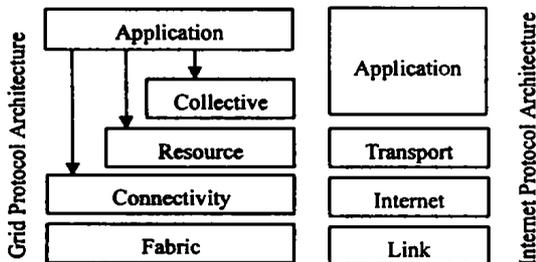


图 1 网格层状体系结构与 Internet 协议结构的关系

在网格系统中,资源的定位、管理和控制以及系统安全等方面面临着巨大的挑战。而资源的管理(RMS)是实现真正意义上的网格计算的关键。RMS 抽象的结构图如图 2 所示。

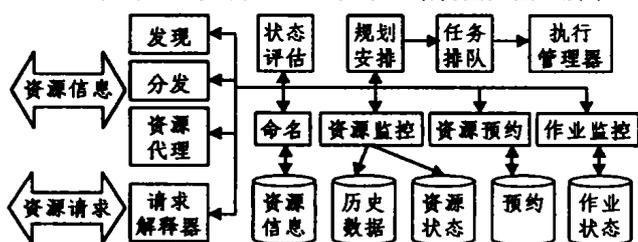


图 2 资源管理系统抽象的结构图

系统安全也是网格系统的重要内容。在已经实现的 Globus 系统中安全采用 x.509 认证系统。在系统安全需求方面,网格计算系统的安全需要做到:(1)单一登录;(2)授权保护;(3)与本地的安全方案协同工作;(4)统一的授权/证书发放机制;(5)提供安全的组通信;(6)支持多种实现方案。

4.2 网络技术的演化

网格技术的思想来源于电力网络。从 Globus 的整个发展来看,可以看到网络的演化历程。网格最开始的目的是为了提供广域范围的高性能科学计算,或者说是为了实现计算网格(Computing Grid);中期,目标转向了数据网格(Data Grid)这个方向,如 Globus 2.0 中的 GridFtp 等都是在这个时候提出的;现在,在 IBM 加入后,又提出了这个 OGSA 标准——Web 服务网格,很明显的标志就是,它又将目标转向了对商业服务的支持。

4.2.1 计算网格 其技术可以追溯到早期的分布式并行处理实验,在 NASA 寻找外部文明 SETI@home 项目中,采用了分布在 224 个地区的约 160 万个参与的计算机,获得了平均每秒钟 10 万亿次操作运算,对项目贡献了相当于 16500 年的计算工作量,用于搜索射电天文望远镜信号中的外星文明迹象<sup>[17]</sup>。网格的目标也从最初的轻量级网格(Lightweight Grids)发展到重量级网格(heavyweight Grids),使网格具有更大的计算能力。由此可见,早期的网格是纯粹为完成复杂计算而产生的分布式计算环境,是面向大规模计算应用需求的。面向的应用主要有:分布式超级计算、高通量计算、即时需求、高强度数据计算以及协同计算等<sup>[18]</sup>。目前大多数为科学计算实验而设计的网格系统如北欧国家的 NorduGrid<sup>[19]</sup>等都是 Globus 基础上实现的网格基础设施。这些网格的目标是为了更广域范围的计算和数据处理。因此,科学计算网格就是提供一种在异构节点上发现资源、协调资源、控制资源、通信控制和分布式协同计算的一种机制。这种机制过去和现在都是计算网格设计开发的基础。从系统结构上来看,计算网络的组成成分如图 3 所示。

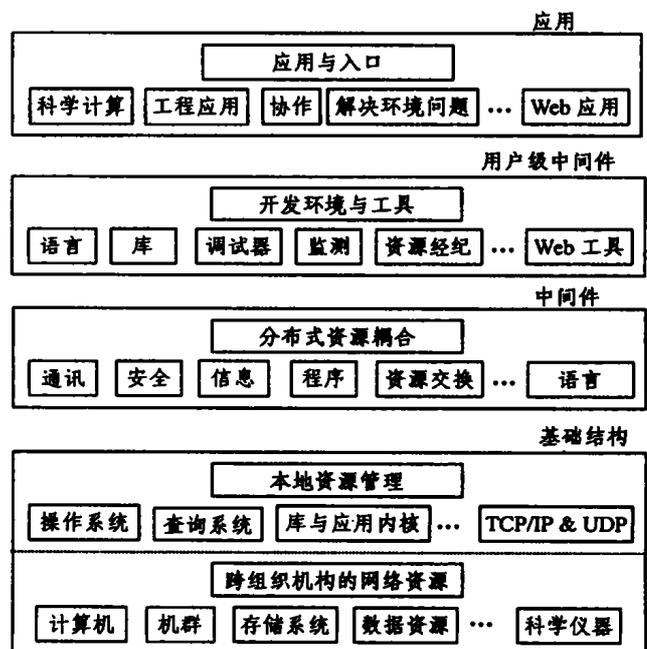


图 3 网络组成成分

4.2.2 数据网格<sup>[20]</sup> 欧洲十分看重他们的数据网格项目,因为这个潜力无限的项目可以为全世界各地的科学家提供轻松获取数据的捷径,同时这些科学家所能够接触到的数据资源也是空前丰富的。欧洲人认为这将会开创一个科学协同研究的新时代。目标就是如何通过网格技术来实现在大范围内,按照空间研究的规律来管理、访问、使用和分配海量的数据。数据网格提供了对跨网格环境的存储资源和数据进行管理的可扩展分布的基础设施。服务被设计支持从高能物理

到基因组计算的各种科学应用,而这些应用需要大量地(从TB级甚至更大)满足各种QoS请求的数据。通过设置一组核心服务如数据传输、安全和目录复制等,可以构造更高级的服务。目前主要通过复制目录信息和用户参数选择来引导从可

选择的存储复制目录中实现高级的复制选择服务。Globus的数据网格体系结构努力提供对分布在网格上的多种存储系统的标准访问。数据网格的体系结构如图4所示<sup>[20]</sup>。

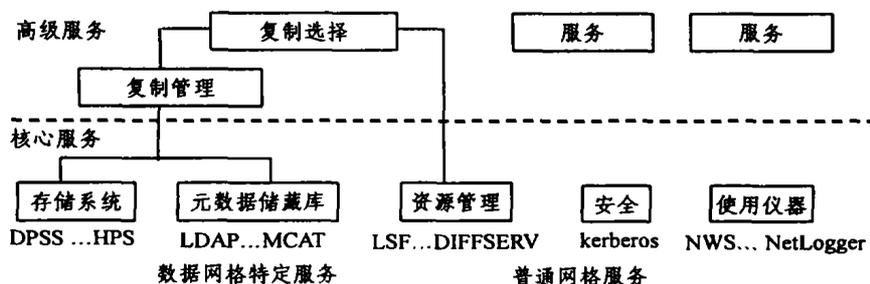


图4 数据网格体系结构

从图中可见,数据网格结构分成两层,即核心服务层和由核心服务层构造的高级服务层。核心数据网格服务寻求存在在网络环境的多种存储系统的抽象,以便为更高级的服务和应用形成统一访问存储系统的形式。这些核心服务包括下面的内容:

- 存储系统和数据访问:这个服务为访问和管理位于存储系统的数据提供基本的机制。这些机制为统一的创建、删除、访问和修改跨存储系统的文件实例提供抽象,而不管它们在什么物理位置。它们瞄准次一级的和文档型的存储系统如UNIX文件系统、高性能存储系统(HPSS)和能够用来访问更复杂的系统如存储请求经纪系统(SRB-storage request broker)。

- 元数据访问:这个服务提供了访问和管理在存储系统的数据信息,各种元数据由这个服务提供。包括:①应用元数据描述文件的内容、数据收集的环境和应用的各种细节。②复制元数据描述文件实例到特定复制位置的映像。③系统配置元数据描述存储系统的容量、性能,因此提供了关于数据网格构造的信息。

元数据服务更进一步提供了各种元数据得以被发布和被访问的统一的方法。以上这些服务自身也是建立在由Globus工具库提供的基本安全和信息服务的基础上的。核心数据网格服务能够用来构建更高级的服务(如图4)。例如:

- 复制管理:这是一个创建和删除在存储系统结点的复制的过程。通常这些复制是原始文件的精确拷贝,创建仅仅是用来获取某种性能益处。一个复制管理器特定地维护包含复制站点地址和文件实例的一个复制目录。

- 复制选择:这是基于一些应用特征从那些分布在网络环境的复制中选择一个复制的过程。一个共同的选择标准将加速访问的速度。

4.2.3 开放式网格服务体系OGSA<sup>[5,11]</sup>——Web服务网格 Foster等人在文[5]中提出了Grid体系结构的概念,在文[11]中提出了OGSA(Open Grid Services Architecture)的概念。文[11]是对文[5]的补充。OGSA是一个新架构。OGSA的目的就是要将Grid一些功能,更确切地说是Globus的一些功能融合到Web Service这个框架中,可见它的主要来源包括了Globus和Web Service两部分。与前期网格不同的是,OGSA是面向服务的架构,服务是第一位的。在这个架构中提出了对VO所支持的服务提供创建、维护的方法。而以前的架构是面向协议的,协议是第一位,服务是第二位的。并

且网格服务是定义在Web服务的基础上的,前者是对Web服务的定义。在OGSA中把所有的一切都定义为服务,计算资源、存储资源、网络、程序、数据等都是服务。所有的服务都联系对应的接口。Foster等已给出了一个OGSA的模型<sup>[11]</sup>。在这个系统中所有的一切服务包括永久性的服务以及临时性的服务。其中永久性的服务是少数的,大多数都是临时性的服务,而OGSA所要管理的也正是这些临时性的服务。所有的服务都必须遵循一定的规范,提供基本的接口。对网格服务这个接口所有的服务定义都要实现。它定义了一些最基本的功能,例如服务信息的查找等,还有一些通用的功能包括服务的创建、生存期的管理、发现、认证通知、升级等一些功能。

OGSA中大量引用了IBM所倡导的Web服务的概念。采用Globus是其提供的一些具体功能,如对其创建服务实例的管理以及一些安全方面的保证。Foster等认为Globus中最能体现网格服务概念的就是GRAM(Grid Resource Access and Management)、MDS(Directory Service)、GSI(Grid Security Infrastructure)这些部分,Globus所要做的下一步工作就是将这些模块融合到网格服务这个框架中来。最后,所有服务都要落实到一些具体的host宿主系统上,如J2EE、.NET、C等。

在OGSA所提到的网格服务中每一个服务都应当支持以下一些接口。主要有:创建(Factory接口)、全局命名(GSH接口)和参照(GSR接口)、生命期管理、注册(Registration)和发现(discovery接口)、授权(Authorization接口)、通知(Notification接口)等。网格环境中一个任务的执行过程是这样的:首先它向特定的注册中心使用Discovery接口找到它所要使用的服务,然后向这个服务的Factory接口发出一个请求,Factory在接受请求后,负责创建它所支持的服务的一个实例,这个事例是一个临时性的服务。实例创建后向客户返回的是一个唯一标识该实例的GSH(Grid Service Handle),为了实现服务与支撑平台的不相关性,GSH中并不包含具体的系统信息,用户为了与该服务进行通信,必须首先通过Mapper接口,获得该GSH所对应的GSR(Grid Service Reference)。在整个服务的生命周期中GSH是唯一的,而GSR是可变的,也就是说,在整个应用处理过程中,可以更换GSR,以达到使用不同的平台绑定。实例创建后在执行过程中,会与客户通过通知(Notification)接口进行交流,接受生命期(Lifetime)的管理。Factory接口主要是为了创建一个具体的服务实例而设立的。前面已经介绍过,一个实例创建后它返回一个全局唯一的

GSH。通过 Mapper 可获得对应的 GSR。Factory 接口如何创建一个实例是由其所依赖的平台具体决定的,如后台是一个 J2EE 或 .NET 那么它们将有自己的创建方法。这一点对发出请求的客户来说是不可见的。同样一个 Factory,可能会调用多个低一层的 Factory 接口,这样就可以转发请求,由低一层的 Factory 具体地创建实例。Mapper 主要是实现 GSH 到 GSR 的一个对应,其中,GSH 在整个应用执行过程中是唯一的,但是由于它不包含具体的平台信息,例如所使用的底层协议,网络的地址等等,客户并不能直接通过 GSH 与其所请求的服务实力进行通信。这时就需要一个 GSR 来完成这一工作,GSR 是一个 WSDL(Web Services Description Language)文档,其中定义了该实例当前时刻所使用的具体的平台相关信息。同时,它也是可以过期的。如对一个网络协议的替换,将导致使用相关协议的 GSR 的过期。GSR 过期后,通过 GSH 可以得到一个新的 GSR,这样就实现了对多种底层平台的绑定。生命期的管理主要是在一个实例创建之后,负责它什么时候结束。一个网络服务的实例在它被创建的时候,客户的请求就和 Factory 接口协商该实例的初始化执行时间,也就是先大致确定一个执行时间。具体创建后,则通过 Soft State 来对该实例进行管理,例如可以设置一个新的结束时间。这样就不会产生一个进程由于没有接收到相关数据而一直在循环等待导致资源的溢出的情况。Registry 就是提供了对服务的一个注册管理。一个服务可以将自己的 GSH 注册在这里,而后客户可以通过这个接口获得与这个服务相关的具体数据。Discovery 是一个应用实例的第一个步骤,通过一个待具体参数的发现请求,获得所需要的特定的服务。通知(Notification)接口有两种功能,简单地说,一个实例通过这个接口可以将它的一些状态信息发出去;同时它也可以通过这个接口获得一些控制信息。在提出了 OGSA 之后 Globus 的结构变化如图 5 所示。左边的是原来的层次性结构对应于前面所提到的网格体系结构。右边则是一种新的结构,首先通过办一些基本的功能抽象为服务,然后在这些服务的基础上构筑更高层的服务。可见左边的是面向协议的,而右边的则是面向服务的一种结构。这里要注意的一点是,它将 GSI 这种安全服务放在最底层,把它剥离于 OGSA 的框架是为了简化 OGSA 的结构,让这些细节由底层的系统负责。

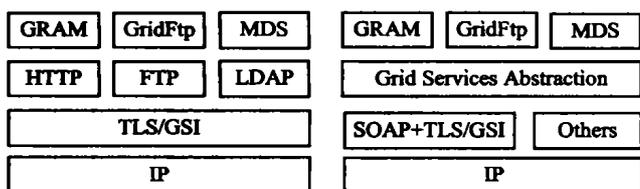


图 5 OGSA 之后 Globus 的结构变化图

## 5 网格基本任务与关键技术

### 5.1 网格计算基本任务<sup>[5]</sup>

在网格计算中,首先要查清网格内所有可用的资源,比如哪些主机可以访问、还空闲多少处理能力、数据库里可供使用的数据是什么、共享的应用程序是否已经准备好、共享主机采用什么文件系统等。用户提交的任务要由系统来分配资源并控制其运行,包括要将其分配到哪些主机上运行、调用哪些数据、启动何种应用程序、何时开始运行等。这样,网格计算至少需要具备三种基本功能(或称网格计算的三要素):任务管理、

任务调度和资源管理。当然在此基础上网格计算要承担的其他任务还有许多。

- 网格资源的管理:确定并监测网格资源状况,收集任务运行时的资源占用数据。

- 任务管理:用户通过该功能向网格提交任务,为任务指定所需要的资源、删除任务并监测任务的执行。

- 任务调度:用户提交的任务由该功能按照任务的类型、所需资源、可用资源等情况安排运行日程和策略。

### 5.2 网格技术关键内容

为解决不同领域复杂科学计算与海量信息服务问题,人们以网络互连为基础构造了不同的网格,如信息服务网格、天文网格、科学计算网格、生物信息网格和数据网格等。它们在体系结构、要解决的问题类型等方面不尽相同,但都需要共同的关键技术。网格的核心观念是一句老话,即“网络就是计算机”。从这个角度看,网格所涉及的技术核心包括以下方面<sup>[5]</sup>:  
 ①如果把整个因特网看成一台计算机,那么它的处理器、存储器、外部设备是什么?它应该采用什么样的体系结构?  
 ②什么是网格这台计算机的操作系统?它的进程和线程是什么?什么是它的地址空间?如何管理它的资源?  
 ③什么是网格的编程环境和使用环境?什么是网格的用户界面?什么是网格的程序设计语言?  
 ④什么是网格的应用?它们有什么样的模式和特征?  
 ⑤从用户的角度看,网格与当前的因特网/Web 有什么不同?它能提供什么样的独特好处?因此,从这 5 个方面来分析,具体就有以下的关键技术内容:

- 网格结点建设:“格点”是网格计算资源的提供者,它包括高端服务器、集群系统、MPP 系统、大型存储设备、数据库等。这些资源在地理位置上是分布的,系统具有异构特性。

- 宽带网络系统建立:在网格计算环境中,提供高性能通信的必要手段。通信能力的好坏对网格计算提供的性能影响甚大,要做到计算能力“即连即用”必须要高质量的宽带网络系统支持。用户要获得网格延迟小、可靠的通信服务也离不开高速的网络。

- 资源管理和任务调度工具设计与开发:计算资源管理工具要解决资源的描述、组织和管理等关键问题。任务调度工具其作用是根据当前系统的负载情况,对系统内的任务进行动态调度,提高系统的运行效率。它们属于网格计算的中间件,要进行研发,支持网格运行。

- 监测工具设计与开发:高性能计算系统的峰值速度可达百万亿次/秒,但是实际的运算速度往往与峰值速度有很大的距离,其主要原因在于高性能并行计算机的并行程序与传统的串行程序有很大差异。为了充分利用各种计算资源,如何帮助使用人员充分利用网格计算中的资源,这就要靠性能分析和监测工具。这对监视系统资源和运行情况十分重要。

- 应用层的可视化工具:网格计算的主要领域是科学计算,往往伴随着海量的数据,面对浩如烟海的数据想通过人工分析得出正确的判断十分困难。如果把计算结果转换成直观的图形信息,就能帮助研究人员摆脱理解数据的困难,这就要研究能在网格计算中传输和读取的可视化工具。并提供友好的用户界面。

## 6 网格技术在我国的应用方向

我国的许多行业,如能源、交通、气象、水利、农林、教育、

(下转第 9 页)

势:它是自描述的,使用 Unicode 编码方式,简单通用,可以描述树型或图型结构的数据,适合于对半结构化数据的管理。这些特点不论是 DBMS 还是 OODB 都是无法取代的。另外,XML 数据库的另一优势还在于数据的移植能力,以及对现有的异构数据库系统的集成能力。NXD 适合管理复杂数据结构的数据集,如果已经以 XML 格式存储信息,则采用 NXD 利于文档的存取和检索,能够提供高质量的全文搜索引擎,特别适合对半结构化数据的管理。

当然,传统的关系型和对象关系数据库也有自己的应用领域,如果要对结构化的数据进行管理,传统的关系数据库还是首选。

对 XML 数据库研究的重点应当放在 NXD 上面,现阶段,NXD 的数据更新以及数据查询、修改能力需要进一步提高,在并行性方面,NXD 还缺乏有效的管理机制,缺乏多重管理能力,协同工作能力、规划能力不强。这些不足之处正是 XML 数据库近期和未来的研究热点所在。

### 参考文献

- 1 Seligman L, Rosenthal A. XML's Impact on Databases and Data Sharing. IEEE Computer, June 2001. 59~66
- 2 XML: DB, XML: DB Initiative for XML Databases. <http://www.xmldb.org/index.html>

- 3 Bourre R. XML Database Products, 14 Aug. 2002. <http://www.rpbouret.com/xml/XMLDatabaseProds.htm>
- 4 Ipedo XML Database. <http://www.ipedo.com/html/products-xml-dat.html>
- 5 Obasanjo D. AN EXPLORATION OF XML IN DATABASE MANAGEMENT SYSTEMS. <http://www.25hoursaday.com/StoringAndQueryingXML.html>
- 6 Clark, James DeRose S. XML Path Language (XPath) 2.0, W3C Working Draft 30 April 2002. <http://www.w3.org/TR/xpath20/>
- 7 Bourett, Ronald. XML and Databases. June 2002. <http://www.rpbouret.com/xml/XMLAndDatabases.htm>
- 8 Chamberlin Don, et al. XQuery 1.0: An XML Query Language (Working Draft). June 2001. <http://www.w3.org/TR/2001/WD-xquery-20010607>
- 9 Cheng, Josephine, Xu, Jane. IBM DB2 Extender. ICDE '00 Conference, San Diego. Feb. 2000
- 10 W3C Working Draft. XML Query Use Cases, 30. April, 2002. <http://www.w3.org/XML/Query>
- 11 Tatarinov I, Ives Z G, Halevt A Y. Updating XML. SIGMOD, 2001
- 12 Jaenicke C. Native XML Databases and Relational Database. <http://www.exln.com/products/whitepapers>
- 13 University of Washington Database Group. Management of XML Data. <http://www.data.cs.washington.edu/xml>
- 14 W3C Technical Reports and Publications. <http://www.w3.org/tr/>
- 15 Airi S, Frank W T. System Desiderata for XML Databases. In: Proc. of the 27th VLDB Conf., 2001

(上接第5页)

环保等对高性能计算网格即信息网格的需求将非常巨大。目前来看,网格计算在我国主要可以提供以下服务:

•资源数据:国家在过去几十年已经花费了大量的财力收集各种地下、地面、大气的资源数据,采用的手段包括地下勘探、地面人工测绘和监测、航空遥感、卫星遥感等。但由于目前这些数据零碎地分散在不同的地方,共享困难,利用率低,将它们放在网格上共享将带来很多好处。

•高性能计算:计算网格的建立有利于各计算中心实现资源共享,充分利用硬件和软件资源,节约成本。它能在基础研究、汽车、大型水电工程、石油勘探、气象气候、航空、交通、金融、医疗等领域发挥空前的作用。

•生物信息:生物科学的信息化和全球化已成为大势所趋,我国的生命科学界对网格也有强烈呼声。以北京大学生命科学院为龙头的一些生物科研机构已经加入了相关的国际组织,并在网格上分享现代生物信息资源以提高工作效率。

•环境保护:我国已经建立了若干环境数据收集网和计算机网,但由于没有实现统一的网上共享和管理,使用极不方便。网格建立后,可以解决这一问题,并且可以提供新闻媒体的缓存服务、目录服务、数据挖掘和分类服务等多种新的应用形式。

### 参考文献

- 1 <http://www.863.org.cn/863-105/active/200205280194.html>
- 2 肖连兵, 黄林鹏. 网格计算综述. 计算机工程, 2002, 28(3): 1~3
- 3 Foster I, Kesselman C, et al. The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, San Francisco, 1999
- 4 <http://www.yesky.com/20020702/1618643.shtml>
- 5 Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. Int'l J. High-Performance

- Computing Applications, 2001, 15 (3): 200 ~ 222. <http://www.globus.org/research/papers/anatomy.pdf> (current June 2002)
- 6 <http://www.st121.com.cn/text/d/210.htm>
- 7 Foster I, Kesselman C, Nick J M, Tuecke S. Grid Services for Distributed System Integration. 15. <http://www.globus.org/research/paper>
- 8 Baker A, Buyya R, Laforenza D. The Grid: A survey on global effort in grid computing. <http://www.byte.com>
- 9 都志辉, 李三立, 等. 网格计算及其原型实现研究. 计算机科学, 2002, 29(8)
- 10 <http://www.computer.org/proceedings/hpdc/0287/02870034a-bs.htm> (current June 2002)
- 11 Foster I, et al. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration; [Tech. report]. Glous Project. <http://www.globus.org/research/papers/ogsa.pdf> (current June 2002)
- 12 Christensen E, et al. Web Services Description Language (WSDL) 1.1. W3C Note, 15 Mar. 2001. <http://www.w3.org/TR/wsd1> (current June 2002)
- 13 Catlett C, Smarr L. Metacomputing. Comm. ACM, June 1992. 44~52
- 14 Paton N W, et al. Database Access and Integration Services on the Grid. [Tech. report UKeS-2002-3]. National e-Science Centre. <http://www.nesc.ac.uk,2002>
- 15 <http://www.globus.org>
- 16 <http://www.gridforum.org>
- 17 Decusatis C. Grid computing: the next (really, really) big thing. <http://www.byte.com>
- 18 Foster I, Kesselman C. Computational Grid. The Grid: Blueprint for a future computing infrastructure. Morgan Kaufmann Publishers, 1998
- 19 Waananen A, Ellert M, Konstantinov A, et al. An overview of grid architecture for scientific computing. <http://arxiv.org/PS-cache/cs/pdf/0205/0205021.pdf>
- 20 Vazhkudai S, Tuecke S, Foster I. Replica Selection in the Globus Data Grid. <http://cn.arxiv.org/abs/cs.DC/0104002>