

网格体系结构详解^{*}

高全泉

(中国科学院数学研究所 北京 100080)

Detailed Description of the Grid Architecture

GAO Quan-Quan

(Institute of Mathematics, Chinese Academy of Sciences, Beijing 100080)

Abstract Grid Technologies proposed by Globus project has been used to establish, manage, and exploit cross-organizational virtual organizations sharing relationships. Grid technologies are structured by the Grid architecture. The Grid architecture is an extensible and open architectural structure which identifies fundamental system components, specifies the purpose and functions of these components, and indicates how these components interact with one another. In this paper, The Globus project and leaders of it are introduced briefly, then description of various layers of the Grid architecture including their purposes and functions are presented in detail, and Grid protocols & services (and APIs) defined in Globus Toolkits and Grid architecture in practice are also discussed.

Keywords Grid computing, Grid architecture, Resources sharing, Virtual organizations, Super computing

1. 引言

在“网格:面向虚拟组织的资源共享技术”一文中,我们对网格的研究领域及基本理念作了介绍,并介绍了网格体系结构的设计思想和体系结构框架。本文我们具体描述由 Globus 项目提出的网格体系结构——一种类似于 Internet 协议体系结构的层次式协议体系结构,并映射到 Internet 协议体系结构上。我们主要介绍构成网格体系结构的基本系统组件,详细规定/说明这些组件的目的与功能,并指出这些组件如何与别的组件交互。

在网格研究领域, Globus 项目可谓网格核心项目,它是产生网格计算的温床,许多关键理念和技术是在该项目的研究中产生的;网格的核心技术集中于网格体系结构,它是实施 VO(虚拟组织)资源共享的基本框架体系; Ian Foster 和 Carl Kesselman 是 Globus 项目领导者,是网格研发的焦点人物,称其为网格计算先行者并不过分。在给出网格体系结构的描述之前,先对相关背景 Globus 项目及其领导者作一简介。

2. Globus 计划及其领导者简介

Globus 是美国能源部 Argonne 国家实验室于 1995 年启动的高性能计算研究计划^[1,2], 美国的 12 所大学和研究机构参与了该项目。Globus 研究不仅着眼于构建计算网格基础结构相关的问题,而且关注在设计和开发使用网格服务的应用中出现的问题。有关“网格”的最早出处,据说是源于专业术语电力供应网“Power Grid”^[10]。“Power Grid”的原意是电力供应商根据用户的需要供应电力,消费者只需支付自己所用的那部分电费,不在意电力源于哪个发电厂。类似的情形用于 IT 领域更适合,即在电脑的处理性能(Computing),如硬件、软件、信息数据库等资源上提供共享能力,根据用户的需求来通过网络提供所需的资源。Globus 的目标是理解一个可用网格的应用需求,并开发满足这些需求所要求的本质技术。此目

标包含以下三个广泛的活动:

- 开发计算性网格所要求的基本技术和工具
- 使用已经开发出的基本技术和工具构建大规模的原型计算性网格(即测试床),并且
- 在原型网格上执行现实的应用以便评价网格概念及技术的实用性

在这三个方面,都已取得令人满意的进展。当前正在开展的研究内容包括

◇资源管理 开发在分布式系统中的有关命名、定位及分配计算和通信资源的一致和可伸缩的机制。

◇数据管理与访问 已经开始实施集体行动,设计并产生关于数据管理的基础结构级体系结构,即所谓的数据网格。

◇应用开发环境 正在将基本的网格服务集成到已有的应用开发框架、环境和语言之中(如, CORBA, Java, Perl, Python)。

◇信息服务 已在网格信息服务的设计及原型方面工作多时,将使得更能适应动态应用配置和适配。

◇安全性 正在开发安全组织通信的安全性算法,信任关系的管理及粒度良好的访问控制的新机制。

Globus 计划中,网格相关软件的开发和制定是重要的活动。Globus 元计算 Toolkit 就是根据网格思想研发的可在各种平台上运行的网格计算工具软件。它定义为构建一个计算网格而需求的基本服务和能力。由于 Toolkit 是 Globus 成果的重要体现,所以我们在第 4 节里将介绍其主要的构成和功能。这里,给出 Globus Toolkit 设计所遵循的原则:

- ✓工具包由组件集构成,这些组件实现安全性、资源位置、资源管理和通信等方面的基本的服务;
- ✓工具包区别本地服务和全局服务,前者被简单地保留以使部署灵活,后者建立在本地服务的顶部且可以更复杂;
- ✓定义了接口以便管理而不是隐藏异质性;
- ✓信息服务是工具包的一个完整的组件

^{*}国家自然科学基金项目,批准号 60072006。高全泉 研究员,研究方向为人工智能、知识工程、软件体系结构、并行/分布计算(中间件)技术等。

√无论对接口还是实现,工具包尽可能使用标准。

Globus 项目的领导者是 Ian Forster 博士和 Carl Kesselman 博士。Ian Foster 在英属哥伦比亚大学获计算机科学博士,同时担任 Argonne 国家实验室科学家和芝加哥大学计算机系合作教授之职。发表专著 3 本,论文及报告 100 多篇,内容涉及并行/分布计算及计算性科学的诸多方面。在 1995 年,领导了关于 I-WAY 网络化试验的软件基础结构的开发。Carl Kesselman 在美国洛杉矶的加州大学获计算机科学博士,担任信息科学研究所的项目领导和南加州大学计算机科学的研究合作教授。研究兴趣涉及高性能分布式计算,并行计算以及并行程序设计语言,与 Ian Foster 共同领导了 Globus 计划。许多关于网格的技术论文都与他们相关。

3. 一般化的网格体系结构描述

我们首先描述在 Globus 中提出的具有一般性的网格体系结构^[3]。由于网格体系结构集中体现网格技术的本质功能与特性,并且网格技术有很高的专业理论 & 技术含量和一定的复杂性,为了体现 Ian Foster 等人的真实意图和易于理解,掌握其精髓,我们将尽可能做到真实再现(因为他们的表述与内容组织极为出色)。

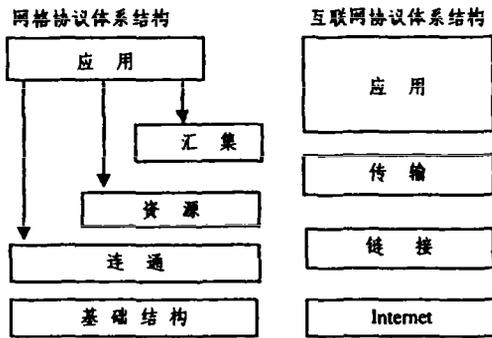


图 1 层次式的网格体系结构以及 Internet 协议体系结构的关系

由于 Internet 协议体系结构从网络扩展到应用,因此从网格层到 Internet 层存在一个映射。对网格体系结构的描述,并不是罗列所有要求的协议(及服务、API 和 SDK),而是标识一般的组件类需求,给出一个可在其中安排关键 VO 需求的解决方案的可扩展的、开放的体系结构。这样的网格体系结构如图 1 所示。这一体系结构及后文的讨论将这些组件组织成层次式的。在每一层中的组件共享共同的特征,并能够建立在由任何低层提供的能力和行为之上。

在规定网格体系结构的各种层次时,遵循的模型是“沙漏(或水漏)”模型^[4]。沙漏的瓶颈定义核心抽象和协议的一个小集合(如在 Internet 中的 TCP 和 HTTP),许多不同的高层行为映射到它们的上面(沙漏的顶部),它们自身也能被映射到不同的基本技术之上(沙漏的底部)。既为瓶颈,则此处定义的协议必须是较小的。在由 Ian Foster 等人提出的体系结构中,沙漏的瓶颈由资源和汇集协议组成,它们简化个别的资源的共享。在这些层设计了协议,因而这些协议能在那些在基础结构层定义的完全不同的资源类型范围之上被实现,并且本身又能被用于构建汇集层上的各种全局服务和应用专门的行为。因在单一资源之上涉及多种资源的协调(“汇集”)使用,我们称之为“汇集”层。

这是一种高级的体系结构,且在设计和实现上很少加以限制。实际的结果是在 Globus Toolkit 中定义的协议集。关于 Globus Toolkit 的协议及服务,在第 4 节里介绍。以下分别描述各层的目的与功能,以及与别的层的交互形式。

3.1 基础结构(Fabric):到本地的接口

网格基础结构提供由网格协议策划的共享访问的资源。基础结构提供的基本资源可以是计算性资源,存储系统,网络资源及传感器等。“资源”可以是逻辑的,如一个分布式文件系统,计算机簇,或分布式计算机池。对此,一资源实现可以涉及内部的协议(如 NFS 存储访问协议或簇资源管理系统的进程管理协议),但这些并非网格体系结构所关心的问题。

基础结构组件实现本地的、资源专门的操作,这些操作作为上层的共享操作的结果出现在专门的物理/逻辑资源上。因此,在基础结构层实现的功能是一方面,它们所支持的共享操作是另一方面,这二者之间存在着紧密和精确的相互依赖关系。基础结构的功能性愈丰富,共享操作就愈能精细;与此同时,如果在基础结构元素上安排少量的要求,则能够简化网格基础结构的部署。例如,资源层支持预先保留使得能以有趣的、别的方式不可能完成的方式为上层服务聚集资源(实际中很少支持,预先保留的需求会增加将新资源合并到网格中的代价)。

由这些网络技术提供的重要新能力的问题/意义体现在,通过非常及时的聚集(共同安排和共同管理),构建大规模的、集成式系统。经验表明,对资源的最起码的要求包括两方面:一是应该实现准许发现它们的结构、状态和能力(如是否支持预先保留)的询问机制,二是应该实现资源管理机制,该机制提供交付的服务质量的某些控制。

一般来说,资源的类型不同,则它们的能力特征也不相同。一些资源专门的能力特征如表 1 所示。

表 1 部分资源专门的能力的特征

资源种类	资源专门的能力特征
计算资源	要求启动程序的机制及监控、控制有效进程的执行的机制。允许控制分配给这些进程的资源的的管理机制,预先保留机制也是有用的。需要能确定硬件、软件特征以及相关状态信息(如在程序机管理的资源情况下,当前负载和队列状态)的询问功能。
存储资源	要求存放和获取文件的机制。第三方和高性能(如剥皮的)传输是有用的。要求读写文件的子集与/或执行远程数据选择或分解的机制也是重要的。允许控制分配给数据传输的资源(空间、磁盘带宽、网络带宽、CPU)的管理机制是有用的,预先保留亦然。需要询问功能,以便确定硬件、软件特征以及如可用的空间和带宽利用的相关的负载信息。
网络资源	对分配给网络传输的资源提供控制(如优先权,保留)的管理机制可能是有用的。提供询问功能以便确定网络特征及负载。
代码仓库	这种专门化的存储资源格式要求管理版本化的源/目标代码,如 CVS 控制系统。
目录	这种专门化的存储资源格式要求实现目录查询和修改操作的机制。如关系数据库。

3.2 连通:顺利并安全地通信

连通层定义网格专门的网络事务所需要的核心通信和认

可协议。通信协议使得能在基础结构层资源之间交换数据。认可协议建立在通信服务之上,以便隐形地提供关于验证用户

和资源的标识的安全机制。因涉及连接与通信两层含义,笔者认为 Connectivity 在这里作连通较通常的“连接”更贴切。

通信需求包括传输、路由和命名。尽管存在别的选择,但我们这里假定,从 TCP/IP 协议栈提取这些协议,特别地,它们出自 Internet 层次式协议体系结构的 Internet (IP 和 ICMP), transport (TCP, UDP) 以及应用 (DNS, OSPF, RSVP

等)。但这并不意味着将来网格通信不要求考虑特别的网络动力学类型的协议。在连通层的安全性方面,应该注意到,安全性问题的复杂性使得只要有可能时,任何解决方案均应建立在已有的标准之上,这点尤为重要。正如通信的情形,许多在 Internet 协议套件中开发的安全性标准是可用的。

VO 环境的认可解决方案的特征如表 2 所示。

表 2 VO 环境的认可解决方案的特征

认可解决方案	基本特征
Single sign on (惟一签字)	用户必须能够“log on”(认可)一次,然后就可以访问在基础结构层定义的多个网格资源,不需要进一步的用户干预。
Delegation (委派)	用户必须能够授予一个程序代表该用户运行的能力,因而程序能够访问用户认可的资源。该程序(任选地)也能有条件地将其权力的子集委派到另一个程序上(有时亦称为受限委派)。
与各种本地安全性解决方案的集成	每个站点或资源提供者可以征用任何不同的本地安全性解决方案,包括 Kerberos 和 UNIX 安全性。网络安全解决方案必须能与这些不同的本地解决方案互操作。从现实来看,它们不可能大规模地替代本地安全性解决方案,而是必须允许映射到本地环境上。
基于用户的信任关系	为了用户一并使用多个提供者提供的资源,安全性系统必须不做如下要求即在配置安全性的环境中,每个资源提供者协作和相互交互。例如,若一用户有使用站点 A 和 B 的权利,则该用户能够一起使用站点 A 与 B,不必要要求 A 和 B 的安全性管理员的干预。

网络安全解决方案也对通信保护提供支持(比如,在保护程度上的控制,不可靠协议的独立数据单元保护,对 TCP 以外的可靠传输协议的支持),并使得能实施在认证决策上的程序机管理的控制,包括以不同的方式限制权力委派的能力。

3.3 资源:共享单个资源

资源层建立在连通层通信和认可协议之上,定义有关单个资源上的安全协商,初始,监控,控制,记账及共享操作的付费的协议(及 API 和 SDK)。这些协议的资源层实现调用基础结构层功能访问并控制本地的资源。资源层协议全面关注单个的资源,因此忽略跨分布的汇集的全局状态和原子动作问题,而此类问题是资源层的上层汇集层关心的问题。

资源层有以下两类基本的协议:

- 信息协议 被用于获得关于一个资源的结构和状态方面的信息。例如,一资源的配置,当前负载以及使用策略(如使用代价)。

- 管理协议 被用于协商访问共享的资源,例如,详细说明资源需求(包括预先保留和服务的质量)以及要执行的操作,例如进程建立或数据访问。由于管理协议对初始共享关系负责,所以它们必须起一个“策略应用点”的作用,确保请求的协议操作与资源被共享的策略相一致。还必须考虑包括记账和付费在内的问题。一个协议也可以支持对一个操作的状态的监控以及对该操作的控制(如,终结该操作)。

虽然能设想许多这样的协议,但由于资源(及汇集)协议层形成沙漏模型的瓶颈,遵循这种原理,资源层将被限制到小且集中的集合。必须选择这些协议,以便捕捉跨许多不同资源(如不同的本地资源管理系统)的共享的基本机制,同时并不过度地限制可以开发的高层协议的类型和性能。

在表 1 中给出的合乎需要的基础结构层功能性之表概括了在资源层协议中需求的主要特点。对该表,增加了关于许多操作的“恰好一次(exactly once)”语义的需要,以及在操作失败时的报错指示。

3.4 汇集:协调多个资源

尽管资源层只着眼于与单个资源的交互,但体系结构的下一层包含的协议(及 API 与 SDK)不与任何一个专门的资源关联,而实际上是全局的并且跨所有资源捕捉相互作用。为

此,我们将体系结构的下一层叫做汇集层。由于汇集组件建立在协议沙漏的狭窄的资源与连通层瓶颈之上,所以它们能够实现各种共享行为而不必在正在被共享的资源之上安排新的需求。例如:

- 目录服务允许 VO 伙伴发现 VO 资源的存在与/或性质。一个目录服务可以允许其用户通过名字与/或通过属性,如类型,可用性或负载来查询资源。资源级 GRRP 与 GRIP 协议用于构建目录。

- 合作分配,进度安排及代理服务允许 VO 参加者请求为一个专门的用途分配一个或多个资源以及在适当资源上为任务安排进度。例子包括 AppLes, Condor-G, Nimrod-G 以及 DEM 代理。

- 监控和诊断服务支持关于失败,对手攻击(“入侵检测”)及超载等方面的 VO 资源的监控。

- 数据备份服务支持 VO 存储(或许也有网络和计算)资源的管理,以使得相对于如响应时间,可靠性以及成本度量来说,数据访问性能极大化。

- Grid-enabled 程序设计系统使得能在网格环境中使用熟知的程序设计模式,能以此类模式使用各种网格服务解决资源发现,安全性,资源分配及其它关键问题。这样的例子有消息传递接口和 manager-worker 框架。

- 工作负载管理系统和集中框架——也就是问题求解环境“PSEs”,用于多步骤,异步的,多组件工作流的描述,使用及管理。

- 软件发现服务根据正在求解的问题的参数,发现并选择最佳的软件实现和执行平台。例子包括 NetSolve 和 Ninf。

- 团体认证服务器实施管理资源访问,生成团体成员能用于访问团体资源的能力的团体策略。通过建立在资源层中的资源信息和资源管理协议,以及连通层的安全性协议,这些服务器提供全局的策略实施。Akenti 解决其中的部分问题。

- 团体记账和付费服务,由团体成员收集用于记账,付费与/或资源用法限制的资源用法信息。

- 集中服务支持在可能出现在大的用户团体中的信息的协调交换,不管同步还是异步。例子是 CAVERNsoft, Access Grid 及日用品组织系统。

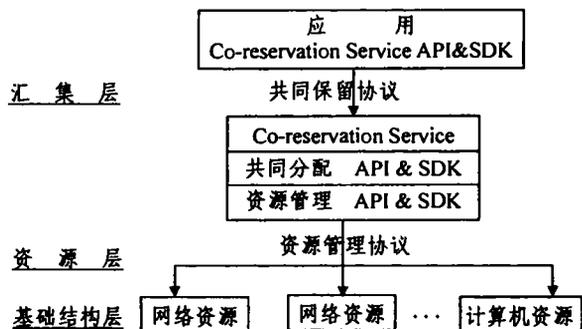


图 2 能以各种方式组合汇集和资源层协议, 服务, APIs 和 SDKs 向应用发送功能

这些例子给出了在实践中遇到的各种汇集层协议和服务。注意, 虽然资源层协议本质上必须是一般的并被广泛地部署, 但是汇集层从一般目的跨越到非常应用或领域专门, 这或许只有在专门的 VO 中才存在。

汇集功能被实现成与相关协议关联的持久性服务/实现成企图与应用链接到一起的 SDK (与 API 关联)。这两种情形下, 它们的实现能建立在资源层 (或其它汇集层) 协议和服务之上。例如, 图 2 显示使用资源层管理协议来操作基本资源的汇集共同分配 (Collective co-allocation) API 及 SDK (中间层)。在此之上, 可以定义一个共同保留服务协议并且实现讲此协议的共同保留服务, 调用共同分配 API 来实现共同分配操作并有可能提供附加的功能性, 比如认证, 容错及日志。应用则可以使用共同保留服务协议来请求端到端网络保留。

可以将汇集组件剪裁到专门的用户团体, VO 或应用领域的的需求。例如, 一个实现应用专门的连贯性协议的 SDK, 或关于网络资源专门集的共同保留服务。其它的汇集组件能更一般化, 例如管理多个团体的存储系统的国际性集中的备份服务, 或为了能够发现 VO 的目录服务。一般地, 目标用户团体越大, 汇集组件的协议和 API 以标准为基础就越重要。

3.5 应用 (Application)

网格体系结构的最终层次由在 VO 环境中操作的应用组成。网格体系结构的应用程序员视图如图 3 所示。应用由在任何层上定义的服务的形式以及对这些服务的调用形式组成。在每一层, 有界限分明的协议, 它们规定: 访问某些有用的服务、资源管理、数据访问、资源发现等等。在每一层中, 也可以定义实现与适当的服务交换协议消息来执行期望的动作用的 API。

API 由软件开发工具包 (套件) SDK 实现, 开发工具依次使用网格协议与向末端用户提供能力的网格服务交互。上层的 SDK 除了实现本地的功能外, 也提供虽未被映射到专门的协议上, 但可以将协议操作与对附加的 API 的调用结合的功能。图 3 中, 实线表示一个直接的调用, 虚线表示协议交互。

需要指出的是, 在图 3 中, 在一个单独层中标出的“应用”及给出的内容, 实际上可在精细的框架和库之上调用 (如, Common Component Architecture, SciRun, CORBA, Cactus, Workflow system), 并以更多内部结构为特点。如果在图中全部记录这些结构, 则那样的图要比现在的图超出许多。这些框架自己可以定义协议, 服务与/或 API (例如简单工作流访问协议)。然而, 这些问题已经超出了本文的范围。本文只着眼于解决网格中所要求的最基本的协议和服务。

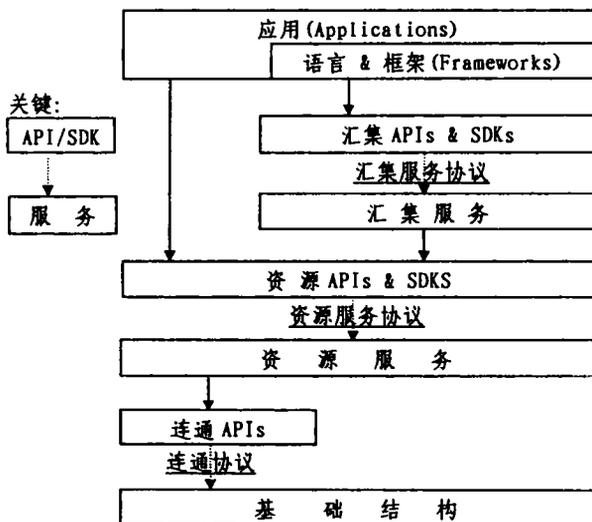


图 3

4. Globus Toolkit 定义的协议及服务

上节描述了 Globus 项目中提出的具有一般意义的网格体系结构。现在, 我们来关注 Globus 开发的能在各种平台上运行的一个实际网格计算工具软件 Globus Toolkit, 介绍其中定义的具体的协议和功能 (有关 Globus Toolkit 的详情, 见文 [2] 及 <http://www.globus.org> 之下的相关文档)。这些协议与功能已用在以下项目中: NSF 的国家技术网格 [5]; NASA 的信息动力网格 [6]; DOE 的 DISCOM [7]; GriPhyN (www.ppdg.net) [8]; NEESgrid (www.neesgrid.org) [9]; 粒子物理数据网格 (www.ppdg.net) [10]; 欧洲数据网格 (www.en-data-grid.org) [11]。

下面分别介绍 Globus Toolkit 中定义的不同层次上的协议及服务等:

- Globus Toolkit 具有使用 (基本地) 已有的基础结构组件的能力, 这些组件包括厂商提供的协议和接口。若厂商没有提供必要的基础结构层行为, 则 Globus Toolkit 引入缺少的这些功能。例如, 它提供查询软件来发现各种公共资源类型的结构和状态信息。查询软件发现如计算机 (比如 OS 版本, 硬件配置, 负载, 进度表队列状态), 存储 (例如可用的空间), 网络 (如当前及可预见的将来的负载) 等资源的结构和状态信息, 并且以一种有利于上层协议, 特别是资源层协议实现的形式将此种信息打包。另一方面, 资源管理一般假设在本地资源管理器的域内。一个例外是关于保留和分配的 (GARA) [12] 的通用体系结构, 它提供能用于实现这样一些资源的预先保留的“槽管理器”, 而这些资源本身并不支持预先保留能力。除此以外, 已发展提高到支持预先保留的可移植批量系统 (PBS) 和 Condor。

- 用于通信的协议来自 TCP/IP 协议栈, 特别是来自 Internet 协议体系结构的 Internet (IP 和 ICMP) 层, Transport (TCP, UDP) 层和应用 (DNS, OSPE, RSVP 等) 层。基于 Public-Key 的网络安全基础性基础结构 (GSI) 协议用于认可、通信保护和认证。GSI 建立在传输层安全协议之上并对其有所扩展, 以解决在 3.2 节连通层中列出的大部分问题, 特别是一次签字, 委派以及与各种本地的安全性解决方案 (包括 Kerberos) 和基于用户的信任的集成。认证的栈持有者控制通过认证工具支持, 该认证工具允许资源所有者通过一般认证和

访问(GAA)控制接口来集成本地策略。在当前的工具文本中未提供对受限委派的丰富的支持,但这些功能在原型中已经演示过了。

• 在资源层,选择了一个小且几乎完全基于标准的协议集。包括

1) 网格资源协议 GRIP (GRIP 当前基于轻量级目录访问协议 LDAP) 用于定义标准资源信息协议和相关的信息模式。一个相关的软状态资源登记协议,即网格资源登记协议(GRRP)用于登记带有 Grid 索引信息服务器的资源^[13]。

2) 基于 HTTP 的网格资源访问和管理协议用于分配计算资源及监控和控制该资源上的计算。

3) 数据访问的管理协议 GridFTP^[14] 是文件传输协议 FTP 的扩展版本。扩展包括使用连通层安全性协议,部分文件访问,高速传输的并行管理。FTP 作为数据传输协议的基础是由于它对第三方传输的支持及其分开的控制和数据通道有利于精细的服务器的实现。

4) LDAP 也被用作目录访问协议。

Globus Toolkit 定义这些协议中的每个协议的客户端的 C 和 Java API 以及 SDKs。为了有利于将各种资源(计算、存储、网络)集成到网格中,也提供各种协议的服务器方 SDK。例如,资源信息服务实现服务器方 LDAP 功能,此种功能带有供任意资源信息发行所用的调出。工具包的一个重要的代表服务器的元素是“看门人”,它提供本质上是讲 GRAM 协议并能用于派发各种本地操作的 GSI 认可的“inetd”。一般的安全服务 API 被用于获取,转交以及验证认证信任书,并提供在这些 SDK 和服务器中的传输层集成性和私有性,使得能在连通层置换候选的安全性服务协议。

• 除了在 3.4 节中列出的服务之外,汇集层许多服务建立在 Globus 的连通和资源协议之上,叫做元目录服务。这种服务引入 Grid 信息索引服务器(GIIS)支持在资源子集上的任意观察,LDAP 用于访问资源专门的 GRIP 以获得资源状态,GRRP 用于资源登记。此外,备份目录和备份管理服务用于在网格环境中支持数据集复制品的管理^[15]。一个在线信任仓库服务(MyProxy)提供关于代理信任的安全存储^[16]。DURO 共同分配库提供资源共同分配的 SDK 和 API^[27]。

5. 网格体系结构实例

我们通过两个实例来说明实际的网格体系结构如何工作。表 3 显示可以用于实现在“网格:面向虚拟组织的资源共享技术”一文中的图 1 中介绍的多学科模拟及射线跟踪的服务。二者基本的基础结构元素是相同的,而且,每种资源讲关于通信和安全性的标准的连通协议。对询问、分配及管理,讲标准的资源协议。在此之上,每个应用使用一般的和更加应用专门的混合的汇集服务。

表 3 用于构建多学科模拟及射线跟踪

	多学科模拟	射线跟踪
汇集层 (应用专门的)	解答者耦合器,分布式数据归档者	检查点,任务管理,失败重复,策划
汇集层 (一般的)	资源发现,资源代理,系统监控,团体认证,证书撤销	
资源层	访问计算,访问数据,访问关于系统结构、状态、性能的信息	
连通层	通信(IP),服务发现(DNS),认可,认证,委派	
基础结构层	存储系统,计算机,网络,代码仓库,目录	

(见文[20]图 1 中两个应用实例的网格服务)

在射线跟踪的例子中,该工作基于大吞吐量计算系统。为了管理 VO 环境中大量独立的任务的执行,系统必须保留现行(活动)的和逼近的任务集的踪迹,找出每个任务的合适的资源的位置,策划那些资源的执行表,探测并响应各种类型的失败。在这里的网格体系结构的上下文中,一个实现除使用标准的资源和连通协议外,它既能使用领域专门的汇集服务(动态检查点,任务池管理,失败重复),也能使用更一般的汇集服务(代理,执行表和公共输入文件的备份)。Condor-G 给出向此目标前进的第一步。

在多学科模拟应用中,在最高层的问题是完全不同的,可以使用某些应用框架(如 CORBA, CCA)来根据它的各种组件构建应用。同时也要求以下机制:发现适合的计算性方面的机制,在这些资源上的保留时间的机制,策划执行表机制,提供对远程存储的访问机制等等。这里,也使用了一些领域专门的汇集服务(例如,解答者耦合器,分布式数据归档者),但基本的支持与射线跟踪是相同的。

结语 网格技术正在全球内作为新的技术浪潮迅猛发展,其对未来 IT 业以及计算机科学研究与发展都将产生重要和深远的影响。

对我国的计算机研究与发展来说,追踪这一最新的前沿学科并迅速投入实际的研究之中尤为重要,因为网格是一种面向虚拟组织的资源共享技术,这种资源是客观已经存在的,并具有明显的特性。比如,信息类软资源,如文件,数据库,知识等,可以被充分共享,不象物质资源那样只能被分享。再如,当两人共享一本书或一台 PC 机时,在某个时间段只能有一个人在看书或只有一人使用 PC 机,只能做到分时共享,本质上是一种分享,因为具有排它性。其它,如能源中的水、煤、油、电等等,也只能分享。这样,信息资源的充分共享特性能通过网格发挥得淋漓尽致。对高性能计算机,网络及航空航天的综合模拟设备等硬资源,往往造价昂贵。虽然其使用具有排他性,但可以分时共享,网格技术可使空闲(如夜晚)或闲置一定时间(如设计及总结期间)的设备物尽其用,发挥其最大效应,达到经济学倡导的资源最佳使用效果。将社会的每个人/团体提供的有限资源形成虚拟社会取之不穷之资源宝藏,网格技术将使世界变得更加美好。

信息时代,人们正在以互联网的速度发展应用。实际上,应用应是一切研究的根本目标。网格计算面向应用的特性尤为突出。在网格研究发展方面,我们面临的机遇与挑战与国际上是相同的。网格技术是资源共享方面重要的创新,影响深远,意义重大。对这一新研究领域,应该说我们与他人的差距不会很大,而事实上在这一领域既存在许多极富挑战性研究探索的问题,也有创建实际网络的发展问题。计算机科学与技术发展的速度令人慨叹知识老化之快,知识更新之必要。传统的卖油翁式的研究受到挑战,远离主流技术前沿的“成果”是永远不能孵化的菜青虫,对我国的发展既不可能对现在起任何积极作用,更不可能对将来产生任何影响,必然为历史所抛弃。我国的“86.3”计划已开始重视网格的研究,以计算所为首的一些研究所和高校已投入了相关的研究^[17,18]并提出了一些自己的见解,这是令人欣慰的。我们追踪网格的目的,正是为了开展相关研究,将人工智能技术与网格技术结合,在网格上,提供软件推理机资源,从而提升网格的问题求解能力和智能应用水平。相关的工作正在进行中。

参考文献

- 1 The Globus Project. <http://www.globus.org>

- 2 Foster I, Kesselman C. The Globus Project: A Status Report. In: Proc. Heterogeneous Computing Workshop, IEEE Press, 1998. 4~18
- 3 Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. Intl. J. Supercomputer Applications, 2001, 15(3)
- 4 Realizing the Information Future: The Internet and Beyond. National Academy Press, 1994. <http://www.nap.edu/readin-groom/books/rtif/>
- 5 Stevense R, et al. From the I-WAY to the National Technology Grid. Communications of the ACM, 1997, 40(II): 50~61
- 6 Johnston W E, Gannon D, Nitzberg B. Grids as Production Computing Environments: The Engineering Aspects of NASA's Information Power Grid. In: Proc. 8th IEEE Symposium on High Performance Distributed Computing, IEEE Press, 1999
- 7 Beiriger J, et al. Constructing the ASCI Grid. In: Proc. 9th IEEE Symposium on High Performance Distributed Computing, IEEE Press, 2000
- 8 <http://griphyn.org>
- 9 <http://www.neesgrid.org>
- 10 <http://www.ppdg.net>
- 11 <http://www.en-datagrid.org>
- 12 Foster I, Roy A, Sander V. A Quality of Service Architecture that Combines Resource Reservation and Application Adaption. In: Proc. 8th International Workshop on Quality of Service, 2000
- 13 Czajkowski K, Fitzgerald S, Foster I, Kesselman C. Grid Information Services for Distributed Resource Sharing. 2001
- 14 Allock W, Foster I, Kesselman C, Tuecke S. Protocol and Services for Distributed Data-Intensive Science. ACAT2000 poceedings, 2000. 161~163
- 15 Allcock B, et al. Secure, Efficient Data Transport and Replica Management for High-Performance Data-Intensive Computing. In Mass Storage Conference. 2001
- 16 Czajkowski K, et al. Resource Management Architecture for Metacomputing Syetems. In The 4th Workshop on Job Scheduling Strategies for Parallel Processing, 1998. 62~82
- 17 李国杰. 信息服务网格——第三代 Internet. 计算机世界, 2001. 11. 15
- 18 徐志伟. 因特网之后是什么? —— 网格技术探讨. 计算机世界, 2001. 11. 22
- 19 严世宗译. IBM 预测 Grid Computing 是因特网的未来. 计算机世界, 2001. 10. 15
- 20 高全泉. 网格: 面向虚拟组织的资源共享技术. 计算机科学, 2002, 30(1)

(上接第 5 页)

识、确定认证及创始共享的机制必须是灵活的和轻量级的,这样才能快速地建立和改变资源共享安排。由于 VO 与现有的机构/技术相辅相成,而不是取而代之,所以不能要求对本地的策略有本质的改变,并且必须允许单独的机构在他们拥有的资源上保留最终的控制权。由于协议管理组件之间的交互,而不是管理组件的实现,所以保留了本地的控制。

3) 服务是重要的(通过服务向 VO 伙伴提供协议行为)一服务完全由它所讲的协议和它实现的行为定义。对于访问计算、访问数据、资源发现、合作安排、数据备份等行为来说,标准的服务定义允许我们增强提供给 VO 伙伴的服务,并将那些妨碍 VO 应用开发的资源专门的细节抽象掉。

4) API(应用程序设计接口)和 SDK(软件开发包)是协议的重要辅助手段 对于 VO 来说,当然有比互操作性、协议和服务更进一步和更细的要求。对开发者来说,他们必须能够开发在复杂的和动态的执行环境中的精细的应用;对用户来说,他们必须能够操作这些应用。因此,应用的鲁棒性、正确性、开发成本及维护成本都是一些重要的需要关心的问题。标准的抽象,API 以及 SDK 是协议的辅助而不是替代者。如果没有标准的协议,互操作性在 API 层能被完成只能通过以下两种方式完成:或者到处使用一个单独的实现,而这在许多有趣的 VO 中根本行不通;或者通过使得每个实现知道其它实现的细节。

结语 本文对公众关心的网格问题进行了一般性的综述。为了理解网格的产生背景,我们从虚拟组织及它们从事的活动出发,概括了虚拟组织的建立及虚拟组织成员间资源共享的特性和需求。并且指出,因已有技术不具备这些特性,不能满足这类需求,故现实中确实存在一类“网格问题”。本文也给出网格讨论中的基本术语/概念的定义,标识了建立虚拟组织共享所要求的基本特性和需求,概括介绍了网格的基本设计思想及关键理念。这些内容基本回答了在本文引言中提出

的前两个问题。至此,网格的定义可以一言以蔽之,即动态的、可伸缩的虚拟组织的可控且可调的资源共享和资源使用。

Ian Foster 等在文[4]中,对网格研究的领域,关注的主要问题及关键技术作了界定,这些定义、术语/概念、原理、关键技术和体系结构,既提供了网格讨论的基本公共语言和交流基础,也提供了实际可以运用和借鉴的理论框架和参考模型。这对全面准确认识网格是必要的和及时的。对网格深层技术,即网格体系结构的需求和功能描述、网格与同时代其它分布计算技术/中间件的区别与关系(以及网格的不同观点的)讨论,我们另文专述。

参 考 文 献

- 1 The Globus Project, <http://www.globus.org>
- 2 Foster I, Kesselman C. The Globus Project: A Status Report. In: Proc. Heterogeneous Computing Workshop, IEEE Press, 1998. 4~18
- 3 孙定. 从理论到实践—Globus 网格计算理论及其应用. 计算机世界, 2001. 11. 5
- 4 Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. Intl. J. Supercomputer Applications, 2001, 15(3)
- 5 Allock W, et al. Data Management and Transfer in High-Performance Computational Grid Environments. Parallel Computing, 2001
- 6 Linn J. Generic Security Service Application Program Interface Version 2. Update I, IETF, RFC 2743, 2000. <http://www.ietf.org/rfe/rfe2743>
- 7 Aiken R, et al. Network Policy and Services: A Report of a Workshop on Middleware. IETF, RFC2768, 2000. <http://www.ietf.org/rfc/rfc2768.txt>
- 8 高全泉. 网格: 面向虚拟组织的资源共享技术. 计算机科学, 2003, 30(1)