

一种支持端到端 QoS 的服务网格体系结构*

胡春明⁺, 怀进鹏, 沃天宇, 雷磊

(北京航空航天大学 计算机学院, 北京 100083)

A Service Oriented Grid Architecture with End to End Quality of Service

HU Chun-Ming⁺, HUAI Jin-Peng, WO Tian-Yu, LEI Lei

(School of Computer Science and Engineering, BeiHang University, Beijing 100083, China)

+ Corresponding author: Phn: +86-10-82316262, Fax: +86-10-82316796, E-mail: huem@act.buaa.edu.cn, <http://act.buaa.edu.cn>

Hu CM, Huai JP, Wo TY, Lei L. A service oriented grid architecture with end to end quality of service. *Journal of Software*, 2006,17(6):1448–1458. <http://www.jos.org.cn/1000-9825/17/1448.htm>

Abstract: Providing non-trivial QoS is one of the primary goals of the grid computing. Since extensive efforts have been made on the development of basic grid components such as execution framework and resource brokering system, there is still a great challenge to build up a QoS-enabled grid architecture for today's service grid. In this paper, a QoS-enabled service grid architecture (QESA) is proposed; the concept of differential guaranteed service is introduced to provide QoS guarantee at service layer; a QoS-aware service discovery and scheduling mechanism are designed to provide QoS guarantee for grid jobs. The performance of the QESA and related mechanisms has been evaluated by a grid application called AREM. The results show that QESA can provide QoS guarantee to grid jobs.

Key words: service grid; middleware; architecture; QoS(quality of service); service discovery; resource capacity reservation; QESA; CROWN

摘要: 随着网格技术研究和应用的不断深入,缺乏服务质量保证已成为制约网格应用的瓶颈之一.如何建立一个开放的、可扩展的体系结构,是当前支持端到端服务质量的服务网格体系结构研究的重要问题.提出了支持服务质量的服务网格层次体系结构(QESA),并通过对网格服务容器、信息服务与调度服务的服务质量扩展实现了该体系结构,设计了基于资源能力管理与服务预留的服务质量保证方法,讨论了服务质量感知的服务发现和服务调度的实现机制.通过中尺度天气系统降水量预报的网格应用(AREM)对其进行了性能评价,结果表明,基于 QESA 的服务网格中间件可为网格作业提供确定的服务质量保证能力.

关键词: 服务网格;中间件;体系结构;QoS;服务发现;资源能力预留;QESA; CROWN

中图法分类号: TP393 文献标识码: A

网格计算的一个重要特征是支持面向 Internet 环境的资源共享与协同问题求解,十年来经历了从面向计算

* Supported by the National Natural Science Foundation of China under Grant No.90412011 (国家自然科学基金); the National Natural Science Funds for Distinguished Young Scholar under Grant No.60525209 (国家杰出青年科学基金); the National High-Tech Research and Development Plan of China under Grant No.2005AA119010 (国家高技术研究发展计划(863)); the National Grand Fundamental Research 973 Program of China under Grant No.2005CB321803 (国家重点基础研究发展规划(973))

Received 2006-01-21; Accepted 2006-03-13

到计算与服务相融合的发展过程,基于服务的开放网格体系结构已成为网格技术发展的重要趋势。随着网格技术和应用的不断深入,网格核心功能与传统的非功能特性共同成为研究应用的焦点,一些关键领域或业务应用场景(如实时计算、可视化等)对网格提出了严格的服务质量(quality of service,简称 QoS)要求,缺乏 QoS 保证逐渐成为制约网格应用的瓶颈,并引起了学术界和工业界的共同关注。

为服务网格设计支持 QoS 的体系结构具有以下特殊需求:首先,网格环境中资源类型多,体系结构应能够提供对不同资源类型的扩展性;其次,网格作业的 QoS 需求多样,这要求该体系结构能够支持不同的 QoS 参数,且当新的 QoS 参数出现时能够尽可能地在已有的体系结构下进行功能扩充;第三,网格最终将会向区域乃至全球的大规模分布计算环境演化,这对网格体系结构之间的互操作性提出了新的要求。因此,如何建立一个开放的、可扩展的体系结构,是当前支持端到端 QoS 的服务网格体系结构研究的重要问题。

针对上述问题,本文借鉴了计算机网络中的端到端 QoS 管理的概念以及分布式构件技术中 QoS 的研究思路,将服务级协定(service level agreement,简称 SLA)作为第一类实体扩展了面向服务的体系结构,提出了支持 QoS 的服务网格体系结构 QESA(QoS-enabled service grid architecture,简称 QESA),并通过对网格服务容器、信息服务与调度服务的 QoS 扩展实现了该体系结构。在网格服务层,本文提出了区分确保型网格服务的概念以及基于资源能力管理与服务预留的实现方法;在具有确定 QoS 保证的网格服务基础上,设计了 QoS 感知的服务发现和服务调度的实现机制,并在我们研制的 CROWN 服务网格中间件基础上扩展实现了 QESA,通过中尺度天气系统降水量预报的网格应用对其进行了性能评价,结果表明,支持 QoS 的 CROWN 中间件可为网格作业提供确定的 QoS 保证能力。

本文第 1 节分析相关研究工作,第 2 节讨论网格端到端 QoS 管理的概念以及支持 QoS 的服务网格体系结构 QESA,第 3 节讨论 QESA 的各个软件体系结构模块的设计思想与主要功能,第 4 节通过基于中尺度天气系统降水量预报应用的一组实验,分析通过 QESA 为网格作业提供确定 QoS 保证的效果及其性能。

1 相关工作

QoS 研究最早起源于 APANET 的性能和可靠性保证需求。20 世纪 90 年代初,随着高速网络技术的发展(如 B-ISDN 技术和 ATM 技术的出现等),以音频、视频流为代表的一类分布式多媒体应用开始出现,针对流媒体对网络性能及端节点处理能力的综合需求,出现了端到端 QoS 的概念^[1]。端到端 QoS 强调对多个层次、多个 QoS 参数的综合管理与控制,主要研究内容包括端到端 QoS 体系结构、QoS 的描述与协商、QoS 在层次间的映射机制以及端到端的接纳控制和资源预留等,出现了许多针对网络系统资源(如网络带宽等)和端系统资源(如 CPU 计算能力、内存和本地存储等)进行预留和分配的研究工作。这些研究结果对网格 QoS 的研究有着重要的指导作用。

分布式系统(如 CORBA, Web 服务等)的 QoS 研究是分布计算环境的重要内容之一,主要研究性能、可靠性、可用性等 QoS 参数的表示、协商和管理机制,提供支持 QoS 的分布式计算环境组件。这些研究工作的基本特点是,在分布构件层提供确定性的行为,并在其上层提供分布构件的选取以及构件执行过程的质量保证能力。如针对 CORBA 环境的 QuO^[2]、uQoS-Mngr^[3]以及针对 Web 服务的 WSLA 及其运行框架^[4,5]、UDDI 的 QoS 扩展^[6]以及 Web 服务性能管理^[7-9]等。

网格的 QoS 研究出现于 2000 年前后,但目前尚未形成明确的技术体系,其典型研究工作包括 QoS 需求的描述、资源的预留与联合分配、支持 QoS 的资源(服务)选取与映射、支持特定 QoS 参数的资源调度以及域间的 QoS 协商与资源管理等机制。

在网格计算的早期,为了保证计算密集型和数据密集型应用的性能,资源预留(resource reservation)和协同分配(co-allocation)成为 QoS 管理的主要研究重点^[10,11]。服务网格的概念提出之后,网格 QoS 研究工作的重点开始转移到面向服务的 QoS 管理方面。它一方面沿用了互联网 QoS 中“服务级协定(service level agreement,简称 SLA)”的概念,通过基于 XML 的 QoS 描述语言描述用户的 QoS 需求;另一方面,通过定义统一的服务接口扩展和服务间交互协议实现服务的 QoS 协商和管理。在网格 QoS 系统和 SLA 管理框架方面的研究工作包括

G-QoS^[13], SLM Engine^[5], CREMONA^[14], Gsmon^[15]等,它们的一个重要特点是开放性和标准化, QoS 的表示与 QoS 管理机制往往通过定义标准规范或向 OGSA 结构中扩充新的 QoS 支撑服务得以实现.但总体上看,目前面向网格作业的 QoS 管理机制还处在起步阶段,尤其是还没有形成明确的、支持端到端 QoS 的服务网格体系结构,网格 QoS 的研究仍缺乏完整、清晰的技术体系,对作为服务网格核心的网格服务的 QoS 控制机制还很少有研究.目前还没有采用 OGSA/WSRF 结构的服务网格中间件能够很好地支持网格应用的 QoS 需求.

2 支持 QoS 的服务网格体系结构

2.1 网格的端到端 QoS 管理

端到端 QoS 是网络 QoS 研究中的概念,它强调了一种总体考虑的视角和通过 QoS 参数逐层映射保持 QoS 策略一致的思想^[1].Campbell 等人在文献[1]中讨论了一个端到端 QoS 系统的基本架构,并给出了端到端 QoS 系统的 5 个基本原则,即集成原则、分离原则、透明原则、异步资源控制原则和合理性能原则.在保证上述 5 原则的前提下,一个端到端 QoS 体系可以划分为 QoS 规范(QoS specification)和 QoS 实现机制两大方面.针对网格作业的特点及其执行过程,面向网格作业的端到端 QoS 也需要提供相应的 QoS 规范和实现机制.

QoS 规范是在 QoS 对象和 QoS 保证系统之间建立的对应用 QoS 需求和管理策略的统一描述.所谓 QoS 对象是指 QoS 管理所针对的主体,在网络 QoS 研究中,网络向上层应用提供传输服务,包(packet)或者流(flow)就是 QoS 对象.服务网格通过 SOA 结构连接各类分布资源,将资源能力在广域网络范围内进行共享与协同,向网格用户提供问题求解服务.类似地,我们将网格用户所提交的作业视为网格端到端 QoS 研究的 QoS 对象.

在描述作业对网格环境的需求时,服务网格中作业的 QoS 规范可以通过一组定性或定量的 QoS 维(QoS dimension,即 QoS 指标)构成,每个 QoS 维描述了所提供的一个侧面的需求.对于不同类型的服务,其 QoS 规范所包含的 QoS 维可能不同.目前,网格 QoS 研究所涉及的 QoS 维包括时间、开销、可靠性、可用性、保真度(fidelity)以及公平性等.从表现形式上看,QoS 规范还可分为基于脚本、对象、属性或基于 XML 语言的^[16].

QoS 实现机制是 QoS 规范的具体实现方式,它可以进一步划分为 QoS 提供机制(QoS provision mechanism)、QoS 控制机制(QoS control mechanism)和 QoS 管理机制(QoS management mechanism)3 个方面.

1) QoS 提供机制:包括 QoS 映射、接纳控制(admission control)和资源预留协议 3 个部分.

QoS 映射自动地在相邻的两个系统层(如操作系统层与传输层、网络层)之间进行 QoS 表示机制的转换,它将高层的 QoS 需求转换为底层的 QoS 实现.在网格的 5 层结构中,作业 QoS 位于应用层,其 QoS 目标的实现需要通过逐层映射最后落实到资源的具体 QoS 控制上.

接纳控制将系统剩余的能力与请求所要求的能力进行比较,在必要时,拒绝超出系统负载的请求以保护系统和之前已接纳的请求.在服务网格中,接纳控制体现为网格系统根据用户提出的作业 QoS 需求判断是否可满足的过程.此外,端到端的接纳控制还要求参与处理的各个环节进行一致、相容的接纳控制检测,任何一个环节无法提供足够的资源能力,都需要拒绝作业请求.

资源预留协议将系统范围内的资源能力合理分配给不同的处理请求,并通过某种机制确保分配的一致性和有效性,主要体现在作业调度和资源能力调度两个层次.作业调度层通过元调度算法将有限的服务资源与具有不同 QoS 需求的用户作业绑定,追求网格环境的整体作业执行效率;资源能力调度通过结合资源自身提供的处理机制进行资源能力管理和预留,向网格上层提供确定的 QoS 保证.

2) QoS 控制机制:主要强调在服务提供过程中进行适当的调节和处理,以满足 QoS 对象的 QoS 需求.针对不同的 QoS 对象,所需的 QoS 控制机制可能完全不同,服务网格的 QoS 控制主要表现在以下两个方面:

(1) QoS 感知的服务发现:它是 Web 服务和网格 QoS 研究的重要内容,也是进行元级调度的基础.在进行服务发现时,需要尽可能地考虑作业的 QoS 需求,并将其作为服务发现的条件对可用服务进行进一步的筛选,要求网格的信息服务体系结构必须具有扩展的信息表示模型和匹配算法以及可接受的性能.

(2) 网格服务的 QoS 管理与元调度:网格服务是服务网格的基本组成模块,对网格服务的 QoS 管理是为网格作业提供端到端 QoS 的基础.其基本思想是:通过对网格服务所封装的资源能力进行建模和管理,对资源能力

进行预留和分配,为网格服务提供区分的 QoS,以满足上层网格作业对网格服务的不同需求,实现对资源能力的高效利用。

3) QoS 管理机制:对服务提供过程中的监控、管理以及出现 QoS 违约(QoS violation)时的处理过程.在服务网格中,它表现为对运行时的服务级协定的维护和管理、对作业执行过程中的 QoS 指标的监控及后续处理(如应用层的自适应、重协商、降级服务、中止服务并进行补偿)等。

2.2 QESA:支持QoS的服务网格体系结构

支持端到端 QoS 的服务网格体系结构 QESA 是一种面向服务网格的特殊应用需求,支持端到端 QoS 管理的软件体系结构,它是面向服务的体系结构(service oriented architecture,简称 SOA)在特定应用需求下的定制和具体化,是一种领域特定的专用体系结构。

SOA 是在分布式环境下建立松耦合软件系统的一种典型的体系结构,提供了按功能组织服务的模式,每一个服务都提供一组定义良好的功能集合.基于 SOA 的应用可被视为一组相互交互的服务,这种基于服务的形式功能描述为系统的灵活性、可扩展性和开放性提供了基础,有效地支持了系统透明性、动态升级和演化。

在面向服务的体系结构中提供 QoS 保证,一个重要特征就是在 SOA 中加强对服务绑定过程的控制,针对服务绑定过程建立 QoS 的约定、监控以及资源分配,在服务提供者与服务请求者之间建立语义清晰、明确的“服务绑定协定”和基于协定的方式进行服务的绑定.扩展后的 SOA 结构保留了原来结构中的简单绑定形式,以兼容基本的尽力而为型服务。

与 SOA 相比,这一扩展的主要特点是:

- (1) 服务注册过程:需要在统一的服务描述语法和语义中增加关于 QoS 的描述信息,以指导服务发现过程。
- (2) 服务发现过程:在基本服务发现的基础上通过增加一些 QoS 限制对所找到的服务提供者进行过滤,得到一组满足基本 QoS 需求的服务提供者。
- (3) 服务绑定过程:在建立绑定之前需要同服务发现过程所获得的可用服务提供者展开 QoS 协商,并最终从中选择一个(或多个,如某些场合需要冗余计算以提高应用的可用性)服务提供者 SLA.在后续的服务绑定和调用中,服务提供者和服务请求者应遵从并努力履行 SLA 的约定.在约定的 QoS 无法达到时,还需要通过重协商或服务请求者的自适应机制将损失减少到最小。

2000 年前后,学术界提出了许多网格体系结构和分类方法,其中最具代表性的工作是以协议为中心、基于“沙漏模型”的层次结构,自底向上依次包括构造层、连接层、资源层、汇聚层和应用层 5 个层次.随着 Web 服务技术的不断成熟以及对网格领域相关标准化工作的开展,如何构造服务网格体系结构已成为网格研究者面临的重要研究内容.2002 年提出的开放网格服务体系结构(open grid service architecture,简称 OGSA)和 2004 年提出的 Web 服务资源框架(web service resource framework,简称 WSRF)是此类研究所取得的重要结果.OGSA 是一个面向服务的体系结构,采用服务作为资源的统一封装和互操作形式,通过定义标准的网格协议使网格成为一个开放系统;WSRF 是对 OGSA 和服务网格概念的进一步发展,它为更好地将 OGSA 与 Web 服务实现结构相结合提出了一组管理结构和交互协议.文献[18]给出了一个基于 SOA 的服务网格层次体系结构,对照网格“沙漏模型”将服务网格区分为资源/构造层、SOA 层、面向应用的工具层和网格应用层.在服务网格的层次化体系结构基础上,QESA 对分布在资源层和汇聚层的软件模块进行了 QoS 感知的功能扩展(如图 1 所示),通过对资源能力的管理和预留,在网格服务层及其上层为网格作业提供 QoS 保证。

图 1 中,资源层和汇聚层的各类网格服务(包括资源封装服务、部分领域相关的服务以及各类网格共性服务)均实现为 OGSA 兼容的网格服务,网格信息服务为这些服务提供注册与发现机制;在汇聚层的上层及应用层的下层是一些支持网格的工具,如针对特定应用模式而开发的应用程序框架及基于 Web 的网格门户等.QESA 针对服务网格的 QoS 管理需求所扩展的部分主要包括:

- (1) QoS 感知的网格信息服务:在 QESA 中,信息服务应当支持 QoS 感知的服务发现和选取,这直接体现为对网格信息服务的信息存储模型、查询接口和服务选取机制的扩展.扩展后的网格信息服务可以为服务存储更多的 QoS 属性(如性能、可靠性等),并将这些属性作为服务选取的条件。

(2) QoS 感知的网格服务容器:在 QESA 中,服务容器应当支持服务级的 QoS 管理,通过服务预约进行接纳控制和资源能力分配,保证已接纳的服务请求的 QoS.首先,需要对 OGSA 网格服务接口进行扩展,定义服务预约接口;其次,需要在服务容器中扩展对 SLA 的管理功能,用于服务调用时 SLA 的创建和维护;第三,需要在服务容器中提供资源能力管理机制,对网格服务所封装的网格资源的能力进行建模,并根据服务请求对资源能力进行预留和合理分配.

(3) QoS 感知的元调度服务:在 QESA 中,对元调度服务的扩展主要体现为作业调度算法的扩展.QoS 感知的网格作业描述语言用来描述用户提交的作业及其 QoS 需求,元调度服务理解用户对作业的 QoS 需求,将其映射为对服务的 QoS 需求,并根据网格服务容器的服务发现结果进行作业分配,努力保证网格作业的 QoS.

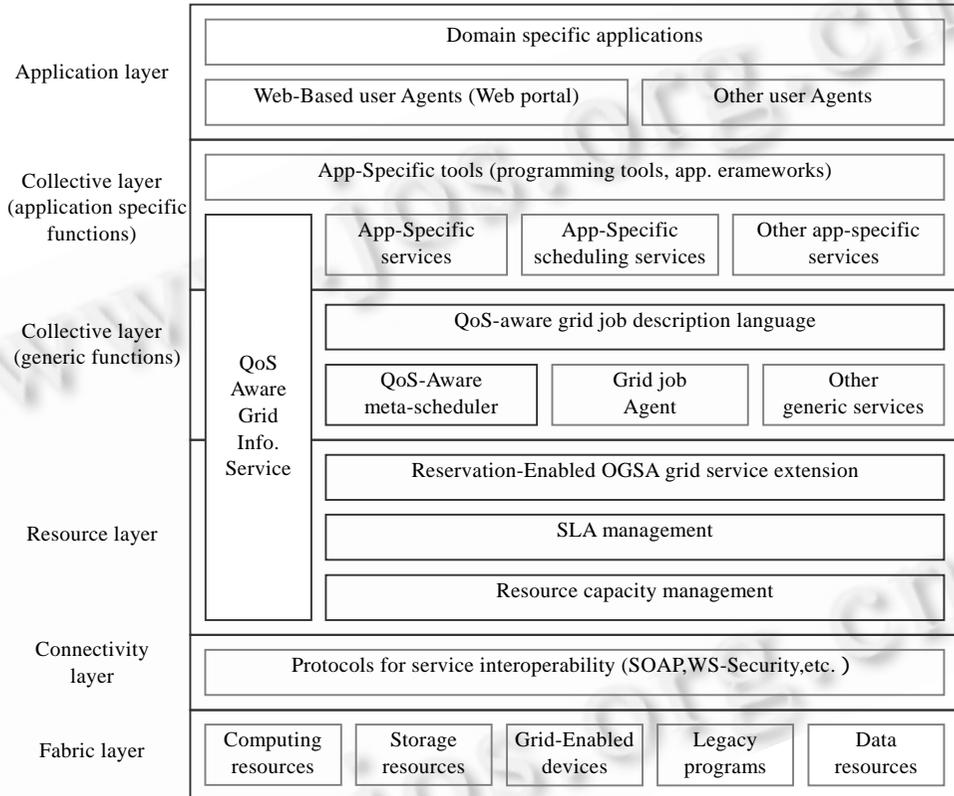


Fig.1 QESA: A layered architecture for QoS-enabled service grid

图 1 QESA:一种支持 QoS 的层次式服务网格体系结构

QESA 具有良好的开放性和可扩展性.其开放性体现在各个软件模块的设计基于已有的 Web 服务标准、服务接口及其扩展形式符合 OGSA/WSRF 的规范定义,易于同其他服务网格中间件集成与互操作;可扩展性一方面体现在通过统一的服务访问形式可屏蔽多类异构资源间的差异,提供对不同资源类型的扩展性,另一方面体现在当需要支持新的 QoS 属性时,只需定义新的资源能力类型并在资源能力管理框架下设计对应的资源能力控制机制,对 QoS 感知的服务发现和作业调度模块基本不必调整.

基于 QESA 的服务网格基本功能如图 2 所示.

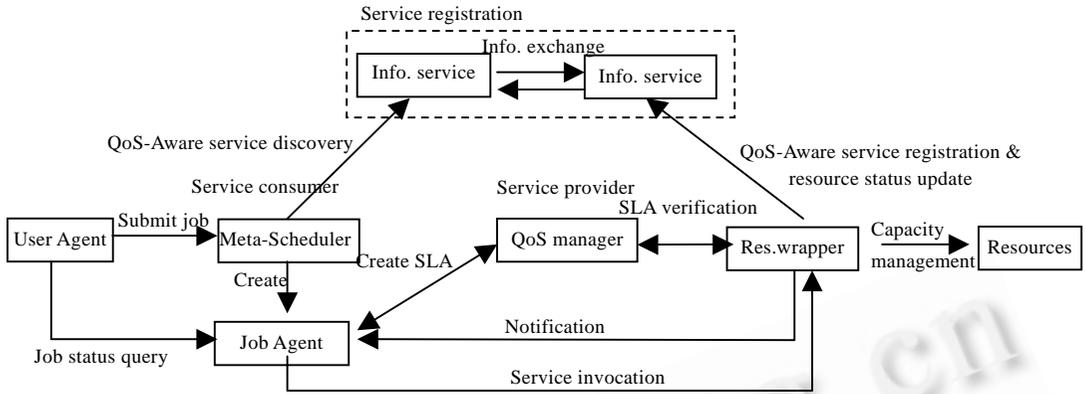


Fig.2 Functionality of a QESA-based service grid

图 2 基于 QESA 的服务网格基本功能

参与网格的资源被封装成资源服务,包括各种 QoS 属性在内的服务元数据和资源状态信息将注册到网格信息服务实例中;网格用户通过用户代理向网格的元调度服务提交作业并通过访问网格作业代理服务查询作业状态、获取作业结果;元调度服务作为 SOA 结构中的服务请求者,通过网格信息服务进行 QoS 感知的服务发现和作业调度;网格作业代理负责调用资源服务并跟踪状态,必要时采用异步通知机制获取资源服务的状态变化.服务提供者通过 QoS 管理模块和资源封装服务对网格资源的能力进行建模和管理,并根据作业需要分配资源能力,为网格服务的执行提供确定的 QoS 保证.此外,WS-*系列协议确保上述交互过程中的非功能特性,如消息安全、服务的访问控制、可靠消息传递和事务处理等.

3 系统设计与实现

QESA 最终通过扩展服务网格中间件的服务容器、信息服务和调度服务得到实现.服务网格系统 CROWN 是由北京航空航天大学研制的一组服务网格中间件系统,它支持面向服务的网格应用开发和运行,包括节点服务器(node server)、资源定位与描述服务(resource location and description service,简称 RLDS)、网格开发工具等 11 类基本构造模块.其中,节点服务器是网格服务的基本运行环境,对底层异构资源进行服务化封装.我们扩展实现了 CROWN 的节点服务器、资源定位与描述服务以及网格调度服务这些构造模块,设计实现了支持端到端 QoS 的服务网格中间件.

3.1 QoS感知的网格服务容器

网格服务容器(grid service container)是服务网格中支持服务运行管理的中间件模块.无论是通用服务(如信息服务等)、资源封装服务或应用特定的服务,都需要运行在网格服务容器中.目前的服务网格容器(如 GT 3.0 等)尚不提供网格服务层的 QoS 管理,基于这些服务容器的网格服务只能以一种“尽力而为(best-efforts)”的方式提供服务(简称 BE 型服务),其主要问题是:(a) 不能区别服务请求的重要性和紧迫性,对所有服务请求同等优先处理,缺乏对服务所封装的资源能力的统一管理和合理分配;(b) 服务的性能受到服务执行时系统负载等诸多因素的影响,响应时间等主要 QoS 指标波动较大,无法为网格上层提供确定的 QoS 保证;(c) 当对服务的请求速率提高、系统负载增大并超出系统处理能力时,缺乏必要的接纳控制和对系统过载的保护.

针对上述问题,我们定义了一种区分确保型服务(differential guaranteed service,简称 DG 型服务),它提供了不同的服务等级,并根据请求的服务级进行区分服务,努力确保各服务级的 QoS 要求得到确保.DG 型服务的一个重要特点是“服务分级”,即 DG 型服务的提供者需要提供多种可选择的 QoS 等级,同时基于服务请求所选择的 QoS 等级在服务执行时赋予合适的优先级并进行合理调度;而不同的服务级与不同的服务开销(或价格)相关联.限于篇幅,价格模型不是本文的研究重点,我们假定 DG 型服务的请求者会根据应用需求选择合适的 QoS 等级.与 BE 型服务的直接访问模式不同,DG 型服务采用“先预约后访问”的模式.在预约阶段,DG 服务客户端需要根据应用需求选择 DG 型服务提供的服务等级,通过服务预约接口向 DG 型服务进行预约.DG 型服务在收到预

约请求时需要进行接纳测试,测试通过后,DG 型服务会向客户端发出一个预约票据作为服务预约的确认.DG 型服务的客户端持该票据进行服务访问,并获得 DG 型服务所承诺的质量保证.

对资源封装服务进行 QoS 管理的目的是通过扩展网格服务容器,支持 DG 型服务的部署与运行管理.具体来说,就是让网格服务容器支持资源的能力建模和管理,支持用户对服务的预约并建立和维护 SLA.通过与底层资源管理器所提供的 QoS 管理功能的集成向上层提供确定的 QoS 保证.为了支持这一特性,需要网格服务容器提供资源能力管理、服务级协定的建立和管理的功能,并对标准的 OGSA 网格服务接口进行扩展,以支持对网格服务进行预约.

(1) 资源能力管理

在服务网格中,资源被封装成网格服务,并以统一的方式提供资源能力.为了对提交给网格的作业提供确定的 QoS 保证,典型的方法是对资源能力进行建模并量化作业对资源能力的需求,实现资源能力的提前预留和分配^[11].通过对资源能力的建模和管理,可以为服务网格中的一类资源封装服务提供确定的 QoS 保证能力.

资源能力是对网格资源功能性指标的一种抽象.网格环境中的资源种类比较多,主要包括计算资源、存储资源、数据资源以及应用程序和网络可访问的仪器设备等.不同类型的资源可具有相互独立的多种资源能力.例如,对计算资源而言,最主要的资源能力是 CPU 的计算能力.此外,可分配的内存容量就是与 CPU 计算能力独立的另一种资源能力.在服务网格环境中,计算资源的能力往往通过将部署在该计算机上的计算密集型应用程序封装为服务而得到共享.

目前,时间槽(time slot)^[17,19]是进行此类资源能力管理的一种有效手段.通过对资源能力进行量化并建立其时间槽模型,我们可以直观地表示资源的可用能力和已分配的能力,并基于这些信息判断一个新的资源需求是否可以被满足.对于具有多种能力的资源(如描述一台部署了几个应用程序的计算机时,需要刻画其 CPU 计算能力和可分配的内存两个侧面)就需要几个这样的数据结构,分别刻画各个不同的能力侧面.因此,一个资源的定义可包括几类不同的资源能力.如果资源能力的请求者同时需要对多种资源能力提出要求,则该请求只有在每一种资源能力均满足需求时被接纳.

(2) 服务级协定的建立和管理

服务预约和 SLA 管理是资源能力管理的客户端.服务消费者通过服务预约接口提出预约请求并进行 SLA 协商,协商过程结束后,如服务预约被接纳,SLA 管理模块向服务消费者发放 SLA 票据.服务所封装的资源也分配相应的能力预留给该服务预约;当服务调用发生时,SLA 管理模块通过验证服务请求中携带的 SLA 票据并按照 SLA 的约定分配资源,确保资源能力.

提供区分 QoS 的分级服务有利于服务的客户端根据应用需求选择服务级别,也使资源能力的管理者可以更好地分配资源能力,以便更加有效地利用资源.每个服务提供者在提供某类服务时可以定义若干个 QoS 等级,不同的等级具有不同的参数.对性能指标而言,不同的 QoS 等级意味着不同的服务执行时间,这些不同的服务执行时间和松弛时间可以映射为相应资源能力需求的开始时间、持续时间和松弛时间.

我们将服务提供者为一个服务所提供的不同 QoS 等级称为 Service Offer,它定义了 QoS 级别和资源能力需求两个方面的内容:前者描述服务执行过程所承诺的 QoS 指标(如服务响应时间等);后者描述为达到该 QoS 指标所需要预留和分配的资源能力需求(如为达到 16 秒的响应时间,必须预留 80% 的 CPU 资源并持续 8 秒,允许的松弛时间是 8 秒).我们对基于 WS-Agreement^[12]的语法结果进行适当的扩展,以描述服务提供者所提供的这些不同的服务等级,并将 SLA 的作用对象具体化为服务的具体操作.

(3) 支持预留的 OGSA 网格服务接口扩展

获得 ServiceOffer 是进行服务预约的第 1 步,客户端需要调用服务端的 getServiceLevelDefinition() 接口向服务端发送所要调用服务的端口类型 portType、操作名 opName,并将可能影响性能的参数列表以名值对的方式序列化并传递给服务端.服务端根据这些信息计算各个服务级的 QoS 参数,返回一组 ServiceOffer 供用户选择.

此后,客户端选择所需的 ServiceOffer 并启动两阶段服务预约协议.采用两阶段服务预留协议的主要目的是

为客户端提供对多个服务进行同步预留的能力.在两阶段预约协议的第 1 阶段,客户端调用 `addServiceReservation()` 接口进行服务预约并建立 SLA 实例.如果成功,返回一个包含了 SLA 票据的 `SLAInstance`;否则,返回 `NULL`,并抛出一个 `SOAP Fault`.在两阶段服务预留协议的第 2 阶段,客户端需要调用 `confirmServiceReservation` 向服务端发送确认信息,完成服务预约.

在服务预约生效之前,客户端还可以更改和删除服务预约.更改服务预约时,需要向服务端传递已建立的 SLA 的有效票据 `oldSLAToken` 以及新选择的 `ServiceOffer`;服务端收到该请求后,首先根据 `newOffer` 进行接纳测试;如果测试通过,则建立新的 `SLAInstance` 并释放原来分配的资源能力;如果接纳测试未通过,则返回 `NULL` 并抛出 `SOAP Fault`,原有的 `SLAInstance` 仍然有效.预约变更后,也需要通过两阶段协议进行确认.删除服务预约时,需要向服务端传递已建立 SLA 的有效票据 `slaToken`;服务端收到该请求后释放已分配的资源能力,删除服务预约.

3.2 QoS感知的网格信息服务

服务发现是服务网格中的重要功能,它通过一组条件的成功匹配,连接服务提供者和服务消费者,并为用户作业选择合适的服务.在网格中,提供服务发现功能的模块是网格信息服务.作为 QESA 的重要组成部分,QoS 感知的网格信息服务支持基于 QoS 属性的服务发现和选取,需要对网格信息服务的信息存储模型、查询接口和服务选取机制的扩展.

为了支持 QoS 感知的服务发现和作业调度,我们扩展信息服务的两类 QoS 属性:一类是服务提供者提供并注册在信息服务的 QoS 属性,如服务的 QoS 等级、响应时间、可靠性、可用性等;另一类是除了将信息服务实例作为服务提供者与服务消费者之外,由独立的第三方观测到的服务提供者的 QoS 属性的可信度.为了支持 QoS 感知的服务发现和作业调度,需要在服务调用过程中增加必要的信息报告与日志.主要工作原理如下:

(1) 在 QoS 感知的服务网格体系结构中,在提供服务之前,服务提供者首先与服务请求者建立服务级协定 (SLA),并在 SLA 中约束服务双方的承诺项.以时间维为例,服务提供者可以通过服务级协定,承诺在一个确定的时间内完成服务执行,并向服务消费者返回服务结果.

(2) 一旦 SLA 建立,服务提供者需要将 SLA 报告给信息服务实例,并在收到用户请求开始执行服务之前和结束执行过程之后,通过向信息服务发送 `SERVICE_INVOKE_START` 和 `SERVICE_INVOKE_FINISHED` 消息汇报状态.

(3) 通过比较上述两个消息的时间差与 SLA 中承诺的服务响应时间评价服务执行是否违约,并根据一段时间内同一服务提供者所有 SLA 的满足情况、信息服务计算该服务类型的 QoS 置信度,作为服务匹配和元调度的依据.

3.3 QoS感知的网格调度服务

在一个提供区分确保型服务的服务网格中,服务具有接纳控制和 QoS 保证的能力.这是元调度服务提供接纳控制和 QoS 保证的基础.元调度服务可依据用户作业对 QoS 的需求进行 QoS 感知的服务发现,并通过可对服务的预约选择调用适当的服务,为用户作业提供 QoS 保证.

元调度器 (metascheduler) 也称社群调度器 (community scheduler),是使网格资源得以更容易地为最终用户所用的关键模块,是网格中间件的重要组成部分之一.它通过一组与底层资源管理器之间的预定义管理协议协调多个资源的能力,将用户作业映射到具体的资源上,为用户进入网格调度系统提供了一个一致的接口.元调度器根据网格应用的需要自动匹配可用资源并与底层资源管理器交互,代理最终用户完成应用问题求解.早期的计算网格中,网格环境通常聚合多台计算资源,它们都部署有自己的本地作业调度系统(如 LSF, OpenPBS 等).网格需要为用户作业匹配可用资源并将其提交给所分配资源上的本地作业调度器,以完成作业执行的过程.这样,网格中的作业调度器就需要同底层的本地作业调度器进行交互,因而称作元调度器.

在服务网格中,服务实现通过采用特定于本地资源管理系统的 API 与本地资源管理系统通信、封装业务逻辑.同时,元调度器也以体现为网格服务的形式,处理的对象是网格作业,通过 SOAP 消息实现服务间的交互.其

基本功能包括:

- (1) 作业管理:通过特定的元调度器接口接受用户作业、管理作业状态并根据需要对作业进行排队;
- (2) 作业调度:根据作业对资源的约束及当前可用资源状态为作业选择合适的资源,形成作业调度方案;
- (3) 作业执行管理:依据形成的作业调度方案与底层资源管理系统交互,将作业提交给底层资源管理系统并监控作业的执行.

服务网格作业的最基本形式是通过一组给定参数对一类特定网格服务的调用,一些应用环境还可通过对该基本作业形式进行组合,形成简单的执行流程.限于篇幅,本文讨论元调度服务时仅针对此类基本作业形式.

元调度服务在接到用户提交的作业请求后,首先识别出作业所请求的服务类型,依据类型访问信息服务进行服务发现,获得可用服务列表;然后,依据一定的服务匹配策略(如随机策略、基于历史信息性能最优策略等)选取服务,并通过作业执行代理发起对服务的调用.

在一个支持区分确保型服务的网格环境中,QoS 感知的元调度在将无 QoS 需求的作业调度到可用服务的同时,还支持接纳控制和服务预约两个基本功能,并通过服务层的 QoS 保证能力确保作业的执行过程满足用户要求.这时,元调度服务不仅要确定将作业调度给哪个可用服务,还需为作业选择一种合适的服务级定义.

如果作业具有 QoS 需求,元调度服务在对作业的处理过程中需要进行适当扩展:首先,需要将 QoS 需求映射为一组服务发现的条件约束,并基于它们进行服务发现;其次,在获得可用服务列表后,需要按照一定的策略对表中服务进行排序,并依次获得所有可用的 Service Offer 列表;然后,元调度服务需要从该列表中依据一定的 Service Offer 匹配策略选取服务级,并通过两阶段服务预约协议建立服务预约.元调度服务重复这个过程,直到建立服务预约为止.

4 性能评价

我们通过 AREM(advanced regional Eta-coordinate numerical prediction model)来检验基于 QESA 的服务网格中间件的运行效果.AREM 是基于中国科学院大气科学和地球流体力学数值模拟国家重点实验室(LASG)的区域 η 坐标系数值预测模型改进和发展的一个针对中尺度天气系统的有限区域的数值预报模式.它通过对国家气象中心转发的 GTS(global telecommunication system)原始气象资料进行同化处理,结合已有的地形资料产生初值,并通过 AREM 模式的时间向前积分来实现.AREM 是一类典型的计算密集型应用,随着考虑因素的增加、预报区域的扩大以及预报精度的提高,AREM 计算过程需要耗费大量的计算力.为了将 AREM 运行在一个服务网格环境并提供确定的 QoS 保证,我们需要将 AREM 计算程序封装成网格服务并部署进网格环境.

AREM 的使用过程可划分为参数获取、前期处理与准备、数值积分以及结果后处理及可视化 4 个阶段:参数获取阶段从命令行读取运行参数;前期处理与准备阶段根据运行参数准备所需的文件和目录,并根据输入参数建立 AREM 模式积分的初始场;数值积分阶段首先根据用户参数编译生成 AREM 可执行文件并执行 AREM 完成计算;后处理阶段通过可视化工具将 AREM 的计算输出文件转换成更加直观的图形.其中:AREM 是用 FORTRAN 语言编写的命令行程序,可运行于 Radhat Linux 平台上.它通过读取特定文件格式的输入参数文件并进行复杂的模式积分过程,生成特定格式的输出文件.AREM 的输出文件可通过 GrADS 绘图软件生成降水量预报图,直观地显示预测结果.

本文所使用的 AREM 模式系统是 AREM 2.4 版,其模式水平分辨率约为 37km.垂直分为 20 层,模式模拟的范围是 15°N~55°N,85°E~135°E,模式初值由 AREM 简单初值系统提供,实际使用的初值场是美国国家环境预报中心的同化分析资料.

为了检验 CROWN 服务网格中间件及其 QoS 保证机制,我们为 AREM 服务定义了 3 种 Service Offer,分别分配 30%,60%和 70%的计算能力,通过在不同系统负载下多次提交具有不同 QoS 需求的 AREM 作业以及无 QoS 需求的 AREM 作业并记录其平均执行时间来比较本文提出的 QoS 保证机制的性能.测试用 AREM 程序部署在多台 CPU 为 AMD 2500+、内存容量 1GB 并安装 10/100M 网卡的测试计算机上,通过 100M 局域网互联,计算能力的预留通过 DSRT^[20]实现,系统负载通过在测试用计算机上运行其他计算密集型应用程序来产生,选

择了 10%,50%,67% 和 80% 4 个采样点.实验结果如图 3 和图 4 所示.

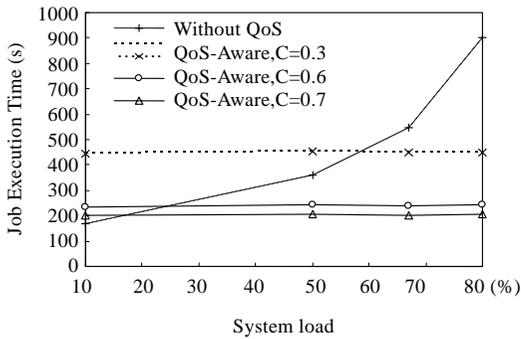


Fig.3 Service execution time at different system load in AREM application

图 3 AREM 在不同系统负载下的服务执行时间

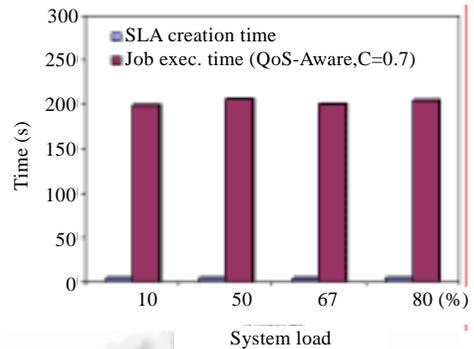


Fig.4 Comparison between SLA creation time and service execution time in AREM application

图 4 AREM 应用中 SLA 建立时间与服务执行时间

在图 3 中,无 QoS 需求的 AREM 作业执行时间随着系统负载的上升迅速增加,在系统负载达到 80% 时,执行时间已经达到 10% 时的 5.4 倍;有 QoS 需求的 AREM 作业则通过有效的资源预留获得了稳定的执行时间.图 4 比较了预留 70% 计算能力时的作业执行时间和服务预约时间,结果表明,建立 SLA 的时间在 3 秒~5 秒左右,这对于 AREM 等用时较长的计算密集型应用而言是完全可以接受的.

5 结束语

本文研究如何建立一个开放、可扩展的服务网格体系结构,以支持面向网格作业的端到端 QoS.在借鉴计算机网络中的端到端 QoS 管理的概念及分布式构件技术中的 QoS 研究思路的基础上,本文提出了一种支持 QoS、开放可扩展的服务网格体系结构 QESA,设计了面向网格作业的 QoS 实现机制;在网格服务层,提出了区分确保型网格服务的概念以及基于资源能力管理与服务预留的实现方法;在具有确定 QoS 保证的网格服务基础上,设计了 QoS 感知的服务发现和服务调度的实现机制,并在我们研制的 CROWN 服务网格中间件基础上扩展实现了 QESA.通过中尺度天气系统降水量预报的网格应用对其进行了性能评价,结果表明,支持 QoS 的 CROWN 中间件可为网格作业提供确定的 QoS 保证能力,为进一步研究面向服务的软件体系结构及其 QoS 保证技术提供了有益的尝试.

References:

- [1] Campbell A, Aurrecochea C, Hauw L. A review of QoS architectures. In: Proc. of the 4th Int'l Workshop on Quality of Service (IWQoS). 1996. <http://citeseer.ist.psu.edu/campbell96review.html>
- [2] Zinky JA, Bakken DE, Schants RD. Architectural support for quality of service for CORBA objects. Theory and Practice of Object Systems, 1997,3(1). <http://citeseer.ist.psu.edu/zinky97architectural.html>
- [3] Tien DL, Villin O, Bac C. Resource manager for QoS in CORBA. In: Proc. of the 2nd IEEE Int'l Symp. on Object-oriented Real-time Distributed Computing (ISORC). 1999. 213-222. <http://ieeexplore.ieee.org/search/>
- [4] Ludwig H, Keller A, Dan A, King RP, Franck R. Web service level agreement language specification (WSLA). IBM Corporation, 2002. <http://www.research.ibm.com/wsla/documents.html>
- [5] Keller A, Ludwig H. The WSLA framework: Specifying and monitoring service level agreements for web services. IBM Research Report, RC22456 (W0205-171), 2002.
- [6] Ran S. A model for web services discovery with QoS. ACM SIGecom Exchanges, 2003,14(1):1-10.
- [7] Farkas P, Charaf H. Web services planning concepts. Journal of WSCG, 2003,11(1). http://wscg.zcu.cz/Rotor/NET_2003/Abstracts/Farkas.htm
- [8] Pacifici G, Spreitzer M, Tantawi A, Youssef A. Performance management for cluster based web services. http://www.research.ibm.com/autonomic/research/papers/pacifici_Tech_Report.pdf

- [9] Xu MW, Hu CM, Liu XD, Ma DF. Research and implementation of web service differentiated QoS. *Journal of Computer Research and Development*, 2005,42(4):669–675 (in Chinese with English abstract).
- [10] Wolf LC, Steinmetz R. Concepts for reservation in advance. *Kluwer Journal on Multimedia Tools and Applications*, 1997,4(3). <http://www.kom.e-technik.tu-darmstadt.de/publications/abstracts/WS97-1.html>
- [11] Foster I, Kesselman C, Lee C, Lindell B, Nahrstedt K, Roy A. A distributed resource management architecture that supports advance reservations and co-allocation. In: *Proc. of the Int'l Workshop on Quality of Service (IWQoS'99)*. 1999. 27–36. <http://ieeexplore.ieee.org/search/>
- [12] Andrieux, A, Czajkowski C, Dan A, Keahey K, Ludwig H, Pruyne J, Rofrano J, Tuecke S, Xu M. Web service agreement specification (WS-agreement). Version 1.1, 2004.
- [13] Al-Ali RJ, Rana O, Walker D, Jha S, Sohail S. G-QoS: Grid service discovery using QoS properties. *Computing and Informatics Journal. Special Issue on Grid Computing*, 2002,21(4):363–382.
- [14] Ludwig H, Dan A, Kearney R. Cremona: An architecture and library for creation and monitoring of WS-agreements. In: Aiello M, Aoyama M, Curbera M, *et al.* eds. *Proc. of the 2nd Int'l Conf. on Service Oriented Computing (ICSOC 2004)*. New York: ACM Press, 2004. 65–74.
- [15] He C, Gu L, Du B, Huang ZC, Li SL. A WSLA-based monitoring system for grid service—GSMon. In: *Proc. of the 2004 IEEE Int'l Conf. on Service Computing (SCC 2004)*. 2004. <http://ieeexplore.ieee.org/search/freeresult.jsp?history=yes&que>
- [16] Jin JW, Nahrstedt K. Classification and comparison of QoS specification languages for distributed multimedia applications [R/OL]. http://uther.dlib.vt.edu/~ncstrlh/cgi-bin/OAINCSTRL_union/UI/show_full_metadata.pl?identifier=oai:ncstrl.uiuc.EDS:ncstrl.uiuc_cs/UIUCDCS-R-2002-2302
- [17] Czajkowski K, Foster I, Kesselman C, Martin S, Smith W, Tuecke S. A resource management architecture for Metacomputing system. In: *Proc. of the 4th Workshop on Job Scheduling Strategies for Parallel Processing*. 1998. 62–82. <http://citeseer.ist.psu.edu/czajkowski97resource.html>
- [18] Hu CM, Huai JP, Sun HL. Web services-based grid architecture and its supporting environment. *Journal of Software*, 2004,15(7): 1064–1073 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/15/1064.htm>
- [19] Casanova H, Obertelli G, Berman F, *et al.* The AppLeS parameter sweep template: User-Level middleware for the grid. In: *Proc. of the Supercomputing 2000*. 2000. <http://citeseer.ist.psu.edu/casanova00apples.html>
- [20] Nahrstedt K, Chu H, Narayan S. QoS-Aware resource management for distributed multimedia applications. *Journal on High-Speed Networking*, IOS Press, 1998. <http://citeseer.ist.psu.edu/nahrstedt98qosaware.html>

附中文参考文献:

- [9] 徐明伟,胡春明,刘旭东,马殿富.一种基于 Web Service 的分级 QoS 的研究与实现. *计算机研究与发展*,2005,42(4):669–675.
- [18] 胡春明,怀进鹏,孙海龙.基于 Web 服务的网格体系结构与支撑系统研究. *软件学报*,2004,15(7):1064–1073. <http://www.jos.org.cn/1000-9825/15/1064.htm>



胡春明(1977 -),男,江西泰和人,博士,主要研究领域为分布式计算,中间件,网格技术,网格 QoS.



沃天宇(1978 -),男,博士生,主要研究领域为分布式计算,网格技术.



怀进鹏(1962 -),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为计算机软件与理论,中间件,网络信息安全,网格技术.



雷磊(1978 -),男,博士生,主要研究领域为分布式计算,网格技术.