

一种网格数据库事务模型的研究^{*})

马新娟^{1,2} 李陶深^{1,3} 李卫玲^{1,4}

(广西大学计算机与电子信息学院 南宁 530004)¹ (山东理工大学计算机科学与技术学院 淄博 255049)²

(中南大学信息科学与工程学院 长沙 410083)³ (洛阳师范学院计算机科学系 洛阳 471022)⁴

摘要 网格事务的管理更是网格数据库管理系统(GDBMS)的关键技术之一。网格环境下对数据库的访问不仅要求短生命周期的事务,而且也要求支持长事务。本文提出了一个适于网格环境下的数据库事务模型,仿真实验证明该模型的设计思想是可行的。

关键词 网格,事务模型,网格数据库服务,网格数据库管理系统

1 引言

开放网格服务体系结构(Open Grid Services Architecture, OGSA)^[1]和 Web 服务资源框架(Web Service Resource Framework, WSRF)推动着网格技术从科学计算走向商业应用,网格的核心是大规模地共享计算机资源,完成协作性的任务^[2]。OGSA 最突出的思想是把网格中一切异构的资源表示成面向服务架构(Service Oriented Architecture, SOA)的服务,这样有利于通过统一的标准接口来管理和使用网格。网格需要在动态的环境下动态灵活地组合分布式资源和服务,与传统的分布式和并行系统相比,网格是一个松散耦合、异构和动态的环境。

目前网格服务的研究主要集中在计算服务与协同服务等领域,而在数据服务领域还没有深入的研究,尤其在网格数据库事务模型的研究领域。由于网格具有并行性、动态性、自治性以及异构性等特点,传统分布式事务都不能直接用于网格中,因为它们要么不能有效地支持长事务,要么不能满足网格的动态特性^[3]。因此,我们提出了一种新的数据库事务模型,它依据 SOA 原理以及服务资源的概念来定义底层数据库。该模型不仅能够满足网格环境中对短时原子操作的要求,而且也能满足涉及海量数据的长时间协作的事务要求。它支持网格动态性和可扩展性。

2 网格环境下的事务模型的设计

2.1 设计思想

本文提出的事务模型是建立在 OGSA 的基础上的。OGSA^[4]为网格服务定义了标准的接口和行为规范。接口解决服务发现、动态服务创建、服务生

命周期管理、通知和变化管理以及鉴别和授权等问题;行为规范解决用户如何与一个网格服务交互,如何实现可靠的调用以及如何管理网格服务等。网格服务是指一个自包含、自描述、遵守 OGSA 接口和规范的 Web 服务。OGSA 将任何松散耦合的网络实体都抽象为服务,包括计算机、程序、数据、仪器设备等,网格服务之间通过消息来传递信息。

分布式的数据库应用层访问接口不具有网格服务动态搜索、动态创建等网格服务特性。它们只能在同构系统中、在特定的编程环境和操作系统平台中运行。基于 OGSA,我们将网格环境中的数据库封装为网格数据库服务(GDS),它可以采用介于用户与 DBMS 之间的中间件作为访问接口。数据库服务接口通过不同的数据库访问引擎(如 JDBC, ODBC 等)与相应的底层物理数据库进行连接,客户端在网格环境中,通过调用基于 OGSA 体系结构的网格中间件服务,与数据库服务访问接口进行连接和数据交换^[5]。其结构如图 1 所示。

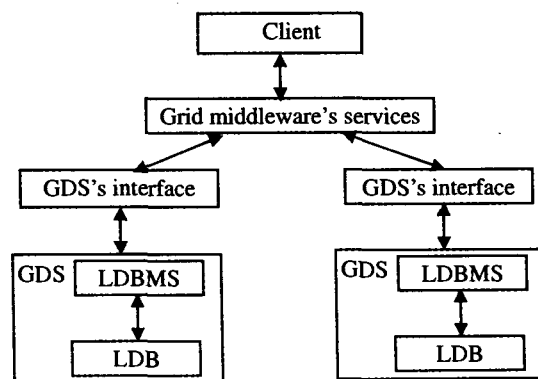


图 1 网格数据库服务访问结构

模型所采用的网格数据库体系结构如图 2 所示,它是一个四层的体系结构,即用户层、应用层、网

^{*} 本文得到广西“新世纪十百千人才工程”专项基金项目(桂人字 2001213)号和广西自然科学基金项目(桂科自 06400026)联合资助。

格服务层和数据层,它们分别在网格事务设计中担任不同的角色。

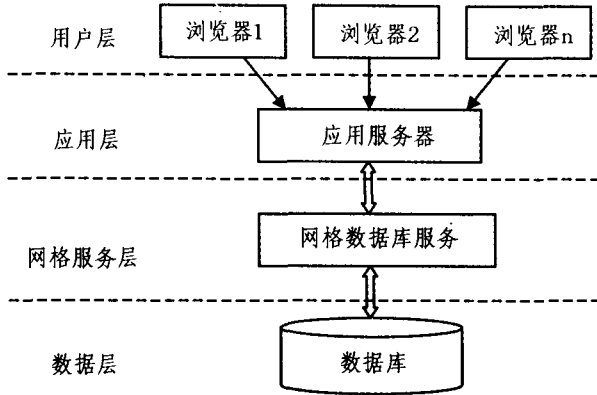


图2 网络数据库体系结构

2.2 基本概念定义

根据网格环境下访问数据库的特点,要求其事务管理机制必须支持长事务的并发执行,并且满足网格的动态过程和用户控制,最大限度地降低局部事务的失败对全局事务的影响。结合分布式嵌套事务模型的描述^[6],网格环境下相关事务的定义如下:

定义1(协调者事务 Agent (C-A)) 定义为五元组 $(C_A_ID, P_A_ID, OP, C_AC, C_AS)$, 其中 C_A_ID 为协调者事务 Agent 的标识; P_A_ID 为组成 C_A 的子事务 Agent 组标识; OP 为协调者事务 Agent 所执行的操作集; C_AC 为组成 C_A 的 P_A 的依赖关系; C_AS 为 C_A 的当前状态集。协调者一般作为父事务。

定义2(参加者事务 Agent (P-A)) 定义为五元组 $(P_A_ID, F_A_ID, OP, P_AC, P_AS)$ 。其中 P_A_ID 为参加者 Agent 的事务标识; F_A_ID 为 P_A 的父事务 Agent (F_A) 标识, 它可以是参加者事务 Agent, 也可以是协调者事务 Agent; OP 为参与者事务 Agent 所执行的操作集; P_AC 为参加者事务 Agent 的约束规则集; P_AS 是事务 Agent 的状态集。能根据子事务自身的特性, 决定是否进行进一步的分解。它可以是父事务, 也可以是子事务, 最底层的参加者为原子事务。

定义3(补偿事务) 一个事务提交后, 可以从语义上“抹掉”其对数据库的影响的事务即为补偿事务。

2.3 事务模型

移动 Agent 是位于网络中并通过迁移或服务接口能与网络中其它程序进行通信的 Agent。由于它具有自治性、交互性、协作性、通信性、移动性等优点, 十分适合于建立大规模复杂的网格事务模型。因此, 结合网格环境的特点, 我们在网格事务模型的研究中采用移动 Agent 技术来实现网格事务, 要求网格中所有的数据资源使用统一的服务接口, 而且

不需要根据事务的不同而改变, 如图3所示。事务模型的形式化描述如下:

事务结构说明: (其中 $\langle operation \rangle$ 为操作对象集合, $\langle constraints \rangle$ 为事务依赖集合, $\langle state \rangle$ 为状态集合)

```

 $\langle C\_A \rangle ::= \langle P\_Ai \rangle +$ 
 $\langle C\_A \rangle ::= \{ C\_A\_ID, P\_Ai\_ID, \langle operation \rangle, \langle constraints \rangle, \langle state \rangle \}$ 
 $\langle P\_Ai \rangle ::= \{ P\_Ai\_ID, F\_A\_ID, \langle operation \rangle, \langle constraints \rangle, \langle state \rangle \}$ 
 $\langle operation \rangle ::= \langle Transaction\_OP \rangle | \langle Application\_OP \rangle$ 
 $\langle Transaction\_OP \rangle ::= BEGIN | COMMIT | ABORT | CANCEL$ 
 $\langle Application\_OP \rangle ::= UPDATE | INSERT | DELETE | QUERY$ 
 $\langle constraints \rangle ::= \langle commit\ rely \rangle \wedge \langle abort\ rely \rangle \wedge \langle other\ rules \rangle$ 
 $\langle state \rangle ::= NOT\_EXECUTED | BEING\_EXECUTED | CHECKPOINT | BEING\_COMMITTED | COMMITTED | BEING\_ABORTED | ABORTED | BEING\_COMPENSATED | COMPENSATED | BEING\_RECOVERY$ 

```

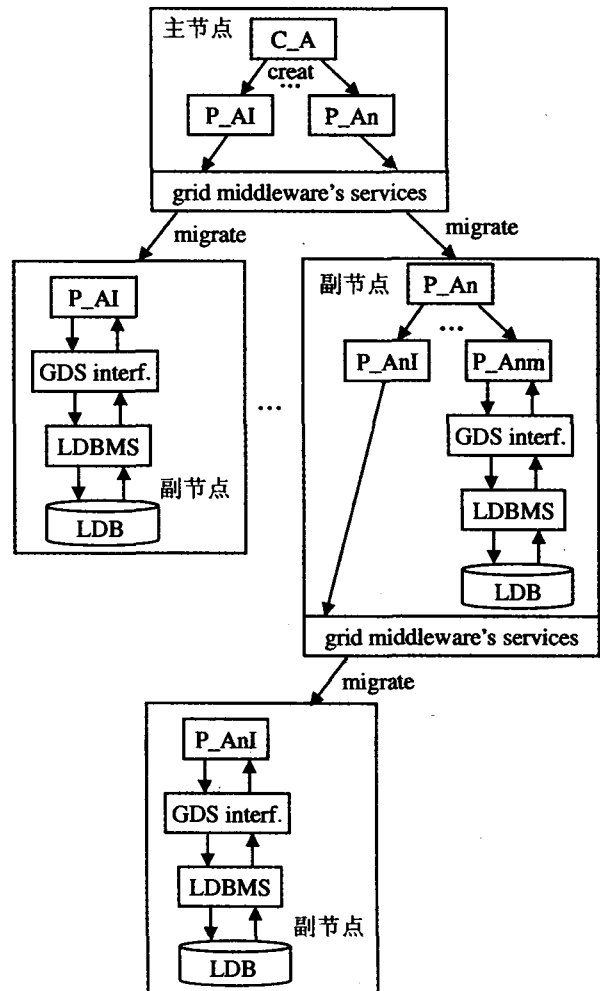


图3 基于网格服务的数据库事务模型

3 模型的仿真实现

在开发实现系统时, 我们选择 Java 作为主要的开发平台, MS SQL Server 2000 作为数据库服务器, 通过建立 JDBC-ODBC 桥实现对数据库的访问。

3.1 仿真实验方案

我们将网格数据库封装为网格数据库服务(GDS),发布并注册到专门的网格服务中心。网格事务基于服务的方式对数据库进行访问。

数据库服务接口通过 Java 数据库访问引擎 JDBC 与相应的底层物理数据库进行连接。网格数据库服务既可以是临时的,也可以是永久的。临时服务的实例是在客户请求时动态创建的,它的生命周期可由用户指定。为了适应临时网格数据库服务的特点,我们将数据库的分布信息存放在一个信息库 gdsinfo. dbc 中,在信息库相应的数据源后加一个时间段变量 timeofuse,采用设置终止时间来模拟数据库服务的动态性。在调用过程中,用户可以通过不断发送“keepalive”信息来延长服务的时间,可以发送“destroy”信息提前销毁它。流程图如图 4 所示。

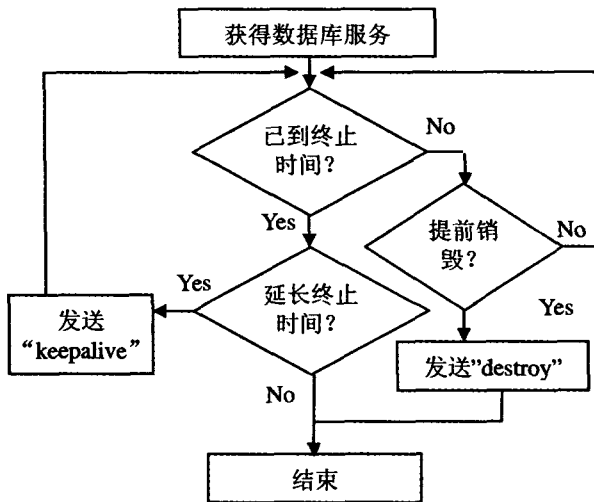


图 4 动态控制时间流程图

我们将事务 Agent 定义为 Agent 类,主要有三部分组成:接口定义、代理编写、功能实现。事务 Agent 根据服务查询模块反馈的信息迁移到合适的节点,执行操作。具体内容在此就不详细描述了。

在网格事务模拟实验中,主要测试进行大规模访问时网格服务器端执行的情况。由于任务的完成时间也是一个很重要的指标,因此我们分别取不同的并发用户个数,来记录任务开始和任务结束的时间,最后算出任务的平均完成时间,由此可测试出服

务器端的性能。实验模型参数的设置如表 1 所示。

表 1 实验模型的参数

参数	参数意义	值
AR	事务到达率	5 事务 /timeunit
NumDB	数据库数量	2
TypeTc	事务类型	网格嵌套事务
CC	嵌套层次	2

3.2 仿真实验结果与分析

实验结果如表 2 所示。通过分析表中的数据可知,在并发用户数增加的同时,任务完成时间也逐渐增加。这是由于随着并发用户的增加,总的事务的通信时间以及事务的处理时间等都会相应增加。但任务完成时间的增加并不随用户个数成比例增加,而是相对要少,这说明测试的数据在合理的范围之内,对并发用户的控制是比较有效的。

表 2 实验结果

用户个数	3	10	20	100
任务完成时间	534.2	862.2	1448.6	5594

结束语 本文分析了基于 SOA 架构的开放网格体系结构,提出了一种网格服务环境下的数据库事务模型。实验证明,此模型的设计思想是可行的,为网格环境下数据库事务管理机制相关技术的研究提供了重要的技术基础。当然,本文提出的事务模型只是初步探讨,它还需要进一步的完善。由于网格环境的复杂特点,寻找一种即简便又实用的事务模型是今后的主要研究工作。

参 考 文 献

- 1 Foster I, Kesselman C. The Grid: Blueprint for a New Computing Infrastructure (Second Edition) [M]. China Machine Press, 2005
- 2 Stockinger Hr. Distributed Database Management Systems and the Data Grid". In: 18th IEEE Symposium on Mass Storage Systems'01
- 3 唐飞龙,李明禄,曹健. 网络环境下的一种事务协调机制及其补偿技术. 计算机研究与发展[J], 2003, 40(12): 1776~1800
- 4 OGSA. <http://www.gridforum.org/ogsi-wg/drafts/GS-Spec-draft03-2002-07-17.pdf>
- 5 曾怡,张万军,杨进,李国庆. 基于 OGSA 的数据库网格化封装方法. 华中科技大学学报(自然科学版)[J], 2005, 33(1): 31~33
- 6 刘云生,李国徽. 实时数据库系统中的嵌套事务. JOURNAL OF SOFTWARE[J], 1999, 5(10): 552~5567