

# 基于 MSA 的 SMIM 实现机制

高志鹏, 孟洛明, 李文璟, 邱雪松

(北京邮电大学网络与交换技术国家重点实验室, 北京 100876)

**摘 要:** 提出基于元数据存储访问的共享管理信息模型实现机制, 以解决异构系统互操作问题, 保证数据一致性。该实现机制基于已有异构系统和已有数据, 使用分层的逻辑划分, 将实现过程划分为数据应用层、数据访问适配层、数据访问代理层、元数据存储层及数据存储层 5 个层次, 并介绍了各个层次的功能和相关算法。

**关键词:** 元数据存储访问; 共享管理信息模型; 数据一致性; 网络管理

## MSA-based Realization Mechanism for SMIM

GAO Zhi-peng, MENG Luo-ming, LI Wen-jing, QIU Xue-song

(State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876)

**【Abstract】**A Metadata Storage Access (MSA)-based realization mechanism for Shared Management Information Model (SMIM) is given to ensure the interoperation between heterogeneous systems, and the data consistency. The mechanism is based on the heterogeneous systems and the data existed. It divides the realization process into five logical layers: data application layer, data access adapter layer, data access agent layer, metadata storage layer and data storage layer. The introduction of the function and the relative algorithm for each layer are given.

**【Key words】** Metadata Storage Access(MSA); Shared Management Information Model(SMIM); data consistency; network management

随着网络新技术的不断出现, 网络结构日益复杂, 各种业务网络日渐繁多, 再加上网络运营面向业务的管理需求, 运营支撑系统(OSS)体系越来越庞大。各OSS系统存在对同一被管对象的重复建模, 这使得同一被管对象在不同OSS系统中的表述方式、描述粒度、属性类型不尽相同, 最终导致信息无法在各系统间共享<sup>[1]</sup>, 无法保证系统间的协同工作和互操作。OSS系统间信息共享的两个要素是建立共享管理信息模型和模型实现机制。共享管理信息模型的建立是实现信息共享的基础, 实现机制是模型应用于OSS的保证。电信管理论坛(TMF)定义了共享信息/数据(SID)模型<sup>[2-3]</sup>, 建议在统一的信息模型描述框架下构建OSS系统, 实现电信运营的数据集成, 支持OSS系统之间互操作, 保证数据的一致性。文献[4]给出了SID在告警相关性分析中的应用。但是TMF并没有给出使用SID实现信息系统数据无缝集成的机制和方法。本文根据共享管理信息模型(SMIM)的特点和SID建模思想, 综合考虑当前异构系统并存及大量业务数据积累现状, 提出了基于元数据存储访问(MSA)的SMIM实现机制, 从而解决了SID在实现机制方面的不足, 保证了SMIM的可行性。

### 1 基于元数据存储访问的 SMIM 实现机制

#### 1.1 SMIM 简介

SMIM是对被管对象抽象信息的模型描述。SMIM着眼于全程全网的信息建模, 由管理需求驱动, 采用SID自顶而下和面向对象的建模方法学, 通过数据抽象、模型表示和数据挖掘 3 个步骤<sup>[5]</sup>, 建立涵盖各专业网络(网络结构、网元信息等)、网络管理(告警信息、配置信息等)、企业运营(人员管理、市场销售等)等领域的信息模型。SMIM具有建模领域全面性、模型数据一致性、模型元素相关性及实现方式无关性的特点。

当前, 网络管理领域对 SMIM 的研究集中在两个方面: SMIM 本身的建模理论和方法研究, 以及 SMIM 的实现机制。

本文是对 SMIM 实现机制的研究。

#### 1.2 基于元数据存储访问的 SMIM 实现机制

本文提出一个基于 MSA 的 SMIM 分层实现机制, 如图 1 所示。数据应用层发出的数据访问请求命令, 经过数据访问适配层的标准化适配, 使用数据访问代理, 对数据存储层和元数据存储层的数据进行操作。元数据存储层存储了数据存储层数据的元数据。下面分节对各层进行说明。

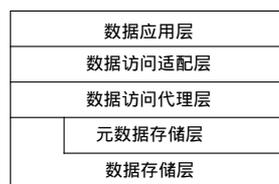


图 1 基于 MSA 的 SMIM 分层实现机制

##### 1.2.1 数据应用层

数据应用层为数据的各种应用的集合, 包括网络管理系统、计费管理系统、电子运维系统、客户管理系统等各种系统, 以及数据采集、通知上报等各种进程。数据应用使用定制采集任务、计算生成、人工输入或批量数据导入的方式, 经过数据访问适配, 通过代理将数据存入数据存储层, 并在元数据存储层记录其元数据; 同时通过适配层和代理层读取相关数据, 对其进行加工、转换、映射、合成等处理, 以需

**基金项目:** 国家自然科学基金资助重大项目(90204002, 90604020); 长江学者和创新团队发展计划基金资助项目(IRT0410); 高校博士点专项科研基金资助项目(20040013002)

**作者简介:** 高志鹏(1980 -), 男, 博士研究生, 主研方向: 网络管理与通信软件; 孟洛明, 教授、博士生导师; 李文璟, 副教授; 邱雪松, 教授

**收稿日期:** 2007-01-20 **E-mail:** zpgao@metarnet.com

要的形式呈现给用户。数据应用层是数据访问请求的原始发起者和数据访问响应的最终消费者。

### 1.2.2 数据访问适配层

数据访问适配层负责对数据访问请求及其响应数据的标准化适配,将异构数据格式统一为标准的数据格式,并对标准化了的数据访问请求做一定的逻辑处理。数据访问适配层提供3个逻辑功能:数据格式一致性约束,应用接口适配,以及数据访问请求处理。

### 1.2.3 数据访问代理层

数据访问代理层由大量的分布式智能代理组成,负责与元数据存储层和数据存储层的数据通话。其内部组织如图2所示。

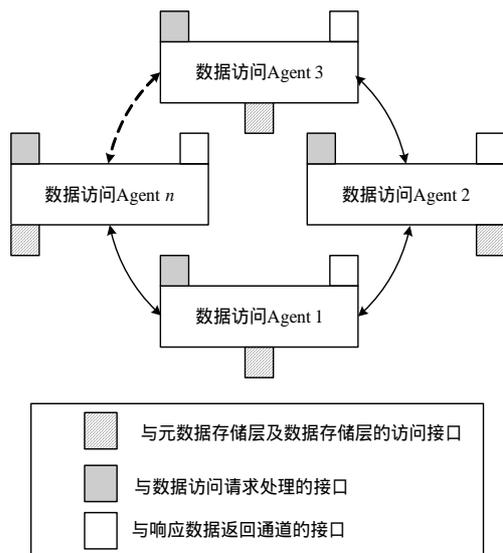


图2 数据访问代理层内部组织

每个Agent有3个层外接口,与数据访问请求处理、存储层及相应数据返回通道相接,分别交互控制信息、访问信息及批量数据。数据访问代理层内部采用环状通信结构,每个Agent只与其相邻的两个代理通信,通信目的在于维护Agent拓扑结构,并保存相邻Agent的任务信息。下面给出实现Agent失败接管的算法。

一旦某个Agent进程死掉,其相邻两个Agent会分别/同时发现此问题。假设问题Agent为Agent2,其相邻Agent分别为Agent1和Agent3。

当Agent1首先发现后,立即将此情况上报数据访问适配层进行数据访问请求处理,同时将Agent2的任务信息上载,并删除自身存储的Agent2的任务信息。数据访问适配层收到信息后立即通知Agent3,同时重新进行任务分配,将Agent2的任务信息分配至其他空闲Agent。Agent3收到通知后,按照数据访问请求处理的要求,将Agent2的访问信息删除,并主动向Agent1发起连接请求。随后Agent1与Agent3建立通信连接,并各自备份对方的任务信息。

当Agent1和Agent3同时发现Agent2的问题后,同时将此情况上报数据访问适配层进行数据访问请求处理。如此情况被数据访问请求处理逻辑并行接收,则数据访问请求处理逻辑根据具体的实现算法,发送通知给其中的某一Agent(假设为Agent1),然后由Agent1主动与Agent3建立连接,互相备份任务信息;如此情况被数据访问请求处理逻辑串行接收,则数据访问请求处理先后下发两道通知(如先发给Agent1,后

发给Agent3),Agent1收到通知后主动与Agent3建立连接。随后Agent3收到与Agent1建立连接的通知后,发现自己正在或者已经与Agent1建立了连接,则丢弃该通知。

如果当3个或3个以上的串连Agent同时发生故障死掉,则丢弃非两端的Agent访问数据,按照一定的算法重新建立活跃Agent之间的连接,并返回故障Agent的数据访问超时信息。Agent通过访问接口可以访问元数据存储层及数据存储层。

### 1.2.4 元数据存储层及数据存储层

元数据存储层及数据存储层都能够被Agent通过访问协议进行操作。元数据存储层存储了共享信息模型及其与数据存储层之间的映射关系。数据存储层存储真实的网络、业务数据。在基于MDA的SMIM实现机制中,Agent一般需先访问元数据存储层,获取要访问的真实数据的逻辑及物理地址、访问权限等重要信息,然后利用此信息对数据访问存储层相应数据进行访问。

元数据与数据的映射关系分为4类:名词映射(Term Mapping, TM),位置映射(Location Mapping, LM),操作映射(Operation Mapping, OM)及鉴权映射(Authentication Mapping, AM)。4种映射关系分别对应了元数据存储层的TM数据、LM数据、OM数据及AM数据等4类数据。Agent在这4类映射数据的作用下对存储数据的访问过程如图3所示。

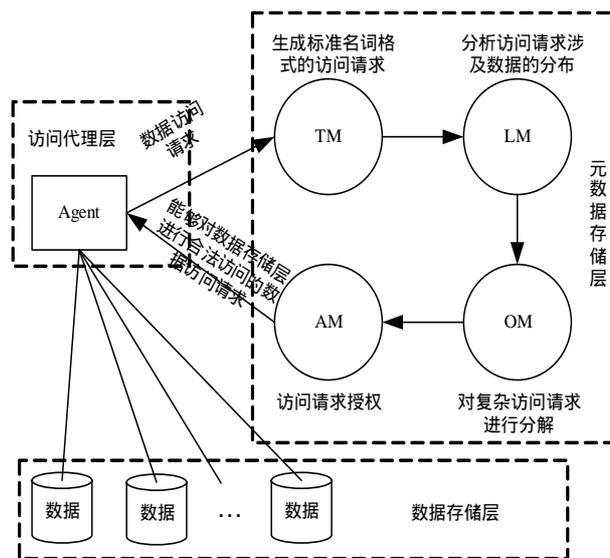


图3 Agent在映射数据作用下对存储数据的访问

TM是SMIM的元素名词与异构模型系统中的名词之间的映射关系,是实现系统之间互操作的语义基础。TM数据有3种情况:(1)名词之间的一一对应,或异构模型与SMIM模型中使用同样的名词,如各个模型都使用名词“Switch”表示交换机;(2)不同模型中的不同名词对应SMIM模型中的一个名词,如SMIM使用名词“device”表示设备,而其他系统可能使用“equipment”或“NE”来表示;(3)不同模型中的同一名词,对应SMIM模型中的不同名词,如在某些系统中,由于建模的不全面性,经常会出现将概念缩水的情况,例如一个软件资源系统建模中“firewall”指的是软件防火墙“softfirewall”,而一个硬件资源系统建模中“firewall”则指的是硬件防火墙“hardfirewall”。前两种情况的出现是由于异构系统的信息模型缺乏统一命名规则,情况(3)是由模型应用领域局限性而导致出现的。

LM 是对访问目的数据的物理、逻辑地址的描述。LM 的使用可以使用户透明地访问自己想得到的数据，而不必关心此数据是否来源于不同的系统、来源于哪些系统。如用户发起的一个数据读取操作，通过 Agent 访问元数据存储层的 LM 数据，发现此访问请求涉及物理上分布在两个不同数据库、逻辑上属于不同业务系统的信息。Agent 得到这个信息后，根据 LM 数据的指示去两个不同的数据库中分别读取相关数据，然后合成为用户需要的数据，反馈给用户。

OM 是对一个数据访问操作的分解，将一个复杂的数据访问请求分解为若干个简单的原子操作。OM 的使用需要配合 LM 的指示。如当用户发起的数据请求格式是一个屏蔽真实数据存储细节的 SQL 标准语句，但是 Agent 在获取 LM 数据后发现要访问的数据分布在若干个不同的物理数据库上，此 SQL 语句无法直接执行，需要将此 SQL 语句分解成若干个相应的 SQL 语句。分解的过程需要有 OM 数据的参与。

AM 保存了对真实数据库访问鉴权的相关数据，将真实数据库、表、视图的操作权限与用户通过 Agent 对数据进行操作的权限进行映射。AM 需要与 LM、OM 配合使用。用户一个简单的数据查询命令，有可能对应多个库、表进行读、写等不同操作命令，因此需要 AM 数据对用户屏蔽底层数据库的鉴权多样化。

## 2 系统的设计与应用

目前，笔者已经将本文提出的基于 MSA 的 SMIM 实现机制的思路应用到某原型系统开发过程中。原型系统的设计按照本文提出的分层实现的思想，将原型系统划分为统一用户界面、数据访问代理、元数据存储控制、数据存储等 4 个模块。

### 2.1 系统结构设计

原型系统采用用户界面(对应数据应用层)、应用层(对应数据适配层和数据访问代理层)和模拟数据层(对应元数据存储层和数据存储层)的 3 层体系结构，如图 4 所示。

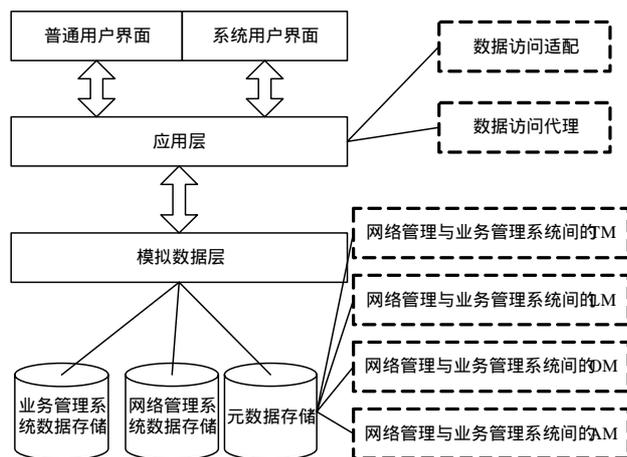


图 4 原型系统的系统结构

为了模拟多信息系统在基于 MDA 的 SMIM 实现机制下的应用效果，笔者选择了两个应用系统：传输网网络管理系统和业务管理系统。界面将用户的界面操作转化为数据操作命令，下发给应用层，应用层采用基于 MDA 的 SMIM 实现机制，与模拟数据库配合对数据进行操作。

原型系统的关键设计包括：

(1)多用户并发数据访问场景仿真及并发控制算法。为切合真实的应用场景，应模拟几十个用户对同一数据的并发操

作，在应用层使用并发控制算法，检验本机制在多用户使用环境下的系统健壮性和数据一致性。本原型系统采用基于加锁协议的并发控制算法。

(2)空闲 Agent 发现及任务分配算法。在对 Agent 分配任务及对任务失败 Agent 进行失败接管的过程中，发现并将任务分配给具有空闲资源的 Agent 至关重要。本系统采用空闲 Agent 注册的方法，建立空闲 Agent 的联系清单，以分配任务。

(3)网管系统信息建模及业务系统信息建模。从网络管理和业务管理的不同角度，对网络、业务信息建模，并在两个模型间建立联系，如网管系统发现设备故障导致连接失败，或带来 QoS 下降，则业务系统应有相应的业务故障预警。

(4)元数据内容及其访问策略。元数据内容是实现机制中最核心的内容，应具有 TM、LM、OM、AM 4 种数据。元数据内容是建立在信息模型的基础上的。另外，由于元数据内容较多，且无法直接以此作为访问网络、业务数据的索引，需要做一定的处理，因此需要使用访问策略合理地挑选、处理元数据。

### 2.2 系统功能

原型系统实现并验证了如下主要功能：

(1)多用户在异构系统中对同一数据的并发读写操作，如数据查询和数据记录；

(2)支持异构系统中具有不同数据操作权限的用户根据自己的权限，对非权限允许范围内数据进行透明的访问，如网络管理系统的用户在需要业务系统数据配合的情况下，可以在后台应用中对业务管理系统的数据进行访问；

(3)支持数据操作的异常处理，并通过失败接管，将异常的数据操作过程重新激活；

(4)通过对元数据的操作，支持跨库、跨系统的数据操作，包括联合查询、联合处理等；

(5)支持系统处理细节记录，通过日志的形式，记录一个数据操作从请求、处理到响应的全部细节，粒度精确到函数，便于对后台应用及算法进行验证。

## 3 结束语

本文提出了基于 MSA 的 SMIM 实现机制，通过数据共享解决系统间互操作的问题，实现异构信息系统集成，保证数据的一致性。本机制最大的特点在于其实现基于已有系统、数据，可以在异构系统并存的情况下实现数据集成，并在访问控制、数据存储等方面具有一定的优势。下一步将完成基于此实现机制的原型系统，并具体解决数据一致性算法、访问鉴权机制、模型进化方法等具体实现理论与技术。

### 参考文献

- [1] John S. DEN-ng: Achieving Business-driven Network Management[C]//Proc. of NMOS'02. Washington D. C., USA: [s. n.], 2002.
- [2] TMF. GB922-2005 SID Business View: Concepts, Principles, and Domains[S]. 2005.
- [3] TMF. GB926-2004 SID System View: Concepts and Principles[S]. 2004.
- [4] 魏丽红, 简思涛, 丁涌, 等. 资源管理 SID 建模方法及在网络管理系统中的应用[J]. 北京邮电大学学报, 2004, 27(6): 90-96.
- [5] 高志鹏, 邱雪松, 芮兰兰. 新的基于 SID 的管理信息建模方法[J]. 北京邮电大学学报, 2006, 29(增刊): 118-121, 178.