

# 达梦数据库 Web 服务技术研究<sup>\*</sup>

朱虹 郭帆 冯玉才

(华中科技大学计算机系数据库与多媒体研究所 武汉 430074)

**摘要** 本文提出了一种将数据库同 Web 服务技术相结合的方法,一方面使得数据库可以将其存储的数据以 Web 服务的形式提供给外部使用,另一方面也可以在数据库内部直接使用 SQL 语句或存储过程调用外部的 Web 服务对数据进行处理或取得数据,该方法具有使得数据库能够不依赖应用程序而直接同 Web 服务进行交互的特点。

**关键词** Web 服务,数据库,面向服务体系结构

## DM Database Management System Web Service Research

ZHU Hong GUO Fan FENG Yu-Cai

(Database and Multimedia Technology Research Institution of Computer Department,

Huazhong University of Science and Technology, Wuhan 430074)

**Abstract** In this paper, we proposed a method to integrate Database management system with Web service, so that database management system can provide its data to other Web applications in the form of web service, it also can invoke other web services in itself with SQL statement or stored procedure. This method make database can interact with Web service directly, do not dependent on application.

**Keywords** Web service, Database management system, Service oriented architecture

## 1 引言

Web 服务技术已经在全世界启动了一波 IT 业的浪潮,基于 Web 服务的应用程序具有松散耦合、面向组件和跨技术实现的特点<sup>[10]</sup>。而数据库作为最主要的数据存储和管理的载体,一直只是作为应用程序的数据提供者而已。随着数据库管理系统的功能越来越强大,各大数据库厂商和研究人员都开始致力于怎样使得数据库管理系统跟上技术发展的步伐,使得数据库可以直接成为面向服务体系结构的组成部分。Bala Iyer, Sharad Mehrotra 提出了一种将数据库作为一种服务提供给外部使用的思想,并实现了一个可以通过 Web 直接访问数据库的应用<sup>[1]</sup>。但是它实现的是将数据库的功能作为服务提供给外部使用,没有解决将数据库中的数据作为服务提供给外部应用的问题,更没有实现在数据库中直接调用外部 Web 服务的功能。随后, DB2 和 Oracle 都在数据库内部实现了将访问数据的 SQL 语句和存储过程包装为 Web 服务的能力,同时提供在数据库内部直接访问外部 Web 服务的能力<sup>[6,7]</sup>。

本文从面向服务的角度,以国产数据库管理系统 DM5 为平台,探讨了利用 Web 服务技术将数据库推向应用的前台的技术,使得数据库独立于应用程序可以将其 sql 语句或存储过程发布为 Web 服务,另一方面扩展了数据库 sql 语句直接来调用外部 Web 服务。

## 2 数据库提供 Web 服务

数据库提供 Web 服务相当于把数据库放到 Web 服务体系结构中的服务提供者的位置。数据库可以将其操作的 SQL 语句或存储过程发布为 Web 服务,从而将数据提供给外

部客户使用;当收到客户调用时,将 SQL 语句或存储过程转换为对数据库的实际操作,并将结果返回给客户。

要使得数据库能够提供 Web 服务,我们需要解决以下几个问题:1. 如何使用户可以方便地将他们要发布为 Web 服务的 SQL 语句或者存储过程表示出来;2. 如何解析用户发送过来的 SOAP 请求,并将该请求映射为对数据库进行操作的 SQL 语句或存储过程调用;3. 如何同数据库建立连接;4. 从数据库取得数据后,如何将数据封装为 SOAP 消息返回给用户。

### 2.1 SQL 语句或存储过程的部署

为了方便用户将 SQL 语句或存储过程发布为 Web 服务,我们给出如下定义。

**定义 1(DBWS 文件)** 一个 DBWS(Database Web Service)文件是一个用户将其要发布的 SQL 语句或存储过程按照达梦数据库 Web 服务规则编写的 XML 文档。主要的规则有:

〈query〉用来标识一个 select 操作的 SQL 语句;

〈update〉用来标识一个 update、delete、insert 类型操作的 SQL 语句;

〈call〉用来标识要调用的一个存储过程;

〈parameter〉它用来标识输入输出参数,以及参数类型信息。

**定义 2(服务组)** 一个服务组是对同一数据库进行操作的 DBWS 文件的集合。

**定义 3(数据库配置文件)** 数据库配置文件提供了访问某服务组下数据库的用户名、密码、JDBC 驱动程序位置等必要信息。

**定义 4(部署描述符)** 一个部署描述符描述了用户要发

<sup>\*</sup> )十五 863 数据库专项“大型通用数据库管理系统 DM V5.6 的研究开发及其应用”(2005AA423030)。朱虹 副博士生导师,博士,主要从事数据库理论与安全方面的研究工作。郭帆 硕士研究生,主要从事数据库技术的研究工作。冯玉才 博士生导师,博士,主要从事数据库理论与技术的研究工作。

布为 Web 服务的某个服务组下的一个 DBWS 文件中的操作和该操作的属性信息,包括服务的名称、DBWS 文件的名称、DBWS 文件所在组名称以及该服务提供的方法等。

在一个服务组中,至少包含一个 DBWS 文件和对该服务组的数据库进行连接的数据库配置文件。

用户写好数据库配置文件,并将要发布为 Web 服务的对该数据库进行操作的 DBWS 文件和服务组建立好之后,最后就是要将定义好的服务部署到数据库 Web 服务(DBWS)引擎上,这样 DBWS 引擎才能根据服务和操作名找到对应的对数据库的操作。用户将部署描述符写好之后,调用 DBWS 引擎提供的工具将部署描述符中指定的对数据库的操作部署到 DBWS 引擎,如用户要撤销某个服务,只需要使用另一个工具即可。

### 2.2 数据库 Web 服务(DBWS)引擎

数据库 Web 服务引擎是数据库提供 Web 服务的核心。其主要功能是接收客户发送的 SOAP-RPC 请求,解析这个请求<sup>[9]</sup>,并将其定位到用户发布的某个 Web 服务上,从而取得该 Web 服务对应的 SQL 语句或存储过程;根据客户提供的参数重组这些 SQL 语句或存储过程,并向数据库发出操作请求;从数据库取得数据,根据用户对该服务的部署,将数据组织为一个有效的 SOAP 消息,返回给客户。

数据库 Web 服务引擎设计的好坏直接关系到用户发布的 Web 服务能否有效地被外部客户调用。为了使得数据库 Web 服务引擎处理流程更加清晰,为了以后能够方便地对数据库 Web 服务引擎进行扩展,我们采用了处理链模型来实现 DBWS 引擎。一个处理链由多个有序的处理部件组成,每个不同的处理部件完成某个特定的功能。

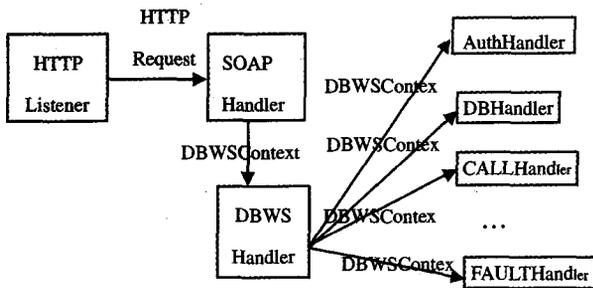


图1 DBWS引擎内部处理链图

从图1可以看出,该处理链由六个处理器部件组成。SOAPHandler 负责从 HTTP 请求中提取 SOAP 消息并将其

解析。DBWSHandler 是一个总控处理器,它负责对其它处理器进行调度。在处理器之间进行流动的是 DBWSContext, DBWSContext 结构包含本次 SOAP-RPC 请求的一些必要信息,例如 Web 服务名、操作名、服务组名等。各个处理器部件负责填充 DBWSContext 结构,或根据结构中的内容进行某些操作。

### 2.3 数据库提供 Web 服务整体框架

图2是数据库提供 Web 服务的一个整体框架,我们将 DBWS 引擎设计为一个中间件的形式,其运行并不依赖于某个特定的数据库。数据库提供 Web 服务的处理流程为:应用服务器接收客户发送过来的请求消息并判断该请求是否被数据库 Web 服务调用,如果是,则将该请求发送给 DBWS 引擎, DBWS 引擎解析该请求,提取出 SOAP 消息,从中得到数据库 Web 服务调用的一些必要信息,根据这些信息检索部署描述文件,从而找到对应的服务组,并从服务组中找到相应的 DBWS 文件,然后取出该组的数据库连接配置信息连接数据库,接着从 DBWS 文件中取出要调用的 SQL 语句或存储过程名,将其和传入的输入参数组合为新的 SQL 语句或存储过程调用,使用 JDBC 操纵数据库进行处理。处理后的结果返回给 DBWS 引擎, DBWS 引擎将结果封装为 SOAP 消息传回给客户端。

此外, DBWS 引擎还提供了部署 DBWS 文件、自动生成 Web 服务描述文档等功能。用户可以根据部署的服务生成相应的 Web 服务描述文档,然后将该文档发布到服务注册中心或客户,从而让开发人员知道如果调用该 Web 服务。

## 3 数据库消费 Web 服务

数据库消费 Web 服务相当于将数据库放在了 Web 服务体系结构的服务消费者的位置。数据库可以通过 SQL 语句或者存储过程调用外部的 Web 服务,这样用户可以直接将数据库的数据交给外部 Web 服务进行处理,或者获取外部 Web 服务的数据到数据库。

为了使数据库能够支持使用 SQL 语句或者存储过程调用外部的 Web 服务,至少有两个问题需要解决:第一, SQL 语句或存储过程如何映射为 Web 服务的调用,然后根据用户提供的参数构造一个 SOAP 消息并发送到指定的 Web 服务提供者;第二,当收到 Web 服务提供者返回的响应之后,必须解析这个响应,提取出返回的数据交给用户。

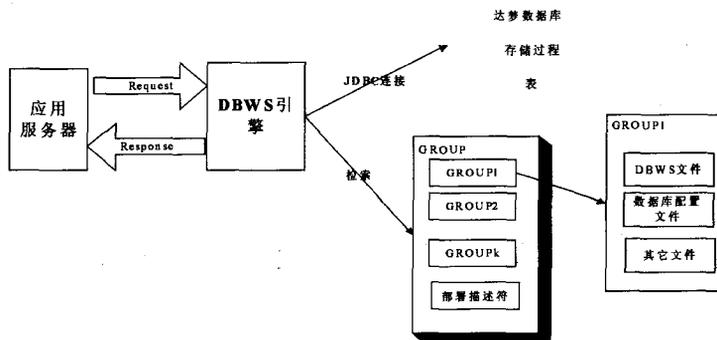


图2 数据库提供 Web 服务

### 3.1 数据库 SQL 语句扩展

为了实现数据库对 Web 服务的调用,需要扩展达梦数据库服务器的 SQL 引擎,在其中加入对 Web 服务调用支持。

调用一个 Web 服务主要需要如下信息:

- 目标对象的 URI(uniform resource identifier),它指明了 Web 服务提供者的位置。

- 要执行的操作名称
- 操作的参数信息

因此,我们增加了两个用来声明 Web 服务及 Web 服务下某个操作的 SQL 语句,另外扩展了 SELECT 语句的语法,从而可以查询或使用外部 Web 服务返回的结果。

### 3.1.1 声明 Web 服务

用户要在数据库中使用 SQL 语句来调用 Web 服务,首先需要在数据库中声明该 Web 服务的定义。

声明 Web 服务的语法为:

CREATE WEBSERVICE Web 服务的 URI Web 服务名;

### 3.1.2 声明 Web 服务下的一个操作

用户声明了一个 Web 服务之后,就要对该 Web 服务下的每个操作进行声明,告诉数据库该操作的名字是什么,输入、输出和返回参数的信息。

声明 Web 服务下操作以及该操作的参数信息的语法为:

CREATE 操作名(该方法参数信息)ON Web 服务名;

方法的参数信息需要指明:参数名 参数数据类型(整形、浮点型等)参数类型(IN,OUT 或 RETURN)。

### 3.1.3 调用 Web 服务下的操作

我们引入一个对 Web 服务进行调用的谓词 SOAP\_CALL。SOAP\_CALL 后面由用户指定要调用的 Web 服务的详细信息:Web 服务名,操作名,该操作的输入参数值。该谓词返回用户指定的 Web 服务下操作所返回的结果集。语法如下:

SOAP\_CALL(Web 服务名,操作名,参数 1,参数 2,...,参数 n);

其次,我们扩展了 SELECT 语句后的 FROM 语法,以前 FROM 谓词后面只能跟一个表或一个子查询,现在我们将其扩展为允许跟一个 SOAP\_CALL 谓词,即对 SOAP\_CALL 返回的结果集进行操作。

下面我们给出一个简单的在 SQL 语句中调用 Web 服务的例子。

例 1 有一个外部 Web 服务用来查询某个商品的价格,输入参数是商品名,返回该商品的价格。如果我们要在数据库中调用该服务,步骤如下:

步骤 1:声明一个名为 GetInfo 的 Web 服务

CREATE WEBSERVICE http://127.0.0.1/webservices GetInfo;

步骤 2:声明 GetInfo 服务下的操作 GetPrice

CREATE GetPrice(name VARCHAR(40)IN, price INTEGER RETURN)ON GetInfo;

步骤 3:调用 GetInfo 服务下的 GetPrice 操作,其输入参数为'DM'

Select \* from SOAP\_CALL (GetInfo, GetPrice, 'DM');

如果商品名为 DM 的价格为 5,则该语句显示结果为:

```
price
5
```

### 3.2 SQL 语句扩展实现

为了实现我们对 SQL 语句的扩展,我们需要对数据字典进行相应的扩展,以记录用户声明的 Web 服务以及该 Web 服务下的操作信息。

SYSWEBSERVICES:该字典表描述了 Web 服务的基本信息。

序号	字段名	类型	说明
1	name	VARCHAR	服务名
2	URI	VARCHAR	服务所在位置
3	id	INTEGER	服务号

SYSMETHODS:该字典表描述了 Web 服务下的操作的基本信息。

序号	字段名	类型	说明
1	name	VARCHAR	方法名
2	serviceid	INTEGER	该方法所属服务 id
3	id	INTEGER	方法号

SYSPARAMETERS:该字典表描述了 Web 服务下的操作的参数信息。

序号	字段名	类型	说明
1	name	VARCHAR	参数名
2	TYPE	VARCHAR	参数数据类型
3	KIND	VARCHAR	参数输入输出类型,取值为 IN,OUT,RETURN。分别代表输入参数,输出参数和返回参数
4	id	INTEGER	参数号
5	methodid	INTEGER	参数所属服务 id

对于声明 Web 服务语句,我们将从语句中分析出所要声明的 Web 服务名及该服务的 URI 信息,然后生成该服务的 id,将这些信息插入到字典表 SYSWEBSERVICES 中。

对于声明 Web 服务下的操作的语句,我们将从语句中分析出所要声明的操作名,该操作所属的 Web 服务名以及该操作的参数信息。通过 Web 服务名,我们检索字典表 SYSWEBSERVICES,可以得到该 Web 服务的 id,然后生成该操作的 id,将这些信息插入到字典表 SYSMETHODS 中。对于每一个参数,都将生成一条记录插入到字典表 SYSPARAMETERS 中,用记录中的 methodid 来标识该参数是哪个操作中的。

而调用 Web 服务的 SOAP\_CALL 谓词则是根据用户输入的信息,检索上面定义的三个字典表,首先根据 Web 服务名检索字典表 SYSWEBSERVICES,得到服务的 id 和服务的 URI;然后根据操作名和服务 id,得到操作的 id;最后根据操作 id 得到操作的所有参数信息,将用户的输入值和得到的参数信息构造一个 SOAP 消息发送给服务提供者。接收到服务提供者返回的信息之后,提取出数据信息,组织为结果集的形式返回给用户。

### 3.3 数据库消费 Web 服务整体框架

图 3 是达梦数据库消费 Web 服务的整体框架。

在图 3 中,如果 SQL 语句或存储过程向 SQL 引擎发出的请求包含外部 Web 服务调用,SQL 引擎会从中取出本次 Web 服务调用的信息,检索我们扩展的三个字典表,得到 Web 服务的定义信息,然后将这些信息组织为 SAOP 请求,再利用 Java 本地接口调用外部 Java Web 服务代理类,代理类通过 SOAP 引擎将该 SOAP 请求发送给 Web 服务提供者,提供者处理完请求后,返回一个回应消息,SOAP\_CALL 函数接收到该回应消息,从中解析出返回数据,并将其插入到结果集中返回给用户。

(下转第 131 页)

机产生的两个挑战数： $r_1(r_{11}, r_{12}, \dots, r_{1p})$ 和  $r_2(r_{21}, r_{22}, \dots, r_{2p})$ ，用户相应地构造出了两个合法的  $RIS_{part}$ 。银行可以通过如下办法把用户的身份揭示出来。

对于随机生成的 2 个  $p$  比特的随机串  $r_1$  和  $r_2$ ，其所有位完全一样的概率为  $(1-2^{-p})$ ，因此大概率上存在至少一位不相等，假设为第  $i$  位不相等，即  $r_{1i} \neq r_{2i}$ 。根据异或函数，银行可以获得用户的身份信息  $ID_U$  为  $x_{1i} \oplus x'_{2i}$  或者  $x'_{1i} \oplus x_{2i}$ 。

#### 4 安全性分析

本系统基于 RSA 盲数字签名的安全性，具有匿名性、不可伪造性、不可重复花费、不可重复存款等安全特性。

从电子支付系统的角度来看，它的安全性体现在：保证消费者的敏感数据的安全，保证电子商店的安全、支付网关的安全以及它们之间链路的安全（他们两两之间的传输数据不会被监听、篡改或伪造）。用户在取款时可能存在如下两种作弊可能。

1) 声称其所取金额（该数目将由银行从用户的账户上扣除）比实际电子现金所含的价值要小。

2) 在盲化的电子现金中包含了错误的用户身份信息。

NB\_ENVELOPE 代表用户创建的信封内所含相同电子现金的数目。如前所述，通过切割选择协议，可以以一定的概率防止这种作弊行为。当 ENVELOPE 取 10 时，银行要求用户把 9 个构建电子现金的信息揭示出来以检验用户身份，那么用户作弊成功的可能性为十分之一，如果 ENVELOPE 取 100，那么用户成功作弊的可能性只有百分之一。显然，用户

不会在小额支付上冒险。

NB\_RIS 代表电子现金中用户的随机身份串信息的配对数目。检测出用户作弊的概率跟 RIS 配对数成几何级数增长。对于两次不同的商家挑战，能够发现作弊用户的概率为两个 RIS 配对数中至少有一个位不同，即发现用户重复消费的概率为  $p=1-\frac{1}{2^n}$ 。因此，如果 NB\_RIS 取 10，则  $p_{10}=1-\frac{1}{2^{10}} \cong 0.999=99.9\%$

**总结** 本文所提出的系统基于 RSA 公钥密码体制，可以满足电子现金最基本的特性：匿名性、安全性、点对点支付、在线购物。它不仅可以直接支付电子现金给其他用户，也可以从其他用户处接收电子现金。同时，还可以支持在 Internet 上进行支付，且易于推广到基于蓝牙、SMS、非接触式智能卡等技术的系统上运行，从而更容易被大众接受。

#### 参考文献

- 1 王常吉, 裴定一, 蒋文保. 一个改进的基于限制性盲签名的电子现金系统. 电子学报, 2002, 29(7): 1083~1085
- 2 Chaum D. Blind Signature for untraceable payment. In: Advances in Cryptology - CRYPTO'82, 1983. 199~203
- 3 Chaum D, et al. Untraceable Electronic Cash. In: Advances in Cryptology - CRYPTO '88, 1989. 319~327
- 4 Manho L, et al. Design and Implementation of Revocable electronic cash system based on Elliptic Curve Discrete Logarithm Problem. Journal of communications and networks, 2002, 4(2): 81~89

(上接第 89 页)

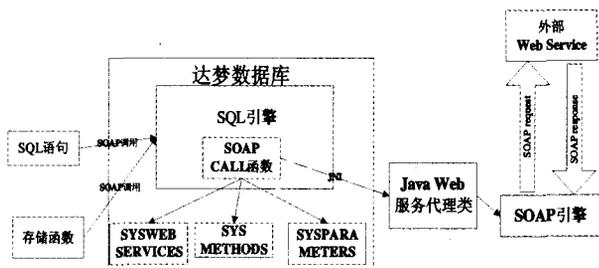


图 3 数据库消费 Web 服务

由于调用外部 Web 服务需要通过网络来传输数据，而网络的好坏和该服务的处理时间直接关系到该事务的处理流程和响应时间。最常用的 HTTP 请求是一种同步请求，在服务器端返回之前，客户端无法继续执行后面的操作。如果我们使用 HTTP 来请求一个外部的 Web 服务是否会影响数据库其它事务的执行呢？

大部分数据库中，每个事务的执行都对对应着一个线程。而在达梦数据库中，一个工作线程并不为某个特定的事务服务，当处理事务由于等待而挂起的时候，处理该事务的线程并不会挂起，而是继续到工作队列中查找是否还有用户请求要处理。因此，即使我们使用 HTTP 来请求一个长时间的外部 Web 服务，我们可以使用事件机制来将该事务挂起，而不影响到整个服务器中其它事务的处理。

**结论** 传统数据库同 Web 应用的隔离造成了要将数据提供给外部访问或者数据库要访问外部数据必须通过编写大量的代码来实现，这一方面造成了大量代码的重复编写，另一方面使得数据库永远只是在后台扮演着数据提供者的角色，而不能直接同外部 Web 应用进行交互。数据库 Web 服务思

想的提出，填平了数据库同 Web 应用的鸿沟，使得数据库从后台直接走到了前台，利用 Web 服务成为了面向服务体系结构的重要组成部分。本文提出的方法很好地将数据库同 Web 服务技术整合在一起，该方法考虑到在 Web 服务调用中，大部分都是通过 HTTP 协议使用 SOAP-RPC 的形式来使用 Web 服务的，因此专门针对这种类型的调用来实现。另外，为了方便用户在数据库中发布和调用 Web 服务，我们也开发了一些便利的工具来帮助用户使用。例如，通过指定要调用的 Web 服务的 WSDL 文档，自动声明该服务以及该服务下的操作；自动根据用户部署的数据库 Web 服务，生成 WSDL 文档等。

#### 参考文献

- 1 Hacigumus H, Iyer B, Mehrotra S. Providing Database as a Service. In: 18th International Conference on Data Engineering(ICDE'02)2002 IEEE
- 2 Simple Object Access Protocol(SOAP). http://www.w3.org/TR/SOAP
- 3 Universal Description. Discovery and Integration(UDDI). http://www.uddi.org
- 4 Web Services Activity, W3C Architecture Domain. http://www.w3.org/2002/wsk
- 5 Web Service Description Language(WSDL). http://www.w3.org/TR/wsdl
- 6 Malaika S, Nelin C J, Qu R. DB2 and Web service. IBM SYSTEMS JOURNAL, 2002, 41(4)
- 7 Database Web Services. ORACLE White Paper, November 2002
- 8 Brogden B 著. SOAP 和 Java 编程指南. 电子工业出版社
- 9 IBM Web Services Architecture Tam. Web services architecture overview. The next stage of evolution for e-business. IBM Technical Document, Web Architecture Library, 2000
- 10 柴晓路, 梁宇奇编著. Web Services 技术、架构和应用. 电子工业出版社