

文章编号:1001-9081(2008)07-1700-02

## 消息机制实现异构数据库的同步更新

雷远平

(武汉理工大学网络信息中心,武汉 430070)  
(leiy@whut.edu.cn)

**摘要:**针对异构数据库需要同步的需求,在研究 JMS 消息机制的基础上提出了一种同步解决方案,利用 XML 语言实现消息的定制,使用 JNDI 接口技术实现跨平台读写数据库中数据,异步地实现了同步数据的更新。实验结果表明这种新方法有效地解决数据中心与身份库同步的问题。

**关键词:**Java 消息服务;XML;轻量级目录访问协议;Java 命名和目录接口

**中图分类号:**TP311.52 **文献标志码:**A

### Message mechanism to achieve heterogeneous database synchronization update

LEI Yuan-ping

(Network Information Center, Wuhan University of Technology, Wuhan Hubei 430070, China)

**Abstract:** In terms of Heterogeneous database synchronization demand, a synchronization solution was proposed based on the study of JMS message mechanism. This solution actualized asynchronously the synchronic data renewal by two parts: messages customization using XML language, database read-write using JNDI interface technology. The experimental results show that this new method effectively deals with the above problems.

**Key words:** Java Message Service (JMS); Extensible Markup Language (XML); Lightweight Directory Access Protocol (LDAP); Java Naming and Directory Interface (JNDI)

## 0 引言

随着信息化和网络技术的迅速发展,企事业单位内部各部门应用系统除了掌握服务自身业务需要的私有数据外,还需要有其他多个部门的公有数据来支撑,这种跨部门公有信息资源共享的需求使数据中心应运而生。与此同时,为了更快更好地服务于有认证需求的所有应用,往往需要把那些服务于全局统一的身份数据在数据中心之外的数据库中单独存放,比如采用流行的 LDAP 数据库形式,从而可以保障各个分散独立的用户能透明地使用各种授权的应用系统,充分享受全局统一的身份数据带来的便利性。这种框架结构势必带来一个问题,就是如何实现数据中心里的身份数据与身份数据库中的身份数据同步更新,特别是当这两种数据库可能是异构时,情况就更为复杂。本文详细讨论了两种异构的数据库中部分数据同步的问题,介绍了所涉及的关键技术,同时结合 J2EE 企业计算平台,设计出了实现框架。

## 1 关键技术介绍

### 1.1 Java 消息服务(JMS)

JMS 是 Sun 公司提出的一种消息中间件系统接口规范<sup>[1]</sup>,它定义了一套通用的接口和相关词义,提供了诸如持久、验证和事务的消息服务,最主要的目的是允许 Java 应用程序访问现有的消息中间件。

JMS 技术采用异步通信模式,发送消息者将需要变更的数据消息提交到消息平台后,就完成了自己的任务,就可以进

行其他的操作,不需要等待服务器端的消息处理结果。这时即使网络出现故障甚至服务器崩溃也不会造成数据的丢失或不一致,消息会保存在消息队列中直到被最终接收<sup>[2]</sup>。

消息平台服务器端的消息驱动 Bean 会监听并接收这条信息,它调用实体 Bean 或会话 Bean 来完成具体的数据库更新操作。消息驱动 bean 实际上是一个 JMS 消息的消费者。

在实际的应用中,数据更新操作的消息一般采用 XML 文件格式。尽管 JMS 规范中没有定义 XML 的消息格式,但大部分厂商的 JMS 实现都支持 XML 消息体。

### 1.2 XML 技术

XML 作为一种可扩展性标记语言,其自描述性使其非常适合不同应用间的数据交换<sup>[3]</sup>,它具有内容的独立性,便于网络传输。XML 具有“跨平台”特性,能在任意系统间传递信息,消除系统间的“异构性”。XML 还具有易表义、可扩展性等优点,可以实现内容与形式相分离。XML 是一种简单的、开放的、被广泛接受的数据标准,这些特性使它作为一种媒介在异构数据库间方便地进行数据交换和处理<sup>[4]</sup>。

### 1.3 Java 命名与目录接口(JNDI)

JNDI (Java Naming and Directory Interface) 是 Java 程序访问基于名字与目录服务的应用程序接口(API),是帮助 J2EE 组件在运行时间地查找其他组件、资源或服务的通用机制,它独立于任何具体的目录服务。应用程序可以通过统一的 JNDI 接口访问不同的名字与目录服务。

JNDI 中两个最重要的包是 javax.naming 和 javax.directory,其中 javax.naming 包主要负责对名字服务的操作;

收稿日期:2007-12-29;修回日期:2008-03-24。

作者简介:雷远平(1967-),男,湖北武汉人,工程师,硕士,主要研究方向:网络信息系统、Web 应用、数据库。

而 javax. directory 扩展了 javax. naming 的功能,加进了访问目录服务的操作,通常调用程序访问 LDAP 数据库用到最多的也就是这个包。

在 JNDI 中,使用一个 InitialDirContext 对象,该对象采用一个 hash 表作为参数,其值为服务提供者的完全合格的类名、LDAP 服务器的主机名和端口号<sup>[5]</sup>。通过关键字 Context. PROVIDER\_URL 连接 LDAP 服务器,指定 LDAP 的认证方式后,对 LDAP 的内容进行操作<sup>[6]</sup>,将身份信息进行更新。

## 2 框架设计

为了保持身份数据的一致性,数据中心中的身份数据与应用系统需要访问的身份库中用户身份信息之间必须同步更新,或者至少能在一定的时间间隔内进行数据同步。由于平台的差异性和数据库的多样性,普通的复制技术就显得乏力,需要一种具有可靠的、通用的数据同步方法。早期采用的套接字方式与 RMI 方式等技术对网络性能等方面的要求比较高,不能很好地满足实际情况。下面介绍的是一种基于 JMS 的数据同步方法,它采用异步的通信模式,屏蔽了平台和底层通讯协议的差异性,满足异构数据库之间的数据同步。

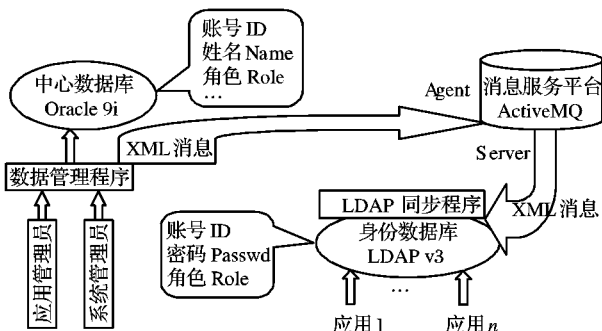


图1 异构数据库之间数据同步过程示意图

当拥有授权的数据中心数据管理员或应用数据管理员通过数据管理程序更新或维护数据中心中身份数据时,数据管理程序首先向数据中心提交一个事务,要求修改数据中心的用户身份信息;同时定制一个 XML 格式的消息,把相关的修改信息以消息的方式提交给消息服务平台,要求身份库 LDAP 中的用户身份信息进行同步更新。消息平台处理消息数据,触发专用修改 LDAP 数据的同步程序来修改 LDAP 中的相关记录,达到数据的更新同步。下面以我校修改用户密码信息同步程序为例介绍数据同步的实现过程。

## 3 数据同步的实现

### 3.1 修改数据中心的数据

...

定义数据工厂,连接特定类型的数据库(此例中实际采用 Oracle 9i 数据库):

```
DAOFactory dao = DAOFactory.getDAOFactory();
ChangePwdPDAO changePwdPDAO = dao.getChangePwdPDAO();
...
```

执行修改操作,改变密码字段,返回成功信息:

```
boolean result = changePwdPDAO.execute(u_id, newPwd);
String msg = "密码修改成功!";
...
```

### 3.2 XML 消息的处理程序

封装 JMS 消息并以 API 接口的方式提供,这样做的好处是可以使二次开发者或其他模块可以直接调用这个 API 接

口,而不再重写 JMS 内部烦琐的实现过程,有利于代码的维护和重用。因此,在实现 JMS 消息的发送和接收时,只需调用提供的接口就可以轻松地实现消息的发送和接收。

本例中用到的接口就有: MessageService、MessageSender、MessageRecevice 等。其中 MessageService 提供的方法有 createObjectMessage(), 用于消息实体的产生; MessageSender 提供的方法有 send(), 用于 JMS 消息的发布; MessageRecevice 提供的方法有 recevice(), 用于消息实体的接收。

#### 1) XML 消息产生的接口实现程序:

```
public class MessageService
{
    ...
    public static void createObjectMessage(String uid, String password)
    throws IOException
    {
        StringBuffer msg = new StringBuffer();
        msg.append("<?xml version = \"1.0\"
            encoding = \"UTF-8\"?>");
        ...
        msg.append("<EMIF_Message >");
        msg.append("<UID >").append(uid).
            append("</UID >");
        msg.append("<PASSWORD >").append(password).append(
            "</PASSWORD >");
        msg.append("</EMIF_Message >");
        ...
    }
}
```

#### 2) XML 消息发送的接口实现程序:

```
public class MessageSender
{
    public static synchronized void send(String ID, String uid, String
    password) {
        ...
        SocketChannel sc = null;
        sc = SocketChannel.open(socketAddress);
        sc.write(charset.encode(msg.toString()));
        sc.close();
        ...
    }
}
```

#### 3) XML 消息接收的接口实现程序:

```
public class MessageRecevice
{
    public StringTokenizer recevice() throws IOException
    {
        ...
        SocketChannel sc = null;
        sc = SocketChannel.open(socketAddress);
        sc.write(charset.encode(strin.toString()));
        StringBuffer recMsg = new StringBuffer();
        for (; sc.read(buffer) > 0; buffer.clear())
        ...
        sc.close();
    }
}
```

### 3.3 应用程序对消息的使用

#### 3.3.1 消息的生产

应用程序如果要求修改身份信息中的密码信息,调用 MessageSender 包中的 send 方法向消息平台发送已产生的修改数据消息:

(下转第 1708 页)

接对普通样本进行建模。而且基于人工免疫的分类算法可以产生容易让人理解的规则,从表 2 中可以看到,IRAA 的离网类准确率在三种分类算法中最高。

### 3.3.3 实验三

为了更好的验证基于人工免疫系统的分类模型的有效性,实验三进行了大数据量的实验,实验数据是将 2006 年 4 月份的全部数据当训练集(231 222 条),5 月份的全部数据当测试集(228 395 条)。这里将 IRAA 与 IFRAIS、贝叶斯、C5.0、神经网络和逻辑回归方法进行了实验比较,结果见表 3。

从表 3 可以看出,IRAA 的离网类准确率达到 32.15%,离网类用户的分类准确性要高于其他 5 种分类算法。同时 IRAA 可以产生容易理解的规则,在实际应用中得到很好的应用效果。

表 3 全部数据各种算法分类性能比较

算法	在网类		离网类		离网类 准确率/%	离网类 召回率/%	规则数
	正确识别	错误识别	正确识别	错误识别			
贝叶斯	190 467	25 661	3 926	8 341	32.00	13.27	-
C5.0	211 429	4 699	950	11 317	7.74	16.82	-
神经网络	211 298	4 830	1 153	11 114	9.40	19.27	-
逻辑回归	213 772	2 356	751	11 516	6.12	24.33	-
IFRAIS	192 637	23 491	3 440	8 827	28.04	12.77	12
IRAA	189 092	27 036	3 944	8 323	32.15	12.73	13

## 4 结语

本文针对模糊规则归纳算法(IFRAIS)的规则进化过程中容易陷入局部最优的弱点,在其克隆选择过程中增加了抗体与抗原的交叉,提出了一种新的基于人工免疫系统的分类方法——抗体抗原交叉的规则归纳算法(IRAA)。对 IRAA、IFRAIS 和传统分类算法进行比较实验,结果证明,这些改进可以提高算法性能。在客户流失模型建立的过程中加入更多的客户属性资料、呼叫模式和付费数据等更多变量进行学习

和推理,对 IRAA 采用模糊系统使其能够处理连续型数据,进一步提高模型的预测精度是下一步将要研究的计划。

本文还是第一次将基于人工免疫系统的分类方法运用到电信客户流失分析中,并且具有不需要过量抽取数据和产生容易理解的规则的特点,在其他行业业务系统中将有广泛的应用前景。

### 参考文献:

- [1] 段云峰,吴唯宁,李剑,等.数据仓库及其在电信领域中的应用[M].北京:电子工业出版社,2003.
- [2] MOZER M C. Predicting subscriber dissatisfaction and improving retention in the wireless telecommunications industry[J]. IEEE Transactions on Neural Networks, 2000, 11(5): 690-696.
- [3] 石永华.电信业务流失建模的研究[J].广东通信技术,2003(6):15-20.
- [4] 叶进,林士敏.基于贝叶斯网络的推理在移动客户流失分析中的应用[J].计算机应用,2005,3(25):673-675.
- [5] 焦李成,杜海峰.人工免疫系统进展与展望[J].电子学报,2003,31(10):1540-1548.
- [6] WATKINS A. AIRS: A resource limited artificial immune classifier[D]. Starkville: Mississippi State University, 2001.
- [7] de CASTRO L N, von ZUBEN F. Learning and optimization using the clone selection principle[J]. IEEE Transaction Evolution Computation, 2001,6(3):239-257.
- [8] ALVES R T, DELGADO M R, LOPES H S, et al. An artificial immune system for fuzzy-rule induction in data mining[C]//Lecture Notes in Computer Science. Berlin: Springer-Verlag, 2004, 3242: 1011-1020.
- [9] 刘芳,戚玉涛,公茂果.免疫克隆分类算法[J].电子学报,2005,33(12):2301-2307.

(上接第 1701 页)

```
...
if (ID != null && !"".equals(ID))
    Debug.out("发送同步消息");
    MessageSender.send(ID, uid, password);
    Debug.out("同步消息发送成功");
```

### 3.3.2 消息的消费

当有消息到达消息中心系统时,服务器端的消息驱动 Bean 会监听到,它对接收的消息进行解析,取出相应的数据库字段,并调用相应的 EJB 组件来完成业务的逻辑处理,更新数据库。本例中就是使用 LDAP 的同步程序通过访问消息平台服务端获取修改数据的消息,并依此修改 LDAP 中的密码信息的,程序如下:

```
...
StringTokenizer st = MessageRecevice.recevice();
if (st != null)
    updateLdap(st);           调用 updateLdap 类更新用户信息
...
```

本例中使用 LDAP 数据库作为身份库,由于 LDAP 库使用树形目录形式,不同于关系型数据库的读写方式,在更新 LDAP 库中数据时需使用 JNDI 技术来实现。同步程序使用 JNDI 的 javax. Directory 包,通过调用 InitialDirContext 对象实例化 ctx 变量来连接 LDAP 库,建立连接后再依据已接收的消息数组 st 中数据修改 LDAP 中的相关数据,实现同步更新。

程序实现比较简单,限于篇幅的原因在此不作详细讨论。

## 4 结语

针对企业级应用中平台、数据库等的多样化,使用 JMS 技术提供了在组件之间进行有效的异步通信机制,采用 JMS 技术的数据同步方案能保证数据准确、高效的传输。本文提出了通过消息机制实现关系型数据中心库到 LDAP 身份库的异步同步方法,既解决了身份统一的问题,同时兼顾了认证身份快速性的要求,很好地满足了实际的需要,提高了系统的运行效率,实践证明是一种行之有效的方法。

### 参考文献:

- [1] HAPNER M, BURRIDGE R. Java 消息服务 API 参考指南[M]. 康博,译.北京:清华大学出版社,2002.
- [2] 罗晓斌,董守斌.基于 JMS 的异步消息处理技术及应用[J].计算机工程,2002,28(12).
- [3] 郑贤中,王乘.XML 的数据建模及其应用[J].计算机工程,2003,29(5).
- [4] 杨晓宇.基于 XML 的数据交换的实现[J].科技情报开发与经济,2004,14(4).
- [5] OpenLDAP Software 2.4 Administrator's Guide[EB/OL]. [2007-05-05]. <http://www.openldap.org/>.
- [6] O'DONAHUE J. Java 数据库编程宝典[M].北京:电子工业出版社,2003.