

传感器实时信息共享的一种实现方案*

苟和平, 冯百明, 邹燕飞, 景永霞

(西北师范大学 数学与信息科学学院, 兰州 730070)

摘要: 针对企业中会存在的各类传感器所产生的关于自然现象的一些实时信息, 如果能够得到及时的分析处理, 并作出相应的决策, 将会极大地提高企业的运作效率。提出了一种在信息网格中集成传感器的方法, 并以智能数字磁场计 HMR2300 为例说明其实现过程。实践表明, 该方法能够更方便、更快捷地将传感器集成在信息网格中, 实现实时信息共享。

关键词: 信息网格; 传感器; Web 服务; REML

中图分类号: TP311.5 **文献标志码:** A **文章编号:** 1001-3695(2008)03-0833-03

Scheme for sharing of real-time information generated by sensors

GOU He ping FENG Bai ming ZOU Yan fei JING Yong xia

(College of Mathematics & Information Science Northwest Normal University Lanzhou 730070, China)

Abstract: There were a lot of sensors in an enterprise the real-time information they generated about the phenomena in the physical world was very important for the enterprise. If they were processed mined as soon as possible to make a reasonable decision correspondingly, high performance could be achieved in enterprise runtime. This paper presented the method of integrating sensor into information grid and described its implementation by an example of HMR 2300. Realworld application shows that the method can make the integrating more convenient and faster and the real-time information shared on time.

Key words: information grid; sensors; Web services; REML

企业中目前还存在着大量企业用户无法访问到的由传感器所产生的信息, 这些实时信息对于企业来说是一笔宝贵的财富, 因此, 如何向信息消费者共享这些信息已经成为当前研究的热点问题之一。近几年来, 人们已经做了大量的努力来建立支持传感器应用和开发的更简单、更快捷的中间件。为 Zebra-Net 项目设计的 Impala^[1], 采用代码移动技术来更新远程传感器功能的同时考虑到应用本身。MILAN^[2]是一个体系结构, 它扩展了网络协议栈并允许网络具体插件将 MILAN 命令转换成具体协议的命令。除了开发支持传感器应用的中间件外, 将传感器网络与信息网格计算技术结合起来构成传感器网络也是很重要的方法之一。Tham 等人^[3]概括了传感器网络的一种基本结构, 并通过分布式信息融合和分布式自主决策算法的一种可能实现的一些例子, 描述了在传感器网络计算方面的一些早期工作。但是, 在这种理想的结构变成现实之前还有很多需要解决的研究问题。Ghanem 等人^[4]介绍了另一种传感器网络集成技术, 利用网格服务包装高吞吐率的传感器, 通过一个标准的方法将这个服务发布到一个注册中心以便被其他用户访问。

由于 Tham 和 Buyya 所提出的结构在目前还没有成熟的系统, 而且对于企业来说, 同时构建网格和传感器网络将是一项企业自身无法承受的工程。本文提出直接将传感器集成到企业信息网格中的结构, 提供给企业用户一个通用的访问接口, 实现传感器实时信息共享, 极大地节约了系统开发成本。

1 基本体系结构

企业信息网格为企业用户提供了一体化的智能信息平台。

在这个平台上, 信息的处理是分布式、协作和智能化的^[5]。用户可以通过单一的入口访问所有信息, 以便能够及时作出决策, 提高企业的运作效率。本文当前实现系统的基本体系结构如图 1 所示。

- 1) 物理层 它是各类传感器, 如温度传感器、图像传感器、磁阻传感器等, 它们都是通过标准的接口与计算机相连。
- 2) 服务层中间件 它是为上层提供基本的服务, 包括传感器注册服务 (sensor register service SRS)、传感器操作服务 (sensor operation service SOS) 和信息传送服务 (information transmission service ITS) 等。
- 3) 开发层中间件 它提供高层应用的访问接口, 包括传感器编程接口、传感器实时信息管理接口等。

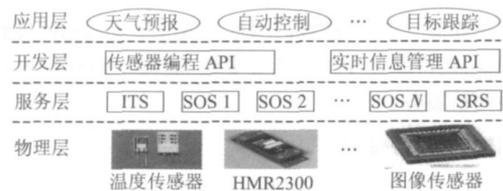


图 1 基本体系结构

2 主要技术问题

2.1 服务层中间件

对于科学计算来说, Web services 是一个极具吸引力的模型, 但在几年前, I Foster 等人^[6]提出了将 Web services 模型集

收稿日期: 2007-01-29; 修回日期: 2007-04-09 基金项目: 甘肃省科学技术攻关计划项目 (2GS047-A52-002-04)

作者简介: 苟和平 (1978-), 男, 甘肃庆阳人, 硕士研究生, 主要研究方向为 Web 服务和网格计算 (gou_he_ping@163.com); 冯百明 (1966-), 男, 甘肃人, 教授, 博士, 主要研究方向为计算机体系结构和网格计算; 邹燕飞 (1981-), 女, 甘肃庆阳人, 硕士研究生, 主要研究方向为 Web 服务和网格计算; 景永霞 (1984-), 女, 甘肃天水人, 硕士研究生, 主要研究方向为数据仓库和数据挖掘。

成在 Globus 中的建议。作为基于 Web services 的网络实现的第一步, Globus 联盟给出了一些规范,使得 Globus 更有效地将应用映射为 Web 服务。在实现了 OGSII(open grid services infrastructure)^[7]规范的“open grid services infrastructure reference implementation”中定义了网格服务是 W3C Web services 的扩展。

在本文实现的基础平台中使用网格服务来包装传感器,即一个传感器对应一个操作本传感器的 SOS 通过一个通用的接口将这个服务发布到注册中心(UDDI)供其他网格用户访问。主要是完成向传感器发送控制信息来控制传感器工作和从传感器接收实时信息。ITS 将从传感器读来的实时信息传送给信息消费者。不同传感器产生的信息格式不同,如温度传感器可能产生的是一个双精度的值;而图像传感器可能产生的是图像的二进制文件。因此,传感器的位置、产生的信息频率及格式、传感器所对应的服务的 RUL 等元信息通过 SRS 向注册中心(UDDI)发布。下面是发布智能数字磁场计 HMR2300 的信息(采用的是下文介绍的 REML):

```
<? xml version="1.0" encoding="UTF-8" ?>
< root>
  < element>
    < name>智能数字磁场计 HMR2300< /name>
    < location>210.26.97.253< /location>
    < info:format>20m< /info:format>
    < info:type>double< /info:type>
    < service>http://210.26.97.253:8080/grid/services/General_dyn_k< /service>
    < method1>display_p< /method1>
    < method2>display_c< /method2>
  < /element>
< /root>
```

服务层中间件是完全符合网格的标准,对其的交互操作也是符合网格的标准协议。但由于本文实现的网格服务和传感器是一一对应的关系,有以下一些特点:

a)功能独立性。每一个传感器的功能均通过网络服务层中间件提供的接口实现,上层网格应用必须开发一个能够对传感器操作的模型来实现和不同传感器的交互。

b)面向资源的服务。不同的传感器具有不同的功能,各自所需要的启动参数及返回参数等一般均不同,因此,提供了一个开发服务的 API。服务开发者通过对传感器的功能和一些主要参数的了解,就能够调用这个 API 生成面向此传感器的特定服务,使得服务开发者能够更快、更简单地开发出符合本系统的网格服务。

2.2 信息格式及编码

为了使信息能够方便被上层应用所识别和利用,提出了信息的一种中间格式——REML。其基本结构如下:

```
< root>
  < element>
    < tag1>...< /tag1>
    < tag2>...< /tag2>
    ...
    < tagn>...< /tagn>
  < /element>
< /root>
```

其中:< element> 标签表示传感器一次产生的信息量。有些传感器(温度传感器等)一次只产生一个双精度的值;还有的传感器一次就产生一组数据信息(无论是一个还是一组,均称之为“帧信息”)。< tag_i> (i=1, 2, …, n) 标签是根据具体传感器

产生信息的类别和数量决定,如智能数字磁场计 HMR2300 可检测磁场的强度和方向,输出 X、Y、Z 三个轴的分量。一帧信息的 REML 如下:

```
<? xml version="1.0" encoding="UTF-8" ?>
< root>
  < element>
    < X-axis>1.562< /X-axis>
    < Y-axis>-7.683< /Y-axis>
    < Z-axis>-5.821< /Z-axis>
  < /element>
< /root>
```

在传感器与消费者之间加入的 REML 格式信息采用 UTF-8 编码。不论是在 Windows 环境还是 Linux 环境,Java 客户端还是 C 客户端,只要将这种 REML 格式的信息在各自的客户端解析成本地代码均能够成功识别,而不会出现乱码导致上层应用无法处理,很好地解决了跨平台访问的问题。其基本结构如图 2 所示。



图 2 传感器与消费者之间的 REML

3 信息消费方式

消费者可选择接收一帧信息(请求-响应或是拉模式)或是连续信息(流或推模式), SOS 提供了这两种方法(display_p 和 display_c),消费者可根据自己的需求选用不同的消费方式进行消费。

1)拉模式(pull) 消费者通过调用 SOS 来请求传感器实时信息,此时传感器返回当前产生的一帧信息,一次服务结束。其基本结构如图 3 所示。

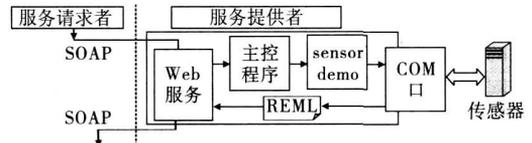


图 3 拉模式的基本实现结构

2)推模式(push) 为了使消费者能够得到可靠的流格式信息,通过 socket 连接来实现实时信息的可靠连续传输。在这种模式下,消费者仍然是通过服务调用的方式来访问传感器信息。但消费者调用服务时需要指明消费频率(f),接收传感器信息的端口号,服务则启动一个线程不断地向接收信息端口发送传感器实时信息,直到消费者明确发出一个停止请求。其基本结构如图 4 所示。

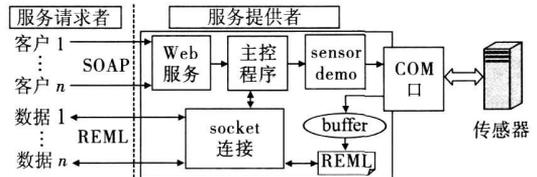


图 4 推模式的基本实现结构

在此模式中,在客户端和服务端分别建立一个缓冲区 buffer_c 和 buffer_s。Buffer_c 位于客户端与上层应用之间,是为了缓存那些已经从传感器中读出而上层应用还没有来得及处理的信息。Buffer_s 位于消费者与传感器之间,解决消费者消费的频率与传感器的发送频率不匹配的问题。

设传感器产生信息的频率为 f, 则

a) $f > f_c$ 按 f_c 的大小从 $buffer_c$ 中读取传感器信息放入 $buffer_r$ 中。

b) $f < f_c$ 按 f 的大小从 $buffer_c$ 中读取传感器信息放入 $buffer_r$ 中。

第一个消费者请求消费时调用 SOS 服务调用读 COM 口程序启动传感器并从传感器中读出实时信息存放在 $buffer_c$ 中。随后的消费者请求消费时, 仍然调用 SOS 但这时服务器检测到传感器已经启动, 就启动一个线程直接从 $buffer_c$ 中按照消费者请求消费的频率读取信息。消费者调用 SOS 获得信息的源代码如下:

```
class Dynamic{
String D_str;
myclient my;
Mon share;
Dynamic(String endpoint String method int a int b){
share=new Mon();
D_clientmythread=new D_client(endpoint method a b); /* 从
COM 口读数据的时间间隔 a 和从缓冲区读信息送到客户端的时间间隔
b 默认的时间间隔是 20 ms 和 20 ms*/
mythread start(); //启动调用 SOS 的线程
my=new myclient(share endpoint);
Thread mycli=new Thread(my);
mycli start(); //启动接收 socket 流信息的线程
}

public String dynam icstr(){
Thread myacc=new Thread(my new access(share));
myacc start(); //启动从客户端缓冲区中读信息的线程
D_str=my getmyst();
return D_str; //返回信息
}
}
```

智能数字磁场计 HMR2300 在两种模式下的访问结果如图 5 所示 (图 5(a) 中访问到的信息用方框标出)。



(a) 拉模式 (b) 推模式
图 5 访问智能数字磁场计 HMR2300 的结果

4 性能分析

两种消费方式是基于不同的传感器及消费者不同的消费方式提出的, 均存在着各自的优缺点。实验数据如图 6 所示。

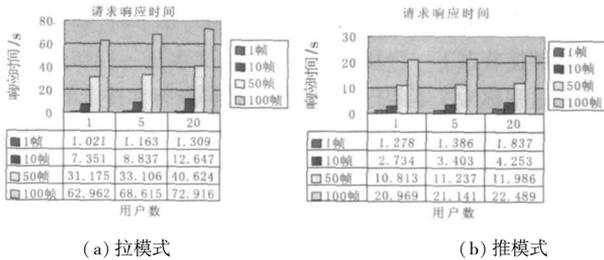


图 6 两种消费模式分析

设调用 SOS 读传感器实时信息的时间为 T_o , 调用 ITS 的时间 T_r , 初始化 COM 口的时间为 T_c , 建立 socket 连接的时间 T_s , 则

在拉模式下, 请求 n 帧信息的响应时间为

$$T_{1n} = T_c + \sum_{i=1}^n (T_o + T_r)$$

当 $n=1$ 时, $T_{11} = T_o + T_r + T_c$
在推模式下, 请求 n 帧信息的响应时间为

$$T_{2n} = T_c + T_o + \sum_{i=1}^n (T_s + T_r)$$

当 $n=1$ 时, $T_{21} = T_o + T_r + T_c + T_s$

由实验数据及分析可以看出, $n=1$ 时, $T_{11} < T_{21}$ 。由于 T_o 包括启动服务的时间、从 COM 读数据的时间、数据的 REML 化的时间, 因此 $T_o \gg T_s$, 随着请求信息的帧数和用户数的增长, $T_{1n} \gg T_{2n}$ 。大量的实验表明, 大约在用户每次请求信息的帧数小于 5 帧时, 选用拉模式消费较好; 反之, 则用推模式。

5 本系统的优点

1) 开放性 本系统采用的是基于 Web services 的网格技术, 网格服务提供了一种与现有应用相集成的有效方式, 能够与现有的企业应用程序整合, 使本系统具有良好的开放性。

2) 松散耦合 客户端与服务端之间以 REML 为交换数据, 使得本系统具有松散耦合性。

3) 重用性 网格服务对外提供的实时信息, 可作为其他信息系统的数据库, 也可供上层应用的二次开发之用。

6 结束语

在企业信息网格中利用网格技术的客户提供传感器实时信息, 实现了企业中传感器实时信息的全面共享, 提高了企业的运作效率。消费者可以根据自己的不同需求选用拉模式或推模式的消费方式请求信息供自己观测或二次开发之用。同时, 采用 REML 的中间格式, 很好地解决了跨平台的问题。本文已经在自己开发的企业信息网格 Loglo 系统中实现了这两种模式。实践表明, 用网格技术来提供传感器实时信息共享是一种行之有效的解决传感器信息孤岛的方法。

参考文献:

- [1] LU T, MARTONOSIM. Inpak: a middleware system for managing autonomous parallel sensor systems[C] // Proc of the 9th ACM SIGPLAN Symp on Principles and Practice of Parallel Programming 2003, 107-118.
- [2] HE NZELMAN W, MURPHY A, CARVALHO H. Middleware to support sensor network applications[J]. IEEE Network Magazine, 2004, 18(1), 6-14.
- [3] THAM C K, BUYYA R. SensoGrid: integrating sensor networks and grid computing[J]. CSICommunications, 2005, 29(1), 24-29.
- [4] GHANEM M, GUO Y, HASSARD J, et al. Sensor grids for air pollution monitoring[C] // Proc of the 3rd UK e-Science All Hands Meeting Nottingham, [s n]. 2004, 106-113.
- [5] 徐志伟, 冯百明, 李伟. 网格计算技术[M]. 北京: 电子工业出版社, 2004, 25-58.
- [6] FOSTER I, KESSELMAN C, NICK J, et al. The physiology of the grid: an open grid services architecture for distributed systems integration (draft) [EB/OL]. (2002-01). <http://www.globus.org/research/papers/ogsa.pdf>
- [7] TUECKE S, FOSTER I, KESSELMAN C, et al. Open grid services infrastructure (OGSI) (draft) [EB/OL]. (2003-02-01). http://www.gridforum.org/ogsiwg/drafts/draft-ggf-ogsi-gridservice-23_2003-02-17.pdf