

基于并发工作流的财务监管数据汇聚模型

那丽春¹, 陈庆奎²

(1. 上海立信会计学院信息科学系, 上海 201620; 2. 上海理工大学计算机与电气工程学院, 上海 200093)

摘要: 企业集团的财务数据通过 Internet 跨越多个大行政区, 集团内的财务软件和数据系统具有异构特性。为有效汇集集团的财务监管数据, 以并发工作流为基础, 提出财务监管数据汇集模型。利用基本财务数据, 构造树形工作流, 使用财务数据汇总服务器树, 构建多级数据汇集机制, 通过树形工作流和多级数据汇集机制实现该模型。分析和实验结果表明, 该模型具有良好汇集效率和近线性的可扩展性, 满足了企业集团的信息化建设需要。

关键词: 财务监管; 并发工作流; 树形工作流; 异构数据汇集

Gather Model Based on Concurrent Workflow for Finance Monitoring Data

NA Li-chun¹, CHEN Qing-kui²

(1. Dept. of Information Science, Shanghai Lixin University of Commerce, Shanghai 201620;

2. School of Computer and Electrical Engineering, University of Shanghai for Science and Technology, Shanghai 200093)

【Abstract】 The financial data of enterprise group covers many big districts through Internet, and the financial software and data system in the group present the heterogeneous characteristic. In order to effectively gathering the financial monitoring data of enterprise group, the concurrent workflow is adopted to construct the gather model for financial monitoring data in this paper. By using of the basic Financial Data(FD), the tree-shape workflow is constructed. By making use of the gather server for FD, the multi-level gather mechanism is proposed. This model is implemented through the tree-shape workflow and the multi-level gather mechanism. Analysis and experimental results show that this model has good gather efficiency and linear extendibility. It can be fit for the need of information building for enterprise group.

【Key words】 finance monitoring; concurrent workflow; tree-shape workflow; heterogeneous data gather

1 概述

随着集团化企业数量的日益增加, 集团财务数据监管变得越来越重要。历史原因和应用技术的不同导致财务软件存在异构特性。以上海豫园集团的财务软件为例, 共包括用友 20 套、金蝶 2 套、新亚 13 套、万能 1 套、立成 20 套、金算盘 23 套。面对此类庞大的异构财务系统, 要实现财务的实时监管只有 2 种途径: (1) 完全更换财务系统, 统一版本。此途径可以解决软件异构问题, 但需要巨大成本, 且对新版软件的培训、使用需要投入巨大精力和财力, 会带来新的应用风险, 甚至牺牲重要历史数据。(2) 保持现有财务系统不变, 通过软件技术使其与新框架融合。此途径省钱、省力且可靠, 但异构系统带来的问题一直难以解决, 通过途径(2)进行财务监管具有较大难度。

在集团财务监管的财务信息中, 存在跨 Internet 的多个并发信息, 此类信息按一定策略, 从各个被监管的财务子系统逐级流动到集团财务中心, 以供集团财务监管应用。上述信息在流动汇总过程中具有时序特性和并发特性, 可以采用并发工作流^[1-2]技术实现监管。目前, 数据整合技术^[3]、网络技术^[4]、并发工作流^[5]和分布式工作流^[6]都为集团财务的监管奠定了基础, 但有关集团财务监管的研究不多。本文利用网络技术、异构数据表示机制、并发工作流理论, 针对分布在 Internet 上的财务信息, 通过分布式数据抓取和协同数据汇总管理技术, 研究集团财务监管数据汇集模型(Gather Model based on concurrent workflow for Finance monitoring Data,

GMFD)。在此基础上, 可以对集团财务信息实施动态实时分析和监控, 对提高集团的生产效率、降低成本具有现实意义。

2 GMFD 模型结构

集团通常是由多个子公司构成的树形层次结构。企业集团的财务监管运作过程主要包括 2 个方面的工作: (1) 所有下属公司的财务系统运营状况的实时汇总, 简称数据汇总; (2) 各个子集团对其下属公司的财务汇总、分析、审批和上报到其父集团的工作, 简称数据汇总同步。第(1)个方面体现并发数据流的特征, 第(2)个方面体现并发工作流的特性。

定义 1(财务数据汇总服务器树) 是一棵由多个分布在 Internet 上的数据库服务器为节点的树 FT , FT 由 3 类节点构成: (1) 一个根服务器节点 RN ; (2) 叶节点 LN , 即运行某个具体财务软件系统的节点(服务器); (3) 中间汇集节点 GN , 介于根节点和叶节点之间的节点, 又称为汇总服务器, 可以是子集团的财务服务器或中间数据汇集、汇总服务器, 它可以完成数据汇总或数据汇总同步工作。 FT 的节点可以表示为八元组 $FTN(fid, IPA, LOC, DB, FA, SONS, SYSS, TYP)$, 其中, fid

基金项目: 国家自然科学基金资助项目(60573108); 上海市教育委员会发展基金资助项目(06QZ002, 07ZZ92); 上海市教育委员会科研创新基金资助重点项目(08ZZ76); 上海市教育委员会科研创新基金资助项目(09YZ428); 上海市教育科学研究基金资助项目(B08056)

作者简介: 那丽春(1967-), 女, 副教授, 主研方向: 财务监管, 网络计算; 陈庆奎, 教授、博士生导师

收稿日期: 2008-11-21 **E-mail:** chenqingkui@tom.com

为节点标识符； IPA 为该节点对应的服务器 IP 地址； LOC 为服务器的物理位置描述； DB 为运行在该节点上的数据库，如果该节点为叶节点，那么 DB 为具体财务系统数据库，否则为汇总数据库； FA 为该节点的父节点，当其为根节点时， FA 为 ϕ ； $SONS=\{C_1, C_2, \dots, C_p\}$ 为该节点所有子节点的集合，当该节点为叶节点时， $SONS=\{\phi\}$ ； $SYSS$ 为该节点的同步策略； TYP 为节点类型，若 $TYP=RN$ ，则为根节点，若 $TYP=GN$ ，则为汇集节点，若 $TYP=LN$ ，则为叶节点。

财务信息汇总树可以客观反映集团服务器分布的层次构架，它表示集团的数据库机器的主从连接关系。其根节点为集团数据库服务器，叶节点为最底层公司的财务数据库服务器，汇总节点对应子集团或专用财务数据汇总服务器。财务信息汇总树的结构如图 1 所示。

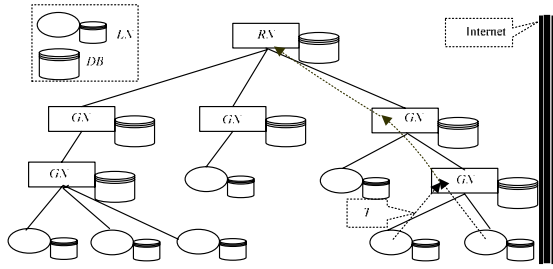


图 1 财务信息汇总树的结构

定义 2(财务数据) 一个财务数据是一个具有同一数据结构的数据的集合 $FD=\{D_1, D_2, \dots, D_n\}$ 。例如，在关系数据库中，一个财务数据可以是一个关系表、一个元组或一个关系表的子集。因此， FD 可以遵从集合运算的所有操作。

定义 3(树形 workflow) 是二元组 $TDF(FD, T)$ ，其中， FD 为一个财务数据； T 为一棵从叶节点指向根节点的有向树，称为 workflow 树， T 的节点结构为 $T(id, fid, LFD, fa, SONS, SYST)$ ，其中， id 为其标识符； fid 为存放该节点的财务数据汇总服务器树的节点标识符； LFD 为隶属本节点的财务数据集合； fa 为其父节点指针； $SONS$ 为该节点所有子节点的集合； $SYST$ 为来自其下游节点的工作流的同步类型，当 $SYST=0$ 时，从该节点所有下游节点到来的 workflow 不必同步汇总，它们经过本地存储后，被立即传输到下游(父)节点，仅完成数据汇总工作，此时经过该节点的数据相当于数据流，当 $SYST=1$ 时，从该节点所有下游到来的 workflow 必须停止流动，待所有分流到齐并加工处理、分析、审批后，才能经过本地传输到下游(父)节点，该 workflow 节点要完成数据汇总同步工作。

workflow 树 T 反映了财务数据 FD 在 T 上经过多次流动、汇总、分析、审批等过程，直至流动到树根的逻辑过程。设 FT 是与 T 同深度的财务数据汇总服务器树。把 T 的根节点部署到 FT 的根节点上，把 T 的叶节点按现场的实际应用规则部署到 FT 的叶节点上，并把 T 的内节点部署到 FT 的汇集节点(数据汇总或数据汇总同步)上。此时，财务数据 FD 可按 T 定义的树形策略，通过 FT 各层节点(服务器)的运行，流动汇总到 FT 的根节点上。 FT 根节点的数据库可以得到 FD 的汇总数据。 T 和 FT 的关系如图 1 所示，其中，虚线表示有向树 T 在 FT 上的分布情形。

定义 4(并发树形 workflow) 设一个集团被监控汇总的财务数据集为 $FDS=\{FD_1, FD_2, \dots, FD_m\}$ ，根据集团的监管业务逻辑的描述为 FDS 中每个财务数据构建一个树形 workflow，并把此类树形 workflow 按一定策略映射到 FT 上，则上述树形 workflow 在 FT 上并发地存在，称它们为并发树形 workflow。一个集

团的财务监管数据集 FDS 的所有财务数据按上述并发树形 workflow 定义的逻辑结构汇总而成。

3 FT 的进程部件描述

FT 每个节点上的进程描述如下：

(1) 并发发送过程 $SENP$ 、并发接收过程 $RECP$ 。 FT 的每个节点都存在 2 个并发的通信进程：接收来自其他节点的数据(命令)接收进程 $RECP$ ，把本地数据(命令)发送到其他节点的数据发送进程 $SENP$ 。 $RECP$ 和 $SENP$ 为 2 个独立于其他进程的进程。

(2) 数据抓取进程 DG 。 FT 的叶节点上运行着一些财务数据抓取进程。设一个集团被监控汇总的财务数据集为 $FDS=\{FD_1, FD_2, \dots, FD_m\}$ ，则对每个财务数据 $FD_i(1 \leq i \leq m)$ 构建一个数据抓取进程 DG_i ， DG_i 负责从本地异构财务系统数据库中提取财务数据 FD_i ，并用统一的 XML 模式表示该财务数据(在树形 workflow 中，所有财务数据都以规范的 XML 模式存放)。

(3) 数据汇集进程 DGH 。 FT 的根节点和汇集节点上运行着一些财务数据汇集进程。设一个集团被监控汇总的财务数据集为 $FDS=\{FD_1, FD_2, \dots, FD_m\}$ ，则对每个财务数据 $FD_i(1 \leq i \leq m)$ 构建一个数据汇集进程 DGH_i ， DGH_i 负责汇总其上游(子)节点发来的异构财务数据，并用统一的 XML 模式表示汇总数据。 DGH 包括 2 个部分，即数据汇总和数据汇总同步，它们根据实际应用的需要来确定。

(4) 节点主控进程 MCP 。 FT 的每个节点上都运行着一个负责控制、协调、管理节点资源和其他进程的进程，即节点主控进程。在 FT 上有存储管理信息的数据库，上述信息以对象关系模型进行存储。

(5) 并发树形 workflow 库。是一个四元组 $CTDB(id, FD, TDF, DG, DGH, XMLD)$ ，其中， id 为标识符； FD 为一个财务数据； TDF 为树形 workflow 描述(见定义 3)； DG 为数据抓取进程； DGH 为数据汇集进程； $XMLD$ 为其数据的 XML 表示框架。

(6) 财务监管数据库。设集团被监控汇总的财务数据集为 $FDS=\{FD_1, FD_2, \dots, FD_m\}$ ，则 FDS 的所有数据通过 $CTDB$ 的结构描述并在 FT 上执行后，汇总在 FT 根节点数据的集合中。

4 GMFD 模型工作过程及其算法

设集团被监控汇总的财务数据集为 $FDS=\{FD_1, FD_2, \dots, FD_m\}$ ，且所有财务数据库服务器按业务逻辑要求构成了 FT 。 $FSS=\{FS_1, FS_2, \dots, FS_p\}$ 为企业所使用财务软件的集合。 $CTDB$ 为并发树形 workflow 库。

算法 1 构建 DG, DGH 和 $XMLD$

```

(1)for  $FD_i \in \{FD_1, FD_2, \dots, FD_m\}$  do (2)
(2)for  $FS_i \in \{FS_1, FS_2, \dots, FS_p\}$  do (3)~(6)
(3)对  $FD_i$  编写其 XML 的表示模式  $XMLD_i$ ;
(4)对  $FS_i$  的  $FD_i$  编写  $DG$  和  $DGH$  得到进程  $DGP$  和  $DGHP$ ;
(5)创建一个  $CTDB$  对象  $O$ ，生成一个  $CTDB$  对象  $newid$ ;
(6) $O.id=newid$ ;  $O.FD=FD_i$ ;
     $O.DG=DGP$ ;  $O.DGH=DGH$ ;
     $O.XMLD=XMLD_i$ ;
     $CTDB=CTDB \cup \{O\}$ ; // 加入到  $CTDB$ 

```

(7)end

树形 workflow 是一个从叶节点到根节点的有向图， TDF 的构建是自底向上的逐层的运作过程。设 $FD=\{FD_1, FD_2, \dots, FD_m\}$ ，则根据 FD 构建 TDF 的算法如下：

算法 2 构建 TDF

(1)获取 FD 在 FT 上的部署策略的语义描述 $DESC$;

(2)把 FD 按 DESC 的描述部署在 FT 的部分节点上;

(3)FT 的 RN 上的主控进程向 FT 的所有叶节点广播命令 Construct TDF for FD 和需要创建的 TDF 的 nid;

(4)所有叶节点收到 Construct TDF for FD 后, 检查自己是否已经被部署了 FD;

(5)如果其已经被部署了 FD, 则创建一个 TDF 节点 T(id, fid, LFD, fa, SONS, SYST), do

{T.id=nid; T.fid=FT 的该叶节点的 id; T.LFD=FD;T.fa=FT 的该叶节点的父节点; T.SYST=0}; T.SONS= ϕ ;

其向其父节点发出 Construct workflow 请求和已经构建好的 T;}

(6)当一个父节点在收到所有来自其子女的 Construct workflow 请求和 T 后, 创建一个 TDF 节点 NT(id, fid, LFD, fa, SONS, SYST), do

{NT.id=nid; T.fid=FT 的该父节点的 id; NT.LFD=FD;NT.fa=FT 的该父节点的父节点;

NT.SONS=隶属该父节点的所有发送 Construct workflow 的子节点构成的集合;

如果该父节点也被部署了 FD, 则 NT.SYST=1, 否则 NT.SYST=0; 其向其父节点发出 Construct workflow 请求和已经构建好的 NT;}

(7)重复执行(6), 直到 FT 的根为止;

(8)end

算法 3 构建 CTDB 的树形工作流 TDF 项

```
for 每个对象 O CTDB do
    {根据 O.FD 得到财务数据 FDi;
    执行算法 2 为财务数据 FDi 构建 TDFi;
    置 O.TDF=TDFi;
    用对象 O 更新 CTDB; }
```

算法 4 FT 上的进程配置

(1)执行算法 1 为所有财务数据构造 DG, DGH 和 XMLD;

(2)执行算法 3 完善 CTDB, 并部署 CTDB 中的 TDF 到 FT 上;

(3)所有节点部署进程 RECP, SENP 和 MCP;

(4)for 每个 O CTDB do
{根据树形工作流 O.TDF 获取树形工作流在 FT 上的分布;
在 O.TDF FT 的汇集节点上配置 O.DGH, O.XMLD;
在 O.TDF FT 的根节点上配置 O.DGH, O.XMLD;
在 O.TDF FT 的叶节点上配置 O.DG, O.XMLD; }

(5)FT 上的进程配置工作结束

执行算法 4 以后, FT 的每个节点上被配置了 MCP, SENP, RECP 和数据抓取、汇集进程, FT 节点上的进程配置结构如图 2 所示。

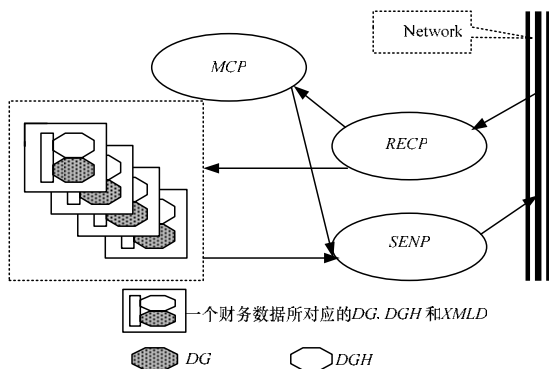


图 2 FT 节点上的进程配置结构

算法 5 FT 上的财务数据抓取和汇集过程

(1)FT 上所有叶节点上的 DG 按用户预定的数据抓取策略(时间频度、数据抓取粒度)对其相应的财务数据 FD 进行抓取, 并形成 FD 对应的 XMLD 表示模式的数据 XFD。该节点上的 SENP 进程根据 FD 的树形工作流的描述把 XFD 发送到其工作流的下游节点(父节点)。

(2)FT 上的所有汇集节点上的财务数据 FD 的 DGH 根据其收到的上游节点(子节点)的财务数据按 XMLD 模式汇总。如果 FD 对应的树形工作流在该汇集节点的同步类型 SYST=1, 则 DGH 需要对 FD 在本地进行同步处理, 并通知本地 SENP 进程发送 FD 的本节点汇总数据到其下游节点(父节点)(如果本汇集节点是根节点, 则不执行该操作)。

(3)FT 上所有节点上 RECP 并行地重复进行接收数据、识别数据并将其发往相应的进程。

(4)FT 上所有节点上 MCP 并行地重复进行接收命令、识别命令、应答命令和执行命令。

(5)FT 的根节点并行地重复监控所有服务器节点的运行、启动阶段命令、管理数据库服务器。

5 分析与实验

5.1 分析

GMFD 模型具有如下特点:(1)采用树形工作流定义财务数据的逐级汇总过程, 采用树形结构连接数据库服务器, 更适合企业集团的现实客观分布结构。可以有效减少远程传输带来的性能下降, 在汇集节点对数据进行预处理, 浓缩发往下游(父)节点的数据量, 减少了下游工作流动量。(2)如果叶节点服务器和根节点服务器的网络距离很大, 则可以在叶节点和根节点间增加汇集节点层次, 并使树形工作流在上述新增加的汇集节点上的同步类型 SYST=1。(3)每一类财务数据拥有一套树形工作流描述、数据抓取进程 DG、数据汇集进程 DGH 以及数据统一表示 XMLD, 极大增加了 GMFD 模型的可扩展性。(4)对财务数据安排独有的 DG, DGH, XMLD, 可以有效支持异构财务系统, 不同财务系统采用的 DG 可以根据其运行环境情况来开发。

5.2 实验

用 7 台计算机作为服务器构建运行 GMFD 的网络环境, 其中, 4 台为 FT 的叶节点, 分别运行用友 U8(A1)、金碟 KIS 标准版(A2)、新亚财务(A3)、一个自主开发的 DOS 环境下的财务软件(A4)。被监管汇总的财务数据为分类账、总账、资产负债表、损益表等。7 台服务器分布在 3 个城市行政区内。A 行政区有 1 台服务器作为 FT 的根。B 行政区有 3 台服务器, 其中, 2 台运行用友 U8 和金碟 KIS 标准版; 1 台作为该区内的汇集节点。C 行政区有 3 台服务器, 其中, 2 台运行新亚财务和自主开发的 DOS 环境下的财务软件; 1 台作为该区内的汇集节点。实验分别在 2 种配置情况下进行:(1)设置汇集节点, 如图 3(a)所示, 此类实验记为 GMFD-GH;(2)没有配置汇集节点, 如图 3(b)所示, 此类实验记为 GMFD-NGH。在 2 种模式下, 上述财务数据的树形工作流对应的树形结构等同于其运行的 FT 结构。为了测试延迟对模型的影响, 设计一个延迟计数器 Delay()。

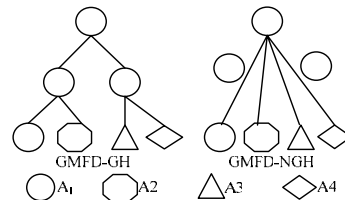


图 3 FT 的 2 种实验结构

在结构为 GMFD-GH 和 GMFD-NGH 的 FT 实验环境下, 分别测试财务数据规模逐渐增加的情况下, 模型的响应时间, 结果如图 4 所示。图 4 横轴为财务数据规模, 其中, 1 表示财务数据为{分类账}; 2 表示财务数据为{分类账、总账}; 3 表示财务数据为{分类账、总账、资产负债表}; 4 表示财务

数据为{分类账、总账、资产负债表、损益表}。此时忽略了网络延迟,由图4数据可知,响应时间和数据规模成线性增加,且GMFD-GH优于GMFD-NGH。说明影响该模型性能的主要因素是数据规模。因此,增加汇集节点可以提高财务数据的汇集效率。

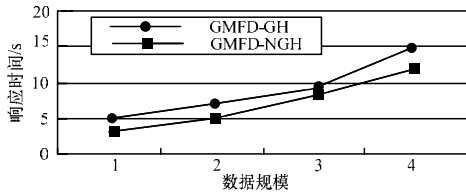


图4 数据规模对GMFD-GH和GMFD-NGH的影响

利用延迟计数器 Delay()人为地增加延迟,分别测试GMFD-GH和GMFD-NGH在延迟为2s(类型1)、4s(类型2)、8s(类型3)、16s(类型4)时的响应时间,结果如图5所示。

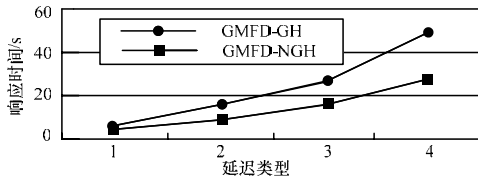


图5 延迟对GMFD-GH和GMFD-NGH的影响

由图5可以看出,延迟对GMFD-GH造成的影响仅增加了延迟时间,而延迟对GMFD-NGH造成的影响随着延迟的增加越来越大。可见,在基于Internet的企业集团财务监管数据的汇集过程中,要得到稳定的响应时间,必须增加中间汇集节点,并适当地对中间数据结果进行浓缩处理。

6 结束语

在基于Internet的集团财务数据监管中,网络延迟和被监管的财务数据规模是影响系统性能的主要因素。异构的财务系统给监管带来了很大困难。本文以财务数据为汇集单元,构造树形工作流并映射到树形结构服务器群上,通过中间汇集节点、XML异构数据解决财务数据汇集问题,为集团财务监管的建设提供了新模式。

参考文献

- [1] 宋宝燕,于戈,葛薇,等.支持复杂应用的工作流并发控制机制[J].东北大学学报:自然科学版,2002,23(1):12-15.
- [2] 李红臣,史美林.工作流模型及其形式化描述[J].计算机学报,2003,26(11):1456-1463.
- [3] Seligman L, Rosenthal A, Lehner P, et al. Data Integration: Where Does the Time Go[J]. IEEE Data Engineering Bulletin, 2002, 25(3): 3-10.
- [4] Cao Junwei, Jarvis S A, Saini S, et al. GridFlow: Workflow Management for Grid Computing[C]//Proc. of the 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid. Tokyo, Japan: IEEE Press, 2003: 55-60.
- [5] Feng Zhilin, Yin Jianwei, He Zhaoyang. Research of Workflow Modeling Based on Concurrent Transaction Logic[C]//Proc. of the 10th International Conference on Computer Supported Cooperative Work in Design. Nanjing, China: IEEE Press, 2006: 1-4.
- [6] 张春海,李忠星.基于扩展ECA的分布式工作流研究与应用[J].计算机工程,2007,33(20):78-79.

编辑 陈 晖

(上接第61页)

当SR-Service推荐一个服务给服务请求者后,服务请求者将根据返回的被推荐服务的信息,通过代理执行调用行为,与此同时SR-Service将会调用PASD模型的观测模块对服务调用过程进行观测,观测过程中系统以提交请求行为对象的形式存储相应的请求,同时系统经由可选的反馈行为来收集反馈信息。该行为描述了代理对返回结果的满意度级别,或者经由调用行为来标记适合请求的服务。如果代理决定使用这些服务中的一个服务的话,系统将会获取更多的信息。观测模块的工作过程如图3所示。

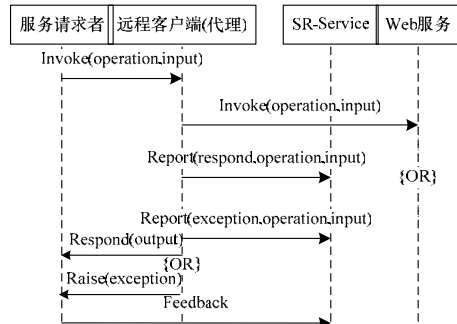


图3 SR-Service服务观测模块工作过程

响应行为用来标记一个服务是可用的,异常行为用来标记一个服务调用异常。当服务请求者收到服务调用的响应消息后,将会产生一个以条件期望行为的外部知识为评判依据

的反馈,如得到的输出是系统所需的,就返回一个认可的反馈。PASD模型是个服务发现的通用模型,具体设计PASD模型系统时,可根据领域的不同,采取不同的领域库。

5 结束语

基于隐式经验约束理论规则的服务发现方法在服务发现的实践中是可行的,尤其是对智能服务发现和服务组合技术的发展能起到积极地推动作用。在今后的工作中可以针对其中的近似度算法作一定的改进,使系统性能进一步得到提高,还能在隐式经验约束理论规则的自动生成方面进一步开展研究工作。与此同时,由于本系统的设计都是在假定服务请求者和发现者之间相互信任的基础上展开的,因此在今后的工作中也可在安全方面对系统做进一步改进和提高。

参考文献

- [1] Sherchan W. A Fuzzy Model for Reasoning About Reputation in Web Services[C]//Proc. of ACM Symp. on Applied Computing. [S. l.]: ACM Press, 2006.
- [2] Claypool M. Implicit Interest Indicators[C]//Proc. of Int'l Conf. on Intelligent User Interfaces. [S. l.]: ACM Press, 2001.
- [3] Birukou E. A Service-oriented Approach to the Development of Recommendation Systems[C]//Proc. of ACM Symp. on Applied Computing. [S. l.]: ACM Press, 2007.

编辑 陈 文

