

迁移 workflow 系统通信机制研究 *

卢朝霞 曾广周 杨公平

(山东大学计算机科学与技术学院 济南 250061)

摘要 迁移 workflow 是解决当前 workflow 管理领域不完全业务过程定义的一个新的研究方向,通信机制是迁移 workflow 系统实现的一个关键问题。本文在分析通信需要解决的主要问题的基础上,提出了一种适应迁移 workflow 系统特征的通信模型,给出了模型框架结构,并详细描述了其中的迁移实例通信算法。模型在实现可靠异步通信的同时,能够保证信件顺序和 exactly-once 提交,降低寻址时间,减小三角路由的带宽占用,增强系统对规模扩大的适应能力,使算法在通信与迁移都较频繁时仍能保持较高的效率。

关键词 迁移 workflow, 迁移实例, 通信, 透明寻址, 信箱 Mailbox

Research on Communication Scheme of Migrating Workflow System

LU Zhao-Xia ZENG Guang-Zhou YANG Gong-Ping

(School of Computer Science & Technology, Shandong University, Jinan 250061)

Abstract Migrating workflow is one of recent highlights to solve the problem of incomplete process specification. Just like other mobile-agent-based systems, communication mechanism is still an important aspect in the research of migrating workflow systems. In this paper, after analyzing the main problems unsolved in communication of migrating workflow, a reliable asynchronous communication model and related algorithms are provided which are aimed at the specific organization characteristics of migrating workflow. The algorithms could ensure reliable communication, but also reduce response time, decrease costs of triangle routing and improve adaptability to the scale enlargement of the migrating system.

Keywords Migrating workflow, Migrating instance, Communication, Transparent addressing, Mailbox

1 引言

迁移 workflow^[1]是为解决当前 workflow 管理中存在的两个问题而提出来的:不完全业务过程定义; workflow 执行环境与流程参数的动态多变造成 workflow 很难维护。针对上述问题的解决,文[2]定义了一种迁移 workflow 模型,模型遵循了利用移动 agent 范型构造迁移实例的技术路线。迁移实例是迁移 workflow 中的行为执行主体,在迁移 workflow 系统中,不同的迁移实例被映射为不同的业务过程,并且可以在不同的工作位置上生成并运行,以动态构建有序结构业务流程。因此迁移 workflow 模型大大提高了 workflow 系统规模定制性以及适应动态环境的灵活性。

迁移实例之间的异步通信作为解决多行为主体合作问题求解的一个重要方面,是迁移 workflow 系统的一个关键问题,它

具有一般移动主体通信的共性,但同时又需要满足特定的系统要求。具体来说,通信算法需要解决的主要问题是:(1)通信容错性和可靠性。算法应能够处理信件的重发与失序以及由于通信目标的迁移而导致的通信失效问题^[3]。(2)效率^[4]与可扩展性^[5]问题。一方面通信算法应尽可能降低地址查询时间和通信响应时间,减少通信带来的额外开销,如地址查询开销,注销/注册开销等;另一方面当系统规模不断扩大时,算法应对服务请求的增多应有较强的适应能力,如能够解决对创建地过于依赖而导致的性能瓶颈,在通信量显著增加的情况下能够保证合理的通信时间和开销。(3)透明寻址^[6]能力。发信方迁移实例应该能够准确而及时地获取收信方的当前位置,而且这种位置映射机制是对被服务方透明的且不因系统规模的扩大而使性能有显著下降。(4)迁移 workflow 应用背景为通信带来的其它问题。如需要保证信件提交的 exactly-

* 国家自然科学基金项目(60473123),山东省科学技术发展计划项目(031110123)。卢朝霞 博士生,主要研究方向为迁移 workflow。曾广周 教授,博士生导师,主要研究领域包括 CSCW、智能计算理论与技术、移动计算及应用技术。杨公平 讲师,在职博士生,主要研究方向为智能计算。

2003. 325~327	Australian HES, 2002
7 Esteves J, Pastor J, Casanovas J. Measuring Sustained Management Support in ERP Implementation Projects; A GQM Approach, 2001	10 Yin R K. Case study research. London Sage Publications, 1994
8 Esteves J, Pastor J, Casanovas J. A Framework Proposal for Monitoring and Evaluating Training in ERP Implementation Projects, 2002	11 Järvinen P. On research methods, Tampere University of Tampere, 1999
9 Laurits J, BinftTech N. Critical success factors for implementing an ERP system in a university environment; A case study from the	12 van Solingen R, Berghout E. The Goal/Question/Metric Method; A practical guide for quality improvement of software development. McGraw-Hill Publishing Company, 1999
	13 Strauss G. The discovery of grounded theory. 1967

once 语义问题^[7]等。

目前众多的对主体间通信的研究中,并没有对上述问题给出较完善的解决方案,许多系统不解决可靠通信问题,仅仅是简单假定通信双方在通信过程中不会发生位置的变化或者将通信失败作为异常交给开发人员来处理,如 Mogent^[8]、Mole^[10]等系统。已有的方法也往往仅仅针对某一个或几个问题的解决,不能充分考虑上述各种需求。此外,目前许多研究采用一种基于 Mailbox 的方法,Mishra^[11]等提出了位置无关的异步通信方法,将 Mailbox 放在 agent Home 实现信件的异步接收,这种方法虽然可以解决通信可靠性问题,但存在着占用带宽较多的三角路由问题,以及 Home 的单元故障和性能瓶颈;文^[8]则将邮箱由 agent 携带,但作阶段性分离,通过预测 agent 下一次迁移可能的通信量来决定 Mailbox 是否迁移,但算法当 agent 通信与迁移都较频繁时显示不出 Mailbox 的优势,而且不能处理信件的重发与失序。

本文在借鉴已有研究成果^[3-5,8]的基础上,针对迁移 workflow 管理系统有管理(即系统中存在一个或多个业务管理域,使得某些业务过程通常只在一定的机构或部门之间发生)的特点,在文^[2]的基础上,定义了一种基于服务域划分的迁移 workflow 系统通信模型,通过改进现有的基于 Mailbox 的通信方案,在保证信件的可靠传递的同时,提高通信效率,增强系统对环境的适应能力和可扩展能力。

本文第 2 节给出系统通信模型的定义及框架结构;第 3 节对主要算法进行了详细描述;第 4 节分析算法的特性及效率;最后给出本文的结论和下一步的工作。

2 系统通信模型与算法设计思想

2.1 通信模型

根据文^[2],在迁移 workflow 系统中,工作位置是迁移实例的运行场所,它为迁移实例提供可靠的本地服务。在 workflow 管理中,机构内或跨机构的工作流程需要按照业务关联度进行组织和管理,从而使得许多业务过程只在一定的范围内发生,换句话说,代表用户执行业务过程的迁移实例的移动与通信范围通常是受限的。因此,在研究迁移实例之间的通信问题时,可以首先按照业务关联度把工作位置划分成不同的服务域,然后再建立域内通信和域间通信的统一模型。一般地说,服务域划分可以遵循下述原则:(1)域内主机提供的资源和服务与域间主机相比具有更高的业务关联度,因而迁移实例访问域内主机的频度比访问域外主机的频度高;(2)域内主机通信与域间主机相比具有更快的速度。基于上述认识,本研究在服务域划分的基础上,对每个服务域设置独立的邮局,从而构成一种多邮局分布式通信模型。

定义 1 设 $Domain = \{d_1, d_2, \dots, d_k\}$ 是迁移 workflow 系统中服务域的集合。域 $d \in Domain$ 是一个五元组 $(Did, WP, Postoffice, Local_MI, Active_MI)$,其中,

Did 为全局唯一的域 d 标识;

$WP = \{wp_1, wp_2, \dots, wp_p\}$ 是域 d 内工作位置的集合;

$Postoffice$ 是域 d 的通信机,称作域邮局;

$Local_MI = \{lmi_1, lmi_2, \dots, lmi_m\}$ 是在域 d 内创建的所有迁移实例的集合;

$Active_MI = \{ami_1, ami_2, \dots, ami_n\}$ 是在域 d 内运行的所有迁移实例的集合,包括 d 内创建并在 d 内运行的迁移实例和从其它域迁入 d 运行的迁移实例两部分。

定义 2 域邮局 $Postoffice$ 是一个五元组 $(Po_id, Po_adr, Mailbox, Address_book, Post_agent)$,其中,

Po_id 为可认证的邮局标识;

Po_adr 为邮局所在工作位置的地址;

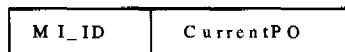
$Mailbox = \{local_Mailbox, active_Mailbox\}$ 是迁移实例的信箱,其中,

$local_Mailbox = \{lmb_1, lmb_2, \dots\}$ 是源信箱集合, $lmb_k \in local_Mailbox$ 是 $lmi_k \in Local_MI$ 的源信箱, $k = 1, 2, \dots$, lmb_k 的生命周期与 lmi_k 等长;

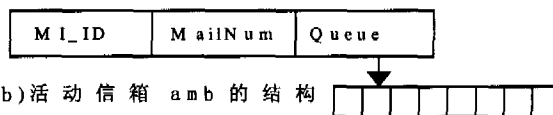
$active_Mailbox = \{amb_1, amb_2, \dots\}$ 是活动信箱集合, $amb_k \in active_Mailbox$ 是 $ami_k \in Active_MI$ 的活动信箱, $k = 1, 2, \dots$, amb_k 的生命周期与 ami_k 在域 d 中的活动时间等长,即 ami_k 在域 d 内启动运行时创建,离开域 d 后注销;

$Address_book = \{ab_1, ab_2, \dots\}$ 是域 d 的熟人地址簿,熟人地址 $ab_k = (MI_ID, CurrentPO)$,其中 MI_ID 表示熟人迁移实例的名字, $CurrentPO$ 表示 MI_ID 当前所在域邮局 $Postoffice$ 的地址;

$Post_agent$ 是邮局的管理者,也是迁移实例的通信代理,它掌握迁移实例当前所处的域状态,并通过对信箱及通信簿的管理实现迁移实例信件的投递与转寄等。



(a) 源信箱 lmb 的结构



(b) 活动信箱 amb 的结构

图 1 Mailbox 的结构

定义 3 源信箱 $lmb \in local_Mailbox$ 是一个二元组 $(MI_ID, CurrentPO)$,如图 1(a)所示,其中,

MI_ID 表示 lmb 所隶属的 $lmi \in Local_MI$ 的名字(命名方法见 2.2,下同);

$CurrentPO$ 表示 MI_ID 当前所在邮局 $Postoffice$ 的地址;

定义 4 活动信箱 $amb \in active_Mailbox$ 是一个三元组 $(MI_ID, MailNum, Queue)$,如图 1(b),其中,

MI_ID 表示 amb 所隶属的 $ami \in Active_MI$ 的名字;

$MailNum$ 是 MI_ID 在域 d 内活动期间接收的邮件数目;

$Queue = \{(\text{信件标识}, \text{序号}, \text{标题}, \text{发送者}, \text{发送邮局}, \text{发送时间}, \text{信件内容})\}$ 是 MI_ID 在域 d 内活动期间接收的邮件列表。

接收者收到信件后,可以根据信件标识确定是否接收过此封信件,以决定接收或拒绝,根据序号将其顺序插入等待队列,从而保证信件的顺序提交。

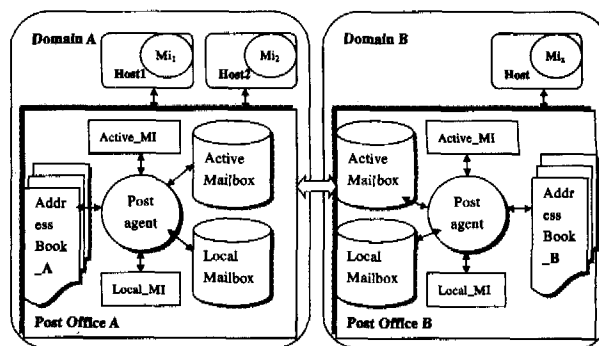


图 2 系统的通信模型

上面定义的通信模型可以用图 2 表示。当迁移实例 mi 被成功创建时,创建主机为其向创建域 Post-agent 提出注册请求。创建域 Post-agent 首先将合法的 mi 在 Local-MI 中注册,然后为 mi 创建一个源信箱 $lmb \in local-Mailbox$,并置 CurrentPO 为创建域邮局。lmb 在 mi 的整个生命周期内一直停留在其创建域邮局中,记录 mi 所经过的邮局地址。迁移实例 mi 启动运行时,所在域 Post-agent 首先将其在 Active-MI 中注册,然后为其创建一个活动信箱 $amb \in active-Mailbox$,以收取 mi 在本域运行时发来的信件。在 mi 迁入新城后,为安全起见,新城 Post-agent 为其重新创建一个活动信箱,再通知原所在域将其 amb 注销,而不是让活动信箱跟随 mi 迁移到新城。

2.2 迁移实例的命名与寻址

迁移实例的名字由三个基本部分组成:创建迁移实例 mi 的域地址 DA (Domain Address)、主机地址 HA (Host Address) 和创建地为 mi 赋予的唯一标识 ID (Identity)。如果网络中同时有多个迁移 workflow 系统运行,则可在迁移实例名字的最前面加上相应迁移 workflow 系统的名称 MWS-Name (图 3 虚线部分),从而方便地实现命名扩展。迁移实例 mi 一旦创建,其名字在整个生命周期内保持不变。

MWS_Name	DA	HA	ID
----------	----	----	----

图 3 迁移实例的名字结构

为实现寻址的快速响应性,减小对创建地的依赖,本文设计了“Domain-Address-book-Home”的寻址机制,迁移实例发信时,按照域内活动迁移实例列表 Local-MI、Address-book 的顺序查找收信方地址,如果找不到,再去查询源信箱。基于域内主机提供服务的相关性,在域内运行的迁移实例所执行的任务也具有相关性,因此设计 Address-book 对本域所有迁移实例所共有,并称其中记录的域外迁移实例为熟人,与此相对应,每个迁移实例都随身携带一张好友域名单,存放曾经通过信的域的地址,其定义如下:

定义 5 如果域 D_i 将 mi 视为熟人,那么称域 D_i 为 mi 的好友域。记 $mi.FriDom = \{D_1, D_2, \dots\}$ 是迁移实例 mi 自身携带的一个好友域名单,其中, D_1, D_2, \dots 是 mi 好友域的域邮局地址,满足下述条件的域 D_i 将被加入到 $mi.FriDom$ 中:

- (1) $mi \notin D_i.Active-MI, D_i.Active-MI$ 为域 D_i 内运行的所有迁移实例的集合;
- (2) mi 与域 D_i 中的至少一个迁移实例完成了通信过程;
- (3) $D_i \notin mi.FriDom$ 。

推论 如果 mi 在域 D 的熟人地址簿中,则 D 一定在 mi 的好友域名单中,反之亦然。

3 主要算法描述

下面讨论的通信算法主要包括:信件发送、转发、接收算法及熟人地址簿管理算法。

3.1 信件发送算法

假设迁移实例 mi_1 向目标迁移实例 $receiver$ 发送信件。为了保证信件可靠送达,信件发送算法需要考虑三种情况:(1) mi_1 和 $receiver$ 在同一个域内运行;(2) mi_1 和 $receiver$ 不在同一个域内运行,但 $receiver$ 是 mi_1 所在域的域外熟人;(3) mi_1 和 $receiver$ 不在同一个域内运行, $receiver$ 也不是 mi_1 所在域的域外熟人。算法描述如下:

```

MailSender(Mail m)
/* 设发信方迁移实例在域 A 运行,收件人当前所在域 S 的邮局管理者为 S.Post-agent。m 为要发送的信件 */
{ receiver=extName(m); //从信头中提取收件人名字
  if find(receiver,A.Active-MI) //情况(1)
    insert(m,receiver,amb);
  else if find(receiver,A.Address-book) //情况(2)
    {S=extrAddr(receiver,A.Address-book);
     //提取收件人所在邮局
    post(m,S,receiver); /* 将信件 m 发往 S,
    S.Post-agent 将信件投递到 receiver 的活动信箱中 */
  }
  else // 情况(3)
    { C=resolve(receiver); //解析收件人地址
     result=query(C,receiver); //查询其当前地址
     post(m,result,receiver);
    //将信件发送至 result 域,投入 receiver 的活动信箱
  }
}

```

3.2 信件转发算法

根据信件发送算法,发给 $receiver$ 的信件将被投递到其活动信箱,如果信件在传递过程中 $receiver$ 正在域间迁移,则算法采用如下措施避免通信失效:在 $receiver$ 向另一个域迁移时,并不立即注销其在迁出域的活动信箱 amb ,而是等 $receiver$ 成功到达并注册了新的活动信箱后再注销。对于在 mi 发生域间迁移时投递到其迁出域活动信箱中的信件,采用信件转发的策略。算法描述如下:

```

MailTransfer(Msg MboxDeregMsg)
/* 当迁移实例 receiver 迁移到达新城,向其迁出域邮局发信箱注销消息时,其邮局管理者执行本算法 */
{ receiver=extName(MboxDeregMsg);
  //提取迁移实例的名字
  curPO=extAddress(MboxDeregMsg);
  //提取迁移实例新城地址
  if receiver.amb.MailNum!=0
    //检查 receiver 活动信箱中的 MailNum 项
    postTo(receiver.amb.queue,curPO,receiver);
  //将所有信件发至 curPO,postTo 为批量投递命令
}

```

3.3 信件接收算法

由信件发送、转发算法可知,迁移实例 mi_i 发给目标收件人的所有信件都被最终投递到 $receiver$ 当前域的活动信箱中,因此 $receiver$ 获取信件的过程完全在当前域内完成。算法采取 Pull^[9]方式投递信件,如下为信件接收算法:

```

MailReceiver(Msg mailReqMsg)
/* 迁移实例 receiver 向所在域 s 的管理者发收信请求,S.Post-agent 执行此算法将信件发至 receiver。 */
{ receiver=extName(mailReqMsg);
  //提取消息发送者的名字
  host=extAddress(mailReqMsg);
  //提取消息发送者的地址
  if receiver.amb.MailNum=0 /* 检查 receiver 的新邮件数,若为 0,则发“无邮件到达信息” */
    sendMsg(host,NoMailMsg);
  else
    postTo(receiver.amb.Queue,host,receiver);
  //将信件全部发给 receiver
}

```

3.4 熟人地址簿管理算法

根据信件发送算法 MailSender,发信方邮局需要根据熟人地址簿快速定位收件人邮局的地址。因此,为了保证熟人地址簿寻址的有效性,必须对熟人地址簿进行两种操作:(1)新熟人项的添加;(2)熟人地址更新。两个迁移实例通信后,会根据定义 5 中的三个条件判断是否需要添加好友域,同时其好友域中的熟人地址簿对应地将该迁移实例的当前所在邮局进行登记,以备以后通信查询。当迁移实例迁至一个新城,其当前邮局发生了改变,要通知所有好友域更新其熟人地址簿中登记的 mi 地址,在实际应用中,由于一个迁移实例负责一个业务流程的执行,任务具有单一性,其域间迁移并不频繁,因此好友域的更新代价不会太高。

```

AddressAdd(MI mi2)
/* mi1 与 mi2 (假设在 B 域) 通信过程结束后执行本
   算法, 以 mi1 所在域的算法执行为例 */
{ if mi2 ∈ Address-book
  addNewItem(mi2, B); /* 在 address-book 中添加新的熟人项, 其
                        中 MI ID= mi2, CurrentPO = B */
}
AddressUpdate(MI mi)
/* 迁移实例 mi 从域 A 迁移到域 X 后, X. Post-agent 执行本算
   法 */
{ for each domain Dk in mi. FriDom
  // 向好友域中的每一项发地址更新消息
  sendUdtInf(Dk, mi, CurrentPO);
}
UdtAdr- Dom(Msg addrUpdMsg)
// mi 的好友域 Dk 收到 mi 的地址更新消息后, 执行本算法, 更
  新 Dk. Address-book 中 mi 项的 CurrentPO
(mi = extName(addrUpdMsg); // 提取迁移实例名字
 dom = extAddress(addrUpdMsg);
 // 提取迁移实例当前所在域地址
 if dom is local /* 如果 mi 当前在本域中, 则置地址簿中 CurrentPO
   为 0 */
   CurrentPO = 0;
 else CurrentPO = dom;
 // 否则更新 CurrentPO 为 mi 的新邮局
}

```

4 算法特性及效率分析

4.1 算法特性分析

性质 1 如果底层通信设施可靠, 算法能够保证信件的可靠提交算法通过两点解决通信失效问题, 一方面当迁移实例在域内活动时, 由于 *amb* 位置固定, 从通信的角度看, 可以认为迁移实例的位置没有发生变化, 信件可以无障碍地发送到 *amb* 内, 然后在合适的时候由迁移实例收取; 另一方面当迁移实例发生域间迁移时, 其在迁出域的 *amb* 并不会立即被注销, 以保证域间迁移期间收到的信件不会发生丢失。

性质 2 如果底层通信设施可靠, 算法能够保证信件提交的 exactly-once 语义和顺序信件在网络中传输时虽然可由相应措施(如 TCP 协议)保证可靠送达, 但由于网络通信状况的不稳定, 可能会导致超时重发或失序, 为此, 算法在接收者一端进行了处理, 通过对信件的初步检查, 抛弃掉被重发的信件, 并且在活动信箱中对信件进行重定序, 从而保证了信件提交的 exactly-once 语义和顺序。

4.2 通信效率分析

影响通信效率的主要因素有: 通信时间与空间(即网络流量)开销。为方便计算, 假定如下: 网络中有 *m* 个域, 每个域中的节点数分别为 $\{n_1, n_2, \dots, n_m\}$, 域间网络带宽的期望值为 δ_{inter} , 域内网络带宽的期望值为 δ_{intra} , 迁移实例 A 向迁移实例 B 发送信件, B 与 A 同域的概率为 ρ_1 , B 在 A 所在域熟人地址簿中的概率为 ρ_2 , B 处于域间迁移状态的概率为 *q*, 信件大小为 C_{mail} , 注销注册等控制消息的大小为 C_{ctl} 。由前述算法, A 与 B 一次通信(A 发 B 收)时间的期望值为:

$$T = \rho_1(t_{intra}) + \rho_2[(1-q)(t_{ur-ml} + t_{intra}) + q(2t_{ur-ml} + t_{intra})] + (1-\rho_1-\rho_2)[(1-q)(2t_{ur-ctl} + t_{ur-ml} + t_{intra}) + q(2t_{ur-ctl} + 2t_{ur-ml} + t_{intra})] \quad (1)$$

其中, $t_{ur-ctl} = C_{ctl}/\delta_{inter}$ 为控制消息的域间传输时间, $t_{ur-ml} = C_{mail}/\delta_{inter}$ 为信件在域间传递时间, $t_{intra} = (C_{ctl} + C_{mail})/\delta_{intra}$ 为 B 从所在域活动信箱 *amb* 收取信件所用时间(包括一次请求消息 C_{ctl} 与一次信件正文 C_{mail} 的传送时间), $q = t_{mgt}/t_{stay} + t_{mgt}$, 其中 t_{mgt} 为域间平均迁移时间, t_{stay} 为域内平均停留时间。由上式可知, 当网络带宽与通信量一定时, ρ_1, ρ_2 越大, 即目标地址在域内以及熟人地址簿中的命中率越高, 对源创建地的依赖越小, *q* 越小, 迁移实例域内停留时间越长, 迁移实

例之间直接通信越多, 需要转发的情况越少, 通信时间也就越短。

通信空间开销包括两部分: 信件传递与域间迁移开销。设 C_{dly} 与 C_{mgt} 分别为迁移实例 A 一次通信传输的信息量和一次域间迁移所产生的通信量。

$$C_{dly} = \rho_1(C_{ctl} + C_{mail}) + \rho_2[(1-q)(C_{ctl} + 2C_{mail}) + q(C_{ctl} + 3C_{mail})] + (1-\rho_1-\rho_2)[(1-q)(3C_{ctl} + 2C_{mail}) + (3C_{ctl} + 3C_{mail})] \quad (2)$$

$$C_{mgt} = (D_{fri} + 2)C_{ctl} \quad (3)$$

信件传递开销的计算同上, 根据(2)式, ρ_1, ρ_2 越大, *q* 越小, 信件传递的信息量越小。迁移实例执行任务期间总的迁移开销为 $C = N_{inter-mgt} C_{mgt}$, 其中 $N_{inter-mgt}$ 为跨域迁移次数的期望值。影响总迁移开销的主要因素有两个: 跨域迁移的频繁程度和通信对象的分散程度, 为此引入两个参量: 跨域迁移率 $\eta = \text{跨域迁移次数} / \text{总的迁移次数}$, 通信分散度 $\xi = \text{通信涉及的域的个数} / \text{总通信次数}$, η 越大, 说明迁移实例跨域迁移越频繁, 注销注册开销越大; ξ 越大, 说明迁移实例通信越分散, 其好友域越多, 则其地址更新开销越大; 反之, 则表示迁移实例的通信和迁移活动较集中, 迁移开销较小。

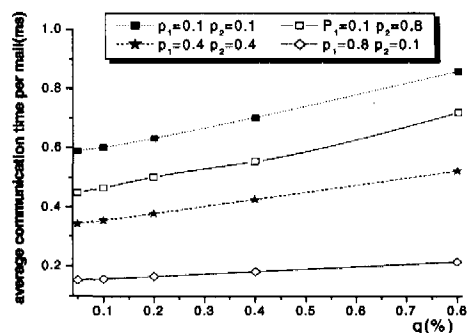


图 4 不同条件下迁移实例单次通信时间比较

为了验证上述分析结果, 对算法进行了模拟实验, 实验假定: 控制消息的大小 $C_{ctl} = 1k$, 信件大小 $C_{mail} = 4k$, $\delta_{intra} = 10\delta_{inter}$ 。图 4 为通信时间受参数 ρ_1, ρ_2, q 变化的影响, 当 $\rho_1 = 0.1, \rho_2 = 0.1$ 时, 信件多数通过源信箱寻址, 当 *q* 较大时(域内停留时间较短, 域间迁移较频繁), 通信时间开销较大。图 5 为模拟迁移实例不同迁移与通信模式下耗费迁移开销的实验结果, 固定迁移总次数为 100 次, 通过产生随机数的方式来控制迁移实例的跨域迁移率 η 保持在 0.05, 0.1, 0.2, 即平均每迁移 20, 10, 5 次跨一域, 每次域间迁移根据通信分散度 ξ 决定更新消息的数目, 系统记录总消息条数。由图可知, 迁移实例的通信和迁移活动越集中, 迁移开销越小。

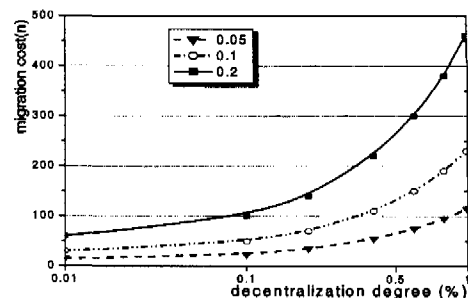


图 5 不同条件下的迁移开销比较

由迁移 workflow 系统的定义可知, 每个服务域提供一定范围的服务, 域内迁移实例执行的任务具有相近性, 由于一个迁

移实例负责一个特定业务流程的执行,其工作性质决定了其域内停留相对时间较长,跨域迁移率较小,通信分散度较小,因此,由上面的分析和实验结果可知,算法对迁移实例间频繁通信和域内频繁迁移有较好的适应能力,能够保证迁移工作流程系统对通信可靠性以及时间和开销等方面的要求。

结论及进一步的工作 本文在分析迁移实例通信需要解决的问题及现有方法不足的基础上,提出了一种针对迁移工作流程系统特定组织模式的通信模型及算法。同现有的方法相比,它具有以下优点:(1)迁移实例的双信箱机制,实现了迁移实例之间可靠异步的通信,保证信件的顺序“exactly-once”提交,并且减小了三角路由的带宽占用和迁移实例的注销注册开销;(2)高效透明的寻址机制,一方面有效降低了寻址时间,减轻迁移实例对出生地的依赖,另一方面能够使算法应对规模扩大有较强的适应性,增强了系统健壮性和效率。初步实验证明算法能够较好地满足迁移 workflow 对可靠性、适应性、快速响应性与效率的特定要求。进一步的工作将在通信安全性方面做深入的研究。

参 考 文 献

1 Cichocki A, Rusinkiewicz M. Providing Transactional Properties for Migrating Workflows. In: IEEE 10th Workshop on database & Expert Systems Applications, Florence, Italy, Sep. 1999. 90~94

2 曾广周,党妍. 基于移动计算范型的迁移 workflow 研究. 计算机学报, 2003, 26(10): 1343~1349

3 陶先平,冯新宇,李新,张冠群,吕建. Mogent 系统的通信机制. 软件学报, 2000, 11(8): 1060~1065

4 杨博,刘大有,杨鲲,张朝辉. 移动 Agent 系统的主动通信机制. 软件学报, 2003, 14(7): 1338~1344

5 Finin T, Labrou Y, Peng Y. Mobile Agents can Benefit from Standards Efforts on Interagent Communication. IEEE Communication Magazine, July, 1998. 50~56

6 王红,曾广周,林守勋. 可移动 agent 系统位置透明通信的一种实现. 计算机学报, 2001, 24(4): 442~446

7 Mishra S, Xie P. Interagent Communication and Synchronization Support in the DaAgent Mobile Agent-Based Computing System. IEEE Transactions on parallel and distributed systems, 2003, 14(3): 290~306

8 Feng XY, Cao JN, Lü J, Chan H. An Efficient Mailbox-Based Algorithm for Message Delivery in Mobile Agent Systems. LNCS2240, Springer-Verlag, 2001, 135~151

9 Cao Jiannong, Feng Xinyu, Lu Jian, et al. Reliable Message Delivery for Mobile Agents, Push or Pull?. IEEE Transactions on Systems, Man and Cybernetics-Part A: Systems and Humans, 2004, 34(5): 577~587

10 StraBer M, Baumann J, Hohl F. Mole—A Java based mobile agent system. In: Proc. Workshop Reader ECOOP'96. dpunkt. Verlag, 1996. 327~334

11 Mishra S, Xie P. Interagent Communication and Synchronization Support in the DaAgent Mobile Agent-Based Computing System. IEEE Transactions on Parallel and Distributed Systems, 2003, 14(3)

(上接第 115 页)

集中式 PUB/SUB 系统通常用可靠的点到点通道来实现发布者与订阅者之间的通信,并在稳定的存储系统中保留事件备份。因此,即使集中式事件代理可能由于延时传送而失败,但事件仍然可以可靠地提交给所有的订阅者。基于一个分布式事件代理的网络系统,通常使用可靠的协议向所有的或者是部分代理传输事件。基于组通信的协议和可靠的应用层多路传输是很好的选择。发布者和订阅者通常使用点对点的通信通道来与最靠近它们的代理进行通信。

结束语 PUB/SUB 是可适用于可扩展要求高、松散耦合系统的分布式交互模型。在此总结了基于三个方面的分类:时间非耦合、空间非耦合和同步非耦合。非耦合是很重要的特征,因为它强调了在抽象层的可扩展性,允许参与者不依赖于另一个而独立操作。然而,在实现层,可扩展性仍存在问题,因为 PUB/SUB 交互可建立在各种通信机制的基础上,容易被不恰当的体系结构所牵制。可扩展性通常与其它特性相冲突,如灵活的订阅要求复杂且开销大的过滤和路由算法。类似地,由于要保留事件日志且丢失的事件要求能被侦听到并可以重传,高可靠性保证就要求很高的开销。特别为广域网开发的协议也不能很好地扩展到订阅者数量大的系统,这是由于消息认可产生的网络流量很大,因而系统资源的开销也就大了。因此,基于事件的 PUB/SUB 中间件的开发

与利用在一定程度上可以提高系统的效率。

参 考 文 献

1 Carzaniga A, Rosenblum D S, Wolf A L. Achieving scalability and expressiveness in an internet-scale event notification service, 1999

2 Eugster P Th, Felber P, Guerraoui R, et al. The Many Faces of Publish/Subscribe. ACM Computing Surveys, 2003, 35(2): 114~131

3 Banavar G, Chandra T, Mukherjee B, et al. An Efficient Multicast Protocol for Content-Based Publish-Subscribe Systems. In: Proceedings of the 19th International Conference on Distributed Computing Systems (ICDCS'99), Austin, 1999

4 Starovic G, Cahill V, Tangney B. An Event Based Object Model for Distributed Programming. In: Proc. of the Intl. Conf. on Object Oriented Information System, London, 1995. 72~86

5 Pietzuch P R, Bacon J M. Hermes: A Distributed Event-Based Middleware Architecture. Submitted to the Workshop on Distributed Event-Based Systems (DEBS), 2002

6 Cugola G, Nitto E D, Fuggetta A. Exploiting an Event-Based Infrastructure to Develop Complex Distributed Systems. In: Proceedings of the 20th International Conference on Software Engineering (ICSE'98), Kyoto, 1998

7 汪洋. 分布事件通知服务的关键技术及其应用研究:[博士论文]. 中国科学院软件研究所, 2004

8 汪洋,王振宇. 一个支持 QoS 的实时 CORBA 中间件 ORB 的结构模型. 计算机科学, 2002, 29(4)