

# 移动 Agent 系统名字解析机制的研究与实现\*

王申康, 滕敏堂

(浙江大学 计算机科学与工程系, 浙江 杭州 310027)

E-mail: wsk@mail.hz.zj.cn; teng\_mt@163.net

http://www.zju.edu.cn

**摘要:** 名字解析机制是移动 Agent 技术研究的重要课题。目前,多数基于域名系统的名字解析机制由于算法自身的复杂性限制,很难实现移动对象的应用级位置透明性访问。全局的、与位置无关的命名和名字解析极大地方便了移动 Agent 系统通信模型和导航模型的建立,具有良好的实用性。通过引入一种基于全局命名模式的名字解析机制,为移动 Agent 系统建立了一种全局统一的命名方法——UON(universal object naming),并从对象的注册、对象的移动和移动对象的访问 3 个方面给出了名字解析的具体实现过程和相关算法,从而有效地解决了移动 Agent 系统应用级的位置透明性问题。

**关键词:** 移动 Agent;名字解析;通用对象命名;对象注册;对象移动;对象访问

**中图法分类号:** TP311 **文献标识码:** A

移动 Agent 技术从 20 世纪 90 年代初期产生,至今已获得了迅猛的发展。但是随着研究的深入,特别是在面向应用的研究中,也暴露出许多问题,其中之一就是移动 Agent 系统中的名字解析机制的问题。移动 Agent 系统包括各种各样的对象,如 Agent、Agent 服务器、资源和用户等,系统必须提供名字解析机制定位一个给定名字的对象的位置的机制。名字解析机制的任务首先是为每个对象分配一个名字,用以唯一地识别它们;其次是在对象发生交互的过程中能够根据名字准确地定位对象。

目前大多数移动 Agent 系统常用的名字解析机制是基于主机名字和端口号对对象进行命名,通过域名系统(domain name system,简称 DNS)来完成名字解析<sup>[1]</sup>,如 Ajanta<sup>[2]</sup>,Agllets<sup>[3]</sup>,Tacoma<sup>[4]</sup>等移动 Agent 系统均采用这种名字解析机制。显然,采用基于主机名字和端口号机制对于静态对象(如移动 Agent 服务器或资源)的命名和名字解析是非常有效的,因为这种机制在唯一命名对象的同时,自身也包含了定位对象的路由信息。但是,对于移动对象(移动 Agent)却存在很大的缺陷,典型的是,当一个 Agent 移动时,它的名字需要同步的改变以反应其最新的位置,这就使应用跟踪 Agent 的移动过程变得非常复杂。因此,在应用级上系统应该提供位置透明的命名和名字解析机制。目前,存在两种解决方法,第 1 种是为移动对象提供本地代理(proxy),把对象的位置封装起来,当对象移动时,系统在本地代理中修改其位置信息,这样,就在应用级实现了位置透明性,Voyager<sup>[5]</sup>移动 Agent 系统就采用这种命名和名字解析机制,但是其静态对象仍使用位置相关的 DNS 命名方式。另一种方法是使用全局的、与位置无关的命名,即当对象的位置发生变化时名字不会随着改变,但是在名字解析时需提供名字服务,把一个符号名字映射到命名对象的当前位置。这就是基于全局命名模式的移动 Agent 系统名字解析机制。由于全局的、与位置无关的命名使对任何对象(尤其是移动对象)的访问都可以以一种统一的方式进行,因此对移动对象的跟踪和通信可以完全实现位置透明性,这极大地方便了移动 Agent 系统导航模型和通信模型等的建立,显然它是解决移动 Agent 系统中名字解析问题的最佳选择。

本文介绍了一种基于全局命名模式的名字解析机制——UON,有效地解决了移动 Agent 系统应用级的位

\* 收稿日期: 2000-05-16; 修改日期: 2000-10-16

作者简介: 王申康(1945 - ),男,上海人,教授,博士生导师,主要研究领域为人工智能,机器学习/专家系统,多媒体/超媒体技术;滕敏堂(1972 - ),男,山东烟台人,博士生,主要研究领域为移动 Agent,软件体系结构,网络管理。

置透明性问题;同时,从对象的注册、对象的移动和移动对象的访问等 3 个方面给出了名字解析的具体实现过程和相关算法。

## 1 通用对象命名方法

为了满足位置无关的命名原则,我们采用了一种称为通用对象命名 UON(universal object naming)的命名方法.通用对象命名是一种静态的与位置无关的标识符,可以用于任何对象的访问和对象属性信息的查询.通用对象命名的语法可以简单描述如下:<UON>:<NSI>:<DON>/<HON>/<OBN>,其中:

UON: 通用对象命名,说明采用的命名方法,类似于 http 地址表示中的 WWW.

NSI: 名字空间标识,用于标识不同的移动 Agent 系统.一般情况下,一个移动 Agent 系统拥有一个名字空间.

DON: 名字空间中某一对象域的域名.

HON: 某一主机的名字,在所在的名字空间中是惟一的.

OBN: 对象名称,在所在的主机上是惟一的.

<DON>/<HON>/<OBN>类似于互联网 URL(uniform resource locator)的编址方法.下面我们给出一个通用对象命名的例子:

```
uon:pns.cs.zju.edu.cn/router_server/router_agent_server
```

其含义为:采用通用对象命名的命名方法,在基于应用的移动 Agent 系统名字空间中,在浙江大学计算机科学与工程系的一台名为 router\_server 的路由服务器主机上运行一个管理路由器的 Agent 服务器对象.

在移动 Agent 系统中,所有的对象分为静态对象和动态对象两种,前者如资源、移动 Agent 服务器等;后者如移动 Agent 等.对于静态对象,由于其位置固定,只要在为其命名时保证全局惟一即可,这相对于位置在不断改变的动态对象要简单得多.在我们的系统中,由于采用了类似于互联网 URL 的编址方式,因此对静态对象的名字解析完全可以按照 DNS(域名服务)的方式进行.下面我们重点说明如何利用通用对象命名为动态对象命名的原理.

通用对象命名方法是以域为单位对对象进行命名管理的,即在每个域中存在一个域服务器,负责为本域内对象的命名.一个域服务器可以认为是一个功能增强的 Agent 服务器,与普通 Agent 服务器不同的是它负责域中所有对象的名字管理(命名和名字解析,另外,域服务器的名字管理还包括公钥分配,即根据对象的 UON 返回其 DSA 公钥,对象访问权限的控制等),而且往往是一台具有永久网络连接的主机.

在每个域服务器上有一个 HR(原籍寄存器)和 VR(访问者寄存器),HR 存放了属于本域的所有移动 Agent 的描述信息,其中最主要的是其原籍 UON 和其当前所在域的域 UON(前面例子 uon:pns.cs.zju.edu.cn/router\_server/router\_agent\_server 中的 uon:pns.cs.zju.edu.cn 为域 UON);VR 中存放了当前正在访问本域的移动 Agent 的信息(移动 Agent 可能来自本域或其他域),包括 VR 为当前访问者分配的新的 UON(称为访问者 UON)和其原籍 UON 等信息.需要说明的是,访问者 UON 的命名与原籍 UON 的命名方式完全相同,这便于名字的统一解析.另外,原籍 UON 在对象的整个生命周期内是不变的,这样对对象的访问者实现了全局的名字服务,即对访问者来说,移动 Agent 的具体位置是透明的.访问者 UON 则随着对象位置的变化而变化,由对象当前所在的主机和域服务器的 VR 进行动态地维护.

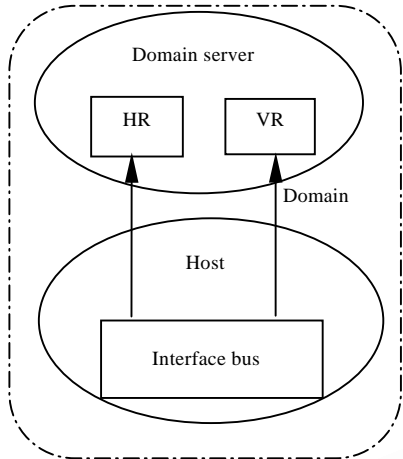
## 2 名字解析

### 2.2 对象的注册

对象的注册过程比较简单,如图 1 所示.移动 Agent 需要向本域的域服务器进行对象注册,域服务器根据其命名规则为当前对象分配全局惟一的 UON——原籍 UON,其域 UON 则为当前域,并将其存入 HR 中.同时,域服务器还将为其分配一个访问者 UON(实际上,这时的访问者 UON 与原籍 UON 相同),并将访问者 UON 与原籍 UON 的对应关系一并存入 VR.

注册过程是主机调用域服务器的 HRRegistration 接口服务实现的(HRRegistration 接口中同时调用了

VRAdd 接口),域服务器的接口服务如下:



域服务器, 域, 主机, 交互总线.

Fig.1 Registration of object

图 1 对象的注册

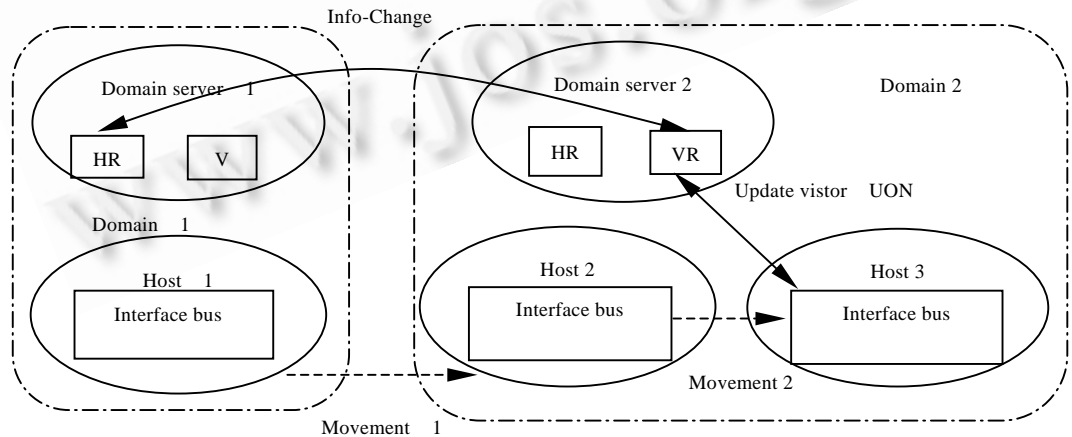
```
interface DOMAIN
{
Public:
    UON HRRegistration (OBJECT obj);
    OBJECT GetObjectInfo(UON uon);
Private:
    BOOLEAN HRDomainUpdate(OBJECT obj);
    UON GetHRDomainUon(OBJECT obj);
    UON VRAdd(OBJECT obj);
    BOOLEAN VRUpdate(OBJECT obj);
    BOOLEAN VRDelete(OBJECT obj);
    UON GetVRUon(OBJECT obj);
    ...
};
```

- 接口 HRRegistration 和 GetObjectInfo 是一组相关操作,前者用于对象向域服务器的 HR 注册,返回对象的 UON;后者则可以根据对象的 UON 返回对象描述信息.这两个接口是开放的(public),任何外部对象或应用都可以访问,并且在访问过程中无须考虑被访问对象的位置,即访问方式是全局统一的.
- 接口 HRDomainUpdate,GetHRDomainUON 用于更新或获取当对象在域间移动时原籍域和目的域间的映射关系,这些是针对 HR 的操作.
- 接口 VRAdd,VRUpdate,VRDelete,GetVRUon 主要用于管理对象在域内部移动时原籍 UON 和访问者 UON 之间的映射关系.

显然,后 6 种接口服务是 DOMAIN 内部对象的管理,即 DOMAIN 为实现应用级对象位置透明性,需要动态地维护对象的位置和 UON 的映射关系.由于这些服务是在应用级之下完成的,有时也称其为系统级服务,其属性是私有的(private).

### 2.2 对象的移动

对象的移动分两种情况,如图 2 所示.



信息交换, 域服务器, 域, 主机, 交互总线, 移动, 更新访问者.

Fig.2 Movement of object

图 2 对象的移动

第 1 种情况是移动 Agent 在域的内部移动,如图 2 中的移动 2.接收移动 Agent 的目的主机 3 在成功地接收当前移动 Agent 后,需要通知本地域服务器 2,域服务器 2 将动态地更新移动 Agent 的访问者 UON(反映出移动 Agent 当前所在的最新位置)以及和原籍 UON 的对应关系,这是域服务器 2 通过调用的 VRUpdate 接口服务完成的.

第 2 种情况是当移动 Agent 在不同域之间移动时,如图 2 中的移动 1.接收移动 Agent 的目的主机 2 在成功接收当前移动 Agent 后,需要通知本域的域服务器 1,域服务器 1 根据其位置信息自动地为其建立访问者 UON,并连同移动 Agent 的原籍 UON 一并存入 VR.同时,目的域服务器 2 还必须通知移动 Agent 的原籍域服务器 1,同步更新当前移动 Agent 的域 UON(原籍域服务器通过调用 HRDomainUpdate 接口服务更新 HR),需要的话,目的域服务器 2 还可能从原籍域服务器 1 的 HR 提取有关移动 Agent 的其他信息,如其拥有者的信息或鉴权信息等(调用域服务器的 GetObjectInfo 接口服务完成).另外,移动 Agent 先前所在的域服务器 1 则自动将其访问者 UON 和相关信息从 HR 中删除.

### 2.3 移动对象的访问

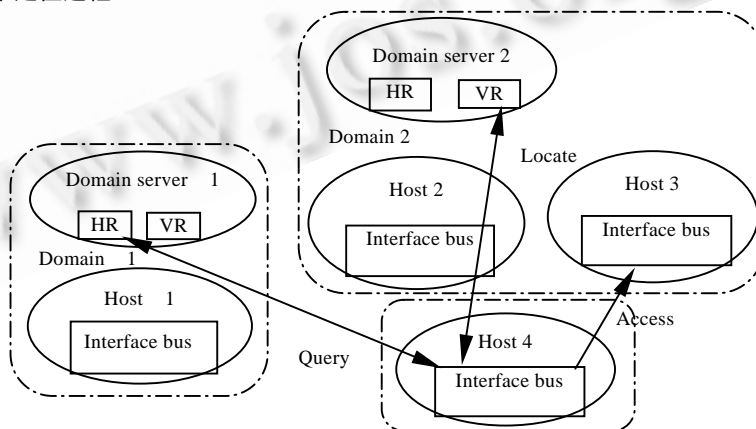
当系统中的某一对象(如位于某一主机上的移动 Agent)需要与某个移动 Agent 通信时,在通信的发起阶段,发起方所在 Agent 服务器的通信服务模块(通信服务模块是组成移动 Agent 系统基础设施的模块之一,它为系统中不同对象间的交互/通信提供服务支持.组成系统基础设施的模块还有代理模型、导航模块、生命周期模型等<sup>[6]</sup>)需要通过名字解析服务首先定位目的移动 Agent.

具体的定位算法如下:

首先,发起方的通信服务模块在本地主机上查找目的对象,这个过程比较简单,主要是通过本地主机的交互总线上寻找来完成的(在我们的移动 Agent 系统中采用了 CORBA 的 ORB 通信总线作为系统的基础通信设施).

若寻找失败,发起方主机请求本域域服务器在本域中查找目的移动 Agent.域服务器根据对象的原籍 UON 调用 GetVRUon 进行查找,一旦成功查找到移动 Agent 的访问者 UON,发起主机则根据域服务器返回的对象访问者 UON 直接与目的对象建立路由连接.

当目的移动 Agent 与通信发起方主机处于不同的域时,如图 3 所示.其定位过程比较复杂.这时,通信发起方主机 4 根据目的移动 Agent 的原籍 UON 首先路由至所在的原籍域服务器 1,原籍域服务器 1 通过调用 GetHRDomainUon 接口服务获得目的移动 Agent 当前所在目的域 2 的位置;发起方主机根据返回的域 UON 改变路由至目的域服务器 2,由目的域服务器 2 调用 GetVRUon 接口服务,查找目的移动 Agent 所在的目的主机 3,从而最终完成整个定位过程.



域服务器, 域, 主机, 交互总线, 定位, 查询, 访问.

Fig.3 Access of mobile object

图 3 移动对象的访问

### 3 算法可靠性分析

建立全局的、与位置无关的命名和名字解析机制的目的是为了能够在各种环境中透明、快速地访问移动对象.由于实际应用环境的复杂性,如目前典型的应用环境 Internet,是一种复杂的异构分布式环境,在这种环境下,网络的可靠性和通信带宽是很难保证的.在我们的算法中充分考虑了这一点,如在第 2.2 节 Agent 移动过程中,只有当 Agent 跨域移动时才需要同步地通知其原籍域服务器更新 HR 中的域 UON,这无疑会大大减少不同域间的通信开销.网络流量,因为 Agent 在域内移动的概率要比其跨域移动的概率高得多;而域内的移动由于范围较小,并且往往是基于某一局域网络的移动(网络的可靠性和网络带宽可以得到保证),因此移动定位算法的可靠性可以得到有效保证.

### 4 结 论

全局的、与位置无关的命名和名字解析机制有效地解决了移动 Agent 系统应用级的位置透明性问题,极大地方便了系统通信模型和导航模型的建立.文中通过引入一种基于全局命名模式的名字解析机制,为移动 Agent 系统建立了一种全局统一的命名方法——UON,并实现了从 UON 到命名对象当前位置的映射,最后从对象的注册、对象的移动和移动对象的访问等 3 个方面给出了名字解析的具体实现过程和相关算法.

#### References:

- [1] Mockapetris, P. Domain names—concepts and facilities. Technical Report, RFC 1034. 1987. <http://www.cis.ohio-state.edu/htbin/rfc/rfc1034.html>.
- [2] Gray, R.S. Agent Tcl: a flexible and secure mobile agent system. In: Proceedings of the 4th Annual Tcl/Tk Workshop (TCL'96). New York: Springer-Verlag, 1996.
- [3] IBM, Inc. IBM aglets documentation web page. 1998. <http://aglets.trl.ibm.co.jp/documentation.html>.
- [4] Johansen, D., van Renesse, R., Schneider, F.B. Operating system support for mobile agents. In: Proceedings of the 5th IEEE Workshop on Hot Topics in Operating Systems (HotOS-V). Los Alamitos, CA: IEEE Computer Society Press, 1995. 42~45.
- [5] ObjectSpace, Inc. ObjectSpace voyager core package technical overview. Technical Report, ObjectSpace, Inc., 1997. <http://www.objectspace.com/voyager/VoyagerTechOverview.pdf>.
- [6] Green, S. Software Agent: a Review. 1997. <http://www.cs.tcd.ie/AI-Labs/Agent/pubreview.pdf>.

## Research and Implementation on a Name Resolution Mechanism of Mobile Agent System\*

WANG Shen-kang, TENG Min-tang

(Department of Computer Science and Engineering, Zhejiang University, Hangzhou 310027, china)

E-mail: wsk@mail.hz.zj.cn; teng\_mt@163.net

<http://www.zju.edu.cn>

**Abstract:** Name resolution mechanism is a very important field in the research of mobile agent technology. At present, for complexity of the algorithm, it is difficult for most traditional DNS-based name resolution mechanisms to implement position independent access of mobile object at application level. But it is convenient to construct communication model and navigation model for mobile agent system using global and position independent name resolution. In order to build a global/universal naming method and resolve position transparent problem at application level in mobile agent system, a name resolution mechanism——UON (universal object naming) is presented in this paper. Finally the detailed implementation and the related algorithm are described in three aspects (object registration, object move and mobile object access).

**Key words:** mobile agent; name resolution; UON (universal object naming); object registration; object move; object access

\* Received May 16, 2000; accepted October 16, 2000