

Java 平台下基于移动 Agent 的信息服务体系

温惠容, 王志良, 牟世堂

(北京科技大学信息工程学院, 北京 100083)

摘要: 运用由纯 Java 语言开发的 Aglet 平台, 进行基于移动 Agent 的信息服务体系的研究, 分析了信息服务体系中移动 Agent 主体结构的实现。该系统实现了移动 Agent 和 Java 结合, 充分发挥了移动 Agent 智能决策、动态控制等特点在信息服务体系中的应用优势。实验表明, 该文提出的信息服务体系是实用有效的。

关键词: 移动 Agent; 信息服务; Aglet

Information Service System Based on Java Mobile Agent

WEN Huirong, WANG Zhiliang, MOU Shitang

(School of Information Engineering, University of Science and Technology Beijing, Beijing 100083)

【Abstract】 This paper studies the information service system based on mobile Agent using the Aglet workbench which is designed in Java. And it analyses how to implement the principal structure of the mobile Agent in this information service system. This system carries out the combination of the mobile Agent and Java. And in the information service system, it sufficiently plays the application advantages of its features such as intelligence strategy, dynamic control and else. The experimental result shows that the information service system proposed in this paper is practical and efficient.

【Key words】 Mobile Agent; Information service; Aglet

1 概述

1.1 移动 Agent 技术

Agent 的研究起源于人工智能领域。Agent 是指模拟人类行为和关系、具有一定智能并能够自主运行和提供相应服务的程序。随着网络技术的发展, 可以让 Agent 在网络中移动并执行, 完成某些功能, 这就是移动 Agent 的思想。

20 世纪 90 年代初, General Magic 公司在推出其商业系统 Telescript 时第一次提出了移动 Agent 的概念, 即一个能在异构网络环境中自主地从一台主机迁移到另一台主机, 并可与其它 Agent 或资源交互的软件实体。由于移动 Agent 可以在异构的软、硬件网络环境中自由移动, 因此这种新的计算模式能有效地降低分布式计算中的网络负载、提高通信效率、动态适应变化了的网络环境, 并具有很好的安全性和容错能力。

由于移动 Agent 技术的优越性, 因此它被广泛应用于电子商务、分布式信息检索、信息发布、个人助手、安全中介、电信网络业务等各个领域。目前移动 Agent 已经进入实用的开发阶段, 并出现了一些移动 Agent 系统的开发平台或执行环境, 比如 General Magic 公司的 Odyssey、IBM 公司的 Aglet、Recursion 公司的 Voyager 等。

Java 是一种面向对象的编程语言, 是移动 Agent 完美的表达语言。Java 的平台独立性、安全机制的改进、动态类库装载、多线程编辑、对象连续性和信息反馈等特点, 都非常符合移动 Agent 的要求, 可以很好地完成移动 Agent 的设计目的。

1.2 基于移动 Agent 的信息服务

现在提供信息服务的手段匮乏, 而且传统的信息服务方式虽然可以在网络中搜索到非常多的信息, 但是很多时候信

息是杂乱无章的, 查询者仍然只能毫无头绪地在大量的信息中碰运气。而基于智能移动 Agent 技术的 Web 信息服务系统, 由 Agent 用户、Agent 宿主系统、Web 服务系统、Agent 转接系统 4 部分组成。Agent Host 和移动 Agent 取代了以往的服务器和客户端。Agent 用户需要搜寻信息时, 用户系统自动创建搜寻移动 Agent, Agent 根据用户需求转移到相应的宿主系统, 并通过 Agent 主机访问 Web 服务器资源。通常宿主系统和资源服务系统在同一台服务器中, Agent 对资源的访问只是同一系统中进程之间的通信, 无须经过网络的传送。由于 Agent 主机是资源服务器上的移动 Agent 支撑环境, 不影响原有的资源服务, 因此系统仍然能够提供系统的 Web 服务, 保持了与传统 Web 系统的兼容。Agent Dock 服务器是为了适应低可靠性和解决拥塞而设置的 Agent 转接系统, 能够向 Agent 提供驻留服务。通过对移动 Agent 的编辑设定, 可以在取得信息的时候进行智能化的处理, 从而使信息符合用户的要求。

2 系统研究

本文中的信息服务系统是以旅游信息服务作为研究背景, 移动 Agent 是系统的信息交互主体。

2.1 系统结构

系统基本框架如图 1 所示。

基金项目: 北京市“现代信息科学与网络技术”重点实验室基金资助项目(TDXX0503); 北京科技大学校重点基金资助项目

作者简介: 温惠容(1980 -), 女, 硕士生, 主研方向: 普适环境下信息服务系统中的多智能体研究; 王志良, 教授、博导; 牟世堂, 教授

收稿日期: 2006-01-17 **E-mail:** happyfairly@163.com

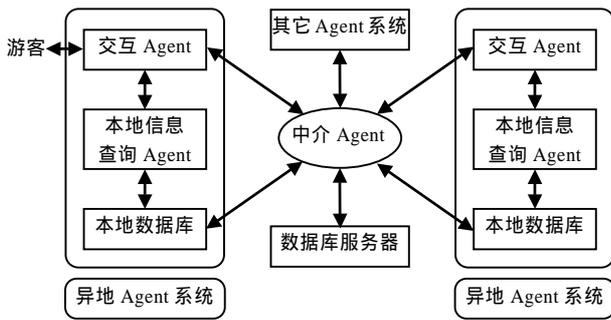


图 1 信息服务系统结构

其中：

(1)交互 Agent：接收游客的要求后，通过基本处理后把信息发送给本地信息查询 Agent。

(2)本地信息查询 Agent：对任务进行分析后，到本地数据库中进行搜索，如果得到所要求的查询结果，则返回所要求的信息给交互 Agent，显示给游客，如果没有符合要求的信息，则返回失败信息给交互 Agent，提示与中介 Agent 的联系。

(3)本地数据库：保存本地(本地地理、服务等信息分类)的信息，并通过中介 Agent 与数据库服务器保持信息一致以及实时更新。

(4)中介 Agent：它在系统中主要完成传递查询信息和查询结果，中介 Agent 得到查询要求后，在系统中迁移，并与各个交互 Agent 通信，得到查询结果后返回到中介 Agent，用中介 Agent 返回信息给信息请求源。中介 Agent 还需要建立数据库服务器与各个本地数据库的联系，来保持数据信息的一致和实时更新。

2.2 Aglet 系统

系统移动 Agent 实现是采用 IBM 公司的 Aglet。Aglet 是由 IBM 公司采用纯 Java 语言开发的移动 Agent 技术，并提供运作平台——Aglet Workbench，在平台下可以开发和测试移动 Agent 系统。Aglet 是一个功能比较全面的系统，它提供了一个简单的移动 Agent 编程模式，为移动 Agent 提供了动态和有效的通信方法，而且具有一套详细又容易操作的安全机制，Aglet 是以线程的形式产生于一台计算机上，可以按照设定，随时暂停正在执行的工作，并把整个 Aglet 迁移到另一台计算机上，执行完任务之后，又返回原计算机，因为是线程，所以不会消耗过多的系统资源。Aglet 的系统框架如图 2 所示。

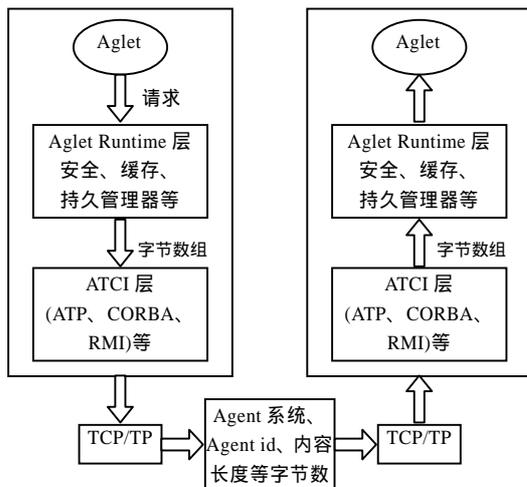


图 2 Aglet 系统示意图

在 Aglet 中设计移动 Agent 系统，设计模式主要有 3 大类，每一类中又包括一些子类^[2]：

(1)巡游模式(Traveling Patterns)：包括 Itinerary, Forwarding 和

Ticket 这 3 种；

(2)任务模式(Task Patterns)：包括 Master-Slave 和 Plan 两种；

(3)交互模式(Interaction Patterns)：包括 Meeting、Locker、Messenger、Finder 和 Organized Group 共 5 种。

在设计中，不同的移动 Agent 交互采用不同的模式编程，在设计分析中再作详细描述。

Aglet 的传输协议使采用由 IBM 提出的 ATP 框架结构(ATP framework)^[3]，它定义了一组原语性的接口和基础消息集，可以看作是一个 Agent 传输协议的最小实现，基本操作如图 3 所示。



图 3 Agent 的基本操作示意图

2.3 系统分析

2.3.1 本地信息处理

在本地信息处理过程的设计中，采用交互模式。本地的驻留 Agent 通过 Meeting 模式在本地主机进行信息交互。这里运用 Aglet 系统中的消息类(The Message Class)实现交互 Agent、本地信息查询 Agent 和本地数据库的信息联系。首先，获得查询要求后，要向 Aglet 代理服务器发出信息：

```
public Object AgletProxy.sendMessage(Message message)
```

Aglet 与 Aglet 之间不是直接交流的，它的方法是通过代理服务器(Proxy)完成交互的，代理服务器就像一道防火墙，避免对 Aglet 的直接访问，同时使 Aglet 的位置信息透明化，这样就保证了 Aglet 之间的安全通信。

通过交互 Agent 对查询关键词的处理后，把信息发送给本地信息查询 Agent 进行查询工作，下面给出基本的信息传递程序部分：

```
public class MessageSend extends Aglet {
    public void run() {
        try {
            AgletProxy proxy = getAgletContext().createAglet (
                getCodeBase(),
                "MessageReceive", null);
            try {
                proxy.sendMessage ( new Message ( "message" ));
            } catch ( Exception e ){
                // Receive failed to handle the message.
            } } catch (Exception e ){
                // Failed to create the receive.
            } } }
    }
```

2.3.2 异地信息处理

当本地的数据库不能满足查询要求时，本地信息查询 Agent 就返回失败提示给交互 Agent，这时交互 Agent 创建一个中介 Agent，中介 Agent 采用巡游模式来完成信息查询。这时就涉及到对 Agent 的控制问题。要在异地主机中获取信息的中介 Agent 可以有两种方式运作。一种是只产生一个中介 Agent，该 Agent 在各个主机中迁移，对每个异地交互 Agent 发出信息请求，在得到正确的信息后返回本地主机，把信息

传递给本地交互 Agent 显示给用户。另一种方式是对每一个异地主机产生一个中介 Agent,没有获得正确信息的子 Agent 在异地主机中自动消亡,获得正确信息的 Agent 则对本地交互 Agent 返回结果,由交互 Agent 输出给用户。下面给出中介 Agent 迁移到异地主机中完成信息请求的基本程序结构:

```
public class MobilityExample extends Aglet{
    public void onCreate ( Object o ) {
        addMobilityListener (
            new MobilityAdapter() {
                public void onArrival ( MobilityEvent e ) {
                    doTask();
                } } );
    }
    void doTask() {
        ... // define the task.
    } }
}
```

2.3.3 数据库信息处理

各个主机的数据库只包含本地的部分信息(比如地理区域信息等),而数据库服务器则包含该信息服务系统所需要提供的全部信息,在信息变更时,首先变更数据库服务器的信息,由数据库服务器产生升级 Agent,对相应的本地数据库进行数据库更新。下面给出升级 Agent 的基本程序结构:

```
public class UpdateFile extends Aglet{
    URL destination = null;
    File dir = null;
    String from = null;
    String to = null;
    public void onCreate(Object args){
        destination = (URL)((Object[])args)[0];
        dir = (File)((Object[])args)[1];
        from = (String)((Object[])args)[2];
        to = (String)((Object[])args)[3];
        addMobilityListener(
            new MobilityAdapter(){
                public void onArrival(MobilityEvent e){
                    replace(args.file,args.from,args.to);
                    dispose();
                } } );
        try{
            dispatch(args.destination);
        }
```

(上接第 68 页)

```
nshort log_nshort[32];
nint log_nint[32]; }LOG;
```

将每一条日志的字段,填入 LOG 结构体中,这样就可以做到对不同日志的统一管理。

3 结束语

根据目前的网络环境现状,提出了一个跨平台综合性的日志审计系统的设计,使得网管人员从纷繁芜杂的日志中寻找网络安全隐患成为可能。不过,目前的设计方案中还没有完全考虑到与审计相关的数据挖掘,因此系统还需要进一步完善。但可以预见的是,在今后的网络环境中,使用日志审计系统来减轻网管人员的工作是十分必要而有效的手段。

```
}catch(Exception e){
    system.out.println("Failed to dispatch.");
} }
void replace(File file,String from,String to){
    ...//define the file used to update
} }
```

2.3.4 信息系统前台交互

移动 Agent 作为此信息服务系统的主体部分,承担信息服务最主要的任务,需要完成信息传递、查询、处理分析和更新等工作,而信息服务系统不能只有后台运作的 Agent 程序,还需要有前台与用户交互的界面。本文中的系统采用 Apache 的 Tomcat 服务器和 SOAP 服务器构建 Web 服务器,架构 B/S 结构的前台 Web 信息交互^[4],并以 Java 作为开发主体语言,这样就使整个信息服务系统的前台界面与后台移动 Agent 运作可以比较容易地结合在一起。

3 结论

本文研究的信息服务系统运用 Java 开发的 Aglet 架构移动 Agent 信息交互网络,充分发挥了 Java 语言的优越性,而且整个系统运用专门为移动 Agent 开发的 Aglet 语言系统来实现,在信息服务系统设计中是一个很大的创新。现阶段已经实现系统信息跨平台迁移交互和信息数据库功能等关键分程序,已经通过 Aglet 实验平台 Tahiti 的测试。接下来的工作主要是对系统的安全性能和数据库维护等部分的继续改进和研究,以及信息智能分析的研究,促使系统运行更安全,更符合用户个性化的需求,达到易于操作,实用性强的目的。

参考文献

- 1 Kotz D, Gray R S. Mobile Agents and the Future of the Internet[J]. ACM Operating Systems Review, 1999, 33(3): 7.
- 2 Lange D B, Oshima M. Programming and Deploying Java^(TM) Mobile Agents with Aglets^(TM)[M]. Addison-Wesley Pub. Co., 1998-09.
- 3 张云勇,刘锦德.移动 Agent 技术[M].北京:清华大学出版社,2003.
- 4 于杰,史小宏,杨斌.基于移动 Agent 和 Web 服务的分布式地理信息查找研究[J].计算机辅助工程,2005,14(1): 56.
- 5 李泽锋.基于多 Agent 技术的 Internet 信息服务模式研究[J].情报杂志,2005,24(4): 69.

参考文献

- 1 Shi F. Generic Algorithms for Feature Selection in an Intrusion Detection Application[D]. Mississippi State: Mississippi State University, 2000.
- 2 Luo J. Mining Fuzzy Association Rules and Fuzzy Frequency Episodes for Intrusion Detection[J]. International Journal of Intelligent System, 2000, 15(8).
- 3 Wierzhon S T. Discriminative Power of the Receptors Activated by K-contiguous Bits Rule[J]. Journal of Computer Science and Technology, 2000, 1(3).
- 4 齐建东.数据挖掘技术在入侵检测中的应用[J].计算机工程与应用,2004,40(6).