

基于移动 agent 的 图书馆信息检索系统设计

张建芬

【摘要】 本文主要以 Grasshopper 作为移动代理平台, 设计了一个基于移动代理的分布式检索系统。该系统将传统的 Web 服务器与移动代理系统集成, 由移动代理系统来完成对各个数据库的检索。

【关键词】 移动代理 信息检索 Grasshopper

Abstract: This paper designs a distributed retrieving system that bases on Grasshopper, a Mobile Agent Platform. This system integrates the traditional Web server with the Mobile Agent system, retrieving is carried out by the mobile agents.

Key words: mobile agent information retrieval Grasshopper

1 引言

图书馆拥有丰富的文献资源, 随着图书馆自动化系统的发展及进一步完善, 绝大多数图书馆 (尤其是高校图书馆) 都实现了自动化, 印刷型的书刊都建立了书目数据库供读者查询。同时随着电子资源的发展, 各图书馆增加了很多电子图书及各种网上数据库。如果读者想全面查找有关课题的资料, 就要打开各种不同的检索界面, 重复输入检索条件。图书馆最好能把这些资源整合在一起, 对读者提供统一的信息检索服务。读者只需在一个检索界面下, 就可以检索到自己所需要的印刷型文献和各种电子资源。然而图书馆这些数据库的服务是逐步开通的, 图书馆的网络是一步步建设起来的, 可能会有网络环境异构的情况。这种整合最重要的一个方面就是整合后的系统能够进行分布式的操作。

在分布式信息检索领域中, 移动代理技术有着特殊的吸引力。通过将移动代理发送到信息源处, 应用程序能够避免传输大量的中间数据; 能够在客户机与网络的连接断开时继续检索任务的执行; 能够在网络中对从各个信息源处检索到的检索结果进行归并和过滤, 而不是将所有的结果都转到客户机上。利用移动代理技术能够很好地满足这些需求, 它能够为网络移动计算提供简单的、适应性的、统一的解决方案。

我们利用移动代理的移动性和跨平台性, 设计了一个基于移动 agent 的分布式异构数据库的检索系统。此系统将传统的 Web 服务器与移动代理系统集成起来, Web 服务器接收用户的检索请求, 为每一个用户检索请求创建一个移动代理, 移动代理去分布在各处的数据库服务器上检索用户需要的数据, 最后将检索的最终结果向用户返回、显示。本系统对客户端没有额外的软硬件要求, 只需要 Web 服务器与移动代理系统交互, 在各个数据库服务器上安装一个移动代理系统, 为代理的执行提供环境。

2 系统开发平台——Grasshopper

整个信息检索系统都是采用 Java 语言在 Grasshopper 这个移动 Agent 开发平台上构建和运行的。Grasshopper^{[1][2]} 是全世界第一个完全实现 OMG MASIF 标准的 MA 平台, 它用 Java 开发, 基于 CORBA 规范, 通过专用的 Grasshopper ORB 互联。理论上 Grasshopper 实现了一个分布式 Agent 环境 (DAE), DAE 由域 (Region)、位置 (Place)、Agent 系统 (agency) 和各类 Agent 构成, Agency 提供了 Agent 能够创建、解释、执行、传送和终止的 MA 平台。Place 提供了 Agency 内具体的执行环境。Place 在 Agency 内对功能和服务进行分组。Region 由多个 Agency 组成, Region 允许多个 Agent 系统代表同一个人或者组织。域提供了与其他域的客户进行通信的抽象层, 通过发布域地址作为访问本域内其他主体的入口。

Grasshopper 的通信基础设计非常巧妙, 可以通过 CORBA 进行通信, 也可以通过 Java RMI 和 Socket (套接字) 连接进行通信, 整个通信结构采用插件技术, 具有很好的扩展性。Grasshopper 不仅实现了 Agent 之间异步和同步通信, 而且实现了多播 (multicast) 通信方式, 使得一个客户能够同时与多个服务 Agent 进行交互。

3 系统的体系结构

该信息检索系统体系结构主要分为 4 层: 用户界面, MA 对象, 分布式 Agent 支持环境以及 Java 虚拟机 (JVM)。最高层是用户界面, 通过用户接口 (GUI) 来进行 MA 对象的创建与管理。MA 对象是基于用户请求而创建的, 该系统是面向信息检索这个特定应用的, 具有自主移动特性, 它在 DAE 中执行来完成在指定地点进行信息检索的任务。而分布式 Agent 支持环境, 它是整个系统体系结构的核心, 并向上为 MA 应用程序提供编程接口, 它实现 MA 对象的移动、生命周期、命名等重要服务。最底层是 JVM, 它为 MA 对象及其支持环境的运行提供环境, 并通过 JDK API 的形式为程序员提供基本编程接口。这 4 个部分及其相互关系构成了该系统的体系结构^{[3][4]}, 如图 1 所示:



图 1 系统体系结构

一个 MA 系统最重要的是 MA 和 MA 的运行环境两部分。在该系统中 MA 的运行环境就是 DAE, 也就是上述的开发平台——Grasshopper。MA 通过 DAE 实现其在网络上的移动和相应信息检索动作; 而 DAE 为 MA 的移动和执行提供必要的执行环境以及相应的发送、接收、恢复、安全管理和服务调用等功能或服务。

4 基于移动 agent 的图书馆分布式信息检索系统的设计^{[5][6][7][8]}

4.1 用户、Web 服务器应用程序、检索代理之间的交互

在本系统中, 用户通过 HTTP 协议与 Web 服务器交互: 在检索服务器上, 本系统将 Web 服务器与移动代理系统集成起来, Web 服务器应用程序与移动代理系统中的检索代理都可被实现为通信的服务器端或通信的客户端。

Web 应用程序将检索的条件和目标分别组织到两个 Vector 类型的变量中, 在检索条件 Vector 中是一个个的检索条件值——对应, 在检索目标 Vector 中是一个个以字符串表示的检索目标。然后创建一个检索代理, 通过调用检索代理的接口类中提供的方法将检索的条件和目标传送给检索代理。三者之间的交互如图 2 所示:

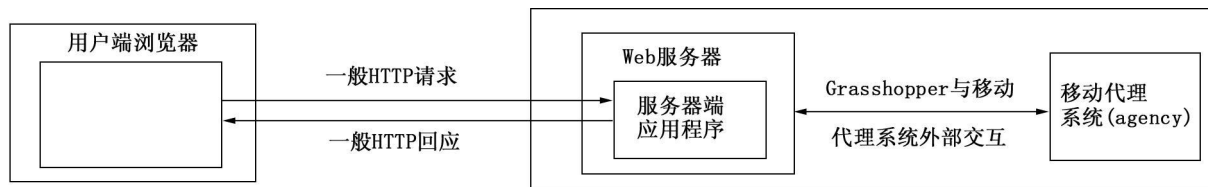


图 2 客户端、Web 服务器与移动代理系统三者之间的交互

图 2 中, 客户端与 Web 服务器之间用 HTTP 协议传输检索的请求和结果, Web 服务器与移动代理系统之间的交互通过 Grasshopper 使用的传输协议——GIOP 或者 IIOP 进行。

在 Grasshopper 中, 代理 A 与代理 B 的通信时, 代理 A 并不是与代理 B 直接通信, 而是引入一个中间实体——Proxy。代理 A 为了与代理 B 通信, 要创建 B 的一个 Proxy, 这个 Proxy 建立到真正的代理 B 的连接。在 Grasshopper 中有三种类型的 proxy: 代理, 代理系统, 代理系统域服务。

为了使 Web 服务器与移动代理系统合作, Web 服务器端的应用程序必须能够与移动代理系统中的代理通信, 也就是 Web 服务器与移动代理系统的集成起来。本人采取了与移动代理系统内部代理通信相类似的方法, 在 Web 服务器上为服务器端应用程序创建一个通信接收器。由于服务器端应用程序除了与代理交互外, 还有其他一些操

作, 所以本人为它加了一个服务器对象 ServerObject, 这个对象相当于服务器端应用程序对移动代理系统的一个通信接收接口, 接收代理传来的检索结果。服务器端应用程序通过检查服务器对象的状态来判断结果是否已到达。这样, 就将 Web 服务器与移动代理系统集成起来。

4.2 检索移动代理的实现

检索移动代理是实现本系统的核心模块, 它承担了所有的信息检索工作。它负责将要进行的检索的条件带给各数据库服务器上的本地服务代理。它在检索服务器和各个数据服务器之间移动, 收集用户需要的信息。在数据服务器上与本地服务代理交互, 得到检索结果并进行归并和过滤, 最后返回检索服务器。在整个过程中, 检索移动代理是完全自治的, 根据所带的地址列表四处移动, 根据自己的功能流程执行任务。检索移动代理由五个模块组成:

4.2.1 代理标识模块

系统中的每个代理都有一个身份标识。身份标识主要用于代理身份的验证, 是代理系统安全性的要求。身份标识也用于代理的管理, 代理系统根据代理的身份标识查找相应的代理。在代理生成时代理系统给它一个唯一的身份标识。代理每移动到一个地方, 当地的代理系统根据它的身份标识进行身份认证和授权, 决定代理在该处对本地资源的访问权限。

4.2.2 对外接口模块

检索移动代理的对外接口负责与检索代理和本地服务代理的交互。

4.2.3 业务处理模块

业务处理模块中定义移动代理在每个位置上进行的操作。因为各种检索移动代理中必然有一部分操作对所有的检索是相同的 (如数据库检索结束后结果的格式转换等), 将这一部分放在各个数据源处的本地服务代理中可以减小检索移动代理的体积。如果检索移动代理是第一次被实例化的, 它首先执行以下的操作: 从检索代理处得到检索的条件和目标, 对变量进行初始化; 从地址列表中取一项; 抽取下一个移动目的地和检索目标, 放到特定的变量中; 而在每一个数据库服务器处, 检索移动代理的主要操作流程如图 3 所示:

4.2.4 地址列表 (LocationList) 模块

地址列表中存放检索移动代理需要移动到的位置以及在相应位置上应该检索的内容。检索移动代理根据自己的地址列表来进行迁移: 每到达一处, 在检索结束后将该处的地址信息从列表中删除, 当地址列表为空时, 检索结束, 移动代理返回到检索代理服务器。

4.2.5 结果集 (ResultSet) 模块

结果集中存放检索的结果。检索结果在每个数据源处被转化为统一的格式。

JAVA 是 Grasshopper 上的编程语言。由于 Java 不能捕获线程的运行状态, Grasshopper 不支持进程的强迁移。当代理迁移时, 实际是代理对象被序列化后传到目的代理系统, 所以代理的数据状态中所包含的变量必须是可序列化的, 为了让移动代理在迁移过程中能够保存需要的变量的值, 这些值必须包含在数据状态中, 不在数据状态中的变量被设为“暂时的 (transient)”类型, 这种变量在代理每一次被实例化时都重新赋值。考虑到代理传输的效率, 代理的数据状态中所包含的数据应当尽量少的。本系统中将需要保持连续状态的变量 (如地址列表、结果集、存放下一个目的地址和检索目标的变量等) 定义为非“暂时的”、可序列化的。为了判断代理的当前状态 (刚刚创建, 在数据库服务器处, 还是检索完毕返回了业务代理系统), 实现中设置一个非“暂时的”变量 state, 根据 state 的值判断代理的处境, 选择应该执行的动作。

为了方便地接收检索代理传送来的检索条件和地址列表, 检索移动代理提供一个可以由检索代理调用的方法。检索移动代理的接口类的实现如下所示:

```
public interface IretriMbbileAgent {
//从外界可以访问到的检索移动代理提供的服务
//检索代理通过调用下面的方法将检索的条件和地址列表传递给
//检索移动代理, 检索移动代理接收到数据后初始化自己的变量
public void getPara (Vector condition, Vector addrServL)。}
```

系统中的每个数据库服务器端都有一个代理系统 (agency), 本地服务代理在这个代理系统中运行, 检索移动代理移动到这个代理系统之后, 与此处的本地服务代理交互。

在本系统中, 检索移动代理与本地服务代理运行在同一个代理服务器中, 它们之间的通信是位置透明的 (即检索移动代理只要提供本地服务代理的代理标识、接口类的名字和检索移动代理类代码的位置就能创建本地服务代理的 proxy, 当检索移动代理调用本地服务代理提供的方法时, 本地服务代理的 proxy 自动通过代理系统的域服务确定本地服务代理的位置)。与前面的 Web 应用程序与检索代理类似, 检索移动代理在这里做通信的客户端, 而本地服务代理做通信的服务器端, 本地服务代理的接口类体现本地服务代理对外提供的服务, 本地服务代理实现这个接口类。检索移动代理调用本地服务代理的接口类中提供的方法。

本系统的实现方法是检索移动代理首先创建一个本地服务代理的实例, 然后根据这个实例通过代理系统创建它的 proxy。当检索结束时, 检索移动代理返回到检索代理所在的移动代理系统, 创建检索代理的 proxy, 调用检索代理的方法, 将结果返回给它。

注释

- [1] 张云勇. 移动 agent 及其应用. 北京: 清华大学出版社, 2002: 9- 11, 16, 244- 245
- [2] Grossshopper Programmers Guide, IKV+ + GmbH <http://www.grossshopper.de>
- [3] 骆正虎, 杨敬安. 基于移动 Agent 的分布式计算模型研究. 小型微型计算机系统, 2002 (3): 300- 304
- [4] [8] 赵苏玮, 刘斌. 移动 agent 在信息检索中的应用与实现. 微处理机, 2009 (4): 82- 85
- [5] 胡春玲. 移动代理技术在分布式信息检索中的应用研究. 北京: 北京邮电大学硕士学位论文, 2001: 36- 45
- [6] 宋国云, 胡月宁, 房燕. 基于移动 Agent 的网络信息检索服务的研究. 科技资讯, 2008 (2): 110
- [7] 张小琴. 移动 agent 在数字图书馆资源检索中的迁移策略. 中南民族大学学报, 2009 (2): 79- 81

张建芬 南京邮电大学图书馆。

(上接第 62 页)

献采访理论知识, 并具有比较丰富采访经验, 但缺乏足够的专业背景知识。而读者尤其是在教学科研一线的教师都是各个学科的专家, 具有良好的学科背景知识, 同时了解相关专业设置及其发展方向, 熟悉教学大纲和专业教学计划, 其所采购的文献更适应教学科研需要, 但他们缺乏文献采访理论知识和采访经验, 而文献资源三级建设体系的建立恰好是整合了图书馆人员文献资源建设技能优势与读者的学科专业优势, 实现了他们之间的优势互补, 同时又尽可能避免了各自的缺陷, 从而使得构建一支结构合理的高素质文献资源建设队伍成为可能。

参考文献

- 1 杨晓秋. 高校图书馆区域文献信息资源保障体系建设研究: 以

黑龙江省高校图书馆为例. 图书馆学研究, 2008 (2)

- 2 储济明. 高校图书馆中文图书采购主体地位的缺失与对策. 图书馆建设, 2009 (1)
- 3 方敏. 高校内部文献资源整体化建设与管理研究工作研究. 图书馆学研究, 2008 (8)
- 4 严扬帆. 新形势下高校图书馆图书采访工作的思考. 图书馆, 2007 (3)
- 5 资芸, 钟叔玉, 董毅明. 高校图书馆图书采访决策模型研究. 情报杂志, 2007 (6)
- 6 王明惠. 基于网络环境的高校图书馆采访工作新思考. 现代情报, 2007 (7)
- 7 韩立栋. 基于书商的高校图书馆“图书两级采访”模式研究. 情报资料工作, 2007 (3)

储济明 徐州工程学院图书馆。