

基于 JXTA 的 P2P 文件共享系统的研究

林 川, 林 鑽

(温州大学计算机科学与工程学院, 浙江温州 325035)

摘 要: JXTA 是一个解决 P2P 计算的开放式网络计算平台, JXTA 提供的内容管理服务 CMS 为所有被共享的内容提供管理功能. 提出了基于 JXTA 的 P2P 文件共享系统的设计方法, 并给出具体的实现.

关键词: JXTA; CMS; 文件共享系统

中图分类号: TP393.02 **文献标识码:** A **文章编号:** 1006-0375(2007)04-0035-07

计算机对等网 (P2P) 技术是目前流行于国际计算机网络技术研究领域的一个热点. P2P 技术通过直接互连实现信息资源、处理器资源以及存储资源的全面共享, 网络中的每个节点是对等的, 这些对等节点具有共同的责任与能力并协同完成服务. 通过网络提供信息资源共享是 P2P 应用最多及最成功的方式之一. P2P 实现信息资源共享的主要目的就是全面实现数据共享, 使用者可以直接从连网的任意节点上检索信息和共享资源, 而不需要连接到服务器上再进行浏览与下载. Sun 公司的 JXTA 技术用于提供 P2P 系统所需的基础服务, 由一系列简单的、开放的协议组成, 通过这些协议, 连接到网络上的任何设备都能够相互通信、协作和资源共享. JXTA 隐藏了复杂的物理网络拓扑结构, 为对等节点构建了一个虚拟网络, 使任意对等点之间能够进行通信. 本文将着重阐述 JXTA 技术在 P2P 文件共享系统中的应用.

1 JXTA 简介

JXTA 是一个解决 P2P 计算的开放式网络计算平台. JXTA P2P 平台允许开发者在其上建立 P2P 的应用. JXTA 标准规范中指出, JXTA 由 6 个协议组成, 这些协议是专为特定的、分布式的对等网络计算设计的. 使用这些协议, 对等点 (peer) 可以互相合作建立自我组织、自我管理的对等组, 不必关心他们在网络中的位置, 并且不需要集中式的管理机构^[1]. 虽然 JXTA 的参考实现是用 Java 语言写的, 但事实上它对当今现有的任何语言、操作系统 (OS) 或平台都适用. JXTA 的体系结构由三层组成. 最底层是 JXTA 核心层 (JXTA Core), 这一层封装了最根本的东西, 包括对等点 (Peer)、对等组 (Peer Group)、Peer 发现、Peer 通信、Peer 监视和相关的安全原语; 服务层 (JXTA Services) 扩展了核心层的功能, 它包括对于 P2P 网络不是必需的、但很通用的功能, 如查找、共享、索引、代码缓存和内容缓存的机制; 应用层 (JXTA Applications) 提供基于 P2P 的应用程序, 比如电子邮件、数据存储. 安全服务贯串于始终, 在每一层都有所体现. JXTA 技术实际上是一组相互独立的协议, 每种协议定义了对等点之间如何交换信息, 交互并管理应用程序. JXTA 不要求每个对等点实现所有协议, 只需实现那些它实际需要使用的协议.

收稿日期: 2007-01-08

作者简介: 林川(1965-), 女, 浙江温州人, 副教授, 硕士, 研究方向: 计算机网络应用

2 基于 JXTA 的 P2P 文件共享系统设计

2.1 系统设计目标

文件下载是目前 P2P 的主要应用, 文件的提供者把文件存储在网络的节点上, 对它具有完全的控制权(创建、修改、删除等), 访问者具有只读权而无修改权力. 本文件下载系统主要实现(1) 对等组创建、搜索功能; (2) 对等点搜索功能; (3) 共享文件发布、查询、下载功能. 系统的体系结构如图 1 所示.

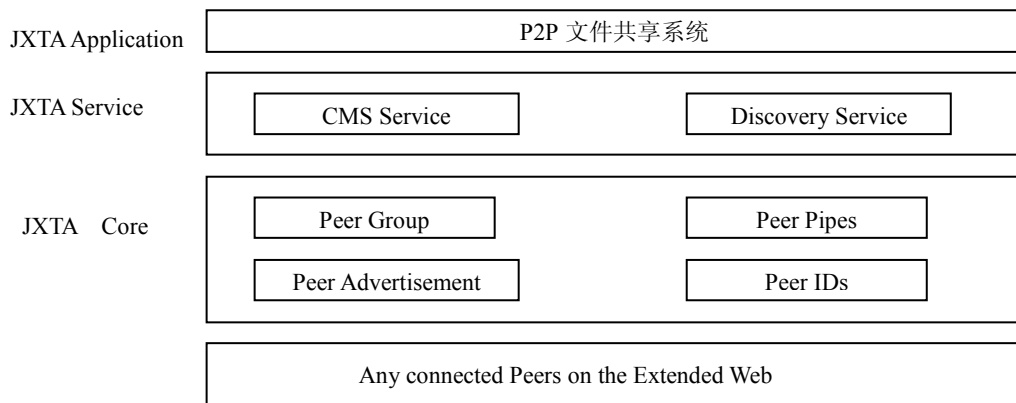


图 1 文件共享系统的体系结构

2.2 网络角色分类

将网络角色分为两大类:

(1) 普通对等点. 可以发送和接受消息, 并把发现的广告缓存在本地. 当接受了查询请求, 会在自己的缓存中查找相关的信息, 并把找到的信息返回给查询发起者, 但它没有传递查询功能.

(2) 集合对等点. 除了拥有普通对等点的功能外, 还能将收到的查询信息传递给同组中的集合点, 直到所设置的消息生存周期结束为止.

2.3 内容管理服务

JXTA 提供了内容管理服务 CMS (Content Manager Service), CMS 充当一个在对等点之间共享并交换内容的框架, 因而对等点不仅可以宿主它与其它对等点共享的内容, 而且可以定位和检索其它对等点中的内容. 任何需要被共享的数据称为内容 (Content). 每一份共享内容都有一个独一无二的内容 ID 和内容广告. 利用内容本身的二进制数据产生唯一的 128 位 MD5 校验和, 通过 MD5 校验和, 可以很容易分辨出两个不同的对等点所共享的两个文件是否相同. 每一份共享内容都有一个与之相关的内容广告 (Content Advertisement), 每个对等点通过查询目的对等点的内容广告来获取对方的共享信息. 内容广告提供与其共享内容相关的元信息 (Meta Information), 这些元信息包括有: 内容 ID、内容名、内容长度、Mime 类型、内容描述, 除了内容 ID 和内容名称之外的所有内容都是可选的内容广告以 XML 的格式存储. 当用户利用 CMS 发布本地共享的文件以后, CMS 管理着一个存储着本地共享文件的引用和与之相关联的内容广告的可持久性存储 (Persistent Store). 此存储存着对共享文件的引用而不是整个文件内容, 一旦共享内容被其它的对等点所存取, 内容 ID 会被重新验证, 以确保上次共享以来没有被改变^[2]. CMS 服务使用 JXTA 管道来请求和获取远程内容, 并不指定搜索内容的对等点路由路径, 只提供内容的本地搜索^[3].

2.4 利用 CMS 实现文件共享的一般方法

2.4.1 共享信息

CMS 通过 JXTA 中的类 `net.jxta.share.CMS` 实现内容管理服务. CMS 的 `startApp` 方法接受用户欲共享的目录名作为输入参数, 该目录代表服务存储有关共享内容的持久信息. 如果用户没有指定共享目录参数, CMS 则在当前 JXTA 应用程序的目录下创建默认的名为 `cms` 的目录来存储信息^[2].

2.4.2 搜索内容

对等点搜索网络上与之相邻的对等点, 发出 `LIST_REQ` 请求消息, 其他对等点在接收到此消息后, 查询本地是否有符合搜索要求的内容. 如果有, 则沿请求消息的发送路径发送 `LIST_RES` 响应消息, 返回包含查询到的所有内容的广告. 无论本地是否存在符合查询请求的文件内容, 其他对等点都会将该查询请求消息通过扩散方式继续在网络中传递, 直至 `TTL` 字段值减为 0 时停止继续转发.

2.4.3 下载信息

当源对等点收到目的对等点发送给它的 `List_RES` 消息后, 通过先前发出的 `List_REQ` 消息中指定的接收请求输入管道发送一个 `Get_REQ` 消息至目的对等点, 要求下载内容数据, 其中包括了用来接收内容响应消息的输入管道广告对等点, 当目的对等点收到 `Get_REQ` 消息后, 首先根据其中的 `CID` (请求内容的 ID) 来检查共享内容是否符合要求. 如果符合则根据 `Get_REQ` 消息中指定的响应输入管道建立一个相应的输出管道, 并发送一个或多个 `Get_RES` 消息予以回应, 其中包括了内容数据.

2.4.4 发现服务

JXTA 使用管道技术, 在建立连接和传输数据时无须考虑协议的问题. 在整个发现服务 (Discovery Service) 中使用消息形式进行信息的传递, 分为查询消息和响应消息. CMS 搜索过程见图 2, 首先普通对等点 1 向所有已知的普通对等点和集合对等点发送查询, 然后集合对等点响应消息, 并将消息传给与它连接的普通对等点, 最后各个普通对等点直接将响应消息发送个原始查询点, 即普通对等点 1.

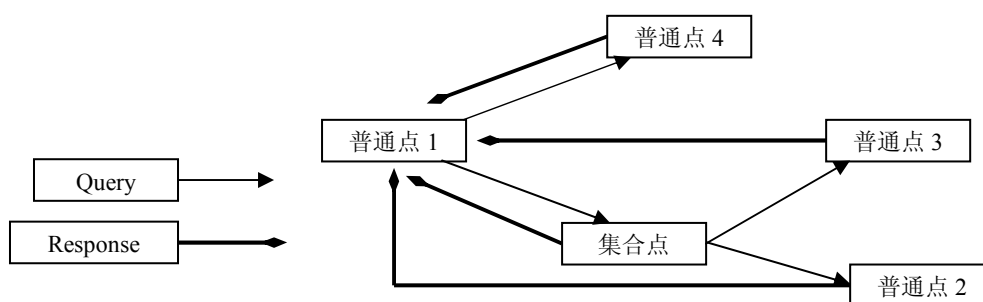


图 2 CMS 发现过程

3 系统的具体实现

系统启动后将在当前目录下生成 JXTA 目录以用于保存各类证书及临时文件. CM 目录建立在 JXTA 目录下, 主要用于缓存在对等点发现过程中传送的通告, 这些通告以通告 ID 作为文件名进行存储. CM 目录按照对等组进行组织, 每个对等组都对应一个子目录, 在组目录中保存组中发现的通告信息和其他关于组的信息, 如成员资格证书等.

3.1 系统启动的实现

系统启动完成几件基本任务,如输入账号和口令、网络环境的配置、加入对等组和启动 CMS 服务等.该实现主要涉及两个类, Jxta 类和 PeerDiscoListener 类, 分别见表 1 和表 2.

表 1 Jxta 类函数及描述

函 数	描 述
joinHaiNet()	加入对等组 HaiNet
joinPublish()	发布自己点的信息
startCMS()	启动 CMS 服务, 并发布共享文件
getPeers()	启动对等点的发现

表 2 PeerDiscoListener 类函数及描述

函 数	描 述
DiscoveryEvent (DiscoveryEvent e)	将发现的对等点显示在窗口上

PeerDiscoListener 类实现了接口 DiscoveryListener. 在 Java 参考实现中, 可以给 Discovery Service 注册一个 listener 对象, 当 Discovery Response Messages 到达时可以调用该 listener 的 discoveryEvent 方法获得通知, 并传递事件本身.

3.2 资源共享的实现

Cms 类: 负责文件资源. 见表 3.

表 3 Cms 类函数及描述

函 数	描 述
Share (File file, boolean isCopy)	共享文件或文件夹, 并确定是否将其复制到目录下
Unshared (File file, boolean remove)	取消文件或文件夹的共享, 并确定是否删除不共享的文件

共享文件夹的代码如下:

```
public void share( File file,  boolean isCopy) {
    if(file.isDirectory())    {
        File[] fileList=file.listFiles();
        for(int i=0;i<fileList.length;i++) {
            share(fileList[i],  isCopy);
        }
        return;
    }
    File to  = new File( Constants.DOWNLOAD,  file.getName());
    FileInputStream fis = null;
    FileOutputStream fos = null;

    if( isCopy) {
        try {
            to.createNewFile();
            fis = new FileInputStream(file);
```

```
        fos = new FileOutputStream(to);

        byte[] buf = new byte[BLOCK];
        int l = 0;
        while ((l = fis.read(buf, 0, BLOCK)) > -1) {
            fos.write(buf, 0, l);
        }
    } catch (FileNotFoundException fnfe) {
    } catch (IOException ioe) {}
    finally {
        if (fis != null) {
            try {
                fis.close();
            } catch (IOException ioe) {}
        }
        if (fos != null) {
            try {
                fos.close();
            } catch (IOException ioe) {}
        }
    }
}
```

```
Content content = null;
```

```
if (cache == null) {
    cache = new HashMap();
}
```

```
try {
    content = contentManager.share(to, "nihao");
    Hai.setInfo( content.getContentAdvertisement().getName(), Constants.PEER_INFO);
    cache.put( to.getName(), content);
} catch (IOException ioe) {}
} //end share
```

3.3 资源搜索的实现

ContentRequest 类: 当用户启动搜索时, 该类负责资源的搜索. 该类实现了 **DiscoveryListener** 接口. 见表 4.

表 4 ContentRequest 类函数及描述

函 数	描 述
activateRequest()	启动搜索
notifyMoreResults()	当有返回信息时调用该函数, 对搜索结果进行处理
cancel()	删除监听
discoveryEvent (DiscoveryEvent e)	对返回信息进行处理

启动搜索的代码如下:

```
public void activateRequest() {
    disco.getRemoteAdvertisements (null,
        DiscoveryService.ADV,
        //NAME_ATTRIBUTE,
        //subString,
        null,
        null,
        DEFAULT_THRESHOLD,
        this);
}
```

对返回信息处理的代码如下:

```
public void discoveryEvent (DiscoveryEvent e) {
    ...
    ...
    while ( enum.hasMoreElements()) {
        str = (String)enum.nextElement();
        if( str == null) continue;

        ByteArrayInputStream stream =
            new ByteArrayInputStream( str.getBytes());

        try {
            adv = AdvertisementFactory.newAdvertisement( Constants.XMLMIMETYPE, stream);
        } catch (IOException e2) {
            // TODO 自动生成 catch 块
            e2.printStackTrace();
        }
        if(adv instanceof ContentAdvertisement) {
            ContentAdvertisement Cadv = (ContentAdvertisement) adv;
            if( Cadv.getName().toUpperCase().contains( subString)) {
                addResult ( Cadv);
                gotSomething = true;}
            }
    }
}
```

```

    }
    if (gotSomething) {
        notifyMoreResults();    }
    }
}

```

3.4 资源下载的实现

GetRequest 类: 该类继承 **GetContentRequest**, 只要提供对等点组, 内容通告和文件存放位置即可完成文件下载功能.

3.5 用户界面的实现

用户界面主要有以下几个类组成. 见表 5.

表 5 用户界面类的名称及描述

类 名	描 述
Info.java	主要显示与系统有关的信息, 如发现对等点的信息, 提示共享信息等.
Search.java	用户在该界面输入搜索字符串, 并显示搜索的结果.
Share.java	在该界面用户可以增加资源共享, 取消资源共享等操作.
Down.java	显示用户下载的信息, 如完成情况等.

4 结束语

本文给出了利用 JXTA 技术实现的文件共享系统. JXTA 技术是 P2P 的新发展, 它提供的通用的平台和协议可以高效灵活地开发 P2P 应用程序, 随着 JXTA 技术的逐步成熟, 软件开发人员将会开发出更多基于 JXTA 的 P2P 网络应用软件.

参考文献

- [1] Oaks S, Rraversat B, Gong L (技桥译). JXTA 技术手册[M]. 北京: 清华大学出版社, 2004. 9-15.
- [2] 许斌. JXTA-Java P2P 网络编程技术[M]. 北京: 清华大学出版社, 2003. 154-169.
- [3] Collab Net. The JXTA Content Manager Service [DB/OL]. <http://cms.jxta.org/cmswhitepaper.html>, 2005-01-30.

P2P File Sharing System Design Based on JXTA

LIN Chuan, LIN Huang

(School of Computer Science and Engineering, Wenzhou University, Wenzhou, China 325035)

Abstract: JXTA is an opening platform of P2P network calculation, by which CMS is defined as any sharable content. Content manager service can be implemented with CMS by JXTA class. This paper discusses the method to realize P2P file sharing system, and offers the specific actualization.

Key words: JXTA; CMS; File sharing system

(编辑: 王一芳)