远程检索在网络科技资源中的研究与应用①

Research and Application of the Remote Search in the Technological Resources of the Network

焦翠花 周明全 张 岩 (北京师范大学 信息科学与技术学院 北京 100875)

摘 要: 针对网络科技资源的分散和不易检索的特点，提出了一种使用远程检索来实现网络之间资源的共享和检索方法。详细介绍了远程检索模块的功能，远程检索涉及到的关键技术、远程检索模块的体系结构以及具体实现远程检索的方法和远程检索的意义等。

关键字: 网络科技资源 远程检索 非阻塞 I/O XML 线程池

远程检索在网络科技资源中研究与应用包括非阻塞 I/O 技术， XML 和线程池等技术，并给出了检索模块的体系结构和实现方法。

1 引言

1.1 文章安排

本文第 2 节介绍网络科技资源的内容。第 3 节就相关的关键技术进行了说明。第 4 节详细介绍了该检索模块的体系结构。第 5 节给出了具体的实现方法。第 6 节给出了结论以及未来工作。

1.1.1 基本介绍

随着网络信息化建设工作的推进，现在存在着大量分散的网络科技资源数据库和应用系统，各数据库和应用系统彼此孤立，相互之间难以实现资源共享和信息传递，致使各个应用系统只能是孤岛式的运行，无法实现资源间、系统间的资源共享和信息联动。另外，科学研究人员可能需要对不同学科、不同资源库的网络科技资源进行综合统计分析和有效评价，获得资源的定位，或者提供综合的数据子集。因此，人们迫切需要实现资源在不同平台间的共享和访问。网络科技资源应用集成环境建设项目中的远程检索模块就是为了实现这一目的而设计的。

2 网络科技资源内容

在网络科技资源应用集成环境建设项目中，科技资源主要指科技信息资源，包括通过科技活动或其他方式获取到的反映客观世界的本质、特征、变化规律等的原始基本数据、根据不同科技活动需要进行系统加工整理后得到的各类数据集、以及其他用于支撑科技活动的数据集。网络科技资源应用集成环境建设项目可汇集的科技资源主要包括平台资源，领域资源，行业资源，科技业务资源，网络中分散的科技资源等方面的数据内容。

可汇集的数据源的数据形式至少应覆盖以下几类:

(1) 文本：即文字描述，这是描述资源数据的最基本、最常见的方式。

(2) 音频：描述语音、音乐数据。应支持常见的音频格式，如.wav、.mp3 等。

(3) 图像：此处指静态图像，应支持常见的图像格式，如.gif、.jpg、.bmp 等。

(4) 视频：即动态图像，应支持常见的图像格式，如.avi、.mpeg、.rm、.rmvb、.asf 等。

3 关键技术

网络科技资源中远程检索[1]功能主要涉及到非阻塞 I/O、XML 和线程池等技术。

3.1 非阻塞 I/O

非阻塞 I/O 的通信方式[2,3]是当一个方法需要处理 I/O 有关的事务时，不要求方法等待 I/O 操作完成

① 基金项目:国家科技基础条件平台建设项目(2005DKA63904)
收稿时间:2009-03-04

即可返回。从而减少了管理 I/O 连接导致的系统开销，大幅度提高了系统性能。通过非阻塞通信方式，我们可以发送和读取任何能够发送读取的信息。

非阻塞 I/O 设计[4,5]的原理是反应器设计模式(Reactor pattern)。分布式系统中的服务器端应用程序必须处理多个向它们发送服务请求的客户端。反应器模式正好适用于这一功能,它允许事件驱动应用程序将服务请求多路分用并进行分派,然后,这些服务请求被并发地从一个或多个客户端传送到应用程序。

远程检索中非阻塞 I/O 主要解决不同主机间通信问题。

3.2 什么是 XML

XML(Extensible Markup Language，可扩展标记语言)是 由 W3C(World Wide Web Consortium)组织于 1998 年 2 月制定的一种通用语言规范。XML 具有如下几个优点[6,7]:

(1) 纯文本;

(2) 开放性;

(3) 可扩展性;

(4) 自描述性;

(5) 很强的链接能力。

远程检索中通过 XML 语言来描述主机间通信的数据包格式。

3.3 线程池技术

线程池技术的核心是多线程技术[8,9]，一个进程内就可以同时容纳多个线程的运行，而多个线程同时共用了进程的地址空间和绝大部分的数据、代码。架构在这样的结构上，再加上相关的线程的创建、撤销、线程间的同步和异步等管理机制，就构成了一个具有基本管理能力的线程的集合，称为线程池技术。

线程池[9,10]为线程生命周期开销和资源不足问题提供了解决方案。通过对多个任务重用线程，线程创建的开销被分摊到了多个任务上。而且，通过适当地调整线程池中的线程数目，也就是当请求的数目超过某个阈值时，就强制其他任何新到的请求一直等待，直到获得一个线程来处理为止，从而可以防止资源不足。应用线程池，能够实现:

提供工作线程池中任务的调度，灵活地调整池的大小;

对线程生命周期进行管理，限制工作队列中任务的数目，以防止队列中的任务耗尽所有可用内存;

提供多种可用的关闭和饱和度策略。

远程检索中使用线程池技术来处理请求队列和响应队列中的包，以解决高并发的问题。

4 体系结构

4.1 体系结构图

远程检索模块的体系结构图如图 1 所示:

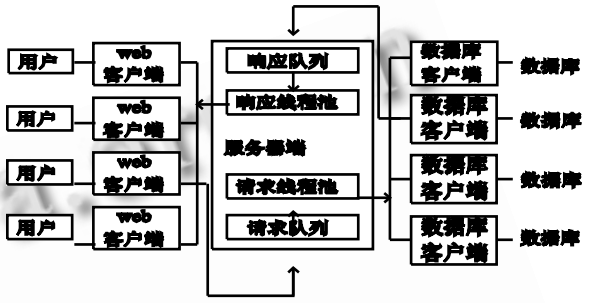

图 1　体系结构图

检索代理模块分为 Web 客户端检索代理模块、服务器端检索代理模块和节点服务器端检索代理模块等。

Web客户端检索代理模块的主要功能有提供用户交互接口、创建请求包并把它发送给服务器端、接收服务器端发来的响应包解析并展现给用户。

服务器端检索代理模块的主要功能有将 Web 服务器发送的响应包分包并发送到对应的数据库客户端、将数据库客户端发送的响应包汇总并转发给 Web 客户端、管理它与数据库客户端的连接，必要时可以删除与某个节点的连接。

节点服务器端检索代理模块的主要功能有根据服务器发来的查询包查询数据库并把查询结果封装成响应包反馈给服务器端。

其中，Web 客户端和服务器端部署在同一主机中，每个数据库客户端部署在不同的节点服务器上，不同的节点服务器连接着不同的网络科技资源平台数据库。

用户通过 Web 页面向检索系统提交查询请求,每发起一个请求就相当于启动一个客户端应用程序，web 客户端将用户的查询请求封装成请求包后向服务器端发包，服务器端经过处理后将查询包发送到对应的节点服务器上的客户端应用程序，这里称为数据库客户端。数据库客户端将查询包的内容封装成 SQL 语

句后查询数据库，再把查询结果封装成响应包回送给服务器端应用程序，服务器端汇总各节点服务器返回的结果集后把响应包发送回 Web 客户端，最后，Web 客户端解析响应包的内容并把其展现在 Web 页面上，这样就形成了一次完整的远程检索过程。

5 实现方法

5.1 包的封装

在远程检索中，不同主机间传送的数据全部封装成包，它的一般结构如下：

```
public class basePacket
{
    public String packetData;
    public SelectionKey selectKey;
}
```

String 类型的 packetData 代表包的内容，默认为空，用 XML 语言来表示描述包的格式，包的类型不同，包的 XML 格式也不同。

SelectionKey 类型的 selectKey 可以存储该连接的状态，每个请求对应一个 selectKey，通过记录 selectKey 可以区分不同的用户，以便将各自的检索结果正确的返回。

5.2 请求包

请求包是指将用户的查询参数封装成的包，其 packetData 的 XML 格式如下：

```
<?xml version="1.0" encoding="gb2312"?>
<packet>
<command>search_request</command>
<client>
    <clientInfo tableName="table1" keyName="key1"ip="192.168.0.1">
    </clientInfo>
    <clientInfo tableName="table2" keyName="key2"ip="192.168.0.1">
    </clientInfo>
    <clientInfo tableName="table3" keyName="key3"ip="192.168.0.2">
    </clientInfo>
</client>
<params>中国　科技　农业</params>
</packet>
```

其中，<packet>标签是整个 XML 的根节点，<command>标签表示该包属于哪种类型的包，search_request 表示该包为请求包。<client>标签表示待查询的数据库客户端信息列表，每个<clientInfo>标签代表一个数据库客户端节点，tableName 代表待查询的数据库表名称，keyName 代表待查询的数据库表中的关键字，ip 代表待查询数据库客户端的 ip 地址。<params>标签代表用户输入的查询关键字。

5.3 响应包

响应包是用户查询返回的结果，其 packetData 的 XML 格式如下：

```
<?xml version="1.0" encoding="gb2312"?>
<packet>
<command>search_response</command>
<DataList>
    <item>结果集 1</item>
    <item>结果集 2</item>
    <item>结果集 3</item>
</DataList>
</packet>
```

其中的<packet>、<command>代表的内容与请求包相同，只不过<command>中的 search_response 代表该包是响应包。<DataList>代表全部结果集的集合，每个<item>代表一条网络科技资源的具体内容。

5.4 包处理

5.4.1 请求队列

由于在某一时刻，可能有多个用户同时检索，所以需要用队列来保存用户的请求，当用户发起一个请求时，就将该请求包加入请求队列，服务器端通过线程池处理队列里的包，将包转发到数据库服务器后将队列里的包移除。

5.4.2 响应队列

既然可能同时存在多个请求，也就需要队列来保存响应包，当接收到数据库客户端发送来的响应包时就将它加入响应队列，服务器端通过线程池处理队列里的包，将包转发到 Web 服务器后将队列里的包移除。

5.4.3 分包

由于用户可能同时查询一个数据库客户端中的两个表，也可能同时查询多个数据库客户端中的表，这种情况下，服务器端要将请求包分包，根据用户的查询请求将查询包以节点服务器为单位分成不同的请求包并转发到相应的数据库客户端上。

5.4.4 包汇总

当数据库客户端返回检索结果时，由于用户可能同时检索多个节点，这种情况下，需要将数据库客户端返回的响应包进行汇总，这里要以用户的单个查询请求为单位，汇总完毕后通过服务器端发送给 Web 客户端进行后续处理。

5.5 数据库表描述

远程检索主要用到三个表：Resource 表、Directory 表和 system_log 表。

资源信息表(resource 表)主要描述了节点服务器中数据库的表(user_table)和该表内网络科技资源分类(metadata_id)的一个对应关系，其具体描述如下：

表 1　Resource 表描述

字段名	数据类型	说明
server_table	Varchar(100)	服务器生成的对应于用户提交表的表名
user_table	Varchar(100)	用户提交的表名
title	Varchar(100)	content 表中 title 字段对应于用户表中的哪个字段
description	Varchar(100)	content 表中 descrip tion 对应于用户表中的哪个字段
user_id	Varchar(20)	用户 id
username	Varchar(100)	提交用户的用户名
create_date	Varchar(100)	提交时间
node_ids	Varchar(5)	所属目录分类，外键，对应于 directory 表的主键
Audit	char(1)	审核是否通过
Metadata_id	Numeric(5)	元数据 id，外键，对应于 metadata 表主键
annotate	char(1)	是否注释，

目录融合表(Directory 表)主要描述了网络资源分类和国民经济分类的一个对应关系，即该资源是属于国民经济分类中的哪一个类别。其具体描述如下：

表 2　Directory 表

字段名	数据类型	字段说明
node_id	numeric(5)	目录标识符
node_name	varchar(50)	目录具体名称
economy_classify	varchar(20)	国民经济分类
society_classify	varchar(20)	社会经济目标分类
subject_classify	varchar(20)	学科分类
area_classify	varchar(20)	地域分类
resource_classify	varchar(20)	资源平台分类
parent_id	numeric(5)	父节点 id
metadata_id	numeric(5)	元数据 id，外键，对应于 metadata 表主键
rlst_rule	varchar(2000)	构成规则，如 SQL 语句等

节点服务器记录表(system_log 表)主要记录了节点服务器的 IP 地址、网络科技资源共享数据库类型等信息。

表 3　system_log 表描述

字段名	数据类型	说明
user_id	varchar(20)	用户 id
username	varchar(100)	用户名称
provide_time	varchar(100)	用户提交的时间
provider_ip	varchar(50)	提交用户的 ip
operate_type	varchar(50)	操作类型
database_type	varchar(50)	数据库类型
database_name	varchar(50)	数据库名称
metadata_id	numeric(5)	元数据 id，外键，对应于 metadata 表主键

用到的 SQL 语句如下所示：

```
select distinct resource.user_table from
resource inner join directory on resource.
metadata_id =directory.metadata_id where
directory.economy_classify= "classifyname"
```

此 SQL 语句的功能是查询用户所选分类所包含的所有网络科技资源数据库表的名称，其中，classifyname 是指某一分类体系下用户所选择的类别。

```
select system_log.provider_ip, system_log.
database_type from system_log inner join
```

resource on system_log.metadata_id =resource.metadata_id where resource.user_table= “ tablename”

此 SQL 语句可以查询出该表所在的节点服务器相关信息，例如节点服务器的 IP、节点服务器中数据库的类型等，其中，tablename 是指第一步中查询到的表名。

5.6 接口定义

远程检索功能主要用到两种接口：用户接口和数据库接口等。

①用户接口

用户接口主要通过 Web 方式展现，提供了如下功能：获取用户的检索条件包括查询关键字和用户感兴趣的类别(如农业，采矿业、制造业等)提交表单等。

应该注意的是，用户感兴趣的可能是多个分类体系或者一个分类体系下的多个子类。

②数据库接口

此接口提供了如下功能：根据数据库连接的协议实现模块与数据库服务器之间的通信。在用户进行信息检索的时候建立用户与数据库之间的联系。

6 结论

网络科技资源仍将以不同的平台或数据库的形式存在，并且通过远程检索可以很好的共享不同平台和数据库中的网络科技资源。因此，远程检索将继续发挥极大的作用。

参考文献

1 Bambara J, Allen PR, Ashnault M.刘垒,等译.J2EE 技术内幕.北京:机械工业出版社, 2002.

2 Beveridge J, Wiener R.侯捷,译.Win32 多线程程序设计.武汉:华中科技大学出版社, 2002.

3 AlurD, Malks D, Crupi J.牛志奇,译.J2EE 核心模式.北京:机械工业出版社, 2002.

4 程超,杨风召.基于Java非阻塞I/O开发高性能网络应用程序.电子工程师, 2006,10(32):71－73.

5 范宝德,马建生.Java 非阻塞通信研究.微计算机信息(管控一体化) ,2006,22(12-3):116－119.

6 李军怀,周明全,耿国华,等.XML 在异构数据集成中的应用研究.计算机应用, 2002,9(10):10－12.

7 王向安,张成洪.XML 的异构分布式数据库集成方案.计算机应用与软件, 2003,20(11):91－92.

8 赵海,李志蜀,韩学为,叶浩.线程池的优化设计.四川大学学报(自然科学版), 2005,42(2):63－67.

9 Andrew S.潘爱民译.计算机网络(第4版).北京:清华大学出版社, 2004.

10 封玮,周世平.Java中的线程池及实现.计算机系统应用, 2004,13(8):16－18.