

基于 XML-RPC 技术的分布式教育资源库系统的设计

肖君¹，张永忠¹，陈祝华²

¹上海远程教育集团，²北京国之源软件公司

摘要：针对目前教育资源分布分散，无法被有效地利用和共享等问题，本文提出了分布式教育资源库系统的设计思想，系统为多个分布式资源库节点建立统一的资源目录中心，采用 XML-RPC 技术解决资源目录中心和分布式资源库节点之间信息的同步问题，有效地实现了教育资源的共建共享。

关键字：分布式 XML-RPC 资源目录中心

1. 背景介绍

随着我国教育信息化的迅速发展，教育资源库的建设开始成为人们关注的焦点。目前，绝大部分地区的教育资源都是分布存储在各个学校或区域内的各个资源库中，如何有效地管理这些分散存储的资源并充分实现它们之间的共建共享，成为了教育资源库建设中的关键问题。

基于这种现状，目前教育资源的存储方式一般采用集中式和分布式存储。集中式存储是将所有物理资源和资源描述信息存储到一个位置，使用统一的管理系统进行集中管理。分布式存储是将资源分布存储在不同的资源节点上，以减轻统一管理的负担。分布式教育资源库有两种组织结构，一种是层次结构，一种是对等结构。层次结构是由不同级别的资源节点组织而成的，物理资源存放在下级资源中心，下级资源中心到上级资源中心注册并同步资源描述信息，用户可以在上级资源中心检索到该节点范围内所有下级资源中心的资源信息。对等结构是指各个资源节点不分级别，都是对等的、独立的系统。用户向某一节点提交查询请求时，该节点中如果没有相关资源，就把请求发送给所有与之相连的其他节点，其他节点收到请求后在自己资源库中进行查询，并把查询结果返回给发出请求的节点。

鉴于集中式存储管理难度大以及对等结构的分布式资源库易造成网络流量负载过重等局限，本文采用教育资源的分布式存储和层次结构管理，通过建立资源目录中心，对分布存储的资源节点进行集中统一管理，实现了分布式资源的快速检索和网内用户的单点登录验证等技术问题。

2. 分布式教育资源库系统结构设计

2.1 分布式教育资源库系统结构

本文提出的分布式教育资源库系统是基于资源的分布式存储和管理架构，采用 XML-RPC 技术解决资源目录中心和分布式资源库节点之间信息的同步问题。系统由资源目录中心、分布式资源库节点、目录中心代理和资源库节点代理等四个主要部分组成，如图 1 所示：

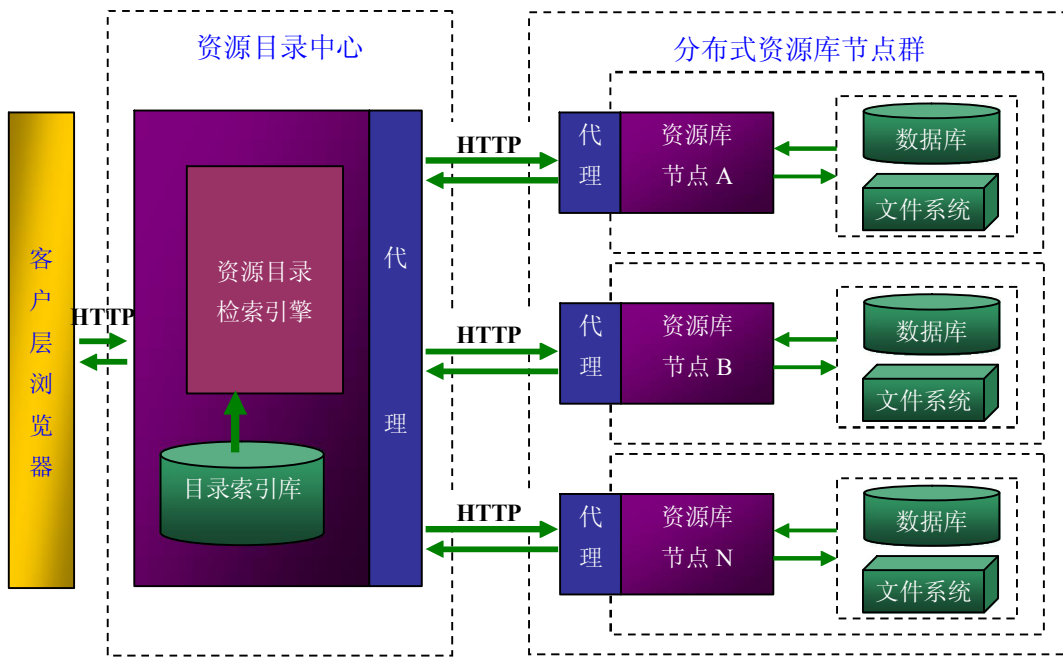


图 1 分布式资源库系统结构图

2.2 系统模块说明

2.2.1 资源目录中心

资源目录中心用于汇集和存放子资源库节点的资源目录信息、对目录信息进行索引，并提供统一的搜索接口。资源目录中心只存放各个资源节点的资源目录信息（即资源的基本描述信息），但是它并不会存储资源物理文件，而相应的资源物理文件则存储在不同的节点服务器上，同时资源目录中心对所有的资源信息建立必要的目录索引库，以利于通过检索引擎对存放在各个节点的资源行快速检索。

2.2.2 分布式资源库节点

分布式资源库节点本身是一个独立的资源库管理系统，具有自己的用户及权限管理系统，是物理资源文件的存放点，多个分布式资源库节点构成分布式资源库节点群。当节点有新的资源入库或资源的更新，它将通过代理来向中心同步新增的或改变的资源信息。资源库节点的架构和实现技术与中心无关，它只要捆绑实现统一的代理程序接口并且资源描述能符合(或转变)为中心统一一描述格式即可。资源在节点的存储形式、查看和播放方式、权限控制等不受中心干预。

2.2.3 目录中心代理

目录中心代理负责接收资源库节点资源更新消息、处理接收的同步数据、向节点发送处理反馈、检测节点代理的运行状态、接收节点的状态探测消息等。

目录中心代理结构见图 2:



图 2 目录中心代理结构

2.2.3 资源库节点代理

资源库节点代理负责侦测节点资源添加/修改/删除变化、将更新资源打包、向目录中心发送资源更新消息、接收中心的处理反馈和状态探测消息、提供本地用户的远程身份验证等。

节点代理结构见图 3:

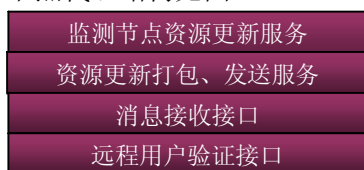


图 3 节点代理结构

2.3 资源库节点间的用户互信访问机制

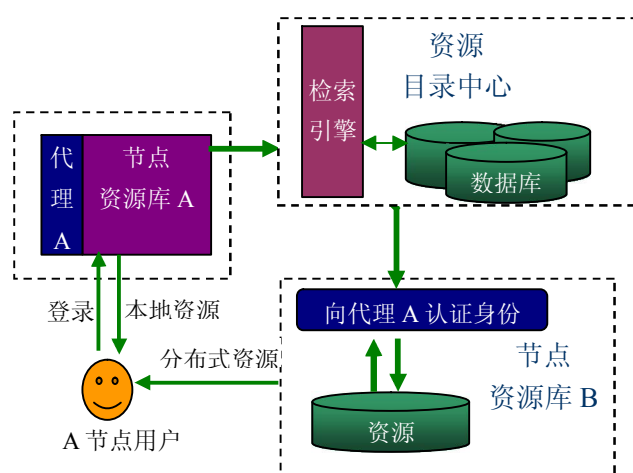


图 4 资源库节点资源访问原理图

分布式用户和权限管理。在整个应用系统信任网内，所有的用户都可以检索中心资源目录。但是对于资源多媒体内容的取得，则根据各个子节点自己的定义和设定决定。

资源库节点资源访问原理见图 4，当 A 节点系统的用户登录 A 资源库节点系统后，若要检索分布式资源，将调用中心的统一检索界面，由远程目录中心提供服务，当检索到第三方 B 节点的资源并且用户需要访问（“拿”）多媒体资源时，中心目录把控制转交给资源所属的 B 节点系统，B 节点接到控制权后，立即向请求用户所在的 A 节点代理进行远程验证。若验证通过，B 节点向用户客户端返回资源，若失败，则显示非法用户，拒绝请求。

3. 系统的主要功能及实现技术

分布式教育资源库系统充分利用了 XML-RPC (Remote Procedure Calling, 远程过程调用) 的特性。XML-RPC 是一套允许运行在不同操作系统、不同环境的程序实现基于 Internet 过程调用的规范和一系列的实现。这种远程过程调用使用 HTTP 作为传输协议，XML 作为传递信息的编码格式。XML-RPC 的定义尽可能地保持了简单，但同时能够传送、处理、返回复杂的数据结构。系统的实现采用了 Apache xml-rpc 的开发包，并应用 XML-RPC 技术封装了远程过程调用的控制和传输的细节。本系统重点应用了以下几点技术：

3.1 异步方式的资源目录同步

资源节点加入分布式资源网必须先为中心进行注册登记。取得中心确认后，中心会把该节点信息广播到各分节点的信任节点列表。节点的代理可以参照代理进行其他方式的实现。但是必须保证中心和其他节点调用的 XML-RPC 接口统一。

中心和节点代理都实现以下 Agent 接口：

```
public Interface Agent
{
    public boolean PingService(); //状态监测
    public boolean ReceiveMsg(String MsgID, String MsgBody, String MsgType,
        String NodeID); //消息接收
}
```

此外，节点还必须实现以下远程用户验证接口：

```
public Interface UserValidate
{
    public boolean UserLoggedIn(String UserName, String SessionID); //用户身份验证
}
```

子节点代理与中心代理之间的通信采用异步通信方式，即当发信方代理将消息（如注册消息，同步消息）传递给接收方后，接收方立即作出收取回应，但不保证接收方代理可以立即处理发信方代理的消息，也不保证发信方代理可以立即收到接收方代理的处理结果。

采用异步通信能够保证通信达到较高的可靠性和可测性。例如，假定某代理暂时与中心代理失去连接，但这些消息都会在节点或中心得以保存，系统可以保证一旦离线的代理与中心建立连接，它的所有消息仍将准确的传递给对方。

中心和节点资源库代理都启动三个服务程序：资源打包服务程序，消息发送服务程序，消息处理服务程序。当子节点有资源添加或者更新时，代理首先将更新的资源缓存起来。节点代理启动的监听服务程序在节点处理器空闲的时候，将缓存资源包装成统一标准的 XML 资源包，并且添加一条待发送的资源同步消息到消息队列。节点代理消息发送服务程序一旦监测到队列有未发送完成的 XML 资源包，在节点处理器空闲的时候通过调用中心 RPC 接口通知中心。

中心收到消息后，会把消息加入消息等待队列，待处理器空闲中心代理的消息处理服务通过资源库节点代理的 XML-RPC 接口获取 XML 资源包，并处理资源包入库。同时调用资源库节点的 XML-RPC 接口反馈处理结果。

3.2 节点资源库用户代理验证

代理之间的身份验证采用同步通信机制，从而保证了验证的实时性。

1. 用户登录所在节点资源库 A (IP:202. 121. 89. 101) 使用内建的 http server 启动代理 XML-RPC 代理服务：

```
XmlRpc.setDriver("org.apache.xerces.parsers.SAXParser");
WebServer server = new WebServer(8585);
server.acceptClient("202. 121. 88. 111"); //设置可信资源库分节点 1IP
server.acceptClient("202. 11. 87. 101"); //设置可信资源库分节点 2IP
...
server.addHandler("NodeAgent", new AgentHandler());
```

```
server.start();
```

```
...
```

2. 用户访问节点 B (IP: 202.121.88.111) 的资源, 资源节点 B 通过 XML-RPC 执行的用户身份验证:

```
...
```

```
try
```

```
{
```

```
    XmlRpc.setDriver("org.apache.xerces.parsers.SAXParser");
```

```
    XmlRpcClient client = new XmlRpcClient("http://202.121.89.101:8585");
```

```
    Vector params = new Vector(); //初始化调用参数
```

```
    params.addElement(strSessionID);
```

```
    params.addElement(strUserID);
```

```
    Boolean logined=(boolean) client.execute("NodeAgent.UserLoggedIn()",  
        params);
```

```
    if (logined)
```

```
    {...//允许访问资源}
```

```
    else
```

```
    {...//拒绝访问}
```

```
}
```

```
catch(ClassNotFoundException e)
```

```
{...}
```

```
catch(MalformedURLException e)
```

```
{...}
```

```
catch(XmlRpcException e)
```

```
{...}
```

```
catch(IOException e)
```

```
{...}
```

```
finally
```

```
{...}
```

3.3 集中式与分布式的协调运用

采用集中式的资源检索。我们知道, 用户的最终目的是尽快得到自己所要的教学资源。这其中包含两个步骤。“找”资源和“拿”资源。其中“拿”资源不管分布式和集中式都是不可避免的要经过公共网络。基于此我们致力于在“找”这个环节来节省用户宝贵时间。

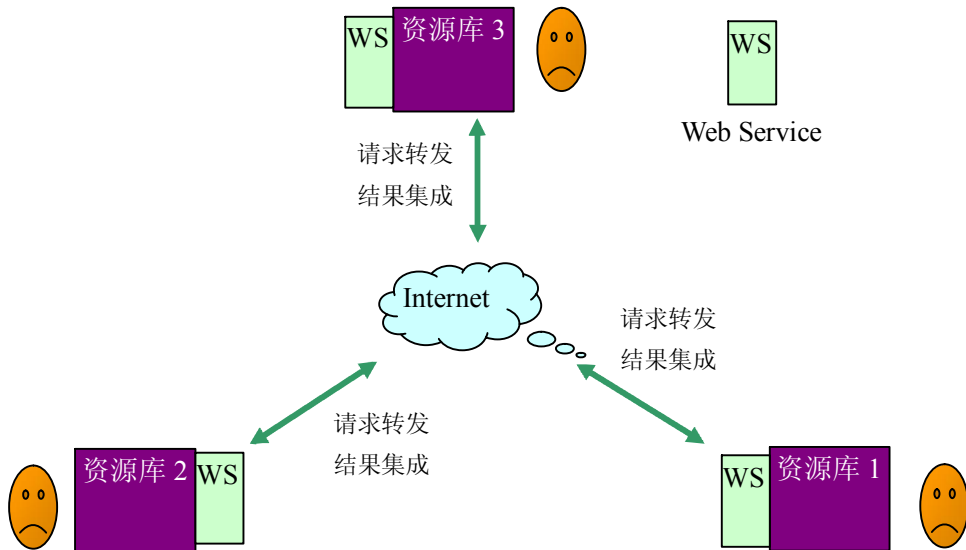


图 5 传统分布式检索示意图

我们知道网络速度是不稳定的，从图 5 中可以看出，传统的分布式检索中的用户请求（“找”）不管是请求本身还是结果返回将会受到网络因素的直接制约。

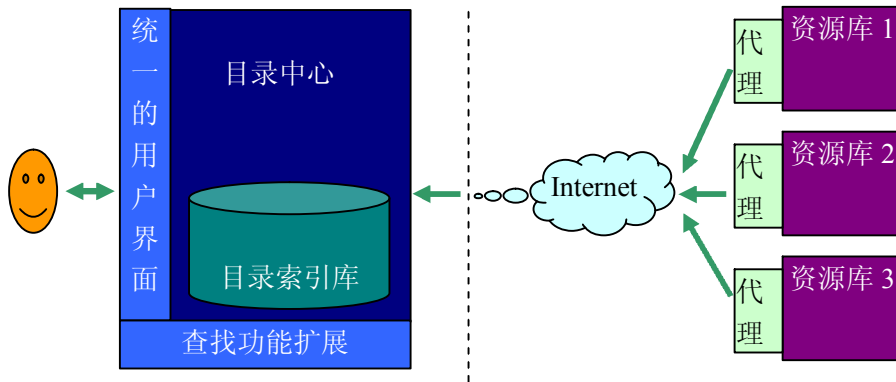


图 6 系统资源集中式检索示意图

本系统中心建立统一的 Web 服务器和数据库服务器对外提供资源检索，见图 6。用户网内资源的需求可以直接在中心进行集中式检索。由于中心预先将目录索引到数据库，不但减少了不必要的网络流量和结果集成环节的延迟还充分利用了 DBMS 的数据检索优势，极大的加快了查找速度。同时由于资源目录由中心统一管理，便于中心针对网内资源查找做灵活的集群、本意匹配等相关的查找功能扩展。

分布式用户和权限管理。对于用户来说，在查看和应用其他站点资源时的验证是透明的，所以用户在网内没有多个分节点的概念，只有所在资源节点和中心的概念。另外由于用户分布于不同节点，免除了中心庞大的用户和权限管理工作。

分布式存储。教学资源分为两个部分：描述信息和多媒体内容。其中描述信息含有检索信息和元数据信息。描述信息主要以文本形式表现。空间占用小易于存储和传输。而多媒体内容含有音频、视频、动画等形态，数据量相对较大，管理和存储代价较高，而且可能造成中心的网路阻塞。而系统中资源分布在各个分节点，更是给管理工作带来了困难。采用分布式存储后，中心只有资源检索的网络流量，没有多媒体访问传输的网络流量，极大的减轻了中心的网络流量负载。另外，如此一来中心不必对各种媒体资源的进行管理工作，而只需维护资源的目录索引信息。

结束语

分布式教育资源库系统是解决各个学校或区域教育资源共享的有效支撑条件。传统教育资源库系统采用资源的完全集中存储方式,当用户访问资源时,易造成网络流量负载过重等问题。本系统则采用资源的分布式存储,并通过建立资源目录中心,实时汇集分布式资源节点的目录信息,有效地解决了上述问题,实现了教育资源的共建共享。

参考文献

- [1] Dave Winer. What is XML-RPC[DB/OL].<http://www.xmlrpc.com/>,1999
- [2] 程建钢,何良春,韩锡斌.分布式网络教育资源库的设计与实现[J].电化教育研究,2004(11)
- [3] M.Tamer Ozsü,Patrick Valduriez. 分布式数据库系统原理[M].北京:清华大学出版社,2002
- [4] Brett McLaughlin 编,孙照林,汪东,王鹏译.Java 与 XML[M].北京:中国电力出版社,2001