

基于 XML-RPC 技术的多系统松耦合平台中的设计和应用

张永忠, 肖君, 凌强
上海远程教育集团

摘要: 针对目前教育资源产品互操作性差, 无法实现产品间数据互通等问题, 本文给出了一种基于 XML-RPC 技术的多系统松耦合平台的技术构架, 并重点介绍了该平台的主要功能和关键实现技术。

关键字: XML-RPC; 多系统松耦合平台; 教育资源库

1. 引言

随着教育资源建设的不断推进, 教育资源库平台得到了相应的发展, 出现了很多成型的教育资源系统和产品, 它们主要实现了教育资源信息的发布、基于资源的活动管理和基于资源的应用等功能[1]。但是, 从我国目前的现状来看, 各个教育资源系统和产品之间相互独立、数据不一致、信息共享程度不高、管理分散、互操作性和可移植性都较差, 难以实现教育资源系统和产品的集中统一管理以及它们之间数据的互通。

针对这种现状, 本文提出了多系统松耦合平台的设计思想, 采用基于 XML-RPC 的系统集成技术, 为各个教育资源系统和产品提供了一整套接口机制, 使得它们可以进行数据的互通, 进而实现了统一用户、统一登录、统一产品入口等重要功能, 是教育资源库平台中各子系统或产品之间进行集成的重要支撑。图 1 说明了多系统松耦合平台在教育资源库系统中的结构关系。

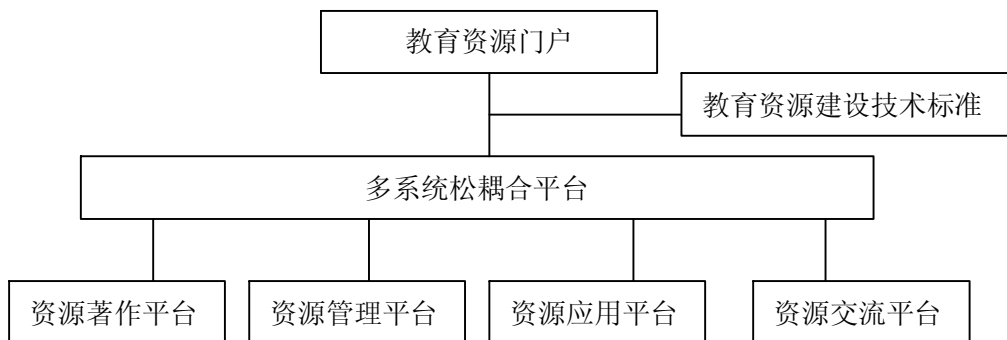


图 1 上海教育资源库系统结构

目前该系统已经通过测试, 并投入到实际的使用之中, 能够跨平台地支持基于 Java、ASP 和 PHP 开发的教育资源系统间的集成。

1. XML-RPC 技术

XML-RPC (XML Remote Procedure Calling, XML 远程过程调用) 是允许各个软件分布运行于不同服务器、不同操作系统、不同环境中, 通过网络进行远程过程调用的一种规约及

工具,它通过利用 HTTP 协议传输 XML 数据来实现通信[2]。XML-RPC 支持所有的操作系统、程序设计语言、动态或静态的环境、开放源代码以及商业应用。它是一种比 SOAP 更早的、简单的轻量级 Web Service 技术,由于 XML-RPC 非常简单,现在大多数编程语言都有了标准的或已经可用的 XML-RPC 实现。

多系统松耦合平台使用 XML-RPC 技术实现教育资源管理和应用平台中各个子系统之间的松散耦合。

2. 多系统松耦合平台的主要功能

多系统耦合平台支持基于 B/S 结构的各类体系结构应用耦合,包括基于 Linux、Windows 等操作系统的各种应用服务器、各种开发平台环境下的应用,能实现跨平台、跨服务器、跨体系结构的系统耦合。它的主要功能包括以下几个方面:

(1) 提供统一的用户数据库

多系统耦合平台提供的用户数据库作为主数据库,所有的用户数据必须存储在主数据库中(如用户注册或修改用户信息时)。各个子系统有自己独立的用户数据表,但其中的数据应从主数据库中同步而来,从而实现了统一的用户数据管理。

(2) 提供统一的用户登录功能

用户登录是权限判断的基本方式。多系统松耦合平台为各个产品提供统一的用户登录功能,用户必须从多系统耦合平台登录,才能使用各个产品,从而实现了用户登录信息的统一管理,提高了系统的安全性。

(3) 提供产品的统一管理/产品入口

针对教育资源库产品多样化以及管理和使用复杂等问题,多系统松耦合平台为用户提供了统一的产品入口,把所有产品的基本信息集中存放并予以分类,从而提高了用户的访问效率;耦合平台还为各个产品提供了统一的管理入口,对产品进行集中的设置和管理,从而提高了管理员的管理效率。

(4) 提供产品的快速部署功能

针对教育资源库产品安装、管理、维护操作复杂,工作量大,专业性要求高等问题,多系统松耦合平台提供了一套产品自动检测机制。需要挂接到耦合平台的每个产品,只需将配置文件(XML 格式)及相关模块的 LOGO 图片复制到耦合平台所在服务器的相关目录下,耦合平台即可自动检测出新产品,并按照配置文件中的信息注册产品,从而实现了产品的快速部署。

3. 系统结构设计

多系统松耦合平台主要由用户登录、用户管理、产品配置、产品部署、用户访问数据统计等功能模块组成。系统设计结构图如图 2 所示:

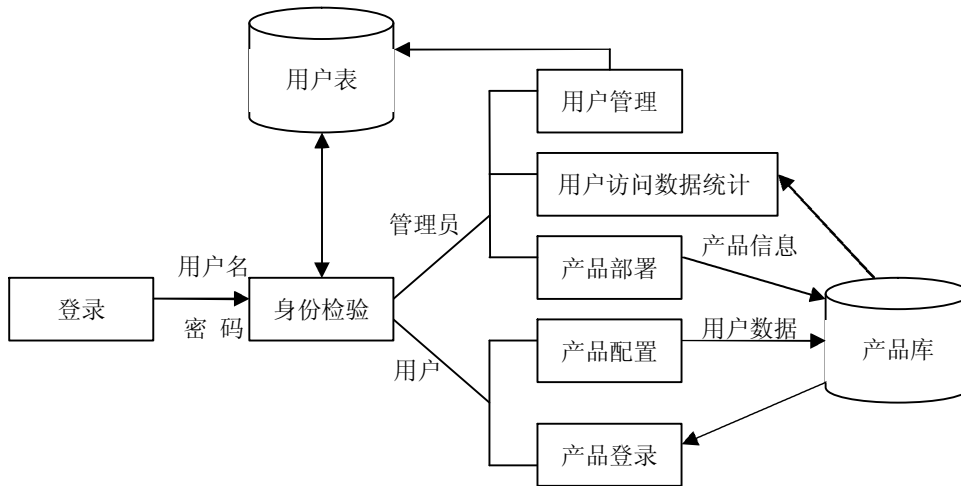


图 2 系统结构图

多系统松耦合平台功能模块见表 1:

模块名称	功能描述
用户登录模块	通过用户登录信息判断用户的身份和权限，并为用户配置相应的产品和管理信息。
用户管理模块	用于定义用户的基本权限、角色。每个角色被赋予相应的权限。管理员具有用户信息查询、用户角色设置等特殊权限。
用户访问数据统计模块	该模块供管理员使用，对耦合平台中所有产品的用户登陆次数进行按月记录，并提供按用户、产品和月份的查询功能。
产品部署模块	该模块由管理员进行配置，用于各个产品的挂接、更新、移除以及相应信息的管理。
产品配置模块	该模块由用户进行配置，用户登录后可以定制需要的产品，并进行帐号同步或帐号配置。

表 1 多系统松耦合平台功能模块表说明

4. 系统的实现

多系统松耦合平台使用 XML-RPC 技术，使耦合平台和产品进行通信，确保通讯过程的安全性，使用户直接登陆选择的 products 中。图 3 显示了用户通过耦合平台访问产品的流程。

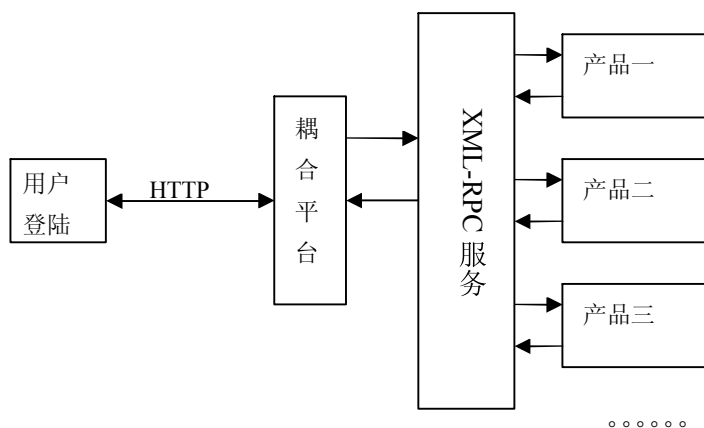


图3 多系统松耦合平台示意图

4.1 产品配置

对于需要挂接到耦合平台中的产品首先必须编写产品配置文件，此文件记录了产品及其模块的相关信息，是 XML 文件。

此 XML 文件名为“产品公司代码.产品代码.xml”，产品公司代码（例如“shzyk”）和产品代码（例如为“pro”）均由小写字母、数字及下划线“_”组成。

此 XML 文件的示例（正式挂接产品时必须将所有注释去掉）及说明如下：

```
<?xml version="1.0" encoding="gb2312"?>
<!-- shzyk.pro.xml -->
<!--
```

<product> 的各项属性是对产品基本信息的描述；
name 属性为产品名称；
version 属性为产品的版本号，必须是两位小数：如：1.00、3.03、4.10……；
outerHost 属性为产品所在服务器的外部 IP 地址或网址；
outerPort 属性为与 outerHost 对应的 WEB 发布端口，默认值为 80（即 HTTP 的默认端口）；

managePath 属性为产品后台管理页面相对于发布目录的路径；

registerPage 属性为产品用户注册页面相对于发布目录的路径（不填写为不需要注册使用的产品）；

guestName 属性为产品通用用户名，产品不需要注册时有效；

guestPsd 属性为产品通用密码，产品不需要注册时有效；

ClientPage 属性为 XML_RPC 客户端页面地址路径；

```
-->
<product name="shzyk pro" version="1.00"
outerHost="www.sherc.net"
outerPort="80" managePath="/pro/app/admin/">
</product>
```

将此 XML 文件放到耦合平台的 products 目录下：

在耦合平台的“产品管理”中使用“检测新产品”功能，将配置文件的信息导入到耦合平台数据库中，即可实现产品的挂接。

4.2 用户统一登录验证

用户统一登录验证是多系统松耦合平台最为重要的功能之一。用户登录页面、登录后台

处理以及验证当前用户是否登录均由耦合平台负责。用户统一登录验证的流程如图 4 所示。

各个产品通过判断当前用户是否已经注册了用户名 `user_id` 的 SESSION, 如果没有注册, 则跳转到耦合平台的登录页面, 同时将当前产品的编号作为参数传递给耦合平台。

用户在登录页面填写用户名、密码登录之后, 耦合平台将对其进行校验, 如果正确则进入耦合平台用户界面, 同时将用户的 SESSION ID 作为参数 `sessid` 传递给相应产品。

产品在获取 `sessid` 之后, 通过耦合平台提供的 XML-RPC 方法 `user.isUserLogin()` (将 `sessid` 作为参数), 验证用户是否登录, 如果用户已经登录, 则获得的返回值为当前耦合登录的用户名, 没有登录或无此用户则获得返回值 `"nousepower"`, 登录超时则获得返回值 `"overtime"`。

最后, 产品以耦合平台用户名和产品编号为参数, 调用耦合平台提供的 XML-RPC 方法 `user.getUserInfo()`, 获取用户的基本信息, 然后在产品所在的当前服务器上注册自身需要使用的各项信息的 SESSION, 包括用户名 `user_id` 的 SESSION。

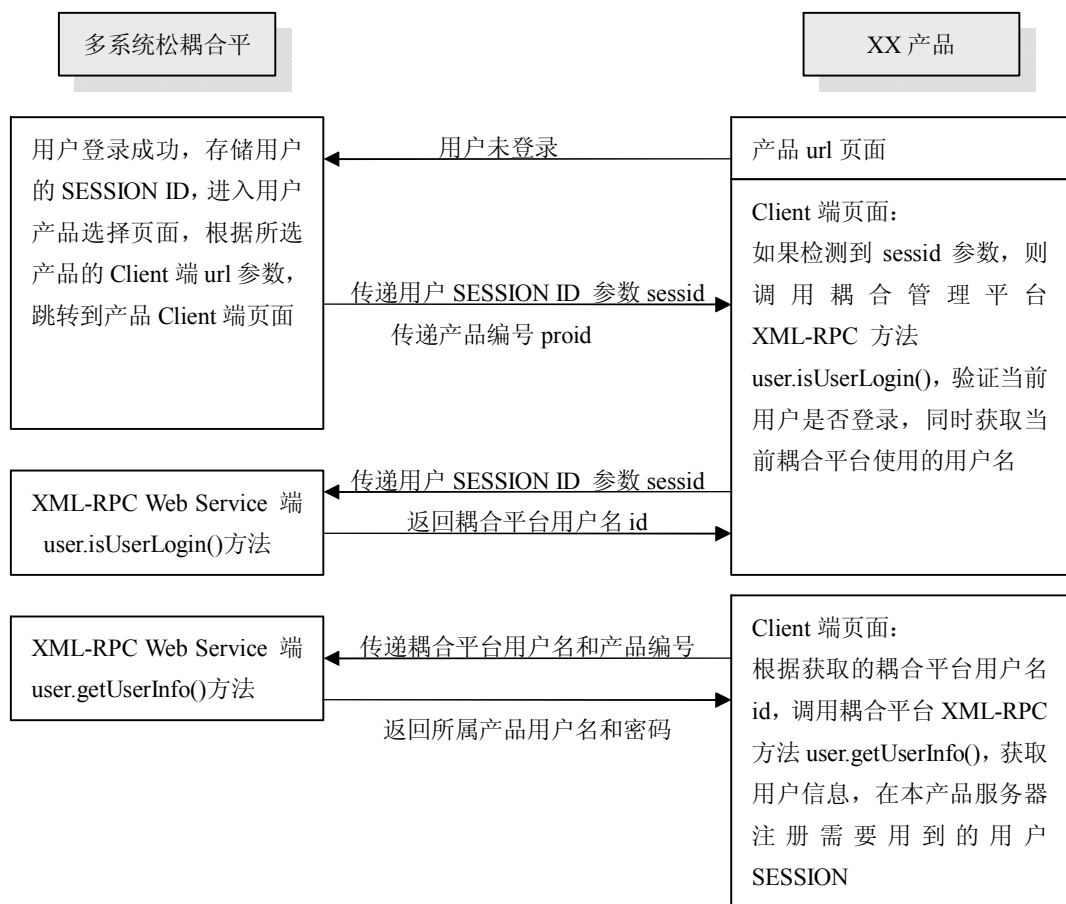


图 4 耦合平台用户登陆认证的流程图

4.3 XML-RPC 程序结构

多系统松耦合平台的具体实现由服务器程序、管理器程序和客户程序三部分组成。服务器程序用于提供 XML-RPC 服务; 管理器程序利用 XML-RPC 服务, 为客户程序提供各种使用服务的方法, 供客户端调用; 客户程序位于各个子产品中, 供耦合平台进行触发调用, 验证完成后, 即可调用管理器中的方法。

(1) 服务器端 Java 代码示例:

```
package xmlRpc;
```

```

import org.apache.xmlrpc.WebServer;
import org.apache.xmlrpc.XmlRpc;
import java.io.IOException;
public class HelloServer {
    public static void main(String[] args){
        if(args.length<1){
            System.out.println("Usage: java HelloServer [port]");
            System.exit(-1);
        }
        try{
            XmlRpc.setDriver("org.apache.xerces.parsers.SAXParser");
            //启动服务器
            System.out.println("Starting XML-RPC Server.....");
            WebServer server = new WebServer(Integer.parseInt(args[0]));
            //register our handler class
            server.addHandler("hello",new HelloHandler());
            System.out.println("Now accepting requests.....");
        }catch(ClassNotFoundException e){
            System.out.println("Could not locate SAX Driver");
        }catch(IOException e){
            System.out.println("Could not start server: "+e.getMessage());
        }
    }
}

```

(2) 管理器 Java 代码示例:

//本示例提供 sayHello 方法, 用户可根据实际需求进行功能扩充使用

package xmlRpc;

```

public class HelloHandler {
    public String sayHello(String name){
        return "Hello " + name;
    }
}

```

(3) 客户端 JSP 代码示例:

<%

//获取产品用户名和密码后进行产品用户身份认证

.....

//创建 XMLRPC 客户端对象实例, 同时指定耦合平台的 XMLRPC 路径

```

    XmlRpcClient client = new XmlRpcClient("http://服务端地址/Servlet");

```

//发送请求进行 XMLRPC 方法 sayHello 的调用

```

    result = (String)client.execute("HelloHandler.sayHello", 方法中所需参
数);

```

%>

4.4 XML-RPC 中文问题处理

在使用 Java 语言使用 XML-RPC 时，如果 request 中包含中文字符，会被自动编码成如下样式“欢”，客户端接收时将无法识别并产生乱码。针对这个问题，程序中引入了 Base64 的类，将所有的中文字符在客户端用 Base64 编码，然后通过 request 指令传送到服务器端，服务器端再用 Base64 解码。从而有效解决了 XML-RPC 中的中文传输问题。

4.5 XML-RPC 安全问题考虑

基于 xml-rpc 技术的耦合平台是一种松散耦合的、与语言和平台无关的、在组织内跨系统、跨防火墙链接的多系统平台。其安全问题主要包括：

网络安全性

支持如 SSL 等提供机密性和完整性的安全传输机制。

采用防火墙技术和用户权限的分配认证机制，保证用户只能访问具有权限的产品，来保证网络访问的安全机制。

XML 消息安全性

XML 数字签名，以便接收方可以证明消息发送方的身份。

XML 加密，提供数据元素的机密性使能够验证交换。W3C 发布 XML 密钥管理服务 (XML Key Management Services, 简称 XKMS) 的备忘录，帮助分发及管理在端点之间进行安全通信所需的密钥。

5. 结论

本文提出多系统松耦合平台的主要目的是为了整合国内优秀的教育资源库系统或产品，使用户通过单点登录模式就能方便地进入需要的教育资源库系统或产品中，用户不必手动注册并记忆多个系统帐号，从而减少了因使用不同系统提供的教育资源而产生的冗余环节，有效提高了用户的访问效率。

参考文献

- [1] 何克抗. 《教育资源建设的问题及其对策》，<http://www.edu.cn/20031126/3094951.shtml>, 2004 年
- [2] Dave Winer. What is XML-RPC. <http://www.xmlrpc.com/>, 2002 年
- [3] (美) Ethan Cerami. Web 服务精髓. 北京：中国电力出版社, 2003 年