

文章编号: 2096-1472(2017)-03-26-03

基于Web Services的数据交换平台的设计与实现

张 军

(湖南铁道职业技术学院资源信息中心, 湖南 株洲 412001)

摘 要: 基于SOAP协议的Web Services已经成为网络应用发展的主流, 利用Web Services技术完成组织内部各业务系统异构数据的交换与共享也成为较主流的解决方案。基于此, 设计了基于Web Services的数据交换平台, 文章对平台系统的框架设计做了重点阐述, 同时对平台系统的功能以及具体实现方法做了说明。平台系统使用Web Services实现对各个不同业务系统之间的数据进行交换与共享, 有效提高了系统的扩展性和模块化程度, 同时降低了开发成本, 提高了开发效率, 对类似系统的建设具有示范及借鉴意义。

关键词: Web Services; 数据交换; 数据共享

中图分类号: TP315 **文献标识码:** A

Design and Implementation of the Data Exchange Platform Based on Web Services

ZHANG Jun

(Information and Technology Center, Hunan Railway Professional Technology College, Zhuzhou 412001, China)

Abstract: As web services based on SOAP protocol has become the mainstream of the development of network applications, the heterogeneous data exchange and sharing among internal business systems through Web Services technology has become a quite commonly adopted solution. A data exchange platform is designed based on Web Services. The paper elaborates on the framework of the platform system, and specifies the functions and the implementation methods of the platform system. Web Services technology is applied in the system to implement the data exchange and sharing among different business systems, which effectively improves the system expansibility and modularity, reduces the development cost, improves the development efficiency, and offers some reference to the construction of similar systems.

Keywords: web services; data exchange; data sharing

1 引言(Introduction)

随着互联网技术的发展, 很多组织先后构建了满足不同功能需求的业务管理系统, 这些系统大多采用不同的开发技术和数据格式, 这就形成了一个巨大的异构数据环境。随着信息化程度的提高, 各业务系统间就会存在各种协议交换和业务流程的关联, 当业务处理过程中所产生的数据不能进行统一的管理, 就会出现数据遗漏、不同步、不统一、不实时等一系列问题^[1]。

数据交换就是要实现在异构的网络环境中, 不同的操作平台、不同地方和不同的数据结构, 以统一的、标准的数据展现给终端使用者^[2]; 解决各个业务系统间共享数据的一致、实时、安全、准确等问题。为企业提供全局的数据视图和完善的的数据权限管理, 让各种数据能够互联互通, 从而产生更为完善的信息服务。本文以湖南铁道职业技术学院数据中心数据交换平台开发为背景, 阐述了利用Web Services等技术实现数据共享与数据交换的基本方法、技术原理和思路。

2 系统设计(System design)

不同业务系统间要实现异构数据的交换主要面临以下两个关键问题, 首先是怎么解决不同业务异构数据源所产生的数据转换问题, 其次是要解决不同业务系统之间的数据传输和交互问题。本文采用了Web Services技术来解决数据交换和互操作问题, 以期实现多业务系统间异构数据的交流和共享。Web Services是利用SOAP协议传输标准的XML数据, 采用这种方式使得系统具有良好的跨平台性^[3]。同时, Web Services也具有较好的复用性和扩展性, 当组织部署了新的业务系统, 需要与原有系统产生数据交换时, 只需要在数据交换平台配置好相关数据源, 发布对应的Web Services接口即可, 不需要对系统功能及数据库进行任何的变动。

2.1 Web Services

Web Services采用基于XML协议格式来封装数据用以在因特网上传输, 使用SOAP协议来实现访问服务。这两项技术的使用使得Web Services具有良好的封装性和松散耦合,

对于使用者而言，只能看到服务的描述，对与调用者来说，Web Services实现的任何操作都是透明的。Web Services也同时具有标准的协议规范以及高度的可集成性等特点^[4]。Web Services体系结构主要有服务请求者、服务注册中心和服务提供者三种角色，交互主要是服务发布、服务查找和服务绑定三种操作。服务提供者完成Web Services的定义和服务的描述并将其发布至服务代理中心，服务请求者先从服务代理中心查找到相关服务的描述，然后根据服务描述完成与服务提供者的绑定，通过SOAP完成服务的调用^[5]。图1给出了Web Services的基本框架结构。



图1 Web Services体系结构图
Fig.1 Web Services architecture diagram

2.2 系统架构设计

数据交换中心采用Web Services技术进行数据操作的封装，不会对原有业务系统的功能和业务操作产生任何影响，只是将业务系统需要公开共享的数据操作当作一种服务，包装成Web Services，通过对Web Services的请求和调用来实现业务系统间对数据的交换和共享的需求。数据交换中心的总体结构是采用当前主流的三层B/S架构。B/S结构能够很好的满足当前网络开放、互联、信息随处可见的需求，采用WWW浏览器技术简化了客户端，使用通用浏览器就可以实现原来专用客户端软件才能实现的强大功能。浏览器端仅处理一下简单的数据展示和交互等相关功能，将系统主要的业务逻辑处理和数据库访问移至应用服务器，从而形成了典型的三层分布式框架结构，即展示层、逻辑处理层及数据库访问层。

数据交换中心不存储各业务系统所提供的数据，只提供所发布的Web Services访问接口和数据定义，用户或其他业务系统只需要发出服务请求，数据交换中心就能够根据请求的数据定义获取数据再返回给数据请求者。当业务系统数据库中的数据定义或是数据类别发生的变化，只需将相关的Web Services接口进行修改或是增删，然后重新发布即可，无需对数据交换中心做任何改动，这就实现了系统的松散耦合性，增强了系统灵活性和扩展性。

平台中数据请求角色(包括用户和其他业务系统)通过Web Services来实现具体的数据请求操作，Web Services通过数据库访问层与业务系统目标数据库完成数据交换，将处理后的数据返回给请求方。结合Web Services的工作原理和机制，图2展示了三层分布式数据交换平台的基本框架结构。

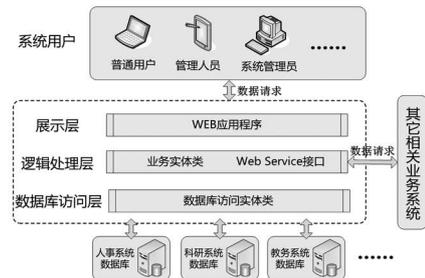


图2 系统框架结构图
Fig.2 System framework chart

3 系统实现(System implementation)

根据前面对平台系统的结构分析，数据交换平台采用微软的ASP.NET技术，使用的开发环境为Microsoft Visual Studio 2012，用C#语言来构造应用程序。为提升系统的界面友好性和交互效果，在平台系统的应用表示层使用了大量的jQuery库中相关插件，平台系统逻辑处理层主要以Web Services的形式组织封装，功能模块都被封装成Web Services发布，供表示层和其他需要数据交换的业务系统调用，大大提高了系统的模块化程度，简化了开发难度，提升了开发效率，而且使得系统具有良好的可扩展性和可维护性。下面我们以获取科研管理系统相关科研数据为例，简要说明一下Web Services的实现数据交换的过程。科研管理系统主要包括科研项目管理和科研成果管理两大核心功能，其产生的科研数据和论文数据需要交付给人事管理系统作为师资职称评定的依据。因篇幅有限，在此仅以获取教师发表论文数据为例来说明服务的实现。下面提供了Web服务端和客户端的关键代码和实现过程。

数据访问层简单查询关键代码如下：

```
public DataTable QuickSelect(string selectSQL,
SqlConnection conn)
{
//selectSQL为SELECT查询语句，conn为
SqlConnection的实例，该实例需要指定一个数据库连接字符串。
DataTable ds=new DataTable();
SqlDataAdapter adapter=new
SqlDataAdapter(selectSQL, conn);
try
{
conn.Open();
adapter.Fill(ds);
}
catch (SqlException ex)
{
this._Message=ex.Message;
}
finally
```

```

    {
        adapter.Dispose();
        conn.Close();
    }
    return ds;
}

```

数据访问层主要采用ADO.NET组件来完成数据库的连接,数据的检索和更新操作。ADO.NET提供了多种对象模型来完成的数据源的不同访问功能,本例使用了SqlDataAdapter对象完成与数据库的交互。该对象通过Fill方法将数据源数据填充到DataSet或者DataTable中,填充完成后就与数据库服务器的连接断开。

```

public class WS_KYData_Query : System.Web.
Services.WebService
{
    protected DBServer Sql_Execute=new
DBServer();
    // 构造一个数据库访问类对象。
    private SqlConnection conn=new SqlConnection
(System.Configuration.ConfigurationManager.ConnectionStrings["KYDBConnectionString"].ConnectionString);
    // 创建一个SqlConnection实例conn,获取Web.
config配置文件中的数据库连接字符串构造该实例。
    [WebMethod]
    public DataTable KYData_Query_Paper(string
paperTitle, string userId)
    {
        DataTable dt_Result=new DataTable();
        string Sql_Query="SELECT b.JSName,a.*FROM
c01_Input_LWFB a,b02_Base_JSK b WHERE a.JSID=b.
JSID";
        if (paperTitle != "")
            Sql_Query+=" AND a.LWMC LIKE
'%" +paperTitle+"%'";
        if (userId != "")
            Sql_Query+=" AND b.JSID='"+userId+"'";
        dt_Result=Sql_Execute.QuickSelect(Sql_
Query,conn);
        dt_Result.TableName="科研管理系统论文信息查
询";
        return dt_Result;
    }
}

```

上述代码利用ADO.NET组件构建了一个数据访问对象,从系统配置文档中获取目标数据库的连接字符串,根据传递的参数创建数据查询语句,然后将数据库访问字符串和查询语句传递给数据库操纵类,完成数据的查询操作。再由Web

Services将查询结果返回给调用者。因篇幅有限,此处仅展示测试页面,如图3所示。

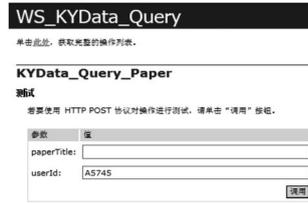


图3 Web Services测试调用页面

Fig.3 Web Services test page

输入相关参数,即可获得响应,如图4所示。

```

<diffgr:diffgram xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" xmlns:diffgr="
- <DocumentElement xmlns=""
- <JSName>领军</JSName>
<ID>762</ID>
<JSID>AS745</JSID>
<ND>2013</ND>
<LWMC>发表的论文——测试数据</LWMC>
<FBKW>计算机工程</FBKW>
<KWJB>南大核心</KWJB>
<QC>2013-12</QC>
<CBNY>2013-10</CBNY>
<YTKTM>依托课题名称</YTKTM>
<LXBH>无</LXBH>
<LXHMC>无</LXHMC>
<ISCFB>否</ISCFB>
<ZZCLSC>1</ZZCLSC>
<DQSHZT>4</DQSHZT>
<BZ>选登——测试数据</BZ>
</DocumentElement>
</diffgr:diffgram>

```

图4 Web Services调用响应

Fig.4 Web Services call response

4 结论(Conclusion)

通过该系统的建设能够全面渗透到组织内部的各项业务管理环节,能够对各个业务系统构建一个沟通交互的桥梁,并能形成一个实时、高效的数据共享平台和数据管理中心。通过实践可以证明,Web Services等技术的应用能够为信息化建设过程中各业务系统之间的数据共享、交互提供有效的解决方案,消除了信息孤岛。该系统目前已经在湖南铁道职业技术学院实施部署,完成了科研、人事、办公系统以及资产等数据的交互与共享,系统运行稳定,满足了各类用户的需求,达到了数据共享交互的目的,同时对类似系统的建设具有示范及借鉴意义。

参考文献(References)

- [1] 王涛.数据共享与数据交换系统的设计与实现[D].大连:大连理工大学,2015.
- [2] 刘嘉.基于XML和Web Services数据交换系统的设计与实现[D].广州:华南农业大学,2016.
- [3] Xiao WX,Liu Z.The Development of Web Service Composition Based on XML Security Models and AJAX Technology.International Journal of Advancements in Computing Technology,2012,4(19):83-90.
- [4] 尚龙华,安毅生,张绍阳.基于Web服务的交通数据交换过程[J].计算机系统应用,2015,24(3):260-264.
- [5] 郭晓阳.Web Services性能测试方法的设计和实现[D].北京:北京邮电大学,2010.

作者简介:

张 军(1984-),男,硕士,讲师.研究领域:可信软件,数据分析与处理。