# 一种基于 Web Service 的分布式应用系统的设计*

周 劲, 刘 洋, 藺永政
(济南大学 信息科学与工程学院, 山东 济南 250022)

**摘 要**: 基于 Web Service 的设计框架为 Web 应用开发提出了一种新的分布式应用体系架构,如何充分利用原有系统组件改造为新的 Web 服务就成为应用设计的重点。论述了将一个传统的基于组件技术的分布式应用系统改造成为一个基于 Web Service 系统的实现过程。

**关键词**: 分布式应用系统; 组件; Web Service; Web 应用
**中图法分类号**: TP311 **文献标识码**: A **文章编号**: 1001-3695(2007)02-0238-02

# Design of Distributed Application System Based on Web Service

ZHOU Jin, LIU Yang, LIN Yong-zheng
*(School of Information Science & Engineering, Jinan University, Jinan Shandong 250022, China)*

**Abstract:** The framework of design based on Web Service has put forward a kind of new distributed application system for Web application development. So how to transform original system components into new Web Services have become the emphases of application design. The realization of the transformation from a distributed application system based on component to the system based on Web Services using an actual application.

**Key words:** Distributed Application System; Component; Web Service; Web Application

随着面向对象技术的发展, Internet/Intranet 应用的不断普及, 基于组件技术(如 DCOM 或 CORBA)的多层分布式体系结构已成为当前应用程序设计的主要趋势, 其共同特点就是通过创建能够在应用程序之间可重用和共享的组件模块, 从而实现应用程序互操作性和伸缩性。但是这种传统的分布式系统设计模式由于各自采用的平台、协议、体系结构的巨大差异, 使得在面对基于异构平台的 Web 应用系统的实施与集成的挑战时, 往往显得力不从心。近年来, 随着以 XML 和 Web Service 为基础的新型分布式应用系统设计模式的崛起和发展, 为解决这一难题提供了新的思路。但是如果将原有传统分布式应用系统中所设计的大量组件全部抛弃, 重新设计新的 Web Service组件, 将造成巨大的重复浪费。为此, 本文将探讨一种在原有传统分布式应用系统基础之上进行改造, 设计成为新的基于 Web Service 的分布式应用系统的方法。

## 1 传统的分布式应用系统设计

随着 Internet/Intranet 应用的不断普及, 基于组件技术的分布式应用体系结构已被广泛采用, 目前流行的分布式对象模型主要包括 DCOM(分布式组件对象模型)[1]、CORBA(公共对象请求代理体系结构)[2]、RMI(远程方法调用)[3]。当前应用程序的设计模式也随之逐渐从传统的 C/S 两层结构转变为基于组件技术的三层分布式体系结构。

三层分布式体系结构(图 1)是将原有传统的两层结构中的客户端应用分为两部分, 即客户应用与服务器应用。新的客户应用负责用户界面和简单固定的业务逻辑; 新的服务器应用中则驻留核心的、易变的、复杂的业务逻辑。这样, 其结构就变为了(Client + Application Server) / Database Server。

图 1 基于组件的三层分布式应用系统架构

(1) 客户层: 负责用户界面及简单固定的业务逻辑。
(2) 应用层: 复杂的、可重用的、可共享的业务逻辑。
(3) 数据层: 提供对数据库的管理。

采用三层分布式体系结构有着多方面的技术优势, 包括: ①逻辑封装性。这是分布式模式中最具诱惑力的特征。当应用软件的业务逻辑规则发生改变时, 只需改变一个应用服务器的程序即可, 而无须更改客户端用户界面, 从而提高工作效率, 便于升级、维护。②性能。三层分布式体系结构将复杂的应用与商业逻辑分离出来, 由专门的应用服务器来处理, 大大减轻了客户机的功能负担, 既可以提高应用的执行速度, 又可以减少网络调用的通信量。③安全性管理。由于大量商业逻辑均驻留在应用服务器端, 信息管理部门就可以十分方便地监控服务器的运行情况, 可以大大简化管理员对系统的管理, 减轻系统维护的工作, 并确保系统的可靠运行。

## 2 基于 Web Service 的分布式应用架构

虽然传统的分布式设计模式在当前应用系统的开发中发挥着重要的作用, 但是由于它们采用的体系结构和协议均各自独立和私有, 结构紧密耦合, 使其并不适用于异构的 Internet/Intranet 环境; 另外, 由于传统组件技术要求动态申请端口, 导致无法通过防火墙的防护。这些问题都是传统分布式应用系统所无法解决的。为此, 基于 Web Service 的设计框架[4,5]通过结合面向组件方法和 Web 技术的优势, 提出了一种新的面向服务的分布式体系结构, 如图 2 所示。

Web Service 是一组通过标准的 Web 协议(HTTP)可编程访

收稿日期: 2005-10-12; 修返日期: 2006-03-27
基金项目: 国家"863"计划资助项目(2002AA4Z3240)

问的 Web 组件。其涉及的相关技术主要包括可扩展标记语言(XML)、简单对象访问协议(SOAP)、Web 服务描述语言(WSDL)、通用描述、发现与集成(UDDI)。其中, XML 是 Web Service 平台中表示数据的基本格式; SOAP 通过 XML 文档形式发送和接收消息实现异构平台上的不同系统的相互通信和共享数据; WSDL 用来描述 Web Service 所完成的功能和 Web Service 所提供的服务; UDDI 用来注册、发布已建立的 Web Service。

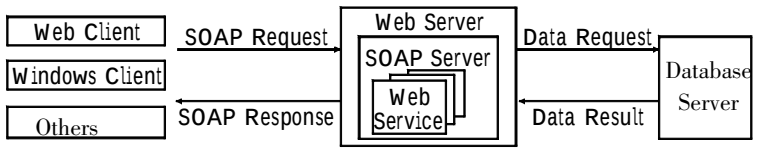


图 2　基于 Web Service 的分布式应用系统架构

从技术角度来看, Web Service 是对象/组件技术在 Internet 中的延伸。Web Service 具有完好的封装性、松散耦合、使用标准协议规范、高度可集成能力等新的特征。Web Service 最突出的优势就在于它不像传统的组件技术那样依赖特定的对象模型协议, 如 DCOM, CORBA, RMI 等, 而是利用通用的 Internet 协议和数据格式来实现服务的访问, 如 HTTP 传输协议、XML 数据格式, 从而真正实现应用系统的跨平台、可扩展和易维护性。

## 3　医疗保险管理系统的改造

### 3.1　系统简介

医疗保险管理系统是一个分别应用于社保机构、医保定点医院、医保定点药店等单位, 通过社会保险信息网络, 实现对投保人员进行统一账户管理、住院管理、结算管理、报销管理等, 以及社保局与医院、社保局与药店进行结算和数据交换管理的分布式应用系统[6]。整个系统的设计模式采用基于组件的三层分布式体系结构, 包括服务器应用程序、客户应用程序和数据库管理三大部分。服务器应用是在 PowerBuilder7.0 + Windows 2000 + MTS 平台上开发的, 以 COM 组件形式实现, 并由 MTS 进行统一有效管理, 主要负责处理大量业务逻辑, 提供相应的公共接口, 供客户应用访问使用; 客户应用则是在 PowerBuilder 6.5 + Windows 98 平台上开发的, 由用户界面、连接对象、远程对象的类定义组成, 通过调用 COM 服务器上的远程对象, 传递参数, 并返回结果; 数据库采用 Oracle 7.0。

### 3.2　系统改造

系统中整个中间业务平台即组件事务服务器的开发非常复杂, 需要实现大量组件, 供客户应用访问使用。主要包括个人账户管理、医疗报销管理、统筹管理、社保医院数据交换、社保药店数据交换、与银行数据传送等。那么, 如果将整个系统中所设计的这些组件全部抛弃, 重新设计新的 Web Service 组件, 将造成巨大的人力和时间的浪费。为此, 选择利用 Microsoft SOAP Toolkit 3.0提供的一整套 COM 对象来实现 COM 组件向 Web Service 组件的转换[7], 这样, 我们既能够实现原有组件的重用, 同时又可以享受到 Web Service 所带来的好处。下面, 以住院登记组件(register.dll)为例介绍实现过程。

#### 3.2.1　应用层实现 COM 组件向 Web Service 的转换

(1) 设置虚拟目录(“d: \webservices \register \”), 以便用户可以通过 HTTP 访问到服务相关的服务描述(WSDL)文件和服务组件描述(WSML)文件。

(2) 利用 Microsoft SOAP Toolkit 3.0 提供的 WSDL Generator 指定 register.dll 中 COM 对象的相应方法(住院登记), 生成与服务相关的服务描述(WSDL)文件和服务组件描述(WSML)文件, 将其改造为对应的 Web 服务。

#### 3.2.2　客户层实现连接访问

(1) 创建 Web Service 代理对象 Proxy_register, 指定 WSDL 文件。

(2) Client 端通过对象 SoapConnection, 连接 SOAP 服务器, 调用相应的 Web Service。

局部变量:

```
SoapConnection l_soapconn//连接对象
Proxy_register l_proxy//代理对象
Long l_ret
String l_url, l_proxy_name
```

程序代码:

```
l_url = "http://localhost/register/"
l_proxy_name = "Proxy_register"
l_soapconn = Create SoapConnection
l_soapconn.SetOptions("SoapLog = ~"soaplog.txt~"")
l_ret = l_soapconn.CreateInstance(l_proxy, l_proxy_name, l_url)
if l_ret < >0 then
   MessageBox("出错","创建实例失败,请检查代理对象是否存在")
   return
end if
try
   l_result = l_proxy.f_register()//调用住院登记方法
catch (SoapException e1)
   MessageBox("出错","调用服务失败")
catch (PBXRuntimeError e2)
   MessageBox("出错","调用服务时出现运行时错误")
catch (RuntimeError e3)
   MessageBox("出错","调用服务时出现不可知错误")
end try
Destroy l_soapconn
```

## 4　总结

本文首先分析了传统的基于组件技术的分布式应用体系结构的不足, 然后提出了一种新的基于 Web Service 的分布式应用体系结构, 并实现了从传统的基于组件技术的分布式应用系统向基于 Web Service 的分布式应用系统的转换。与传统的分布式应用体系结构相比, 基于 Web Service 的分布式应用体系结构不仅能更好地适应现代 Web 应用的飞速发展, 还能更好地体现面向对象思想, 也更适用于异构平台的 Web 应用系统的实施与集成。

参考文献:

[1] irtland M. Designing Component-based Applications[M]. Microsoft Press, 1999.

[2] The Common Object Request Broker: Architecture and Specification, Revision 2.0[EB/OL]. http://www.omg.org/corba/corbiiop.htm, 1995-07.

[3] DCE1.1: Remote Procedure Call Specification[EB/OL]. http://www.rdg.opengroup.org/public/pubs/catalog/c706.htm, 1995.

[4] 柴晓路, 梁宇奇. Web Service 技术、架构和应用[M]. 北京: 电子工业出版社, 2003.

[5] 程炜, 杨宗凯. 基于 Web Service 的一种分布式体系结构[J]. 计算机应用研究, 2002, 19(3): 105-107, 111.

[6] 周劲, 李景洲, 张世栋. 基于组件的数据库应用设计[J]. 南京大学学报, 2000, (36): 111-113.

[7] 林弘之. Web Service 原理与开发实务[M]. 北京: 电子工业出版社, 2003.

作者简介:

周劲(1976-), 男, 河北乐亭人, 讲师, 硕士, 主要研究方向为软件工程、数据挖掘、数据库应用; 刘洋, 女, 硕士研究生, 主要研究方向为软件工程、数据挖掘; 蔺永政, 男, 讲师, 硕士, 主要研究方向为数据库应用。