

【理论与探索】

基于 .NET Remoting 的文书收发系统的设计与实现*

田海江¹, 王德才², 唐业军², 孙玉萍²

(1. 重庆邮电大学, 重庆 400065; 2. 防化指挥工程学院, 北京 102205)

摘要: 阐述了分布式应用开发的历史及现状, 阐述了使用 .NET Remoting 技术进行分布式应用开发的优点, 以及使用 .NET Remoting 实现远程通信的基本原理. 使用 .NET Remoting 技术开发文书收发系统, 给出了系统的总体设计及功能的实现思路和方法.

关键词: .NET Remoting; 分布式应用; 文书收发; 设计与实现

中图分类号: TP391.9

文献标识码: A

文章编号: 1006-0707(2008)02-0091-03

文书收发系统是网上对抗演练系统及指挥信息系统的重要组成部分, 是一个典型对等结构的分布式应用系统. 当前用于分布式应用系统开发的技术较多, 本研究选择具有快速开发、可扩展性好的 .NET Remoting 开发文书收发系统.

1 分布式应用开发与 .NET Remoting

随着网络技术的飞速发展, 目前大多数应用系统都需要依托网络运行, 因此分布式应用开发在目前的应用程序开发中占据重要的地位.

1.1 分布式应用开发. 简单地讲建立在计算机网络上、由多台计算机组成的应用系统都可以称为分布式应用系统. 计算机网络出现之后, 基于计算网络的应用不断涌现, 也就出现了最初的分布式应用^[1].

分布式应用开发最初是使用套接字进行通信的. 尽管套接字可以完全控制通信, 但是开发人员必须自己构建消息传递系统, 建立并解释数据流, 对大多数通用的分布式应用程序来说, 这种工作太单调乏味、且比较困难. 后来出现了远程过程调用(RPC), 开发人员可以通过 RPC 使用执行本地过程调用所需的相同语义, 实现远程通信. RPC 使远程通信比套接字编程更加友好.

随着时间的推移, 当编程领域从过程性编程转移为面向对象开发之后, 又出现了分布式对象技术. 分布式对象技术是伴随网络技术发展起来的一种面向对象的技术. 分布对象技术允许运行在一台机器上的应用程序或对象访问另一台机器上运行的对象. 正如 RPC 让远程过程好像本地过程

一样, 分布式对象技术让远程对象看上去像本地对象.

在 .NET Remoting 出现之前, 有代表性的主流分布式对象技术主要有对象管理组织(OMG)提出的 CORBA(Common Object Request Broker Architecture)、Microsoft 的 DCOM(Distributed Component Object Model)和 SUN 公司的 Java RMI. 其中 Java RMI 主要是面向 Java 应用的, CORBA 仅仅是一个规范, 而不是一个实现. 在 .NET Remoting 出现之前, DCOM 是开发 Microsoft 平台上分布式应用程序的首选方法, 但是对于一般的开发人员来说, 理解和使用 DCOM 都不是很容易. 正如 .NET Framework 取代 COM 成为建立组件的首选方式一样, .NET Remoting 取代了 DCOM 成为使用 .NET Framework 建立分布式应用程序的首选方式^[2].

1.2 .NET Remoting. 从微软的产品角度来看, 可以说 .NET Remoting 就是 DCOM 的一种升级, 它改善了很多功能, 并极好的融合到 .Net 平台下. .NET Remoting 提供了一种允许对象通过应用程序域与另一对象进行交互的框架. 在 Remoting 中通过信道(channel)来实现两个应用程序域之间对象的通信, 如图 1 所示.

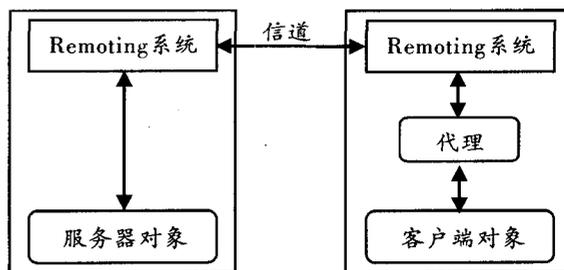


图 1 通过 .NET Remoting 进行 2 个应用程序域之间的通信

* 收稿日期: 2008-03-05

基金项目: 北京市人防局人防指挥训练系统科研基金资助项目.

作者简介: 田海江(1981—), 男, 黑龙江哈尔滨人, 硕士研究生, 主要从事计算机技术研究.

客户端通过 Remoting 访问信道以获得服务端对象,再通过代理解析为客户端对象.这样,客户端对象就能像操作本地对象一样操作服务器对象,包括调用它的方法.客户端在获取服务器端对象时,并不是获得实际的服务端对象,而是获得它的引用,在客户端称为“代理”.这既保证了客户端和服务端有关对象的松散耦合,同时也优化了通信的性能^[3-5].

Remoting 的信道主要有 2 种: Tcp 和 Http. Tcp 信道提供了基于 Socket 的传输工具,使用 Tcp 协议跨越 Remoting 边界传输序列化的消息流. Tcp Channel 类型默认使用二进制格式序列化消息对象,因此它具有更高的传输性能. Http Channel 提供了一种使用 Http 协议,使其能在 Internet 上穿越防火墙传输序列化消息流. 默认情况下, Http 信道使用 Soap 格式序列化消息对象,因此它具有更好的互操作性. 通常在局域网内,更多地使用 Tcp Channel; 如果要穿越防火墙,则使用 Http Channel.

在访问远程类型的一个对象实例之前,必须通过一个名为 Activation 的进程创建它,并进行初始化. 这种客户端通过信道来创建远程对象,称为对象的激活. 在 Remoting 中,远程对象的激活分为两大类: 服务器端激活和客户端激活. .Net Remoting 把服务器端激活又分为 Single Ton 模式和 Single Call 模式 2 种. Singleton 为有状态模式,如果设置为 Single Ton 激活模式,则 Remoting 将为所有客户端建立一个对象实例. 如果想要在所有客户之间共享一些数据,则可以使用这种对象类型. Single Call 是一种无状态模式,每一次调用远程对象时,都会常见一个新的实例. 有些情况要求远程对象实例的每个客户端引用都是清楚明确的, .NET Remting 为此提供了客户端激活.

2 系统设计^[6-8]

为了便于实现和扩展文书收发系统,整个系统采用多层结构. 从逻辑上将其划分为 3 个层次: 用户界面层、文书收发服务层、底层数据支持层. 其中用户界面层负责和用户进行交互,为用户提供操作界面,文书收发服务层是整个系统的核心,该层利用 .NET Remoting 技术实现具体的文书发送和接收等基本功能接口,这些功能接口供用户界面层调用,数据支持层为整个系统提供数据支持,例如保存文书发送与接收信息,记录各收发席位的 IP 地址等.

系统采用完全对等的结构,即任意席位之间可以相互发送文书,该系统的一个运行实例即可以向其他运行实例发送文书,也可以接受其他运行实例发送过来的文书. 因为采用完全对等的模式,没有一个各个席位都知道的主服务器,所以必须有一个机制让各个席位之间可以相互找到对方,在本研究开发的系统中,通过预定义配置参数来实现,具体方法是在本地数据表中保存下所有席位名称和 IP 地址,当需要向某个席位发送文书时,根据其席位名称查找对应机器的 IP 地址,然后向指定 IP 地址的机器上运行的程序实例发送文书.

3 系统实现^[9-10]

本研究中的文书收发系统具体采用 Visual Studio. NET 2005 作为开发环境,编程语言为 C# .NET. 系统实现主要包括文电收发服务类的实现、文书收发功能的实现及系统其他功能的实现 3 部分.

3.1 文电收发服务类的实现. 为了利用 .NET Remoting 技术实现文书收发功能,在系统中定义了文电收发服务类 WendienShoufaServer, 该类是一个远程对象类,用于支持文书收发功能的实现. 在 .NET Remting 中,远程对象类必须继承自 MarshalByrefObject 的基类. WendienShoufaServer 类是整个系统实现的基础,比较简单,其核心是成员函数 ReceiveWenshu (Wenshu wenshu), 该函数为文书收发功能的实现提供了基础,其参数类型是自定义的一个结构,包含了文书的具体内容及细节信息,例如文书的主题提示、密级、发送席位、目标席位、发送时间、以及分类等信息. 为了将整个文书结构对象而不是对象的引用传递到客户端,必须使该结构对象串行化地通过信道,所以在定义该结构时需要使用 [Serializable] 特性将其标记起来.

ReceiveWenshu (Wenshu wenshu) 成员函数的具体功能是将其参数中包含的具体文书信息保存到本地的数据表中,将文书内容以文件的形式保存到硬盘上. 核心代码为:

```
//将收到文书记录文书保存到数据库中,代码略
.....
//将文书保存到相应文件夹下
string filePath = "收到文书 \ \" + newrow["FENLEI"].ToString() + " \ \"
+ newrow["WENJIANMING"].ToString();
System.IO.FileStream stream = new System.IO.FileStream
(filePath, FileMode.OpenOrCreate);
stream.Write (wenshu. neirong, 0, wenshu. neirong.
Length);
stream.Close();
```

此外,在 WendienShoufaServer 类中还实现了 ChaiyueWenshu() 方法,当目标席位拆阅文书时,该方法用于通知文书的发送席位: 该文书已经被拆阅.

3.2 文书收发功能的实现. 文书收发功能是该系统的基本功能,其实现原理如图 2 所示(席位 1 向席位 2 发送文书).

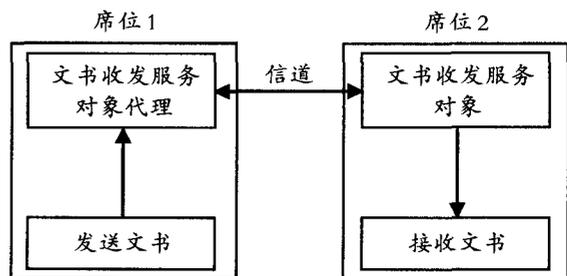


图 2 文书收发实现原理

当系统启动时,会自动注册一个文书收发服务远程对

象,用于监听其他席位的请求,代码为:

```
//创建信道
TcpChannel MyChannel = new TcpChannel(9001);
//注册信道
ChannelServices.RegisterChannel(MyChannel);
//注册远程对象
RemotingConfiguration.RegisterWellKnownServiceType (
typeof( WendianShoufaServer ),
"WendianShoufa", WellKnownObjectMode.Singleton);
```

在图2所示的情况下,无论是运行于席位1还是席位2上的系统实例,在启动时都会注册一个文书收发服务远程对象.当席位1向席位2发送文书时,首先创建一个Wenshu结构的实例wenshu,并为其各字段设置相应的值,保存文书的具体内容,然后激活席位2中文书收发对象的代理,并以wenshu作为参数,通过代理调用文书收发服务对象的接收文书方法,即上面提到的ReceiveWenshu()方法,把文书的具体内容及其他信息保存到席位2所在的机器中,从而席位1实现了向席位2发送文书的功能,而席位2也实现了接收来自席位1文书的功能.核心代码为:

```
//加载文书内容
System.IO.FileStream stream =
new System.IO.FileStream(txtFileName1.Text, System.IO.
FileMode.Open, System.IO.FileAccess.Read);
byte[] fileBuffle = new byte [stream.Length];
foreach(object tempObj in lbMubiaoXiwei1.Items)
{
//为目标席位构造一个文书对象
Wenshu wenshu = new Wenshu();
//为文书对象设置相应的字段,代码略
.....
//添加发送记录,代码略
.....
//发送文书
SendWenshuTread threadObj = new SendWenshuTread
((XIWEI)tempObj, wenshu);
.....
Thread thread = new Thread(new ThreadStart(thread-
Obj.ThreadProc));
thread.Start();
}
```

为了实现同时向多个席位发送,上面的代码采用了多线程技术,每个线程负责向一个席位发送文书,即通过代理调用目标席上位文书收发服务远程对象的ReceiveWenshu()方法.线程函数threadObj.ThreadProc()的核心代码为:

```
//激活目标机器文书收发服务对象
string ipStr = mubiaoXiwei.IPAddress;
WendianShoufaServer mubiaoServer =
(WendianShoufaServer) Activator.GetObject(typeof(Wen-
dianShoufaServer),
"tcp://"+ ipStr + ":9001/WendianShoufa");
```

.....

```
//调用目标席位文书收发服务对象的 ReceiveWenshu
()方法,接收发送过来的文书
mubiaoServer.ReceiveWenshu(sendWenshu);
```

3.3 其他功能的实现.除了文书收发这一基本功能外,作为一个完整的文书收发系统还有许多其他具体的功能需要实现,例如文书拆阅功能,即当目标席位拆阅了文书之后,要通知发送席位,这样发送席位就知道目标席位已经阅读了文书.拆阅文书功能实现的基本思路是,当目标席位拆阅文书时,首先打开文书供目标席位阅读,然后获取发送席位的文书收发服务对象,调用其ChaiyueWenshu()方法,通知发送席位目标席位已经拆阅文书,其原理和文书收发功能的实现原理基本相同.

为了顺利实现文书收发及拆阅功能,还需要确保在每个席位上只能运行一个程序实例.该功能的实现思路是检查当前运行的所有进程,查看文书收发系统的进程是否已经运行,如果已经运行则不再创建新的程序,只是简单使原来运行的文书收发系统处于活动状态,从而使其主界面在屏幕的最前端显示,如果没有文书收发系统的进程已经运行则创建一个新的程序实例.

此外,为了改善界面的友好性,当不发送和接收文书时,将文书系统隐藏到托盘中.当单击关闭按钮关闭系统时,系统会给出提示,询问是将系统放入托盘还是退出系统;当接收到文书或单击托盘时,则在屏幕上显示系统的主界面.

4 结束语

使用.NET Remoting开发分布式应用程序的优点在于可以使编程变得非常简单,它将远程调用的技术细节隐藏起来,使得可以像调用本地对象一样调用远程对象.而且如果将.NET Remoting应用程序配置成最佳性能,其速度可比得上DCOM的速度,这个速度是非常快的.此外,.NET Remoting还是用于对等通信的好技术.所以本研究选择使用.NET Remting来开发文书收发系统.目前,已经将本研究中开发的文书收发系统集成到人防指挥训练系统中,用于人防指挥训练,并取得了很好的效果.

参考文献:

- [1] 胡晓峰.战争模拟引论[M].北京:国防大学出版社,2004.
- [2] 张昆琪.Microsoft.NET Remoting权威指南[M].北京:机械工业出版社,2003.
- [3] 黄承安,谢东文,许聪.CJHJ网络应用案例导航[M].北京:中国铁道出版社,2003.
- [4] 李训勤,董连强,宗先贵.网上对抗演习[M].北京:军事科学出版社,2002.
- [5] 李敏波.CJHJ高级编程[M].北京:清华大学出版社,2006.

(下转第126页)

习题等资源放到网站上,方便学生进行参考学习。

3 个性化与整体相结合,改变教学方式

以往的课程教学都是针对所有的学生,从头到尾都是采取同样的方式进行教学,但由于学生的层次和水平的不平衡,对于教学方式的喜好也不尽相同,使得教学的效果始终不能有所突破。在该门课程的教学,我们按照个性化与整体相结合的原则,改变了以往单一的教学方式,采用讨论式、开放式和研究式等多种教学形式,对于重点章节进行“订单式”讲授。在课程开始前让学生先查阅相关资料进行预习,有兴趣的学生可制作 PPT 进行讲解,再进行课堂集中教学,这样的教学方式,既满足了全体学生的教学要求,又激发了部分学生参与课堂学习的兴趣,使教学方式呈现多元化。

4 实践教学与理论教学相结合,增强教学效果

学生在课堂上获取的知识只有在实践中才能消化、理解和掌握,并在实践中获得创新^[4]。“材料表面工程”课程是一门理论与实践相结合的课程,生产实际应用性强。我们主要从实验教学、生产实习及科研助手 3 个方面来增强学生在表面工程技术方面的实践能力,强化教学效果。

重庆工学院表面技术设备(如热喷涂设备、等离子渗氮炉、磁控溅射镀膜仪等)较为齐全,开设有大量综合性实验如“热喷涂涂层的制备与性能测试”“气相沉积薄膜的制备与分析”“电刷镀涂层的制备与性能测试”“金属基涂料的制备及其组织性能分析”等,学生可根据自身的兴趣爱好进行选择和设计实验。生产实习是重要的实践教学环节之一。重庆工学院与长安集团、东风汽车有限公司等大型企业签订了实习基地协议,进行生产实习。在时间安排方面,以前是先实习再进行理论学习,在实习中因不具备表面技术的专业基本知识,效果不太理想。改革后将先进行理论知识的学习再实习,实习中理论与实践相结合,进一步深化教学内容,强化教学效果。我们还鼓励有兴趣的学生在大二、大三阶段就成为教师的科研助手,积极参与教师的相关科研活动,培养动手及创新能力。

5 多种考核形式相结合,科学进行教学评价

成绩评定是课程建设的重要环节之一,是维护学生学

习积极性的重要激励机制,是帮助学生对知识进行建构的重要手段,也是对教学效果的反馈,可以促进教学工作的改善。用考核和评价的方式推动学生的学习朝最佳方向发展,不仅要考核学生的知识和解决实际问题的能力,更要考核学生得出结论的过程,强调评价的真实性、全面性^[5]。在传统的理论教学中,考试大多是在课程结束时进行,结果导致学生平时没有一点压力,很多学生不能主动学习,仅靠考前突击,对知识一知半解,更不会灵活应用。改革后的“材料表面工程”课程的考试采用小论文和 PPT 制作答辩、课外作业、单元测试、期末考试等多种考核形式,全面检测学生的学习效果,科学地评价学生的综合素质和创新能力。

6 结束语

经过不断的教学改革和实践,“材料表面工程”课程教学中各章节重点、难点的教学安排更趋于合理,学生学习该门课程的兴趣更加浓厚。将课堂上所学的理论应用到科研与实践教学中,不断深化了教学内容,学生的动手能力及创新意识得到了明显提高。然而这门课程的教学理论及实践性非常强,并受到实习基地、实验设施等各方面条件的限制,我们仅在该门课程教学改革中做了一点尝试,“材料表面工程”课程教学改革还需要进行更进一步的探索。

参考文献:

- [1] 赵新平.谈课程建设评估[J].长治学院学报,2006,23(3):63-64.
- [2] 时蕾,卓克垒,李建平.“有机化学”精品课程建设的探索与实践[J].河南师范大学学报:教育科学版,2006,25(3):73-75.
- [3] 曹红霞,杨树.《分析化学》课程理论与实践教学方法探析[J].开封大学学报,2006,20(3):46-47.
- [4] 朱斌,陈呈频,何兴.构建大学生创新教育长效机制的思考[J].中国大学教学,2006(1):16-18.
- [5] 胡忠青.《媒介素养》课程建设探讨[J].鞍山科技大学学报,2006,29(1):81-84.

(上接第 93 页)

- [6] David Talbot. Introducing .NET Remoting[EB/OL]. [2005-04-21]. <http://www.csharp4help.com/archives/archive157.html>.
- [7] Microsoft Corporation. .NET Remoting Overview. [EB/OL]. [2003-08-11]. [http://msdn2.microsoft.com/en-us/library/kwdt6w2k\(vs.71\).aspx](http://msdn2.microsoft.com/en-us/library/kwdt6w2k(vs.71).aspx).
- [8] Building a .NET Remoting Application[EB/OL]. [2002-05

- 11]. <http://www.beansoftware.com/NET-Tutorials/Remoting-Application.aspx>.

- [9] .NET Remoting Tutorial [EB/OL]. [2002-11-13]. <http://www.beansoftware.com/NET-Tutorials/NET-Remoting-Tutorial.aspx>.
- [10] Mark Strawmyer. .NET Remoting[EB/OL]. [2004-07-24]. <http://www.developer.com/net/cplus/article.php/1479761>.