

COM和CORBA的桥接与应用

钟 灿* 钟本善 周熙襄

(成都理工大学信息工程学院 成都 610059)

【摘要】根据CORBA规范,阐述了桥接组件的体系结构和六种桥接模型,并引申出八种桥接方式,比较了各种桥接方式的性能差异,利用自定义桥接组件,把建立在UNIX平台上的系统扩展为跨UNIX和Windows平台的分布系统。经测试验证,系统符合设计的要求,改善了原系统的不足。

关键词 桥接; 组件对象模型; 公共对象请求代理结构; 接口描述语言; 接口; 分布系统
中图分类号 TP311 文献标识码 A

COM/CORBA Bridging and Applications

Zhong Can Zhong Benshan Zhou Xixiang

(School of Information Engineering Chengdu University of Technology Chengdu 610059)

Abstract According to the specification of CORBA, the architecture of bridge component and six kinds of models of the bridge were expatiated. These models lead to eight ways to create the bridge component. After analyzing the performance of these kinds of bridge, a user-defined bridge was applied to create a distributed system under UNIX and Windows that was originally set up under UNIX. The system was testified to be successful.

Key words bridge; component object model; common object request broker architecture; interface description language; interface; distributed system

在众多的分布式对象中,OMG的CORBA和微软的COM/DCOM是两种竞争激烈、占据主导地位的技术。在 Windows 平台下,COM/DCOM主宰服务器和客户市场,微软提供了一系列强有力的服务,来创建基于COM/DCOM的分布系统。但COM所支持的平台仅为Windows系列,COM/DCOM在Java内的使用必须有微软的Java虚拟机(微软JVM)支持,而且微软支持的COM/DCOM的JVM并不支持非Windows平台。CORBA与COM/DCOM有一系列相同点,是主流的远程体系结构,侧重于分布式对象的远程方法调用,提供了健壮的跨语言和跨平台支持,从而可以在众多的操作系统平台上创建服务器。在异构平台分布式对象中,COM/DCOM和CORBA互相补充,两者都起到了重要的作用。桥接COM和CORBA可以沟通COM与CORBA的世界,变两者的差异为互补。

1 桥接的体系结构和实现方式

1.1 桥接组件的体系结构

OMG的CORBA规范指出,COM和CORBA的一系列相同点有利于连接体系结构的创建,其最重要的相同点是COM和CORBA都依赖于接口,为COM和CORBA服务器提供了外部的入口点。在大多数情况下,可以直接作CORBA接口和COM自定义接口(非自动接口)之间的映射。CORBA接口和COM自定义接口都允许用户自定义数据类型,如结构、联合以及数组,也允许各种原始数据类型。连接规范还阐明CORBA和COM

2002年11月14日收稿

* 男 28岁 硕士生 主要从事中间件的开发与应用方面的研究

自动接口之间的映射,涉及到视图(View)和目标(Target)概念。在规范中,一个视图与客户机相联系,而目标与服务器相连接,且有下面几种视图与目标的映射关系:

- 1) CORBA目标的COM自定义接口视图;
- 2) COM自定义接口目标的CORBA视图;
- 3) CORBA目标的COM自动视图;
- 4) COM自动接口目标的CORBA视图。

系统B中的对象映射到系统A的过程如图1所示。根据规范,桥接的模型有六种,如图2所示。按桥接组件物理位置的不同,可以分为本地和远程桥接。图2中,模型1、2、3、5为本地桥接,模型4、6为远程桥接。由于网络传递数据时间上的开销大,远程桥接中一次函数的调用(包括参数传入和函数的返回值与传出参数的传出)的开销比本地桥接中高一个数量级。

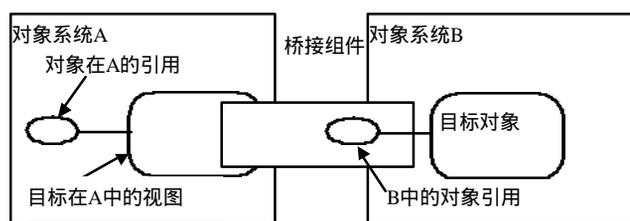


图1 对象映射

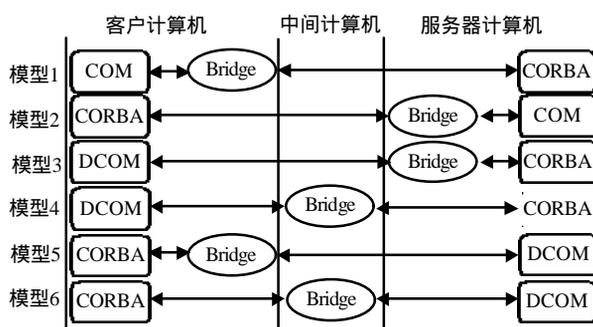


图2 桥接方式

1.2 桥接组件的结构

两种桥接组件的类型如图3所示。



图3 桥接组件两种类型

1.3 桥接方式

根据图2和图3,引申的桥接方式有以下八种:

- 1) DCOM客户、CORBA服务器和桥接组件在同一台计算机上;
- 2) DCOM客户、CORBA服务器通过IIOP连接,桥接组件在客户端计算机上;
- 3) DCOM客户与远程CORBA服务器通过DCOM连接,其中桥接组件在服务器端计算机上;
- 4) DCOM客户与远程CORBA服务器通过DCOM和IIOP连接,桥接组件在第三方计算机上;
- 5) CORBA客户、DCOM服务器和桥接组件在同一台机器上;
- 6) CORBA客户与远程DCOM服务器通过IIOP连接,桥接组件在服务器端计算机上;
- 7) CORBA客户与远程DCOM服务器通过DCOM连接,桥接组件在客户端计算机上;
- 8) CORBA客户与远程DCOM服务器通过IIOP和DCOM连接,桥接组件在第三方计算机上。

由于COM/DCOM客户和服务器在同一台计算机上时可以分为进程内和进程外服务器,因此桥接方式1)、2)、5)、6)可以分为进程内桥接组件和进程外桥接组件。经比较,进程内桥接组件安全性低但时间开销少,进程外桥接组件由于跨进程要通过存根和代理,需进行数据的列集和解列,时间开销较多。在下面的性能分析中,对于桥接方式1)、2)、5)、6)均采用进程内桥接组件。

1.4 桥接组件的实现方法

自定义桥接法,实现桥接COM和CORBA最简单的方法是编写一个特殊的组件,在内部既可以使用COM又可以使用CORBA,并能够将一种模式映射为另一种模式。创建组件时,必须拥有支持COM和CORBA的

软件开发环境：1) 基于Windows平台的Visual C++，不仅提供对COM的极好支持，而且主要的CORBA产品都支持Visual C++(后面桥接的测试和实现都采用这种开发环境)；2) 微软Java虚拟机(微软JVM)，支持COM，也可以用CORBA/Java ORB创建自定义桥接组件。此外，COM的纯Java实现和CORBA/Java ORB 可以允许桥接组件用于非Java虚拟机上。

商业化桥接法，其产品主要都由CORBA开发的软件系统，而且一般都为自己的CORBA ORB产品定做，其中有Iona的OrbixCOMet，一个功能齐全的桥接产品；PeerLogic提供的DAIS COM2CORBA，重点放在CORBA和COM之间的自动化映射上；VisualEdge的ObjectBridge，不提供专用的ORB，重点放在广泛的CORBA、COM和Java中间件的应用上。

企业应用服务器提供对COM和CORBA共同实现的客户和服务器的支持，解决分布式系统的基础框架问题，允许开发者将重点放在业务问题上，而不是放在基础框架中，主要有微软的COM方法与CORBA/EJB(Enterprise JavaBeans)组合的方法。

1.5 六种桥接方式的性能分析

1) 测试软件：

Windows 2000 server ,
Microsoft Visual C++ 6.0 ,
Inprise Visibroker CORBA ORB ;

2) 参数类型：String , Double , Long ;

3) 主要函数：GetString(),GetDouble(),
GetSum(),Increase() ;

4) 调用过程如图4所示；

5) 测试结果如表1所示；

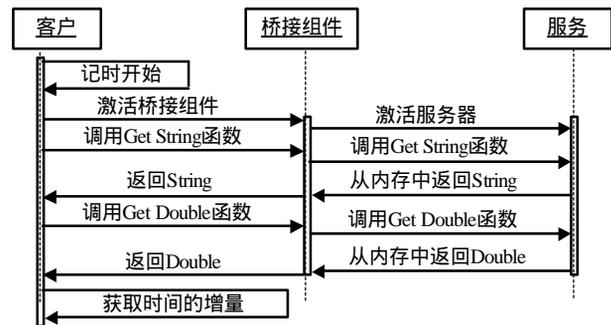


图4 调用顺序图

表1 测试结果

桥接方式	1	2	3	4	5	6	7	8
时间/ms	5.148	5.208	5.237	5.217	5.761	5.624	5.321	5.939

6) 结果分析

当DCOM客户调用CORBA 服务器的函数时，使用GetObject()和 CoCreateInstanceEx() 来获取对象。客户程序把桥接组件的地址传入CoCreateInstanceEx()，并且获取ICORBAFactory对象。客户把服务器的地址传入ICORBAFactory的GetObject()，并且获取IUnknown对象。通过客户来确定CORBA 服务器和桥接组件的位置信息，开发程序时就有足够的灵活性。当CORBA客户调用DCOM服务器的函数时，使用_bind()获取对象时灵活性教小，时间开销较多。桥接组件在第三方计算机上时，无论是DCOM客户调用CORBA 服务器还是CORBA客户调用DCOM服务器都比相当条件下所花时间长。

2 桥接的应用

2.1 应用背景

某公司希望改进其软件Creative System(CS)——一个针对门窗生产厂商的生产及销售管理系统。原系统建立在UNIX上，其中数据库公司买断了FairCom的数据库的源代码并作了修改，涉及价格的模块由于税务体制分为Federal tax、State tax、County tax、Local tax 四种不同的税率使得价格的计算相当复杂，而且原有的价格计算模块非常成熟、稳定。现在希望把原来的系统扩展为UNIX和Windows多平台多客户端的系统，即前台客户端充分利用Windows友好的界面和良好的可操作性，保留数据库和价格计算模块。

2.2 解决方案

构架一个分布式系统，利用COM和CORBA两种分布式技术，基于前面的测试结果，选择模型1方式2)，在前台客户端利用COM技术构建组件化的应用程序；后台保留图5中UNIX系统下的数据库，把价格计算模块封装到一个CORBA组件，再开发一个自定义的桥接组件沟通UNIX和前台客户端应用程序。考虑到数据

库的访问需要一个数据库访问的中间件,系统构架如图5所示,其中主要的组件有CORBA下的PriceComputer和DataServer,COM下的BridgeComponent。PriceComputer和BridgeComponent之间桥接的部分功能如下:

PriceComputer :

- 1) 访问数据库;
- 2) 计算价格;
- 3) 返回结果。

BridgeComponent :

- 1) 管理PriceComputer的生存;
- 2) 传递用户信息、订单号和货物ID;
- 3) 访问数据库;
- 4) 控制数据库的更新,设置数据库访问信号灯。

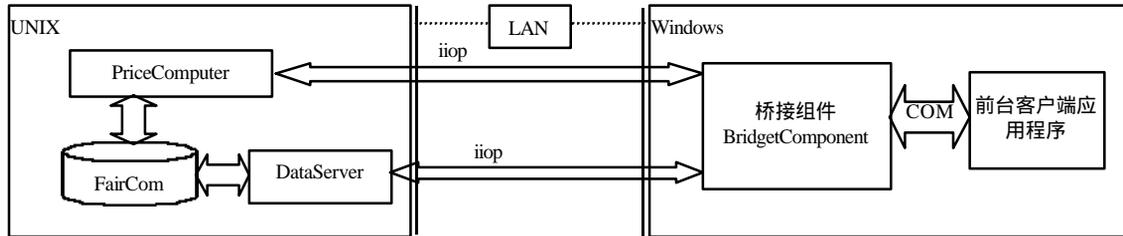


图5 系统构架

2.3 PriceComputer组件和BridgeComponent组件的IDL描述

服务器PriceComputer的 CORBA IDL如下:

```
module PriceComputer
{
    interface CorbaServer
    {
        void CustomerInformation(in long cut_uid, in long shp_uid);
        void QuoteInformation(in long qut_num, in string bom_cod);
        void GetCost(out string* p_cost);
        /*数据库访问的系列函数*/;
    };
    interface CorbaServerFactory
    {
        CorbaServer createCorbaServer();
    };
};
```

桥接组件BridgeComponent中COM服务器的IDL描述如下:

```
import "oidl.idl"
import "ocidl.idl"
[ object, uuid, dual, pointer_default(unique) ]
interface IBridgeComponent : Idispatch
{
    [id(1)] HRESULT Init([in] BSTR corbaFactoryIOR);
    [id(2)] HRESULT CustomerInformation([in] long cut_uid, [in] long shp_uid);
    [id(3)] HRESULT QuoteInformation([in] long qut_num, [in] string bom_cod);
    [id(4)] HRESULT GetCost([out, retval] string* p_cost);
    /*数据库访问的系列函数*/;
};
[ uuid(5102F13F-57F1-11D2-B54C-000000000000),
  version(1.0), helpstring("BridgeComponent") ]
library BridgeComponentLib
{
    importlib("stdole32.tlb");
    importlib("stdole2.tlb");
    [ uuid(4448B930-3EF0-11D2-B54B-000000000000) ]
```

28: 167-173

- [6] Lulli G, Albertazzi E, Nipoti R. The monte carlo binary collision approximation applied to the simulation of the ion implantation process in single crystal SiC: high dose effects[C]. Materials Science Forum, 2001
- [7] Posselt M, Schmidt B, Murthy C S. Modeling of damage accumulation during ion implantation into single-crystal silicon[J]. J. Electrochem. Soc., 1997, 144: 1 496-1 504
- [8] Zhao J H, Tone K, Weiner S R, *et al.* Evaluation of Ohmic contacts to *p*-type 6H-SiC created by C and Al coimplantation[J]. IEEE Electron Device Lett. 1997, 18: 375-377
- [9] Tone K, Zhao J H. A comparative study of C plus Al coimplantation and Al implantation in 4H- and 6H-SiC[J]. IEEE Trans. Electron Device, 1999, 46(3): 612-618
- [10] Zhang Yuming, Luo Jinsheng, Zhang Yimen. Au/NiCr Ohmic contacts to *n*-type 6H-silicon carbide[J]. Chinese Journal of Semiconductors, 1997, 18(9): 718-720

编辑 漆 蓉

上接第191页

```
coclass BridgeComponent
    {[default] interface IbridgeComponent };
};
```

2.4 效果

在测试中,通过客户端程序发送一个客户信息和定货号,通过桥接组件连接到CORBA服务器调用函数计算价格,然后CORBA服务器返回价格到客户,总共时间开销为5~6 ms,符合开发的要求。在实际开发中要解决一些问题,如控制CORBA服务器的生存期、数据库的同步、获取数据及类型的完整性等。

3 结束语

创建跨平台分布式系统已成为开发大型软件系统的一种重要选择。可以充分发挥UNIX等操作系统的稳定性,又可以利用Windows平台下图形化界面良好的可操作性,还可以避免对某一种平台的过分依赖。本文利用COM与CORBA的桥接,成功地把原有的系统扩展为跨平台的分布系统,充分利用UNIX和Windows的优点,改善了原系统的不足。

参 考 文 献

- [1] 潘爱民. COM原理与应用[M]. 北京:清华大学出版社, 1999
- [2] Don B. Essential COM[M]. 北京: 中国电力出版社, 2001
- [3] Jason P. COM与CORBA本质与应用[M]. 北京: 机械工业出版社, 2002
- [4] OMG/美国对象管理组织. CORBA Service[M]. 北京: 电子工业出版社, 2002
- [5] 朱其亮 邓 斌. CORBA原理及应用[M]. 北京: 北京邮电大学出版社, 2001

编辑 徐培红