

李智芳 吴晓云 (广州中山大学电子系 510275)

摘要: 传统的分布式系统是基于RPC(远程过程调用)标准的, 存在许多缺陷, 限制了分布式技术的进一步发展。CORBA规范弥补了传统的分布式技术的不足, 是分布式技术和面向对象技术结合的产物。本文在阐述CORBA系统的体系结构的基础上, 研究了基于CORBA系统的应用系统的实现方法。作为CORBA应用系统开发实例, 以Orbix 2000为平台, 开发了一个简单企业职工信息管理系统。

关键词: CORBA 面向对象 Orbix 2000 ODBC 信息管理系统

1 引言

随着计算机网络技术日益发展和普及, 人们越来越多的关注各计算机之间的互操作性, 达到软硬件及数据资源共享, 同时将应用程序放在不同的机器上, 以便在不同计算机上平衡负载。传统的分布式系统是基于RPC(远程过程调用)的, RPC是ISO/OSI组织制定的关于远程过程通信的标准, DCE(分布式计算环境)是基于RPC机制实现分布式计算的标准, 使用面向过程的分布式模型。RPC本身没有固有的对象模型。客户程序和服务器程序必须用RPC或DCE的实现语言来编制, 因为这些程序需要同RPC或DCE提供的库函数或接口界面捆绑在一起。这样就大大限制了用户的可选语言范围, 用户不得不去学习一种新的程序设计语言, 为的是能够使用RPC或DCE提供的服务, 这些缺点限制了RPC和DCE的进一步发展。

CORBA (Common Object Request Broker Architecture) 是OMG (Object Management Group) 制定的分布式对象技术标准, 它是分布式计算和面向对象技术结合的产物。CORBA的核心是ORB (对象请求代理)。实际上ORB是一种软件总线

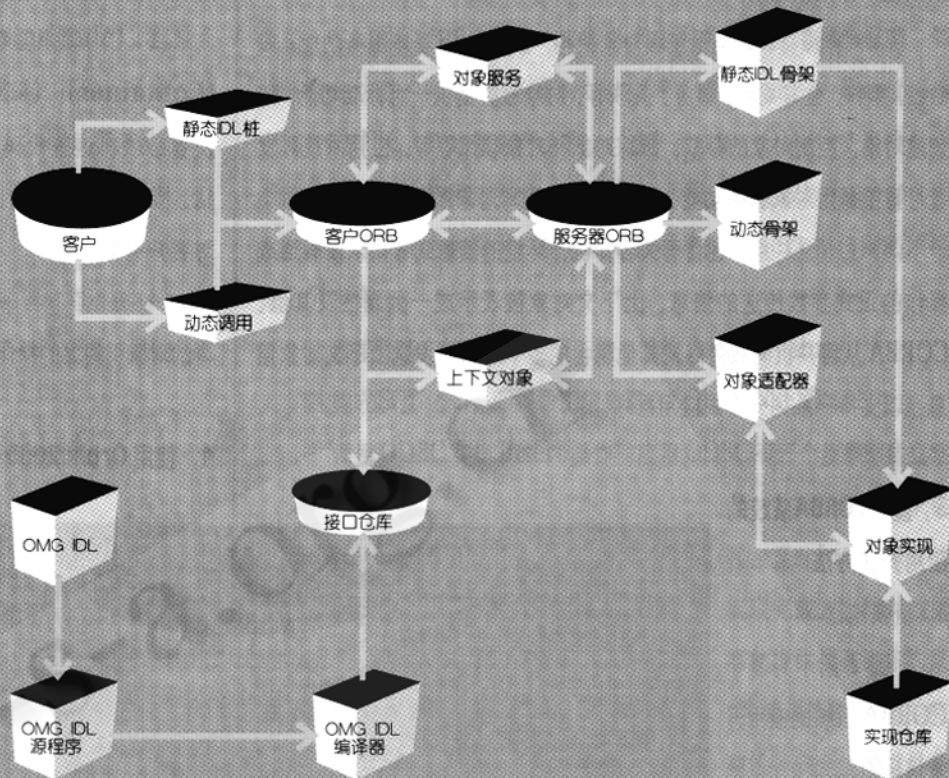


图1 CORBA系统体系结构

(software bus)。系统各个部件只要是按照CORBA规范开发都能够通过ORB连接到系统上, 实现即插即用。使用CORBA构造分布式系统用户不必考虑对象实现的语言, 存在的操作系统及在网络上的位置, 用户只需向ORB发出服务请求, ORB自动地在网络上查找相应的服务对象, 并返结果。

2 CORBA概述

如图1所示为CORBA系统体系结构图, 下面对各部分简要描述。

(1) 接口仓库。包含描述对象服务及如何调用对象服务的实时信息。客户从接口仓库中找到所需要的服务, 及如何调用服务。

(2) IDL桩和骨架。IDL编译器产生的目标代码分别在客户端和服务端产生映射, 客户端的映射称为桩(Stub), 服务器端的映射称为骨架。

(3) 动态调用和动态骨架。动态调用允许

对象请求动态创建和调用。客户使用动态调用发送请求给对象与客户使用桩发送请求具有相同的语义。一个请求由对象引用、操作和参数表组成。对于请求ORB使用封装的原则, 动态骨架允许动态的处理对象调用, 即不是通过某一操作的骨架来调用对象, 而是通过提供操作名和参数的接口来达到对象实现, 与客户的动态调用相似。

(4) 对象实现仓库。实现库包含一些使ORB能定位及激活对象的实现的信息, CORBA使用实现库来联系对象引用和实现, 正如它使用接口仓库来联系对象引用和接口一样。BOA(对象适配器)期望描述实现的信息存放在实现仓库里。

(5) 对象适配器(BOA)。提供执行对象可以访问ORB服务的基本接口, 其作用是产生及解释对象引用、安全交互、登记和执行等。

(6) ORB(对象请求代理)。ORB隐藏了服务器对象和客户对象的所有的通信细节, 实际上

ORB是一个对象总线。

(7) 对象服务。定义系统级对象架构, 扩展ORB基本服务功能, 如命名、事件、生命周期、事务、关联等。对象服务提供一组标准功能, 如创建对象、控制对象的存取、跟踪对象和对象参照等等。应用系统开发者可直接调用对象服务功能, 而无需自己编写对象服务程序。

(8) 上下文对象。包含有关客户机、环境的信息或者请求没有当成操作的正式参数传送的信息。上下文对象被表示成一系列属性及其值。CORBA从客户机向服务器或其他隐含的多个服务器发送上下文对象信息, 以支持一系列分布式应用程序。

3 基于CORBA系统的应用开发方法

3.1 开发流程

图2所示为基于CORBA系统的应用开发流程图。

3.2 开发步骤

(1) 定义IDL接口。CORBA IDL用于描述企业应用中的对象接口。对象接口为对象潜在的客户描述对象的属性、操作及签名。IDL定义的对象能以IDL可映射的任何语言实现如C++、JAVA及COBOL。通过一个公共语言的包装对象接口, IDL使得两对象之间能交换作用, 而与对

象的实际实现无关。因此用IDL编写对象接口是使二个不同语言及平台之间交互操作, 这是CORBA的中心目标。CORBA定义了从IDL到多种语言的标准映射, 包括C++、JAVA及SMALLTALK。每个IDL映射描述一个IDL接口如何与某一语言的实现相对应。IDL编译器使用这些映射将IDL定义转换成遵从某一语言语法的定义。

定义IDL接口就是标识应用需要的对象, 以IDL定义他们的公共接口, 来表示对象需要的一些功能等。

(2) 编译IDL定义。编译器产生C++头文件和源文件, 这些文件

是实现客户程序和服务器程序所需要的。

(3) 开发服务器程序。服务器作为各种CORBA对象的容器(Container), 每个CORBA对象都支持一个IDL接口。服务器开发者必须将这些代码加入, 以为每个CORBA对象类型提供商业逻辑。服务器通过将输出对象引用到某个可知的位置, 使CORBA对象对客户有效。

(4) 开发客户程序。客户使用IDL编译器产生的映射调用从服务器获得的对象引用上的操作。

4 使用Orbix 2000研制基于CORBA的职工信息管理系统

4.1 Orbix 2000简介

Orbix 2000是IONA公司研制的基于库的CORBA规范实现, 支持Windows、OS/2、各种UNIX等20余种操作平台。它提供了C++语言映射器和C++代码产生器。为了能够在互操作方面做进一步的工作, IONA已向微软公司购买了COM许可证, 以期能够提供实现COM/CORBA(DCOM/CORBA)互操作完美的解决方案。IONA的系列产品包括OrbixNames(名录服务)、OrbixTrader

(交易服务)、OrbixSecurity(安全服务)、OrbixOTS与OrbixOTM(事务服务)、OrbixTalk(事件服务)和Orbix Notification(通知服务), 它们实现了CORBA对象服务规范中规定的许多重要的对象服务。此外, 还有OrbixSSL(实现SSL上的IIOP加密传输)、OrbixCOMet(COM/CORBA互操作)、Orbix Database Adapter Framework(Orbix的数据库接口)等。该公司还推出了Orbix的Java版OrbixWeb。

该系统提供多种方便的开发工具如提供了命名行式的代码自动产

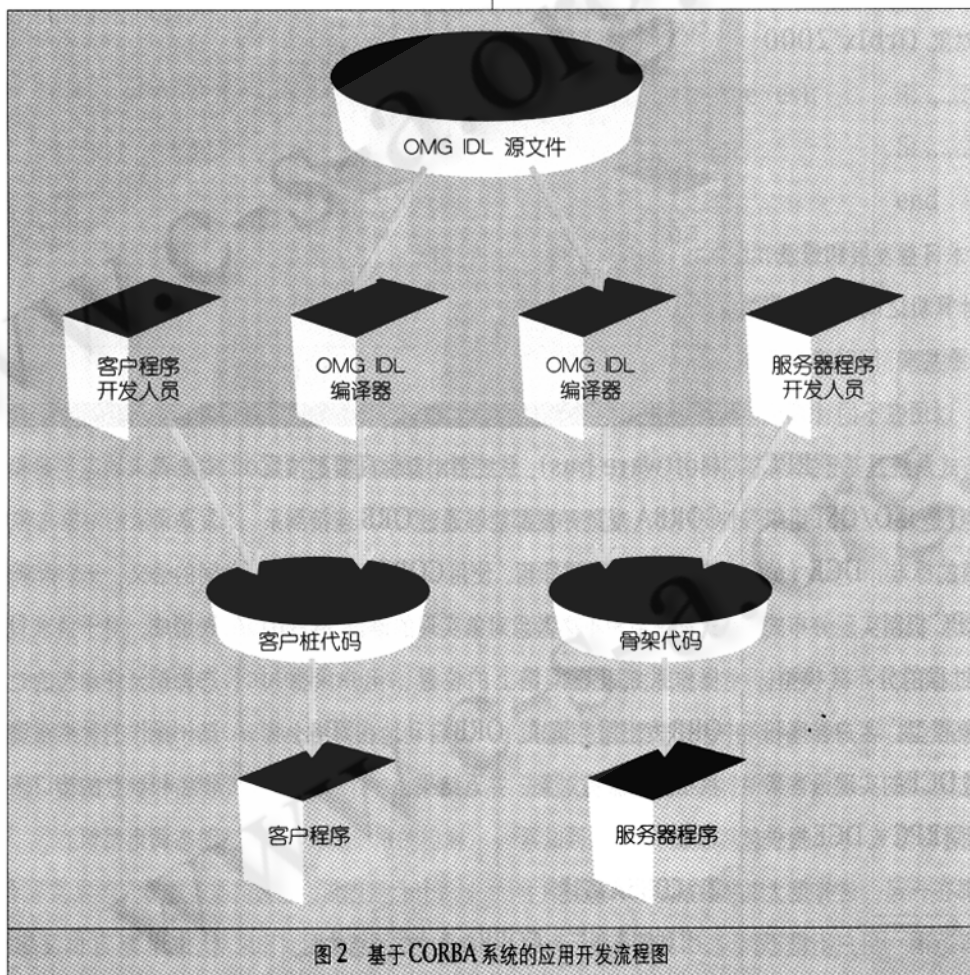


图2 基于CORBA系统的应用开发流程图

生器。能够与VC整合,使用VC的开发平台开发应用程序。这些工具都为用户产生应用程序的起始点代码(Start point code)。起始点代码其实相当于应用程序框架,有了应用程序框架,开发人员通过向框架中添加代码,定制所需的应用系统。这样大大的缩减了开发时间。

4.2 职工信息管理系统的设计与实现

4.2.1 系统需求

系统功能如下:

(1) 查询浏览服务:模糊查询、上一个记录、下一个记录、跳到头记录、跳到尾记录。

(2) 改变数据的服务:插入记录、修改记录和删除记录。

(3) 统计打印:按各种条件进行统计并打印报表。

数据库包括以下信息手段:工资号、姓名、部门、办公电话、移动电话、性别、生出年份、出生月份、家庭电话、身份证号。

4.2.2 系统结构

如图3所示。

4.2.3 系统实现

(1) 开发客户端应用程序。在VC开发环境下,选择建立新项目属性页中的IONA Orbix 2000Client/

Server Wizard项目类型,在后面的选项中Client即可产生客户端应用程序框架。然后修改各个调用的实现。

由于篇幅有限,这里仅以Insert服务为例说明。

要插入一个新的职工记录,按下列步骤进行。

① 定义一个EMPDATA结构类型,用于传递职工信息数据即EMPDATA Record-value;

② 将职工信息数据赋于Record-value的各个信息字段。

③ 调用Insert操作。Record-value作为操作参数传送服务器的真正的Insert的实现,对数据库实施该操作。即Insert(Record-value);

(2) 开发服务器端应用程序。在VC开发环境下,选择建立新项

目属性页中的IONA Orbix 2000 Client/Server Wizard项目类型,在后面的选项中Server即可产生服务器端应用程序框架。然后修改各功能的实现。服务器端对数据库实施各种操作的是通过调用中间件ODBC(开放数据库互联)[5]来完成的。ODBC使用的是标准的SQL语言[4]实施操作。由于篇幅有限,这里仅以Insert(插入)方法的实现为例说明。要完成对数据库的Insert操作则必须经由下列步骤。

① 将接受客户传送过来的职工信息结构Record-value。

② 构成SQL语句。即将职工信息溶入SQL的INSERT语句中。

③ 调用ODBC实施Insert操作。应用ODBC有三个步骤即建立链接、数据存取和结束链接。

至此Insert操作的实现完成。当然对数据库的操作是有许多约束的,错误处理,数据的完整性,安全性等都是非常重要的,而且必须考虑的。由于篇幅有限在此略过。

5 结束语

CORBA规范已成为分布计算的最新标准。越来越多的厂商正在按照CORBA标准开发具有自己知识产权的软件产品。这样,不同

厂商生产的面向对象的软件产品能够跨越网络和操作系统,从而推动了CORBA技术的发展。因此,以CORBA系统为支撑的应用系统,也必然成为未来发展的主流。

参考文献

1 OMG,The Common Object Request Broker: Architecture and Specification,Revision 2.0,July 1995 Updated July 1996.

2 IONA INC.,Orbix Programmer's Guide C++ Edition <http://www.iona.com>.

3 IONA INC.,Orbix Programmer's Reference C++ Edition <http://www.iona.com>. ■

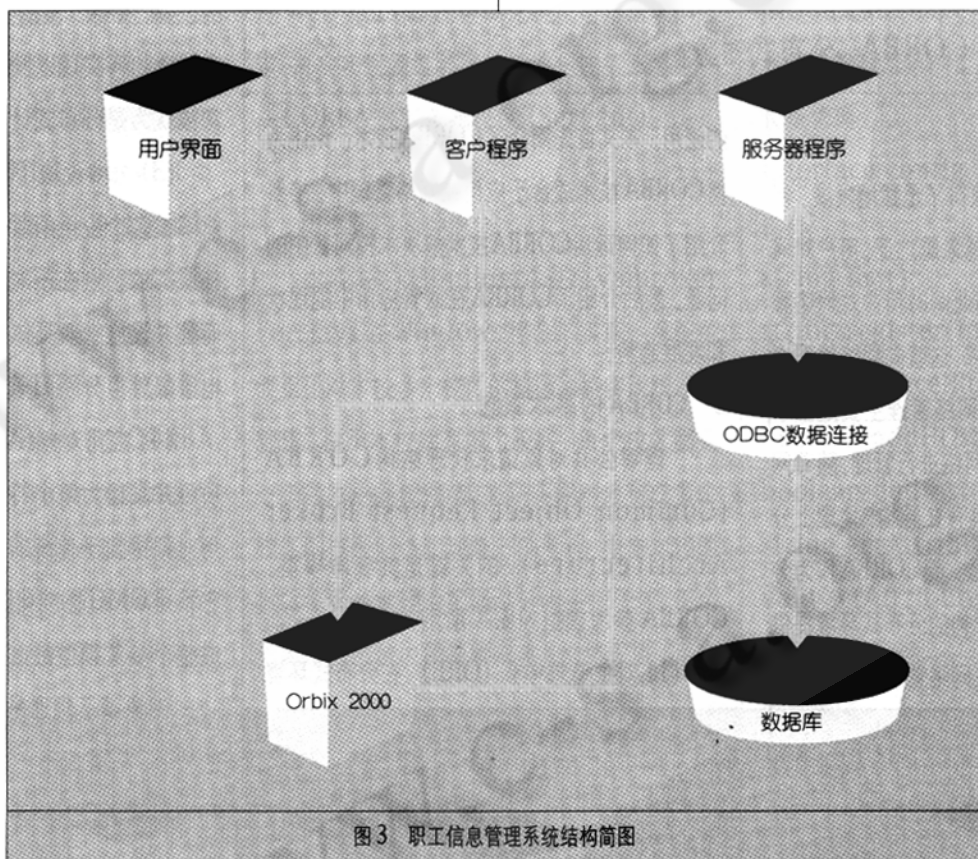


图3 职工信息管理系统结构简图