

文章编号:1000-582X(2006)02-0012-04

基于 CORBA 和 Java 的协同设计通信平台*

孙冬梅¹, 陈小安¹, 罗天洪¹, 方义平²

(1. 重庆大学机械传动国家重点实验室, 重庆 400030; 2. 荆州江汉机械研究所, 荆州 434000)

摘要:通过分析目前协同设计的现状以及存在的问题,提出了一种基于 CORBA 原理与 Java 分布式编程技术的通信体系结构. 利用 CORBA 和 Java 技术的互补性,将 CORBA 和 Java 技术集成,进行 WWW 分布式应用开发,为分布式协同设计提供了一种更有效的设计平台,实现了安全高效的通信.

关键词:CORBA; Java; 通信平台; 协同设计

中图分类号:TP393

文献标识码:A

随着市场竞争的日趋激烈以及全球化的快速发展,越来越要求企业以最快的速度、提供最好的服务响应用户的需求,这就使得传统的设计制造方法无法适应市场快速变化的需要,工作组成员之间的协同工作就显得越来越重要了. 目前很多企业实现了生产制造资源的全球配置,企业的管理、设计制造、市场营销人员分布在全国各个角落,因此,现代产品的协同设计也逐渐由设计人员面对面的协同设计发展到跨地区甚至跨国界的异地协同设计,这就要求有一种技术来为异地分布的协同人员提供协同支持. 随着计算机网络技术和 CSCW(Computer Supported Cooperative Work)技术的快速发展,出现的计算机网络支持的协同工作(Network Supported Cooperative Work, NSCW)技术为异地协同工作的研究与应用提供了有利的技术支持^[1-2],然而现代设计中分布式用户界面通常处于异构的网络环境之中,在这种环境下,系统的部件可能处于不同的物理平台上,运行于不同的操作系统中,部件可能用不同的编程语言编写,相互通信时采用不同的通信协议(TCP/IP等). 因此,异构环境中协同用户之间安全高效的通信就成为一个急需解决的问题.

1 基于 CORBA 和 Java 的通信体系结构

传统的协同设计人员之间的通信主要是以 HTML 技术为代表、基于客户/服务器两层模型的 Web 体系. 这种结构模式难以维护和升级,难以扩展到大企业广

域网或国际互联网,大大阻碍了系统的发展,存在着由 HTTP 协议所造成的网络瓶颈和事务处理能力不足等缺陷. 随着经济全球化的快速发展,需要协同完成的设计项目越来越大,协同设计人员分布越来越广,同时庞大而复杂的网络变得更大而且更复杂,因为技术变得更复杂了. 在这种环境下,人们对 Web 的应用剧增,对 WWW 技术也提出了越来越高的要求. 于是出现了 Web 中间件技术的 3 层 Web 体系. 然而目前用于协同设计通信的很多中间件技术其内部接口和实现并不透明,缺乏技术上的中立性和稳定性,不利于分布式应用的开发和维护. 针对上述问题,作者进行了基于 CORBA 和 Java 的协同设计通信平台的探索和研究^[3].

1.1 CORBA 技术的特点

CORBA 是一种面向对象的中间件技术,从本质上讲,它是一种结构,一种通用的对象请求代理体系结构. 这种体系结构为开发大型的分布式异构应用软件系统提供了独立于硬件平台、独立于编程语言、独立于操作系统和独立于网络传送协议的平台^[4]. 它定义了对象之间通过对象请求代理(Object Request Broker, ORB)透明的发送请求以及接收响应的机制,保证了在异构系统中对象之间的互操作性. 具体化体现在用接口定义语言(Interface Definition Language, IDL)进行对象建模,以及采用 IIOP(Internet Inter-ORB Protocol),即因特网 ORB 互通协议进行协同用户和服务器之间的通信. CORBA 将面向对象技术与分布式计算

* 收稿日期:2005-09-23

基金项目:高等学校博士点专项科研基金资助项目(20040611006); 欧盟 Asia link 项目(ASV137-301/98/679-023); 亚欧 IT&C 项目(A31/B7301/3150299/7155)

作者简介:孙冬梅(1975-),女,湖北黄冈人,重庆大学硕士研究生,主要从事网络化协同设计方面的研究.

相结合,提供了软总线机制,CORBA 技术的核心作用就是为异构环境下的分布式系统中分布在不同节点上的对象提供一个对象总线以及相应的总线服务,分布式系统中各分布对象只要按 CORBA 规定的接口方法接上软总线,便可以安全方便地实现对象间的互操作性.

1.2 Java 技术的特点

Java 是一种比较新的面向对象程序设计语言,具有安全、多线程、可移植性和跨平台性等特点,其设计目标是支持分布式异构环境下的应用,Java 程序可以在任何地方运行.而且,它还提供了数据库访问接口,即 JDBC,JDBC 提供了一个标准的应用程序访问接口(API),它为数据库开发提供了标准的应用程序编程接口.此外,Java Applet 大大拓展了 Java 与 WEB 技术的结合,Java 小应用程序可被下载到使用 Java 的浏览器上,并能够同其它与 CORBA 标准兼容的对象和服务器进行通信交互.

1.3 CORBA 和 Java 技术相互补充

CORBA 和 Java 都是 WWW 的重要技术,也是当今分布式网络技术的研究热点. CORBA,Java 各有其优点,将它们结合起来,就可以让它们实现优势互补.为了解决协同设计过程中异构网络环境下构件间的互操作问题—即来源不同的构件之间能相互通信、相互提供服务,共同完成一项复杂的任务,采用了中间件技术.在众多解决方案中,首先选择了 CORBA,因为它具有可扩展性和业界广泛的支持.目前,国内 CORBA 技术的理论研究虽然已经有很长时间了,但集成 JAVA/CORBA 到 Web 中应用的研究却刚刚起步,基于 Java/CORBA 的 Web 协同设计系统更是少见.

CORBA 与 Java 技术相结合使用,一方面,CORBA 用分布式对象基础结构扩展了 Java 语言,使得 Java 应用程序可以通过远程方法调用,进行跨地址空间的通信;另一方面,Java 语言使 CORBA 能在 WEB 环境下运行,并且扩充了 CORBA 的代理结构^[5-6].

1.4 通信体系结构

基于 CORBA/Java 的分布式协同设计通信方案,通过将 CORBA 和 Java 技术结合,实现异构环境下协同设计人员之间的通信,其通信体系结构如图 1 所示.从图 1 中可以看到 CORBA 通过对象引用实现相互间的交流,通过 ORB 和 IDL 实现了协同用户方和服务器之间的互操作.

实现过程如下:

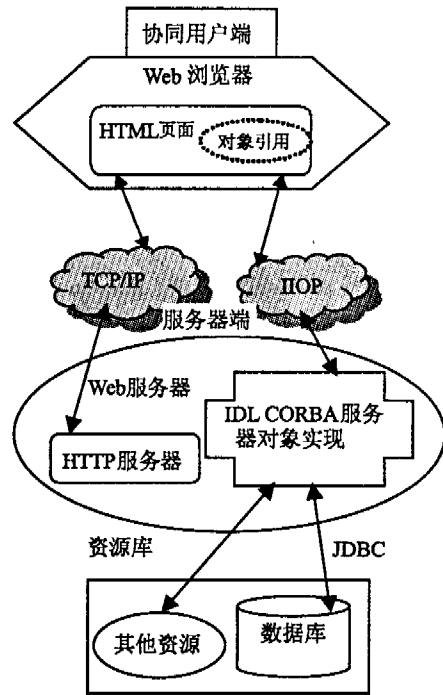


图 1 通信体系结构

1) 协同用户通过 Web 网进入服务器项目设计主页并下载静态 HTML 文本和 Java Applet 以及 ORB 运行组件,然后协同用户才能使用 IIOP 协议与对象实现方通讯.

2) 浏览器运行 Applet,即 Applet 初始化协同用户端 ORB,然后通过 IIOP 访问协同网络内的服务器端对象,利用 ORB 寻找到 CORBA 服务器对象,获得该服务对象的对象引用.小应用程序 Applet 根据接口 IDL 产生的客户存根,就可以使用对象引用发送调用请求了.

3) 协同用户通过本地浏览器向服务器发出请求,请求的网页内包含有起桥梁作用的 Applet 小应用程序.该 Applet 由 CORBA 协同用户程序的 JAVA 语言实现,并将它与 CORBA 的 ORB 捆绑.

4) ORB 将请求送给对象适配器,然后对象适配器执行调用 IDL 静态框架或动态框架接口中的动态实现例程来调用对象实现的操作,如通过 JDBC 调用数据库.

5) 在对象实现的操作执行完毕后,ORB 将负责把结果或异常反馈给协同用户,即通过 IIOP 协议将请求结果返回给浏览器^[7].

2 基于 CORBA 和 Java 的协同设计应用实例

对于一个具体的设计系统,其原型系统的模块设置和外部显示由服务器/客户程序和 IDL 接口实现,在建立该原型系统时采用了 JAVA 语言.建模之前在服

务器和用户端分别安装好 JAVA 软件并设置 CORBA 服务器,协同设计过程通过权限设置和用户密码验证来保证安全性.

原型系统的创建过程如下^[8]:首先针对服务器和客户端在协同设计中需要实现的不同功能建立各自的对象模型,然后根据对象模型用 JAVA 语言表达出来,实现用户程序和服务器程序,最后编写 IDL 接口实现连接和互操作.

协同设计中功能模块的建立以及服务器和用户的交换示例.

2.1 服务器端的建立和实现

服务器的功能是将异地分布的协同用户联系在一起,统一交换,统一管理,本例中如图 2 所示服务器端需建立 7 个功能模块—协同用户管理模块、多数据库系统管理模块、数据格式转换系统模块、消息管理模块、共享文件列表模块、装配图显示模块、专家审核系统模块.



图 2 服务器系统

建模完成之后,根据这些功能模块用 Java 语言编写 Design Server.java 服务器端程序并保存在虚拟目录下,然后设置 CORBA 服务器,服务器端 ORB 初始化如下:

```
java.util.Properties orbpro = null;
orbpro.setProperty("ORBInitialHost", "202.202.10.47");
```

全部初始化完之后,等待协同用户端的请求和调用,并处理相关操作.如图 2 为服务器和协同用户之间的交互,服务器在接受油泵设计用户的参数修改数据.

2.2 客户端的建立和具体实现

首先建立客户端对象模型,确定其功能模块,如图 3 所示协同用户系统包括 4 个功能模块—服务器装配图显示模块、共享文件浏览并下载功能模块、本地图形

共享和上传功能模块、信息交流即消息对话功能模块.

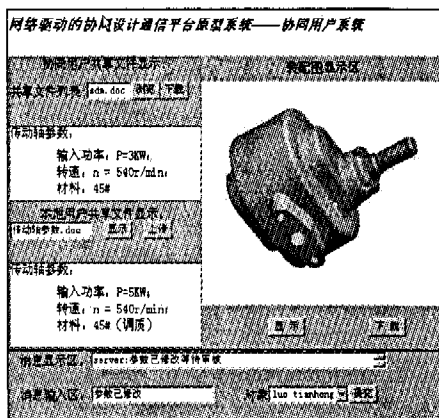


图 3 协同用户系统

然后根据这些功能模块编写用户程序, CORBA 规范对具体实现的编程语言没有什么特别的限制,但考虑到 CORBA 和 JAVA 语言结合的优越性,采用 JAVA 语言.完成程序编写后再编写 IDL 接口.在编写接口时,对接口中的数据类型进行必要的描述之后,以关键字 interface 开始对接口内容进行定义,并将对象需要提供给外界的属性和操作添加到接口中来.

design.idl 文件

```
// design: the module of collaborative design
module designapp {
.....
interface Customer {
.....};
interface Server {
readonly attribute string name;
readonly attribute string passwd;
.....};
interface Design {
.....
};};
```

完成上述工作后就可以对系统进行编译、调试、运行了.原型系统运行后的服务器和协同用户界面如图 2 和 3 所示.协同用户通过 Web 网进入服务器主页,然后输入用户名和密码,通过验证后进入如图 2 所示协同用户系统.

图 3 为设计用户和服务器的协同过程,从图上可看到协同用户可以看到服务器上最新显示的该协同项目的装配图,此时协同用户正浏览和下载服务器上的共享文件,了解协同设计项目的进度、参数、设计更改

和人员调整等信息;同时将本地更改的设计参数上传到服务器供协同网络内其他设计人员查看,并通过信息交流提示服务器等待其审核更新,显示最新版本。

与此同时如图 3 所示服务器接收到了该用户的更改数据并进行相应的处理。

从上例可以看出,协同设计的整个设计和实现过程由两条主线将客户端和服务端紧密地联系在一起,如图 1 所示,一条是以 TCP/IP 为纽带的外线,另一条是以 IIOP 为纽带的内线,它们相辅相成,使分布式协同设计透明而有条不紊地进行着。

3 结 论

在协同设计中实现异构环境下资源共享,安全而高效地通信是很难回避的技术难题。由于异构的复杂性,很少有软件能较好地加以解决。而从以上研究分析中可以看到,将 CORBA 技术和 Java 技术结合应用于协同设计通信,实现了异构环境中协同用户之间安全高效的通信。

参考文献:

- [1] 曹健,张有良,黄双喜. 计算机支持下的协同设计及其环境研究[J]. 计算机工程与应用,1999,(5):39-41.
- [2] 田凌,童秉枢. 网络化产品协同设计的理论与实践[J]. 工程与应用,2002,(5):3-6.
- [3] 李志均. 基于 WWW 的 Java/CORBA 中间件技术及应用研究[D]. 广东 广州:广东工业大学,2000.
- [4] 朱其亮,郑斌. CORBA 原理及应用[M]. 北京:北京邮电大学出版社,2001.
- [5] 段江娇,薛永生. 基于 CORBA/Java 技术的分布式异构数据库应用系统[J]. 计算机工程与应用,2002,(2):160-162.
- [6] KONG X A, ZHAN J F. A Parallel Distributed FEM Computing Circumstance Based on CORBA and Java Techniques[J]. Journal of Southwest Jiaotong University, 2000, 8(2):136-143.
- [7] 罗天洪,陈小安,林利红,等. 网络协同设计动态数据交换系统[J]. 重庆大学学报(自然科学版),2003,26(12):1-5.
- [8] 朱庆华,程涛. CORBA 规范在分布式制造系统中的应用[J]. 中国机械工程,2000,(3):307-309.

Communication Platform Based CORBA and Java in Collaborative Design

SUN Dong-mei¹, CHEN Xiao-an¹, LUO Tian-hong¹, FANG Yi-ping²

(1. State Key Laboratory of Mechanical Transmission, Chongqing University, Chongqing 400030, China;

2. Jiangnan Machinery Research Institute, Jingzhou 434000, China)

Abstract: The present status and problem for collaborative design are analyzed. A communication system structure based CORBA principle and the Java distributed programming technology is developed. The capability of CORBA and Java supplying is compensated each other in this structure system. The integration of CORBA and Java is proved to be a more effective designing platform for distributed collaborative design in the area developing WWW-based applications. Highly efficient safe communication is realized for collaborative design.

Key words: CORBA; Java; communication platform; collaborative design

(编辑 成孝义)