

JAVA 与 CORBA 结合实现分布式异构系统中间件

朱 斌¹, 刘子欣¹, 朱海云²

(11 华南理工大学, 广东广州 510640; 21 广东省电信公司科学技术研究所, 广东广州 510630)

摘 要: 分布式异构系统的中间件是当前研究的热门课题之一, 也是电力系统中人们极为关注的问题. 本文把 JAVA 和 CORBA 各自的优势结合起来, 设计了用于国内某电力调度中心分布式异构系统的中间件. 本方案以 JAVA 为外壳, 以 CORBA 为内核, 在国内电力系统中首次成功地实现了这一设计思想. 实验证明, 上述设计思想是正确的, 且该中间件已可实际应用于电力调度中心异构系统的数据交换. 本文重点讨论了该中间件设计的几个关键技术问题.

关键词: 分布式异构系统; 中间件; I/O 性能; 连接池

中图分类号: TP311152 **文献标识码:** A **文章编号:** 03722112 (2003) 091313204

Integrate JAVA with CORBA to Implement the Middleware of the Distributed Systems of Heterogeneity

ZHU Bin¹, LIU Zi2xin¹, ZHU Hai2yun²

(11 South China University of Technology, Guangzhou, Guangdong 510640, China;

21 Guangdong Telecom Academy of Science and Technology, Guangzhou, Guangdong 510630, China)

Abstract: The middleware for the heterogeneous distributed systems is one of the hotspots for research. And it is also fearfully focused in the electric power field. The article concludes that it is valid to integrate the respective advantages of JAVA and CORBA to implement the middleware, which is applied to the heterogeneous distributed systems in the electric power control center. That is the first time to be attempted in national electric power field. The crust of the middleware is JAVA and the kernel is CORBA. This idea is proved to be correct in practice. And the middleware for the data exchanging can be suitably applied to the electric power control center. At last, the keys of the project are discussed.

Key words: distributed systems of heterogeneity; middleware; the capability of Input/Output; connection pool

1 引言

现代计算机技术迅速发展, 特别是 Internet 的出现, 使计算机的应用范围更为广阔, 许多应用程序需在网络环境的异构平台上运行. 这一切都对新一代的软件开发提出了新的需求, 如何把分布式异构系统集成起来, 并开发新的应用是一个非常现实而困难的问题. 为解决分布异构问题, 人们提出了中间件(middleware)的概念. 中间件是位于平台(硬件和操作系统)和应用之间的通用服务, 这些服务具有标准的程序接口和协议, 针对不同的操作系统和硬件平台, 它们可以有符合接口和协议规范的多种实现. 在数据交换时, 处于分布异构环境的用户只需向网管数据访问平台指明他需要什么数据, 便可由平台通过中间件取得数据, 并发送给用户^[1]. 在现代化的大型企业中中间件的作用尤为重要, 特别是在我国的电力系统中, 把 JAVA 和 CORBA 结合实现分布式异构系统的中间件尚无成功前例. 高效高性能的分布式异构系统中间件将为优化电力调度, 顺利完成“十五”计划/西电东送的任务提供强有力的

保障, 所以本研究具有一定的实际意义.

2 电力调度中心的体系结构与工作方式

某电力系统由几个跨省的大电网组成, 其电力调度中心现有 EMS 系统(Electronic Management System), 以及独立运行的子系统: 功角监测系统、水调自动化系统. 子系统的设计采用了不同的方案, 水调系统用的是 WINNT+ MS SQL, 而功角系统则用 WINNT+ 数据文件. 系统彼此间的差异非常大, 实现方法语言也各不相同. 随着电网的发展和电力市场的拓展, 再加上系统本身的缺陷和设备老化, 原有的 EMS 系统已经越来越难以适应发展要求. 主要存在的问题是: 各子系统之间互相独立, 无法实现信息共享; 原系统为集中式结构, 可靠性和灵活性不如分布式系统; 原系统为非面向对象系统, 不适合将来系统的扩展. 为此, 必须新建一个功能强大的 EMS 系统(以后简称新 EMS 系统). 新 EMS 系统是一个分布式的面向对象的系统, 必须保证电网的安全运行; 实现有关自动化的系统集成; 具有良好的可靠性、可伸缩性和可维护性. 而更为重要的

是必须把新 EMS 系统与原有的两个子系统整合起来,与之进行数据交换,并且做到有良好的可扩充性(结构体系如图 1)。这势必要开发一个高性能并且安全可靠的中间件,为分布式异构系统之间的数据交换提供支持。这种中间件在我国的电力系统中尚无成功前例,它既要满足新 EMS 系统的需要,同时又要保证原有子系统可以正常使用,还要兼容以后可能扩展的新的子系统。

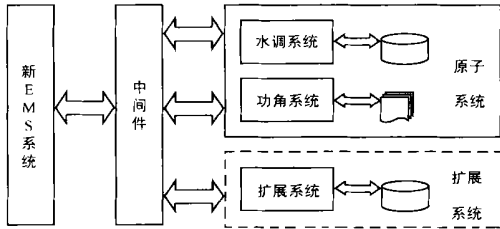


图 1 电力调度中心体系结构图

3 中间件方案论证: JAVA 和 CORBA 结合

在当今中间件技术中 CORBA(Common Object Request Broker Architecture), DCOM(Distributed Component Object Model) 和 EJB(Enterprise Java Bean) 这三种用得最为广泛。我们对这三种典型的技术进行分析与比较,认为 DCOM 技术不宜采用,因为它有两个重大缺点:(1) 它由单一开发者(微软)定义并控制,这大大限制了 DCOM 使用者的选择范围(比方说开发工具和风格)。(2) DCOM 缺乏众多的平台支持,这极大程度地制约了代码的可重用性和 DCOM 应用的可扩展性。我们认为 JAVA 和 CORBA 各有千秋。JAVA 提供了一个概念清晰、结构紧凑的分布计算模型和构件互操作的方法,为构件应用开发提供了相当的灵活性。但由于它还处于发展时期,因此其形态很难界定。CORBA 是一种集成技术,而不是编程技术。它提供了对各种功能模块进行构件化处理并将它们捆绑在一起的粘合剂。JAVA 和 CORBA 在很大的程度上可以看作是互补的。CORBA 使 JAVA 应用可以跨越网络、语言以及操作系统,并为 JAVA 提供了一组分布服务,如分布式自我观察、动态发现、事务、关系、安全和命名等。JAVA 不仅是一种语言,还是一个动态代码系统,对运动对象来说是一个可移植的虚拟机(JVM);为开发、管理、发布 Client/Server 应用提供了更简单的方式。只要将应用放在一个 Web 服务器上就可将这一应用发布给成千上万的用户,而不必关心它的安装和升级^[3]。JAVA 还非常适合服务器的开发,可以动态地将服务代码移向最需要它们的地方。JAVA 将会使 CORBA 对象能够运行在从主机、网络计算机到蜂窝电话等可编程的各种机器上,并简化了大型 CORBA 系统的代码发布。对客户和服务对象来说, JAVA 是很理想的编程语言, JAVA 内置的多线程、垃圾收集和错误处理使编写健壮的网络对象变得很容易。这两种对象模型可以很好地相互补充, CORBA 处理网络的透明性, JAVA 处理实现的透明性, CORBA 为 JAVA 可移植应用环境提供了一个分布式的结构^[2]。由此看来 CORBA/JAVA 技术紧密结合将成为 Object Web 技术的主要形态。因此本文的结论:电力调度中心分布式

异构系统中间件的开发采用 JAVA 与 CORBA 结合的技术,使用内置了 CORBA2.1.3 标准的 JAVA 发展工具组(Java Development Kit)进行开发。该工具组已包含了一组纯粹由 JAVA 所发展的 CORBA ORB(Object Request Broker)。它已经把用 IDL 编写的 CORBA 常用构件映射到 JAVA 平台上并写成类的形式,只要用 JAVA 语言把这些常用的接口类有机的组合起来,就可以初步地实现 CORBA 与 JAVA 技术的结合,从而解决中间件的实现问题。(要说明的是这种结合技术只是 CORBA 与 JAVA 技术的一种初步的结合。将来的 Java RMI 将建立在 IIOP 之上,那时 CORBA 不只是对象请求代理 ORB,也是一个非常完整的分布式对象平台,这是将来两种技术在更深层次上结合的新趋势^[4]。)

采用两种技术结合的方案,以 JAVA 为外壳,以 CORBA 为内核,集中了两种技术的优点,是现今中间件软件发展的新趋势。对电力系统中间件而言,更是没有前例的大胆尝试。综合来讲,该技术具有五大优点:(1) 平台无关性。CORBA 可以创建平台无关的分布式服务,用户不必将自己捆绑在某一种硬件或操作系统上,可以根据自身的情况选择合适的硬件、操作系统、数据库;(2) 可移植性。因为 J2EE 是一种组件技术,已完成的组件能被方便地移植到任何其他地方;(3) J2EE 作为中间件,提供了强大的功能,使开发人员只需要关注商业逻辑,因而能加快开发速度,提高系统的运动效率和稳定性;(4) 保护用户投资。当用户更换平台时,由于基于 J2EE 的系统能方便地移植到其他平台上,而无须重新开发,因此能有效地保护用户的投资;(5) 广泛的支持。J2EE 技术规范得到了从国际性大公司到自由软件开发者的广泛支持, SUN, IBM, ORACLE, NETSCAPE, BEA 等纷纷推出基于 J2EE 的软件产品。

4 中间件的实现方案与成功实现的关键

4.1 中间件的实现方案

整个中间件分为两大模块, Server 模块和 Client 模块。

Client 模块在新 EMS 系统上运行,主要负责与水调系统,功角系统等服务器通讯,并向对应的服务器发送相关的数据查询命令,并对服务器传送过来的数据进行处理等。主要有 3 个方法,其中 Receive Socket() 负责监听指定端口,接收 Client 模块发送的命令; Send SQL() 负责执行相关命令,查询数据库并向 Client 模块发送相关记录; Send File() 负责执行相关命令,查询数据文件并向 Client 模块发送相关数据文件。

Server 模块在水调系统和功角系统上运行,主要负责与新 EMS 系统通讯,接收新 EMS 系统发送的数据操作命令,并把相关数据传送给新 EMS 系统。主要有 3 个方法,其中 Send Socket() 负责向 Server 模块发送命令; Receive SQL() 负责接收 Server 模块发送的数据库记录; Receive File() 负责接收 Server 模块发送的数据文件。

Client 模块和 Server 模块通过 TCP/IP 协议相互激发,使用 Data Stream 为接口实现数据交换。Server 模块使用 JDBC 连接数据库,实现相应的操作;使用 File Reader 接口实现对文件的操作。整个系统主要模块和接口如图 2。

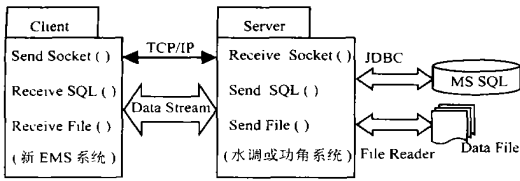


图 2 系统模块与接口图

41.2 成功实现的关键技术

图 2 中的系统模块用 JAVA 实现起来并不困难,但在开发过程中,我们发现要成功地实现整个方案有几个关键地方需要特别注意:

41.2.1 优化 I/O 性能,解决速度问题

由于众所周知的原因 JAVA 程序总是比 C++ 程序慢,特别是频繁的 I/O 操作,为了保证中间件的高效率和高性能,必须优化 I/O 性能,解决速度问题.实验结果表明,经过精心优化后 JAVA 程序可以达到 C++ 中类似程序的效率.要提高 Java 的 I/O 性能,使用缓冲是最为有效的方式之一.可以使用标准的 BufferedReader 和 BufferedStream 类,或者使用块读取方法,一次读取一大块数据.前者快速简单,能够有效地改进性能,且只需少量地增加代码,出错的机会比较少.后者也即自己编写代码,复杂性略有提高))当然也谈不上困难,但它能够获得更好的效果.我们设计程序时,就是用 BufferedReader 封装 File Reader 作为源文件与程序的 I/O 接口,用 BufferedStream 封装 DataStream 作为数据 I/O 的接口,从而优化 JAVA 的 I/O 性能,达到 C++ 中类似程序的效率,解决速度问题.现以分布异构环境中传送数据文件为例,下面是 3 种方案的效率的比较:一种是没有优化 I/O 的 JAVA 程序实现的中间件,另一种是优化了 I/O 的 JAVA 程序实现的中间件,还有一种是用 IDL 对 CORBA 构件进行描述,再用 C++ 实现的中间件,见表 1.

表 1 三种方案效率对照表

| 实现方案 | 文件大小 | 并发用户数 | 总共耗时 |
|------------|------|-------|---------------|
| 优化前 JAVA | 1M | 1 | 1930~ 2010 毫秒 |
| 优化后 JAVA | 1M | 1 | 1372~ 1392 毫秒 |
| CORBA, C++ | 1M | 1 | 1484~ 1552 毫秒 |
| 优化前 JAVA | 2M | 1 | 3205~ 3360 毫秒 |
| 优化后 JAVA | 2M | 1 | 2614~ 2632 毫秒 |
| CORBA, C++ | 2M | 1 | 2715~ 2820 毫秒 |

从表 1 可见,经过优化 I/O 的 JAVA 程序实现的中间件,在效率上大大提高,与用 IDL 对 CORBA 构件进行描述再用 C++ 实现的中间件相比也毫不逊色,甚至超出.它在分布异构环境中传送数据文件速度达到 0.18 M/秒左右,完全可以应付电力调度通信中心日常的需要.

41.2.2 创建连接池,优化资源配置,提高效率

在电力调度中心日常的工作中,往往有多个并发用户同时请求连接相同的服务器,传送相同的数据.如果仅仅简单地运行多线程,每当有客户机申请一个连接时都在一个新 Thread 中创建一个新 ConnectionHandler.这意味着可能有一捆 Thread 躺在我们周围,而且创建 Thread 的系统开销并不是微

不足道的,这样当多个用户同时连接时将大大影响系统的效率^[5].为了解决这一问题,我们创建一个可以维护的连接池,一定数量的 ConnectionHandler 将为它提供服务.这种设计限定了允许同时连接的数目,只需启动 ConnectionHandler Thread 一次.在服务器启动时创建一定数量的 ConnectionHandler,把进入的连接放入连接池中并让 ConnectionHandler 处理剩下的事情.池中可能永远不会有多个连接,因为事情将很快就被处理掉.如果池中有一个以上连接,那么其他处理程序也不必等待系统添加新的连接到池中,当它们检查 pool.isEmpty() 条件发现其值为假时,就会从池中攫取一个连接并处理它.这样既保证了系统不会因大量用户同时进入而瘫痪,也提高了系统处理多用户的综合效率.

图 3、图 4 是实际的测试结果,分别以文件传输和数据库查询为例,其中文件大小为 500 K,数据库中的记录为 100 条,连接池的最大连接数取 20,并发用户数分别为:1, 5, 10, 20, 40, 60.

从图 3 和图 4 可见,创建连接池优化资源配置后,无论是传送数据文件还是查询数据库记录,系统都不会因大量用户同时进入而瘫痪,并且提高了系统处理多用户的综合效率.即使面对 60 个并发用户的情况,系统在分布异构环境中传送数据文件速度可达到 1M/秒#用户左右,查询数据库记录的速度可达到 1000 条/秒#用户,这已经完全可以应付电力调度通信中心日常的需要.

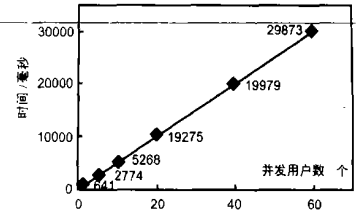


图 3 多用户文件传输效率图

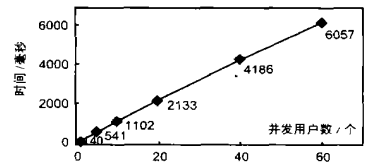


图 4 多用户数据库传输效率图

5 结论

中间件的设计采用 JAVA 与 CORBA 结合的方案,集中了各自的优点,同时实现了中间件的平台无关性和可移植性.本文采用了优化 I/O 性能和创建连接池两种关键技术,解决速度问题,优化资源配置,保证了中间件的高效率.经过实验测试,在单用户的情况下,分布异构系统间交换数据文件的速度为 0.18M/秒左右,查询数据库记录的速度为 720 条/秒左右;在通常 20 个并发用户的情况下速度分别为 1M/秒#用户左右和 930 条/秒#用户左右.这样的性能已完全适合在电力系统中应用,比较好地解决了电力调度中心分布式异构系统间数据交换的问题.

参考文献:

[1] 韦乐平,薛君敖,孟洛明. CORBA 系统结构、原理与规范 [M]. 北京:电子工业出版社,2001. 62- 83.
 [2] Balen Henry. Distributed Object Architectures with CORBA [M]. Cambridge, New York: Cambridge University Press SIGS Books, 2000. 200- 230.

- [3] Vlada Matena, Beth Steams. J2EE 平台上的 EJB 组件开发 [M]. 瞿裕忠, 等, 译. 北京: 机械工业出版社, 2001. 296- 302.
- [4] David Reilly. Java 与 CORBA) 一种光滑的融合[OL]. 阿霏, 译. <http://www.corba.com.cn/download/200122603/index.html>.
- [5] Michael Orey. SQL Server 2000 开发指南 [M]. 北京: 清华大学出版社, 2002. 122- 143.

作者简介:



朱 斌 男, 1940 年 3 月出生于云南个旧, 教授, 研究生导师, 中国电子学会高级会员, 1963 年毕业于华南理工大学计算机专业, 毕业后至今长期在华南理工大学计算机学院从事教学和科研工作, 参加和主持过国家自然科学基金, 国家九五攻关, 广东省多项科研, 主要研究方向是计算机应用, 智能控制, 曾获国家教委, 省科委科技进步奖.

进步奖.



刘子欣 男, 1978 年 12 月出生于广东广州, 2001 年毕业于广东工业大学信息工程学院并获得学士学位, 现为华南理工大学硕士研究生, 主要研究方向是智能控制, 计算机应用.



朱海云 女, 1977 年 8 月出生于广州, 助工, 1999 年毕业于华南理工大学计算机系并获得学士学位, 毕业后于广州市, 广东省电信公司科学技术研究院从事电信及计算机方面的技术研究及开发.