

基于 Windows 平台的企业应用集成 解决方案的研究*

刘永壮, 刘 萍

(北京科技大学 信息工程学院 计算机系, 北京 100083)

摘 要: 首先阐明了企业应用集成的意义以及需要解决的问题, 并由此引出了 EAI 解决方案, 并对 MSMQ, XML, Biztalk Server, COM 等相关技术给予了简要介绍, 结合方案设计图重点阐述了如何综合以上技术去构建企业应用集成方案。

关键词: MSMQ; Biztalk Server; 企业应用集成; B2B; 分布式应用

中图法分类号: TP311 **文献标识码:** A **文章编号:** 1001-3695(2004)11-0175-03

Study of the Solution to Enterprise Application Integration Based on the Windows Platform

LIU Yong-zhuang, LIU Ping

(Dept. of Computer, College of Information Science & Engineering, Beijing University of Science & Technology, Beijing 100083, China)

Abstract: The paper points out the purposes of Enterprise Application Integration(EAI) and the problems in EAI at first, and then derives the solution to EAI discussed in the text. Later the paper introduces the interrelated technologies including MSMQ, XML, Biztalk Server and COM. Finally with the graph the paper discusses how to design the solution to (EAI) using the technologies above-mentioned in detail.

Key words: MSMQ; Biztalk Server; EAI(Enterprise Application Integration) ; B2B; Distributing Application

1 引言

为什么要进行企业应用集成(EAI)? 特别是被用来进行企业运作的关键任务的应用程序, 代表着已证明的功能块, 它们能否正常运转, 关系着组织的业务能否正常开展。它们代表着订购供应、生产进度的制定、客户订单的执行以及公司财务的协调。

这些业务应用程序在许多公司内部是各自独立作用的。但是实际上这些业务过程很少是单独发生的, 因此就需要人工进行集成。例如, 业务部用自己的系统打印出订单, 然后交到产品部, 产品部再将订单内容录入产品部软件系统。这势必会对组织的最终产品或服务造成时间的延迟与费用的增加。因此任何能够用自动化信息处理取代人工集成的做法, 都会加快公司的运作并减少开销, 所以我们希望能够从工业革命时代进入互联网时代。

从更高的角度看这个问题, 其实公司本身就是整个过程的步骤。几乎没有哪个公司去做从原料到成品之间的每一步工作。公司在业务关系网络中运作, 一个公司为其他公司提供他们所需要的东西。对于单个组织内的处理, 公司多数通过人工过程交互。但是, 如果一个公司内部的库存应用程序能够直接

利用一个可信任的供应商的订单输入系统下订单的话, 整个进程就可以进行流水作业了。本质上, 这就是在公司间实现的 EAI, 也常常被称为电子商务(B2B)。

由此可见, 随着互联网技术的日益成熟, 企业应用集成势在必行。但企业应用集成在具体的实施中面临着很多问题: 要保持原有各应用程序的独立性, 不能影响业务的正常运转, 因此对原有应用程序的修改要尽量少; 由于历史原因, 原有各系统的开发工具和数据格式可能千差万别; 如网络风险问题等。本方案旨在解决以上问题, 建立一个基于 Windows 平台的企业应用集成的通用性的架构。本方案涉及到以下主要技术及工具: MSMQ, XML, Biztalk Server, COM。

2 MSMQ 简介

从概念上讲, 消息队列是用来建立分布系统的一种简单模型。应用程序创建一个消息并把它送到消息队列中。而另一个应用程序从该消息队列中读入该消息, 然后读入该消息的应用程序可以把一个消息发送到另一个消息队列。该消息队列可以由最初消息的发送者来读取, 也可以由其他应用程序来读取。消息队列是异步的, 一旦某个消息放入了消息队列, 发送该消息的应用程序就可以转去做其他的工作, 而不必等待其他应用程序来读取该消息。消息队列还允许脱机工作, 也就是说即使某个应用程序还没有接入网络时, 也可以发送消息, 一旦该应用程序重建了网络连接, 该消息就可以进入消息队列。消

息队列是十分可靠的技术,可以把消息队列存储在永久存储器中,在系统崩溃后恢复运行时,可以重建有关的队列操作。

图 1 是消息机制的示意结构。



图 1 消息机制示意图

3 XML 简介

XML(eXtensible Markup Language)中文可以译成可扩展标记语言。在 Web 和 Internet 日益发达的今天,人们对互联网的依赖也越来越深。不同的操作平台,不同的应用系统,甚至不同企业间的数据互通,都需要一个共同的数据格式定义,XML 的出现可以解决这一切的问题。XML 提供了一套跨平台、跨网络、跨程序语言的数据描述方式,各行各业的人都可以针对自己的需求定义所要的文件格式,并且以 XML 文件当作数据交换的标准格式。所以在目前以及将来,XML 将在 Web 中扮演举足轻重的角色。

单纯的 XML 只是一个可以自由定义文件格式的语言,但是与 XML 相关的技术和标准却很多,这些 XML 相关技术和标准与 XML 构成完整的 XML 架构。新一代分布式应用,企业整合,或是各种 B-to-C 和 B-to-B 电子商务,甚至是网络服务,都将是 XML 为基础的数据格式,再搭配支持 XML 的各种标准与相关技术来完成。

4 Biztalk Server 简介

Microsoft Biztalk Server 是 Microsoft 公司 .NET 企业服务器产品之一,它使开发者、专业 IT 工作者和商业分析家非常轻松地在 Internet 建立一个跨平台、跨应用、跨企业的动态业务过程。NET 框架是微软公司组件对象模型(Component Object Model, COM)的精华,将它们与松散连接计算的精华有机地结合起来,生成强大、高效的 Web 组件系统,达到简化程序员的管道操作。Biztalk 深入地集成了安全性,引进了基于互联网的操作系统,极大地改善了应用程序的可靠性和可扩展性。Biztalk Server 是 Microsoft 公司推出的数据与商业流程集成的服务器,旨在促进企业内部及企业之间电子商务流程的协作。它提供了强大的基于 Web 的开发和运行环境,这个环境集成了企业内部和企业与企业之间的商务流程。Microsoft Biztalk Server 有助于组织快速建立并管理与其他组织的 Internet 关系。它使得这些组织可以与其他任何伙伴组织自动交换文档,而无须考虑转换要求和所使用的数据格式。这为企业之间集成商务流程提供了一个经济的方法。

5 COM 简介

COM 是 Microsoft 创建的一种二进制和网络标准,它允许任意两个组件互相通信,而不管它们是在什么计算机上运行(只要计算机是相连的),不管各计算机运行的是什么操作系统(只要该操作系统支持 COM),也不管该组件是用什么语言编写的。COM 组件对内部对象进行了很好的封装,组件与应

用程序或其他组件之间通过接口进行通信(接口通常是一组函数)。在 Windows 系统平台上,一个 COM 组件可以是一个 DLL(Dynamic Linking Library,动态链接库)文件,也可以是一个 EXE(可执行程序)文件。一个组件程序可以包含多个 COM 对象,并且每个 COM 对象可以实现多个接口。

6 企业应用集成解决方案

环境需求:Windows 2000 Server 或更高版本;SQL Server 2000,这是安装 Biztalk Server 的需要,因为 Biztalk Server 本身需要 SQL Server 2000 支持管理它的数据;Biztalk Server 2000 或更高版本(如 2002,2003 等);支持 COM 技术的开发语言,如 Visual Basic 6.0, Visual C++ 6.0, Delphi 等。

本方案采用 MSMQ(消息队列)作为各应用程序的通信工具,用 XML 标准作为消息定义的规范,采用 Biztalk Server 作为消息模板定义、转换及消息传递的服务器。MSMQ 技术帮助我们规避了网络风险,保证在网络暂时不通的情况下,各子系统还可以继续运转。XML 技术解决了各子系统不同的数据格式问题。而 Biztalk Server 增强了系统的稳定性并减少了对原有应用程序的修改工作。

假设某企业现有 A 系统与 B 系统两个部门系统,现在我们采用本方案对两个系统进行集成。首先,给出方案架构图,如图 2 所示。

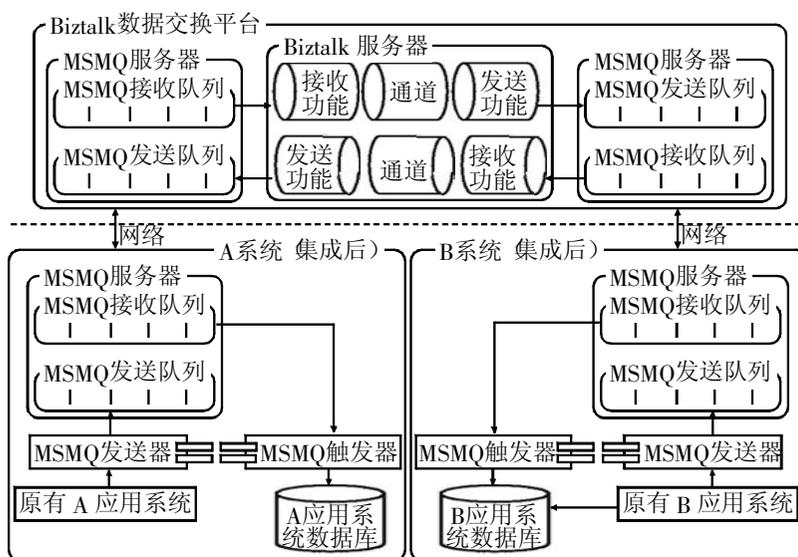


图 2 方案架构图

下面我们结合图 2 对方案设计进行说明。整个框图分为三部分:A 系统(集成后)、B 系统(集成后)、Biztalk 数据交换平台。我们先看改造后的 A 系统。图 2 中包括了原有 A 应用系统及原有 A 应用系统数据库。另外,增加了 MSMQ 服务器。MSMQ 服务器是 Windows 2000 Server 自带的附属产品,如果当前没有安装,可通过控制面板 添加或删除程序 添加或删除 Windows 组件,从 Windows 2000 安装盘中安装。在 MSMQ 服务器中,我们可以看到有 MSMQ 发送队列和 MSMQ 接收队列。当通过消息队列向网络中的其他计算机发送消息时,消息首先将到达本机器的发送队列中,然后再到达目标机器的接收队列。如果消息正常到达目的地,则本机器的发送队列清空;否则如果网络不通,不能到达目的地,消息会一直存储在本机器的发送队列中。此时,MSMQ 服务器会检测网络的情况,一直

等到网络畅通时, 它会将发送队列的消息取出, 发送到对方机器的接收队列。如果发送成功, 清除自己的发送队列中相应消息, 整个发送过程结束。可见, 在发送消息的过程中, MSMQ 服务器起着决定性的作用。它协调发送队列和接收队列, 保证整个工作的圆满完成。除了 MSMQ 服务器, 还有 MSMQ 发送器和 MSMQ 触发器。如果熟悉 UML, 通过框图符号就可以知道这是两个 COM 组件。MSMQ 发送器是一个启动 MSMQ 服务, 向消息队列发送消息的功能模块。它指定接收队列的地址(消息队列的地址由 IP 地址或机器名及队列名确定), 与 MSMQ 服务器一起, 完成整个消息传递的过程。原有的应用系统, 将在合适的时机调用 MSMQ 发送器, 向其他系统发送消息。MSMQ 触发器同样是一个 COM 组件, 当指定的接收队列有消息到达时, MSMQ 触发器被触发, 完成赋予它的任务。通常, 它的任务之一就是将接收到的消息处理成自己需要的数据写入自己的数据库。至此, 关于 A 系统(集成后)的部分说明完毕。至于图 2 右边的 B 系统, 情况完全一样。

Biztalk 交换平台部分: 图 2 中的左右两边各有一个 MSMQ 服务器, 用来与 A, B 两个子系统进行通信。关于 MSMQ 服务器前面我们在对 A 系统(集成后)阐述时已经进行了介绍。这里我们重点说明框图中的中间部分, 即 Biztalk 服务器。Biztalk 的接收功能通常通过接收函数(Receive Function)来实现。接收函数需要在 Biztalk 中配置, 配置时需要指定接收队列的地址及通道的名称。Biztalk 的最大特点就是用简单的配置代替编程。通道由三个部分组成: 模板一 映射 模板二。模板一和模板二分别是 A, B 两个子系统的模板。模板是一个 XML 文档, 它定义了消息的规范。通常为了方便, 我们可以按照子系统的数据库结构定义模板。模板定义以后, 各子系统发送的消息格式必须要遵守模板的规范。这样各子系统只需按照自己系统的特点定义自己的模板, 而不用花费精力去研究对方的数据库结构(而且, 一般各子系统的数据库结构也不愿意向别人公开)。不同模板之间的转换交给 Biztalk 服务器完成(同样, 模板映射需要在 Biztalk 中配置完成)。发送功能是通过在 Biztalk 中配置一个消息端口来实现的, 消息端口与通道相连接, 同时要指定接收消息队列的地址。Biztalk 服务器的整个活动过程是这样的: 当有消息到达时, 接收函数取走消息并传递到通道, 通道首先检查消息的格式, 是否与模板一符合, 如不符合停止传递, 并将错误信息写入错误日志, 将消息放入 Biztalk 的悬挂队列(Suspended Queue); 如符合, 将消息的格式由模板一的格式转换为模板二的格式, 然后再将转换后的消息发送到另一系统的消息队列中。

下面我们以 A 系统向 B 系统发送消息为例说明整个过程。原有 A 应用系统调用 MSMQ 发送器按照事先定义好的模

板格式发送消息(消息为一个 XML 格式的字符串), 消息到达 Biztalk 数据交换平台的消息队列。接收函数接收消息并将消息传递到通道(Channel)。通道根据模板检验消息的数据格式, 然后对消息进行模板转换, 并将转换后的消息发送到 B 系统的接收队列。此时, B 系统的 MSMQ 触发器被触发, 触发器程序执行解析消息、入库及其他业务逻辑处理等操作, 并决定是否要向 A 系统回复消息。同样, B 系统向 A 系统发送消息的过程类似, 在此不再赘述。

限于篇幅, 我们只进行了两个系统的集成。对于更多系统的集成, 也只是量上的差别, 如在图 2 中再重复一个 C 系统(集成后)或 D 系统(集成后)。无论增加多少, 原理都是一样的。当然, 在具体实施中, 还有很多技术问题, 读者可参考后面列出的参考文献, 它们可以帮助我们掌握相关技术细节。

7 结论

通过上面的讨论不难看出, 应用本方案构建企业应用集成(EAI), 不需要对原应用程序进行多少修改就可以做到无缝集成, 而且可以解决在企业应用集成中遇到的大部分问题。本方案聚集了 MSMQ, XML, Biztalk Server, COM 等当前先进技术, 在 EAI 中, 每一项关键技术都充分展现出了它的魅力, 起着至关重要的作用。当然, 在具体实施中, 上述技术也并非缺一不可, 我们可以根据实际情况对设计方案进行适当修改。比如, 当在 Internet 上进行 B2B 集成时, 我们可用 HTTP 传输取代 MSMQ 传输。再如, 对于企业内部集成, 当需要集成的系统彼此熟知对方的数据结构同时又无保密性可言时, 可以不用 Biztalk Server。但是, 通常情况下, 集成本文提到的各项技术设计出的方案是一个比较不错的选择。

参考文献:

- [1] 林锦雀. 最新 XML 入门与应用[M]. 北京: 中国铁道出版社, 2001.
- [2] 余英. Visual C++ COM 和 COM+ 篇[M]. 北京: 中国铁道出版社, 2001.
- [3] Robert J Oberg. 深入学习: COM+ 高级编程[M]. 北京: 电子工业出版社, 2001.
- [4] Stephen Mohr, Scott Woodgate. Biztalk 高级编程[M]. 北京: 清华大学出版社, 2002.

作者简介:

刘永壮(1976-), 男, 河北乐亭人, 硕士研究生, 主要研究方向为分布式应用及系统集成; 刘萍, 女, 北京人, 副教授, 主要研究方向为计算机仿真及应用。

(上接 174 页) Directed Graphs[J]. Software: Practice and Experience, 1988, 18(11): 1047-1062.

[2] P Luder, R Emst, S Stille. An Approach to Automatic Display Layout Using Combinatorial Optimization Algorithms[J]. Software: Practice and Experience, 1995, 25(11): 1183-1202.

[3] 孙昌爱, 刘超, 金茂忠. 一种有效的软件结构图的布图算法[J]. 北京航空航天大学学报, 2000, 26(6): 706-708.

作者简介:

欧胜高, 硕士研究生, 主要从事软件工程、软件测试方面的研究; 刘超, 教授, 主要从事软件工程、软件测试和面向对象技术方面的研究。