

文章编号: 1001 - 9081 (2010) 06 - 1658 - 03

面向企业应用集成的 ESB 框架的研究

林泳琴<sup>1</sup>, 黄晨晖<sup>2</sup>

(1. 广州大学 网络与现代教育技术中心, 广州 510006; 2 广东女子职业技术学院 网络中心, 广州 511450)  
(lyq03148@gzhu.edu.cn)

**摘 要:**企业服务总线 (ESB) 是基于事件驱动的消息传递机制的标准软件架构, 消息总线是 ESB 的基础。为实现高效、安全、可靠的 ESB, 分析了传统 ESB 框架中消息传递的关键组件及其工作原理, 并在此基础上设计并实现了消息切分、压缩、加密、断点续传, 以及集群队列等功能的企业级消息传递系统。实验结果表明, 通过改进消息传递模型, 可进一步提升 ESB 的性能。

**关键词:**企业服务总线; 面向服务架构; 消息切分; 断点续传

**中图分类号:** TP311; TP393      **文献标志码:** A

Research on ESB framework for enterprise application integration

LN Yong-qin<sup>1</sup>, HUANG Chen-hui<sup>2</sup>

(1. Network and Modern Educational Technology Center, Guangzhou University, Guangzhou Guangdong 510006, China;  
2. Network Center, Guangdong Women's Polytechnic College, Guangzhou Guangdong 511450, China)

**Abstract:** Enterprise Service Bus (ESB) is a standard software architecture based on an event-driven messaging engine, of which message bus is the foundation. In order to achieve an effective, secure and available ESB, this paper analyzed the key components and working principles of messaging on traditional ESB, and then proposed the design and implementation of an enterprise level messaging system, which had the functions of message splitter, message compression, message encryption, message resume, and queues cluster. The application result shows that improving the messaging model can further improve the performance of ESB.

**Key words:** Enterprise Service Bus (ESB); Service-Oriented Architecture (SOA); message splitter; message resume

0 引言

在文献 [1] 中对应用集成模式进行了分类, 包括文件传输、共享数据库、远程过程调用和消息 4 种模式。

1) 文件传输: 两个系统生成文件, 文件的有效负载就是由另一个系统处理的消息。比如: 针对文件轮询目录或 FTP 目录, 并处理该文件。

2) 共享数据库: 两个系统查询同一个数据库以获取要传递的数据。比如: 两个 EAR 应用, 它们的实体类 (JPA、Hibernate 等) 共用同一个表。

3) 远程过程调用: 两个系统都暴露另一个能调用的服务。比如: EJB 服务, 或 SOAP 和 REST 服务。

4) 消息: 两个系统连接到一个公用的消息系统, 互相交换数据, 并利用消息调用行为。应用案例: 中心辐射式的 JMS 架构。

企业应用集成 (Enterprise Application Integration, EAI) 的集成级别经历了数据级集成、消息级集成和现在的过程级集成, 面向构架服务 (Service-Oriented Architecture, SOA) 是目前 EAI 领域中应用最广的体系结构, 而企业服务总线 (Enterprise Service Bus, ESB) 则是实现 SOA 的基础设施, 图 1 显示 ESB 在 SOA 参考架构中位置<sup>[2]</sup>, 图 1 表明, ESB 是 SOA 参考架构的中心构件, 实现服务注册、管控和服务之间的交互, 是一种在松散耦合的服务和应用之间标准的集成方式。在集成技术中, 相比于 CORBA、COM/DCOM 等技术具有代价低、耗时短、

集成工作量小等优点, 另外, 它还弥补了传统的基于 Web Service 集成技术点对点连接的缺陷, 进一步实现了异构环境中服务的解耦, 为企业应用提供了廉价、便捷、高效部署的集成方案。

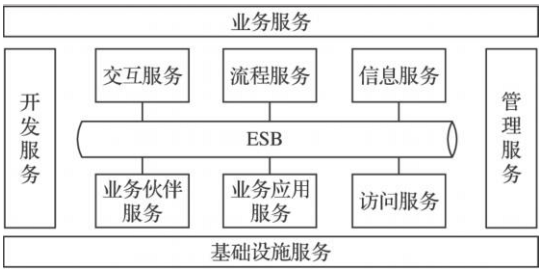


图 1 ESB 在 SOA 参考架构中的位置

ESB 在实现上是基于服务构件组的, 通常包括数据转换构件组、文件传输构件组、Web Service 接入/调用构件组、DB 接入/输出构件组等。另外, 它还提供规则引擎 (Rules Engine, RE)、消息格式转换 (Message Format Transformation, MFT)、服务编排引擎 (Service Orchestration Engine, SOE)、服务注册库 (Service Registry Repository, SRR) 等功能。

目前大多数 ESB 产品一般都实现了基于标准的消息通信架构、基于标准的互连 (如 Web 服务、J2EE 和 .NET 适配器)、基于标准的数据转换引擎 (如 XSLT 和 XML Query)、应用部署的 SOA 方式以及基于标准的安全性 (如 LDAP 和 SSL), 以提供路由、消息传递、调解、复杂事件处理、调用、日

收稿日期: 2009 - 12 - 28; 修回日期: 2010 - 03 - 12。

作者简介: 林泳琴 (1978 - ), 女, 广东珠海人, 高级工程师, 硕士, 主要研究方向: 软件工程; 黄晨晖 (1978 - ), 男, 广东梅州人, 讲师, 硕士, 主要研究方向: 网络。

志、审计、安全认证和管理等功能。

其中:消息总线是 ESB 的基础,所有服务的交互最终都转化为消息的传递来实现,消息传递机制是其核心。近年来对 ESB 中消息传递模型的研究如文献 [3-6],它们的研究侧重于消息的路由规则和路由过程的实现,对传输的内容和传输性能欠缺考虑。本文将给出 ESB 中消息传递的主要组件及传递原理,并在此基础上给出改进的能实现消息切分、压缩、加密、断点续传和集群队列等功能的企业级消息传递 ESB。

## 1 ESB 中的消息传递模型

### 1.1 模型中的核心概念及组件

消息 (Message):是传递到端点的有效负载。为传输数据,应用将数据打包成一个或多个 Packet,将每个 Packet 包装成一个 Message,然后将其传输到消息通道 (Channel)。消息在实现上可以只是一个字节流,这个字节流可以是一个 XML 文档、一个序列化的 Java 对象、一个文本字符串,甚至是一条空消息。

消息队列 (Queue) 和主题 (Topic): 消息队列是一个安全的存储消息的地方。主题可以被认为是消息的传输中介,发布者 (Publisher) 发布消息到主题,订阅者 (Subscriber) 从主题订阅消息。主题使得消息订阅者和消息发布者保持互相独立,不需要接触即可保证消息的传送。在实现上,每一个队列和一个程序相联系,一个程序可以有一个或者多个接收队列,而且可以有多个输出队列。

消息通道 (Channel):是两个端点间被命名的连接。消息在经过处理管道的过程中,会在总线内被转化、路由至其他通道、分发、响应,或者被中断并发送到一个死消息通道中去。

端点 (Endpoint):是消息通道 Channel 的终点。大多数的应用程序没有内建的能力来同一个 Message 系统交互,因此它们必须包含一个中间层,它知道应用系统如何工作,也知道 Message 系统如何工作,并桥接两个系统。

路由器 (Router):可分为入站路由器、出站路由器及异步回复路由器等,分别用于控制服务如何处理入站消息,控制如何分发经服务处理过的消息,用于 Request/Response 场景中返回响应结果。另外,所有的路由器都支持过滤器 (Filter)。目前大多 ESB 产品路由功能的实现通常由消息中间件 (Message Oriented Middleware, MOM) 服务器来完成路由功能。

### 1.2 消息传递原理

消息模型本身基于元模型,元模型是一种表示消息内容的方法。消息元模型的一个示例是 XML 模式定义语言,事实上,这是 ESB 产品中最常用的消息元模型。消息模型是元模型的特定应用;内容模型的一个示例是某个特定的 XML 模式。ESB 必须显式支持至少一种消息元模型,并且能够支持多种消息元模型。

图 2 为 ESB 中的消息传递示意图,消息客户端发送消息到消息系统中的队列或主题。消息驱动 Bean 在部署到容器中时关联指定的消息队列或主题,并由容器对消息的队列或主题进行监听;当容器从队列或主题中接收到消息之后,将消息作为事件的一部分通知容器中相应的消息驱动 Bean 实例并由它根据消息的内容调用相应的业务逻辑或其他组件。

ESB 中的消息总线通常都支持以下两种传输模型。

1) PTP (Point-to-Point) 模型。该模型是基于队列的,发送方发消息到队列,接收方从队列接收消息。PTP 模型定义了客户端如何向队列发送消息,如何从队列接收消息和如何浏

览队列中的消息等。如图 3 (a)。

2) PUB/SUB 模型。该模型定义了如何向一个内容节点发布和订阅消息,这些节点被称为主题。发布者发布消息到主题,订阅者从主题订阅消息。主题使得消息订阅者和消息发布者保持互相独立,不需要接触即可保证消息的传送。如图 3 (b)。

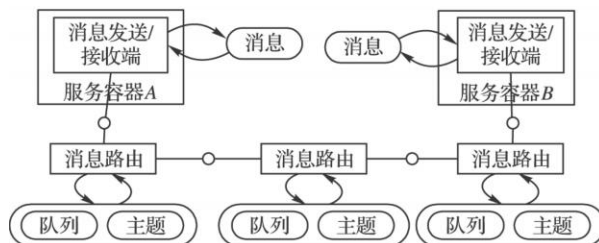


图 2 ESB 的消息传递示意图

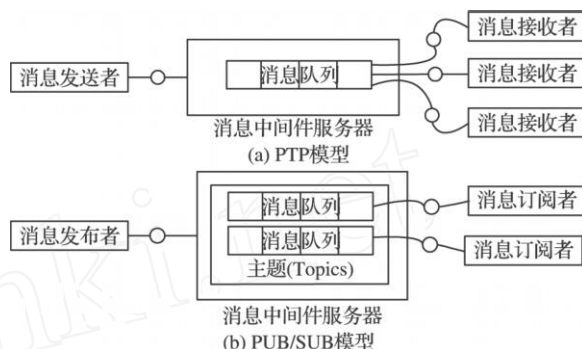


图 3 PTP 与 PUB/SUB 模型

## 2 改进的 ESB

JMS (Java Message Service) 是 J2EE 规范中提出的消息服务规范,提供应用程序间异步或同步的消息传递和管理服务,为进一步提高 ESB 在可靠、稳定、安全、高效等方面的功能,本文基于 JMS 对其消息中间件进行了改进,提供了实现消息分解、消息压缩、消息加密和断点续传等功能的具有可靠消息传输的 ESB。消息切分、压缩和加密如图 4 所示。

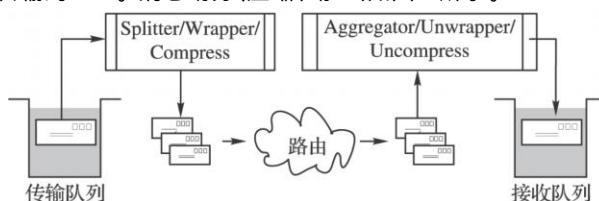


图 4 消息切分、压缩和加密

在消息分解功能中,支持将一个大的消息切分成多个小块发送,在目的节点重新组合成完整的消息。消息的切分通过 Splitter 组件将消息按固定长度分解成多个部分,并将它们路由到同一个端点来实现。消息的组合基于关联 ID 通过 Aggregator 组件重新将消息块聚合在一起来实现。另外,还支持将输出消息分解成多个部分,再将它们分发到配置在路由器上的不同端点。例如,在某些应用中,如果想将消息分解后的不同部分分发给不同的服务去处理,这就需要使用到基于列表的消息分解器或基于表达式的分解器的实现。切分的消息充分利用网络带宽,提高了大消息的传输效率,通过消息的切分,每秒最大可以传输几十吉字节的大消息。

在消息压缩功能中,提供数据压缩 API,方便用户在收发消息时对消息进行压缩和解压缩,以提高消息传输效率和服务器的吞吐量。消息的压缩和解压缩通过 Compress/Uncompress 组件实现。

对需要加密的数据,通过 Wrapper/Unwrapper组件的加解密功能来实现。它支持 SSL 管道安全机制,以保护整个通信的安全。考虑到对机密数据的加密,基于 HTTP的安全是不够的,它仅能解决点到点的安全问题,因此为实现消息在转发中跨多个跃点传递时通信的安全性,系统添加了基于消息级别的安全性实现方案:XML 标准加密,它可以用于加密 XML 消息或仅加密消息的一部分,且用此方法加密消息,在转换为 JMS格式时安全信息不会丢失。为提高加解密性能,使用非对称加密生成和保护要共享的密钥,使用对称加密来执行数据加密。当收到加密的消息时,接收方使用私钥解密会话密钥,然后使用此会话密钥解密消息。这种机制确保了只有受信方才可以执行消息的解密,也就是说,这种机制可以有效地保证某个主题的数据的机密性,只有拥有正确私钥的订阅者才能正确执行解密。

另外,系统还支持集群队列,支持多个节点的多个不同的队列,在逻辑上形成一个集群队列,实现节点网络中物理队列的透明性。普通消息和大尺寸对象消息的断点续传也是本文所支持的功能之一,通过断点续传,解决由于网络原因导致消息传输中断的问题,使得当网络恢复时,不需要重新传输整个消息,而只需从中断的位置传送即可。

3 应用实例

本校在实现数字化校园各系统集成中采用了该 ESB 作为基础架构,实现了教务系统、OA 系统、人事档案系统、学工系统、迎新系统及其他遗留系统之间的无缝集成,很好地解决了不同系统间相关数据不一致和跨系统业务集成的问题。ESB 系统基于 Linux操作系统,分别采用 Apusic 6.0和 Oracle 10g作为应用服务器和中心数据库服务器,LoadRunner测试结果表明,ESB 系统 90%事务响应时间在 1 s内,事务平均响应时间为 0.6 s,系统资源占用方面数据库服务器 CPU 平均占用不超过 20%,系统可以承受不低于 2000个在线用户,不低于 100个并发用户;系统在大量用户使用和长时间反复运行

中,未出现 CPU、内存占用过高、内存泄露等不良反应,系统运行良好。

4 结语

本文给出了 ESB 框架中消息传递的关键组件及其工作原理,并在此基础上提出实现消息切分、压缩、加密、断点续传以及集群队列等企业级消息传递系统的设计与实现,从而提供高效、安全、可靠的 ESB 实现,并用之构建了本校数字化校园建设中各应用系统的集成平台。实践表明,该数据集成方案一方面保持了原有系统的独立运行,另一方面,又很好地实现了各系统之间相关数据的互连互通和一致性,底层可靠、高效、安全的传输机制为 ESB 很好地支持跨部门的数据和业务集成提供了保证。

参考文献:

[1] GREGOR H, BOBBY W. Enterprise integration patterns: Designing, building, and deploying messaging solutions [M]. England: Addison Wesley, 2003.

[2] ASHOK I, VNOD J, MICHELE C. Websphere business integration primer: Process server, BPEL, SCA, and SOA [M]. New York: BM Press, 2007.

[3] 李秀林,张祖平. 基于动态消息路由的 ESB 框架的研究与应用[J]. 计算机系统应用, 2009, 18(9): 132 - 135.

[4] BAI XIAOYING, XIE JIHUI, CHEN BEN, et al. DRESR: Dynamic routing in enterprise service bus [C]// ICBE'07: IEEE International Conference on e-Business Engineering. Washington, DC: IEEE Computer Society, 2007: 528 - 531.

[5] WU BEN, LIU SHIJUN, WU LEI. Dynamic reliable service routing in enterprise service bus [C]// 2008 IEEE Asia-Pacific Services Computing Conference. Washington, DC: IEEE Computer Society, 2008: 349 - 354.

[6] GULNOZA Z, EUNMIC, DUGKIM. Content-based intelligent routing and message processing in enterprise service bus [C]// International Conference on Convergence and Hybrid Information Technology 2008. Washington, DC: IEEE Computer Society, 2008: 245 - 249.

(上接第 1657 页)

关联度较小时相似度值与经验值相差较大。但是一般在选取扩展词时会忽略关联度较小的关联词,因此对查询扩展的影响不大。

最后在水稻生产经营过程本体随机选取了 20 个概念,在该语义检索系统和未加入查询扩展的 Lucene检索系统中分别进行了检索,比较结果如表 2 所示。

表 2 系统查准率比较

检索系统	$P_1$	$P_2 / \%$	$P_3$	$P_4 / \%$
本文系统	27.2	90.6	78.3	78.3
原 Lucene系统	22.4	74.6	61.2	61.2

注:  $P_1$  为前 30 篇中相关文档数;  $P_2$  为前 30 篇查准率;  
 $P_3$  为前 100 篇中相关文档数;  $P_4$  为前 100 篇查准率。

本文系统的检索结果的前 30 篇和前 100 篇查准率与未进行查询扩展的 Lucene检索系统分别有 21%和 28%的提高。

5 结语

本文在开源检索工具包 Lucene的基础上设计并实现了一个语义检索系统模型。该模型利用领域本体对用户查询进行扩展,丰富了用户查询语义,并在 Lucene原有的检索结果相关性算法中加入了语义信息,经在实验中分析和验证,能有

效提高小型领域信息检索时的准确率。在概念相似度计算过程中存在大量的权值设定,权值设定的不同对结果有很大的影响。如何选择最合适的权值是未来工作的重点。

参考文献:

[1] 郑世明,任在安,宋自林,等. 基于 Ontology 的语义查询分析研究[J]. 南京师范大学学报, 2008, 8(4): 63 - 67.

[2] 熊桂喜,王开锋. 基于语义的查询扩展研究[J]. 微计算机信息, 2008, 24(10): 177 - 178.

[3] TVERSKY A. Features of similarity [J]. Psychological Review, 1997, 84(4): 327 - 352.

[4] RESNIK P. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity and natural language [J]. Journal of Artificial Intelligence Research, 1999, 11: 95 - 130.

[5] LEACOCK C, CHODOROW M. Combining local context and wordnet similarity for word sense identification [M]. Cambridge, MA: MIT Press, 1998: 265 - 283.

[6] 黄果,周竹荣,周亭. 基于领域本体的语义相似度计算研究[J]. 计算机工程与科学, 2007, 29(5): 112 - 116.

[7] 张承立,陈剑波,奇开悦. 基于语义网的语义相似度算法改进[J]. 计算机工程与应用, 2006, 42(17): 165 - 166.

[8] 李鹏,陶兰,王弼佐. 一种改进的本体语义相似度计算及其应用[J]. 计算机工程与设计, 2007, 28(1): 227 - 229.

