

基于组件群的分布式第三方物流信息系统的研究

陈 冈

(武汉科技学院经济管理学院 武汉 430074)

摘 要 第三方物流信息系统的建设应充分注意可复用组件技术的应用和推广,提出了物流信息系统五个紧密相连的组件群构成。结合第三方物流的特点,研究了第三方物流信息系统的三层分布式体系结构的设计。分布式数据处理有利于物流信息系统性能的提升,讨论了分布式数据库的结构形式、存储机制和访问策略。

关键词 组件群 分布式 第三方物流 信息系统

第三方物流一般是指企业的物流劳务交由供方、需方之外的第三方去完成的物流运作方式。它是物流业发展到一定阶段的产物,是物流专业化的一种形式。物流信息系统是第三方物流企业运作的基本条件。物流信息系统可以适应物流信息源点多、分布广、信息量大、动态性强、信息的价值衰减速度快、及时性要求高的特征,缩短物流的传输长度,实现物流各环节的有效沟通,必须加快建设第三方物流信息系统的建设。

1 基于组件设计的必要性

近年来,我国的物流业发展很快,物流信息化也已经取得了可喜的成绩,但是中国物流信息化整体水平偏低,信息技术在物流方面的应用还比较少。从现实情况来看,许多物流企业和软件专业厂商已经组织开发了一些第三方物流信息系统,极大地推动了物流业的发展。但是,随着软件系统规模的扩大和复杂性的增加,软件开发时所耗费的人力、物力和运行时所占用的硬件资源及运算时间的增加,软件系统的可扩展、可维护性却明显降低。当物流系统的结构和规模发生变动时,特别是物流数据的积累导致数据库结构的变动时,由于程序之间的关联性很强,信息系统通常要做繁杂的修改甚至是整个代码的重写,而且这种物流信息系统的专用性很强,往往是某个物流企业的“专用品”,移植到其他物流企业或物流系统上的代价很高,难以实现复用。随着物流系统规模的日益扩大、网络用户数量的急剧增长和数据的不断积累,传统物流信息系统的开发和维护工作将会面临越来越大的困难。

物流企业信息系统不仅要求能够完成基本业务内容,还必须是可靠的、易扩展的、易维护的和安全的。随着经济全球化进程的推进,物流企业重新整合和并购也迫在眉睫。同时,基于互联网的各种技术也迅速发展起来,物流企业规模扩大,因此要求物流软件在功能、架构上作相应调整。目前的物流信息系统体系结构基本上都是采用 C/S 或者 C/S 与 B/S 结构结合,物流系统移植代价高昂,不能复用,已经不能适应物流业务复杂多变的特性。因此,物流信息系统的设计和实现应充分注意可复用组件技术的应用和推广,以便适应物流系统变化迅速的特点。

国内物流软件的组件化程度普遍比较低,因此在物流信息服务领域的一大难点是软件之间难以兼容、物流软件的适应性较

差。根据有关统计表明,采用合理的复用组件可以使得开发效率提高 5~8 倍。采用组件思想使应用开发有了明确的分工,一部分人专注于事务逻辑层组件的开发、测试工作,另一部分人则根据商业逻辑的需要,在 Web 页面选择和使用组件,使用组件的统一对外接口而无须了解其内部实现细节。

2 组件群的设计

软件复用是一个系统工程,包括软件生命周期各个阶段,从需求分析、设计、编码及测试和维护等都可以成为复用的对象。组件复用则是软件复用的核心,一个理想的可复用的组件必须是抽象的、可移植的,不能依赖于某一具体的硬件、编译程序和操作系统,并且是标准化的。目前,制定组件实现规范的机构主要有 OMG(Object Management Group)组织、Microsoft 公司和 Sun Microsystems 公司,其组件模型分别是 CORBA(Common Object Request Broker Architecture)模型、COM/DCOM(Component Object Model/Distributed Component Object Model)模型和 EJB(Enterprise Java Bean)模型。这三种组件模型都是针对二进制代码级组件制定的,为基于组件的软件开发提供了框架。

网络技术的应用、电子商务的发展,要求物流信息系统是可重用用于多个应用和不同环境的,或者是同一应用领域不同应用之间,因此需要采用可复用的组件模型。分析物流企业物流业务的层次结构,有利于更有效地建立物流信息系统的组件结构。第三方物流:涵盖了运输、仓储作业、采购、销售、库存计划、调度计划、销售预测等各个环节,是一个完整的分销资源计划和运作系统。根据第三方物流业务的特点,第三方物流信息系统由 5 个紧密相连的组件群构成,每个组件群由若干组件组成,每一个组件又可进一步地分解为输入接口、输出接口和任务处理三大部分。输入接口接收外界的信息输入;输出接口则向外界输出信息;任务处理完成物流业务逻辑的处理。物流信息系统组件群的划分,需要注意组件粒度问题。粒度过大可能失去可复用性,过小又可能增加系统开发和维护的复杂程度。基本原则是遵循一个组件对应一个逻辑上不可分割的物流业务单元的划分原则。具体划分如下:

2.1 业务运作组件群是业务处理系统中所有业务处理组件的集合 该组件群是物流运作的中枢神经系统,其核心职能是在满

足客户服务水平的前提下,实现物流总成本的最小化。它包括订单处理、运输、仓库运作以及预测和计划各环节中所有的流程控制和作业自动化。主要由订货管理、订货处理、订单执行、配送作业、运输/装船作业、采购作业等等组件组成。

2.2 计划处理组件群是计划管理系统的所有组件集合 该组件群以计划管理为核心目标,由销售预测和库存计划中目前库存的水平定期制订出进货计划,在进货计划被工厂或供应商调整和确认后,根据每个商品品种的物流服务水平要求和当前的库存状况,可以制订出所有产品库存水平和分布计划。对所有仓库库存的及时监控和订单执行情况的实时监控,将产生出短期的补货计划。主要包括战略计划、能力计划、物流计划、存货分布计划、制造计划、进货计划、产品分布计划、采购计划、补货计划等组件。

2.3 销售预测组件群是销售预测的所有组件集合 该组件群以市场信息、销售数据以及运作系统所产生的中间结果数据为依托,选择和建立预测模型,进行销售预测。有了良好的销售预测,库存和补货计划才有可能完成,同时也便于集中控制管理整个分销网络的库存水平、库存分布和补货策略。主要由预测模型、预测指标、销售预测等组件组成。

2.4 仓储管理组件群是满足完善的仓储管理要求的所有组件集合 该组件群主要确定配送渠道中制成成品库存位置、补充订货的时间和数量,进行仓库收货和存货控制管理。主要包括货位控制、出库控制、入库控制、仓储保管等组件。

2.5 货运管理组件群是实现货运管理的所有组件集合 该组件群主要实现多种产品的组合运输以及合理的车辆进出调度。包括承运商管理、货物配载控制、运输作业查询、车辆调度等组件。

上述各个组件群相互支撑,共同构成了一体化物流信息系统的运作流程。

3 分布式体系结构的设计

目前物流信息系统体系结构大多数采用 C/S 结构。C/S 体系结构将用户交互界面和业务应用处理与数据库处理相分离,服务器与客户端之间通过消息传递机制进行对话,对数据库的操作通过远程数据库访问的方式由后台数据库服务器完成。C/S 结构是一种较典型的集中式结构,各分支节点相连,各分支节点之间只能通过中心节点进行联系,由于应用系统的企业逻辑通常写在客户端的应用程序之中,使得系统的维护成本非常高,限制了对业务处理逻辑变化适应和扩展的能力,显然难以适应电子商务时代对物流数据处理的需要。

分布式多层系统把主要的企业逻辑写在应用服务器上,解决了企业组件复用的问题。同一逻辑功能写成组件形式后,Web 应用程序或一般的应用程序(Windows 应用程序)都可以复用。在分布式多层应用系统中,客户端程序/应用服务器和远程数据库服务器分布在不同的机器上。其中客户端程序主要是提供用户界面(User Interface),并且向应用服务器(或 Web 服务器)提出数据请求,而应用服务器又向远程的服务器提交数据请求。多级分布式体系结构利用应用中间件专门完成应用业务处理逻辑,客户端的功能注重在与用户交互和数据表征。后台数据库完成数据访问和数据管理。这种结构将复杂的业务处理分割成相互之间可交互通信的若干组件对象,并将其分配到多个应用服务器中间件上实现负荷均衡。因此,这是一种集中和分散相结合的体系结

构,这种结构使得数据处理安全、稳定、易维护、易扩展,比较符合第三方物流数据处理的要求,是物流信息系统发展的主要方向。

结合第三方物流的特点,提出第三方物流信息系统的体系结构如图 1 所示。

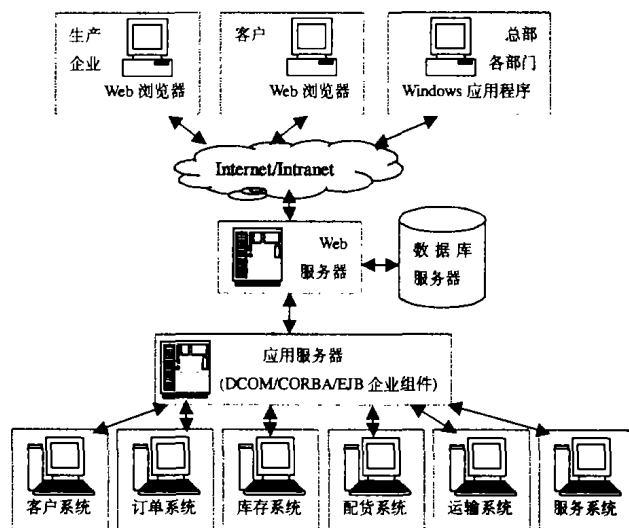


图 1 3PL 信息系统的体系结构

整个体系结构可以划分为三个层次。第一层是 Web 表现层。用于收集和显示信息,并利用网络浏览器实现对物流信息的浏览和各种物流指令的下达。第二层是商业应用层。它调用程序并与数据库连接实现对客户端信息的处理。可以利用 Java 具有多线程机制的 Servlet 进行逻辑控制,负责处理 HTTP 请求,通过 DCOM / CORBA / EJB 组件实现业务逻辑,访问数据库,以 JSP 实现表示功能。当客户端请求传递到商业应用层后,由控制 Servlet 首先处理。控制 Servlet 根据实际情况,进行数据的合法性检验或者转换成系统要求的形式,然后调用 DCOM / CORBA / EJB 组件访问数据库,完成具体的业务逻辑处理,最后通过 JSP 页面把处理结果反馈回客户端。JSP 页面只负责信息显示功能,一般不参与处理业务逻辑。这种应用逻辑和表示逻辑相互分离的形式,使得应用逻辑组件具有更好的独立性,也给代码重用带来了极大方便。这种分离的优势还在于创造了更多的可移植性、可重用性、平台无关性组件,大大提高了系统可维护性、扩展性和集成功能。第三层是数据库访问层。通过各种关系数据库对各种物流数据进行分布式管理,完成对各种数据的查询、更新操作。商业应用层的企业组件通过 JDBC 连接数据库,完成数据访问与存取逻辑。JDBC 定义了 Java 与数据库的接口类库,它是支持基本 SQL 功能的 Java 的通用底层应用编程接口,很容易方便地实现对异构数据库的访问。

上述分布式三层体系结构具有较多的优点:a. 系统易于集成。应用程序可以很容易地定位和共享应用程序服务器中的应用逻辑和数据,应用程序服务器能无缝地连入应用开发环境中。b. 系统易于移植和演进。可以在不改变应用程序代码的情况下改变计算机底层硬件、操作系统或通信协议,也可以进行局部修改而不会影响到系统的其它部分。c. 系统具有高可靠性。由于应用程序服务器具有接管和恢复的功能,能够保证事务及关键性业务不会被丢失。d. 系统还易于使用。应用程序服务器能和同构或异构环境下的多种数据源通信,同时它能管理数据间的公共逻辑约束,它将用户从复杂的平台、网络、数据库中解放出来,维护和升级方便,可跨平台操作,具有良好的开放性、可扩充性和安全性。

4 分布式数据处理

物流信息系统的数据处理可以集中在一台大型主机系统上进行,也可以分散在不同服务器的多个数据处理中心进行。同样,数据分布既可以存在于不同地域的数据库中,也可以只存在于一个中央数据库中。选择何种数据处理和分布结构,应该结合物流信息系统的计算结构、体系结构、企业的组织结构和功能结构进行。例如,对于一个全球性航运物流企业而言,既有总部的数据处理需求,也有区域和代理各自的数据处理需求,单纯采用集中式数据处理和分布结构显然难以满足需要。因此,需要注意分布式数据库技术的设计与应用。

分布式数据库系统是指物理上分散而逻辑上集中的数据库系统。分布式数据库的基本原理是利用计算机网络将地理位置分散,而管理和控制的需要不同程度的集中的多个逻辑单位连接起来,共同组成一个统一的数据库系统。因此,分布式数据库系统可以看成是计算机网络与数据库系统的有机组合,它可以由同一类型的数据库构成,也可以由异构数据库共同构成。近年来,分布式数据库已经成为信息处理中的一个重要领域,其主要特点是具有多层控制的结构。也就是说,由全局数据库管理员担负管理整个数据库的中央职责,而由若干本地数据库管理员负责管理各自的本地数据库。分布式的这种多层控制结构是当前多种应用系统体系结构中的主流,能支持更多的用户,对于 Internet 上的应用机制,有着更为廉价的系统维护成本。物流信息系统的应用范围很广,往往需要较多的节点来支持较多的客户端。因此使用分布式数据库技术处理物流信息系统的处理和分布结构,容易实现全局数据的一致性和完整性。这种既有集中又有分散的数据处理和分布结构,适应了物流企业地理上分散的子公司或部门对于数据库应用的需求。

从物流企业的现状和发展来看,物流企业都有若干个子分公司或相互关联的部门。虽然在业务上它们独立处理各自的数据,但彼此之间数据的交换和处理日趋频繁,存在着相互制约的关系。因此,进行分布式数据库设计时,需要首先确定数据的物理位置以及各个数据表的最佳存放位置,每个数据库中还要根据实际需求建立有关的数据库对象,确立数据存取机制,以及如何访问和存取其它数据库中的数据、如何实现不同数据库中数据表的链接规则等等,从而使数据库的分布更加合理。实际实施时,可以对每个独立场点建立一个数据库,同一场点也可以建立多个数据库。各个部门处理本场点的数据,中央数据库则存放所有业务数据,对数据进行完整性和一致性的检查,并在不同场点存储同一数据的多个副本,从而提高系统的可靠性和可用性以及局部应用的效率。这些物理上分散在异地的数据库系统,其逻辑上是一个整体,它们共同组成了完整集成的分布式数据库系统。这样实现的分布式数据库的优点是:由于数据存放在经常使用的地点,减少了请求响应的时间,也减少了网络通信代价。而且分布式数据库中数据副本的存在,客观上起到了增加数据可靠性的作用。

分布式数据库的数据存储机制。如上所述,在分布式数据库系统中,数据存放在不同的物理计算机上,根据应用的需求,不同的数据表可以存放在不同的场所,同一个数据表也可以存放在不同的场所,因此数据的访问处理需要利用数据分片技术。在中央数据库与各独立场点(各分公司、职能部门)的数据关系中,由于各独立场点的业务数据是公司业务数据的子集,可以按照一定的

方法将数据表中的记录分成多个集合,每个集合为一个片段,系统将这些记录片段合理地分配在不同的存储场所,在应用时通过并运算得到完整的数据关系。在中央数据库的数据关系中,数据是按照其业务流程来划分的,因此可以按照数据表的列把它分为若干子集,并将这些子集存储在不同计算机的存储介质上,也就是采用垂直分片的方式,应用时可以通过连接的方法利用分片恢复完整的关系。整个分布式数据库的设计如图 2 所示。

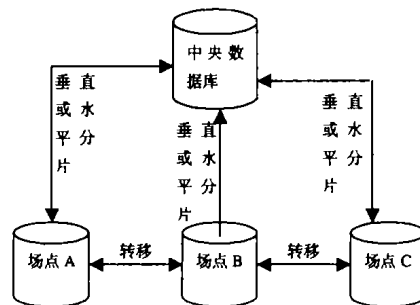


图2 分布式数据库的结构形式

由图2可以看出,由于大部分的应用是面向局部数据库的操作,为了减少网络通信量,根据应用的功能已将数据关系进行分片存放在各数据库中,但是全局性的查询仍需要多个数据库的数据支持。

分布式数据库的数据访问策略。采用事务驱动式数据库访问策略。借助于关系数据库中事务处理的概念,我们将用户输入的简单查询处理条件作为一个事务。系统根据用户的查询条件,构造成任务数据库中的事务任务,提交到数据库,获得处理结果。这种数据访问方式可以使得用户在拥有条件的情况下,增加可选择字段的自由度,实现对数据库的更高级访问。因此,其操作界面简单友好、易于使用。管理员可以添加新的事务,编辑现有的事务,删除不需要的事务任务,并灵活地为访库任务设置访库参数,也能实现对多种文件数据库格式的访问。

分布式数据库有利于整个物流信息系统性能的提升。a. 分布式数据库有利于系统执行效率的提高。由于本地数据库的数据规模小于集中式数据库,因此本地数据库上的各种事务处理将会运行得更快,网络活动也显著降低。b. 减少了对硬件性能的需求。由于单个子系统用户的减少,因此支持这种系统的硬件也可以相对要求较低。c. 可以提高系统的可靠性与可用性。由于数据和应用分布在不同场点上,当某一场点发生故障时,其它场点仍然处于运行状态并且是可访问的。d. 通过分布式数据库中数据副本的存在,也大幅提升了物流信息系统的灾难恢复能力。

参考文献

- 1 张翔,周明全,耿国华. 软件复用与基于 Java 的 COM 组件实现. 计算机应用与软件, 2003; (7)
- 2 康瑞华,尹帆,薛胜军. 基于 B/S 模式和分布式数据库技术的物流信息系统. 武汉理工大学学报, 2003; (12)
- 3 吴洪庆,段成华. 一种物流信息系统体系结构的探讨. 计算机应用研究, 2002; (8)
- 4 幸小敏,张李义. 构建一体化物流信息系统. 情报杂志, 2004; (2)
- 5 唐友三. 2003 年中国物流信息化回顾. 物流与供应链, 2004; (1~2)
- 6 厉小军,朱鸿斌,胡上序. 基于 B/S 结构的第三方物流系统设计与实现. 计算机工程, 2003; (1)
- 7 杜献峰. 组件及其复用技术研究. 电脑开发与应用, 2004; (5)
- 8 曾益,熊志勇,郭顺生. 可重构第三方物流信息系统的实现. 计算机工程与应用, 2004; (5)

(责编:梅亦愚)