

三种零售企业数据通信传输方法比较

马越, 罗汉文

(上海交通大学 电子信息与电气工程学院, 上海 200240)

摘要: 伴随着零售行业的迅速发展,数据的种类和复杂程度不断增加,尤其单个数据文件之间的数据量差别极大.如何能够在成本固定的基础上,进行准确、实时、高效的数据传输是一个非常重要及现实的问题.针对大量商业数据文件的有效传输问题,对现有的3种数据传输方式进行了分析、对比,结合企业数据通信系统功能等方面的要求,最终比较得出一种能够满足企业日常业务发展需求,并且具有良好扩展性的数据通信传输方法.

关键词: 零售业; 数据传输; Webspere MQ

中图分类号: TN 919.3 **文献标志码:** A **文章编号:** 1000-5137(2016)02-0142-08

0 引言

随着市场零售行业的迅速发展,企业数据的种类和复杂程度也在不断增加,尤其单个数据文件之间的数据量也存在很大的差异.因此,企业级数据传输对用于数据传输的软件往往有着以下几点要求:

(1) 软件质量要求高.企业要求软件不仅具有高运行效率、高正确性等运行性能,亦注重软件的非运行性能,可变更性、可移植性、可重用性.

(2) 软件结构复杂.文件传输发送端的地理分布范围广,且发送端和接收端的软件运行环境涉及大量异构的网络和硬件,企业要求软件必须分布运行,并且能够方便移植,此外还要求服务器端协调好接收文件与数据处理两个环节,结果使软件结构越来越复杂.

(3) 涉及大量底层技术.文件传输及数据处理软件涉及到网络传输、数据持久化、分布式事务处理、负载均衡、容错等等底层技术.

据了解,目前国内外大部分企业在数据传输的过程中主要采用 Database Link(DBLink)或文件传输的方式,不会使用到中间件产品.而对于部分企业,他们则选择借助中间件产品进行数据传输,WebSphere(WMQ)是其中较为典型的一款中间件产品.

本文作者将模拟日常企业数据通信的模式,对比上述3种数据传输模式,验证 WMQ 在日常企业数据传输中的优越性.

1 Database Link(DBLink)传输原理

1.1 DBLink 简介

DBLink 指通过连接到其他数据库系统的表或者视图,实现 Oracle 对 Oracle 数据库的同步.当需要

收稿日期: 2016-03-03

通信作者: 罗汉文,中国上海市闵行区东川路 800 号,上海交通大学电子信息与电气工程学院,邮编:200240, E-mail:hwluo@sjtu.edu.cn

跨本地数据库访问另外一个数据库表中的数据时,本地数据库中就可以创建远程数据库的 DBLink,通过 DBLink,本地数据库可以像访问自身一样访问远程数据库表中的数据,优点是建立同步操作方便快捷,既可以保证数据的自动更新,又可以保证数据库间的数据保持实时一致,实现两个业务系统数据的有效集成^[1-2].

假设有数据库服务器 A 和 B,这两个数据库服务器要做数据同步,如果不使用 DBLink,只能先登录到其中一台服务器上,将数据导出后再将数据导入到另一台服务器上^[3].

1.2 DBLink 数据同步步骤

DBLink 与数据库中的视图 (view) 类似,建立 DBLink 的时候需要知道待读取数据库的 IP 地址,服务集标识符以及数据库账号^[4].通过 DBLink 建立数据库间的数据同步,需完成如图 1 所示的几个步骤.

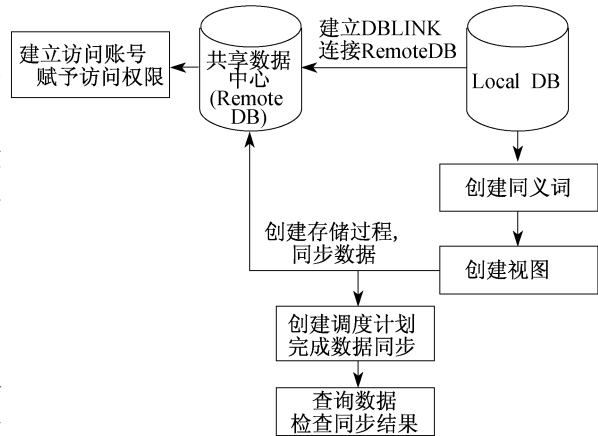


图 1 DBLink 数据同步流程图

2 文件传输协议 (FTP) 传输原理

FTP 是使用最为普遍的网络协议之一,由最早的简单文件传输协议 (TFTP) 发展而来,经过不断的修正与完善,已成为网络间文件传输的重要协议^[5].

从网络结构上讲,FTP 协议属于 C/S (客户/服务器) 结构,由 FTP 服务器与 FTP 客户机两部分组成,根据文件传输协议的规范,客户机向服务器发出请求,服务器根据请求作出相应的应答。“请求”以命令行的形式发送;根据不同的“请求”,服务器作出相应的“应答”响应.

一个典型的 FTP 会话使用两个 TCP 连接来完成文件传输操作.这两个 TCP 连接分别为控制连接和数据连接^[6].FTP 服务器在端口 (协议口 21) 执行被动打开,并等待客户连接.相应地,FTP 客户机也在这一端口与 FTP 服务器联系,在程序协调下完成一个 TCP 连接.控制连接在整个 FTP 处理过程中保持主动状态.客户机和服务器通过控制连接交换 NVTASCII 命令串及应答码.FTP 建立一个分立的数据连接来进行文件传送操作^[7].

图 2 显示了一个 FTP 操作的典型配置.

此操作的核心环节是协议解释 (PI) 和数据传输过程 (DTP).客户机和服务器有它们各自的协议解释和数据传输过程.数据传输过程负责建立和管理数据连接.协议解释部分负责解释命令并通过客户 PI 在 FTP 会话一开始建立的控制连接进行通信^[8].

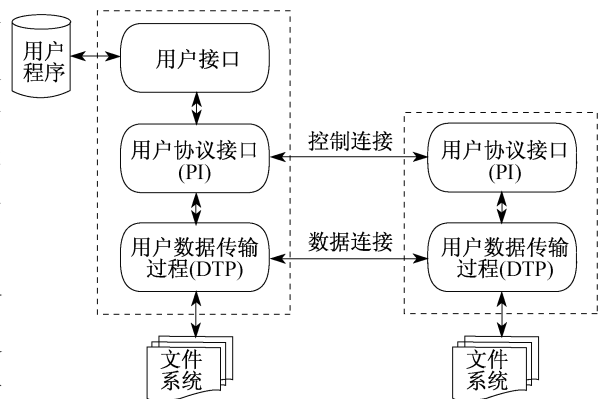


图 2 FTP 操作的典型配置

首先,FTP 客户像其他 TCP 连接一样与一个 FTP 服务器建立连接.客户和服务器在一端口 (协议口 21) 建立连接.FTP 服务器接收连接并传送 220 号应答码:220 Service ready.

在 FTP 客户接收到一个 220 应答码之后,FTP 登录过程开始,通常 FTP 服务器主机要求 FTP 客户发送一个用户名及口令字.然后 FTP 服务器对每个发送返回一个应答信息.典型的登录过程中的 FTP 命令和服务器应答交换为:USER username;331

User name ok;need password k;PASS secret k;230 User logged in.

登录之后,用户可以发送 FTP 命令以改变其在 FTP 服务器主机上的工作目录.用户在找到要检索的文件之后,FTP 要求客户和服务器建立第二个 TCP 连接以在文件传送中传送数据.若 FTP 客户要求下载服务器上的文件,则要传送一个 FTP PORT 命令,告知服务器应与它的哪一个协议端口连接,然后,客户传送一个 RETR 命令来启动文件传送操作.服务器则显示如下相应信息:RETR filename k;150 File status okay,about to open data connection.

如果文件传送成功,服务器会传送一个 226 号应答码为:226 Closing data connection,file transfer successful. FTP 客户向服务器传送文件的过程与此相类似.不同的客户传送的是 STOR 命令而不是 RETR 命令.对于一个匿名 FTP 登录来说,通常服务器不接受来自客户的文件,这时,客户和服务器之间的交互将为:STOR filename k;550 Access denied.

当用户决定结束 FTP 会话时,客户可以发送退出(QUIT)命令,且服务器将做出如下应答:QUIT k;221 Good-bye.

3 WMQ 传输原理

WMQ 是 IBM 的商业通讯中间件.WMQ 提供一个具有工业标准、安全、可靠的消息传输系统.它的功能是控制和管理一个集成的商业应用,使得组成这个商业应用的多个分支程序(模块)之间通过传递消息完成整个工作流程.WMQ 基本由一个消息传输系统和一个应用程序接口组成,其资源是消息和队列(Messaging and Queuing).有关 WMQ 的几个基本概念如下:

消息:消息就是一个信息单元,这个信息单元可以是一个请求(Request message),也可以是一个应答(Reply message),或者是一个报告(Report message)或一份报文(Datagram message).一个消息包含两个因素——消息描述(用于定义诸如消息传输目标等)和数据消息(如应用程序数据或数据库查询等).

队列:一个安全的存储消息的地方,消息的存储一般是顺序的,队列使消息分阶段地传送和接收.因为消息存放在队列中,所以应用程序可以以不同的速度,在不同的时间,在不同的地点相互独立地运行.

消息传输系统:用于确保队列之间的消息提供,包括网络中不同系统上远程队列之间的消息提供,并保证网络故障或关闭后的恢复.

应用程序接口:应用程序和消息系统之间通过 WMQ API 实现的接口.WMQ API 在所有 WMQ 平台上是一致的.API 只有 14 个调用,2 个关键词:发送(PUT)和接收(GET)^[9].

如图 3 所示,虽然应用程序 A 和应用程序 B 运行于同一系统 A,它们不需要直接的通讯.应用程序 A 向队列 Q1 发送一条消息,而当应用程序 B 需要时就可以得到该消息.如果消息传输的目标改为在系统 B 上的应用程序 C,这种变化不会对应用程序 A 产生影响,应用程序 A 向队列 Q2 发送一条消息,系统 A 的 WMQ 发现 Q2 实际上在系统 B,它将消息放到本地的一个特殊队列——传输队列(Transmission Queue).系统 A 的 WMQ 随后建立一条到系统 B 通讯联接,传递这条消息到系统 B,并等待确认.只有 WMQ 接到系统 B 成功地收到消息的确认后,才从传输队列中移走消息.如果通讯线路不通,或系统 B 不在运行,消息会留在传输队列中,直到被成功地传送到目的地.这是 WMQ 最基本而最重要的技术——可靠消息传输.

事实上,WMQ 具有特殊的技术防止消息重复传送,确保消息一次且仅一次(once-and-only-once)传递.

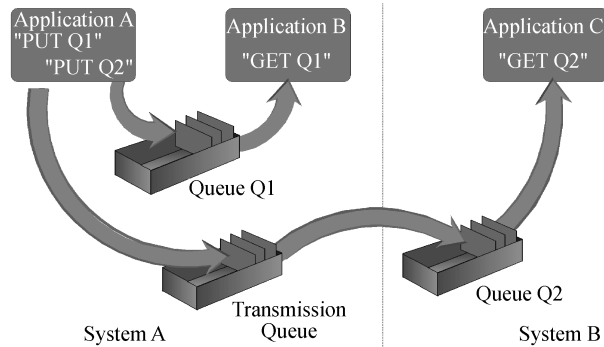


图 3 WMQ 消息传输示意

4 3 种传输方式的比较

4.1 准确性

DBLink:在数据传输的过程中,如果有涉及到 DDL 语句,整个用户级别的操作,或者某些特殊规则,就需要自定义大量的触发器来实现,而这些操作的效率和准确性都是不确定的,需要花大量时间去验证.

FTP:不考虑文件自身的内容,一味使用 ASCII 模式传输数据是不合理的.虽然现在许多 Linux 和 Windows 客户端已经支持自动传输模式,但多达数代的 UNIX 和 Windows 客户端都默认使用 ASCII 传输模式,这种传输模式可能会造成文件损坏^[10].

WMQ:WMQ 包含逻辑工作单元支持(logical unit of work),备份和恢复机制,大信息传递和高性能等特点.其中最重要的是确保信息传输,意思是一旦 WMQ 接受一个信息传输的任务,会确保信息被传送到目标平台.信息的传输是一次且仅一次.另外,强健的中间件机制保证业务数据一致性,并可在系统发生故障时,及时恢复,业务不会受到影响.

分布式队列管理通过通道两端的同步点协调来确保消息被正确地传输.如果协调过程发现了错误,将关闭通道,把消息安全地存放在传输队列中,直到通道被重新启动.当通道启动时,同步过程可以恢复可疑(in-doubt)状态.只有意外的情况,才会出现通道可疑的情况,一般情况通道能自动解决可疑状态.

4.2 实时性

DBLink:针对 DBLink 的数据库间传输效率问题,对一张具有数据记录数约 5 万条且含有大字段 CLOB 的表进行数据拷贝. DBLink 复制表的时间约为 14 s.

测试过程中会面临一个问题,当利用 DBLink 直接查询表的时候,若表中含有 CLOB 大字段,查询会报错:“ORA-22992:无法使用从远程表选择的 LOB 定位器”,但可用该语句创建表.若同步数据量多的话,oracle 不具备断点续传功能,如果对方数据库出问题(undo 表空间不足、临时表空间不足、其他错误等)、网络问题的话,则需要重新同步所有数据,这就涉及到了效率的问题.

FTP:传统 FTP 采用单线程上传文件,在网络带宽有限的条件下,对带宽的抢占优势小,导致上传速度慢;如果上传文件容量巨大,使用传统单线程 FTP 上传方式将耗时更长;而且在传输过程中,可能会发生意外断电或网络连接中断等不可预知的情况,导致文件重新上传的操作频繁,极大影响了数据共享的效率^[11].

此外,从 FTP 服务器上检索一个文件,包含繁复的交换握手步骤,需要经过三次握手来建立一

条 TCP Socket 连接. 传输一个文件, FTP 需要往复 10 次, 这也会对传输效率造成一定的影响^[12].

WMQ: 针对测试 flow “StdTransfer_Test”, 使用了包含各种大小 (KB 级别和 MB 级别) 跟类型 (xml, pdf, txt 等) 的测试数据.

表 1 反映了 flow “StdTransfer_Test” 在 2016. 01. 29 当天每个小时内传输的数据量. 该 flow 在当日每小时中最大负载的极限数据处理量为 67840 条数据.

表 1 WMQ 中测试 flow 数据传输速度

flow	date	start	end	messages	messagevol/MB	execution time/sec
StdTransfer_Test	2016. 01. 29	00:00:16	01:00:14	377	9,089	31,789
StdTransfer_Test	2016. 01. 29	01:00:14	02:00:14	457	13,483	43,195
StdTransfer_Test	2016. 01. 29	02:00:14	03:00:14	420	9,728	38,626
StdTransfer_Test	2016. 01. 29	03:00:14	04:00:17	647	12,393	52,010
StdTransfer_Test	2016. 01. 29	04:00:17	05:00:03	1,541	28,714	107,107
StdTransfer_Test	2016. 01. 29	05:01:10	06:00:17	580	12,478	59,392
StdTransfer_Test	2016. 01. 29	06:00:17	07:00:17	288	8,786	33,519
StdTransfer_Test	2016. 01. 29	07:00:17	08:00:18	579	12,203	63,514
StdTransfer_Test	2016. 01. 29	08:00:18	09:00:17	580	14,320	56,200
StdTransfer_Test	2016. 01. 29	09:00:17	10:00:19	45,328	93,101	3,057,137
StdTransfer_Test	2016. 01. 29	10:00:19	11:00:18	54,214	106,436	3,576,429
StdTransfer_Test	2016. 01. 29	11:00:18	12:00:19	54,947	108,703	3,577,674
StdTransfer_Test	2016. 01. 29	12:00:19	12:40:00	37,872	74,568	2,366,238
StdTransfer_Test	2016. 01. 29	12:41:05	13:40:12	55,225	110,161	3,526,372
StdTransfer_Test	2016. 01. 29	13:40:12	14:40:13	44,754	88,699	3,582,493
StdTransfer_Test	2016. 01. 29	14:40:13	15:40:12	48,232	96,193	3,579,581
StdTransfer_Test	2016. 01. 29	15:40:12	16:40:12	59,347	126,358	3,578,613
StdTransfer_Test	2016. 01. 29	17:55:15	18:51:34	61,302	505,006	3,329,884
StdTransfer_Test	2016. 01. 29	18:51:34	19:51:33	65,252	918,186	3,575,728
StdTransfer_Test	2016. 01. 29	19:51:33	20:51:33	67,840	636,340	3,480,803
StdTransfer_Test	2016. 01. 29	20:51:33	21:51:31	315	45,490	29,016
StdTransfer_Test	2016. 01. 29	21:51:31	22:51:31	80	46,612	11,484
StdTransfer_Test	2016. 01. 29	22:51:31	23:51:32	75	14,265	9,493
StdTransfer_Test	2016. 01. 29	23:51:32	00:51:32	457	26,468	38,532

而针对单条数据而言, 测试表明数据从源系统发送到目标系统时, 在 WMQ 中的处理时间极短, 几乎没有延迟:

由图 4 可以看出, WMQ 对单条数据的处理速度在微秒级别, 即数据在到达 WMQ 后, 几乎同时就可以被发送往目标系统, 延迟极小.

INSERT_DATE	CURRENT_EVENT_ID	FLOW_NAME	SUB_ID	CORRELATION_ID
1 12-4月 -16 01.41.00.232312000 上午 +02:00	6045376380 BRKP.I219.01__MCC_CN_X_X_4_StdDispatcher_MPOS	started		flow_d88a7772-00^
2 12-4月 -16 01.41.00.257455000 上午 +02:00	6045376380 BRKP.I219.01__MCC_CN_X_X_4_StdDispatcher_MPOS	routed		flow_d88a7772-00
3 12-4月 -16 01.41.00.263235000 上午 +02:00	6045376380 BRKP.I219.01__MCC_CN_X_X_3_StdTXInvoiceUpload	started		disp_d88e4e56-00
4 12-4月 -16 01.41.00.278459000 上午 +02:00	6045376380 BRKP.I219.01__MCC_CN_X_X_3_StdTXInvoiceUpload	info		GenericMappingV4
5 12-4月 -16 01.41.00.281006000 上午 +02:00	6045376380 BRKP.I219.01__MCC_CN_X_X_3_StdTXInvoiceUpload	receipt_status		disp_d88e4e56-00
6 12-4月 -16 01.41.00.322188000 上午 +02:00	6045376380 BRKP.I219.01__MCC_CN_X_X_3_StdTXInvoiceUpload	info		XSLT done

图 4 WMQ 单条数据发送效率示意

4.3 安全性

DBLink: 这里所说的数据库安全性指针对非授权用户的数据库保护措施. 安全性问题涉及的层面很多, 非法用户可以从以下各层面对数据库进行窃取或篡改:

(1) 数据库系统层面: 非法用户可以通过获取高级用户数据库权限的手段对数据库进行窃取或篡改;

(2) 操作系统层面: 非法用户可以通过获取操作系统 root 权限的手段, 对数据库(包括但不限于数据库)进行窃取或篡改;

(3) 网络层面: 非法用户可以通过网络窃听和身份伪装窃取线上机密数据;

(4) 物理层面: 非法用户可以从物理上进行数据窃取和破坏存储设备.

一般 DBLink 的数据库安全性管理从 3 个角度入手: 服务器操作系统(OS) 登陆身份验证; 数据库管理系统(DBMS) 登陆身份验证; 数据库(DBMS) 权限管理.

权限管理属于自定义式(Discretionary)的访问控制, 还有一种机制是强制(Mandatory)访问控制(从安全模型、数据对象标签等角度), 一般数据库具有以下权限: 读数据权限; 插入数据权限; 修改数据权限; 删除数据权限; 创建删除索引权限; 关系创建权限; 关系修改权限; 关系删除权限…….

前文提及的视图算是一种权限管理机制(用户访问符合授权的视图), 除此之外用授权视图来表示权限管理等级也是常用方法(用户和数据库管理员(DBA)均参与).

还有一个常见的数据库安全管理措施叫做审计跟踪(Audit Trail), 即 DBMS 在特殊文件或数据库中记录每个用户的具体操作, 那么 DBA 可以查看记录是否越权等,

FTP: 在互联网早期, 文件传输协议(FTP)并没有对密码安全做出规定. 在 FTP 客户端和服务端, 数据以明文的形式传输, 任何对通讯路径上的路由具有控制能力的人, 都可以通过嗅探获取用户的密码和数据. 当然, 可以使用安全套接层(SSL)封装 FTP, 但 FTP 是通过建立多次链接进行数据传输的, 即使保护了密码安全, 也很难保护数据传输的安全性.

FTP 诞生在网络地址转换(NAT)和防火墙之前, 那时的网络还不存在恶意攻击. 当今大多数最终用户的 IPv4 地址已不可路由, 这是因为防火墙的使用和 IPv4 地址的短缺. 如果 FTP 客户端 IP 地址不可路由, 或者位于防火墙之后, 那么就只能使用被动传输模式进行数据传输. 如果服务器端的 IP 地址也不可路由, 或者位于防火墙之后, 则 FTP 将无法进行数据传输^[13].

现在, 许多防火墙适用于 NAT 环境, 可以使用一些特殊的技巧(hacks)允许 FTP 在防火墙之后正常工作. 当然, 这需要对防火墙进行配置^[14].

WMQ: WMQ 队列管理器传递的信息可能都是有价值的信息, 因此需要使用授权系统来保证未授权的用户无法访问定义的队列管理器. 所以 WMQ 提供以下安全性控制: 管理 WMQ 的权限, 可以定义一些用户通过命令来管理 WMQ. 使用 WMQ 对象的权限, 可以定义哪些用户(通常是应用程序)可以使用 MQI 调用和 PCF 命令执行以下操作: 连接到队列管理器; 访问类似队列、名字列表和进程的对象, 访问这些对象的方式; 访问 WMQ 消息; 访问与消息相关联的上下文信息. 通道安全性: 需要确保用于将消息发送到远程系统的通道可以访问必需的资源, 还需要确保该通道被已授权的用户操纵.

4.4 可扩展性

DBLink: 如果要将几个 oracle 数据库里的数据集中到一个综合的数据库中, 需要综合的数据库有 SQLSERVER 以及 ORACLE, 但是两者的字符集又有所区别. 如果仅使用 DBLink 的话那么就需要手动更新; 此外, 数据库的版本也可能会影响到数据传输能力, 比如“ORA-00600”错误, 就是由于数据库版本匹配问题造成的.

FTP:基于FTP协议的简化文件传输组件具有一定的可移植性,但现有的FTP客户端软件大多独立运行,不能很好地嵌入到既有系统之中,且存在无法定制功能的缺陷^[15].

WMQ:WMQ作为标准的商业中间件,企业可以借助WMQ很容易地在现有或遗留系统之上或之外增加新的功能模块或商务构件,并将它们与原有系统无缝集合.

此外,从经济及时间成本的角度分析,WMQ中间件还具有以下优势:

应用开发:The Standish Group分析了一百个关键应用系统中的业务逻辑程序、应用逻辑程序及基础程序所占的比例,发现业务逻辑程序、应用逻辑程序仅占总程序量的30%,而基础程序却占了70%.若是以新一代的中间件系列产品来组合应用,同时配合以可复用的商务对象构件,则应用开发费用可节省至80%.

系统运行:没有使用中间件的应用系统,其初期投入的资金及运行费用要比同规模的使用中间件的应用系统多一倍.

开发周期:时间限制是所有应用系统开发项目的天敌,而基础软件的开发又是一件极耗时的工作,若使用标准商业中间件则可缩短开发周期50%~75%.

合理运用资金:借助标准的商业中间件,企业可以很容易地在现有或遗留系统之上或之外增加新的功能模块,并将它们与原有系统无缝集合.

系统维护:每年维护自我开发的基础(中间件)软件的开支是当初开发费用的15%~25%,每年应用程序的维护开支则仅需要当初项目总费用的10%~20%.

质量:基于企业自我建造的基础(中间件)软件平台上的应用系统,每增加一个新的模块,就要相应地在基础(中间件)软件之上进行改进.The Standish Group在调研过程中,曾在某个企业中的一个应用系统里,发现了有多达17000多个模块接口,而标准的中间件在接口方面都是清晰和规范的,可以有效地保证应用系统质量及减少新旧系统维护开支^[9].

5 结 论

就DBLink和FTP这两类数据传输方式而言,由于传输软件开发的抽象层次不高,开发人员更多地精力集中在建设底层设施、保证传输的正确性和提高系统的执行效率上.软件系统的开发效率低下,软件系统的可重用性、可扩展性不高,不能适应用户对高质量软件系统的要求.

WMQ则以一致的、可靠的和易于管理的方式来连接应用程序,并为跨部门、企业范围的集成提供了可靠的基础.它为重要的消息和事务提供可靠的、一次且仅一次的传递,同时也可以处理复杂的通信协议,并动态地将消息传递工作负载分配给可用的资源.

综上所述,WMQ由于集成了DBLink与FTP的功能,在准确性、实时性、安全性上具有良好的表现,且具有强大的可扩展性,并有效地降低了时间和经济成本、提升了项目开发的成功率,是零售企业用于数据通信传输的最佳工具.

参考文献:

- [1] Chen S J, Chen X J, Zhang P. Research and implementation of data synchronization based on DBLink [J]. Computer programming skills and maintenance, 2015(8): 71-72.
- [2] Zheng Y J. Application integration research in digital campus [J]. Journal of Nanningpolytechnic, 2010, 15(2): 96-100.

- [3] Shi Z Y, Feng Q Q. Application of heterogeneous data integration in the digital campus [J]. Journal of Air Force Radar Academy, 2010, 24(1) : 54 - 57.
- [4] Zhang J. Analysis and design on college heterogeneous data integration [J]. Science & Technology Economy Market, 2010 (7) : 3 - 5.
- [5] GrayCat. FTP Must Die [EB/OL]. (2016 - 2 - 15) [2016 - 3 - 1]. <http://mywiki.woledge.org/FtpMustDie>.
- [6] Gong L, Mao X J. Implementation of FTP client based on C# [J]. Computer Programming Skills and Maintenance, 2010 (24) : 94 - 95.
- [7] Li L. Design and implementation of distributed FTP system [J]. Journal of Liaoning Technical University, 2005, 24(2) : 232 - 234.
- [8] Yu Y S. Web common transceiver system development based on FTP protocol [J]. Modern Computer, 2009, 26(9) : 183 - 185.
- [9] Gan Q. IBM WebSphere MQ basic course [M]. Beijing: Publishing House of Electronics Industry, 2004.
- [10] Zhang G X, Chen H. Research of automatically updates for cross-platform systems based on FTP and message mechanism [J]. Computer security, 2010, 31(9) : 16 - 18.
- [11] Wu K, Lu B L, Zhang L. Implementation of client software based on FTP protocol [J]. Journal of Shenyang Institute of Aeronautical Engineering, 2006, 23(4) : 39 - 42.
- [12] Qu T. Data communications and network infrastructure [M]. Wuhan: Huazhong University of Science and Technology Press, 1996.
- [13] Zheng M. Classic examples for C + + Builder and Windows API [M]. Beijing: Tsinghua University Press, 2002.
- [14] Lei Z J, Ma Y X. Computer network [M]. Xi'an: Xi'an Electronic Science & Technology University Press, 1996.
- [15] KrisJamsa D, Wright C. C# programming tips and techniques [M]. Osborne: McGraw-Hill, 2001.

Comparison of three retail data communication and transmission methods

MA Yue, LUO Hanwen

(School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract: With the rapid development of retail trade, the type and complexity of data keep increasing, and single data file size has a great difference between each other. How to realize an accurate, real-time and efficient data transmission based on a fixed cost is an important problem. Regarding the problem of effective transmission for business data files, this article implements analysis and comparison on 3 existing data transmission methods, considering the requirements on aspects like function in enterprise data communication system, we get a conclusion that which method can both meet the enterprise daily business development requirement better and have good extension ability.

Key words: WebSphere MQ; retail trade; data transmission

(责任编辑:包震宇)