

J2EE 集群的持久对象缓存同步的研究

曾志常¹, 杨文伟^{1,2}, 李锦棠¹

(1. 广东工业大学 计算机学院, 广东 广州 510006;

2. 广东省教育厅, 广东 广州 510631)

摘要:对利用组播进行 J2EE 集群服务器持久对象的同步原理进行介绍, 讨论在现行 J2EE 集群系统中的实现情况; 分析需要编写通信中间件进行同步 J2EE 集群的持久缓存的情形及以非组播方式实现的方法; 最后给出了以 JMS 实现集群持久缓存同步的案例。

关键词:J2EE 集群; 持久对象; 缓存; 同步; JMS; 组播

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2006)11-0143-03

Research on Synchronization of Cache for Persistent Object in J2EE Cluster

ZENG Zhi-chang¹, YANG Wen-wei^{1,2}, LI Jin-tang¹

(1. Faculty of Computer, Guangdong University of Technology, Guangzhou 510006, China;

2. Department of Education of Guangdong Province, Guangzhou 510631, China)

Abstract: Introduces the principle of synchronization cache for persistent object in J2EE cluster by multicast, the realization in present is also discussed in detail. Then analyses the cases of requirement of developing communication-oriented middleware to synchronize the persistent object caches and the realization means except multicast. Finally presents a case which realizes the synchronization by JMS.

Key words: J2EE cluster; persistent object; cache; synchronization; JMS; multicast

0 引言

随着 Web 应用的迅猛发展, J2EE 集群服务器以其可扩展性、高可靠性, 为 Web 应用系统提供了新的实现方案。对于 J2EE 集群系统前端的负载均衡、Http Session 的容错和恢复等, 业界许多 J2EE 平台已提供了实现方式。在前端 Web 应用服务得到良好解决后, 后台持久层数据的存取往往成为 J2EE 系统的瓶颈, 为此, 许多 J2EE 服务器对读取频繁的非关键数据采用缓存技术, 以减少应用程序读取持久数据的时间。然而, 由于各节点的持久缓存运行在各自的 JVM 环境中, 当单个节点对全局性的持久数据进行事务性更改后, 需对各缓存同步, 以保持一致性。目前通常的同步方式是通过组播实现, 但由于受网络环境、集群方式等多方面影响, 采用组播同步缓存需具备一定的条件才能得以进行。为此, 文中对组播进行同步 J2EE 集群的持久缓存进行分析, 并研究指出 JMS 也是一种实现缓存同步的良好选择。

1 集群的组播原理及其现行实现缓存同步的分析

1.1 集群的组播原理

组播, 即多播, 集群节点间以组播方式通信时, 数据包的目的地址是以 1110 开头的 32 位 D 类 IP 地址, 代表一组目标主机。集群单个节点发送数据到其他节点时, 不论集群成员数量的多少, 数据源只发送一次数据包, 就可向那些需要数据包的主机发送数据, 而集群中所有的节点就会收到该组播地址上的每个分组拷贝。

由于主机可随时加入或退出群组, 非集群成员也可以向集群发送组播数据包, 因而采取组播通信的集群可扩展性强, 集群规模越大, 组播的优点越能体现出来。当然, 集群采取组播方式进行通信时, 也带来了安全性和管理性等问题; 而生成组播 MAC 地址时, 由于组播 IP 的 28 位有效地址位不能全部映射到只有 23 位的 MAC 地址空间中, 致使一个 MAC 地址表示 32 个 IP 组播地址, 如图 1 所示。这种不明确性造成网络中有多个组播组时, 接收主机需要检查收到数据包的 IP 部分是否为其所需的组播组, 因而对 CPU 的性能要求较高。

1.2 现行组播实现持久缓存同步的分析

许多 J2EE 应用服务器都提供了集群的功能, 通过组播的方式, 实现集群节点间的通信。EJB 是 J2EE 的核心技术, 而持久层的数据缓存可在实体 EJB 中实现, 支持集

收稿日期: 2006-02-26

作者简介:曾志常(1977-), 男, 广东清远人, 硕士研究生, 研究方向为计算机网络与分布式系统; 杨文伟, 副教授, 研究方向为网络与分布式系统。

群 EJB 的服务器也不再需要考虑持久缓存的同步问题。然而, EJB 全面考虑了分布式事务、安全权限控制、多线程访问、数据缓存等多方面的服务, 这导致 EJB 的灵活性最终以性能上的牺牲为代价^[1], 因此实体 EJB 的持久缓存功能受到一定的限制。而适合实体 EJB 的应用领域比较狭窄, 设计、开发和部署实体 EJB 也较为复杂, 致使实践中实体 EJB 并没有得到广泛的应用, 持久层的数据缓存则更多地 Web 服务器或其他 O/R 映射框架中实现。

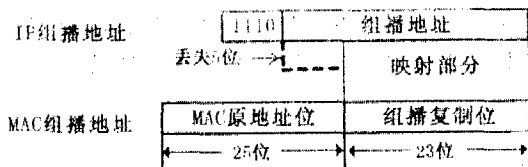


图 1 IP 组播地址到 MAC 地址的映射

当持久层数据缓存放在前端 Web 应用服务器中实现时, 因集群 Web 机制不同而对持久缓存的同步产生一定的限制或问题: Tomcat 集群时, 进行多节点复制, 所有节点都一样, 为集群系统提供了高容错性, 因而利用其 Web 集群通信机制实现的持久缓存的同步会一致, 但因为大量的网络通信, 对节点数量将有所限制, 很易成为集群系统的瓶颈, 扩展性差^[2], 实际中很少使用。主流产品 Weblogic, Websphere 采取节点成对复制方式^[3], 为 Web 集群系统提供了高性能、高扩展性的平台。采取这种方式同步持久缓存, 当单个节点对持久数据改变后, 只有和该节点成对的另外一个节点的持久数据得到同步, 而其他节点的则得不到同步。

JBoss Cache 是一个支持集群系统内分布式事务的复制式 (Replication) 缓存, 缓存 3 个 Mbeans (其中一个用于采用缓存机制的实体 bean^[2]) 以树状结构提供服务, 方便使用、同步和管理; JBoss 官方资料表示, JBoss Cache 可作为一个独立的插件存在而嵌入到其他 J2EE 服务器中使用, 且具有面向方面编程 (AOP) 的优点。然而, 虽然开源软件 JBoss Cache 实现了分布式数据缓存, 并且不会像 Weblogic 那样实现集群时需要另外购买昂贵的 License, 又可以嵌入到非 JBoss 的集群 J2EE 平台中, 持久缓存的同步问题也在软件中得以实现, 但到目前为止, JBoss 官方资料只提供了 JBoss Cache 嵌入到 Weblogic 的一个测试资料, 要真正广泛而稳定地应用于非 JBoss 的集群 J2EE 平台, 还需较长一段时间。另外, 虽然 JBoss Cache 具有良好的功能, 但需要依赖通信组件 JGroups, 并且 JGroups 和 JBoss Cache 的配置和使用较为复杂, 一般需要向 IBM 购买较为昂贵的服务。

2 编写通信中间件同步集群持久缓存的情形

对于集群系统, 如果其持久层的缓存同步已在产品中实现, 如 Weblogic 的集群 EJB, 或者持久缓存本身已提供集群功能, 如 JBoss Cache, 则不存在同步持久缓存的问题。而对于以下情况, 则需另行增加通信件来同步集群服

务器的持久缓存。

1) 产品缓存并不支持集群的功能, 如现行热门的 Hibernate, 其二级缓存 EH Cache 还无法做到分布式缓存。

2) 持久层缓存不是 J2EE 平台提供的, 而是自定义开发实现的, 而在开发时只考虑到单服务器的环境。

3) 集群系统只是单纯的负载均衡, 各节点互为独立, 需手工进行或编写单独的通信插件进行同步节点间的持久缓存。

4) 系统没有使用实体 EJB 或集群 EJB, 没有开启缓存而把缓存放置在 Web 服务器中实现。

当需要增加中间件来同步集群的持久缓存时, 其实现可采取通常的组播方法来进行。然而, 在以下情况, 需要考虑其他方式来同步持久缓存。

(1) 虽然 J2EE 集群平台有提供组播通信的方式, 但只是对平台厂商自身的产品进行通信, 而没有对开发者提供使用的接口; 或者, 集群系统或网络不提供组播机制, 那就需要其他的同步方式同步缓存。

(2) 有些网络结构不支持组播。如当采取 IP 组播时, 若集群内的各节点处于不同的 IP 网段而节点间的路由器没有开启组播功能。

(3) 虽然支持组播, 但考虑到组播所产生的消耗将会超过采取缓存的性能改善。如组播间隔的时间太短, 将导致集群节点间发送和接收的数据量太多, 甚至会导致网络中的数据阻塞; 而映射组播 MAC 地址的问题, 将可能增加接收节点对组播信息的处理。

(4) 要求集群系统能较易部署、升级和维护, 或对系统的稳定性、健壮性较高。这对于许多开源中间件, 以上要求是难以满足的。如作为非集成性产品的 JGroups, 在部署或应用时, 需清楚组播的底层网络技术和 JGroups 的各项策略及多个参数的意义。而根据文中对各个缓存软件相关文档的查阅, 一些开源集群缓存软件, 如 Swarm Cache 和 JBoss Cache, 当集群规模较大时, 都不同程度地存在阻塞或死锁的可能。

(5) 组播技术带来负面的影响, 如组播病毒的传播。

3 RMI 与 JMS 同步集群持久缓存的分析

RMI (Remote Method Invoke) 远程方法调用, 封装了底层的分布式处理细节, 提供了透明的远程对象方法调用机制, 较之 Socket 而易于使用, 可以完整无缺地传送对象。但 RMI 采取点对点的基于请求/应答的同步通信模式, 不支持于多个节点之间进行交互, 对网络状况要求很高, 对延时也非常敏感。客户机调用服务器上某个分布对象的一个方法, 在方法调用返回之前, 该客户机被阻塞, 直到方法调用结束才能执行下一条指令^[4]。在集群节点数量不大的情况下, 这对发送缓存同步信息的节点的服务性能影响不会很大; 但当集群规模较大时, 需要考虑采取这种方式同步缓存所带来的影响, 尤其是若只是静态地同步集群节点缓存, 而集群的结构不稳定或故障节点较多时,

发送节点因需等待到超出故障节点的 RMI 服务时间后才能继续下面服务而影响服务性能。

在构造分布式系统特别是松耦合系统的时候,消息中间件(MOM)是一种很常见的构建模块。在采用 MOM 时,松耦合的组件通过异步交换消息的方式进行通信,为独立的组件提供可靠的消息传递服务。MOM 的异步特性使它适合于将高度自治性作为设计目标的情况,若同步缓存采取这种通讯机制,可使集群节点发出缓存的同步信息后,不必等待其他节点的接收确认信息而可以继续服务。Java 消息服务(JMS)是 MOM 的 Java 接口规范,它给出了使用 Java 语言在 MOM 中创建、发送、接收、订阅消息的统一编程方式,并定义了一套消息通信模型。JMS 消息的类型可以是串行化的 Java 对象,因而,对于同步集群持久缓存采用节点间复制的方式,JMS 也提供了良好的支持。JMS 提供两种最普遍的消息通信模型:点对点(Point-To-Point)和发布/订阅(Publish-Subscribe)模型。对于集群系统来说,进行同步持久缓存有多个接收者,因此 Pub/Sub 的方式更为适合。而 Message-Driven-Bean 则兼备了 JMS 和 EJB 的功能,在消息生产者与消费者分离的同时,简化了创建 JMS 消息消费者的过程,并且支持事务性的消息处理。当同步持久缓存采取这种方式时,除有 JMS 的特点外,还可以把持久缓存的同步作为一个事务来处理,这对于复杂的持久缓存来说尤为重要。

4 JMS 实现集群持久缓存同步的案例

广东省考试中心的普通高考、成人高考等多个报名和录取应用系统均部署在有 6 个节点的 Weblogic 集群服务器中,后台为独立磁盘冗余阵列支持的单个 Sybase 数据库服务器。由于各系统的应用时效性很强,如普通高考报名,须在向社会公开的时限内完成 50 多万考生的报名,在持久数据的读取本来就成为瓶颈的情况下,Web 应用系统要求对数据库读取的频率就更为限制。因此,缓存技术在各系统中的采用尤为必要。虽然 EJB 中的 CMP 可以通过制定乐观并发锁定等策略来配合启用持久缓存技术^[5],但考虑到实体 EJB 的性能问题,因此各系统开发的过程中就避免采用实体 EJB,而以 HashMap,ArrayList 等容器直接实现高效的持久缓存。然而,由于集群系统只是简单的作为负载均衡,节点间 Weblogic 实例各自独立,因此,原来当某一节点对持久数据发生改变后,产生缓存不一致的问题。现行通过 JMS 技术,较好地解决了以上问题,同步缓存的协作图如图 2 所示。

集群节点既是同步持久缓存的信息发布者,也是信息的订阅者,而 JMS 服务器则部署在 Weblogic 的 Domain Administrator Server 上。当集群任一节点发生事务操作

而导致 Sybase 的持久数据发生改变后,节点发送同步信息到 JMS 服务器,而集群所有节点获知更新信息来读取数据库数据达到同步目的。由于集群的策略要求各节点间尽量独立,同步缓存采取通知的方式,从数据库读取数据来更新缓存。采用 JMS 同步缓存首先在普通高考报名系统中实现,并通过简单的修改而移植到其他系统中。由于 JMS 不是一种实现方案,而是将 API 嵌入带有适应性的消息系统的一种规范,在工业界已得到广泛的实现,因此,采用 JMS 开发的同步集群的持久缓存插件将有良好的移植性^[6],是同步缓存的一个良好方案。

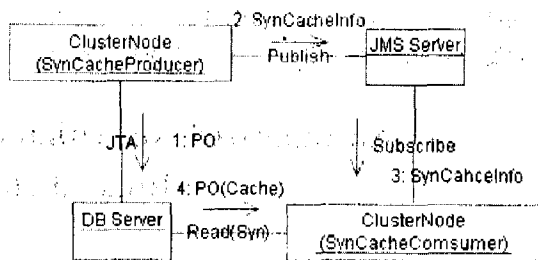


图 2 JMS 同步集群持久缓存协作图

5 结束语

对目前常用的组播同步集群持久缓存的原理进行介绍,指出现行 Web 服务器实现集群持久缓存的实现情况;并分析其他需要增加通信中间件来同步持久缓存的情形,对于不适宜采用组播同步缓存而采取其他同步方式的可行性,也作了详细的研究;最后给以广东省考试中心集群服务器的持久缓存同步的实现案例,验证了除组播方式外,JMS 也是实现集群持久缓存同步的一个良好选择。

参考文献:

- [1] 朱岸青,王会进,高河福.影响 EJB 性能的若干因素及其优化方法[J].计算机应用,2004,24(6):181-184.
- [2] TheServerSide. Uncover the hood of J2EE Clustering[EB/OL]. 2005-08. <http://www.theserverside.com/articles/article.tss?l=J2EEClustering>.
- [3] JBoss. JBossCache and JGroups Services[EB/OL]. 2005-10. <http://docs.jboss.org/JBossas/JBoss4guide/r4/html/JBossCache.chapt.html>.
- [4] 白琳. JMS 与 RMI 技术在数据库监测系统中应用研究[J]. 计算机应用研究,2005(12):208-210.
- [5] BEA Systems. Managing Entity Bean Pooling and Caching[EB/OL]. 2005-12. <http://e-docs.bea.com/wls/docs81/cjb/entity.html#1194003>.
- [6] 李中良. 分布式环境下事件自动处理机制的实现[J]. 计算机工程,2005,31:92-94.

欢迎订阅,邮发代号 52-127。