第33卷 第15期 计 算 机 工 程 2007年8月
Vol.33 No.15 Computer Engineering August 2007
·博士论文· 文章编号：1000—3428(2007)15—0043—03 文献标识码：A 中图分类号：TP391

基于对等模型的分布式协同设计系统

陈冬芳[1]，薛继伟[1]，王 征[2]
（1. 大庆石油学院计算机与信息技术学院，大庆 163318；2. 电子科技大学计算机学院，成都 610054）

摘 要：为了解决分布式协同设计系统中的信息快速检索以及多副本同步等问题，引入了对等模型，给出了该系统的功能模型、信息检索模型等。提出了DHT的对等信息检索方法，保证了用户能够在分布式协同设计系统中快速共享资源。采用基于DHT的分布式互斥等算法作为协同数据的一致性维护方法，给出了系统的具体实现方法和实例。

关键词：分布式协同设计；对等；分布式哈希表

Distributed and Collaborative Design Systems Based on P2P Model

CHEN Dong-fang[1], XUE Ji-wei[1], WANG Zheng[2]
(1. School of Computer and Information Technology, Daqing Petroleum Institute, Daqing 163318;
2. Computer Science and Engineering College, University of Electronic Science and Technology, Chengdu 610054)

【Abstract】 To deal with problems such as fast information retrieval, multi-replica synchronization and so on, a P2P model is presented for distributed collaborative design systems. Some function models, information retrieval models and so on are proposed to construct the system. A P2P information retrieval method based on DHT and interest clustering is used to ensure the fast retrieval of users. And a distributed mutual exclusion algorithm based on DHT is given as the method to synchronize the collaborative data. And the implement method and an instance system are given.

【Key words】 distributed and collaborative design; P2P; DHT

1 概述

目前，计算机辅助设计工具的分布式集成传统的CAD/DFX 软件工具基本都是面向单机单用户的[1,2]。这些工具有广泛的用户群和应用基础，如何整合这些资源是推行分布式协同设计研究的热点。实现这些工具的分布集成就是对它们进行封装，并向它们提供与外部世界的通信接口。目前，分布式协同设计系统采用的主要方法有2种：(1)直接封装，使用CORBA、DCOM 以及基于Java 的软件技术将这些工具中对外开放的服务封装为分布式对象，供异地用户进行远程调用；(2)间接封装，即基于Agent 的封装，通过Agent 实现CAD/DFX 工具之间的通信与交流，具有较好的适用性，便于实现不同CAD/DFX 工具之间的互操作和信息交流。实现CAD/DFX 工具分布式集成的关键在于工程共享信息表达的一致性和不同CAD 系统的API 函数的一致性。

计算机支持的分布式协同设计(以下简称为分布式协同设计)方式下[2~4]，分布在不同地点的产品设计人员通过网络采用计算机辅助工具协同地进行产品设计活动。目前分布式协同设计系统，如Liao等的ASP模式的协同设计系统[2]等，其处理共享信息的主要方法是：将其设置为“共享”，用户手动在服务器间切换，并在所登录的服务器上查找/访问。这种方法效率低下、缺乏统一的接口，而且无法保证分布式环境下的多副本的一致性[5]。

目前，基于对等模型P2P(peer to peer)的系统被广泛地研究应用[2,3]，例如：Chord, PASTA等。P2P系统用对等互助模式替代C/S(client/server)模式，系统中的节点存储信息也充当信息检索中的资源路由，从而能够利用系统中各个节点中的资源。本文针对协同分布式设计系统的特点，提出了基于对等模型的分布式协同设计系统，解决传统信息共享系统的诸多问题。

2 系统模型

2.1 对等系统模型

基于对等模型的分布式协同设计系统的节点模型主要模块有(见图1)：

(1)资源浏览器(resource browser)/客户接口（client service interface）/服务接口（server service interface）：提供接口/工具给协同设计系统中的用户/子系统。其中，登录P2P 节点的用户通过资源浏览器检索/获取系统中的可用信息资源，如CAD 模型等；CSI 提供本地/远程调用接口给客户端程序，该接口通过向CAD 等软件进行二次开发，使其集成到分布式协同设计系统中，对于异地编辑等操作，CAD 软件发送的命令及编辑信息将通过；管理员通过SSI 配置检索/资源路由器的初始化路由等信息。

(2)检索器/资源路由是节点模型的核心部分：它主要处理经过哈希化的信息（节点号；信息的 Key 标识等）；同时，它负责对收集到的信息进行分类，协调其它模块的工作；本地共享信息通过分布式哈希表DHT(distributed hash table)获得对应的关键字Key，由资源路由将其Key标识扩散到同一关键字空间中的相关节点；异地共享信息的Key通过资源路由分类存储在本节点中。

基金项目：黑龙江省自然科学基金资助项目(A0312)
作者简介：陈冬芳(1972－)，男，博士研究生，主研方向：动力学仿真；薛继伟，博士、副教授；王 征，博士
收稿日期：2007-04-20 **E-mail：**dongfongc@163.com

(3)路由表/叶子集/邻居集：每个节点在进入系统时都通过 DHT 分配了一个 128 位的节点号；本地节点获得的异地节点号/IP 地址存入 3 个集合中：通常，叶子集保存了节点号最接近（通常是同一广播域内的节点）的信息；路由表/邻居集保存了其它节点信息；本节点模型根据分布式协同设计中信息聚类分布/访问的特点。对上述 3 个集合作了按用户访问兴趣/资源特征聚类的改进，具体方法见下文。

(4)对等节点管理/对等通信：前者控制本地节点主动将节点号/IP 地址扩散至同广播域的节点，同时也通过广播获得其它节点的信息；对于不在同一个广播域的节点，本地节点需要系统管理员配置或从其它节点获得；同时该模块还负责分布式多副本一致性的维护工作。后者发送消息、建立信息传输通道；完成信息检索、资源路由、文件共享等一系列工作。

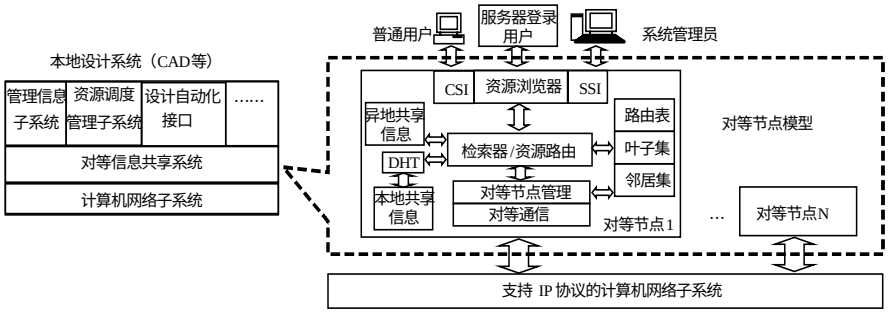

图 1　对等分布式协同设计系统节点模型

2.2　信息共享系统运行流程

该系统的运行是一个 3 段流程，包括：(1)信息的搜集/发布，主要将共享信息的元数据、用户兴趣、聚类特征等检索信息在系统中收集/规范化/扩散；(2)预处理，主要是关键词提取、索引化/聚类操作等；(3)服务，包括检索、获取信息、多副本同步等。下文以一个设计图的获取流程说明该系统的运行流程：

Step1 在单个节点上生成设计图，将该图设置为“对等共享”或者保存在对等共享的文件夹中；本地节点的信息共享系统自动提取该图的元数据(文件名、类型等属性)；由 DHT 运算生成该图的 Key 值，封装<Key,URL>二元组，其中 URL 是本地节点号；检索器/资源路由在将其保存入本地共享信息的同时，也将其扩散至叶子集中的各个节点，以及节点号和该 Key 接近的节点中。

Step2 节点接收到其它节点扩散的<Key,URL>二元组，将其按照类型保存在异地共享信息中，完成信息的搜集/发布及信息收集。

Step3 系统中某节点发起协同请求，遍历设计表单，发现需要该设计图时，同样通过该文件的元数据信息（文件名等）由 DHT 生成 Key 值；首先查找异地共享信息，如果有 Key 值相等的表项，则可以直接从 URL 表征的节点中获取该图；如果没有，进一步向节点号最接近 Key 值的节点发送检索消息<Key, URL>。

Step4 当某节点接收到检索消息后，检查本/异地共享信息中是否存在接收到的 Key，如果有返回查询结果给检索节点；否则继续前递(Forward)消息给叶子集/路由表中的关键字空间距离最近的节点。注意，为了限制搜索深度，消息中带有 TTL(Time to Live)标识：每次路由该消息时，将 TTL 减 1；当 TTL=0 时，清除该消息。

Step5 当检索设计图的节点接收到检索应答消息时，一方面更新异地共享信息，插入<Key,URL>项，必要时也修改路由表/邻居集；另一方面该节点的对等通信模块自动从设计图所在节点获取该图的副本。

Step6 当该图的多个副本扩散在系统中，多个用户对该图进行写操作（插入、删除等），则副本所在节点通过节点管理模块进行分布式互斥操作，保证多副本的一致性。

Step7 若该图的副本不可扩散，多个异地用户对该图进行写操作时，拥有该图的节点自动建立本地的互斥队列；根据请求所携带的时间戳，对接收到的异地编辑请求进行排队，保证数据的一致性。

3　分布式数据一致性

分布式协同设计等应用环境需要进行数据一致性操作；这也是分布式协同设计系统的一个难点。通过在信息共享系统中引入 P2P，使该问题变得容易解决。

如上文所述，本系统采用了两种分布式互斥算法进行数据一致性维护；这两种算法均采用 Lamport 逻辑时戳进行优先级判断，该逻辑时戳的描述如下：

(1)每个节点i中保持一个时钟C_i。

(2)当系统发送请求时，它将时钟加 1，请求消息的格式为(m, T_i, i)，其中，m是请求内容；T_i是该请求的时戳；i是本节点标志。

(3)当节点j接收到请求时，作如下操作：$C_j = 1+\max(C_j, T_j)$。对于同一节点接到的两个不同的消息a和b；如若$T_a<T_b$，或者$T_a=T_b$且$a<b$，则认为a在b之前。

3.1　异地编辑一致性

当多个节点通过分布式协同设计系统编辑某一文件时候，可能引发共享数据的不一致问题。为此，本系统采用了 DHT 化的优先队列进行异地编辑一致性操作，其算法描述如下：

Step1 当多个节点需要编辑某一节点中的设计对象时候，首先发送“请求”消息给设计对象所在节点，该消息携带<Key, URL, Pos>三元组和 Lamport 逻辑时戳，其中，Pos 是异地节点需要编辑的位置标志（如一条管道的拐点等等），该元素同样通过 DHT 进行哈希化处理。

Step2 当设计对象所在节点接到多个节点的编辑请求时，根据请求所携带的 Pos 元素建立多条请求队列（Key 和 Pos 元素相同的信息入同一条队列），队列中根据 Lamport 逻辑时戳进行排列，执行 FIFO 操作（先进先出 First In First Out）。

Step3 设计对象所在节点根据请求队列中出队列的异地请求，完成一定的编辑操作，并返回“应答”消息给请求节点。

3.2　副本编辑一致性

分布式多副本同步算法的实质是分布式互斥，用以保证分布式系统的一致性，即确保每次只有一个节点进入临界区，其关键在于如何选择决策节点；传统分布式互斥的算法很多，例如 Lamport 的时间戳算法、Maekawa 的请求集算法。本系统中将传统的“锁”机制扩展为“对等锁”，解决一致性问题，“对等锁”算法描述如下，如图 2。

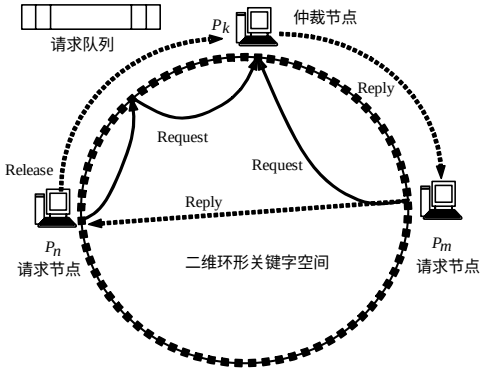

图 2　基于 DHT 的分布式互斥示例

Step1 需要进入临界区的节点P_m通过DHT路由基发送“请求”消息给主本所在节点P_0；系统中可能已经存在多个副本，部分副本

可能由其他副本复制生成，这些副本不能视为主本；因此，本算法中节点将“请求”消息发送给主本所在节点。

Step2 P_0收到多个“请求”，P_0根据逻辑时戳对这些消息进行排序，存入请求队列；主P_0给优先级最高的节点P_m发送“应答”消息，允许它进入临界区，P_0即被点P_m锁住；接到“释放”消息前，P_0不得发送“应答”给其他节点。

Step3 优先级最高的节点P_m退出临界区后，不需再次查找，直接发送“释放”消息给P_0。

Step4 主本所在节点P_0接到“释放”消息时，释放锁，删除消息队列中P_n产生锁的记录，并发送“应答”给下一个节点；重复上述Step2、Step3，直到请求队列为空。

Step5 副本所在节点进行操作的同时，进行数据一致性刷新，将编辑命令发送给其他节点，由这些节点完成一致性操作。

传统副本同步算法需要在系统中广播/组播大量的消息；即使在节点全互联的网络中，Lamport算法也具有$O(N)$的消息复杂度，Maekawa算法具有$O(\log_2 N)$ 消息复杂度；在多跳网络中，这些算法的消息复杂度还将上升；而在本系统中，为获取一次进入临界区的机会，节点最多将传递$\log_2{}^b N$个消息用于查找主本所在节点；返回“应答”和“释放”，最多各需要$\log_2{}^b N$个消息(全互联网络仅需 1 个消息)，因此，其消息复杂度为$O(\log_2{}^b N)$，因为b大于 1，所以“对等锁”算法的消息复杂度远低于其他算法。

4 系统性能测试

本系统在某企业的虚拟专用网络 VLAN（virtual LAN）中得以实现 ;该系统由 4 个局域网组成,局域网间通过 Internet 互联，单个局域网由 6~20 台服务器组成，节点间可以通过 IP 进行直接通信 ;服务器安装有 Windows 2000 Server 及 Red Hat Linux 9 等操作系统。为适应异构环境，对等信息共享系统提供了若干操作系统的版本，但实现机理是一致的。

该系统在实际应用中得到了应证。图 3 显示了在本系统与该企业最初使用的国外某型系统中进行“排水管道 247.DWG”对象进行搜索和同步的平均时间延迟。

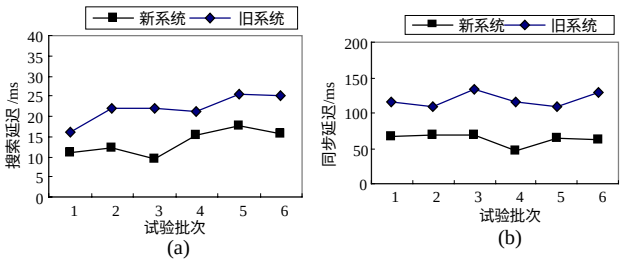

图 3　新旧系统性能对比

从图 3 可以看出，本系统由于采用了对等模型，大大缩短了共享信息的搜索时间和数据一致性的维护时间，从而提高了系统整体的性能。

5 结论

综上所述，基于对等模型的分布式协同设计系统，保证了用户能够在分布式协同设计系统中快速共享资源；采用基于 DHT 的分布式互斥等算法作为协同数据的一致性维护方法。同时，本文给出了系统的具体实现方法和实例与验证。该系统已经在生产单位得到了应用，生产实践证明该系统的性能良好，运行稳定，取得了良好的经济效益。

参考文献

1 冯相忠, 高　禹. 基于多 Agent 技术的分布式协同设计结构的研究[J]. 计算机应用, 2006, 26(9): 2182-2183.

2 黎水平, 望　超. 基于 P2P 的分布式协同设计模式研究[J]. 矿山机械, 2006, 34(2): 99-101.

3 吕建明, 刘　悦, 丁　林, 等. P2P 与信息检索[J]. 信息技术快报, 2005, 3(2): 1-12.

4 高曙明, 何发智. 分布式协同设计技术综述[J]. 计算机辅助设计与图形学学报, 2004, 16(2): 149-157.

5 应　华, 李凯里. 三维 CAD 分布式协同设计的方案研究[J]. 机械设计与研究, 2006, 22(6): 77-80.

（上接第 19 页）

求一个元素的逆需要较大的运算量，因此，方案 1 是 3 个方案中最优的。

表 1　3 个方案的效率比较

阶段 / 方案	签名	验证
方案 1	5E+5M	2E+1M
方案 2	5E+5M+1I	2E+1M
方案 3	5E+5M+1I	2E+1M

2.5 抗篡改协定信息攻击

篡改协定信息攻击是指用户在获得签名人的签名后，试图将事先协定的信息 c 篡改为 c^*，同时保证签名的有效性。本文的方案使用了 Hash 函数，因此可以抵抗这种攻击，考虑 2.1 节的一般方案，若 (r,s,c) 是一个有效的部分盲签名，假设攻击者企图把 c 篡改为 c^* 后使签名仍然有效，则有：

$$rg^s = y^{r+h(m,c)} \bmod p$$

$$rg^s = y^{r+h(m,c^*)} \bmod p$$

因此，攻击者必须求出 c^*，使得 $h(m,c)=h(m,c^*)$，这是很困难的，因为 h 是密码学 Hash 函数。

3 结论

在部分盲签名中，签名人在盲签名时可以加入自己的信息或者和用户协商的信息，消息提供者在得到签名后不能对签名人加入的信息篡改，这比盲签名更加实用，部分盲签名可以看作是盲签名的扩展，盲签名可以看成是没有协定信息的部分盲签名，是部分盲签名的一个特例。本文完整地解决了 Harn 签名的部分盲化问题，得到了 3 个部分盲签名方案，并分析了提出的方案满足部分盲签名的所有性质。

参考文献

1 Chaum D. Blind Signatures for Untraceable Payments[C]//Proceedings of Crypto’82. Prenum Publishing Corporation, 1982: 199-204.

2 Abe M, Fujisaki E. How to Date Blind Signatures[C]//Proceedings of Asiacrypt’96. Berlin: Springer-Verlag, 1996: 244-251.

3 Abe M, Okamoto T. Provably Secure Partially Blind Signatures[C]// Proceedings of Crypto’00. Berlin: Springer-Verlag, 2000: 271-286.

4 Zhang F, Safavi-Naini R, Susilo W. Efficient Verifiably Encrypted Signature and Partially Blind Signature from Bilinear Pairings[C]// Proceedings of INDOCRYPT’03. Berlin: Springer-Verlag, 2003: 191-204.

5 Harn L. New Digital Signature Scheme Based on Logarithm[J]. Electron. Lett., 1994, 30(5): 396-398.

6 黄振杰, 林宣治, 周豫萍. 新的基于离散对数盲签名方案[J]. 漳州师范学院院报, 2004, 17(3): 21-24.