

基于 JXTA 的 Super-peer 搜索方法设计

李歆海 李善平

(浙江大学人工智能研究所 杭州310027)

Design JXTA Based Super-peer Search Method

LI Xin-Hai LI Shan-Ping

(Department of Computer Science and Engineering, Zhejiang University, Hangzhou 310027)

Abstract Efficient resource search method has significant impact on the scalability and availability of P2P network. Generally there are two search methods, pure Peer-to-Peer method and central index method. Recently, some search methods with super-peer concept are appearing, which are the compromise of those two methods and have favorable scalability and availability.

In this paper, we compare the advantage and deficiency of these three kinds of search methods, and based on JXTA platform design the super-peer search method.

Keywords Super-peer search, Peer-to-Peer, JXTA

1 介绍

Peer-to-Peer 网络由具有对等地位的主机组成,资源(包括主机计算能力,存储空间,服务,内容)以非集中方式存在于各个对等体(Peer)内,使得资源具有很高的可用性。但是,对等体计算能力和网络带宽的差异性,给有效发现资源从而利

用资源造成了障碍。因此,设计高效,低代价的资源搜索方法成为开发 P2P 网络应用潜力的重要问题。

P2P 搜索的三种方法:纯 Peer-to-Peer 方法,集中索引方法和 super-peer 方法,是以对等体在搜索中承担任务的多少划分的,图1显示了它们的这种不同。

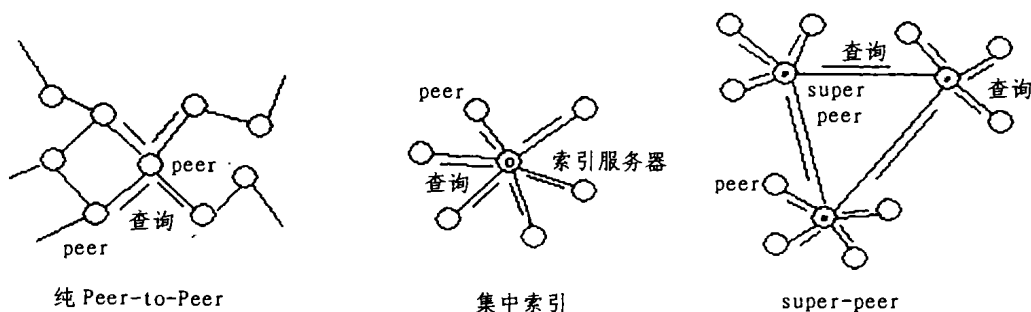


图1 P2P 搜索网络

在纯 Peer-to-Peer 方法中,如 Gnutella, Freenet, 每一个对等体具有完全相同的作用和责任,自一个对等体发起的查询可在整个 P2P 网络中传播,每一个收到查询的对等体有责任向发起者作出回应。在集中索引方法中,如 Napster, 索引服务器存储来自客户对等体的资源信息并处理查询。在 super-peer 方法中,如 Morpheus, super-peer 的作用类似于集中索引方法中的索引服务器,但对等体只向多个可获得的 super-peer 中的一个发送资源更新信息及查询。Super-peer 间是对等的关系,共同完成查询任务。

由于缺少公共平台,在目前的 P2P 应用的开发和使用中,开发者必须重复创建完成基本功能的系统原语,用户为参加不同的 P2P 网络必须提供不同的软件,并为这些软件的运行提供包括操作系统、网络连接等方面的支持。JXTA 作为一种高度开放的 P2P 系统平台,具有高度互操作性,平台独立性,及可实现于各种设备的特点,使在其上实现 P2P 搜索成为合理的选择。

2 搜索方法比较

在纯 Peer-to-Peer 方法中,所有的对等体都是查询的发起者和回应者。它的优点体现在最大的可用性,即单一对等体的失败,仅使自身的资源不被其他对等体利用,而不会造成别的影响。其缺点表现在扩展性的不足,即当对等体数量增加时,每一对等体须接收、处理和转发的查询数量急剧增加,因此那些计算能力或网络带宽很低的对等体,将造成 P2P 网络的总体响应时间和资源利用率恶化。

在集中索引方法中,索引服务器缓存来自客户对等体的资源信息,并根据这些缓存的信息处理查询,客户对等体不负担查询任务。它的优点体现在查询的高效率,且网络上传递的查询和回应消息数量少。它的缺点是索引服务器只能为数目有限的客户提供服务,因此扩展性不足,并且索引服务器的单点失败将使整个搜索网络失败,这就要求索引服务器必须有很高的可用性。

李歆海 研究生,主要研究方向:计算机网络应用。李善平 教授,博士生导师。

在 super-peer 方式中,客户对等体在某一时刻仅与一个 super-peer 连接,并向其发送自己的资源情况和查询。super-peer 缓存客户资源信息,并执行查询。由于查询任务在 super-peer 间得到分担,因此搜索可用性提高。而且查询消息仅在 super-peer 间传播,可大量减少查询消息数量,因此搜索扩展性提高。但是相比前两种方法,super-peer 方法的算法更为复杂,难点包括如何控制 super-peer 数量,如何有效地在 super-peer 间传播及处理查询和回应等。

随着 P2P 用户数目的快速增加,对查询的扩展性和可用性提出了很高的要求,super-peer 搜索方法适应了这种要求。并且,super-peer 的两种特殊形式,单一 super-peer(仅一个对等体作 super-peer)和全 super-peer(所有对等体均是 super-peer),分别对应集中索引方法和纯 Peer-to-Peer 方法,使其也能满足某些特殊应用的需要。

由上所述,super-peer 方法是最合理的搜索选择。

3 Super-peer 搜索算法分析及设计原则

基于性能高的对等体发生网络连接断开和系统崩溃的可能性较低的假设,采用以下的分析而设计的算法能使搜索网络逐步自组形成稳定状态,性能高的对等体承担更多的搜索任务,且搜索效率在总体上得到提高。算法设计主要涉及以下方面。

作为客户对等体还是 super-peer 一个高性能的 super-peer 处于低负载状态,不利于提高总体性能。但是让所有对等体都尝试发现并连接高性能 super-peer,其代价太高。为解决这两个问题,本算法采用以下设计原则:正与一个 super-peer 保持连接的客户对等体不尝试发现并连接别的 super-peer。super-peer 总是尝试发现并连接处于低负载且性能高于自己的 super-peer,然后转为客户身份运行,并与其原有客户断开。处于高负载的 super-peer 要求它的一个或一些客户与自己断开。当客户对等体在与 super-peer 发生断开时或刚进入 P2P 网络时,立即尝试连接低负载且性能高于自己 super-peer,若不成功则以 super-peer 身份运行。

信息的缓存 当客户对等体拥有的资源很多时,若每次与 super-peer 连接都上传全部资源信息,会对系统性能造成不利。通过让对等体缓存一些信息可较好地解决这个问题。这些信息包括:本地的资源状态,对等体作为客户时最近连接过的几个 super-peer,对等体作为 super-peer 时最近收到的来自客户对等体的资源信息。基于这些信息,客户对等体总是优先连接最近连接时的 super-peer,在此被连接的 super-peer 仅需要客户发送最近发生变化的资源信息。当然,存储空间有限的对等体可选择作信息缓存。

客户对等体与 super-peer 的通信 由客户发往 super-peer 的消息主要包括:建立连接、查询、资源更新。由 super-peer 发往客户的消息主要包括:允许连接、查询回应、要求客户退出。

super-peer 间的通信 super-peer 间的通信主要是查询

和查询回应。为使一条查询不被同一个 super-peer 多次处理,并将回应返回初始查询者,本算法使用查询路径来解决。另外,可通过在 super-peer 间复制客户资源信息以缩短查询时间。本算法选择的策略是当一个 super-peer 在收到来自别的 super-peer 的查询回应时,可决定是否缓存回应中的资源信息。

4 基于 JXTA 的算法设计

4.1 JXTA 平台体系

JXTA 平台体系包含以下基本概念:标示符(Identifiers),广告(Advertisements),对等体(Peers),消息(Messages),对等体组(Peer Groups),管道(Pipes)。在 JXTA 体系中,所有实体(包括对等体、管道、对等体组等)及消息都被赋予标示符,对等体参与对等体组,并在组内发布和接收广告,广告描述了实体的各种信息。根据获得的广告中的内容,对等体使用管道与别的对等体互发消息。

对等体为了参与对等体组,发布广告,发现组内别的对等体及其提供的服务,在对等体间发送消息,需要遵守 JXTA 定义的以下协议:对等体发现协议(PDP),对等体解析器协议(PRP),对等体信息协议(PIP),对等体成员资格协议(PMP),管道绑定协议(PBP),终端点路由协议(ERP)。

PDP 使对等体能发现其他对等体的广告,进而发现对等体、对等体组、管道和服务等实体。在目前的 JXTA1.0 版中,发现机制包括:1. 基于局域网的发现,即通过在子网内的广播完成;2. 基于邀请的发现,即当对等体收到邀请时,可通过包含于邀请中的对等体信息发现别的对等体;3. 级联发现,即若一个对等体发现了另一个对等体,后者可向前者报告自己发现的对等体;4. 通过集结点的发现。集结点是一个特殊的对等体,其他对等体可通过与集结点的通信来发现别的对等体。PRP 使对等体可以完成最基础的查询功能,PIP 使对等体可获得别的对等体的信息和状态,PMP 为需要安全审核的对等体组提供成员安全证书管理,包括证书的申请、刷新和撤销。PBP 允许对等体将管道广告捆绑到管道终端点,从而在对等体间建立消息连接。ERP 允许非直接连接的两个对等体发送消息。

4.2 算法设计

对应 JXTA 体系,客户对等体和 super-peer 均为 Peer,并由 Identifier(以下简称 ID)标示,且加入同一 Peer Group。对等体利用 PRP 发现 super-peer,利用 PIP 获得别的对等体的信息广告,利用 PBP,ERP 并采用 Pipe 机制通信。

以下先介绍重要的数据类型和数据项,再描述算法。

4.2.1 关键数据类型和数据项 CachedSuperPeerList: 以与当前时间接近者优先为序保存最近连接过的 super-peer 的 ID。LocalResourceStatusList: 针对 CachedSuperPeerList 中每一个 Super-Peer 记录对等体最近一次同其断开以来自身的资源状态变化。

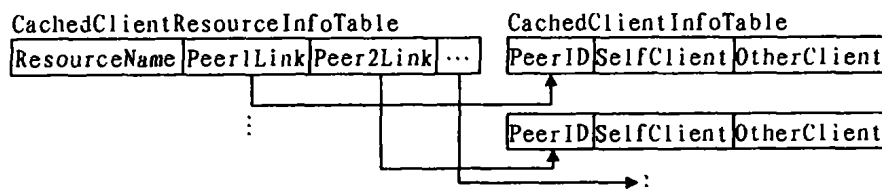


图2 缓存的客户资源信息与客户信息的关系

CachedClientResourceInfoTable, CachedClientInfoTable: 两者均以表的形式表示, 图2给出了它们的表项的关系。前者的表项记录了资源和资源提供者链接。后者的表项记录了客户对等体 ID, 并以 SelfClient 表示它为自己的客户, 以 OtherClient 表示它不是自己的客户。

PeerInfo Advertisement: 包含对等体性能信息, 若是 su-

per-peer, 还包含负载信息。

QueryMessage, ResultsMessage, QueryRoute: 如图3, 其中 TTL 控制查询传播范围; ResultsNumber 表示查询者期望的结果数; 在需要多个 super-peer 处理一条查询时, 将使用 QueryRoute 以使结果最终返回初始查询者。



图3 查询消息、结果消息

AliveMessage, AliveClientList: 客户发送 AliveMessage 消息, 以避免 super-peer 认为客户已断开。Super-peer 在 AliveClientList 中记录保持活动的客户 ID。

4.2.2 搜索算法 首先描述算法中的几个关键部份:

Super-peer 间查询及回应处理: 涉及如何在 super-peer 间传播查询及将结果返回最初查询者, 并避免同一查询被同一 super-peer 多次处理。

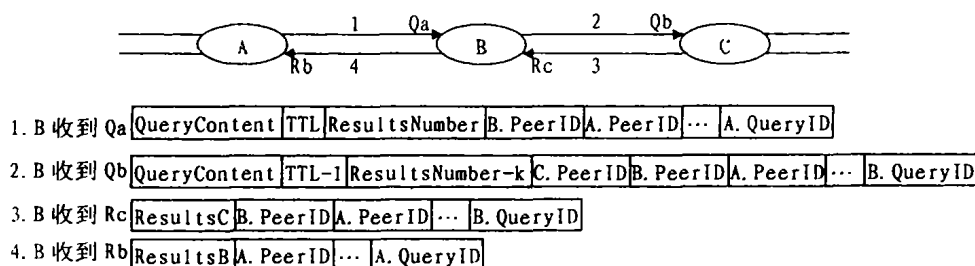


图4 super-peer 的查询及回应处理

如图4, 第一步中, super-peer B 收到查询消息 Qa 后, 将尝试在本地缓存中找到 ResultsNumber 个结果, 并最终找到了 k 个结果 (以 LocalResults 表示结果)。然后判断以下条件: $(k < \text{ResultsNumber}) \text{ AND } (\text{TTL} \neq 0) \text{ AND } (\text{能找到一个 super-peer (以 C 表示)}) \text{ AND } (\text{C.PeerID 不在 Qa.QueryRoute.PeerIDList 中})$, 若不满足转第4步处理, 若完全满足按第二步处理。

第二步中, B 生成本地唯一的查询 ID (以 B.QueryID 表示), 并在本地保存 B.QueryID, LocalResults 和 Qa 中的 A.QueryID 及 LocalResults, 然后向 C 发送查询消息 Qb。

第三步中, B 收到来自 C 的结果消息 Rc 后, 根据 Rc 中的 B.QueryID 取出保存的 A.QueryID, LocalResults, 合并本地结果和 Rc 中的结果 $\text{LocalResults} = \text{LocalResults} + \text{Rc.ResultsC}$ 。然后按第四步处理。另外, ResultsC 将按下文所述方法缓存。

第四步中, B 发送 Rb 到 A, Rb 中的 ResultsB 为 LocalResults。

客户资源信息的缓存和撤销: 资源信息及其提供者由 CachedClientResourceInfoTable 和 CachedClientInfoTable 记录。对等体只在以 super-peer 身份运行时, 才更新这两张表。

对表的操作包括: (1) UpdateSelfClient()。客户与 super-peer 保持连接时, MyClient 被设为 TRUE, OtherClient 设为 FALSE。当客户添加或删除资源时, 资源表项的相关 PeerLink 被设置或删除; (2) CleanUpSelfClient()。当 super-peer 与客户断开时, 断开客户的 MyClient 设为 FALSE; (3) UpdateOtherClient()。当按上文所述收到结果消息并缓存资源信息时, 若已有客户表项存在, 且 MyClient 为 TRUE, 则操作结束。否则, 设定 MyClient 为 FALSE, OtherClient 为 TRUE, 并加入资源信息; (4) DeleteOtherClient()。所有 Other-

Client 为 TRUE 的客户表项被删除, 且资源表项中指向它的 PeerLink 被删除。

从以上操作可见, 在 (1)(2) 操作中, 会为新客户、新资源新建表项。在 (1)(3)(4) 操作中, 若资源表项已无 PeerLink, 则将它删除。在 (4) 中, 才会清除客户表项。

针对以上操作对其他处理的影响需说明两点: 1. 当 super-peer 在本地缓存中查找资源时, 仅将 MyClient == TRUE 或 OtherClient == TRUE 的客户提供的资源列入查找范围。当然, super-peer 自己的资源也在查找范围内; 2. 在 (1) 操作中, 若尚无客户表项存在, 则要求客户报告完整资源信息, 否则只要求客户报告资源变化信息。

Super-peer 负载计算方法: 本文采用一种简单的负载评价方法, 即当 super-peer 需要计算负载时, 选择与自己连接的且性能最高的若干客户, 若自己客户的性能都很低, 则须从网络中发现若干目前与别的 super-peer 连接的性能高的客户。然后要求客户发送给定查询并记录从发出查询到收到回应的的时间间隔, super-peer 以客户报告的时间间隔值中的最小值作为负载值。判定对等体性能高低及是否处于高负载的具体值可根据应用的特定要求设定。

选择连接最佳的 super-peer: 需要以下条件同时满足:

1. 一个 super-peer 不会向别的已处于高负载的 super-peer 提出成为其客户的要求; 2. 在所有发现的 super-peer 中选择性能最高者, 且其性能高于自己。

以下根据对等体的运行过程, 简要描述算法。

对等体启动: 从 CachedSuperPeerList 的头部开始, 尝试连接未处于高负载的 super-peer。一旦成功则对等体开始以客户身份运行, 否则以 super-peer 身份运行。

客户身份运行: 立即执行的任务有: 根据最近一次以客

(下转第30页)

与主动 RSA 签名结合起来提供的问题,有入侵检测发现攻击后如何通知主动系统的问题,以及如何在被攻击者控制的机器上维护主动系统代码本身的安全性等问题。

参考文献

- Ostrovsky R, Yung M. How to withstand mobile virus attacks. In: Proc. of the 10th ACM Symposium on the Principles of Distributed Computing, 1991. 51~61
- Herzberg A, et al. Proactive Secret Sharing or: How to Cope With Perpetual Leakage? Advances in Cryptology--Crypto 95 Proceedings, LNCS, 1995, 963: 339~352
- Canetti R, et al. Proactive Security: Long-term protection against break-ins. RSA CryptoBytes, 1997, 3: 1~8
- Shamir A. How to Share a Secret. Communications of the ACM, 1979, 22: 612~613
- Gennaro R, et al. Secure Distributed Key Generation for Discrete-Log Based Cryptosystems. Eurocrypt '99, pp. 295~310
- Boneh D, Franklin M. Efficient Generation of Shared RSA Keys. Crypto 97, pp. 425~439
- Frankel Y, MacKenzie P D, Yung M. Robust Efficient Distributed RSA-Key Generation. PODC 1998. 320
- Canetti R, Halevi S, Herzberg A. Maintaining Authenticated Communication in the Presence of Break-ins. Journal of Cryptology, 2000, 13: 61~105
- Gennaro R, et al. Robust Threshold DSS Signatures. Eurocrypt '96, LNCS 1070. 354~371
- Herzberg A, et al. Proactive Public Key and Signature Systems. ACM Conference on Computers and Communication Security, 1997
- Frankel Y, Gemmell P, MacKenzie P, Yung M. Proactive RSA. Crypto'97, 1997. 440~454
- Rabin T. A Simplified Approach to Threshold and Proactive RSA. Crypto'98, LNCS vol. 1462, Springer-Verlag, 1998. 89~104
- Gennaro R, Jarecki S, Krawczyk H, Rabin T. Robust and Efficient Sharing of RSA Functions. Journal of Cryptology, 2000, 13(2): 273~300
- Frankel Y, MacKenzie P, Yung M. Adaptively-Secure Optimal-Resilience Proactive RSA. ASIACRYPT, 1999. 180~194
- Frankel Y, Gemmell P, MacKenzie P D, Yung M. Optimal-resilience proactive public-key cryptosystems. IEEE Symposium on Foundations of Computer Science, 1997. 384~393
- Frankel Y, MacKenzie P, Yung M. Adaptively-Secure Distributed Public-Key Systems. European Symposium on Algorithms, 1999. 4~27
- Frankel Y, MacKenzie P, Yung M. Adaptively-Secure Optimal-Resilience Proactive RSA. ASIACRYPT'99. 180~194
- Lysyanskaya A. Efficient Threshold and Proactive Cryptography Secure Against the Adaptive Adversary. Manuscript, 1999
- Barak B, Herzberg A, Naor D, Shai E. The Proactive Security Toolkit and Applications. ACM Conference on Computer and Communications Security 1999. 18~27
- Garay J A, Gennaro R, Jutla C, Rabin T. Secure Distributed Storage and Retrieval. Theoretical Computer Science, 2000, 243(1-2): 363~389
- Chow C-S, Herzberg A. Network Randomization Protocol: A Proactive Pseudo-Random Generator. In: Proc. of the Fifth USENIX UNIX Security Symposium, Salt Lake City, Utah, June 1995

(上接第12页)

户身份退出网络以来本地资源状态变化情况,更新 LocalResourceStatusList; 将当前连接的 super-peer 设为 CachedSuperPeerList 的头; 根据 super-peer 的要求及自身缓存, 报告资源信息(完整资源信息或仅是资源变化信息); 发布 PeerInfo Advertisement. 在需要时执行的任务有: 1. 发送查询(需在 QueryMessage 中设定初始 TTL, ResultsNumber, 自己的 PeerID, QueryID), 接收回应; 2. 发送 AliveMessage; 3. 为 super-peer 提供负载值; 4. 本地资源发生变化时, 向 super-peer 报告; 5. 被要求断开时, 转“对等体启动”。

在退出客户身份时不执行任何处理。

Super-peer 身份运行: 基本任务包括接受客户连接, 更新客户资源, 记录客户连接状态, 处理查询消息和结果消息。另外需周期性执行以下任务: 1. 根据 AliveClientList, 调用 CleanUpSelfClient; 2. 调用 DeleteOtherClient; 3. 发现并连接最佳的 super-peer, 成功时退出 super-peer 身份, 转“客户身份运行”; 4. 计算负载, 高负载时要求性能最高的客户断开, 并为其执行 CleanUpSelfClient; 5. 发布 PeerInfo Advertisement。

在退出 super-peer 身份退出前必须执行 CleanUpSelfClient 和 DeleteOtherClient。

结束语 高效的资源搜索方法对 P2P 网络扩展性和可用性的提高有着重要影响, 具有 super-peer 概念的搜索方法是提高扩展性和可用性的可行途径之一。本文基于 JXTA 技术设计了一种 super-peer 搜索方法。由于基于 JXTA, 使其能适应更广的应用环境。在设计中, 以性能高的对等体发生网络连接断开和系统崩溃的可能性较低为假设, 使用它们来共同承担搜索任务; 通过客户及 super-peer 的信息缓存, 有效减

少对等体间消息发送数量; 利用 super-peer 负载计算, 合理控制网络中 super-peer 的数目。

为使本算法更加完善, 在未来的工作中仍有一些问题需要解决, 其中主要包括如何使 super-peer 与客户的组织更符合物理网络拓扑逻辑, 如何提高 super-peer 间信息缓存复制的有效性。

参考文献

- Yang B, Garcia-Molina H. Comparing hybrid peer-to-peer systems. In: Proc. of the 27th Intl. Conf. on Very Large Databases, Sep. 2001
- Ratnasamy S, Francis P, Handley M, Karp R, Shenker S. A scalable content-addressable network. In: Proc. ACM SIGCOMM, Aug. 2001
- Yang B, Garcia-Molina H. Designing a Super-Peer Network: [Technical report]. Stanford University, 2002. Available at: <http://www-db.stanford.edu/~byang/pubs/superpeer.pdf>
- JXTA v1.0 Protocols Specification. Available at: <http://spec.jxta.org/v1.0/docbook/JXTAProtocols.html>
- Ripeanu M. Peer-to-Peer Architecture Case Study: Gnutella Network: [Technical report]. University of Chicago. Available at: <http://www.cs.uchicago.edu/files/tr-authentic/TR-2001-26.pdf>
- Saroiu S, Gummadi P, Gribble S. A measurement study of peer-to-peer file sharing systems. In: Proc. of the Multimedia Computing and Networking, January 2002
- Yang B, Garcia-Molina H. Improving efficiency of peer-to-peer search. In: Proc. of the 28th Intl. Conf. on Distributed Computing Systems, July 2002