P2P 系统资源查询机制研究综述*)

王 丹

(北京工业大学计算机学院 北京100022)

摘 要 随着计算机网络的迅速发展和个人计算机处理能力的不断提高,P2P 技术已高度重视其新的特性。但是,P2P 系统的高度动态性和资源的广泛分布性使其难以共享资源。本文讨论了 P2P 系统的资源查询机制,描述和分析了一些方法,给出了查询机制的要求和目标。

关键词 P2P,查询,DTH

The Research Overview of Resource Search Mechanisms in P2P System

WANG Dan

(College of Computer Science, Beijing University of Technology, Beijing 100022)

Abstract With the rapid development of computer network and the increasing processing ability of personal computer, Peer to Peer technology has attracted much attention for its new feature. But high dynamic features and broadly distributed resources in Peer-to-Peer system make it difficult to share resources. Resource Search Mechanism in Peer-to-Peer System is discussed. Several approaches are described and analyzed. The requirements and goal about search mechanism are given.

Keywords Peer to Peer, Search, DHT

1 引言

P2P 是 Peer-to-Peer 的简写,一般称为对等或点对点,表示用户可以直接连接到其他用户的计算机,进行文件共享与交换,从而使各网络节点既是客户,又是服务器。P2P 系统的核心任务仍然是实现资源共享,即一个对等者(Peer)在网络中提供并获得资源和服务。和客户/服务器(C/S)服务模式相比,P2P 系统中的节点之间能够直接进行交互,而不依赖于中心环节。目前个人计算机处理速度和存储设备容量不断增长,价格不断降低,P2P 技术的出现则可以使人们充分利用这些位于网络边缘的资源、并具有降低平均网络负载,消除单点故障等诸多优点。

由于 P2P 网络的高度动态性(节点可以自由加入与离开)及资源的分布性,如何提供一种高效的资源查询机制是这类系统所面临的一个问题。由于每个节点都拥有大量的可被其它节点共享的文件,资源查询请求的形式也可以是多种多样的,如可以是基于文件标识符的,或者按一定规则表达的关键字。因此,查询机制就是要考虑搜索能力、带宽消耗、安全等诸多方面的问题,并优化查询消息从一个对等者传递到另一个对等者的路径,使系统能够快速有效地找到并返回用户所需的数据,或指向数据的指针。

2 P2P 系统的查询机制

P2P 系统的查询工作过程就是从用户接收查询请求,定位文件所在的位置,返回结果(文件或指向文件的指针),之后源与目的节点直接相连进行文件下载。查询请求通常是以规则形式表述的关键字,并且可能是在文件的不同部分(例如头部、标题、图元数据)中定义的,也可能是其它形式,如有基于语义的信息。按文件索引信息的存放方式的不同,本文将现有系统中的查询机制归以下几类进行描述:集中式查询、广播式查询和分布式查询三类。

2.1 集中式查询

集中式搜索使用专门的中心索引节点,这些节点上保留了在 P2P 系统中可得到的文件索引,需要查询的节点都先连接到中心索引节点上,即先将查询请求发送给此中心索引节点,由它在本地节点的索引表中查找所需资源所在的节点,并将此节点地址返回给请求者或者返回查询失败的消息,然后请求者与此目标节点连接,实现上下载文件的操作。

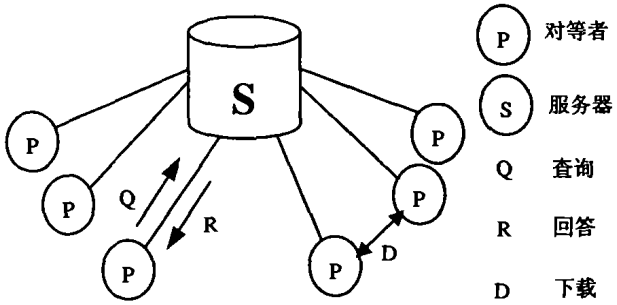

图1 集中式索引机制

以 Napster[1]系统为例,查询和响应如图1所示。①当一个节点加入系统时,就发送一份该节点上的所有文件的名单给作为中心索引节点的 Napster 服务器,它保存每个用户加入系统时提供的文件列表名单。②当中心索引节点 Napster 服务器收到用户的查询请求时,就在它保存的索引文件列表中判别是否含有相关内容的节点,如果有就将这些节点的地址名单返回给用户。一般中心索引节点会查找与查询匹配最佳

*)基金资助:北京工业大学博士启动基金资助。王 丹 博士,主研方向:分布式计算,P2P,移动代理。

的对等者(最佳对等者可能是速度最快的、代价最低的等,是由用户的需求来决定的)。③用户与最佳对等者之间直接传送文件,而不再通过中心索引节点。

在这类P2P系统中,文件索引是集中式存放的,文件的存储和服务是分布式的。此方法查询效率高,网络上传递的查询和响应消息数量较少。然而中心服务器只能为有限的客户提供服务,扩展性不足,并存在单点失效问题,且容易受到攻击。

2.2 广播式查询

在这类系统中,既没有存放文件索引信息的服务器,也没有对网络拓扑结构和文件的放置的任何要求。查询请求消息被广播到当前节点的所有邻居节点中(或者在一定的半径内),以期找到与查询对应的节点。Gnutella[2]系统是典型代表之一。在Gnutella网络中,节点中没有索引信息,共享文件没有在任何地方进行类似广告之类的发布,节点仅仅通过本地信息决定要连接到网络的哪个部分上。文件的查询机制是通过向所有的邻居主机广播带有非零TTL的查询来实现的(是查询message历经的最大Hop数),工作过程如图2所示。Gnutella使用的是一种广度优先搜索技术,每个收到查询请求的节点将向它的所有邻居节点转发请求,直到超过TTL的要求为止。因此,想要查找某个文件就必须不断地从节点到邻近节点传播查询,直到发现匹配的文件。用户可以看到一小部分其他用户的文件,其它用户再依次可以看到其它用户的文件,类似于菊花链效应。在Freenet系统中,则使用类似于深度优先搜索技术[3]。每个节点只是向它的一个单个邻居转发请求,等待一定的响应时间,如果没有得到满意的结果,再向另一个邻居转发请求。

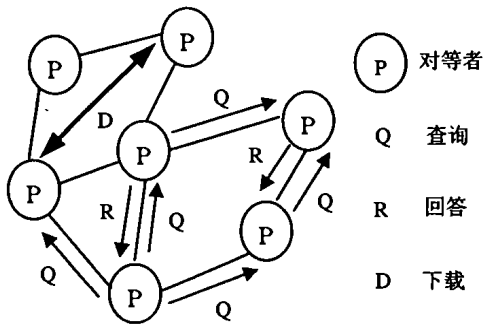

图2 无索引机制

广播式查询机制的优点是简单强健,可以对节点的动态加入和离开由一定的弹性控制,一个节点的失效不会影响查询的实现,但产生的巨大网络流量加重了网络负载,而且不容易度量。Ritter分析了Gnutella搜索协议之后得出[8]:在一个拓扑为树型、分支为8、TTL为8的Gnutella系统中,为了搜索一个18字节的字符串可能仅仅进行文件定位就会产生1.2GB的网络流量。因此不适用于大规模系统,但是在一个小型网络中,比如公司的内部网络中,这种机制还是可取的[8],因为会很快得到查询结果。对Freetnet使用的深度优先机制来讲,因为各个节点是顺序处理查询的,只要满足结果要求,此次查询就可以结束,因此,代价比较小。但是,这种顺序查询方式使得响应时间较长,最坏情况下随路径的增长呈指数级增长。

2.3 分布式查询

这种方法是在每个节点上都保存有部分资源的索引信息,这些索引表分布到网络中的各个节点上,其中的索引信息给出了文件所在的"方向",而一般不是文件所在的真实位置。下面是几种不同的实现方法。

2.3.1 基于DHT的关键字查询方法

基于DHT(Distributed Hash Table)的方法是通过分布式哈希函数,将指定的文件关键字唯一映射到某个网络节点上,然后通过某些路由算法同该节点建立连接来实现文件查询[6~8,13]。在该模型中,赋予节点一随机标识符(NodeID),并且每个节点均知道一定数量其它节点的NodeID。当资源需要在某一个点发布共享时,采用分布式Hash算法对其进行Hash运算,生成一个资源标识符,再按照一定的算法,该资源被存储到相关的节点中。节点在请求资源时,向其已知的节点发送查询请求,该对等节点则根据该资源的标识符将请求信息发送到与该标识符最接近的节点,直到找到存储资源的节点为止。

Chord[9]和CAN[10]等系统是这类模型的代表之一。Chord由MIT提出,它提供一个适合于P2P环境的分布式资源发现服务。该模型通过维护相邻节点的路由表(Routing)来进行信息资源定位和查找;其中的网络节点按照一定的方式分配一个唯一节点标识符(NodeID),资源对象通过Hash运算产生一个唯一的资源标识符(Object ID),且该资源将存储在节点ID与之相等或者相近的节点上。需要查找该资源时,通过Hash运算可定位到存储该资源的节点。通过使用分布式Hash路由表技术使得查找指定对象只需要维护$\log N$长度的路由表,但仍存在命名冲突及消息的传递效率不高及可扩展性差的问题。

CAN(Content Addressable Networks)项目是由Berkeley开发的,采用N维几何分割的拓扑结构实现资源按内容寻址。其中,网络中每一个节点在每一维标识符空间中均保存与自己(逻辑上或者物理上)相连的节点的信息。标识符空间中的每一个ID代表一个节点,该节点存储一对相应的(Key,Value),其中Key通过Hash运算映射为节点对应Node ID。当需要查询时,只需采用相同的方法对查询的关键字Key进行Hash运算,得到一个Node ID,从该节点即可找到存储与关键字对应内容(Value)。N维几何分割的实现方法使得系统具有很好的可扩展性较好,但理论上比较复杂。

基于DHT的分布式查询协议,可以针对不同关键字,建立多个关键字服务器。用户输入待查的关键字,并通过Hash函数映射到对应的关键字服务器,通过多个关键字对应的不同的关键字服务器之间的协同完成查询。如果合理分布关键字,可以减轻数据的传输量。但是,DHT方法适合于准确的查找,目前也无法解决多关键字查询的场合。

2.3.2 基于域模型的查询方法

在基于域(Domain)模型的P2P系统中,一方面尽可能充分利用组成网络的对等节点性能不一的特点,将整个网络中的对等点划分为超级节点(Super-Peer)和普通节点两类,一方面根据某些原则将节点组织在若干个域中。各区域有一个中心控制节点,一般由性能较高的Super-Peer节点充当,作为信息搜索的中心,它负责收集某一区域内多个对等节点的信息,但各个Super-Peer之间构成一个纯的P2P网络。

如图3所示,域内的节点查询请求都向其Super-Peer发出,如果待查的信息存在于本地域内,则可以有效地找到。若信息不能在本地域内找到,则由Super-Peer将请求向其它域的Super-Peer广播,之后转移到其它域中进行查询。为了防止域中的Super-peer出现故障,有的系统从域中任选一个普通Peer代替,或者采用选举的方法找到另外一个Peer充当。

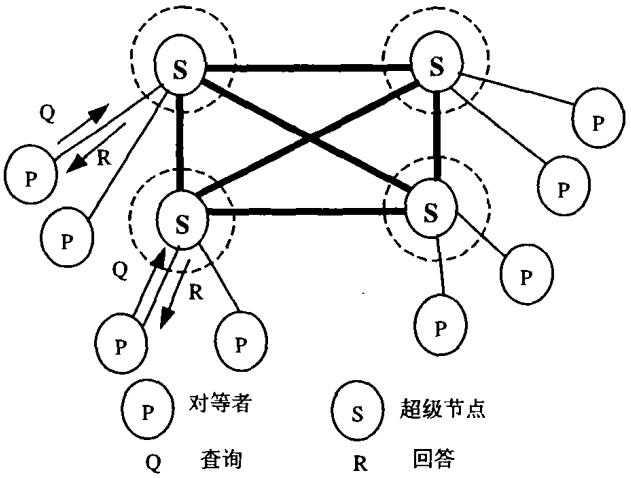

图3 基于 super-peer 域的查询

这种基于域的方法要考虑如何在网络中形成域，如何控制 Super-peer 的数量，如何有效地在 Super-peer 间传播以及处理查询和回答。文[12]提出了一种基于各个节点的 Interest 划分区域的方法，Interest 相近的节点被认为是 good peer，系统自动在 good peer 之间连接。各个节点对相邻节点的 Interest 的了解是根据对它们对查询消息的应答结果来得到的。通过对各 peer 之间的 Interest 的计算，将这些具有同样 Interest 的节点连接在一个域中。基于这些信息，节点还可以动态决定与哪些节点连接，以及什么时候建立或断开一个连接。因此当网络拓扑结构改变时，可以根据节点的 Interest 比较容易地对重新划分域。

2.3.3 基于 Routing Indices 的方法 文[4]提出了一种称为“路由索引”(Routing Indices－RI)的方法，使得查询请求的传递更有目的性，尽可能传给那些可能会有正确结果的节点。实现时，每个节点都维护本地的一个 RI，其中存放此节点的 r 个邻居节点中的共享资源的索引信息(r 是系统定义的，被称为索引半径)。当一个节点接收到查询请求时，如果它无法找到相应的结果，基于它的 RI，就向它的邻居节点的子集递交查询，即有 r 个节点处理查询。因此，只通过在少数几个节点上执行查询，就能够查询到多个节点上的数据，即可以满足用户的要求，又可以获得低的代价。

同时，用户除了提交查询请求之外，还提交停止条件(例如所期望的结果数目)。收到查询请求的节点首先在本地进行查询，并将结果返回给用户。如果没有达到停止条件，该节点就选择一个或更多个它的邻居，把查询请求(还有一些中间状态信息)传给它们。每个邻居依次以近似的方式进行查询，把结果指针返回给用户，并把查询传给邻居们。如此继续，直到达到停止条件，或者确知不可能达到停止条件，此时停止传递查询，得到查询结果。以图4为例。节点 A 最初收到一个查询，A 检查本地结果并且把结果发送给请求节点。然后，假设停止条件还没有被满足，节点 A 选择节点 D 作为解决查询的最佳邻居并把查询传递给它(箭头)。一般来讲，为了让节点能够证实是否达到了停止条件，需要把目前为止发现的结果数目作为状态信息包括进入每条发送查询的消息。然后 D 处理查询，并选择 I 为继续解决查询的最佳邻居。现在让我们假设：I 已经处理了查询，但是没有发现足够达到停止条件的结果，在此情况下，I 将查询转给 D，D 再转给下一个最佳邻居(此情况为 J)。假设 I 发现了足够的结果，则停止继续查询。

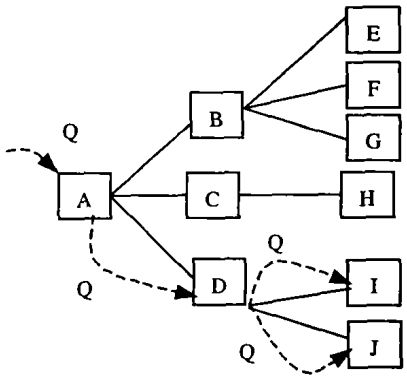

图4 分布式索引机制

在这类系统中，要根据网络状态信息及时更新索引表，当节点加入、离开或更新数据时，要创建和维护索引表。(1)当节点 X 要加入到网络中时，它发出一个 *join* 消息，并给出 r 的信息(相当于 TTL)以及它拥有的资源描述信息。当其它节点接收到 X 发出的 *join* 消息时，它也直接向 X 发出一个 *join* 消息，其中包含它的资源的描述信息，从而使两个节点都将它们的资源信息加入到各自的索引 RI 中。(2)当节点离开网络时，包含此节点信息的节点就从它的索引表中去掉此节点的资源描述信息。(3)当用户更新节点上的信息时，它的节点就向 r 范围内的节点发送一个更新消息，包含所有被修改的文件的资源描述信息，所有节点接收到此消息后随后就立即更新它们的索引表。

如果能对返回的结果信息进行排序，就可以提高数据质量，文[5]提出了用良好度(Goodness)对各节点的返回结果进行排序的方法及具体的计算公式。良好度的解释有所不同，但通常它反映了邻近节点中的共享资源的数目。

为了及时反映的节点上的资源信息，当节点加入、离开或更新数据时，要创建和维护索引表，这个工作量是很大的，要花费一定的时间。

3 P2P 系统查询机制的设计要求和目标

3.1 设计要求

一个良好的 P2P 系统中查询机制应该是为用户有效地定位所期望的数据，即要综合考虑资源的可获得性、网络拓扑的变化和响应时间等相关因素。然而 P2P 网络的高度动态性以及各个对等者的高度自治性等使得这一目标的实现存在诸多的困难。文[11]讨论了在设计时要考虑下面几个因素：

(1)拓扑结构：定义了对等者相互连接的方式。在某些系统(如 Gnutella)中，对等者可以连接它想连接的任何对等者。而在有些系统中，如基于域的系统中，对等者被组织成严格的结构，其中连接的数目和性质在协议中进行了严格的描述。定义严格的拓扑可以提高效率，但是会限制自治性。

(2)数据排列：定义了数据在对等者网络中分布的方式。例如，在 Gnutella 中，每个节点只存储它自己的数据集合。而在超对等者(Super-Peer)网络中，相关对等者集合的元数据信息被集中排列在单一的超对等者中。

(3)消息路由：定义了消息在网络中传播的方式。当一个对等者提交了查询请求时，查询消息被发送给该对等者的邻居们，邻居们再依次顺序或者并行地把消息传递给它们的一些邻居，如此继续。何时、把消息传给谁，是由路由协议描述的。通常，路由协议可以利用已知的拓扑和数据排列形式，来减少发送消息的数目。

(下转第65页)

〈COMMENT=description/〉
〈/LOCAL MANAGER〉
〈/LM〉

节点管理层主要实现本地或远程节点管理的各项功能，它也是采用动态装配方法实现在网络管理中服务的分布。在节点管理层中的各个管理器既可以访问建模层提供的功能，也可以访问动态 MIB 库，它们是主动网络管理系统的基本构成模块。其主要构成模块有：

①节点管理器(Node Mgr)：管理本地的硬件、软件与带宽资源，它可以配置和监视这些资源的性能并处理它们的操作故障。

②虚拟网络管理器(VAN Mgr)：配置和监控虚拟主动网络的性能，包括创建和配置一个虚拟网络、分配本地资源、监控虚拟网络的性能和故障并进行恢复处理。

③执行环境管理器(EE Mgr)：配置和监控 EE 的性能，包括配置 EE 环境、分配 EE 所需资源、将 EE 与 VAN 进行链接、监控 EE 的性能以及主动网元在 EE 中的执行情况。

④主动应用管理器(Active Mgr)：可以扩展用以配置和监视各种特定的主动应用。

结束语 网络管理是一个庞大、复杂的系统工程。网络管理涉及到分布式计算、面向对象技术、人工智能等多种领域。由于主动网络中节点具有独立的计算和处理能力，因此可以实现网络管理的分布化并减少网络中的冗余信息。

本文探讨了一种基于节点的主动网络管理模式的设计，并对其结构、性能与管理机制分析。这种管理模式具有稳定的层次结构、各层独立性强，而且便于扩充，满足了主动网络中节点的灵活性和主动应用的扩展性要求，是一种较为理想的网络管理方案。

参 考 文 献

1 Mahon H, Bernet Y, Herzog S, Schnizlein J. Requirements for a Policy Management System, Internet Draft. Nov. 2000
2 Al Shaer E. Active Management Framework for Distributed Multimedia Systems. Journal of Networks and Systems Management, 2000, 8(1)
3 Kawamura R, Stadler R. Active Distributed Management for IP Networks. IEEE Communications Magazine, 2000, 38(4)
4 Salamanca E, Serrat J, Vivero J. Active policy-based management. In: OPENSIG 2001 Workshop, Sep. 2001
5 Kiwior D, Zabele S. Active Resource allocation in Active Networks. IEEE JSAC, 2001, 19(3): 452～459
6 Schwartz B Y, et al. Smart Packets: applying Active Networks to Network Management. ACM Transaction on computer Systems, 2000, 18(1)

(上接第59页)

然而在实际系统中，满足上述要求的整体模型是以不同形式出现的，原因就在于各个系统的要求不同。这些特定要求主要被划分成以下几个方面：

(1)表达：在描述查询请求消息时，系统使用的查询语言必须能够充分描述所期望的数据。

(2)复杂度：在一些系统中，返回任意一个结果就足够了，但是在另一些系统中可能需要返回所有的结果。因此前一种系统只需要相对简单的搜索机制，而后一种系统需要复杂的搜索机制。

(3)自治性：每种搜索机制都要定义对等者相对于拓扑结构、数据排列和消息路由的行为，然而，对等者的自治性会受到限制。例如，某对等者可能只想与它的朋友们或者其它在相同组织中可以信任的节点相连接，或者该对等者期望控制那些可以存储它的数据的节点。根据系统的目的和用户需要，搜索机制要与对等者自治程度相一致。

3.2 设计目标

一个设计良好的搜索机制在必须满足系统要求的同时，还要提高以下目标：

(1)效率：当查询消息通过网络广播时，各个节点需要使用 CPU 资源来转发请求，处理它，并使用网络带宽来发送和接收请求，快速响应，因此一般多用带宽、处理能力、响应时间等衡量效率。

(2)服务质量：因为应用的不同，所以可以用不同的标准(如结果数量、响应时间等)来衡量服务质量。服务质量常指用户察觉到的质量，而效率关注达到特定服务程度所利用的资源代价(如带宽)。一般来讲，P2P 网络的 QoS 问题包括以下几个方面：①结果数量：符合要求的结果集的大小；②响应时间：用户需要的信息可能在多个节点同时存放，如何选择一个处理能力强、负载轻、带宽高的节点需要用户考虑。③满意程度：用户可能共享出无用或者违法信息，造成信息垃圾充斥网络。因此，网络应该能够控制用户共享的信息，提高用户获得有用信息的效率。

(3)鲁棒性：鲁棒性指的是失败时的稳定性，强健性，即系统中的对等者离开网络或者搜索失败时仍保持的服务质量和效率。我们知道，P2P 系统中各个对等者是自治的，节点可以连续地加入或者脱离网络，因此网络规模和网络拓扑结构是不可预知的，变化很快。例如，用户可能会突然关闭其它人正在访问的电脑设备，出现其他人无法访问的情形，同时也增加了大量的不确定带宽，服务器资源和分布式存储的需求，这些动态特性要求网络中的节点能够按照网络信息的变化准确地配置和改变自身，所以必须提高系统的鲁棒性。

结束语 查询机制是 P2P 系统研究的关键问题之一。本文所描述几种常见的查询方法，给出了查询机制的一般要求和设计目标。随着 P2P 用户数量的快速增加，对资源查询机制的扩展性和可用性提出了很高的要求。随着技术和理论的发展，查询机制和实现方法会向着更加智能高效的方向发展，如一些系统采用智能代理等实现手段提高查询效率和网络带宽的利用率。

参 考 文 献

1 Napster website. http://www.napster.com
2 Gnutella website. http://www.gnutella.com
3 Freenet website. Http://www.freenet.com
4 Crespo A, Garcia-Monila H. Routing indices for peer-to-peer systems. In: Proc. of the Intl. Conf. On Distributed Computing Systems . 2002
5 Finding Good Peers in Peer to Peer Networks
6 Yang B, Garcia-Monila, H. Improving search in peer-to-peer systems. In: Proc. of the International Conference On Distributed Computing Systems . 2002
7 Lv Q, Cao P, Cohen E, Li K, Shenker S. Search and replication in unstructured peer-to-peer networks. In: Proc. of Intl. Conf. on Supercomputing . 2002
8 Milojicic D S, Kalogeraki V, Lukose R. Peer-to Peer Computing: [Technical Report]. HP Laboratories Palo Alto. HPL-2002-57. 2002
9 Stoica I, et al. Chord: A scalable peer-to-peer lookup service for internet applications. In: Proc. of ACM SIGCOMM'01, Aug. 2001
10 Ratnasamy S, et al. A scalable content-addressable network. In: Proc. of ACM SIGCOMM'01, Aug. 2001
11 Daswani N, Garcia-Monila H, Yang B. Open Problems in Data-Sharing Peer-to-Peer Systems
12 PeerSearch: Efficient Information Retrieval in Peer-to -Peer Networks. HPL-2002-198. July 12th, 2002
13 Sylvia R, Scott S, Ion S. Routing Algorithms for DHTs : Some Open Questions[C]. In: 1st Intl. Workshop on Peer-to-Peer Systems. March 2002