一种用于大规模 P2P 点播系统的拓扑结构

武广柱[1,2],王劲林[2]
WU Guang-zhu[1,2],WANG Jin-lin[2]

1.中国科学院 声学研究所,北京 100080
2.中国科学院 研究生院,北京 100080
1.Institute of Acoustics,Chinese Academy of Sciences,Beijing 100080,China
2.Graduate University of Chinese Academy of Sciences,Beijing 100080,China
E-mail:wugz@dsp.ac.cn

WU Guang-zhu,WANG Jin-lin.Hierarchical topology structure for large-scale P2P VoD system.Computer Engineering and Applications,2008,44(17):21-23.

Abstract: This paper presents a hierarchical DHT topology structure based on dynamic time coordinate for large-scale P2P VoD system.In this coordinate system,peer's coordinate maintains constant unless the peer's play occasion jumps to another point of the stream.Thus DHT based lookup protocols are adopted to track buffer information of peers with small overhead.Simulations show that the design achieves good performance.

Key words: peer-to-peer;VoD;DHT;resource locating

摘 要:提出了一种基于动态时间坐标的分层 DHT 拓扑结构,解决了因大规模 P2P 点播系统要求细粒度追踪而难以应用 DHT 的问题。在动态时间坐标系中,节点的播放点坐标不再随着节点的播放而移动,从而使得 DHT 能够用于追踪点播系统节点缓存位置。仿真结果证明了方法的有效性。

关键词: Peer-to-Peer;VoD;DHT;资源定位
DOI:10.3778/j.issn.1002-8331.2008.17.006 **文章编号:**1002-8331(2008)17-0021-03 **文献标识码:**A **中图分类号:**TP301

1 前言

相比直播而言,P2P 点播系统[1-7]的一个难点在于资源定位:节点的缓存状态难以追踪,搜索合作节点困难[1]。

对于直播系统,处于同一频道的各个节点基本处于同一播放点,在节点缓存大小合适的条件下,这种同步性使得合作节点发现相对简单。索引服务器记录下频道内的所有节点,当新的节点加入时,索引服务器随机返回一些节点,这些节点都能够成为新加入节点的合作节点(本文不涉及合作节点选择的优化问题)。因为在直播系统中节点不会进行快进、快退、拖动等 VCR(Video Cassette Recorder)操作,故节点无需经常性地报告自己的播放点,索引服务器仅仅记录节点所属的频道即可。这种索引粒度为"频道"的索引方式和无 VCR 操作使得服务器压力较小。

但在 P2P 点播统中,合作节点的发现具有挑战性。点播系统具有异步特性。点播系统中的用户随时加入系统并从节目开始或者任意位置播放,此外用户还会在播放过程中进行快进、快退、拖动等 VCR 操作。在节点缓存受限的情况下,两个节点要想协作,则其播放点必须相近,播放点距离较远的节点不能相互协作。例如,一个频道中现有三个节点,P_a 在节目的第 1 分钟播放,P_b 在第 2 分钟位置播放,P_c 在第 30 分钟位置播放。现在新加入一个节点 P_d 从第 28 分钟处播放,则只有节点 P_c 才可能成为 P_d 的合作节点。而 P_d 想要发现 P_c,索引粒度为"频道"是不行的,系统就必须提供一种较细粒度的索引机制。而这种较细粒度的索引可能是分布式的也可能是集中式的。集中管理在节点数量不大时最为有效:节点定期向服务器报告其缓存位置,需要查找合作集的节点向索引服务器查询。然而,一旦大规模部署,节点播放位置的实时刷新以及大量节点的跳转请求会使得服务器将难以承担。另外,单点失效也是集中管理方式的一个弱点。分布式解决可能采用 Session[2]、Generation[3]等"关系内嵌"技术,将节点的播放点关系内嵌到节点的逻辑拓扑联系中。而这种方法索引时需要多次顺次跳跃才能到达目标,效率不高。

结构化的 DHT[8]网络结构是 P2P 搜索技术中较为成熟高效的方法,然而,DHT 并不能用于 P2P 点播系统来追踪节点缓存。这是由于点播系统的索引不像 BitTorrent[9]等粒度为"文件"

基金项目:国家高技术研究发展计划(863)(the National High-Tech Research and Development Plan of China under Grant No.2005AA1032);中国下一代互联网示范工程(Supported by China Next Generation Internet Foundations(CNGI) No.CNGI-04-15-2A)。
作者简介:武广柱(1979-),男,博士生,主要研究领域为宽带多媒体通信,嵌入式系统;王劲林(1964-),男,博士生导师,主任研究员,主要研究领域宽带多媒体通信。
收稿日期:2008-01-29 **修回日期:**2008-03-21

的系统。在 BT 等内容分发系统中,仅仅依靠文件名哈希就可以完成索引。点播系统中合作节点的发现必须依靠播放点,索引粒度变小,这就需要对频道号和播放点一同进行哈希计算。然而节点的播放点是时刻变化的,这样节点就需要不停地向 DHT 注册当前播放点位置,并撤销原注册位置。这种频繁的注册、撤销操作会给整个网络带来巨大的开销。

本文在 VoD 系统中提出了动态时间坐标系的概念,系统中各频道都建立一个动态时间坐标系,在该坐标系下节点播放点坐标不进行 VCR 操作时为常数。利用这一性质,设计了基于动态时间坐标的分层 DHT 拓扑组织方法,解决了因点播系统要求细粒度追踪而难以应用 DHT 的问题,并且借助分层使得 DHT 查找复杂度由原来的 O(log(N))减小为 O(1),N 为系统中节点的个数。

2 相关工作

P2P 直播系统已经走向成熟并投入商业运营。但 P2P 点播系统却因异步特性而仍然存在诸多挑战。资源定位问题便是其中之一。下面对几个典型 P2P 点播系统的资源定位方法做一个回顾。

在 P2Cast[2]中,加入系统的时间相近的节点构成一个 Session。对于每一个 Session,媒体服务器和本 Session 中的节点通过单播构成一棵应用层组播树,称为基础树。对于一个新加入的节点,如果其父节点没有缓存其所需要的内容片段,则节点直接从服务器下载,也可以从本 Seesion 内具有该片段的其它节点下载。这种打补丁的工作方式使得 P2Cast 比传统的 C/S 模式能够服务更多的用户。新节点 P 加入系统首先要联系服务器,以便服务器进行节点追踪。如果 P 属于一个已经存在的 session,则 P 加入基础树并选择一个补丁服务节点,如果成功,这节点被接纳。否则,如果因节点带宽不足或者找不到合适的补丁服务节点进而需要服务器来提供补丁服务而服务器带宽又不足,则节点被拒绝。如果节点需要新开一个 Session 而且服务器带宽允许,则允许节点加入,否则拒绝节点。P2Cast 的节点加入完全依赖于服务器进行索引,是一种中心管理方式。P2Cast 并未对用户 VCR 操作时如何定位资源进行讨论。

在 P2VoD[3]中节点依据它们的加入时间组织成多等级群组,数据流沿重叠树进行转发。每个节点从其上级群组中的一个父亲节点接收数据并将数据转发到位于其低层群组的子节点。新节点加入系统可以尝试加入低的群组或形成一新的最低群组。如果它不能从组播树中找到一可用的父亲节点,且服务器有足够的带宽,则它直接连接到服务器。可见,P2VoD 采用的是一种关系内嵌式的索引方法。P2VoD 没有对用户 VCR 操作造成的节点跳跃索引做优化。如果一用户启动跳转请求,节点需要顺序地搜索其上群或下群组,由于群组数量很大,搜索开销是非常高的。

OBN[1]抛弃了 P2Cast 和 P2VoD 将资源定位内嵌到内容分发拓扑中的做法。OBN 构建了称作重叠缓存网络的拓扑结构,利用了节点流畅播放时各节点播放点的差相对固定这一性质进行资源定位。

Vmesh[4]将 DHT 索引引入了 P2P 点播系统。一个新加入节点可以使用 DHT 搜索它所关注的分段。然而,VMesh 应用 DHT 解决的是相对固定的硬盘数据索引。Vmesh 中的节点要存储一些频道媒体文件的数据块并保留较长时间,这些数据块并不随播放点的移动而动态变化。Vmesh 没有解决播放缓冲区的数据的索引。

3 动态时间坐标系

系统中每一个频道都有一个动态时间坐标系 DTCS(Dynamic Time Coordinate System)。假设频道中有一个节点从某时间点开始循环播放本频道节目,这一假设节点当前正播放的位置称为本频道的虚拟播放点。动态时间坐标系的坐标原点和本频道的虚拟播放点相等。

设 C 为系统中所有频道的有限集。设系统中的各频道 $C_j \in C$ 的虚拟结点从绝对时间 TS_j 开始不断循环播放本频道的影片,本频道整个影片的播放时间是 T_j。对于 C_j,可以计算出当前时间的虚拟播放点播放循环播放的遍数 N_j 以及其播放位置 TP_{curr},TP_{curr} 即为 C_j 的动态时间坐标系原点 O_j:

$$N_j = floor(T_{curr} - TS_j)\%T_j \tag{1}$$

$$O_j = TP_{curr} = (T_{curr} - TS_j)\%T_j \tag{2}$$

每一个频道的影片都被按照一固定时间 TL 划分为多个片段,称为桶。设 B 为系统中所有桶的集合,$B_j \in B$ 是频道 $C_j \in C$ 的所有桶的集合。给定一 C_j 中任意播放点 TP_{jx},设当前时间是 T_{curr},则 TP_{jx} 所属的桶为:

$$D_{jx} = \begin{cases} TP_{jx} - O_j & \text{if} \quad N_j \text{为偶数} \\ TP_{jx} - O_j + T_j & \text{else if} \quad N_j \text{为奇数且} \ TP_j < O_j \\ TP_{jx} - O_j - T_j & \text{else} \end{cases} \tag{3}$$

$$B_{jx} = floor\left(\frac{D_{jx}}{TL}\right) \tag{4}$$

其中,TP_{jx} 是从影片开始到 P_{jx} 的时长。

可以看出,在节点步进行 VCR 操作的一段时间内,节点在动态时间坐标系下所属的桶号不随播放发生变化。

4 基于动态时间坐标的分层 DHT 拓扑组织方法

基于动态时间坐标的分层 DHT 拓扑组织方法解决了因点播系统要求细粒度追踪而难以应用 DHT 的问题,并且借助分层使得 DHT 查找复杂度由原来的 O(log(N))减小为 O(1),N 为系统中节点的个数。

在动态时间坐标系下,节点坐标(播放点)在未进行 VCR 操作的时间段内成为一个恒量。这样,节点无需频繁向系统注册坐标,DHT 就可以容易地应用到点播系统了。一种简单的组织方式就是将节点坐标的哈希值作为 ID,并将这一 ID 注册到 DHT 网络。给定一 $C_j \in C$ 中任意播放点 TP_{jx},则定义 TP_{jx} 的 ID 为:

$$ID(TP_{jx}) = Hash(Name(C_j) + floor\left(\frac{D_{jx}}{TL}\right)) \tag{5}$$

为了进一步降低查找复杂度,采用了分层的 DHT 结构。如图 1 所示,整个逻辑拓扑被分为 4 层。其中,IS(Index Server)是索引服务器,提供时间同步、缓存索引、节目热度统计、缓存统计等功能;RM(Root Manager)是频道的根管理员,负责管理其

下各个 BM，并周期性的将本频道的统计信息报告给 IS;BM(Bucket Manager)是一个频道的桶管理员,负责跟踪和统计其管理的一般节点的播放缓存和磁盘缓存等信息。每个频道都有一个 RM 和多个 BM。BM 和 RM 由符合一定条件的超级节点担任,这些条件包括:不在 NAT(Network Address Translation)和防火墙后、具有足够强的 CPU 处理能力、具有足够大的内存、具有足够的网络带宽、较长的稳定在线时间。超级节点按照 DHT 组织拓扑。

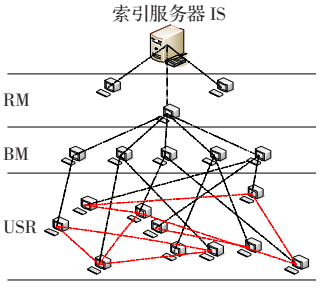

图 1 分层拓扑结构

4.1 RM 的生成(RM_CREATE)

IS 负责指定 RM。当一个频道的 RM 不存在时,IS 根据频道名计算其频道 ID,根据此 ID 通过 DHT 的 FINDNOD 算法找到节点并指定其为 RM。

$$ChannelID(C_j)+Hash(Name(C_j)) \quad (6)$$

4.2 BM 生成(BM_CREATE)

仅 RM 有权负责发起 BM_CREATE 指令。当一个频道某桶的 BM 不存在时,RM 根据频道名及其桶号计算其 $ID(TP_{jx})$,根据此 $ID(TP_{jx})$通过 DHT 的 FINDNOD 算法找到一个节点并指定一个作为 BM。

4.3 节点加入系统(JOIN_SYSTEM)

为了减小路由跳数并减小节点动态带来的拓扑维护开销,并不是所有节点都加入 DHT 网络，而只有那些符合一定条件的较稳定的超级节点才加入 DHT。不能加入 DHT 网络的节点不负责任何 RM、BM 的事务。作者在其他论文中详述超级节点的选择算法。

4.4 节点加入频道(JOIN_CHANNEL)

当节点 P 点播某频道时,P 向 IS 请求本频道的 RM 及其最近加入本频道的节点集 S。P 收到回复后向 S 中的节点请求节目最开始的内容（每个节点必须保存所在频道最开始 10 s 的内容),同时 P 向 RM 请求 BM 的地址。如果 IS 返回的 S 不足以使得 P 加入系统,则 P 向相应的 BM 请求节点集。P 加入频道后向合作节点请求频道的初始片段，在缓存数据达到 10 s 时开始播放并向 BM 注册坐标。

4.5 节点 VCR 操作(VCR)

在某频道内的节点如果进行 VCR 操作，则首先计算其目标播放点和原播放点是否具有相同的动态坐标 ID。如果相同,则节点将播放点移动到目标播放点处;如果 ID 不同,则节点向相应的 BM 请求合作节点集并注销原动态坐标和注册新的动态坐标。

5 实验结果

为验证算法在大规模 P2P 点播系统中的性能,假设点播系统提供 1 000 套节目，每套节目的平均播放时间为 4 000 s,系统中有 20 万用户,用户平均每 500 s 将进行一次拖动。网络中节点间的平均延迟为 100 ms,带宽为 10 M。索引服务器平均响应时间是 5 ms,普通节点的平均响应时间是 20 ms。

本文对 C/S 模式方法、P2VoD 采用的关系内嵌方法、基于动态时间坐标系 DCS 的分层 DHT 方法进行了仿真。图 2 为节点系统对节点拖动和加入的响应时间(s)。可见,由于 C/S 模式下服务器不但要处理节点加入频道还要处理节点的拖动请求,响应时间很长;关系内嵌方式因为需要多跳也需要较长的响应时间;基于 DCS 的分层 DHT 方法因将索引负担从服务器分布到普通节点并且不需要关系内嵌方式的多跳到达,响应时间最短。图 3 为索引服务器的负载情况(请求数/s)。可见,因动态时间坐标系方法和关系内嵌方法不需要服务器处理拖动情况,服务器压力较小。

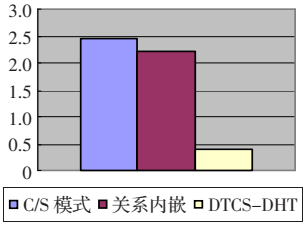

图 2 响应时间

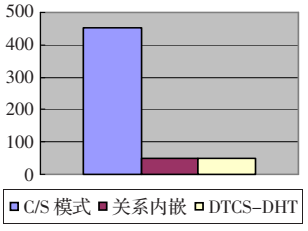

图 3 服务器负载

6 结束语

通过动态时间坐标系的概念，本文将 DHT 应用到了点播系统的节点跟踪问题上，使得合作节点查找的复杂度变为 O(log(N)),N 为 DHT 中节点的数量。为了进一步减少复杂度,将频道和频道的各桶进行了分层管理,使得节点查找和注册复杂度进一步减小到 O(1)。即使 BM 或 RM 甚至 IS 出现暂时故障,节点仍然可以依靠标准 DHT 的 FIND 操作找到合作节点。基于动态时间坐标的分层 DHT 拓扑组织方法具有很好的查找效率和很高鲁棒性。这一鲁棒性得益于 DHT 的鲁棒性。关于各种 DHT 算法的鲁棒性论文不做讨论。

当一个频道的用户加入速率很高时,RM 和 BM 压力较大。作者将在其他论文中给出一种基于槽的重叠 ID 负载均衡 DHT 拓扑组织方法。

参考文献:

[1] Liao Chi-Shiang,Sun Wen-Hung,King Chung-Ta,et al.OBN:peering for finding suppliers in P2P on-demand streaming systems[J]. ICPADS,2006(1):235-242.

[2] Guo Y,Suh K,Kurose J,et al.P2cast:peer-to-peer patching scheme for VoD service[C]//Proc of the 12th Int'l Conf on World Wide Web.New York:ACM Press,2003:301-309.

[3] Do T,Hua K A,Tantaoui M A.P2vod:providing fault tolerant video-on-demand streaming in peer-to-peer environment[C]//IEEE International Conference on Communications,Paris,France,2004:1467-1472.

[4] Ken Yiu W-P,Xing Jin,Gary Chan S-H.Distributed storage to support user interactivity in peer-to-peer video streaming[C]//IEEE International Conference on Communications,2006,1:55-60.

(下转 32 页)