

# 基于 P2P 的流媒体系统模型研究<sup>①</sup>

## Research on the Model of P2P – Based Media Streaming

李莉 韩慧健 (山东经济学院计算机科学与技术学院 济南 250014)

**摘要:**基于 P2P 方式的流媒体系统可以充分利用节点资源,提高系统的可扩展性,但同时由于 P2P 中节点的不稳定性,也给 P2P 方式流媒体系统的设计带来很多挑战性问题。本文在分析 P2P 流媒体传输系统特点的基础之上,重点归纳了 P2P 流媒体系统一些典型模型的工作原理和优缺点,最后指出了进一步的研究方向。

**关键词:**P2P 流媒体 模型

### 1 引言

近年来,由于对等网络技术的迅速发展,基于 P2P (Peer to Peer) 方式的流媒体传输的研究逐渐引起了人们的注意。P2P 方式的核心思想是网络中各节点可以直接通信,每个节点在利用其他节点资源的同时也为其它节点提供服务,也就是充当了 C/S 模式中服务

这种不稳定性也给基于 P2P 的流媒体传输系统的实际应用带来很多挑战性问题。目前,对于 P2P 流媒体系统的研究还处于起始阶段,人们从多个角度开展了研究,包括文件搜索机制、QoS 控制机制、激励机制等<sup>[1]</sup>。其中流媒体系统模型是需要解决的最基本问题之一。

本文首先分析了 P2P 流媒体传输系统的特点,接着介绍了 P2P 流媒体模型的分类,重点归纳了一些典型模型的工作原理和优缺点,最后给出了结论。

### 2 P2P 流媒体传输系统概述

流媒体传输系统主要包括以下几个部分<sup>[2]</sup>,如图 1 所示。

(1) 媒体编码和制作工具。对其它格式的媒体文件或外设采集的实时媒体数据进行编码、编辑生成流媒体数据格式。可以将流格式文件存储在存储器中,或直接将媒体流发布到流媒体服务器上。

(2) 流媒体服务器。用来管理、传输和控制流媒体数据,以及实现对客户授权、登录、退出等多种管理。流媒体服务器可以响应客户请求,并通过流媒体传输协议将流媒体数据传输给客户。

(3) 网络。是实现流媒体服务的底层基础设施。要求支持流媒体的传输。

(4) 媒体播放器。供客户浏览流媒体文件。其基本工作过程为:从网络中接收传输过来的各类媒体数据流,存入一个缓冲队列,然后对其中的每一帧数据调

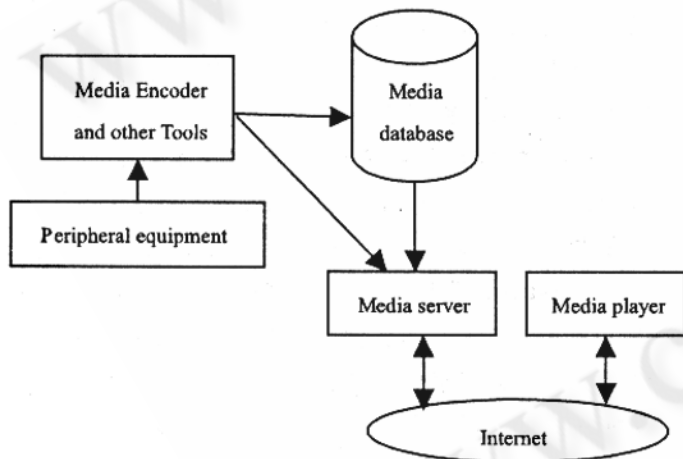


图 1 流媒体传输系统的组成

器和客户两种角色。P2P 系统通过把服务器的负载分散到每个节点中,可以有效地减轻服务器的负担,提高系统的可扩展性,因此利用 P2P 网络技术可以实现大规模流媒体数据的传输和播放,从根本上解决基于 C/S 模式的流媒体系统的扩展性差、容错性差等问题。但 P2P 网络中节点的加入和退出具有很大的随机性,

① 基金项目:山东省自然科学基金项目(Y2006G31);山东经济学院校级科研项目(01610854)

用各类解码器重建成原始的数据格式,最后经同步后在设备上播放出来。

P2P 流媒体系统是基于 P2P 网络来实现流媒体的传输。所谓 P2P 网络是指利用 P2P 计算模式在 Internet 物理拓扑结构的基础之上建立的逻辑覆盖网络,也称为 P2P 覆盖网络 (Overlay Network),它决定了节点连接的拓扑结构和节点与相邻节点保持连接时的行为规范。基于 P2P 网络构建流媒体传输系统,要求覆盖网络能够为上层的流媒体服务提供良好的支持,它的设计目标包括:为用户提供高质量的网络连接、支持大规模的节点同时在线、用户可以高效的搜索媒体文件等。基于 P2P 模式进行流媒体传输时,服务器不是以单播方式向所有节点发送数据,而只是把数据发给部分节点,再由享受服务的这些节点向其它节点提供服务。从以上分析可以看出,P2P 流媒体传输系统并没有改变现有的流媒体传输协议和流媒体传输系统架构,只是改变了 C/S 模式下的服务方式和数据传输路径。一般只需在现有流媒体系统的基础之上,增加一些新的模块和功能,主要用来实现对流媒体传输系统所基于的 P2P 网络进行节点管理、拓扑结构优化以及数据调度分配等功能。

### 3 P2P 流媒体系统模型

流媒体系统模型决定了节点之间的关系和数据传输的方式。根据节点构造的拓扑关系,可以把 P2P 流媒体模型分为两类:树状模型和网状模型。

#### 3.1 树状模型

树状模型系统建立在应用层组播技术基础之上,属于一对多传输方式,也就是一个节点向多个节点发送数据。树状模型又可以分为单树模型和多树模型。

(1) 单树模型。单树模型是指系统只构建一个组播树,典型的模型有 SpreadIt<sup>[3]</sup>、ZigZag<sup>[4]</sup>等。

SpreadIt 模型是最简单的树状模型。服务器节点和请求同一内容的客户节点组织成一个组播树,服务器节点是组播树的根,第一层节点从服务器获取数据,第二层节点从第一层节点获取数据,依此类推。如图 2 所示。在这棵组播树中,n1 和 n2 直接从服务器获取数据,n3 和 n4 从 n1 节点获取数据,n7 则从 n3 节点获取数据。因为每个节点只从父节点接收数据,所以这种分层树状结构,可以有效地避免节点之间互相发送

相同数据而引起的数据风暴。

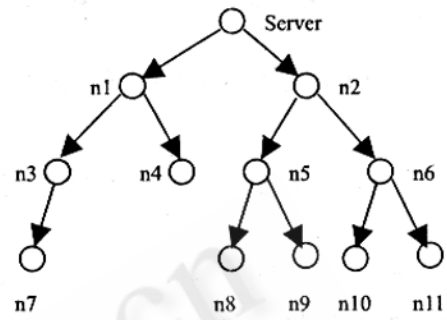


图 2 SplitIt 组播树举例

在建立应用层组播树时,要解决节点的加入和离开问题。SpreadIt 模型中,节点的加入和离开算法都很简单。如节点加入算法可描述如下:有新节点 n 加入,首先向 server 发送请求,如果服务器有能力,则可以直接向节点 n 提供服务,否则把请求发给它的一直接子节点,子节点再判断自身是否有能力给 n 提供服务,依此类推,直到 n 节点找到父节点。转发请求给哪些节点,可以选择不同的策略,如可以选择物理位置和 n 最接近的节点,这种策略考虑了节点的物理位置,使组播树更接近节点的物理拓扑。

在组播树中,节点离根节点越远,数据时延就越大,而因每个节点的处理能力有限,不可能给太多子节点提供服务,所以大量新增加的节点必导致树深度的增加。SpreadIt 模型,节点的管理规则简单,并不能有效保证组播树深度和宽度的平衡。而 ZigZag 模型通过定义一整套完整的树构建规则,保证了组播树的深度维持在  $O(\log N)$ ,N 为系统中节点的数量。

单树模型存在以下问题:每个节点单一的从父节点接收数据,因而父节点的变化对子节点服务质量的影响很大,且组播树中的最下层节点(称为叶子节点)不再参与数据的分发,系统的资源未得到完全利用。为了解决以上问题,近来的研究提出了使用多棵树代替单一树的多树模型。

(2) 多树模型。多树模型是指系统同时构建两个或多个组播树,以 Splitstream<sup>[5]</sup>和 CoopNet<sup>[6]</sup>为代表。

Splitstream 模型的核心思想是把媒体数据分成 k 个码流 (Stripe),每个码流构建自己的组播树。节点可以根据自身的能力选择转发其中几个码流而加入多个

组播树。组播树在构建时要求每个节点只在某棵树中为中间节点,而在其它树中都为叶子节点。图 3 为 Splitstream 组播树的一个例子。图中源节点 1 产生了两个 Stripe,分别构造了两个组播树,如对应 Stripe1 的组播树,节点 2 把数据发给 3、4,节点 3、4 再发给 5、8、6、7。所有节点满足在一棵树中为中间节点,而在另一棵树中为叶子节点。

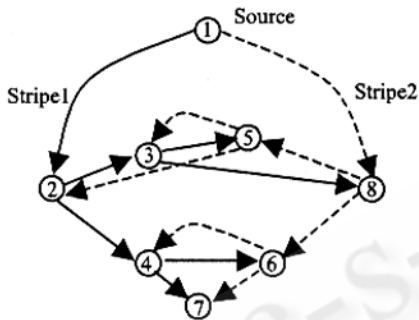


图 3 Splitstream 组播树举例

CoopNet 模型同样也是多树模型,利用了多描述编码 (MDC) 技术。和 Splitstream 模型不同的是 CoopNet 模型是基于客户/服务器模式,也就是组播树的构造、维护等都是由服务器完成的,通过引入功能强大的服务器使得组播树的构造可以简单化。

多树模型中构造了多个组播树,可以使每个节点都进行了数据的转发,降低了对节点的带宽要求,平衡了树的负载,同时减少了节点失效对系统的影响。但多树模型的缺点也很明显:需要同时维护多个组播树,系统开销增大,而且要解决多路同步问题。

### 3.2 网状模型

网状模型中没有明确的定义节点之间的关系,每个节点维护了其它部分节点的信息,按照一定数据调度算法从多个节点或服务器下载数据,属于多对一的传输方式。我们以 Bullet 模型<sup>[7]</sup>和 DoNet<sup>[8]</sup>模型为例来介绍网状模型。

Bullet 实际是基于树状模型提出的一种改进模型,目的是通过垂直下载 (Perpendicular downloading) 技术来增大通过树获得的带宽。Bullet 模型中的节点首先构建一组播树,每个节点根据其孩子节点 (Children Nodes) 的带宽能力给它们发送一组数据块,然后按照一定的算法快速寻找能给它们提供所缺数据的伙伴节

点,从这些节点下载剩余的数据。我们举个例子说明数据传输过程,如图 4 所示,节点 S、A、B、C、D、E 构成组播树,S 为服务器(称为源节点)。每个节点根据自身带宽大小从其父节点接收一组数据,如:节点 D 由于带宽限制从父节点 A 处单位时间只获取了三个数据块 1、2、5,然后寻找到能给它提供所缺数据的两个节点 C、E,从 C、E 处下载剩余数据块 3、4、6。

Bullet 模型通过对树状模型的改进,使得每个节点不仅仅从父节点获取数据,还可以通过“垂直链路”从伙伴节点下载数据,使数据传输带宽大大增加。同时,Bullet 模型设计了高效、分布式算法,能够使数据块均匀的散布在所有参与的节点中,平衡了树的负载,还可以避免定位最后一个数据块的问题,取得持续高带宽的数据发送。

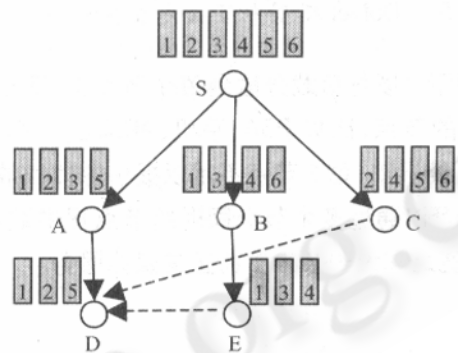


图 4 Bullet 模型数据传输过程举例

DoNet 模型中节点关系可以如图 5 所示。如:节点 E 从 F、B、H 三个节点获取数据。为了选择从哪些节点来获取数据,DoNet 模型中每个节点要维护伙伴节点的信息,为此每个节点定义了一个唯一标识符,同时在缓存中保存了其它节点的标识符列表。对于新加入的节点,它首先请求服务器,源节点随机在其缓存中选择一个节点作为新节点的代理节点,新节点从其代理节点处获取其它节点标识符列表。为了适应动态的网络环境,节点之间会周期性的交换宣告自己存在的消息,每个节点依此来更新自己的标识符列表。

DoNet 模型中,节点的伙伴关系和数据传输路径并不固定,它们是根据缓存中的数据情况进行数据交换,因此节点之间需要知道相互缓存的数据内容。模型通过定义一个缓存映射 (Buffer Map) 来表示节点中是否拥有某个数据片断。例如:BM 用 120 个比特表

示,每个比特对应一个数据片断,值为 1 表示有这个数据片断,0 表示没有。但 DoNet 模型是半同步系统,不同节点的 BM 中表示的数据不是完全一样的,因此用另外 2 个字节表示 BM 中第一个数据片断的序列号。每个节点和它的伙伴节点之间通过连续的交换 BM 来了解彼此缓存的情况,然后根据一定的数据调度算法从对应的伙伴处来获取某个数据片断。

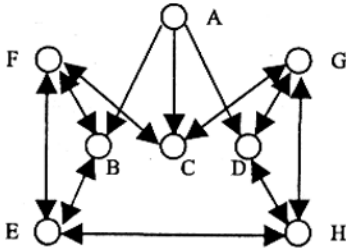


图 5 DoNet 模型中节点关系举例

网状模型可以把负载分散到每个节点上,适用于能力比较低的节点,比如 PDA、手机、ADSL 上网的 PC 机等,在这些情况下,单个节点不足以提供媒体回放所需的带宽  $R$ ,可以通过多个节点同时给节点提供数据,使其上行带宽之和大于  $R$  来支持正常媒体服务。

#### 4 结论

本文对 P2P 方式下流媒体系统模型进行了总结,我们把它分为两类:树状模型和网状模型。树状模型是一对多的传输方式,而网状模型是多对一传输方式。树状模型的关键问题是组播树的构建和维护:单树模型中组播树构造简单,但因每个节点只有一个父节点,父节点对子节点的影响很大,并且组播树中大量的叶子节点只接收数据,不再提供上传数据的服务,容易导致树负载的不平衡;多树模型可以在一定程度上解决单树模型带来的问题,但系统构造比较复杂,开销较大。网状模型相对于树状模型可以更好的适应 P2P 的网络环境,保证流媒体服务质量,但同时存在占用较大缓存,时延较大等问题。

#### 参考文献

- 1 龚海刚、刘明、毛莺池等, P2P 流媒体关键技术的研究进展[J], 计算机研究与发展, 2005; 42(12): 2033-2040.
- 2 武健. 流媒体技术原理及应用. 太原大学学报[J], 2005; 6(1): 47-51.
- 3 H. Deshpand, M. Bawa, H. Garcia - Molina. Streaming live media over a peer - to - peer network. [R]. Stanford Database Group Technical Report, 2001; 30.
- 4 Duc A. Tran, Kien A. Hua, Tai Do. ZigZag: An efficient peer - to - peer scheme for media streaming. [C] In: Proc. of IEEE INFOCOM, 2003.
- 5 M. Castro, P. Druschel, A. Kermarrec, et al. Split-Stream: High - bandwidth content distribution in a cooperative environment [C]. In: Proc. of IPTPS, 2003.
- 6 V. N. Padmanabhan, H. J. Wang, P. A. Chou, et al. Distributing streaming media content using cooperative networking [C]. In: Proc. of NOSSDAV, 2002; 177-186.
- 7 Kostic D, Rodriguez A, Albrecht J, et al. Bullet: High bandwidth data dissemination using an overlay mesh [C]. In: Proc. of the 19th ACM SOSP, 2003; 282-297.
- 8 X. Zhang, J. Liu, B. Li, et al. CoolStreaming/DoNet: A data - driven overlay network for efficient live media streaming [C]. In: Proc. of INFOCOM, 2005; 3: 2102-2111.
- 9 X. Jiang, Y. Dong, D. Xu, et al. GnuStream: A P2P media streaming system prototype [C]. In: Proc. of the 4th ICME, 2003.
- 10 M. Hefeeda, A. Habib, B. Botev, et al. PROMISE: A peer - to - peer media streaming using CollectCast [C]. In: Proc. of ACM MM, 2003.