基于P2P和CDN的流媒体直播系统的设计与实现

任立勇[1] 王 焘[1] 段翰聪[1] 周 旭[2]
(电子科技大学计算机科学与工程学院 成都 610054)[1]
(中国科学院声学研究所高性能网络实验室 北京 100080)[2]

摘 要 对现有流媒体播放系统的相关技术进行了分析比较,指出了各自的优点以及存在的问题。结合CDN与P2P两种技术的优点,改善传统内容分发网络拓扑结构,把P2P的扩展能力和CDN的可靠性、可管理性有效地结合起来,设计并实现一种效率高、可扩展性好、稳定性强、可管理的流媒体直播系统。并且对系统的节点管理、缓存管理、数据调度等关键技术进行了介绍,最后对该系统的特性进行了分析。仿真实验表明,在大规模网络环境中,该流媒体直播系统比单纯的P2P系统在性能上有明显的提高。

关键词 对等网,内容分发网,流媒体,直播

Design and Implement on a Live Media Streaming System Based on Peer-to-Peer and CDN

REN Li-yong[1] WANG Tao[1] DUAN Han-cong[1] ZHOU Xu[2]
(Dept. of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China)[1]
(High Performance Network Lab in Institute of Acoustics, Chinese Academy of Sciences, Beijing 100080, China)[2]

Abstract Current technologies adopted by live media streaming system were compared and analyzed, and many problems of these technologies were proposed. To solve the problems of exiting live media streaming systems, combined with the merits of CDN and P2P network architectures, this paper proposed and implemented a new live media streaming system which improves the network topology of traditional content distribution networks. And a solution for critical issues was given, such as the user peer management strategy, buffer management strategy and data schedule scheme, then the merits of this new live media streaming system were analyzed. Simulation studies show that, in large-scale networks environment, the proposed system has improved performances than the current exiting P2P live media streaming systems.

Keywords P2P, CDN, Media streaming, Live

1 前言

随着Internet的飞速发展,人们的工作和生活有了全新的改变,网络流媒体技术作为Internet的主要应用之一,日益成为研究的热点。Internet上的传统流媒体系统大多采用C/S(Client/Server)模式[1],由于传输流媒体占用的带宽大,持续时间长,而服务器可利用的网络带宽资源有限,加之其处理能力、缓存大小、I/O速率等因素的影响,在有大量用户请求获取数据的情况下,服务器将不堪重负,成为整个系统的瓶颈。为了改善网络效率,IP组播技术[1]被加入TCP/IP协议簇,通过路由器复制数据包,避免数据在链路上冗余传输,取得较高的网络效率。然而,由于协议的复杂性以及拥塞控制、可靠性管理等方面的不足[2],IP组播应用极其困难。CDN[3]使用户能够从位于本地的服务器上获取流媒体文件,从而提高用户访问性能,并减轻骨干网络流量,同时增加系统容量,但由于整个系统依然受到C/S架构的因素制约,代理服务器能够提供服务的能力有限,同时增加代理服务器数量的建设成本昂贵。单纯的P2P[4]流媒体分发系统虽然成本低、可扩展性好,但由于P2P系统的开放性、匿名性、节点不为自身行为承担责任等特点,导致系统服务质量(QoS)严重下降,更有甚者,恶意节点滥用P2P资源传播广告、病毒等文件来危害其他节点[5],同时P2P应用的盛行带来网络流量风暴、监管缺失、涉及版权等问题。P2P和CDN在几个关键点上存在着互补的特点,如果能将两种技术有效地结合起来,在CDN网络中引入P2P技术,必然是一种更加完善的系统。通过这种模式,可以在不增加成本的同时有效提升CDN服务能力,有效避免P2P应用的诸多弊端。

2 系统结构设计

系统的核心设计思想是在CDN网络中引入P2P自治域,由单个或若干个缓存服务器与其覆盖的用户节点作为对等节点,共同构成一个P2P自治域。在域内利用P2P技术实现资源共享,而自治域之间的用户节点不发生流量交换。并且将CDN的缓存服务器以P2P的方式组织,利用P2P的目

到稿日期:2008-08-25 返修日期:2008-10-12 本文受中国下一代互联网示范工程(CNGI-04-12-1D)资助。
任立勇 博士,副教授,主要研究方向为高性能网络、P2P网络,E-mail:lyren_cs@uestc.edu.cn;王 焘 硕士,主要研究方向为P2P网络、多媒体网络;段翰聪 博士,主要研究方向为P2P网络、多媒体网络;周 旭 博士,研究员,主要研究方向为并行计算、P2P网络。

录服务和多点传输能力，实现 CDN Cache 设备之间的内容交换，提升 CDN 的内容分发能力。同时将 CDN 的管理机制和服务能力引入 P2P 网络，形成以 CDN 为可靠的内容核心，以 P2P 为服务边缘的架构，通过这种架构可以在不增加 CDN 成本的同时有效提升其服务能力，实现了 CDN 技术与 P2P 传输的有效结合。

系统拓扑结构如图 1 所示，网络架构采用 3 层结构，分为控制层、中心层和应用层。在控制层部署路由选择服务器及服务管理中心，实现对系统进行统一管理；中心层部署缓存服务器和索引服务器，实现内容的分发和传送，节点发现及引导互连；而应用层实现 P2P 的内容服务、视频发布与观看等功能，并通过缓存服务器和索引服务器进行管理、控制和服务保障。

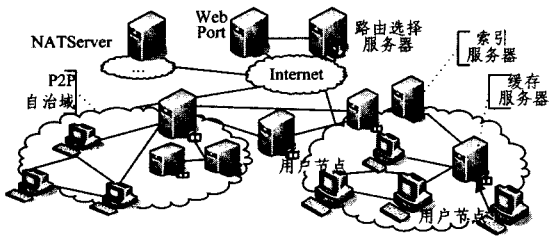

图 1　系统拓扑结构图

路由选择服务器：接受用户节点下载或上传流媒体的请求，返回其所在域内的索引服务器地址。接受索引服务器获取流媒体的请求，返回给具有相应数据的其他域缓存服务器的信息列表。

索引服务器：索引当前本域内具有流媒体数据的在线用户节点，管理节点的位置信息，为节点加入时提供域内节点的信息列表。当接收到节点的请求，如果所请求的流媒体数据已经在本域中某台缓存服务器中存在，则记录下该节点信息，返回其所需要的节点信息列表，并将请求转发到相应的缓存服务器为其提供服务。如果域内的缓存服务器中没有需求的数据，则访问路由选择服务器，获取其他域中具有此数据的缓存服务器信息列表，根据当前状况决定启动某台负载低的缓存服务器并将列表传递给它，使缓存服务器间以 P2P 方式建立连接，获取该数据。

缓存服务器：从用户节点或其他域的缓存服务器获取流媒体数据，作为域内数据源为普通用户节点提供服务。在普通节点看来，缓存服务器也是 P2P 网络中的一个对等节点。

用户节点：从域内其它节点获取流媒体数据，或者发布流媒体数据到缓存服务器。首先访问路由选择服务获得 PeerID 和域内索引服务器地址。获取数据时，通过索引服务器获得域内节点信息列表，节点间互连以 P2P 方式共享下载数据；发布数据时，请求域内索引服务器分配一台缓存服务器，与其建立连接并上传数据。

3　关键技术与策略

3.1　节点管理策略

节点管理最基本的思想是提高覆盖网络与底层物理网络的匹配程度，使得其上层网络拓扑和节点之间的物理网络拓扑尽量匹配，达到在流媒体网络中相邻近的节点在物理网络中也邻近的效果，从而有效减少流媒体数据传输的负载压力和端对端传输延迟。

3.1.1　节点标识(PeerID)的生成

Internet 上所有结点对之间的信息是巨大的，难以存储和计算，但可以依据 ISP、自治域或地址前缀等信息将其划分为互不相交的区域，每个区域都为一组 IP 地址的集合。PeerID 数据结构如图 2 所示，用 56 位二进制值表示，利用固定的位来分别表示国家、网络类型、地区、城市等信息。

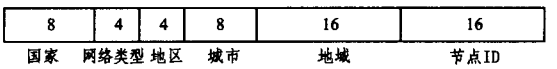

图 2　PeerID 各字段的表示

路由选择服务器中包含一个 IP 地址到地理位置信息转换数据库，该数据库中包含了 IP 地址的地理信息，其中，网络类型根据国内的七大网络运营商进行划分，节点 ID 字段随机生成。当系统中有新节点请求加入，为请求节点生成一个 PeerID 以唯一标识此节点。

3.1.2　用户节点的加入

根据 SCAMP[6] 协议，为了保证节点拥有的网络局部视图同覆盖网规模保持一致性增长，根据式(1)可计算出维护的覆盖网局部视图大小。

$$ViewNum=(c+1)\times\log N \tag{1}$$

其中，$ViewNum$ 为节点局部视图大小，N 为覆盖网节点总数，c 为参数。当用户节点加入网络的时候，从索引服务器获得域网络规模大小，此后，当索引服务器发现节点数量发生一定规模变化时，将网络规模信息通知所有节点，使各个节点能够动态调整局部视图大小。

用户节点中管理的节点分为成员节点和伙伴节点，分别在成员列表和伙伴列表中管理。成员节点只保存节点相关的主机信息，而没有建立网络连接，成员列表由用户加入网络时访问索引服务器所获得的节点列表消息进行初始化。伙伴列表是成员列表的子集，选取其中与自己 PeerID 接近的节点建立连接，进行数据的交互。

用户节点加入网络时，首先访问路由选择服务器，获得域索引服务器的地址和自己的 PeerID，然后向索引服务器发出请求，获得域内缓存有相应流媒体数据的节点列表，加入到成员列表中。系统运行过程中，成员列表会被动态更新，用户节点通过自己向伙伴节点周期性地发出请求以获得新的节点信息并将其加入成员列表中。同时，定期向索引服务器发送心跳报文汇报保活信息。

用户节点从成员列表中选取节点加入伙伴列表时，先从与自己 PeerID 中国家、网络类型、地区、城市、地域字段完全相同的节点中选，如果数量不够，再从国家、网络类型、地区、城市字段完全相同而地域字段不同的节点中选，依次类推。向伙伴列表中的节点发起连接请求，节点之间建立连接就形成一个网络拓扑结构，如图 3 所示。

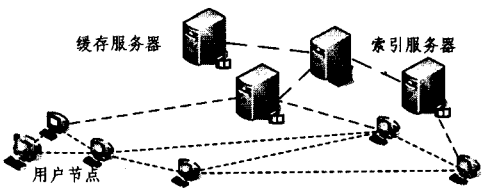

图 3　域内用户节点连接拓扑结构

3.1.3　用户节点的退出

当一个用户节点正常退出时，向伙伴节点发出退出请求，

收到请求的节点将其信息从自己的伙伴列表和成员列表中删除。同时，节点周期性地向伙伴节点发送心跳消息，这样如果用户节点在没有发出退出请求的情况下非正常退出，与其连接的节点在一段时间没有收到心跳信息，就会把其从自己的伙伴列表和成员列表中删除。

3.1.4 节点信息的动态更新

当收到其他节点的连接请求时，如果连接数小于由式(2)计算出的门限值，则接受连接请求，建立伙伴关系，将其信息加入伙伴列表和成员列表。如果超出了门限值，则将自己的伙伴列表发送给请求节点。

$$Limit=BW/S-1 \tag{2}$$

其中，$Limit$ 为设定门限值，BW 为节点的可用带宽，S 为流媒体数据的编码率。

周期性地对节点当前的成员列表进行监控，如果发现当前成员数量少于 $3\log N$(N 为域内节点数量)，则向当前的伙伴发送请求，获取其伙伴列表信息，添加到自己的成员列表当中。当发现当前的成员数量大于 $3\log N$ 的时候，会淘汰部分节点，使成员数量基本保持在 $3\log N$。同时，定期根据伙伴的性能来动态替换表现不佳的节点，并从成员列表中随机选取相应的节点作为新的伙伴，使其伙伴数量基本保持在 $\log N$。其中伙伴性能评估指标由式(3)得出。

$$PW=((1-p)\times upload+p\times download)/durTime \tag{3}$$

其中，PW(Peer Weight)为节点的权值，$upload$ 为上传数据包的数量，$download$ 为下载数据包数量，$durTime$ 为经历时间，p 为节点下载所占的权重($0<p<1$)。

分数低的伙伴被淘汰替换，每次淘汰的数量为：$k*curNo$，其中，$curNo$ 为现在伙伴列表中节点数量，k 为参数，$0<k<1/2$。由于保持连接的节点数量波动比率为 k，可以保证媒体播放连续性不因为节点的替换而受影响。

3.2 缓冲区的管理

3.2.1 数据的表示

系统是基于 Gossip 协议的 P2P 流媒体网络，数据的传输方向不固定，节点从一个或多个伙伴节点中获取自己需要的数据，同时向伙伴节点提供自己拥有的数据。这样节点之间根据各自缓存数据情况进行数据交换，流媒体数据分割成相同时间的片段，用一个缓存映射来表示节点中是否拥有某个片段数据，节点之间交换数据是通过检查彼此的缓存映射来进行的。

每个节点都要缓存一定大小的数据，缓存的数据越多数据传输的可靠性越能得到保证，但是和服务器的延迟会越大。同时，缓存区分为若干个数据片段，分别向伙伴节点索取，片段越小描述片段的头部信息所占数据的比重越大，导致节点间链路开销增大。相反，若片段太大就无法分配较小的数据量，降低了获得数据的及时性。缓存区用一个滑动窗口来存储 256 个数据片段，每个片段大约是 200ms 数据片，用 256 个 bit 的 Bitmap 来表示这 $256\times200ms=51.2s$ 数据，bit 值为 1 表示该节点拥有此片段，为 0 表示没有。缓存区存储固定时间长度的数据片，占用的空间大小呈动态变化。为了保证系统中表示的数据同步，用两个字节来记录 bitmap 中第一个片段的序号，每个片段序号是发布节点、发布数据时打的标记。

3.2.2 缓冲区的组织

缓冲区组织结构如图 4 所示，每个片段是 200ms 的数据量，整个窗口每 200ms 向前滑动一次。Bitmap 中被移出缓冲区的那个 bit 被认为是“已过期”的数据，本地不用再缓存它，新移入的那个 bit 被认为是自己现在还没有而期望缓冲的数据，初始的时候缓存映射为全 0，随后它将从伙伴节点获取若干片段把相应位置为 1。

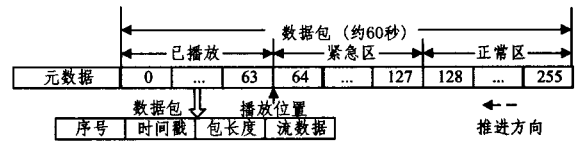

图 4 缓存区结构

当缓存映射最前面有 48 个连续的 1 时，启动滑动机制，这样每个 bit 在本地缓冲区至少有 $48\times200ms=10s$ 的存活时间，在它的存活期内，可以给其伙伴节点提供该 bit 的数据，同时任何一个 bit 至少有 10s 的时间用来从网络中搜索该 bit 的数据并获取它。在每个 Gossip 周期，节点会将本地的 Bitmap 信息状况写入一张表内，记录了每个片段在该节点保存的情况，节点之间就是通过这张表的传递来相互感知对方数据存储的情况。

3.3 数据调度

3.3.1 请求数据片的数量

节点根据自己和伙伴节点的 Bitmap，从伙伴获取所需要的数据。向伙伴节点请求数据片的数量是依据邻居结点上次完成的情况周期性动态计算，由式(5)得出。

$$H=\frac{N}{M/k} \tag{4}$$

其中，M 为伙伴节点数量，N 为需要数据片的总数，H 为每次请求数据片的上限，k 为参数($1<k<M$)。

$$CurRqst=\begin{cases}\min\{2\times R,H\} & (F=R)\\ \max\{L,F-(R-F)\} & (R/2\leqslant F<R)\\ 0 & (0\leqslant F<R/2)\end{cases} \tag{5}$$

其中，R 为上次请求的数目，F 为上次完成的数目，$CurRqst$ 为本次将请求的数目，H 为每次请求数据的上限，由式(4)得出，L 为每次请求数据片的下限($1<L<H$)。

3.3.2 选择数据块

为了保证数据的及时性，把缓冲窗口分为 3 个部分，如图 5 所示。当前播放位置之前的数据供其他节点索取，之后 1/3 为紧急区，区中数据必须立即获取，此后为正常区，依据最少优先的选取策略[7]请求。即首先选择伙伴节点数目最少的片，如果遇到多个片有相同的伙伴节点数目就随机选择，这样使得数量少的片段能够增多，从而在网络中达到平衡。

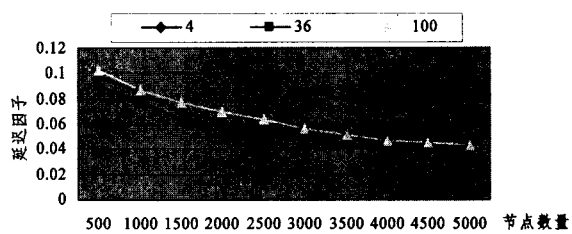

图 5 节点间延迟

4 仿真实验

采用基于周期驱动的 P2P 网络仿真系统 Peersim 模拟大规模网络环境，实现了覆盖网建立和节点之间数据交换协议。

4.1 数据传输延时

数据传输延迟是流媒体数据在伙伴节点间交换数据时从一个节点到另一个节点的延迟，该性能反映出系统中数据传输的实时性，延迟越小系统的实时性就越高，节点能及时获取请求的数据，系统的 QoS 就能得到保证。

如图 5 所示，纵坐标用节点延迟因子来表示数据传输的延迟，这是在模拟实际物理网络中节点间数据传输的平均时间得到的数值，延迟因子越小，系统延时就越小，实时性越高。横坐标表示域内节点数量，模拟分别在 4，16，36，64，100 个物理网络区域中，分布从 500 到 5000 个节点的情况。从图中可以清晰看到，每个区域内节点数目多少对数据传输的延迟影响并不大，随着网络中节点数量的增加，节点的数据传输延迟会明显减少，这是由于覆盖网中观看节点的伙伴节点优先选取和自己处于相同区域内的节点，当网络中节点的规模增大，网络距离非常接近，数据传输延时较小。

与未采用 CDN 的基于 Gossip 协议的 P2P 流媒体分发系统进行比较，如图 6 所示，当在具有 16 个区域的物理网络环境中，采用 CDN 的 P2P 系统中具有 500 个节点时延时因子只有 0.1 左右，并且随着节点数量的增加而逐步减小，而未采用 CDN 的 P2P 系统却始终在 0.4 到 0.5 之间。

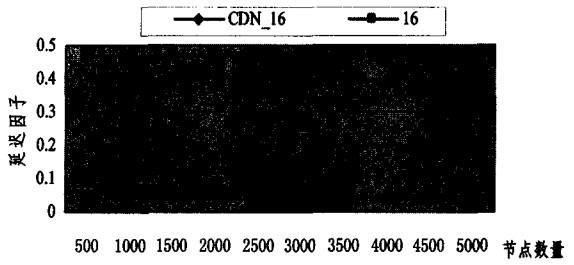

图 6 节点间延迟比较

4.2 区域间流量

系统根据底层物理网络划分为若干个区域，区域以骨干网为边界，而节点之间数据传输的流量一部分就分布于区域内部，一部分分布于区域之间，这样区域间的流量就构成了骨干网上的流量。当主干网的带宽资源占用过大会影响整个网络系统的性能，因此要尽量减少主干网络带宽的占用。每个节点需要观看节目，就需要占用编码率的带宽 m，整个系统需要 $m\times N$ 的带宽（N 为节点数），主干网带宽占用可以通过统计单位时间内区域间数据的传输量得到，从而可以计算出主干网带宽占用总共需要带宽的比例，比例越小，占用主干网络的带宽越少，网络的性能会越好。如图 7 所示，相同数量的节点分布在越多的物理区域，区域间流量会越大，同时，随着网络中节点数量的增加，区域间流量的比例会减少，最后达到一个平衡状态。

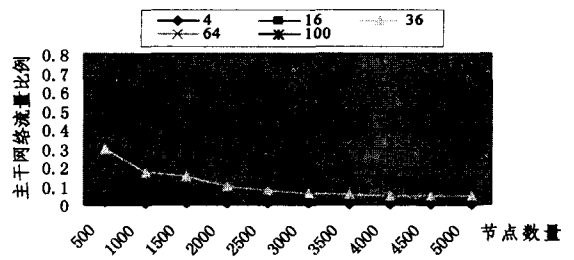

图 7 主干网流量分布

如图 8 所示，采用 CDN 的 P2P 系统根据底层物理网络划分为若干个区域的选点策略，与未采用 CDN 的 P2P 系统相比，当节点分布在 4，16 个区域时，区域间流量明显减少，大大缓解了主干网络带宽的压力。

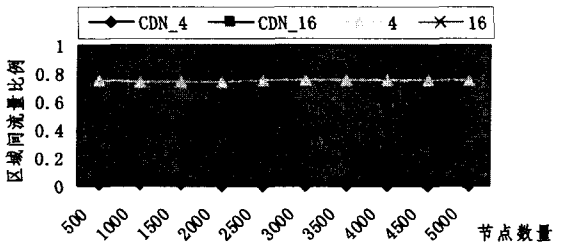

图 8 主干网流量比较

结束语 P2P 技术充分利用了用户的闲置上行带宽，提高了流媒体传输效率，这样就大大降低了边缘服务器的压力，可以通过更少的边缘服务器，提供更多的业务量为更多的用户服务，降低了原来 CDN 模式的成本。通过对网络进行 P2P 自治域的划分，将 P2P 的流量严格限制在同一区域内，避免骨干网上的流量无序性和风暴，可以实现对用户的监控、流量的监管，增强了网络的可管理性。采用分布式系统体系结构，可方便地为自治域添加缓存服务器，实现域的扩容，同时只要在路由选择服务器中简单地更改域划分设置，即可增加新域，轻松满足整个系统扩容的需求。鉴于 P2P 和 CDN 网络技术优缺点的互补性，本文把 P2P 的扩展能力和 CDN 的可靠性、可管理性结合起来，设计并实现一种效率高、可扩展性好、稳定性强、可管理的可靠的流媒体直播系统。另外，用户节点和缓存服务器中缓存替换算法、缓存服务器间的协作机制、安全认证机制等内容还需要进一步的研究。

参 考 文 献

[1] Stephen E D，Deborah E，Dino F，et al. An Architecture for Wide-area Multicast Routing[J]. IEEE/ACM Transaction on Networing，1996，4(2)

[2] 刘亚杰，窦文华. P2P 流媒体：一种新型的流媒体服务体系[J]. 计算机科学，2004，31(4)

[3] Kangasharju，Jussi A T. Internet Content Distribution[D]. April 2002

[4] Banerjee S，Bhattacharjee B，Kommareddy C. ScalableApplication Layer Multicast[C]//ACM Sigcomm 2002. 2002，8

[5] 候孟书，卢显良，等. P2P 系统的信任研究[J]. 计算机科学，2005，32(4)

[6] Ganesh A，Kermanec A-M，Massoulid L. Peer-to-peer membership management for gossip-based protcols[J]. IEEE Transactions on Computers，2003，52(2)

[7] Baldoni R，Piergiovanni S T. Group Membership for Peer-to-Peer Communication[C]//Dipartimento di Informatica a Sistemistica Universitpa di Roma "LaSapienza". Via Salaria 113，00198 Roma，Italia