

基于 P2P 的点播服务系统的设计

孙龙军, 高 岭, 牛建涛

(西北大学 信息科学与技术学院, 陕西 西安 710127)

摘要: P2P 技术是一种比较成熟的技术, 在实际中有着广泛的应用。通过分析研究常见 P2P 技术的应用即视频直播和文件传输, 结合实际需求将视频点播系统与 P2P 技术相关联, 提出了一种基于 P2P 技术的边下边看的视频点播系统。阐述了系统的构成, 对服务器端视频文件格式, 客户端结点的选取和存储空间的开辟等问题进行了分析讨论。

关键词: P2P; Open - while - Downloading; BT; 视频点播

中图分类号: TP393

文献标识码: A

文章编号: 1673 - 629X(2008)12 - 0053 - 03

A Design of Video on Demand System Based on P2P Technology

SUN Long-jun, GAO Ling, NIU Jian-tao

(Information Science and Technology Institution, Northwest University, Xi'an 710127, China)

Abstract: The P2P technology is one kind of quite mature technology, and has the widespread application in reality. Analyzes the P2P technology's application which was the live video streaming system and the file transfer, combines the realistic demand which is connected the video on demand system and the P2P technology, and proposes one kind the video on demand system which is open - while - downloading based on the P2P technology. Elaborates system's constitution, to the server end video document format, customer end node's selection and storage space's questions and so has carried on the analysis discussion.

Key words: P2P; open - while - downloading; BT; VOD

0 引言

近年来, 基于 Internet 音视频传输的流媒体应用和业务得到了迅速的发展^[1]。目前针对影音文件的视频点播 (Video - On - Demand) 服务已经成为了现在互联网主要应用之一。

现在绝大多数的视频点播系统都是采用 C/S (客户机/服务器) 模式^[2], 在这种模式下, 每一个用户的点播服务请求服务器都会单独来响应并与其建立单独的通信。在这种情况下即使是使用处理能力超强的服务器, 也只能响应处理有限的用户请求, 同时当用户到达一定数量时, 服务器很难向所有用户提供有保证的服务。如果只是通过靠服务器机群来解决, 在这种模式下随着网络用户的积聚增加使得视频服务器的负载增大, 将会导致较大的成本与管理开销^[3]。为了改变现有的视频点播系统的问题, 就必须提出一种有别于 C/S 的系统结构。

1 P2P 技术和应用的研究分析

P2P 就是“Peer - To - Peer”, Peer 意为“(地位能力等) 同等者”或“伙伴”等, 因此 P2P 可以理解为“伙伴对伙伴”, 或称为点对点式对等网络。P2P 提出了一个全新网络传输理念: 这个网络中不存在中心服务器, 每一个 Peer 的权利和义务都是对等的; 这样就打破以往通过服务器或浏览下载的 C/S 模式。在 P2P 模式中, 所有网络上的设备都可以建立 P2P 对话, 信息的需求者同时也是该信息的提供者, 同一信息的访问者越多, 速度就越快, 这使得信息传输的速度和效率得到了极大的提高^[4]。正是因为 P2P 网络中不存在中心服务器和每一个 Peer 的权利和义务都是对等的特点, 所以 P2P 网络具有如下优势: (1) 负载均衡; (2) 丰富的信息资源; (3) 冗余和容错; (4) 基于内容的寻址; (5) 有效的搜索; (6) P2P 相对于 C/S 结构有更多的优势^[5]。

P2P 从诞生以来就迅速风行, 目前, P2P 应用已经占据了整个互联网流量的 2/3 以上, 它被应用于广泛的领域, 例如文件共享、协同工作、即时通讯、文件传递等。其中主要应用是文件共享。在文件共享中它又可以分为文件下载和视频服务。

收稿日期: 2008 - 03 - 12

基金项目: 陕西省自然科学基金项目 (2005f36)

作者简介: 孙龙军 (1980 -), 男, 陕西西安人, 硕士研究生, 研究方向为网络应用; 高 岭, 教授, 研究方向为网络应用。

1.1 文件下载技术

BitTorrent(BT)为 P2P 网络目前最热门且有效的分布式文件传输技术之一,并被称为第三代的 P2P 技术。BT 在 2001 年由美国软件工程师 Bram Cohen 发明,它跟传统的 P2P 有两大区别:没有中央搜寻系统,多点对多点传输。BT 也没有采用分散服务器来追踪文件的搜寻,因此在网络上不会造成大量的搜寻流量,BT 是利用一个用 torrent 为扩展名的文件,将上载者(Seed)和下载者(Peer)连接起来,BT 的另一特点是多点对多点传输。传统 P2P 虽然将服务器的工作分流,但仍然未脱离单点对单点的下载方式,如果 Peer 增加就需要排队下载,仍然会降低下载效率,而 BT 则利用了用户在下载时没用到的上载带宽,在下载的同时也进行上传的动作,换句话说,同一时间的下载者越多,上载者也越多。它的基本思想是将一个大的文件分成许多小的片断,下载用户非顺序地从网络中其他用户处下载文件的各个分片。当用户下载完某个完整的分片后,又可将该分片共享,供其他用户下载。与传统的 C/S 方式文件下载相比,这种 P2P 下载方式的特点是用户越多下载速度越快,可以最大限度地发挥网络传输能力。

1.2 视频服务

基于 P2P 的直播系统主要采用 C/S 模式。每一个客户端都有一个程序来实现以下主要功能:

1)将该客户共享的视频资源分块编号并且将相关信息通过 IP 多播方式发送数据信息包;

2)对本机的网络能力、处理能力等做出对应的服务质量评估并发送给服务器;

3)监控本机播放视频的缓存使用情况。

服务器端要完成的主要任务有:

(1)提供用户注册、视频等基本服务;

(2)接收用户端信息,将所有信息归类并及时更新;

(3)当用户点播某视频时,服务器向用户发送该视频提供者的用户群信息。

例如:目前常见的 P2P 直播软件都是使用 WMV 格式的视频文件,通过模拟 HTTP 服务器的方式,把 WMV 数据流传送给 Media Player 播放器来实现直播的。在服务器端需要将 Media Encoder 发出的 HTTP 数据进行拆包,然后组成 30 秒~1 分钟的 P2P 数据块,通过 P2P 方式将数据块发送给客户端,客户端模拟成 HTTP 服务器将传输完成的数据块发送给 Media Player 来实现观看^[6]。

2 P2P 视频点播应用的设计

P2P 网络的优势为解决传统点播系统中的问题提

供了一个途径,在这里参考 BT 下载技术的优势和直播系统的设计模式提出一种将文件下载系统与视频直播系统相结合的边下边看(Open - while - Downloading)^[7]的视频点播系统。如图 1 所示。

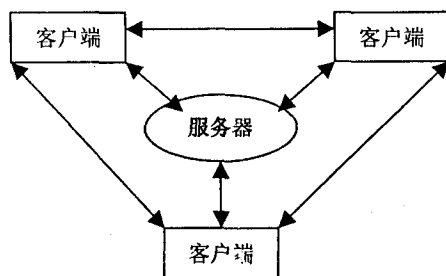


图 1 P2P 视频点播系统构件关系图

如图 1 所示,此系统采用 C/S 模式,由服务器端和客户端组成。服务器端可以与客户端进行信息传输,同时客户端与客户端之间也可以相互传输信息。视频点播的服务器端是一个文件服务器,其主要功能是用来存放影音文件和维护用户列表。视频点播的客户端主要功能就是根据用户的需要下载所要影音文件并将影音文件送入视频播放器中。

2.1 服务器端的设计

传统的视频文件并不像直播系统服务器发出的经过 Media Encoder 编码的 HTTP 数据那样可以单独播放。在点播系统的服务器端不能存放原始的视频文件,必须对原始的视频文件进行改造,使其和经过 Media Encoder 编码的 HTTP 数据一样可以单独播放。对原始文件进行时间分片处理,如图 2 所示。

原始的视频文件									
0	1	2	3	4	5	6	7	8	9

图 2 服务器端视频文件分片图

如图 2 所示,原始的视频文件被分成了 10 时间分片,分出来的每一个时间分片都能单独播放。在进行时间分片的过程中需要对每一个时间分片都加入其原始文件中的头信息、结尾信息或索引信息等关键数据。然后将这些时间分片和分片详细情况表组合在一起形成一个新的文件,存放在服务器上。时间分片的大小决定了点播系统的精确度和系统运用 P2P 技术运转所需的最小用户。点播系统精确度的大小由时间分片的大小所决定,它们之间成反比,即时间分片越小点播系统的精确度越高,时间分片越大点播系统的精确度越低。系统运转的最小用户与时间分片成反比,即时间分片越小,所需要的用户越多;时间分片越大所需要的用户越少。所以点播系统的精确度和系统运转所需的最小用户成反比。利用 BT 文件下载技术为将时间分片和时间分片列表的详情组合的视频文件生成一个 torrent 为扩展名的文件。当用户要点播某个节目

时,服务器端将相应节目文件的 torrent 扩展名的文件和观看的用户列表发送给客户端。服务器端定期地向用户发送消息来确定用户是否还在网和用户观看的文件情况,来更新服务器端的用户列表。

2.2 客户端的设计

点播系统的客户端接收到服务器端发送来的 torrent 扩展名的文件和用户列表,按照用户的需要从服务器端或客户端下载所需要观看的时间分片文件,客户端不采用 BT 软件或边下边看软件那样将整个文件都下载到磁盘上的方式,这种方式会在点播系统中对用户造成大量不必要的开销。点播系统的客户端的用来存储下载文件的区域由三部分组成,如图 3 所示。

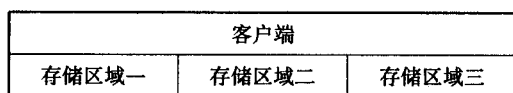


图 3 客户端存储空间图

如图 3 所示,存储区域的第一部分用来下载将要收看的时间分片,若下完了将其移入播放器的缓存中播放,开始下载接下来的时间分片文件;存储区域的第二部分用来存放刚看过的分片,为需要的用户提供下载服务;存储区域的第三部分用来存放所看视频文件中观看者最少的时间分片。这一部分是为了使文件保持完整性不至于在那些关键结点退出后使文件变得不完全或是造成网络拥塞。存储区域的大小根据 torrent 扩展名的文件中的时间分片列表来动态地开辟空间。

点播系统相对于直播系统的优势就在于点播系统可以随意拖动,而直播系统中无法进行拖动。当用户在播放器时间条上进行拖动时,根据用户拖动的位置来确定用户所需要观看的视频文件的开始时间,根据从服务器端接收到的时间分片列表信息,找到所对应的时间分片,下载该时间分片文件,来实现视频点播系统的拖动功能。

为了向用户提供有保证的点播服务,使用户在点播中不出现断流或停止等影响观看的现象,客户端从服务器端下载来视频文件的时间分片列表后,根据时间分片列表中的时间间隔大小和时间分片文件的大小,计算出下载每个时间分片文件的最小速度。从计算出的下载时间分片文件最小速度中选出最大的速度,作为为用户提供有保障的点播服务的最小速度。为了保证服务的最小速度,在下载结点的选取上采用带反馈机制询问方式。网络的负载情况是时时刻刻都在变化的,结点所能提供的带宽也是在不断变更的,在第一次下载时间分片文件的时候,随机地选取几个结点使它的下载速度大于算出的服务最小速度,下载最

初的视频文件,同时向其它结点发送消息,来得到其它结点的还可以利用的带宽。然后根据结点的反馈的信息将用户列表重新排列,将可利用带宽大的用户放到前面,形成新的列表。当下载接下来的时间分片的时候优先考虑这些排在前面结点。每隔固定的时间都要向所有没有选中的结点发送消息,来获取结点可利用带宽的反馈数据,重新排列结点列表。如果因为网络原因造成从结点列表按默认连接数选取结点无法达到服务最小速度的话,点播系统将在操作系统允许的范围内自动增加结点的连接数,使其达到最小速度,以保证点播观看的流畅。

3 结束语

通过对现有 P2P 直播技术和 BT 下载技术的研究与分析,提出了一种将两种技术相结合的视频点播系统。

点播系统充分利用了现有的 P2P 应用系统的设计原理,合理地将文件下载和视频直播相结合。在点播系统对传统的视频文件进行了改进,对其进行时间分片,成为可以单独播放的视频文件,为点播拖动提供了支持;在文件下载时只下载所看的视频时间分片并根据时间分片的大小动态地分配存储空间,为用户节省了存储空间;在结点的选取上采用了带有反馈的动态的结点选取方式,根据网络情况合理地改变连接数的多少,使网络的带宽可以得到充分利用。

参考文献:

- [1] 郭红方,林子松,王宗敏.基于 NGI 的 ALM 视频直播系统的实现[J].微计算机信息,2008(3):17-19.
- [2] 沈建强,耿兆丰,邹 轩.视频点播系统的设计与实现[J].计算机工程,2006,32(2):209-211.
- [3] 张伟文,金 鑫,吴国新.一种基于 P2P 的视频点播系统的设计[J].计算机技术与发展,2007,17(2):159-163.
- [4] 宋 芬,陆建德.c/s 和 p2p 相结合的视频点播系统的研究[R].苏州:苏州大学计算机科学与技术学院,2006:43-46.
- [5] 杨国燕,张国印.基于 P2P 技术的网络教学互动系统研究与设计[R].哈尔滨:哈尔滨工程大学,2007.
- [6] 陈添丁.使用 BT 技术测试视频点播系统的可行性[J].中国有线电视,2005(24):2392-2397.
- [7] Xiang Z, Zhang Q, Zhu W, et al. Peer-to-peer based multimedia distribution service[J]. IEEE Transactions on Multimedia, 2004, 6(4):343-355.