

基于网络计算机的流媒体播放器的研究与实现

樊承泽, 陈蜀宇, 杨新华

(重庆大学 软件学院, 重庆 400044)

摘要:针对目前网络计算机等网络终端设备在处理流媒体信息时没有专门的播放软件,且缓冲区长度设置不合理和占用网络带宽较大的问题,描述了一个以网络计算机为基础平台的流媒体播放器技术实现方法;利用RTSP协议,实现了对网络流媒体文件播放的实时控制功能。实践证明该播放器不但能良好地播放网络中的各种流媒体文件,而且占用较小的网络带宽,在设备资源的占用率与流媒体播放效果之间做出了良好的平衡,在一定程度上提高了资源受限的网络终端设备对大容量流媒体资源的处理效果。

关键词:流媒体;RTSP;网络计算机;播放器;缓冲区设置

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2010)04-0195-04

Research and Implementation of Streaming Media Player Based on Network Computer

FAN Cheng-ze, CHEN Shu-yu, YANG Xin-hua

(College of Software Engineering, Chongqing University, Chongqing 400044, China)

Abstract: Aiming at the current network computer terminal device information in dealing with streaming media player without special software, and the length of the buffer zone set up unreasonable and occupied by the larger issue of network bandwidth, describes streaming media player technology method that based on network computer; Use RTSP protocol to achieve the streaming media files on the network playing real-time control functions. Practice has proved that the player will not only play well in a variety of network streaming media files, but also occupy less network bandwidth, so it made a good balance between equipment resources in the occupancy rate and the effects of playing streaming media files, also improved the effect between resource-constrained network computer and high-capacity streaming media resources.

Key words: streaming media; RTSP; network computer; player; buffer settings

0 引言

随着计算机网络的迅速发展,网络计算机(Network Computer, NC)作为一类基于网络环境的计算设备,最早出现于20世纪90年代,是一种在高速网络环境下的信息访问处理终端设备。它使客户终端和网络主设备等相结合,利用网络技术,提供各种实时/非实时、并行/串行、集中/分布式的计算和处理;具有易管理、低成本、安全等特点^[1]。

与此同时,网络中针对流媒体资源的各种应用也进入了一个新的时期。伴随着网络技术的逐渐成熟和

各种网络媒体服务(如在线电视、在线直播等)种类的不断增多,专用的网络计算机在提供传统信息处理功能的基础上也需要具有强大的流媒体资源处理能力。因此,描述了一个针对网络计算机环境下的流媒体播放器的实现方法。

1 流媒体传输协议的介绍

在流媒体播放技术中,流媒体传输协议占有十分重要的地位;RTSP(Real Time Streaming Protocol:实时流协议)是应用层协议,控制实时数据的发送^[2]。SDP(Session Description Protocol:会话描述协议),它的目的是为了描述在会话公告、会话邀请和其它一些形式的多媒体会话应用的初始化时所需的信息^[3]。RTP(Real-time Transport Protocol:实时传输协议),它在多点传送(多播)或单点传送(单播)的网络服务上提供端到端的网络传输功能,适合应用程序传输实时数据^[4]。流媒体播放器的网络协议结构如图1所示。

收稿日期:2009-09-01;修回日期:2009-12-27

基金项目:教育部新世纪优秀人才支持计划项目(NCET-04-0843);重庆市信息产业发展基金项目(200611009)

作者简介:樊承泽(1984-),男,四川大竹人,硕士研究生,研究方向为网络终端及嵌入式Linux;陈蜀宇,教授,博导,研究方向为嵌入式系统、多媒体网络技术、网络通信、网络计算。

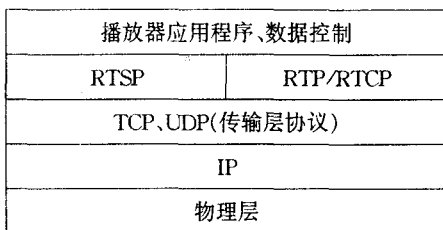


图 1 流媒体播放器的网络协议结构

RTSP 的控制是通过单独的协议发送的,没有绑定到传输层连接(如 TCP 连接),所以可以使用 TCP 发送 RTSP 请求;但此外,也可以使用无连接传输协议,比如用 UDP 发送 RTSP 请求^[5,6]。

2 基于网络计算机的流媒体播放器的设计实现

流媒体播放器已经有很多年的历史,但是早期的播放器只能单向被动的获取服务器上的流媒体资源,而不能和服务器进行直接交互,所以在播放流媒体文件的同时不能进行暂停、快进、快退、慢进以及拖动等操作,这种播放模式/操作模式影响了流媒体播放技术的进一步发展。在 RTSP 协议出现以后,为流媒体播放添加了实现播放控制功能,较好地解决了前面提到的问题。文中在应用程序的设计中,将实现如下功能:

1) RTSP 远程点播功能。

用户首先输入希望播放的流媒体文件 URL,通过播放器与流媒体服务器建立连接,然后从流媒体服务器获取媒体文件的 SDP 描述,同时通过 RTP 信道接收媒体数据进行播放。

2) 播放器远程控制功能。

设计的播放器有播放和暂停、恢复播放、快进、快退、拖动和停止等远程控制功能。

2.1 播放器的整体框架

播放器主要由四个模块组成,它们分别是 RTSP 播放控制模块、音视频解码模块、实现流媒体协议的 RTSP 模块和 RTP/RTCP 模块。播放器的模块框架图如图 2 所示。

RTSP 播放控制模块的实现依赖于 RTSP 模块提供的接口函数,它将用户操作转化为 RTSP 信令与服务器交互,从而实现对远程文件的本地操作。此模块也控制音视频解码器模块何时运行、何时停止,包含的播放控制功能有获取用户输入的 URL 地址、播放、暂停、快进、快退、拖动以及停止。

音视频解码器模块从 RTP/RTCP 模块接收流媒体数据,进行解码,还原为波形文件,再写入音视频设备进行播放。

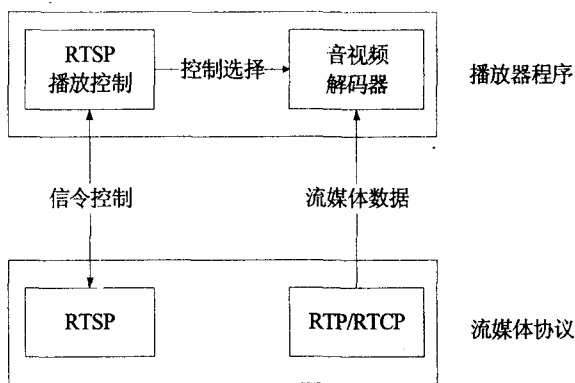


图 2 播放器的模块框架图

RTSP 模块和 RTP/RTCP 模块是实现流媒体传输的核心模块。RTSP 模块负责在播放器与服务器之间建立信令通道,实现 RTSP 消息的交互。RTP/RTCP 模块负责在播放器与服务器之间建立媒体数据通道,让播放器接收从服务器传来的媒体数据,并通过 RTCP 的反馈机制对传输网络的 QoS 进行控制。

2.2 接收流媒体数据的实现

为了获取网络流媒体资源,首先分析播放器与服务器的交互过程,RTSP 中所有的操作都是通过服务器和播放器的消息应答来完成的。播放器对 RTSP 的控制有很多实现方法,以下几种方法在信息的交互以及流资源的分配上起着重要的作用:SETUP、PLAY、PAUSE、TEARDOWN。播放器与服务器的交互过程如图 3 所示。

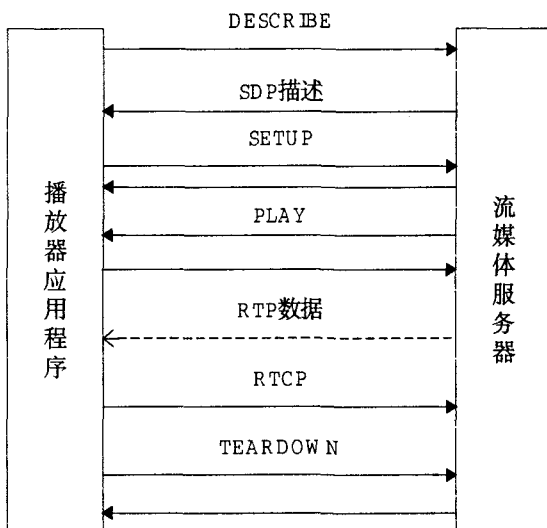


图 3 播放器与服务器的交互过程

1) 当播放器通过 RTSP 的 DESCRIBE 方法向服务器发送请求时,服务器会根据用户请求的资源情况,采用 SDP 的信息组织方式返回资源的描述信息,包括媒体的组成、名称、地址、编码方式、时间长度等。

2) 之后播放器向具体媒体地址发送 SETUP 命令。如果服务器正确建立连接,就返回一个独立的会

话序列号。根据这个序列号,播放器可以同服务器进一步通信。

3)连接成功后,播放器向服务器发送 PLAY 请求,打开 RTP 接收端口。此时,播放器就收到媒体的音视频 RTP 包。

4)播放器可以通过发出 PAUSE 和 TEARDOWN 命令来暂停和结束同流媒体服务器的连接^[7]。

2.3 播放器两个重要线程的设计

因为播放器应用程序需要同时处理两方面的信息:一方面是来自用户的播放控制操作,另一方面是从网络到来的媒体数据流;因此在应用程序中需要开启两个线程来同时处理这两个方面的任务。文中把处理来自用户播放控制操作的线程称作 RTSP 播放控制线程,把接收、处理网络流媒体数据的线程称作流媒体数据处理线程,这两个并发的线程控制着播放器的各个模块良好地工作。

2.3.1 RTSP 播放控制线程

此线程通过图形化界面的消息传递和回调函数机制,处理来自用户的播放控制操作,实现播放控制功能。RTSP 播放控制线程如图 4 所示。

当播放器应用程序启动之后,首先创建主窗口以及依附于主窗口上的各个控件,然后调用函数将完整的图形界面显示出来;接着进入外部事件监听循环中,当来自用户的播放控制操作发生时,就调用该事件对应的回调函数进行处理;处理完成之后,则继续监听下一个事件的到来。播放控制的各个回调函数,将调用 RTSP 协议所定义的播放、暂停、快进/快退、停止等方法,向流媒体服务器发出请求;在收到流媒体服务器的正确响应之后,即完成了控制播放的功能。

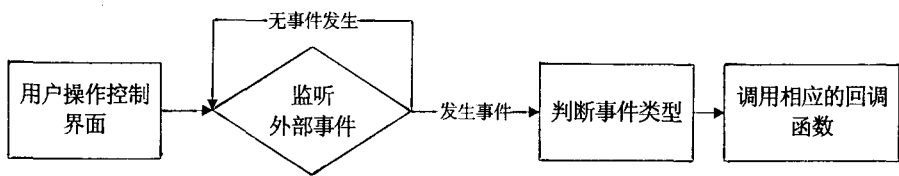


图 4 RTSP 播放控制线程

2.3.2 流媒体数据处理线程

此线程实现的功能是对来自网络的流媒体数据包进行处理,解析出流媒体数据,并交给音视频解码程序进行播放。流媒体数据处理线程如图 5 所示。

用户通过图形用户界面输入 URL,启动一个管道,用来传递流媒体信息;RTSP 播放控制模块向管道中写入数据,而音视频解码模块则从管道中读取数据。管道创建之后,需要为管道的读写两端分别设置两个进程,而在网络计算机环境下,要使两个进程并发运行,需要调用 fork() 函数。在 fork() 调用完成后,两个

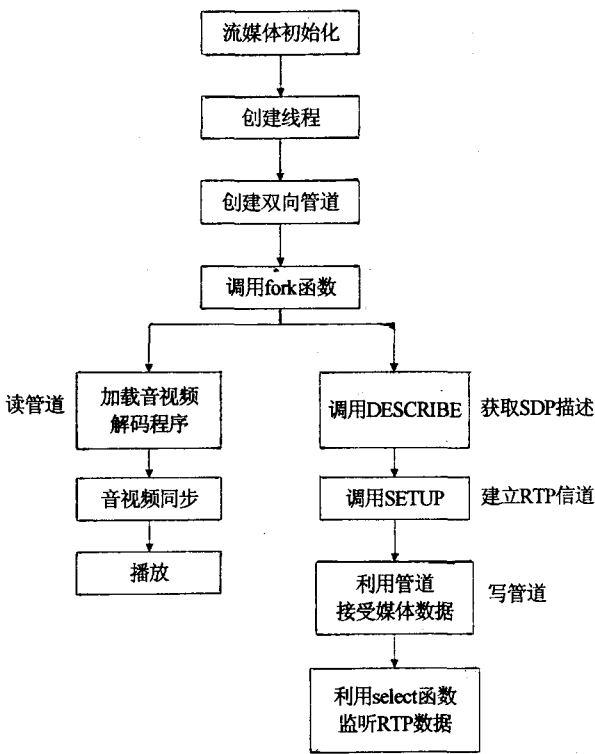


图 5 流媒体数据处理线程

进程事实上共享同一块存储器,因此新进程可以更改原进程的数据及堆栈信息;在 fork() 系统调用完成后,原来的进程进入睡眠,直到新的进程执行 execvp(), 从管道中读取写入的流媒体数据,加载音视频解码程序,完成播放功能。当新的进程正确开始执行后,将唤醒原进程,使得原进程继续往后执行,调用 RTSP 播放控制线程,建立播放器与流媒体服务器的连接,开启 RTP 信道。随后流媒体数据处理线程将利用 select() 系统调用监听 RTP/RTCP 包的到来,对 RTP/RTCP 包进行处理,将媒体数据写入到管道中。通过 fork() + execvp() 的方式,加载音视频解码程序到内存中与流媒体数据处理线程一起运行。

2.4 播放器缓冲控制策略

在网络中传输实时流媒体数据时,网络状况经常变化,因此,视音频数据到达终端的网络延迟是不确定的。为避免播放产生空隙,在播放器端设定合适的缓冲区大小就显得尤为重要。如果缓冲区设置过小,当网络传输速度比较慢的时候,流媒体文件播放的流畅性将会受到很大的影响,出现跳帧等情况。相反如果缓冲区设置过大,而网络传输速度又比较快,那么该播放器不但占用了过多的内存空间而导致了整个网络计算机设备的性能的降低,也造成了对网络传输速度的浪费。

但是已有的缓冲区计算方法比较复杂,不容易在嵌入式设备上实现。为了更好地解决这个问题,设计了一种“缓冲区动态设定”的方法^[8]。该方法在每次确定缓冲区大小之前,先由播放控制线程打开一个新的探测线程,该探测线程执行一段简单的媒体数据抓取代码,连续五次抓取服务器上的媒体数据,并将结果返回给播放器。接下来播放器首先根据返回结果计算每次抓取中的数据传输率(即数据量与时间的比值),接着运用概率理论进行分析。将数据传输率看作一个可重复采样的独立样本,采用了样本均值与样本方差这两个样本特征值来进行分析。所以可以得到如下这两个计算公式:

$$\text{样本均值: } \bar{X} = \sum_{i=1}^5 T_i / 5$$

$$\text{样本方差: } S^2 = \sum (T_i - \bar{X})^2 / 5$$

对应流媒体传输的参数,可以看到,样本均值反映了流媒体传输的平均速率 \bar{V} ,而样本方差则较好地反映了流媒体传输的延时抖 T_{jitter} 。接下来采用下面这个公式估算缓冲区的长度:

$$\text{缓冲区长度计算: } L_{\text{buffer}} = \beta \bar{V} T_{\text{jitter}}$$

其中 β 为一个辅助系数,可通过经验尝试获得。当播放控制线程获取了恰当的缓冲区长度设置之后,该线程将缓冲区长度作为一个播放参数传递给流媒体处理线程,流媒体处理线程根据指定的长度分配合适的缓冲区数据块长度用于网络流媒体数据的处理。

3 软件的测试播放

根据文中对流媒体播放器的整体设计和关键模块的设计,以网络计算机为硬件载体,配以一台网络流媒体服务器完成播放流媒体文件的测试。

1) 播放测试结果。

测试显示播放效果良好,播放视音频流畅并且同步,顺利完成播放和暂停、恢复播放、快进、快退、拖动和停止等功能性操作,达到预期目标。

2) 网络负载测试。

为了验证播放器缓冲区算法对网络带宽的占用情况影响,以典型网络计算机局域网应用为背景进行测试:其中流媒体服务器提供网络多媒体资源,不同数量的网络计算机通过播放器以 RTSP 协议的形式请求网络媒体服务,播放过程中网络带宽资源的消耗情况如图 6 所示。

测试表明,30 台网络计算机同时播放服务器上的流媒体资源时,对带宽的要求不到 4MB/s,远没有达到局域网的带宽上限,保证了良好的网络环境。

4 结束语

文中在阐明流媒体播放器开发原理的基础上,描述了一个针对网络计算机环境下的流媒体播放器的技术实现方法,实现了对网络流媒体文件播放的实时控制功能。该播放器不但使网络计算机平台的多媒体功能更加丰富灵活,而且在一定程度上提高了资源受限的网络终端设备对大容量流媒体资源的处理效果。目前该播放器软件已经成功应用于重庆网舟网络技术公司生产的 e-boat 网络计算机中,并取得较好的应用效果。

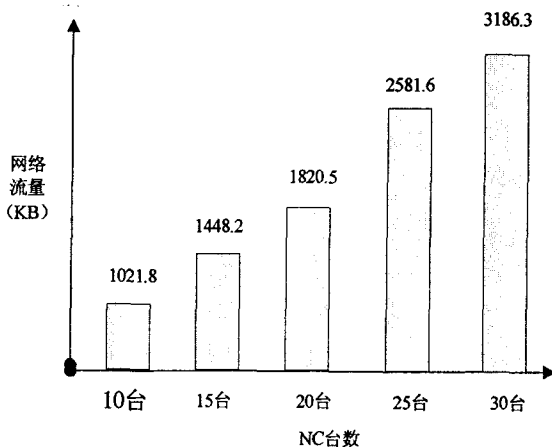


图 6 网络负载测试结果图

参考文献:

- [1] 陈蜀宇,曹俊勇,于 渤. 网络计算机中嵌入式 Linux 体系结构的研究[J]. 计算机工程,2007,33(6):74-76.
- [2] Network Working Group. Real Time Streaming Protocol(RTSP)[S]. RFC2326,1998.
- [3] Handley M. Session Description Protocol(SDP)[S]. RFC 2327,1998.
- [4] Network Working Group. A Transport Protocol for Real-Time Applications(RTP)[S]. RFC 3550,2003.
- [5] Stevens W R, Stephen R. Advanced Programming in the UNIX Environment. [M]. 2nd ed. 北京:人民邮电出版社,2006.
- [6] Cavusoglu B, Schonfeld D, Ansari R. Real-time Low-complexity adaptive approach for enhanced QoS and error resilience in MPEG-2 video transport over RTP network[J]. IEEE Trans. Inf. Theory,2005,15(11):1604-1614.
- [7] 金 海,邵艳明,韩宗芬. 基于实时流协议的流媒体播放器[J]. 计算机工程,2004,30(11):192-194.
- [8] 文远保,林建明. 嵌入式流媒体播放器缓冲控制策略研究[J]. 华中科技大学学报:自然科学版,2005,33(10):84-86.