

基于 RTSP 协议的流媒体迁移系统实现

宋欣,宋飞,周华春

(北京交通大学电子信息工程学院,北京 100044)

摘要:个人终端设备(包括智能手机、平板电脑、笔记本等)的多样化,使用户对服务移动性的要求越来越高。目前的研究多数集中在服务器层面的被动迁移,而面向客户端的主动迁移正在成为研究热点。作为互联网的典型应用,流媒体服务在发展过程中急需解决如何实现主动迁移的问题。文中在分析了 RTSP 协议和其他迁移系统的基础上,分别构建了流媒体服务模块和连接管理模块,并以服务状态信息作为迁移内容,实现了用户在不同终端上的流媒体迁移。

关键词:实时流协议;流媒体;迁移;C/S

中图分类号:TP39

文献标识码:A

文章编号:1673-629X(2012)12-0059-04

Implementation of Streaming Media Migration System Based on RTSP

SONG Xin, SONG Fei, ZHOU Hua-chun

(School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China)

Abstract: With the variety of personal terminal devices, including smartphone, pad, notebook etc, service mobility is increasingly needed for users. Nowadays the migration is focused on service for servers, and the migration for applications of personal terminal is becoming hot topic. Streaming media service is the typical application of internet, and it has a great desire of migration. It is based on the analysis of real time streaming protocol and some parts of other migration systems, making the system consists of two modules are streaming media service model and joint management model, and using the status parameters of streaming media as migration contents to achieve the streaming media migration for one user in different terminals.

Key words: RTSP; streaming media; migration; C/S

0 引言

随着信息技术的高速发展,个人终端不再局限于传统的台式计算机和笔记本。特别是近年来由 iPhone、iPad、Android 等移动终端设备引领的开发浪潮^[1],使智能手机和平板电脑的功能大大增强,许多原本只能在电脑上获取的服务,如今可以凭借这些便携式设备轻而易举的实现。通过便携式终端上网看视频、听音乐、浏览网页等已经成为了人们的生活习惯。当然与固定式终端相比,便携式终端也存在如屏幕小、音质小、电池容量不足等弊端。各类终端的优缺点使用户在不同情形下选择不同设备,比如用户处于移动状态,往往会选择便携式终端;用户处于固定场所,则习惯使用固定式终端。但当用户终端发生改变时,连

续的服务却不能随之改变,此情形导致用户希望在不同终端上持续获取同一服务,迁移需求由此而生。

流媒体服务是当今互联网的主流应用^[2],其实现方式大多采用实时流协议 RTSP (Real Time Streaming Protocol)^[3]。因为流媒体服务具有持续性的特点,当用户中断流媒体服务后,再次获取服务需要重新查找资源、资源定位等,所以流媒体服务的迁移需求较为强烈。文中将在 RTSP 协议的基础上,利用流媒体资源的统一资源定位符 URL (Uniform Resource Locator) 和正常播放时间 NPT (Normal Play Time) 作为服务状态信息,传输给迁移对象,从而实现用户在不同终端的流媒体迁移。

1 研究背景

1.1 迁移研究分析

迁移系统的研究,按层次划分主要包括服务层迁移和用户层迁移^[4]。服务层迁移系统主要针对服务不可达、服务器发生变化等情况,其研究重点在于预判迁移发生、选择迁移内容、决策迁移目标^[5];用户层迁移则不同,因为迁移发生时间和迁移目标都是由用户直

收稿日期:2012-04-05;修回日期:2012-07-08

基金项目:中央高校基本科研业务费专项资金(2011JBM018);国家自然科学基金资助项目(60833002)

作者简介:宋欣(1988-),男,硕士研究生,研究方向为互联网应用和移动互联网;宋飞,博士,讲师,硕士生导师,研究方向为下一代互联网体系架构、网络协议分析和优化;周华春,博士,教授,博士生导师,研究方向为计算机网络安全和移动互联网。

接决定的,所以用户层迁移主要集中在迁移内容的选择上。文中将重点研究用户层迁移。

用户层迁移的实现,主要工作是客户端设计特定规范或协议。凭借近年来“云”概念的提出和移动互联网的发展,用户层迁移正得到越来越多的关注。网络上用户层迁移的应用有 Google 推出的 Chrome 云端数据同步服务^[6]。该服务以浏览器为平台,所有通过浏览器访问产生的服务状态信息,都保存在云端服务器。当用户放弃当前终端的客户端,转而使用新终端客户端访问服务时,云端服务器将用户信息推送给新的客户端,以此实现用户的信息同步。该服务属于 web 迁移,其特点是对服务实时性要求不高,并且实现较为简单。而流媒体迁移更注重实时性,而且悬挂和恢复流媒体服务较为复杂。

用户层迁移的内容,包括数据流、控制流和服务状态。从实现的代价考虑,服务状态的迁移是最优选择^[7]。流媒体的具体内容大致可以分为三类^[8],根据此分类内容,服务状态信息有三种描述方式。对于流式存储音频/视频,服务状态至少应包括 URL 和 NPT 参数;对于流式实况音频/视频,服务状态则只要求 URL;对于交互式音频/视频,服务状态还要结合应用层的具体需求,比如 SIP(Session Initiation Protocol)协议的 URL^[9]等等。在迁移系统中,这三类流媒体的实现方式完全一致,差别只在于服务状态信息的内容不同。文中将以流式存储音频/视频为例,采用中间件的方式,利用服务器和客户端程序来管理和转发服务状态信息。

1.2 RTSP 协议

RTSP 主要作用是传输流媒体控制信息,流媒体数据信息的传输则依赖于某种传输层,如 RTP(Real Time Transport Protocol)协议。RTSP 基于 C/S(Client/Server)模型,客户端与服务器的交互主要通过请求和应答方式进行,请求和应答消息都使用 RFC822 中实体传输的规定格式。RTSP 请求与响应的语法格式如图 1、图 2 所示。其中空白部分表示空格,CRLF 表示回车换行。

方法	URL	版本	CRLF	请求行
首部字段名	:	值	CRLF	
⋮				
首部字段名	:	值	CRLF	
CRLF				首部行 (可选)
...				
主体(可选)				

图 1 RTSP 请求消息格式

RTSP 协议的请求行中,规定了多种方法用于实现

流媒体服务建立、播放、暂停、结束等功能。客户端通过这些请求方法向 RTSP 服务器申请服务,服务器的响应则返回状态码和短语^[10],这里的状态码解释参照了 HTTP 的响应方式。状态码的第一位数字决定了响应的类别,后两位可以自定义。一般的响应以“2”起始表明了操作被成功理解和接受,比如 200 OK;“4”表明客户端出错;“5”表明服务器出错。在 RTSP 协议消息格式中有很多参数,其中和流媒体资源紧密相关的参数是 URL 和 NPT。URL 是用来表示从因特网上得到资源位置和访问这些资源的方法。NPT 则指明了流相对于开始的位置,精度可以到毫秒级别。当流媒体资源为实况直播时,NPT 默认无效。文中流媒体迁移的实现方式是迁移源向迁移目标传输 RTSP 协议的 URL 和 NPT 参数,迁移目标则利用此参数信息重新与服务器建立连接。

版本	状态码	短语	CRLF	状态行
首部字段名	:	值	CRLF	
⋮				
首部字段名	:	值	CRLF	
CRLF				首部行 (可选)
...				
主体(可选)				

图 2 RTSP 响应消息格式

2 系统实现

2.1 系统结构模块

流媒体服务按结构分为 C/S 模式和 P2P 模式。在流媒体迁移中采用 C/S 模式比 P2P 模式更合适,具体原因包括三点。首先是 C/S 模式下的流媒体迁移更快捷有效。流媒体服务中 P2P 模式的优势是大规模下的数据流传输^[11],用户可以提前分块下载流媒体资源。可是当迁移发生后,原客户端所下载的资源都将浪费,用户在新的客户端需要重新下载资源。所以流媒体迁移中,用户更需要的是稳定、可靠、有实时保证的服务器。其次,流媒体迁移注重控制流的可靠传输,而 RTSP 协议的控制交互流程正是基于 C/S 模式设计的。最后,迁移系统在 P2P 模式下难以保证用户的安全和隐私。迁移系统中,服务器保存了用户的所有终端信息,特别是每个终端的 IP 地址,而 P2P 模式会将这些信息泄露给多个资源提供者。在 C/S 模式基础上,本系统主要分两大功能模块进行实现:流媒体服务模块和连接管理模块。每个模块在服务器和客户端都有相应的功能子模块,如图 3 所示。

流媒体服务模块的功能是提供并获取流媒体服

务。在流媒体服务器端,准备好的资源文件要添加到资源索引当中,供客户端实现资源查询。资源的描述信息包括资源 URL、资源格式、资源大小、时间长度等。在查询到用户需要的资源信息后,流媒体播放器向流媒体服务器端提交 RTSP 请求,在收到服务器的正确响应后开始播放流媒体资源。

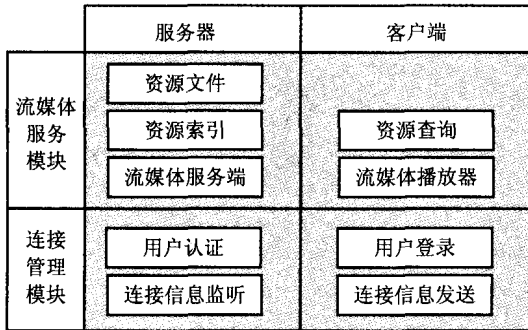


图 3 系统组成模块

连接管理模块的功能是监听所有客户端的连接信息,并将这些信息推送给迁移源,让用户决定迁移目的地。为了确保客户端的用户是属于同一用户,连接管理模块还要有用户认证,因为属于同一用户的其他客户端才是迁移的目标。当用户发起迁移请求时,服务器首先记录已登录某一账户的所有客户端,然后将记录结果发送至当前客户端。用户选择某一客户端后,迁移源客户端将服务状态信息发送至迁移目标的客户端。在此过程中,当用户只打开两个客户端时,系统可以优化迁移流程,即不需要用户选择迁移目标,而直接将服务状态信息发送至另一客户端。在迁移使用中,此状况也是最典型的案例。

2.2 迁移流程描述

系统流程可以用一台服务器和两台客户端之间的交互来阐述,其中客户端 A 为迁移源,客户端 B 为迁移目标,正确的流媒体迁移流程如图 4 所示。步骤描述如下:

1) 客户端 A 向服务器端发送登录请求。如账号信息正确,服务器端将返回确认信息和推荐的资源列表;如账号信息错误,服务器端返回登录错误,要求用户重新登录。

2) 客户端 A 在资源列表选取某一流媒体资源后,向服务器发送播放请求。此交互过程主要通过 RTSP 协议的方法实现,涉及到四个方法:OPTION 方法询问

服务器有哪些方法可用;DESCRIBE 方法用于描述流媒体资源;SETUP 方法是让服务器给流分配资源,启动 RTSP 会话;PLAY 方法将启动流媒体的数据传输。方法的响应信息用状态码返回,“200 OK”表示操作正确接收,其他状态码的含义见 2.2 小节。

3) 客户端 B 向服务器发送登录请求。此处登录的账户信息应当与步骤 1 中的账户信息相同,以此来确保两台客户端的使用者都是同一用户。服务器的响应信息与步骤 1 相同。

4) 客户端 A 向服务器发送迁移请求。服务器将该账户的其他客户端 IP 地址发送到客户端 A。迁移系统中可能存在两个以上的客户端迁移(图 4 以两个客户端为例),这些客户端都有可能是迁移目标,所以其他客户端的 IP 地址都要发送至迁移源。

5) 客户端 A 选择迁移目标(图中选择客户端 B),发送服务状态信息。当客户端只有两台时,程序从收到的 IP 列表中自动选择另一台作为迁移目标。流媒体迁移系统中的服务状态信息主要是 URL 和 NPT。

6) 客户端 B 根据收到的服务状态信息,重新向服务器发送播放请求。此次的播放请求将恢复客户端 A 中断的流媒体服务,实现方式与步骤 2 大致相同,不同之处仅在于 PLAY 方法中的 NPT 参数采用步骤 5 中收到的参数值。

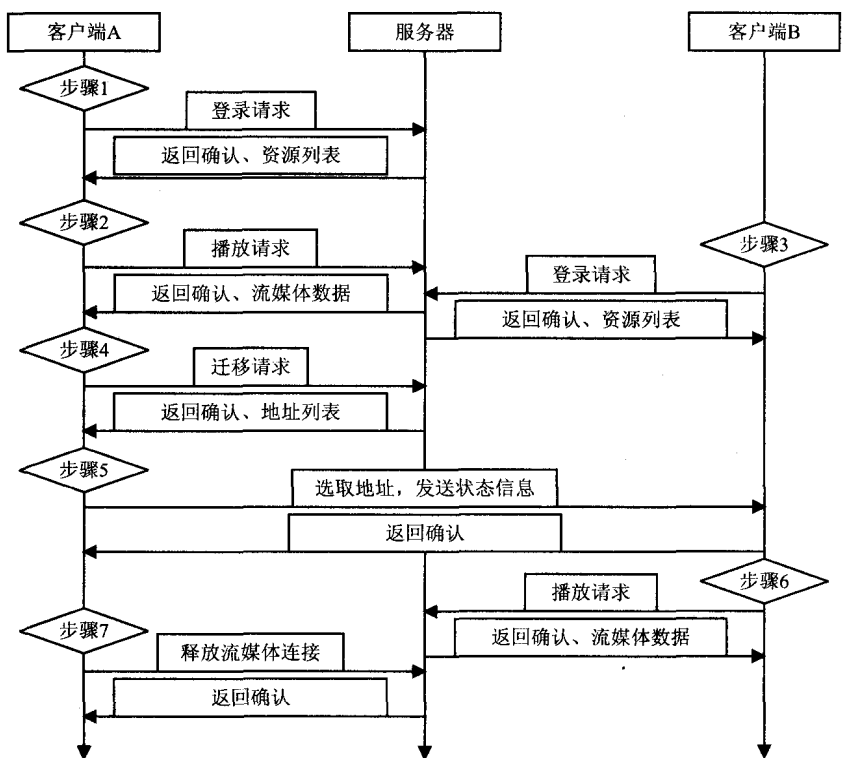


图 4 流媒体迁移流程

7) 客户端 A 收到迁移完成的确认信息后,释放流媒体连接。流媒体连接的释放,通过 RTSP 协议中的 TEARDOWN 方法实现,此方法将释放流占用的资源,

RTSP 会话停止。

2.3 实现方法

按照 2.1 小节,流媒体服务模块分为服务器和客户端两部分实现,服务器采用了开源程序 Live555 Media Service,此程序支持 RTSP 协议下的流传输并且可以实现视频跳转;在笔记本电脑客户端通过调用 vlcj 项目^[12]的 API 实现了将 vlc 流媒体播放器内嵌到客户端程序。在 Android 手机客户端通过调用 vlc 的函数库来实现流媒体播放功能。在流媒体的格式选取过程中,建议使用 MPEG-2 标准下的 TS(Transport Stream)格式^[13]。原因是 TS 流格式在流传输中性能可靠,而且 TS 视频文件可以产生一个后缀名为 TSX 的文件,此文件用于索引 TS 视频,可以很好地实现视频快进、后退、跳转等功能。当然,如果流媒体资源是直播而不是点

播时,视频不存在快进、后退、跳转功能,代表播放时间属性的 NPT 值也将默认不会对迁移产生作用,此时可以选取任意支持流媒体播放的视频格式。

连接管理模块同样分为服务器和客户端实现,编程语言采用 Java。用户认证实现较为简单,服务器端采用了 Mysql 数据库保存账户信息,并设定端口监听客户端的登录请求。为了实现连接管理,服务器端先监听一个固定端口,收到客户端的请求信息后,根据接收数据中的用户信息,推送给迁移源的客户端。迁移源客户端与服务器建立通信 socket,用于登录服务器和接收迁移目标地址信息;迁移源客户端与迁移目标客户端建立通信 socket,用于实现服务状态信息的发送和接收。

3 测试

本系统的测试环境为一台 Android 智能手机、一台笔记本和一台服务器,两台计算机操作系统为 Windows XP。服务器要求安装 Live555 媒体服务器软件, Wireshark 抓包软件和流媒体迁移服务程序;笔记本要求添加 vlcj 和 libvlc 等库,还有流媒体迁移客户端;智能手机要求导入 vlc 函数库和流媒体迁移客户端。测试视频为 TS 视频文件格式,RTSP 版本信息为 1.0。服务器的 IP 地址为 192.168.161.6,智能手机和笔记

本的 IP 地址分别为 192.168.161.7,192.168.161.8,如图 5 所示。本测试通过在服务器抓包来分析迁移过程中,RTSP 协议的交互流程是否正确,以及 URL 和 NPT 是否按照系统要求传送给迁移目的地,抓包结果见图 6。

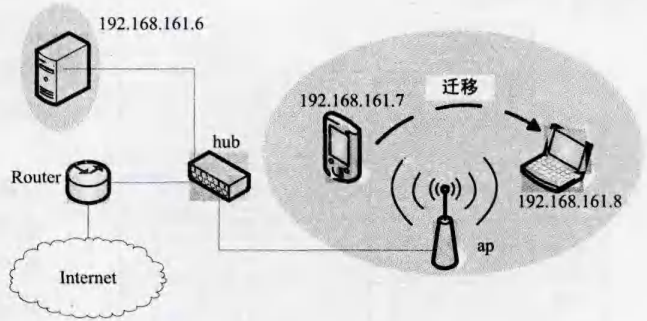


图 5 流媒体迁移测试环境

Source	Destination	Protocol	Info
192.168.161.7	192.168.161.6	RTSP	OPTIONS rtsp://192.168.161.6/all.ts RTSP/1.0
192.168.161.7	192.168.161.6	RTSP	DESCRIBE rtsp://192.168.161.6/all.ts RTSP/1.0
192.168.161.7	192.168.161.6	RTSP	SETUP rtsp://192.168.161.6/all.ts/track1 RTSP/1.0
192.168.161.7	192.168.161.6	RTSP	PLAY rtsp://192.168.161.6/all.ts RTSP/1.0
192.168.161.7	192.168.161.6	RTSP	GET_PARAMETER rtsp://192.168.161.6/all.ts/ RTSP/1.0
192.168.161.7	192.168.161.6	RTSP	TEARDOWN rtsp://192.168.161.6/all.ts/ RTSP/1.0
192.168.161.8	192.168.161.6	RTSP	OPTIONS rtsp://192.168.161.6/all.ts RTSP/1.0
192.168.161.8	192.168.161.6	RTSP	DESCRIBE rtsp://192.168.161.6/all.ts RTSP/1.0
192.168.161.8	192.168.161.6	RTSP	SETUP rtsp://192.168.161.6/all.ts/track1 RTSP/1.0
192.168.161.8	192.168.161.6	RTSP	PLAY rtsp://192.168.161.6/all.ts RTSP/1.0

Real Time Streaming Protocol		Real Time Streaming Protocol	
Request: PLAY rtsp://192.168.161.6/all.ts/	CSeq: 5\r\n	Request: PLAY rtsp://192.168.161.6/all.ts/	CSeq: 8\r\n
User-Agent: LibVLC/2.0.0 (LIVE555 Streaming)	Session: C2844D84	User-Agent: LibVLC/2.0.0 (LIVE555 Streaming)	Session: E226AED5
Range: npt=0.000-\r\n		Range: npt=28.949-\r\n	

图 6 测试抓取数据包

为了使结果易于分析,在抓取数据包中过滤了服务器向客户端返回的正确状态码信息,过滤规则为“rtsp and ip.src != 192.168.161.6”。从图 6 可以看到:迁移前播放的流媒体信息,IP 地址为 192.168.161.7,PLAY 方法中的 URL 为 rtsp://192.168.161.6/all.ts,NPT 为 0.000-;而迁移后播放的流媒体信息,IP 地址为 192.168.161.8,PLAY 方法中的 URL 为 rtsp://192.168.161.6/all.ts,NPT 为 28.949-。上述对比中,IP 地址有变化说明了迁移发生;而 URL 保持一致说明了接受的服务相同;NPT 从 0.000 变化到 28.949 说明了迁移到新客户端后,流媒体播放将从原来的断点处继续。测试证明,在 RTSP 协议的支持下,流媒体迁移可以较好地实现。

4 结束语

文中分析了用户的迁移需求和迁移的研究状况,设计了流媒体迁移系统的模块和交互流程,实现了面向用户的流媒体迁移系统,并在实际环境中对流媒体迁移系统进行了测试。迁移系统的实现还可以拓展到

(下转第 66 页)

但在处理时有时会过多地合并原图像的灰度级而降低了图像整体的灰度层次感,丢失了一些细节^[12]。针对这样的问题,根据原图像灰度级的灰度值总体偏高还是总体偏低两种情况,有两种保留灰度级的 HE 算法:保留高灰度级的 HE 和保留低灰度级的 HE。运用 MMTD 来判断原图像灰度级的灰度值是总体偏高还是总体偏低进而决定用哪种保留灰度级的 HE 算法。

采用保留灰度级 HE 处理后的图像在一定程度上改进了传统的 HE 算法的缺点,即由于灰度级合并而降低整体的灰度层次感。经过该方法处理过的图像能包含更多的细节信息,图像看起来显得更加连续,视觉效果比较柔和。由此认为,中介真值程度的数值化度量方法对直方图均衡化图像增强还是有一定帮助的,下一步将继续运用 MMTD 的数学方法来尝试改进一下直方图规定化(匹配)的传统方法,争取能获得比较好的改进效果。

参考文献:

[1] 赵晓捷,穆冬,薛斌党.反距离加权插值自适应图像直方图均衡化算法[J].中国体视学与图像分析,2010(15):8-11.

[2] Menotti D, Najman L, Facon J, et al. Multi-histogram equalization methods for contrast enhancement and brightness preserving[J]. IEEE Transaction on Consum Electronics, 2007,

53(3):1186-1194.

[3] Kim J Y, Kim L S, Hwang S H. An advanced contrast enhancement using partially overlapped sub-block histogram equalization[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2001, 11(4):475-477.

[4] Chang D C, Wu W R. Image contrast enhancement based on a histogram transformation of local standard deviation[J]. IEEE Trans on Medical Imaging, 1998, 17(4):518-531.

[5] Gonzalez R C, Woods R E. 数字图像处理[M].阮秋琦,阮宇智译.第2版.北京:电子工业出版社,2008.

[6] 乔闹生.一种改进的直方图均衡化[J].光学技术,2008,34(S1):141-142.

[7] 张志龙,李吉成,沈振康.一种保持图像细节的直方图均衡新算法[J].计算机工程与科学,2006,28(5):36-38.

[8] Chen S D, Ramli A R. Contrast enhancement using recursive mean-separate histogram equalization for scalable brightness preservation[J]. IEEE Transactions on Consumer Electronics, 2003, 49(4):1301-1309.

[9] 洪龙,肖奚安,朱梧楦.中介真值程度的度量及其应用(I)[J].计算机学报,2006,29(12):2186-2194.

[10] 洪龙,肖奚安,朱梧楦.中介真值程度的度量及其应用(II)[J].计算机学报,2007,30(9):1551-1558.

[11] 姚静,武文波,康停军.直方图均衡化的 Matlab 实现[J].辽宁工程技术大学学报,2007,26(S2):60-62.

[12] 尹立敏,刘艳滢,贺琳,等.基于多尺度灰度直方图的图像增强的研究[J].电子器件,2006(2):431-433.

(上接第 62 页)

其他服务,搭建一个包容各类服务的迁移平台,将是下一步的研究方向。

参考文献:

[1] 百度无线移动开放平台.百度移动互联网发展趋势报告 2011Q4[EB/OL]. 2012-03-01. <http://open.shouji.baidu.com/?page=mireport&action=download&f=2011q4>.

[2] Sandvine Incorporated. Global Internet Phenomena Report Fall 2011[EB/OL]. 2011-10-26. [http://www.sandvine.com/downloads/documents/10-26-2011_phenomena/Sandvine Global Internet Phenomena Report - Fall 2011. pdf](http://www.sandvine.com/downloads/documents/10-26-2011_phenomena/Sandvine%20Global%20Internet%20Phenomena%20Report%20-%20Fall%202011.pdf).

[3] Schulzrinne H, Rao A, Lanphier R. Real time streaming protocol[S]. IETF RFC2326, 1998.

[4] Liu Haiqin, Sun Yan, Kim M S. Provider-level content migration strategies in P2P-based media distribution networks[C]//IEEE CCNC 2011. [s.l.]:[s.n.], 2011:337-341.

[5] 苏俭,郭伟,刘强.移动 Ad Hoc 网络服务迁移的研究与实现[J].计算机工程,2009,35(9):110-113.

[6] Paul R. Google reveals plans for Chrome cloud synchronization

[EB/OL]. 2009-08. <http://arstechnica.com/open-source/news/2009/08/google-reveals-plans-for-chrome-cloud-synchronization.ars>.

[7] 周俊,李文中,陆桑璐,等.利用网格技术实现流媒体服务迁移[J].计算机科学,2005,32(8):109-113.

[8] 谢希仁.计算机网络[M].北京:电子工业出版社,2007.

[9] Rosenberg J, Schulzrinne H, Camarillo G, et al. Session Initiation Protocol[S]. IETF RFC3261, 2002.

[10] Fielding R, Gettys J, Mogul J, et al. Hypertext Transfer Protocol[S]. IETF RFC2616, 1999.

[11] Ma Liqun, Shen Haihong, Zhang Qishan. The key technologies for a large-scale real-time interactive video distribution system[C]//IEEE ICACC 2010. [s.l.]:[s.n.], 2010:471-476.

[12] Lee M. Java Bindings for vlc[EB/OL]. 2012-02-13. <http://code.google.com/p/vlcj/>.

[13] Huttunen A, Defee I. Performance of desktop software MPEG-2 TS decoder[C]//ISCAS 99. [s.l.]:[s.n.], 1999.