



(12)发明专利申请

(10)申请公布号 CN 108494792 A

(43)申请公布日 2018.09.04

(21)申请号 201810312365.6

(22)申请日 2018.04.09

(71)申请人 四川百视通科技有限公司
地址 610000 四川省成都市高新区天府五街200号1号楼8楼A、B区

(72)发明人 李超

(74)专利代理机构 成都睿道专利代理事务所
(普通合伙) 51217

代理人 贺理兴

(51) Int. Cl.

H04L 29/06(2006.01)

H04L 29/08(2006.01)

H04N 21/4402(2011.01)

H04N 21/845(2011.01)

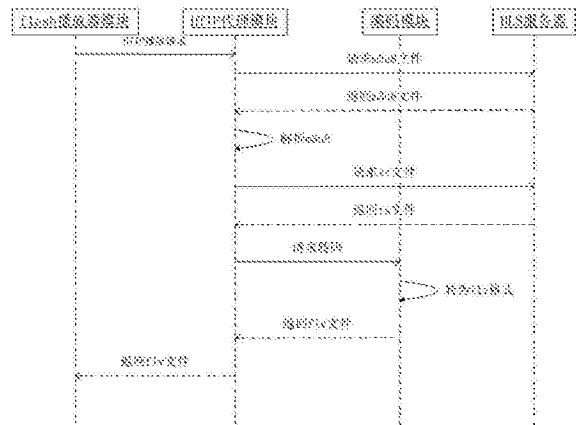
权利要求书2页 说明书7页 附图3页

(54)发明名称

一种flash播放器播放hls视频流的转换系统及其工作方法

(57)摘要

本发明公开了一种flash播放器播放hls视频流的转换系统及其工作方法,flash播放器模块进行视频地址分析,根据得到的视频播放地址向http代理模块发送http播放请求,然后向hls服务器请求m3u8文件,hls服务器返回相应的m3u8文件;http代理模块解析m3u8文件并向hls服务器请求相应的切片文件,hls服务器返回相应的切片文件返回;http代理模块向编码模块发送待转码的切片文件,编码模块将待转码的切片文件转换为flv格式,然后通过http代理模块返回至flash播放器解码并播放。本发明只需要一种满足hls协议的ts格式的文件就能满足flash播放器和移动终端的需求,减少了直播服务器系统的复杂性和降低了系统的不稳定性因素,减少了视频格式转换的时间和步骤,同时节省了磁盘空间。



1. 一种flash播放器播放hls视频流的转换系统,其特征在于,它包括:

Internet网络;

分别与Internet网络连接的hls服务器和直播前端(1);所述hls服务器通过CDN网络连接用户端(6);

所述的直播前端(1)包括与Internet网络连接的直播编码器(4)和与直播编码器(4)连接的若干台摄像机(5);

所述的播放端(6)包括flash播放器、IOS播放器、Android播放器和H5播放器;

所述的hls服务器包括hls直播服务器(2)和hls点播服务器(3)。

2. 根据权利要求1所述的一种flash播放器播放hls视频流的转换系统,其特征在于所述的flash播放器包括:

与所述hls服务器连接的编码器模块和http代理模块,所述编码器模块用于将切片文件按照flv封装格式进行重新封装;

与所述http代理模块连接的flash播放器模块;

所述编码器模块与http代理模块连接;

所述的flash播放器模块还包括协议解析模块。

3. 根据权利要求2所述的一种flash播放器播放hls视频流的转换系统,其特征在于所述的http代理模块包括:

http代理模块,用于接收所述flash播放器的http请求,并响应请求;

http客户端模块,用于向所述hls服务器请求m3u8文件和切片文件;

m3u8文件解析模块,用于解析所述hls服务器返回的m3u8文件,获取切片文件表;

所述的http代理模块还包括动态决策模块,所述动态决策模块用于判断所述http客户端模块采用并行或串行方式向hls服务器请求切片文件。

4. 一种flash播放器播放hls视频流的转换方法,应用权利要求1所述的一种flash播放器播放hls视频流的转换系统,其特征在于,它包括以下步骤:

步骤一:flash播放器模块进行视频地址分析,根据得到的视频播放地址向http代理模块发送http播放请求;

步骤二:http代理模块根据http播放请求向hls服务器请求m3u8文件,hls服务器返回相应的m3u8文件;

步骤三:http代理模块解析hls服务器返回的m3u8文件,得到切片文件列表;

步骤四:http代理模块根据切片文件列表向hls服务器请求相应的切片文件,hls服务器将相应的切片文件返回至http代理模块;

步骤五:http代理模块向编码模块发送待转码的切片文件,编码模块将待转码的切片文件转换为flv格式,然后通过http代理模块返回至flash播放器;

步骤六:flash播放器解码flv格式切片文件并播放。

5. 根据权利要求4所述的一种flash播放器播放hls视频流的转换方法,其特征在于所述步骤三中得到切片文件列表之后还包括判断当前http播放请求为直播请求或点播请求,若当前请求为点播请求,启用动态决策算法。

6. 根据权利要求5所述的一种实现flash播放器播放hls协议的视频方法,其特征在于所述启用动态决策算法具体包括以下步骤:

步骤101:预设带宽初始期望值,初始状态时,采用串行方式向hls服务器请求第一个切片文件;

步骤102:每隔一定时间 n 统计数据接收率 R ,数据接收率 R 的计算公式为: $R=S/n$,其中, S 代表一个时间间隔 n 内接收数据的大小;

步骤103:根据数据接收率 R 判断采用串行方式或并行方式向hls服务器请求切片文件,当数据接收率 R 大于或等于带宽初始期望值,切换到并行方式,当数据接收率 R 小于带宽初始期望值,切换到串行方式;

所述串行方式为依次向hls服务器请求切片文件列表中的切片文件;

所述并行方式为每次向hls服务器请求多个切片文件列表中的切片文件。

7.根据权利要求6所述的一种flash播放器播放hls视频流的转换方法,其特征在于采用并行方式请求切片文件具体包括以下步骤:

步骤201:预设第二带宽初始期望值;

步骤202:定义每次请求切片文件的数量为 m , $2 \leq m \leq 3$;

步骤203:当数据接收率 R 满足大于或等于带宽初始期望值,小于或等于第二带宽初始期望值时, $m=2$;

当数据接收率 R 满足大于第二带宽初始期望值时, $m=3$ 。

8.根据权利要求7所述的一种flash播放器播放hls视频流的转换方法,其特征在于所述的带宽初始期望值为 8Mb/s ,所述的第二带宽初始期望值为 15Mb/s 。

9.根据权利要求4所述的一种flash播放器播放hls视频流的转换方法,其特征在于步骤四所述的将切片文件转换为flv格式具体包括以下步骤:

步骤301:接收一个完整的切片文件,获取切片文件的音频ID和视频ID;

步骤302:解析音频ID和视频ID,得到音频数据和视频数据;

步骤303:查找音频数据和视频数据的解码时间戳DTS和显示时间戳PTS;

步骤304:根据解码时间戳DTS和显示时间戳PTS计算flv格式切片文件的参考时间戳
 $\text{Composition Time} = (\text{PTS} - \text{DTS}) / 90$,

其中,90为切片参考时钟 90KHz ;

步骤305:预设flv解码时间戳初始值为 40ms ;

步骤306:将音频数据和视频数据根据flv解码时间戳和参考时间戳按flv的格式重新封装,得到flv格式的音视频数据;

步骤307:将flv格式的音视频数据通过http代理模块返回至flash播放器模块。

10.根据权利要求5所述的一种flash播放器播放hls视频流的转换方法,其特征在于判断http播放请求为直播请求或点播请求具体为:当m3u8文件以“#EXT-X-ENDLIST”结束时或m3u8文件中包“#EXT-X-PLAYLIST-TYPE:VOD”,则当前请求为点播请求。

一种flash播放器播放hls视频流的转换系统及其工作方法

技术领域

[0001] 本发明属于互联网音视频技术领域,具体涉及一种flash播放器播放hls视频流的转换系统及其工作方法。

背景技术

[0002] 目前在PC端播放视频还是采用页面嵌入flash播放器的方式,flash播放器目前支持的协议有http、rtmp、rtmpc等,但不支持hls协议,支持播放的视频格式有flv、mp4、f4v、3gp。随着移动互联网视频播放技术的快速发展,尤其是由苹果公司提出的视频播放协议hls协议已经统领了移动端终端的视频播放协议。在直播领域,为了满足不同的终端,直播服务器通常会开通两路协议,一路支持PC端的flash播放器协议,另一路支持移动终端的hls协议;在视频点播领域,为了满足不同的终端,通常会支持两种点播协议,一种是支持PC端flash播放器的协议,另一种是支持移动终端的hls协议。由此可见,hls协议既支持直播,还支持点播。

[0003] hls协议播放视频的优点如下:

[0004] 1、性能高。采用HTTP协议传输数据。

[0005] 2、穿防火墙能力强。hls协议的本质是采用了HTTP协议进行数据传输,

[0006] 目前互联网中的防火墙都开放HTTP协议的端口80。

[0007] 3、原生支持很好:IOS、Android平台支持完美。

[0008] 4、采用hls协议播放视频的优势在于:将大文件切片,任意的切片视频不依赖于其它切片视频。由于hls协议采用切片,因此读取一个切片比读取一个大文件花费的时间要小。因此,hls协议在视频点播中支持的快进、快退的体验非常好,达到所见即所得的效果。

[0009] 5、采用hls协议的直播优势在于:由于hls协议采用了切片机制,一个切片不依赖于另一个切片,因此当某个切片在网络传输过程中发生数据错误时,播放影响也仅在出现数据错误的某个切片,不会影响其它切片和全局视频播放。当某个切片不存时,会自动播放下一个切片,因此,hls协议的容错能力比其它协议播放视频的容错能力强。

[0010] 当前PC端flash播放器播放视频还存在如下问题:

[0011] 1、直播问题:

[0012] PC端flash播放器直播主要采用flv的视频格式进行播放,但是flv视频格式是一个连续的视频流,由若干个tag标签组成,每个tag标签都依赖前一个tag标签,因此当其中的某个tag标签出现问题的时候,视频播放就会出现问题且不能自我恢复,需要重新请求播放视频。

[0013] 2、视频点播问题

[0014] PC端视频点播目前通常采用将mp4、flv文件进行分段,通常视频按每段时长15分钟或30分钟进行分段处理。但是在实现播放过程中,当某一段即将播放完成,进入下一分段视频时,会出现卡顿或黑屏现象。某一分段依赖于前一个分段,因此,在视频快进、快退时,体验不友好。

发明内容

[0015] 本发明的目的在于:解决上述现有技术中的不足,提供一种flash播放器播放hls视频流的转换系统及其工作方法,只需要一种满足hls协议的ts格式的文件就能满足flash播放器和移动终端的需求,显著减少了直播服务器系统的复杂性和降低了系统的不稳定性因素,减少了视频格式转换的时间和步骤,同时大大节省了磁盘空间。

[0016] 为了实现上述目的,本发明采用的技术方案为:

[0017] 一种flash播放器播放hls视频流的转换系统,它包括:

[0018] Internet网络;

[0019] 分别与Internet网络连接的hls服务器和直播前端;所述hls服务器通过CDN网络连接用户端。

[0020] 进一步的,上述的直播前端包括与Internet网络连接的直播编码器和与直播编码器连接的若干台摄像机。

[0021] 进一步的,上述的播放端包括flash播放器、IOS播放器、Android播放器和H5播放器。

[0022] 进一步的,上述的hls服务器包括hls直播服务器和hls点播服务器。

[0023] 进一步的,上述的flash播放器包括:

[0024] 与hls服务器连接的编码器模块和http代理模块,编码器模块用于将切片文件按照flv封装格式进行重新封装;

[0025] 与所述http代理模块连接的flash播放器模块;

[0026] 所述编码器模块与http代理模块连接。

[0027] 进一步的,上述的flash播放器模块还包括协议解析模块。

[0028] 进一步的,上述的http代理模块包括:

[0029] http代理模块,用于接收所述flash播放器的http请求,并响应请求;

[0030] http客户端模块,用于向所述hls服务器请求m3u8文件和切片文件;

[0031] m3u8文件解析模块,用于解析所述hls服务器返回的m3u8文件,获取切片文件表。

[0032] 进一步的,上述的http代理模块还包括动态决策模块,动态决策模块用于判断所述http客户端模块采用并行或串行方式向hls服务器请求切片文件。

[0033] 一种flash播放器播放hls视频流的转换方法,应用权利要求1所述的一种flash播放器播放hls视频流的转换系统,包括以下步骤:

[0034] 步骤一:flash播放器模块进行视频地址分析,根据得到的视频播放地址向http代理模块发送http播放请求;

[0035] 步骤二:http代理模块根据http播放请求向hls服务器请求m3u8文件,hls服务器返回相应的m3u8文件;

[0036] 步骤三:http代理模块解析hls服务器返回的m3u8文件,得到切片文件列表;

[0037] 步骤四:http代理模块根据切片文件列表向hls服务器请求相应的切片文件,hls服务器将相应的切片文件返回至http代理模块;

[0038] 步骤五:http代理模块向编码模块发送待转码的切片文件,编码模块将待转码的切片文件转换为flv格式,然后通过http代理模块返回至flash播放器;

- [0039] 步骤六:flash播放器解码flv格式切片文件并播放。
- [0040] 进一步的,上述步骤三中得到切片文件列表之后还包括判断当前http播放请求为直播请求或点播请求,若当前请求为点播请求,启用动态决策算法。
- [0041] 进一步的,上述启用动态决策算法具体包括以下步骤:
- [0042] 步骤101:预设带宽初始期望值,初始状态时,采用串行方式向hls服务器请求第一个切片文件;
- [0043] 步骤102:每隔一定时间n统计数据接收率R,数据接收率R的计算公式为: $R=S/n$,其中,S代表一个时间间隔n内接收数据的大小;
- [0044] 步骤103:根据数据接收率R判断采用串行方式或并行方式向hls服务器请求切片文件,当数据接收率R大于或等于带宽初始期望值,切换到并行方式,当数据接收率R小于带宽初始期望值,切换到串行方式;
- [0045] 串行方式为依次向hls服务器请求切片文件列表中的切片文件;
- [0046] 并行方式为每次向hls服务器请求多个切片文件列表中的切片文件。
- [0047] 进一步的,采用并行方式请求切片文件具体包括以下步骤:
- [0048] 步骤201:预设第二带宽初始期望值;
- [0049] 步骤202:定义每次请求切片文件的数量为m, $2 \leq m \leq 3$;
- [0050] 步骤203:当数据接收率R满足大于或等于带宽初始期望值,小于或等于第二带宽初始期望值时, $m=2$;
- [0051] 当数据接收率R满足大于第二带宽初始期望值时, $m=3$ 。
- [0052] 进一步的,上述的带宽初始期望值为8Mb/s,所述的第二带宽初始期望值为15Mb/s。
- [0053] 进一步的,步骤四中将切片文件转换为flv格式具体包括以下步骤:
- [0054] 步骤301:接收一个完整的切片文件,获取切片文件的音频ID和视频ID;
- [0055] 步骤302:解析音频ID和视频ID,得到音频数据和视频数据;
- [0056] 步骤303:查找音频数据和视频数据的解码时间戳DTS和显示时间戳PTS;
- [0057] 步骤304:根据解码时间戳DTS和显示时间戳PTS计算flv格式切片文件的参考时间戳Composition Time = (PTS-DTS)/90,
- [0058] 其中,90为切片参考时钟90KHz;
- [0059] 步骤305:预设flv解码时间戳初始值为40ms;
- [0060] 步骤306:将音频数据和视频数据根据flv解码时间戳和参考时间戳按flv的格式重新封装,得到flv格式的音视频数据;
- [0061] 步骤307:将flv格式的音视频数据通过http代理模块返回至flash播放器模块。
- [0062] 进一步的,判断http播放请求为直播请求或点播请求具体为:当m3u8文件以“#EXT-X-ENDLIST”结束时或m3u8文件中包含“#EXT-X-PLAYLIST-TYPE:VOD”,则当前请求为点播请求。
- [0063] 由于采用了上述技术方案,本发明的有益效果是:
- [0064] 本发明只需要一种满足hls协议的ts格式的文件就能满足flash播放器和移动终端的需求,显著减少了直播服务器系统的复杂性和降低了系统的不稳定性因素,减少了视频格式转换的时间和步骤,同时大大节省了磁盘空间。

[0065] 本发明还解决了播放器快进、快退时需要长时间等待的问题和传统视频分段方法中播放器在分段之间切换时出现黑屏的问题。

[0066] 本发明还采用了一种动态决策算法,应用于hls协议播放视频点播的情况,用于解决如何提高flash播放器端与视频服务器端带宽利用率最大化问题。

附图说明

[0067] 图1为本发明的数据传输流程示意图。

[0068] 图2为本发明的flash播放器模块结构示意图。

[0069] 图3为本发明的数据传输时序示意图。

[0070] 图4为本发明的结构示意图。

[0071] 附图标记:1-直播前端,2-hls直播服务器,3-hls点播服务器,4-直播编码器,5-摄像机,6-播放端。

具体实施方式

[0072] 参照附图1-4,对本发明的实施方式做具体的说明。

[0073] 一种flash播放器播放hls视频流的转换系统,包括以下模块:

[0074] flash播放器模块、HTTP代理模块、编码模块组成。

[0075] flash播放器模块:主要功能有播放请求地址解析,获取音视频数据,解码音视频数据,呈现画面,播放声音。所述flash播放器模块还包括协议解析模块,根据传入的视频地址进行判断是否请求hls协议播放视频。

[0076] HTTP代理模块包含以下模块:

[0077] HTTP服务端模块:充当HTTP服务角色,接收所述flash播放器的http请求,并响应请求。

[0078] HTTP客户端模块:充当HTTP客户端角色,向hls服务器请求m3u8文件及ts文件。

[0079] m3u8文件解析模块:解析hls服务器返回的m3u8文件,获取ts文件列表。

[0080] 动态决策模块:采用动态决策,支持采用并行还是串行方式向hls服务器请求ts文件。

[0081] 所谓串行方式:依次向hls服务器请求ts文件列表中的ts文件。

[0082] 所谓并行方式:每次向hls服务器请求ts文件列表中的多个ts文件。

[0083] 根据用户和hls服务器的带宽情况,动态判断采用哪种方式,且在两种方式可自由动态切换。

[0084] 编码模块主要是将ts文件按照flv封装格式进行重新封装。包括以下模块:

[0085] ts解析模块,时间戳转换模块,flv封装模块。

[0086] ts解析模块:收到一个完整的ts文件后,抽取其中的音视频数据。

[0087] 时间戳转换模块:由于ts文件与flv文件使用的时间戳不同,所以要进行转换。

[0088] flv封装模块:将从ts文件抽取的音视频数据按flv文件格式重新封装。

[0089] 图1是一种实现flash播放器播放hls协议的视频方法流程图,包括:

[0090] 步骤S101,flash播放器请求播放视频地址,所述flash播放器进行视频地址分析,若视频地址包含以“m3u8”字符串结尾,则表示当前请求播放的是一个hls协议的视频。

- [0091] 步骤S102,根据视频播放地址,向hls服务器请求m3u8文件。
- [0092] 假如现在请求播放中央电视台一台,其m3u8地址为:http://IP:port/live/cctv.m3u8。
- [0093] hls服务器返回m3u8文件为:
- [0094] #EXTM3U
- [0095] #EXT-X-TARGETDURATION:10
- [0096] #EXT-X-MEDIA-SEQUENCE:0
- [0097] #EXT-X-VERSION:3
- [0098] #EXTINF:10,
- [0099] HTTP://IP:PORT/CCTV1/2018/03/22/19/0.ts
- [0100] #EXTINF:10,
- [0101] HTTP://IP:PORT/CCTV1/2018/03/22/19/1.ts
- [0102] #EXTINF:10,
- [0103] HTTP://IP:PORT/CCTV1/2018/03/22/19/2.ts
- [0104] #EXTINF:10,
- [0105] HTTP://IP:PORT/CCTV1/2018/03/22/19/3.ts
- [0106] #EXTINF:10,
- [0107] HTTP://IP:PORT/CCTV1/2018/03/22/19/4.ts
- [0108] #EXTINF:10,
- [0109] HTTP://IP:PORT/CCTV1/2018/03/22/19/5.ts
- [0110] 步骤S103,解析ts文件地址,根据所述步骤S102中返回的m3u8文件,自上而下解析出所有ts文件地址。
- [0111] 步骤S104,请求ts文件。在所述S104步骤之前,所述步骤S103之后,进一步包括:
- [0112] 判断当前请求是直播请求还是点播请求,当m3u8文件以
- [0113] “#EXT-X-ENDLIST”结束时或m3u8文件中包含
- [0114] “#EXT-X-PLAYLIST-TYPE:VOD”表示当前请求播放点播;
- [0115] 若为点播,启用动态决策算法;
- [0116] 动态决策算法如下:
- [0117] 初始状态时,我们采用串行方式,请求ts文件列表中的第一个ts文件,每n秒统计数据接收率 $R=S/n$,S代表n秒接收数据大小,n代表一个正整数值,定义在 $\{3 \leq n \leq 5\}$ 。当R的值 $\geq 8\text{Mb/s}$ 时,代表播放器与hls服务器之间带宽值 $\geq 8\text{Mb/s}$,此时切换到并行方式。因此 $R \geq 8\text{Mb/s}$ 时,作为串行与并行方式之间动态切换的临界值。 $R \geq 8\text{Mb/s}$ 作为我们一个带宽的初始期望值,这个值可以根据实际带宽环境进行调整。
- [0118] 并行方式下,一次请求的ts文件数量m定义在 $\{2 \leq m \leq 3\}$ 若 $8\text{Mb/s} \leq R \leq 15\text{Mb/s}$,则一次请求ts文件数量m的值为2,若 $R \geq 15\text{Mb/s}$ 时,m的值为3。同理,当 $R < 8\text{Mb/s}$ 时,切换至串行方式。
- [0119] 根据所述动态决策算法,以下m3u8文件将按以下步骤请求ts文件。
- [0120] #EXTM3U
- [0121] #EXT-X-TARGETDURATION:10

- [0122] #EXT-X-MEDIA-SEQUENCE:0
 [0123] #EXT-X-VERSION:3
 [0124] #EXTINF:10,
 [0125] HTTP://IP:PORT/CCTV1/2018/03/22/19/0.ts
 [0126] #EXTINF:10,
 [0127] HTTP://IP:PORT/CCTV1/2018/03/22/19/1.ts
 [0128] #EXTINF:10,
 [0129] HTTP://IP:PORT/CCTV1/2018/03/22/19/2.ts
 [0130] #EXTINF:10,
 [0131] HTTP://IP:PORT/CCTV1/2018/03/22/19/3.ts
 [0132] #EXTINF:10,
 [0133] HTTP://IP:PORT/CCTV1/2018/03/22/19/4.ts
 [0134] #EXTINF:10,
 [0135] HTTP://IP:PORT/CCTV1/2018/03/22/19/5.ts
 [0136] #EXT-X-ENDLIST
 [0137] 首先采用串行方式请求:HTTP://IP:PORT/CCTV1/2018/03/22/19/0.ts文件,当 $R \geq 8\text{Mb/s}$ 时,则动态切换至并行方式请求,则开启一个新的线程请求HTTP://IP:PORT/CCTV1/2018/03/22/19/1.ts文件。当 $R \geq 15\text{Mb/s}$ 时,则开启新的线程请求HTTP://IP:PORT/CCTV1/2018/03/22/19/2.ts。直至已经开始下载的文件下载完成,当R值发生变化,再做出相应的动态决策。动态决策表如下:

[0138]

	$8\text{Mb/s} \leq R < 15\text{Mb/s}$	$R \geq 15\text{Mb/s}$	$R < 8\text{Mb/s}$
并行	并发下载两个TS文件	并发下载三个TS文件	
串行			下载一个视频TS文件

- [0139] 步骤S105,按以下方式将ts文件转flv格式的文件:
 [0140] 1、接收一个完整的ts文件;
 [0141] 2、搜索ts同步字节0x47,查找PMT,PAT表,进一步查找音频ID,视频ID;
 [0142] 3、根据2步中的查找到音频ID和视频ID,解析出音频数据和视频数据;
 [0143] 4、查找DTS(解码时间戳)和PTS(显示时间戳);
 [0144] 5、根据4中的值计算出flv需要的Composition Time(参考时间戳) = $(\text{PTS} - \text{DTS}) / 90$,其中90为ts参考时钟90KHz。flv TimeStamp(解码时间戳)初始值为0,每一个视频tag按的TimeStamp在前一个TimeStamp的值上面加40,其中40的单位为ms(毫秒),为什么加40呢,这是因为当前视频帧率基本在20-30帧/秒,所以一帧播放的时间基本在40毫秒左右。
 [0145] 6、根据音视频数据和TimeStamp以及Composition Time按flv的格式重新封装音视频。

[0146] 步骤S106,将封装的flv数据返回给flash播放器。

[0147] 步骤S107,flash播放器解码flv文件,解码音视频文件,呈现画面和播放声音。

[0148] 综上所述,本申请在不增加任何设备,不需要任何技术改造的情况下,只需要一路hls协议的直播流就可以满足flash播放器和移动端播放直播的需求,解决了传统直播服务器要分别为flash播放器和移动端提供直播流的问题,本申请显著减少了直播服务器系统的复杂性和降低了系统的不稳定性因素。

[0149] 相比传统视频点播,分别为flash播放器和移动端提供不同视频格式的文件,既同一个视频文件要转码成两种不同的格式的视频文件,一种满足flash播放器的要求,另一种满足移动终端的需求。本申请只需要一种满足hls协议的ts格式的文件就能满足flash播放器和移动终端的需求。本申请减少了视频格式转换的时间和步骤,同时大大节省了磁盘空间。相比传统视频点播,本申请还解决了播放器快进、快退时需要长时间等待问题,以及传统视频分片方法中,播放器在分片之间切换时出现黑屏问题。

[0150] 本申请还采用了一种动态决策算法,用于解决如何提高flash播放器端与视频服务器端带宽利用率最大化问题。

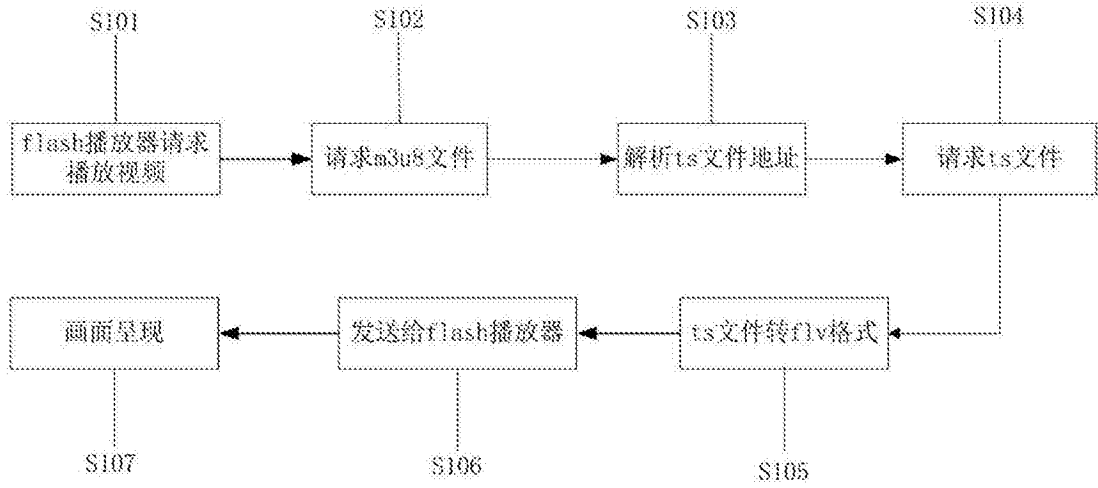


图1

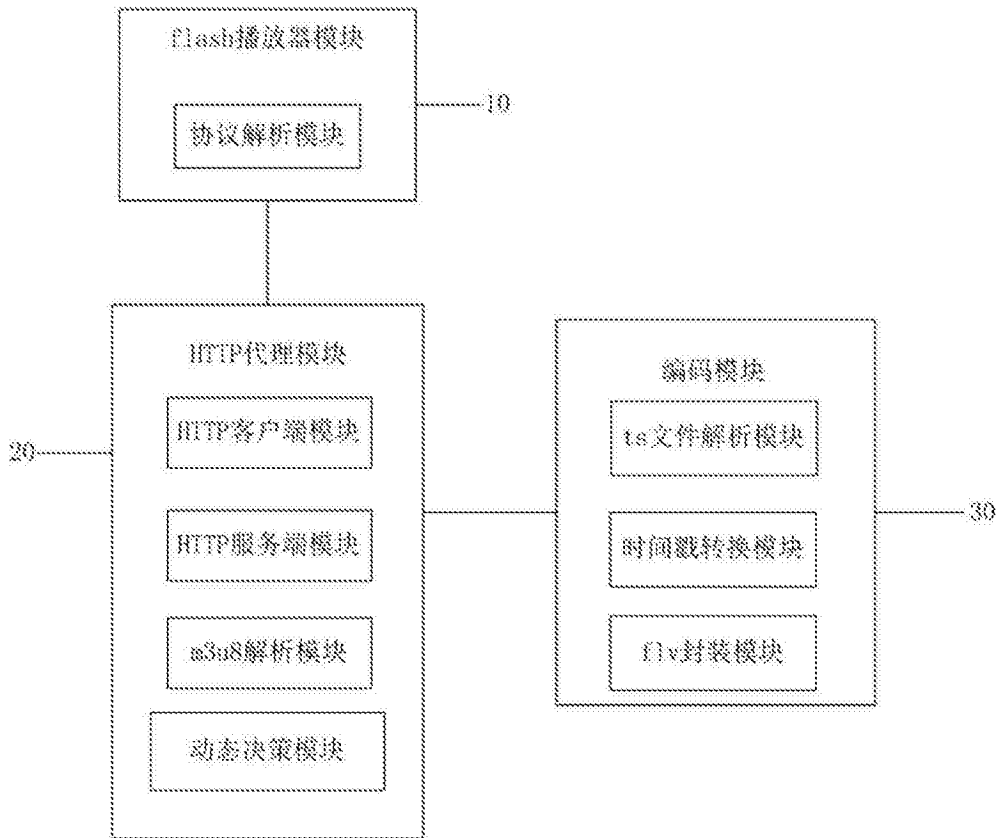


图2

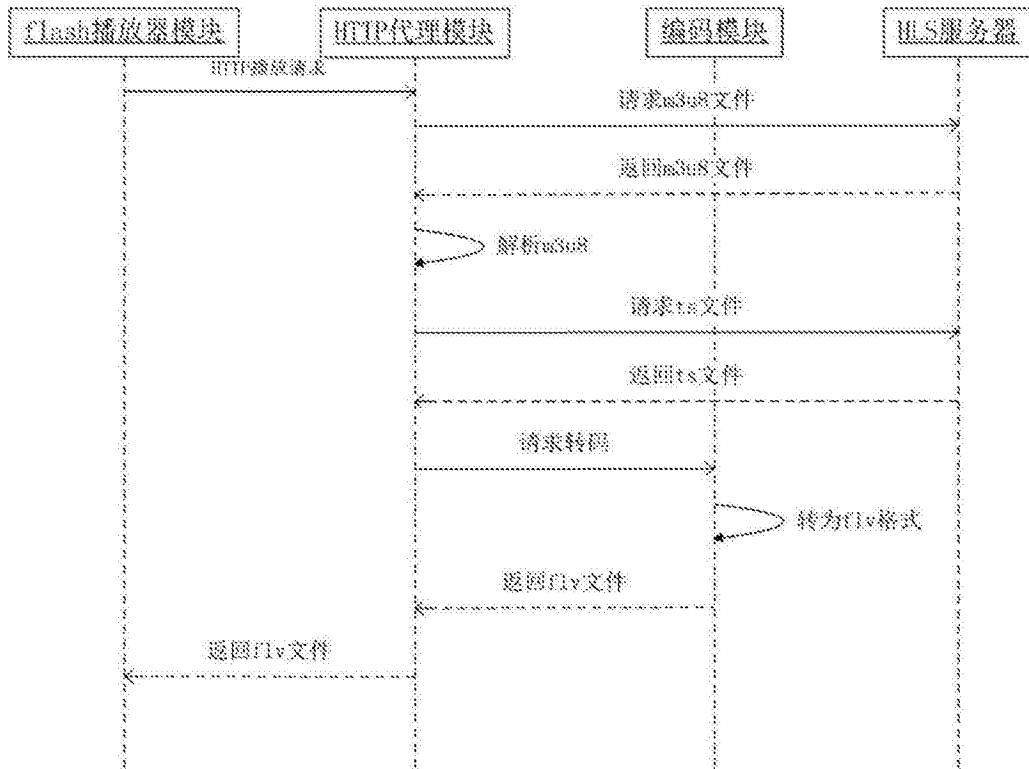


图3

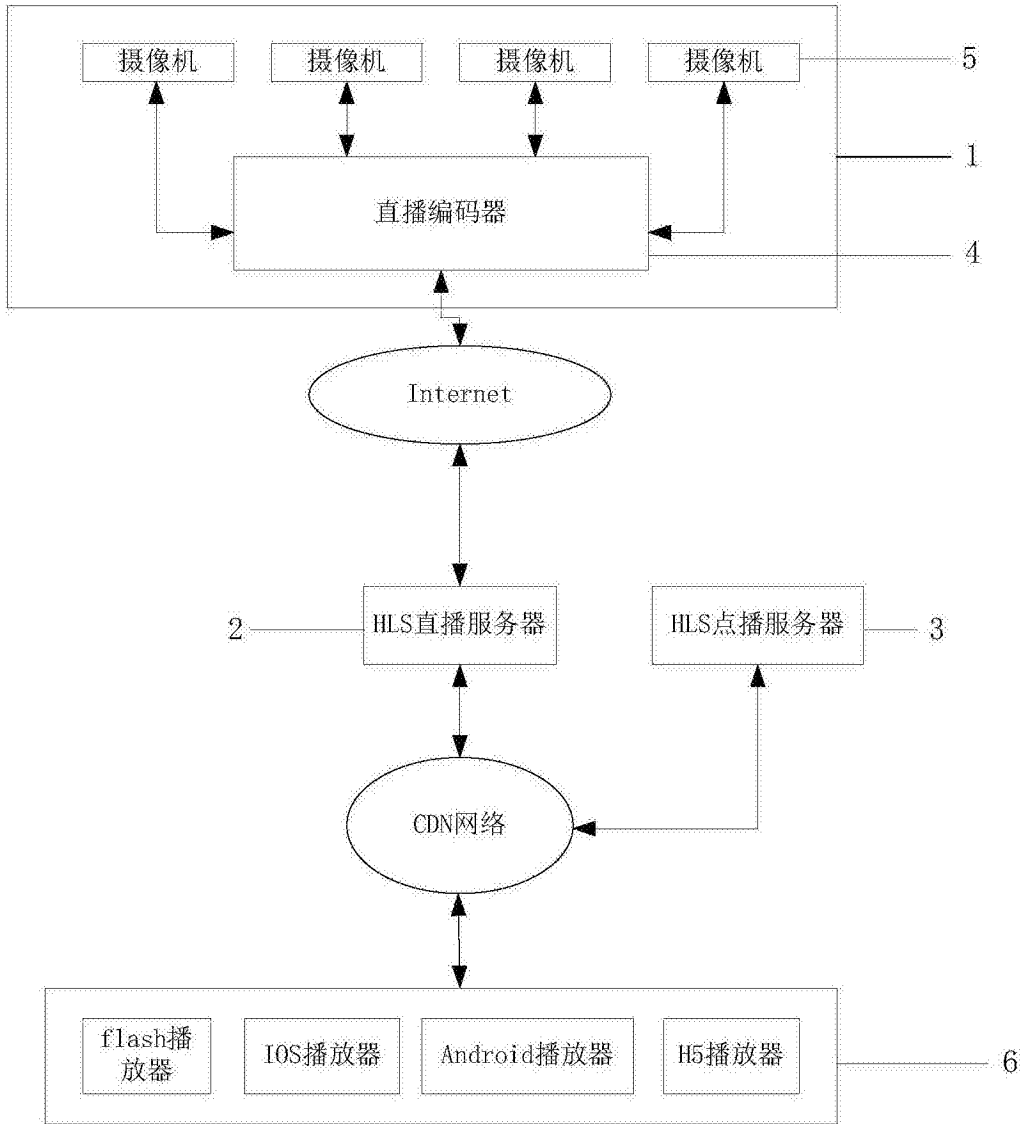


图4