

JD视频存储服务器设计方案

作者: 曾昊

时间: 2015-2-26

引言

概述

JD视频存储服务器是JD智能云视频服务器的一个子系统，为JD智能云的视频转发服务器提供视频存储服务功能，实现将流媒体转换为mp4视频文件并存储到京东云。

读者

本文档的读者仅限于京东智能云的项目管理人员和视频转发服务器的开发人员，未经允许请不要转发或引用本文档。

参考文档

《京东流媒体服务器详细设计》	李纯
《视频存储概要设计》	张政

需求分析

概述

JD视频转发服务器是JD智能云的一个子系统，主要为视频采集设备提供基于RTSP流媒体的转发和分发服务，这个视频采集设备多用于实时的监控业务，因此一段时间内的视频数据的存储和回放是基本的业务功能。

视频存储服务器将基于目前的流媒体转发服务器的架构和功能，为JD智能云的视频采集设备提供视频存储功能。

功能需求

视频存储服务有以下功能：

1. 作为RTSP的客户端接收来自智能云服务器的参数并从视频转发服务器上取得视频流；
2. 把RTSP的视频流在保持音视频编码的条件下以mp4的文件格式保存到指定的存储器；
3. 有定时功能，只存储指定时间长度的视频；
4. 可以把保存的视频文件的基本参数上传到智能云服务器；
5. 能监测和实时上报单个视频存储服务进程的健康状态到智能云服务器；

以上是视频存储服务的基本功能，在视频存储服务的第一个开发周期完成，如果有新的需求在后续继续开发。

注：

1. 目前视频存储服务只支持H264和AAC格式;
2. 视频文件目前只能保存到京东云存储;
3. 基本参数包括视频的时间长度, 开始记录的时间、文件在云存储对应的key和桶名、文件的大小

性能需求

考虑到视频下载和视频转换对于带宽和CPU的占用比较大, 所以视频存储服务器需要满足以下性能上的需求:

1. 单个视频存储服务进程所使用的内存上限是1G;
2. 单台服务器上活跃的视频存储服务进程数上限是CPU核心数;
3. 如果可能应该使用硬件编解码器进行视频格式转换;

约束

视频存储服务器的约束如下:

1. 存储的文件格式是mp4, 视频的编码器是h264, 音频编码器是AAC;
2. 单个视频文件的大小不超过120M;
3. 对单个用户视频文件的累计大小不超过10G;

概要设计

服务部署的软硬件环境

视频存储服务器部署按设计要求需要部署到CentOS 64bit的操作系统上, 建议采用以下配置:

- CPU核心数不少于8;
- 物理内存不小于16G;
- 网卡速率不小于1000Mbps;
- 硬盘可用空间不小于100G;

服务开发的软硬件环境

视频存储服务器的开发环境配置如下:

- CentOS6.3 64bit服务器;
- CPU核心数为8;
- 物理内存16G;
- gcc4.2以上版本
- 库支持: yasm、libx264、libfdk_aac、jss-sdk-c;
- 编译管理工具waf1.8以上版本;
- 版本管理svn1.6以上版本;

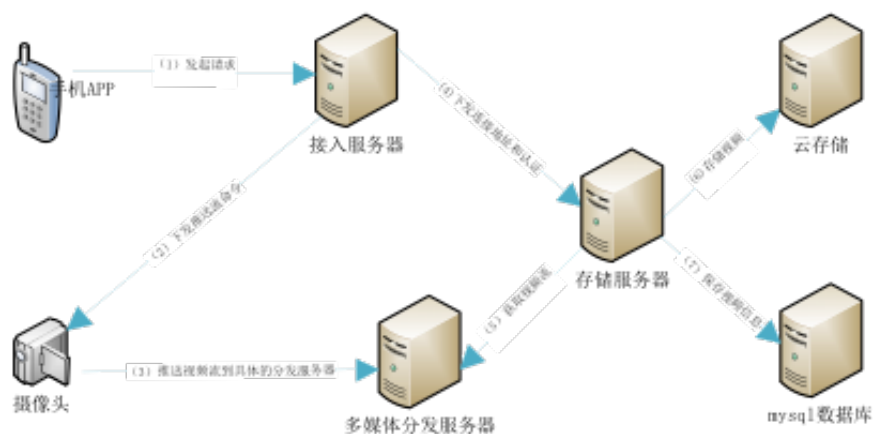
系统结构



注:

- ffmpeg不会被完整的编译，只打包里面基本的模块，如：libavformat, libavcodec等；
- common里面包括基本的参数解析，socket模块，健康状态这些基本业务模块；
- 视频存储服务目前只支持x264和aac编码的输入和输出；

网络结构



上图来自《视频存储概要设计》，基本原理是视频存储服务作为RTSP的客户端连接到视频转发服务器，从视频转发服务器上取得对应的视频流并进行转换和存储。

数据结构

服务器心跳数据格式

```
{
  ip: xxx.xxx.xxx.xxx;
  mem: xxx;
  cpu: xxx;
  pro: xxx;
  core: xxx;
}
```

注:

- mem是当前已经使用的内存数，单位是MB；
- cpu是当前已经使用的CPU，单位是%；
- pro是当前活跃的存储服务进程数；
- core是服务器的核心数；

存储服务异常数据格式

```
{
  ip: xxx.xxx.xxx.xxx;
  feed_id: <feed id>;
  uid: <user id>;
  url: <rtsp url>;
}
```

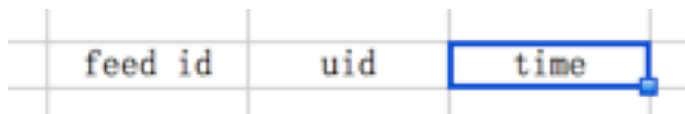
视频记录命令下发格式

```
{
  feed_id: <feed id>;
  uid: <uid>;
  duration: -1;
  command: start/stop;
  url: rtsp://xxxx/xxx.sdp?feed=xx&token=xxx;
}
```

视频文件信息上报格式

```
{
  feed_id: <feed id>;
  bucket: <bucket name>;
  key: <key>;
  duration: xxxx sec;
  size: xxxx;
  start: 2015-4-11 12:12:12;
}
```

视频文件key生成格式



注：

- feed id是视频采集设备的识别码；
- uid是用户的id；
- time是记录的开始时间，是精确到毫秒的长整型；
- 各个值之间以中划线'-'作为分割符

例如：

1001234-051987-1089898834134

详细设计

概述

视频存储服务器包括主要的3层，最低层是第三方的编解码器，直接编译成静态库后打包即可，不需要开发，主要的开发工作集中在第二层，ffmpeg的库的定制和common工具模块。第一层是视频存储服务器的主要逻辑代码。

common模块

概述

common里面包含视频存储服务的一些基本的业务功能模块，如看门狗功能模块，socket服务监听模块等，下面分别描述。

socket服务器

http客户端

HA服务模块（高可靠性服务）

HA包括两个部分，一个是一个独立的HA服务器，另一个部分是集成在视频存储程序里的HA模

块。

HA服务器

HA模块

云存储文件上传

JSON解析器和编码器

ffmpeg模块

从ffmpeg中抽取需要的模块，在jd-mss项目中编译成静态库，jd-mss直接打包到代码中即可。

主要的工作是抽取ffmpeg的libav开头的文件夹到jd-mss工程中，通过修改wscript文件的内容去编译静态库，然后在主工程中引用这些静态库。

ffmpeg的依赖关系

ffmpeg中包含libavfilter、libswscale、libavformat、libavcodec、libswresample、libavutil、libavdevice这几个主要的库，系统需要除了libavdevice以外的所有的库，库之间的依赖关系如下：

- libavutil: none
- libswresample: libavutil
- libswscale: libavutil
- libavresample: libavutil
- libavcodec: libswresample libavutil
- libavformat: libavcodec libswresample libavutil
- libavfilter: libswscale libavformat libavcodec libswresample libavutil

jd-mss程序

概述

基本业务流程图

重要逻辑流程图