

## 1. 使用 ffmpeg 建立直播系统

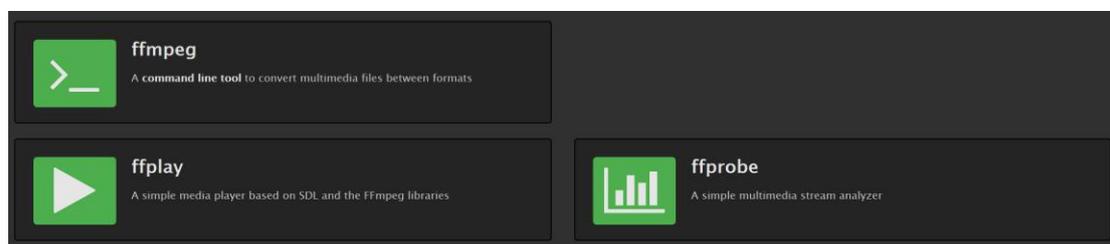


FFmpeg 是一套可以用来记录、转换数字音频、视频，并能将其转化为流的开源计算机程序。采用 LGPL 或 GPL 许可证。它提供了录制、转换以及流化音视频的完整解决方案。它包含了非常先进的音频/视频编解码库 libavcodec，为了保证高可移植性和编解码质量，libavcodec 里很多 code 都是从头开发的。

FFmpeg 在 Linux 平台下开发，但它同样也可以在其它操作系统环境中编译运行，包括 Windows、Mac OS X 等。这个项目最早由 Fabrice Bellard 发起，2004 年至 2015 年间由 Michael Niedermayer 主要负责维护。许多 FFmpeg 的开发人员都来自 MPlayer 项目，而且当前 FFmpeg 也是放在 MPlayer 项目组的服务器上。项目的名称来自 MPEG 视频编码标准，前面的“FF”代表“Fast Forward”。

实验过程使用“FFmpeg Tools”中的三个程序。

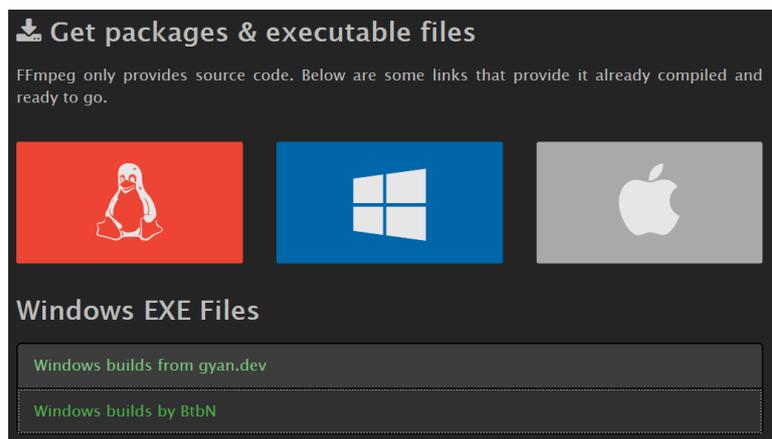
<https://ffmpeg.org/about.html>



### 1.1 实验环境准备

#### 1.1.1 下载 ffmpeg 工具包可执行文件

<https://ffmpeg.org/download.html>



建议如下 2 个编译版本之一（含程序的使用说明文档），

<https://www.gyan.dev/ffmpeg/builds/>

<https://github.com/BtbN/FFmpeg-Builds/releases>

#### 1.1.2 下载 H.264 视频 & mp3

- (1) 下载一个 mp3 音频文件。

- (2) 下载一个 H.264 格式的视频文件。

[http://clips.vorwaerts-gmbh.de/big\\_buck\\_bunny.mp4](http://clips.vorwaerts-gmbh.de/big_buck_bunny.mp4)

<http://vjs.zencdn.net/v/oceans.mp4>

<https://media.w3.org/2010/05/sintel/>

<http://mirror.aarnet.edu.au/pub/TED-talks/>

## 1.2 使用 ffprobe 程序统计流信息

- (3) 阅读 ffprobe 帮助文档 (<https://ffmpeg.org/ffprobe.html>)。
- (4) 用 ffprobe 统计一个 mp3 文件采样频率，码率等基本信息并记录。
- (5) 用 ffprobe 统计一个 mp4 文件的比特流中含有多少个媒体流，各个媒体流所占的比例、码率、时长等基础信息并记录。
- (6) 统计视频的分辨率、帧率等基本信息并记录。
- (7) 分析该 mp4 文件中的 DTS 和 PTS 信息（选一个片段记录 DTS 和 PTS，并进行细致分析）。
- (8) DTS 和 PTS 数值的单位是？

## 1.3 使用 ffmpeg 编辑视频

- (9) 阅读 ffmpeg 帮助文档 (<https://ffmpeg.org/ffmpeg.html>)
- (10) 运行 “ffmpeg -codec”，记录结果中 3 种视频和 3 种音频格式（课程讲过的）。
- (11) 将一个视频文件的帧率翻倍，观察音视频的同步关系并记录。如果出现不同步，请予以解决，并记录解决过程。
- (12) 将一个 mp4 文件中的视频和音频分离存储至两个不同的文件，记录所使用的命令行。
- (13) 自选一个 MP4 格式的视频，抽取其中 10 秒钟视频信息转换为 gif 格式的动图。
- (14) 给一个视频添加水印（文字或者图片）。
- a) 可参考 [https://www.cnblogs.com/leisure\\_chn/p/10434209.html](https://www.cnblogs.com/leisure_chn/p/10434209.html)

## 1.4 使用 ffplay 播放视频

- (15) 阅读 ffplay 帮助文档 (<https://ffmpeg.org/ffplay.html>)
- (16) 播放一个 mp4 文件。
- (17) 播放过程中控制操作（暂停、继续、增加音量、减小音量...）的按键是？
- (18) 改变显示窗口的大小。
- a) 提示：参数 -x 、 -y
- (19) 尝试不同的音视频同步方式播放 mp4，记录使用的命令并简要分析结果。
- a) 提示：ffplay 中音视频同步的实现方式其实有三种，分别是：以音频为主时间轴作

为同步源，以视频为主时间轴作为同步源，以外部时钟为主时间轴作为同步源。

- b) 参数 -sync

## 1.5 使用 ffmpeg 和 ffprobe 建立简易直播系统

(20) 基于 ffmpeg 将一个 H.264 格式的视频 RTP 推流。

- a) 可参考 <https://blog.csdn.net/zhoubotong2012/article/details/86711097>
- b) 示例：ffmpeg -re -i d:\videos\1080P.264 -vcodec copy -f rtp rtp://127.0.0.1:1234

(21) 用 ffplay 接收上述 RTP 流并播放。

- a) 示例：ffplay -protocol\_whitelist "file,udp,rtp" -i rtp://127.0.0.1:1234

(22) 基于 ffmpeg 采集本机摄像头视频并 RTP 推流。

- a) 可参考 <https://blog.csdn.net/zhoubotong2012/article/details/102774983>

## 1.6 nginx + ffmpeg 流媒体服务器搭建

(23) 上述简易的直播系统仅可以有一个接收端，不能满足多用户同时访问的需求。搭建 nginx 的流媒体服务，将 ffmpeg 采集本机摄像头视频推流，可实现多用户同时访问。

- a) 可自行查阅有关资料完成流媒体服务器搭建，如
  - i. <https://blog.csdn.net/bvngh3247/article/details/80405423>
  - ii. <https://www.jb51.net/article/107405.htm>
- b) 流媒体播放器可选择 VLC media player (<https://www.videolan.org/vlc/>)。

## 1.7 思考题

(24) 请解释，使用 ffplay 接收 ffmpeg 推送的 RTP 视频流时，SDP 文件的作用。

(25) ffmpeg 参数-vcodec 作用？有哪些可能的参数值？

(26) ffmpeg 参数-acodec 作用？有哪些可能的参数值？

(27) Nginx 所起的作用？