

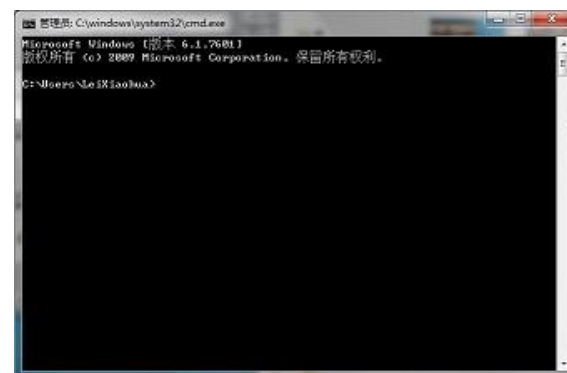
# 基于 FFmpeg + SDL 的视频播放器的制作

## (5) FFmpeg+SDL视频播放器

雷霄骅

leixiaohua1020@126.com

<http://blog.csdn.net/leixiaohua1020>



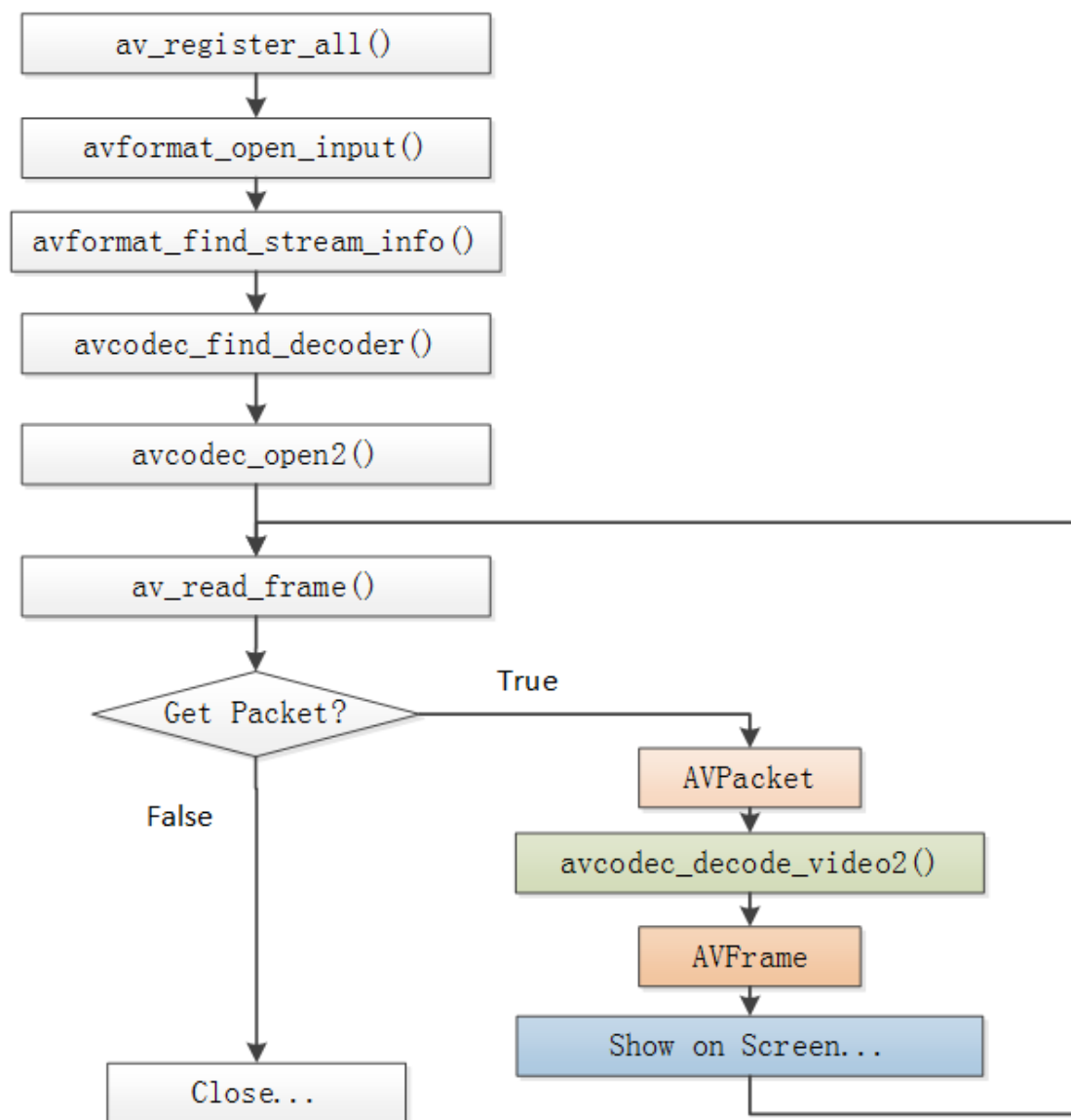
# 目录

- FFmpeg知识回顾
- SDL知识回顾
- FFmpeg和SDL的整合实现视频播放
- 进阶：脱离开发环境的独立播放器
- 练习

# FFmpeg 知识回顾

- 演示

- FFmpeg解码的流程图如下所示



## • FFmpeg解码函数简介

- `av_register_all()`：注册所有组件。
- `avformat_open_input()`：打开输入视频文件。
- `avformat_find_stream_info()`：获取视频文件信息。
- `avcodec_find_decoder()`：查找解码器。
- `avcodec_open2()`：打开解码器。
- `av_read_frame()`：从输入文件读取一帧压缩数据。
- `avcodec_decode_video2()`：解码一帧压缩数据。
- `avcodec_close()`：关闭解码器。
- `avformat_close_input()`：关闭输入视频文件。

## • FFmpeg数据结构简介

### ▫ AVFormatContext

- 封装格式上下文结构体，也是统领全局的结构体，保存了视频文件封装格式相关信息。

### ▫ AVInputFormat

- 每种封装格式（例如FLV，MKV，MP4，AVI）对应一个该结构体。

### ▫ AVStream

- 视频文件中每个视频（音频）流对应一个该结构体。

### ▫ AVCodecContext

- 编码器上下文结构体，保存了视频（音频）编解码相关信息。

### ▫ AVCodec

- 每种视频（音频）编解码器（例如H.264解码器）对应一个该结构体。

### ▫ AVPacket

- 存储一帧压缩编码数据。

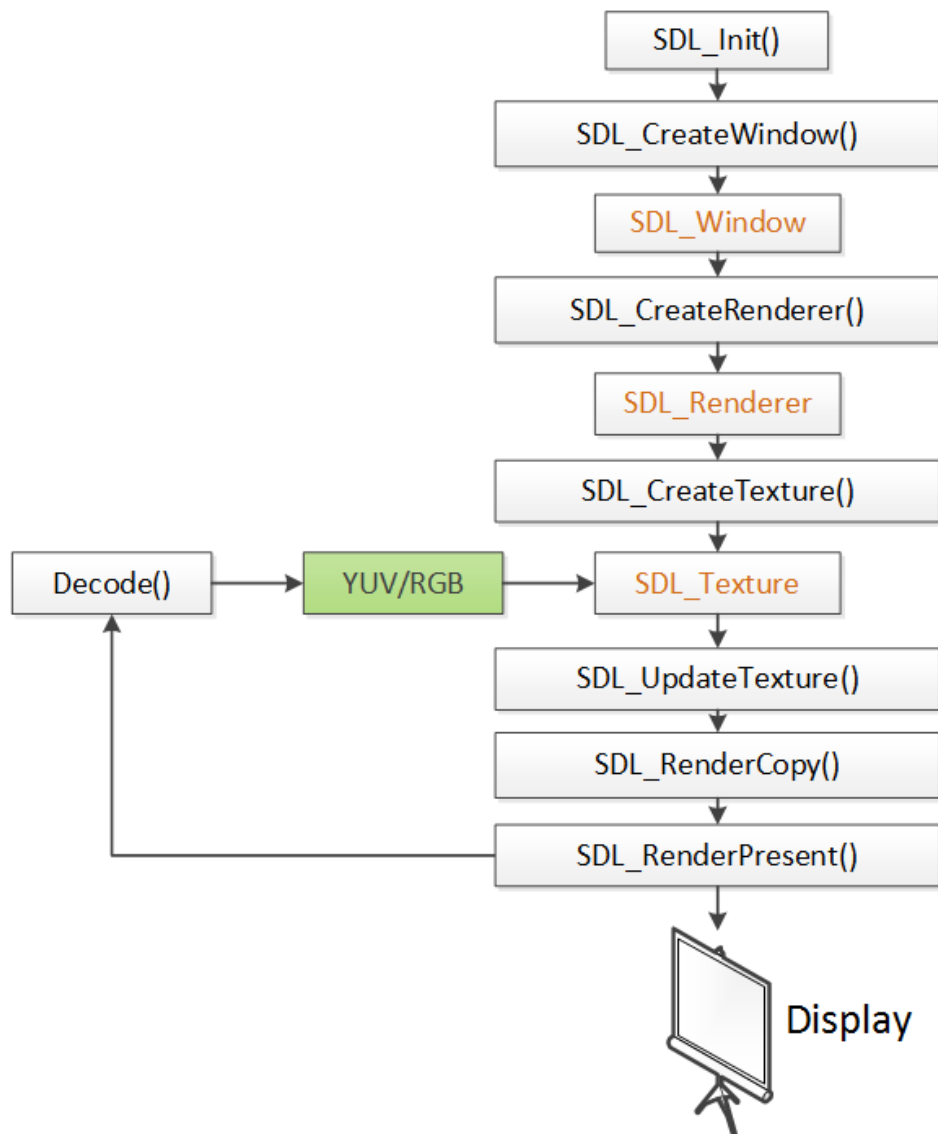
### ▫ AVFrame

- 存储一帧解码后像素（采样）数据。

# SDL 知识回顾

- 演示

- SDL视频显示的流程图如下所示





## • SDL视频显示函数简介

- `SDL_Init()`：初始化SDL系统
- `SDL_CreateWindow()`：创建窗口SDL\_Window
- `SDL_CreateRenderer()`：创建渲染器SDL\_Renderer
- `SDL_CreateTexture()`：创建纹理SDL\_Texture
- `SDL_UpdateTexture()`：设置纹理的数据
- `SDL_RenderCopy()`：将纹理的数据拷贝给渲染器
- `SDL_RenderPresent()`：显示
- `SDL_Quit()`：退出SDL系统

- SDL数据结构简介

- SDL\_Window

- 代表了一个“窗口”

- SDL\_Renderer

- 代表了一个“渲染器”

- SDL\_Texture

- 代表了一个“纹理”

- SDL\_Rect

- 一个简单的矩形结构

# FFmpeg和SDL整合实现视频播放器

- 整合方式

- FFmpeg解码器实现了：视频文件→YUV
- SDL视频显示实现了：YUV→屏幕
- FFmpeg+SDL整合之后实现了：视频文件→YUV→屏幕

# 进阶：脱离开发环境的独立播放器

- `main()` 函数的参数
  - `argc argv`: 全称为ARGument Counter 和 ARGument Vector。其中`argv`存储了来自于命令行的参数；而`argc`存储了参数的个数。
  - 例如在命令行中输入“`ffmpeg -i test.mkv test.ts`”，则`argc`取值为4，而`argv[]`数组取值如下：  

```
argv[0]="ffmpeg"  
argv[1]="-i"  
argv[2]="test.mkv"  
argv[3]="test.ts"
```
- 动态链接库（\*.dll）
  - 动态链接库不能被编译进应用程序。因而使用应用程序的时候必须在相同目录下保存用到的动态链接库文件。

# 练习

- 修改源代码。整合《Ffmpeg视频解码器》和《SDL视频显示器》形成视频播放器，要求
  - 可以提取并打印视频文进相关的信息
  - 可以脱离开发环境，在命令行下播放任意一个视频文件
  - 可以暂停/继续播放