

基于 FFmpeg + SDL 的视频播放器的制作

(4) SDL 视频显示

雷霄骅

leixiaohua1020@126.com

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



目录

- 视频显示知识
- SDL简介
- VC下SDL开发环境的搭建
- 示例程序运行
- SDL视频显示的函数
- SDL视频显示的数据结构
- 进阶-示例程序运行
- 进阶-SDL中多线程和事件
- 练习

视频显示知识

- 视频显示的流程
 - 视频显示的流程，就是将像素数据“画”在屏幕上的过程。
 - 例如显示YUV，就是将YUV“画”在系统的窗口中。

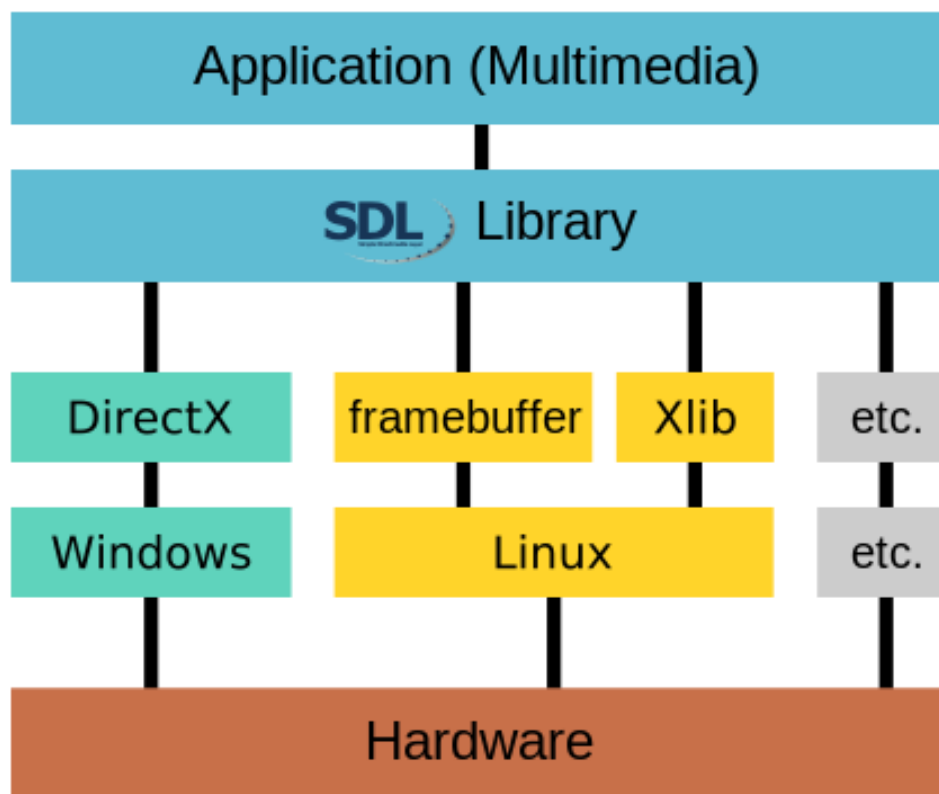
SDL 简介

- 作用
 - SDL (Simple DirectMedia Layer) 库的作用说白了就是封装了复杂的视音频底层交互工作，简化了视音频处理的难度。
 - 本课程中只涉及到SDL库的一小部分——视频显示部分。
- 特点
 - 跨平台
 - 开源

PS：尽管SDL已经简化了视音频底层交互工作，对于新手来说学习还是有一些难度的。

- 结构

- SDL结构如下所示。可以看出它实际上还是调用了DirectX等底层的API完成了和硬件的交互。



VC下SDL开发环境的搭建

- 新建控制台工程
 - 打开VC++
 - 文件->新建->项目->Win32控制台应用程序
- 拷贝SDL开发文件
 - 头文件 (*.h) 拷贝至项目文件夹的include子文件夹下
 - 导入库文件 (*.lib) 拷贝至项目文件夹的lib子文件夹下
 - 动态库文件 (*.dll) 拷贝至项目文件夹下

- 配置开发文件

- 打开属性面板

- 解决方案资源管理器->右键单击项目->属性

- 头文件配置

- 配置属性->C/C++->常规->附加包含目录，输入“include”（刚才拷贝文件的目录）

- 导入库配置

- 配置属性->链接器->常规->附加库目录，输入“lib”（刚才拷贝文件的目录）
 - 配置属性->链接器->输入->附加依赖项，输入“SDL2.lib; SDL2main.lib”（导入库的文件名）

- 动态库不用配置

• 测试

▫ 创建源代码文件

- 在工程中创建一个包含main()函数的C/C++文件（如果已经有了可以跳过这一步），后续步骤在该文件中编写源代码。

▫ 包含头文件

- 如果是C语言中使用SDL，则直接使用下面代码

```
#include "SDL2/SDL.h"
```

- 如果是C++语言中使用SDL，则使用下面代码

```
extern "C"  
{  
#include "SDL2/SDL.h"  
}
```

▫ main() 中调用一个SDL的接口函数

- 例如下面代码初始化了SDL

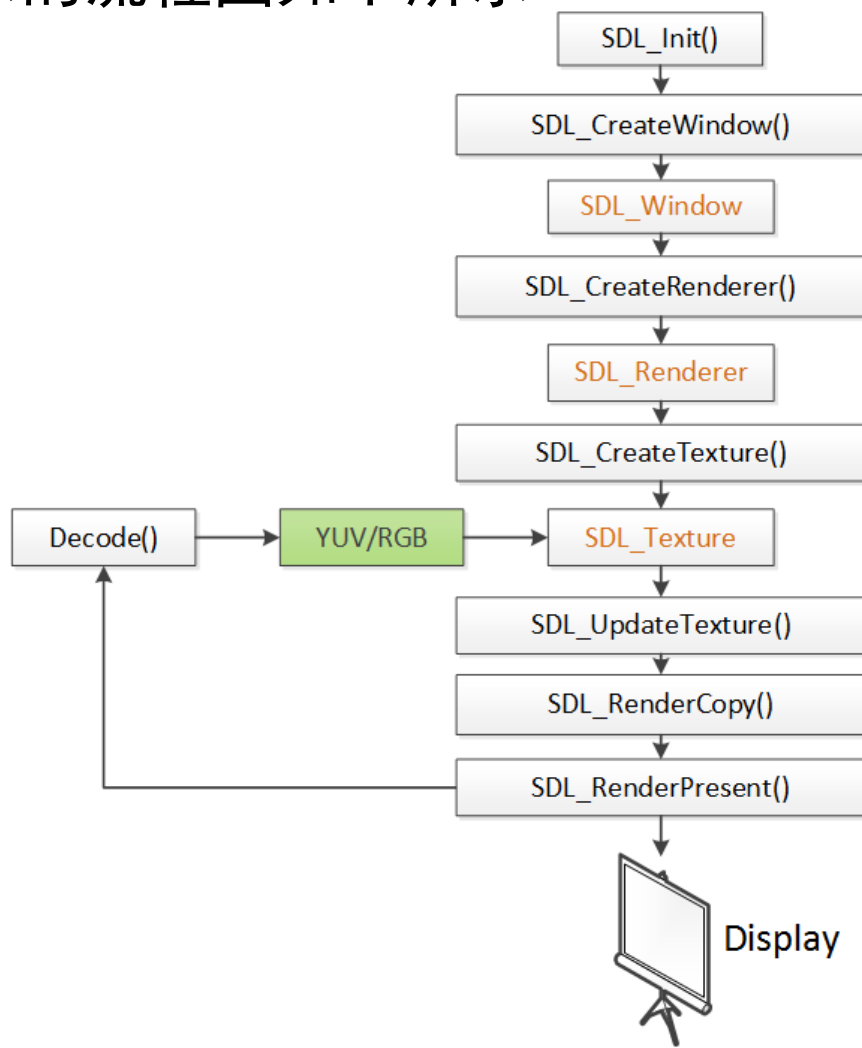
```
int main(int argc, char* argv[]){  
if(SDL_Init(SDL_INIT_VIDEO)) {  
    printf("Could not initialize SDL - %s\n", SDL_GetError());  
} else{  
printf("Success init SDL");  
}  
return 0;  
}
```

如果运行无误，则代表SDL已经配置完成。

示例程序运行

SDL视频显示的函数

- SDL视频显示的流程图如下所示



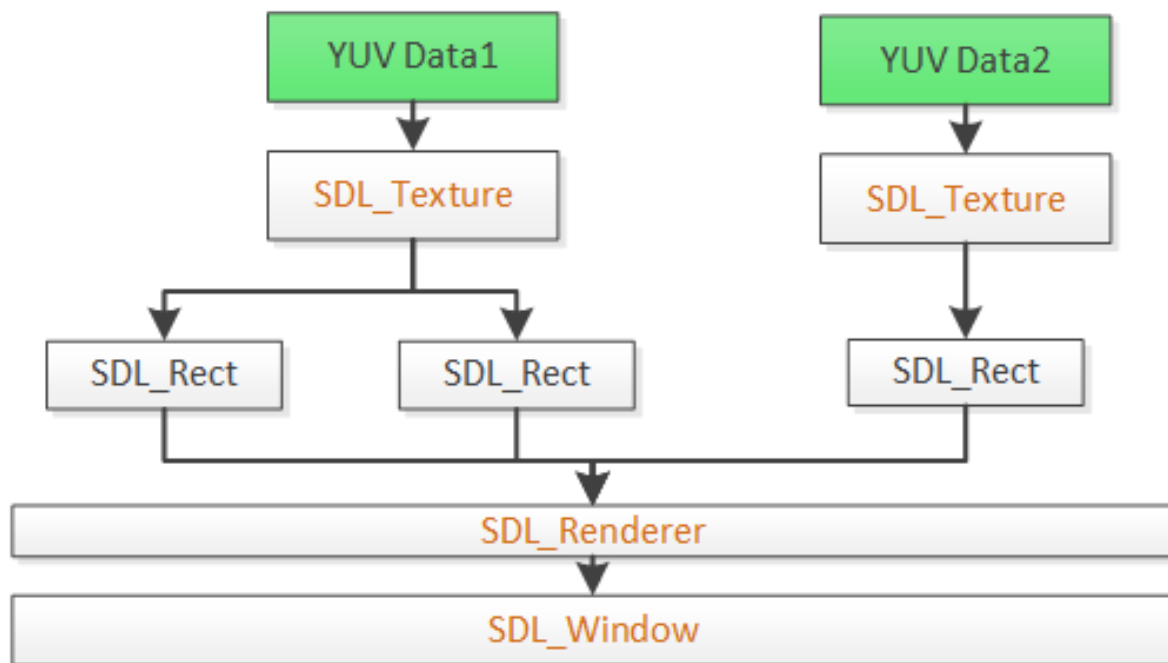
• SDL视频显示函数简介

- `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_Delay()`：工具函数，用于延时。
- `SDL_Quit()`：退出SDL系统

PS：初次学习，一定要将这些函数名称熟记于心

SDL视频显示的数据结构

- SDL视频显示的数据结构如下所示



- 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

PS：初次学习，一定要将这些结构体名称熟记于心

进阶-示例程序运行

进阶-SDL中事件和多线程

- SDL多线程

- 函数

- `SDL_CreateThread()`：创建一个线程

- 数据结构

- `SDL_Thread`：线程的句柄

- SDL事件

- 函数

- `SDL_WaitEvent()` 等待一个事件
 - `SDL_PushEvent()` 发送一个事件

- 数据结构

- `SDL_Event`：代表一个事件

练习

- 给源代码中每个SDL函数添加中文注释
- 修改源代码。对于测试文件，实现以下几种显示
 - 基本练习
 - 二倍速度
 - 二倍宽高
 - 窗口大小固定为500x500
 - 视频周围包围10像素的“黑框”
 - 换一段测试YUV素材进行播放
 - 进阶练习
 - 窗口可以移动
 - 窗口可以调整大小
 - 按下空格键（SPACE）后暂停，再次按下空格后继续播放
 - 显示黑白图像