

## 最简单的基于DirectShow的示例：视频播放器

2015年01月10日 18:29:47 阅读数：12893

最简单的基于DirectShow的示例文章列表：

[最简单的基于DirectShow的示例：视频播放器](#)

[最简单的基于DirectShow的示例：视频播放器图形界面版](#)

[最简单的基于DirectShow的示例：视频播放器自定义版](#)

[最简单的基于DirectShow的示例：获取Filter信息](#)

本文记录一个最简单的基于DirectShow的视频播放器。DirectShow是一个庞大的框架，可以在Windows下实现多种多样的视频处理需求。但是它的“庞大”也使得新手不太容易学习它的使用。本文的例子正是为解决这一问题而做的，它只包含了使用DirectShow播放一个视频文件所需要的最重要的函数。

## 流程图

最简单的使用DirectShow播放视频文件的流程如下图所示。

流程图中涉及到几个接口如下所示。

IGraphBuilder：继承自IFilterGraph，用于构建Filter Graph。相比于IFilterGraph来说IGraphBuilder提供了一些更加“智能”的方法，例如RenderFile()方法。

IMediaControl：提供和播放控制有关的一些接口。

IMediaEvent：用来处理Filter Graph发出的事件。

流程图中关键函数的作用如下所示。

CoInitialize()：初始化COM运行环境。

CoCreateInstance(...,pGraph)：用指定的类标识符创建一个Com对象。在该播放器中类标识符为“CLSID\_FilterGraph”，用于创建IGraphBuilder。

pGraph->QueryInterface(...,pControl)：通过QueryInterface()查询某个组件是否支持某个特定的接口。在这里查询IMediaControl接口。

pGraph->QueryInterface(...,pEvent)：同上。在这里查询IMediaEvent接口。

pGraph->RenderFile("xxx.mkv")：为指定的文件智能的构建一个Filter Graph。

pControl->Run()：开始运行Filter Graph中的所有Filter。

pEvent->WaitForCompletion()：等待Filter Graph处理完所有数据。

CoUninitialize()：释放CoInitialize()初始化的COM运行环境。

注意上述几个函数是构建一个基于DirectShow的视频播放器所必须的函数，除了上述几个接口之外还经常用到以下几个接口：

IBasicVideo：提供和视频有关的一些接口。

IBasicAudio：提供和音频有关的一些接口。

IVideoWindow：提供和窗口有关的一些接口。

IMediaSeeking：提供和播放位置有关的一些接口。

## 源代码

```
[cpp]
1.  /**
2.   * 最简单的基于DirectShow的视频播放器
3.   *  Simplest DirectShow Player
4.   *
5.   *  雷霄骅 Lei Xiaohua
6.   *  leixiaohua1020@126.com
7.   *  中国传媒大学/数字电视技术
8.   *  Communication University of China / Digital TV Technology
9.   *  http://blog.csdn.net/leixiaohua1020
10.  *
11.  *  本程序是一个最简单的基于DirectShow的播放器。
12.  *  适合初学者学习DirectShow。
```

```

13.  *
14.  * This example is the simplest Player based on DirectShow.
15.  * Suitable for the beginner of DirectShow.
16.  */
17.
18.  #include "stdafx.h"
19.
20.  #include <dshow.h>
21.  #include <atlconv.h>
22.
23.
24.  #define OUTPUT_INFO 1
25.
26.  //Show Filter in FilterGraph
27.  int show_filters_in_filtergraph(IGraphBuilder *pGraph){
28.      printf("Filters in FilterGraph=====\n");
29.      USES_CONVERSION;
30.      IEnumFilters *pFilterEnum=NULL;
31.      if(FAILED(pGraph->EnumFilters(&pFilterEnum))){
32.          pFilterEnum->Release();
33.          return -1;
34.      }
35.      pFilterEnum->Reset();
36.      IBaseFilter * filter = NULL;
37.      ULONG fetchCount = 0;
38.      //Pin Info
39.      while (SUCCEEDED(pFilterEnum->Next(1, &filter, &fetchCount)) && fetchCount){
40.          if (!filter){
41.              continue;
42.          }
43.          FILTER_INFO FilterInfo;
44.          if (FAILED(filter->QueryFilterInfo(&FilterInfo))){
45.              continue;
46.          }
47.          printf("[%s]\n",W2A(FilterInfo.achName));
48.          filter->Release();
49.      }
50.      pFilterEnum->Release();
51.      printf("=====\n");
52.      return 0;
53.  }
54.
55.
56.  int _tmain(int argc, _TCHAR* argv[])
57.  {
58.      IGraphBuilder *pGraph = NULL;
59.      IMediaControl *pControl = NULL;
60.      IMediaEvent *pEvent = NULL;
61.      //Get some param-----
62.      HRESULT hr1;
63.      IBasicVideo *pVideo=NULL;
64.      IBasicAudio *pAudio=NULL;
65.      IVideoWindow *pWindow=NULL;
66.      IMediaSeeking *pSeeking=NULL;
67.
68.
69.      // Init COM
70.      HRESULT hr = CoInitialize(NULL);
71.      if (FAILED(hr)){
72.          printf("Error - Can't init COM.");
73.          return -1;
74.      }
75.
76.      // Create FilterGraph
77.      hr=CoCreateInstance(CLSID_FilterGraph, NULL,CLSCTX_INPROC_SERVER, IID_IGraphBuilder, (void **)&pGraph);
78.      if (FAILED(hr)){
79.          printf("Error - Can't create Filter Graph.");
80.          return -1;
81.      }
82.      // Query Interface
83.      hr = pGraph->QueryInterface(IID_IMediaControl, (void **)&pControl);
84.      hr = pGraph->QueryInterface(IID_IMediaEvent, (void **)&pEvent);
85.      // RenderFile
86.      hr = pGraph->RenderFile(L"cuc_ieschool.mov", NULL);
87.      if (FAILED(hr)){
88.          printf("Error - Can't Render File.");
89.          return -1;
90.      }
91.      #if OUTPUT_INFO
92.      //Get some information-----
93.      long video_w=0,video_h=0,video_bitrate=0,audio_volume=0;
94.      long long duration_l=0,position_l=0;
95.      REFTIME avgttimeperframe=0;
96.      float framerate=0,duration_sec=0,progress=0,position_sec=0;
97.      //Video
98.      hr1=pGraph->QueryInterface(IID_IBasicVideo, (void **)&pVideo);
99.      pVideo->get_VideoWidth(&video_w);
100.      pVideo->get_VideoHeight(&video_h);
101.      pVideo->get_AvgTimePerFrame(&avgttimeperframe);
102.      framerate=1/avgttimeperframe;
103.      //pVideo->get_BitRate(&video_bitrate);

```

```

104. //Audio
105. hr1=pGraph->QueryInterface(IID_IBasicAudio, (void **)&pAudio);
106. //Mute
107. //pAudio->put_Volume(-10000);
108. printf("Some Information:\n");
109. printf("Video Resolution:\t%d\t%d\n", video_w, video_h);
110. printf("Video Framerate:\t%.3f\n", framerate);
111. //Window
112. hr1=pGraph->QueryInterface(IID_IVideoWindow, (void **)&pWindow);
113. pWindow->put_Caption(L"Simplest DirectShow Player");
114. //pWindow->put_Width(480);
115. //pWindow->put_Height(272);
116. //Seek
117. hr1=pGraph->QueryInterface(IID_IMediaSeeking, (void **)&pSeeking);
118. pSeeking->GetDuration(&duration_1);
119. //time unit:100ns=0.0000001s
120. duration_sec=(float)duration_1/10000000.0;
121. printf("Duration:\t%.2f s\n", duration_sec);
122. //pSeeking->SetPositions();
123. //PlayBack Rate
124. //pSeeking->SetRate(2.0);
125.
126. //Show Filter in FilterGpagh
127. show_filters_in_filtergraph(pGraph);
128. //-----
129. #endif
130.
131. printf("Progress Info\n");
132. printf("Position\tProgress\n");
133. if (SUCCEEDED(hr)){
134.     // Run
135.     hr = pControl->Run();
136.     if (SUCCEEDED(hr)){
137.         long evCode=0;
138.         //pEvent->WaitForCompletion(INFINITE, &evCode);
139.         while(evCode!=EC_COMPLETE){
140.             //Info
141. #if OUTPUT_INFO
142.             pSeeking->GetCurrentPosition(&position_1);
143.             position_sec=(float)position_1/10000000.0;
144.             progress=position_sec*100/duration_sec;
145.             printf("%.2fs\t%.5f%%\n", position_sec, progress);
146. #endif
147.             //1000ms
148.             pEvent->WaitForCompletion(1000, &evCode);
149.         }
150.     }
151. }
152. // Release resource
153. pControl->Release();
154. pEvent->Release();
155. pGraph->Release();
156. CoUninitialize();
157. return 0;
158. }

```

## 运行结果

程序运行后即可开始播放一个“cuc\_ieschool.mov”文件。程序运行时候的截图如下所示。由图可见运行的同时程序在控制台中打印出了两种信息：

- (1) 该视频的相关信息
- (2) 播放该视频的 Filter Graph中的Filter（该功能通过函数show\_filters\_in\_filtergraph()完成）。

□

可以通过定义在代码最前面宏OUTPUT\_INFO控制是否输出视频的信息。定义成“0”的话则不会输出视频的信息。如下所示。

```

1. #define OUTPUT_INFO 1

```

## 下载

### Simplest DirectShow Example

#### 项目主页

SourceForge：<https://sourceforge.net/projects/simplestdirectshowexample/>

Github：[https://github.com/leixiaohua1020/simplest\\_directshow\\_example](https://github.com/leixiaohua1020/simplest_directshow_example)

开源中国：[http://git.oschina.net/leixiaohua1020/simplest\\_directshow\\_example](http://git.oschina.net/leixiaohua1020/simplest_directshow_example)

CDSN下载地址：<http://download.csdn.net/detail/leixiaohua1020/8348163>

本程序包含了DirectShow开发的示例程序。适合DirectShow初学者进行学习。

它包含了以下几个子程序：

**simplest\_directshow\_player: 最简单的基于DirectShow的视频播放器。**

simplest\_directshow\_player\_custom: 最简单的基于DirectShow的视频播放器（Custom）。

playerGUI: 最简单的基于DirectShow的播放器-图形界面版。

simplest\_directshow\_info: 最简单的Directshow信息显示例子。

simplest\_directshow\_filter: 目前还未完成。

版权声明：本文为博主原创文章，未经博主允许不得转载。 <https://blog.csdn.net/leixiaohua1020/article/details/42372419>

文章标签：[DirectShow](#) [播放器](#) [开发](#)

个人分类：[我的开源项目](#) [DirectShow](#)

此PDF由spygg生成,请尊重原作者版权!!!

我的邮箱:liushidc@163.com