

最简单的基于DirectShow的示例：获取Filter信息

2015年01月12日 21:03:08 阅读数：13718

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](#)

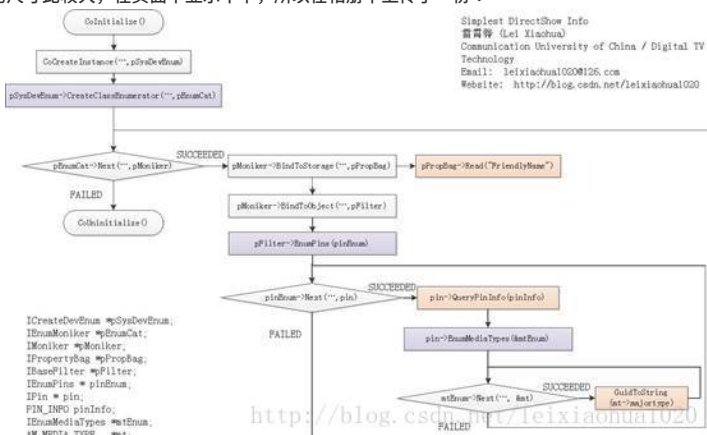
[最简单的基于DirectShow的示例：获取Filter信息](#)

本文记录一个获取DirectShow的Filter信息的程序。该程序可以遍历当前系统中的DirectShow Filter并且将它们的信息打印到控制台中。通过本程序可以搞清楚DirectShow中Filter, Pin, MediaType等几个关键结构之间的关系，以及它们的信息的获取方法。



流程图

该程序的流程图如下所示。由于该图的尺寸比较大，在页面中显示不下，所以在相册中上传了一份：



[查看大图](#)

接口

该流程图中涉及到以下接口：

ICreateDevEnum *pSysDevEnum：设备枚举接口。

IEnumMoniker *pEnumCat：Moniker（别名）枚举接口。

IMoniker *pMoniker：Moniker（别名）接口。

IPropertyBag *pPropBag：存储属性值的接口。

IBaseFilter *pFilter：Filter接口。

IEnumPins *pinEnum：Filter枚举接口。

IPin *pin：Pin接口。

PIN_INFO pinInfo：存储Pin的信息的结构体。

IEnumMediaTypes *mtEnum：MediaType枚举接口。

AM_MEDIA_TYPE *mt：描述媒体类型的结构体。

流程图

该流程图中涉及到以下函数：

【初始化】

CoInitialize()：初始化COM运行环境。

CoCreateInstance(...,pSysDevEnum)：用指定的类标识符创建一个Com对象。在该示例中类标识符为“IID_ICreateDevEnum”，用于创建ICreateDevEnum。

【Filter的枚举】

pSysDevEnum->CreateClassEnumerator(...,pEnumCat)：通过ICreateDevEnum查询IEnumMoniker枚举接口，枚举指定类型目录下的设备Moniker（别名）。

pEnumCat->Next(...,pMoniker)：通过IEnumMoniker查询下一个IMoniker接口。

pMoniker->BindToStorage(...,pPropBag)：通过IMoniker查询IPropertyBag接口（用于获取Filter信息）。

pPropBag->Read("FriendlyName")：通过IPropertyBag获取"FriendlyName"属性的值。

pMoniker->BindToObject(...,pFilter)：通过IMoniker查询IBaseFilter接口（用于获取Filter，注意和BindToStorage()区别）。

【Pin的枚举】

pFilter->EnumPins(pinEnum)：通过IBaseFilter查询IEnumPins枚举接口。

pinEnum->Next(...,pin)：通过IEnumPins查询下一个IPin接口。

pin->QueryPinInfo(PinInfo)：通过IPin获取Pin的信息。

【MediaType的枚举】

pin->EnumMediaTypes(&mtEnum)：通过IPin查询IEnumMediaTypes枚举接口。

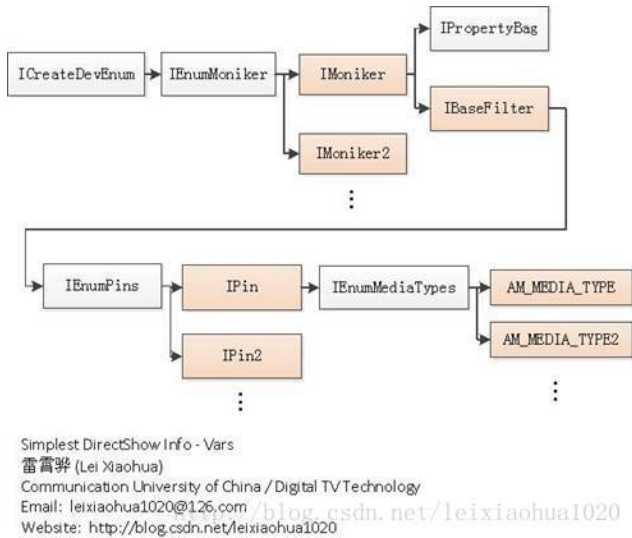
mtEnum->Next(..., &mt)：通过IEnumMediaTypes查询下一个AM_MEDIA_TYPE。

GuidToString(mt->majortype)：把AM_MEDIA_TYPE的GUID转换成字符串（方便输出）。

【释放】

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

再附上一张代码中涉及到的接口之间的关系：



可以看出从上到下他们之间顺序的排列如下所示：

ICreateDevEnum-->IEnumMoniker-->IMoniker-->IBaseFilter-->IEnumPins-->IPin-->IEnumMediaTypes-->AM_MEDIA_TYPE

源代码

```
[cpp]
1.  /**
2.  * 最简单的Directshow信息显示例子
3.  * Simplest DirectShow Info
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. * 本程序是一段获取DirectShow滤镜信息的代码。通过本代码可以获得
12. * DirectShow滤镜信息。适合初学者学习DirectShow。
```

```

13.  *
14.  * This code can be used to get Directshow Filter's information.
15.  * Suitable for the beginner of DirectShow.
16.  */
17.
18.  #include "stdafx.h"
19.  #include <dshow.h>
20.  #include <atlconv.h>
21.
22.  #define OUTPUT_PIN      1
23.  #define OUTPUT_MEDIATYPE 1
24.
25.
26.  char* GuidToString(const GUID &guid)
27.  {
28.      int buf_len=64;
29.      char *buf =(char *)malloc(buf_len);
30.      _snprintf(
31.          buf,
32.          buf_len,
33.          "{%08X-%04X-%04X-%02X%02X-%02X%02X%02X%02X%02X}",
34.          guid.Data1, guid.Data2, guid.Data3,
35.          guid.Data4[0], guid.Data4[1],
36.          guid.Data4[2], guid.Data4[3],
37.          guid.Data4[4], guid.Data4[5],
38.          guid.Data4[6], guid.Data4[7]);
39.      //printf("%s\n",buf);
40.      return buf;
41.  }
42.
43.
44.  int _tmain(int argc, _TCHAR* argv[])
45.  {
46.      USES_CONVERSION;
47.      // Init COM
48.      HRESULT hr=NULL;
49.      hr= CoInitialize(NULL);
50.      if (FAILED(hr)){
51.          printf("Error, Can not init COM.");
52.          return -1;
53.      }
54.      printf("=====Directshow Filters =====\n");
55.      ICreateDevEnum *pSysDevEnum = NULL;
56.      hr = CoCreateInstance(CLSID_SystemDeviceEnum, NULL, CLSCTX_INPROC_SERVER,
57.          IID_ICreateDevEnum, (void **)&pSysDevEnum);
58.      if (FAILED(hr)){
59.          return hr;
60.      }
61.
62.      IEnumMoniker *pEnumCat = NULL;
63.      //Category
64.      /*****
65.      Friendly Name          CLSID
66.      -----
67.      Audio Capture Sources   CLSID_AudioInputDeviceCategory
68.      Audio Compressors        CLSID_AudioCompressorCategory
69.      Audio Renderers          CLSID_AudioRendererCategory
70.      Device Control Filters    CLSID_DeviceControlCategory
71.      DirectShow Filters        CLSID_LegacyAmFilterCategory
72.      External Renderers        CLSID_TransmitCategory
73.      Midi Renderers            CLSID_MidiRendererCategory
74.      Video Capture Sources     CLSID_VideoInputDeviceCategory
75.      Video Compressors         CLSID_VideoCompressorCategory
76.      WDM Stream Decompression Devices CLSID_DVDHWDecodersCategory
77.      WDM Streaming Capture Devices AM_KSCATEGORY_CAPTURE
78.      WDM Streaming Crossbar Devices AM_KSCATEGORY_CROSSBAR
79.      WDM Streaming Rendering Devices AM_KSCATEGORY_RENDER
80.      WDM Streaming Tee/Splitter Devices AM_KSCATEGORY_SPLITTER
81.      WDM Streaming TV Audio Devices AM_KSCATEGORY_TVAUDIO
82.      WDM Streaming TV Tuner Devices AM_KSCATEGORY_TVTUNER
83.      WDM Streaming VBI Codecs AM_KSCATEGORY_VBICODEC
84.      *****/
85.      hr = pSysDevEnum->CreateClassEnumerator(CLSID_VideoCompressorCategory, &pEnumCat, 0);
86.      //hr = pSysDevEnum->CreateClassEnumerator(CLSID_VideoInputDeviceCategory, &pEnumCat, 0);
87.      //hr = pSysDevEnum->CreateClassEnumerator(CLSID_AudioCompressorCategory, &pEnumCat, 0);
88.      //hr = pSysDevEnum->CreateClassEnumerator(CLSID_AudioInputDeviceCategory, &pEnumCat, 0);
89.      //hr = pSysDevEnum->CreateClassEnumerator(CLSID_MediaMultiplexerCategory, &pEnumCat, 0);
90.      //hr = pSysDevEnum->CreateClassEnumerator(CLSID_LegacyAmFilterCategory, &pEnumCat, 0);
91.
92.      if (hr != S_OK) {
93.          pSysDevEnum->Release();
94.          return -1;
95.      }
96.
97.      IMoniker *pMoniker = NULL;
98.      ULONG monikerFetched;
99.      //Filter
100.      while(pEnumCat->Next(1, &pMoniker, &monikerFetched) == S_OK){
101.          IPropertyBag *pPropBag;
102.          VARIANT varName;
103.          IBaseFilter *pFilter;

```

```

104.     hr = pMoniker->BindToObject(0, 0, IID_IPropertyBag, (void**) &pPropBag);
105.     if (FAILED(hr)){
106.         pMoniker->Release();
107.         continue;
108.     }
109.     VariantInit(&varName);
110.     hr = pPropBag->Read(L"FriendlyName", &varName, 0);
111.     //"FriendlyName": The name of the device.
112.     //"Description": A description of the device.
113.     //"Filter Info"=====
114.     printf("[%s]\n", W2A(varName.bstrVal));
115.     VariantClear(&varName);
116.     //=====
117. #if OUTPUT_PIN
118.     hr = pMoniker->BindToObject(NULL, NULL, IID_IBaseFilter, (void**) &pFilter);
119.     if (!pFilter){
120.         continue;
121.     }
122.
123.     IEnumPins * pinEnum = NULL;
124.     IPin * pin = NULL;
125.     ULONG pinFetched = 0;
126.     if (FAILED(pFilter->EnumPins(&pinEnum))){
127.         pinEnum->Release();
128.         continue;
129.     }
130.     pinEnum->Reset();
131.     //Pin Info
132.     while (SUCCEEDED(pinEnum->Next(1, &pin, &pinFetched)) && pinFetched){
133.         if (!pin){
134.             continue;
135.         }
136.         PIN_INFO pinInfo;
137.         if (FAILED(pin->QueryPinInfo(&pinInfo))){
138.             continue;
139.         }
140.         printf("\t[Pin] ");
141.         switch(pinInfo.dir){
142.             case PINDIR_INPUT: printf("Dir:Input \t"); break;
143.             case PINDIR_OUTPUT: printf("Dir:Output \t"); break;
144.             default: printf("Dir:Unknown\n"); break;
145.         }
146.         printf("Name:%s\n", W2A(pinInfo.achName));
147.
148.         //MediaType
149. #if OUTPUT_MEDIATYPE
150.         IEnumMediaTypes * mtEnum=NULL;
151.         AM_MEDIA_TYPE * mt=NULL;
152.         if( FAILED( pin->EnumMediaTypes( &mtEnum ) ) )
153.             break;
154.         mtEnum->Reset();
155.
156.         ULONG mtFetched = 0;
157.
158.         while (SUCCEEDED(mtEnum->Next(1, &mt, &mtFetched)) && mtFetched){
159.
160.             printf("\t\t[MediaType]\n");
161.             //Video
162.             char *MEDIATYPE_Video_str=GuidToString(MEDIATYPE_Video);
163.             //Audio
164.             char *MEDIATYPE_Audio_str=GuidToString(MEDIATYPE_Audio);
165.             //Stream
166.             char *MEDIATYPE_Stream_str=GuidToString(MEDIATYPE_Stream);
167.             //Majortype
168.             char *majortype_str=GuidToString(mt->majortype);
169.             //Subtype
170.             char *subtype_str=GuidToString(mt->subtype);
171.
172.             printf("\t\t Majortype:");
173.             if(strcmp(majortype_str, MEDIATYPE_Video_str)==0){
174.                 printf("Video\n");
175.             }else if(strcmp(majortype_str, MEDIATYPE_Audio_str)==0){
176.                 printf("Audio\n");
177.             }else if(strcmp(majortype_str, MEDIATYPE_Stream_str)==0){
178.                 printf("Stream\n");
179.             }else{
180.                 printf("Other\n");
181.             }
182.             printf("\t\t Subtype GUID:%s", subtype_str);
183.
184.             free(MEDIATYPE_Video_str);
185.             free(MEDIATYPE_Audio_str);
186.             free(MEDIATYPE_Stream_str);
187.             free(subtype_str);
188.             free(majortype_str);
189.             printf("\n");
190.         }
191.     }
192. #endif
193.     pin->Release();
194.
195.     }

```

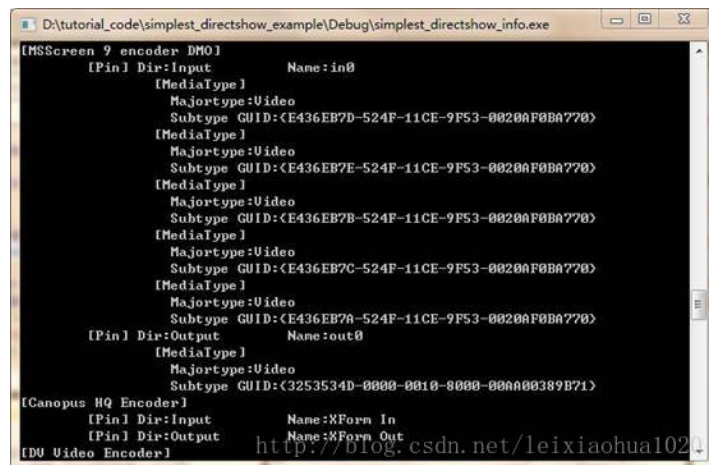
```

196.         pinEnum->Release();
197.
198.         pFilter->Release();
199.     #endif
200.
201.         pPropBag->Release();
202.         pMoniker->Release();
203.     }
204.     pEnumCat->Release();
205.     pSysDevEnum->Release();
206.     printf("=====\\n");
207.     CoUninitialize();
208.     return 0;
209. }

```

运行结果

程序运行的结果如下图所示。从图中可以看出，程序打印出了系统中DirectShow的Filter信息。每个Filter的信息中包含了它的Pin的信息。每个Pin中又包含了Pin中的Media Type信息。



可以通过定义在代码最前面宏 控制输出的Filter信息的类型。定义成"0"的话则不会输出该类信息。如下所示。

```

1. #define OUTPUT_PIN 1
2. #define OUTPUT_MEDIATYPE 1

```

下载

Simplest DirectShow Example

项目主页

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

Github : https://github.com/leixiaohua1020/simplest_directshow_example

开源中国 : 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。

文章标签：[DirectShow](#) [Filter](#) [Pin](#) [Mediatype](#)

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