

原 传媒大学媒体中心资源批量获取工具的制作

2013年10月25日 17:00:49 阅读数：3853

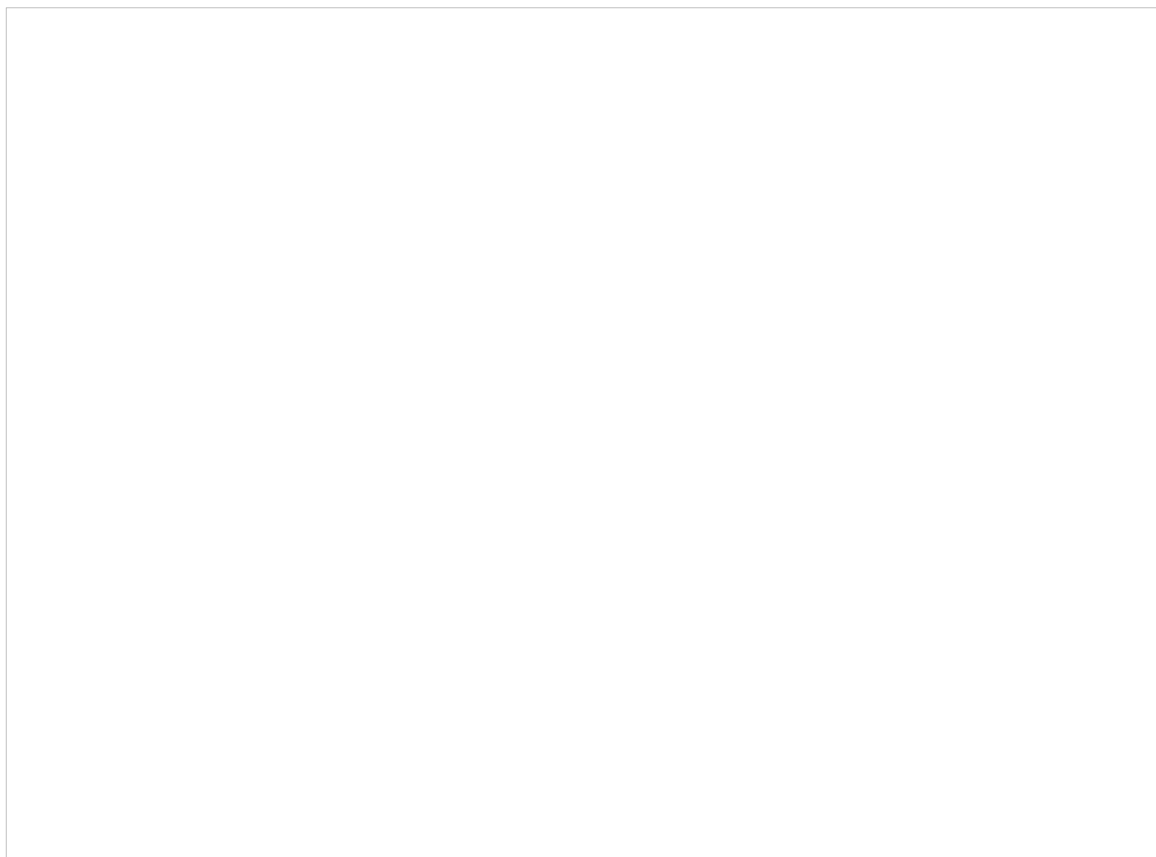
中国传媒大学媒体中心（<http://media.cuc.edu.cn/>）是中国传媒大学媒体资源最集中的地方，各种电影，电视剧，音乐等等，可以说是丰富多彩。然而它有一个缺点，就是只能在线看，不能下载。这导致想把自己喜爱的视频保存下来是比较困难的。为此我课余时间进行了一个小研究，做了一个MFC的小程序，可以实现媒体中心中资源URL的提取和保存，在此记录一下自己的制作过程。

该工具主要涉及以下三个技术：

- 1.发送HTTP请求，获取网页的源代码
- 2.查找具有特定标记的字符串，并提取出来
- 3.数据写入xml文件

下面先看看实际情况

首先登录媒体中心，打开一个视频，截图如图所示：



查看一下网页的源代码，看看有没有视频URL。在网页里搜索了关键字“.flv”，竟然找到了。

地址就是：

<http://202.205.20.12:2048/02/video/2012/5/111309/1336492554022.flv>

只要把上述地址粘贴到迅雷，快车里面就可以下载视频了。

现在有一个问题，就是下载一个视频可以这样翻网页源代码找一找，但是每次这样操作有点太过麻烦了。因此需要编程实现一个小工具。当输入一个视频播放网页的地址的时候，就可以输出该视频实际的下载地址。当然，最好可以批量输入页面地址，然后批量解析视频的实际地址。

其实这个东西已经实现出来了，最终界面如下所示：

在这个工具中，贴入视频所在的网页，就可以解析出视频的标题以及视频的地址。而且下半部分还提供了批量解析的功能，输入视频ID（随后会解释）的范围，就可以探测出该范围内所有的视频资源，同时输出成XML或者TXT。

批量解析输出成XML如下所示：

```
[html]
1. <URLList>
2.   <URL id="4">
3.     <name>恐惧拉斯维加斯</name>
4.     <link>http://202.205.20.12:2048/02/video/2008/6/恐惧拉斯维加斯/恐惧拉斯维加斯.flv</link>
5.     <type>vod</type>
6.     <protocol>http</protocol>
7.   </URL>
8.   <URL id="5">
9.     <name>一球成名2CD1</name>
10.    <link>http://202.205.20.12:2048/02/video/2008/7/一球成名2CD1/一球成名2CD1.flv</link>
11.    <type>vod</type>
12.    <protocol>http</protocol>
13.  </URL>
14.  <URL id="6">
15.    <name>一球成名2CD2</name>
16.    <link>http://202.205.20.12:2048/02/video/2008/7/一球成名2CD2/一球成名2CD2.flv</link>
17.    <type>vod</type>
18.    <protocol>http</protocol>
19.  </URL>
20.  <URL id="7">
21.    <name>深海寻人</name>
22.    <link>http://202.205.20.12:2048/02/video/2008/7/深海寻人/深海寻人.flv</link>
23.    <type>vod</type>
24.    <protocol>http</protocol>
25.  </URL>
26.  <URL id="28">
27.    <name>0h_My_Friend</name>
28.    <link>http://202.205.20.12:2048/02/video/1970/1/28.BIGBANG_3rd_MINI_0h_My_Friend_MV/BIGBANG_3rd_MINI_0h_My_Friend_MV.flv</link>
29.  </URL>
30.  <URL id="30">
31.    <name>once in a lifetime</name>
32.    <link>http://202.205.20.12:2048/02/video/2008/9/30.once_in_a_lifetime/once_in_a_lifetime.flv</link>
33.    <type>vod</type>
34.    <protocol>http</protocol>
35.  </URL>
36. </URLList>
```

或者TXT格式：

1.	视频ID:4	视频名称：恐惧拉斯维加斯	视频地址：http://202.205.20.12:2048/02/video/2008/6/恐惧拉斯维加斯/恐惧拉斯维加斯.flv
2.	视频ID:5	视频名称：一球成名2CD1	视频地址：http://202.205.20.12:2048/02/video/2008/7/一球成名2CD1/一球成名2CD1.flv
3.	视频ID:6	视频名称：一球成名2CD2	视频地址：http://202.205.20.12:2048/02/video/2008/7/一球成名2CD2/一球成名2CD2.flv
4.	视频ID:7	视频名称：深海寻人	视频地址：http://202.205.20.12:2048/02/video/2008/7/深海寻人/深海寻人.flv
5.	视频ID:28	视频名称：0h_My_Friend	视频地址： http://202.205.20.12:2048/02/video/1970/1/28.BIGBANG_3rd_MINI_0h_My_Friend_MV/BIGBANG_3rd_MINI_0h_My_Friend_MV.flv
6.	视频ID:30	视频名称：once in a lifetime	视频地址： http://202.205.20.12:2048/02/video/2008/9/30.once_in_a_lifetime/once_in_a_lifetime.flv

介绍完毕。现在简要介绍一下单个视频解析url的制作过程。

第一步：发送HTTP请求，获取网页的源代码

曾经写过一篇发送HTTP请求获取网页源代码的文章：[C++发送HTTP请求获取网页HTML代码](#)

第二步：查找具有特定标记的字符串，并提取出来

曾经写过一篇查找字符串并提取出来的方法的文章：[C++从文件中查找特定的字符串，并提取该字符串](#)

在此需要综合前两篇文章的方法，实现对特定url的网页源代码的请求，以及对特定字符串的查找和提取。

首先观察一下网页源代码，发现视频地址是在一对<script>标签里：

```
[html]
1. <script type='text/javascript'>
2.   var s1 = new SWFObject('/v/Y29uZmlnP59tcGMuanNwP2lpZD0xMTEzMDEk=', 'mediaplayer', '480', '380', '10');
3.   //var s1 = new SWFObject('/scripts/player_4.4.swf?config=mpc.jsp?mid=111309', 'mediaplayer', '480', '380', '9');
4.   //s1.addParam('flashvars', '&file=http://202.205.20.12:2048/02/video/2012/5/111309/1336492554022.flv&streamer=lig
   httpd');
5.   s1.useExpressInstall('/scripts/expressinstall.swf');
6.   s1.addParam('allowfullscreen', 'true');
7.   s1.addParam('allowscripaccess', 'always');
8.   s1.write('player_box');
9. </script>
```

而地址开头"http://"前面是"('flashvars','&file=", 地址结尾".flv"后面是"&streamer=lighttpd"。以这两个字符串作为标志, 就能找到视频url地址。

视频标题的开头前面是"", 地址结尾" "。以这两个字符串作为标志, 就能找到视频的标题。

下面贴上这部分的源代码

注意: 本工程中使用了3个CString变量关联到3个Edit Control控件:

CString m_htmlurl;//输入页面url

CString m_videourl;//输出解析出来的视频url

CString m_videoname;//输出解析出来的视频名称

```
[cpp]
1. void Csocket_http_dialogDlg::OnBnClickedOk()
2. {
3.     // TODO: 在此添加控件通知处理程序代码
4.     //地址-----
5.     char stringsearch_before[]="('flashvars','&file=";
6.     char stringsearch_after[]="&streamer=lighttpd";
7.     //标题-----
8.     char stringsearch_before1[]="<span class=text_bl>";
9.     char stringsearch_after1[]="</span> ";
10.    //url_search_before位置, 代表找到了相应字符串
11.    const char *mark=NULL;
12.    //开始和结束
13.    const char *stringstart=NULL;
14.    const char *stringend=NULL;
15.    //结果
16.    char url[200]={0};
17.    char vname[200]={0};
18.
19.    char *content_temp=NULL;
20.    char *string_temp=NULL;
21.    //-----
22.
23.
24.
25.    CInternetSession session;//建立对话
26.    CHttpFile *file;
27.    //CException *e;
28.    UpdateData(true);
29.    CString URL = m_htmlurl.GetString();
30.    if(URL==""){
31.        AfxMessageBox("网页地址为空!");
32.    }
33.    try{
34.        file=(CHttpFile*)session.OpenURL(URL);//打开文件
35.    }catch(...){
36.        file = 0;
37.    }
38.    if (file){
39.        DWORD dwStatus;
40.        file->QueryInfoStatusCode(dwStatus);
41.        if(dwStatus == HTTP_STATUS_OK){
42.            CString content;
43.            CString data;
44.            while (file->ReadString(data)){
45.                content+=data+"\r\n";
46.            }
47.            content.TrimRight();
48.            //MessageBox((LPCTSTR)content);
49.            //处理数据,数据位于content之中-----
50.            //获得地址
51.            mark=strstr(content,stringsearch_before);
52.            if(mark==NULL){
53.                AfxMessageBox("没有找到地址...本软件只适用于媒体中心");
54.                goto end;
55.            }
56.            //注意要-1, 此处获得string开始
57.            stringstart=mark+sizeof(stringsearch_before)-1;
58.            //注意此处获得string结束之后的1位, 因此最后一位应改为\0
59.            stringend=strstr(stringstart,stringsearch_after);
60.            string_temp=url;
61.            for(content_temp=(char*)stringstart;content_temp!=stringend;content_temp++,string_temp++){
62.                *string_temp=*content_temp;
63.            }
64.            string_temp='\0';
65.            //获得标题-----
66.            mark=strstr(content,stringsearch_before1);
67.            if(mark==NULL){
68.                AfxMessageBox("没有找到标题...本软件只适用于媒体中心");
69.                goto end;
70.            }
71.            //注意要-1, 此处获得string开始
```

```
72.         stringstart=mark+sizeof(stringsearch_before1)-1;
73.         //注意此处获得string结束之后的1位，因此最后一位应改为\0
74.         stringend=strstr(stringstart,stringsearch_after1);
75.         //vname是最终输出
76.         string_temp=vname;
77.         for(content_temp=(char*)stringstart;content_temp!=stringend;content_temp++,string_temp++){
78.             *string_temp=*content_temp;
79.         }
80.         string_temp='\0';
81.
82.         //-----
83.         m_videourl=url;
84.         m_videoname=vname;
85.         UpdateData(FALSE);
86.
87.     }else{
88.         MessageBox("dwStatus!=HTTP_STATUS_OK");
89.     }
90.     end:
91.     file->Close();
92.     delete file;
93. }
94. session.Close();
95.
96.
97. }
```

第三步：数据写入xml文件

曾经写过一篇数据写入成xml的文章：[TinyXML：一个优秀的C++ XML解析器](#)

在这里就不多说了，方法类似。

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/12974945>

文章标签：[http](#) [xml](#) [字符串](#) [网页源代码](#)

个人分类：[计算机网络](#)

此PDF由spygg生成,请尊重原作者版权!!!

我的邮箱:liushidc@163.com