

# jmxiaocai的专栏

彩虹天堂

 RSS 订阅

#### 个人资料



imxiaocai

访问: 59190次

积分: 815

等级: BLOC 3

排名: 千里之外

原创: 15篇 转载: 40篇

译文: 0篇

评论: 12条

#### 文章搜索

## 文章分类

vc/c++ (37)

oracle (5)

php (5) asp.net (1)

c# (10)

mysql (2)

sqlite (4)

Access (2) mssql (0)

## 文章存档

2012年12月 (1)

2012年09月 (1)

2012年06月 (2)

2012年05月 (3)

2012年04月 (2)

#### 阅读排行

c# Post數据到web服务器 (5243)
VC 下Microsoft Speech (4781)
tinyxml 解析XML字符串, (4273)
c# WebClient 下载远程图 (3413)
Oracle 查询当天、当周、(2520)

oracle 循环更新表的数据 (2021)

【免费公开课】模板匹配的字符识别(OCR)算法原理 有奖试读—漫话程序员面试求职、升职加薪、创业与生活

## VC 下Microsoft Speech SDK开发语音识别

标签: microsoft preprocessor null xml 引擎 command

2011-12-02 18:18 4783人阅读 评论(0) 收藏 举报

**Ⅲ** 分类: vc/c++(36) -

### 1.首先开发得需要Microsoft Speech SDK的支持,以下是下载地址

http://msdn.microsoft.com/code/sample.asp?url=/msdn-files/027/000/781/msdncompositedoc.xml

Speech SDK 5.1 (68 MB)

http://download.microsoft.com/download/speechSDK/SDK/5.1/WXP/EN-US/speechsdk51.exe

5.1 Language Pack (81.5 MB)

http://download.microsoft.com/download/speechSDK/SDK/5.1/WXP/EN-US/speechsdk51LangPack.exe

Redistributables (128 MB)

http://download.microsoft.com/download/speechSDK/SDK/5.1/WXP/EN-US/speechsdk51MSM.exe

Documentation (2.28 MB)

http://download.microsoft.com/download/speechSDK/SDK/5.1/WXP/EN-US/sapi.chm

## 2.下载后,执行安装

下载完毕后首先安装SpeechSDK51.exe,然后安装中文语言补丁包SpeechSDK51LangPack,然后展开speechsdk51MSM.exe,这些都是自解压文件,解压后执行相应的setup程序到你要的目录,默认C:/Microsoft Speech SDK 5.1.对应的开发参考手册为sapi.chm,详细描述了各个函数的细节等.

### 3.VC的环境配置

在应用SDK的开发前当然得需要对工程环境进行配置,我用的是VS2003(其他情况类似),配置的过程如下:工具->选项->项目->VC++目录,在"显示以下内容的目录"下拉框中选择"包含目录"项,添加一项C:/Program Files/Microsoft Speech SDK 5.1/Include到目录中去。再选择"库文件"项,添加一项C:/Program Files/Microsoft Speech SDK 5.1/Lib/i386到目录中去。

#### 4.其他准备项

基础的配置已经完成,那么接下来的工作就是要包含编译的头文件了,所以先将头文件和库文件包含进来

#include <sapi.h>

#include <sphelper.h>

#include <spuihelp.h>

#pragma comment(lib,"ole32.lib") //Colnitialize CoCreateInstance需要调用ole32.dll

#pragma comment(lib,"sapi.lib") //sapi.lib在SDK的lib目录,必需正确配置

具体其他函数所需要的头文件可参考sapi.chm手册.

#### 5.源文件修改项

看上去上面的部分配置完成后就大功告成了,其实还不全是,当你编译时就会出错:

c:/program files/microsoft speech sdk 5.1/include/sphelper.h(769) : error C4430: missing type specifier - int assumed. Note: C++ does not support default-int

c:/program files/microsoft speech sdk 5.1/include/sphelper.h(1419) : error C4430: missing type specifier - int assumed. Note: C++ does not support default-int

c:/program files/microsoft speech sdk 5.1/include/sphelper.h(2373) : error C2065: 'psz' : undeclared identifier c:/program files/microsoft speech sdk 5.1/include/sphelper.h(2559) : error C2440: 'initializing' : cannot convert from 'CSpDynamicString' to 'SPPHONEID \*'

No user-defined-conversion operator available that can perform this conversion, or the operator cannot be called

```
VC++中计时器timer的运 (2012)
                            c:/program files/microsoft speech sdk 5.1/include/sphelper.h(2633) : error C2664: 'wcslen' : cannot convert
C# 按指定字符截取字符》(1228)
                             parameter 1 from 'SPPHONEID *' to 'const wchar t *'
vc++ 下载远程图片
                  (1198)
                             Types pointed to are unrelated; conversion requires reinterpret_cast, C-style cast or function-style cast
Oracle中执行sql语句时排 (1163)
                             Speech代码编写时间太早,语法不严密。而VS2003(及以上)对于语法检查非常严格,导致编译无法通过。修改
                             头文件中的以下行即可正常编译:
评论排行
                            Line 769
                              修改前: const ulLenVendorPreferred = wcslen(pszVendorPreferred);
c# WebClient 下载远程图
                     (5)
                              修改后: const unsigned long ulLenVendorPreferred = wcslen(pszVendorPreferred);
tinyxml 解析XML字符串,
                     (3)
                            Line 1418
消息函数SendMessagef
                     (1)
                              修改前: static CoMemCopyWFEX(const WAVEFORMATEX * pSrc, WAVEFORMATEX ** ppCoMemWFEX)
vc 屏蔽esc键和回车键
                     (1)
                              修改后: static HRESULT CoMemCopyWFEX(const WAVEFORMATEX * pSrc, WAVEFORMATEX **
c# Post数据到web服务器
                     (1)
                            ppCoMemWFEX)
c++/c 结构体定义 typede
                     (1)
                            Line 2372
装IIS的时候遇到的访问 I
                     (0)
                              修改前: for (const WCHAR * psz = (const WCHAR *)IParam; *psz; psz++) {}
VC++ 四步实现最小化托
                     (0)
                              修改后: const WCHAR * psz; for (psz = (const WCHAR *)|Param; *psz; psz++) {}
oracle 循环更新表的数据
                     (0)
                            Line 2559
char tchar wchar_t WCH
                     (0)
                              修改前: SPPHONEID* pphoneld = dsPhoneld;
                              修改后: SPPHONEID* pphoneld = (SPPHONEID*)((WCHAR *)dsPhoneld);
推荐文章
                            Line 2633
*4月28--30日: 一起去北展揭开
电信网络诈骗的神秘面纱
                               修改前: pphoneId += wcslen(pphoneId) + 1;
*Spring和MyBatis实现数据的读
                               修改后: pphoneId += wcslen((const wchar_t *)pphoneId) + 1;
                             好了, 编译通过, 下面可以正式编写程序了。
* Android几种常见的多渠道(批
量)打包方式介绍
                             6: 语音朗读: SAPI实现TTS(Text to Speech)
*Untiy Native Render Plugin在
VR中的绘制(二): 透明排序
                                • 1. 首先要初始化语音接口,一般有两种方式:
*随机过程--Metropolis-Hastings
                                   ISpVoice* pVoice;
                                   ::Colnitialize(NULL);
* Android源码浅析(一)
   -VMware Workstation Pro和
                                   HRESULT hr =CoCreateInstance(CLSID SpVoice, NULL, CLSCTX ALL,IID ISpVoice, (void **)
Ubuntu Kvlin 16.04 LTS安装配置
                                   然后就可以使用这个指针调用SAPI函数了,例如
最新评论
                                   pVoice->SetVolume(50);//设置音量
c++/c 结构体定义 typedef struct ,
                                   pVoice->Speak(str.AllocSysString(),SPF_ASYNC,NULL);
skyeagles123: 过了一遍博主的
总结,明朗很多了,以前常常搞混
                              另外也可以使用如下方式:
c# Post数据到web服务器
新亿: 这个方法 貌似对创建
                              CComPtr<ISpVoice> m_cpVoice;
cookie 不起作用了。
                              HRESULT hr = m_cpVoice.CoCreateInstance(CLSID_SpVoice );
tinyxml 解析XML字符串,可加载迈
kenshin007: m_http是个什么对
                              在下面的例子中都用这个m_cpVoice变量。CLSID_SpVoice的定义位于sapi.h中。
象啊???求解释
vc 屏蔽esc键和回车键

    2. 获取/设置输出频率。

将革命讲行到底: 很好!
tinyxml 解析XML字符串,可加载访
                             SAPI朗读文字的时候,可以采用多种频率方式输出声音,比如: 8kHz 8Bit Mono、8kHz 8BitStereo、44kHz
open520yin: m_http 你这个是什
                             16BitStereo等,在音调上有所差别。具体可以参考sapi.h。
c# WebClient 下载远程图片--超简
                              可以使用如下代码获取当前的频率配置:
jmxiaocai: 挺好的东西
                              CComPtr<ISpStreamFormat> cpStream:
c# WebClient 下载远程图片--紹作
                              HRESULT hrOutputStream =m_cpVoice->GetOutputStream(&cpStream);
jmxiaocai: @Usuda:大幅度的
                              if (hrOutputStream ==S_OK)
c# WebClient 下载远程图片--超简
imxiaocai: 稳稳当当
                              {
c# WebClient 下载远程图片--超简
                                CSpStreamFormat Fmt;
imxiaocai: @Usuda:大文件就不
用这个,小小的文件还是可以用
                                hr = Fmt.AssignFormat(cpStream);
                                if (SUCCEEDED(hr))
```

c# WebClient 下载远程图片--紹作

Usuda: 这个东西会出现操作超时

的情况

{

SPSTREAMFORMAT 是一个ENUM类型,定义位于sapi.h中,这样eFmt就保存了获得的当前频率设置值。每 一个值对应了不同的频率设置。

SPSTREAMFORMAT eFmt = Fmt.ComputeFormatEnum();

```
通过如下代码设置当前朗读频率:
 CComPtr<ISpAudio> m cpOutAudio; //声音输出接口
 SpCreateDefaultObjectFromCategoryId( SPCAT_AUDIOOUT,&m_cpOutAudio ); //创建接口
 SPSTREAMFORMAT eFmt = SPSF_8kHz8BitMono; //SPSF_8kHz 8Bit Mono这个参数可以参考sapi.chm手册
 CSpStreamFormat Fmt;
 Fmt.AssignFormat(eFmt);
 if (m_cpOutAudio)
   hr = m cpOutAudio->SetFormat(Fmt.FormatId(), Fmt.WaveFormatExPtr() );
 }
 else hr = E FAIL;
 if(SUCCEEDED( hr ) )
   m_cpVoice->SetOutput( m_cpOutAudio, FALSE );
   • 3. 获取/设置播放所用语音。
 引擎中所用的语音数据文件一般保存在SpeechEngines下的spd或者vce文件中。安装sdk后,在注册表中保存
了可用的语音,比如英文的男/女,简体中文的男音等。位置是:
 HKEY_LOCAL_MACHINE/Software/Microsoft/Speech/Voices/Tokens
 SAPI的缺点是不能支持中英文混读,在朗读中文的时候,遇到英文,只能逐个字母读出。所以需要程序自己进
行语音切换。
(1) 可以采用如下的函数把当前SDK支持的语音填充在一个组合框中:
 // SAPI5helper function in sphelper.h
  CWnd* m_wnd = GetDlgItem(IDC_COMBO_VOICES);
  HWND hWndCombo = m_wnd->m_hWnd; //组合框句柄
  HRESULT hr =SpInitTokenComboBox( hWndCombo , SPCAT_VOICES );
  这个函数是通过IEnumSpObjectTokens接口枚举当前可用的语音接口,把接口的说明文字添加到组合框中,
并且把接口的指针作为LPARAM 保存在组合框中。
  一定要记住最后程序退出的时候,释放组合框中保存的接口:
 SpDestroyTokenComboBox(\ hWndCombo\ );
  这个函数的原理就是逐个取得combo里面每一项的LPARAM数据,转换成lUnknown接口指针,然后调用
Release函数。
(2) 当组合框选择变化的时候,可以用下面的函数获取用户选择的语音:
  ISpObjectToken* pToken = SpGetCurSelComboBoxToken( hWndCombo );
(3) 用下面的函数获取当前正在使用的语音:
  CComPtr<ISpObjectToken> pOldToken;
  HRESULT hr = m_cpVoice->GetVoice( &pOldToken);
(4) 当用户选择的语音和当前正在使用的不一致的时候,用下面的函数修改:
 if(pOldToken != pToken)
   // 首先结束当前的朗读,这个不是必须的。
   HRESULT hr = m_cpVoice->Speak( NULL,SPF_PURGEBEFORESPEAK, 0);
   if (SUCCEEDED (hr))
     hr = m_cpVoice->SetVoice( pToken );
(5) 也可以直接使用函数SpGetTokenFromId获取指定voice的Token指针,例如:
  WCHAR pszTokenId[]
=L"HKEY LOCAL MACHINE//Software//Microsoft//Speech//Voices//Tokens//MSSimplifiedChineseVoice";
  SpGetTokenFromId(pszTokenID, &pChineseToken);
   • 4 开始/暂停/恢复/结束当前的朗读
要朗读的文字必须位于宽字符串中,所以从文本框中读取的字符串类型CString必须转换成为WCHAR型,如下
(m_strText为文本框变量):
  CString strSpeak;
 m\_strText.GetWindowText(strSpeak);
```

```
WCHAR wChar[256];
 memset(wChar ,0,256);
 MultiByteToWideChar( CP_ACP , 0 , strSpeak , strSpeak.GetLength() , wChar , 256);
  这样就将文本框中的字符串strSpeak转化为WCHAR型的wChar变量中了.
   开始朗读的代码:
  hr =m_cpVoice->Speak( wChar, SPF_ASYNC |SPF_IS_NOT_XML, 0 );
  如果要解读一个XML文本,用:
  hr =m_cpVoice->Speak( wChar, SPF_ASYNC |SPF_IS_XML, 0 );
  暂停的代码: m_cpVoice->Pause();
  恢复的代码: m_cpVoice->Resume();
  结束的代码: (上面的例子中已经给出了)
  hr =m_cpVoice->Speak( NULL, SPF_PURGEBEFORESPEAK,0);
      • 5 跳过部分朗读的文字
  在朗读的过程中,可以跳过部分文字继续后面的朗读,代码如下:
  ULONG ulGarbage = 0;
  WCHAR szGarbage[] =L"Sentence";
  hr =m_cpVoice->Skip( szGarbage, SkipNum,&ulGarbage );
   SkipNum是设置要跳过的句子数量,值可以是正/负。
   根据sdk的说明,目前SAPI仅仅支持SENTENCE这个类型。SAPI是通过标点符号来区分句子的。
      • 6播放WAV文件。SAPI可以播放WAV文件,这是通过ISpStream接口实现的:
 CComPtr<ISpStream> cpWavStream;
  WCHAR szwWavFileName[NORM_SIZE] = L"";
  USES_CONVERSION;
  wcscpy( szwWavFileName, T2W(szAFileName ) );//从ANSI将WAV文件的名字转换成宽字符串
  //使用sphelper.h 提供的这个函数打开wav 文件,并得到一个 IStream 指针
  hr = SPBindToFile(szwWavFileName, SPFM_OPEN_READONLY, &cpWavStream);
  if( SUCCEEDED( hr ) )
  {
       m_cpVoice->SpeakStream( cpWavStream, SPF_ASYNC, NULL);//播放WAV文件
  }
      • 7 将朗读的结果保存到wav文件
             TCHARszFileName[256];//假设这里面保存着目标文件的路径
            USES CONVERSION;
            WCHAR m_szWFileName[MAX_FILE_PATH];
             wcscpy( m_szWFileName,T2W(szFileName) );//转换成宽字符串
  //创建一个输出流,绑定到wav文件
   CSpStreamFormat OriginalFmt;
   CComPtr<ISpStream> cpWavStream;
  CComPtr<ISpStreamFormat> cpOldStream;
  HRESULT hr =m_cpVoice->GetOutputStream(&cpOldStream );
  if (hr == S_OK) hr =OriginalFmt.AssignFormat(cpOldStream);
  else hr =E_FAIL;
  // 使用sphelper.h中提供的函数创建 wav文件
  if (SUCCEEDED(hr))
     hr = SPBindToFile(\ m\_szWFileName,\ SPFM\_CREATE\_ALWAYS, \&cpWavStream, \&OriginalFmt. Formatld and the support of the support 
(),OriginalFmt.WaveFormatExPtr());
  if( SUCCEEDED( hr ) )
     //设置声音的输出到 wav 文件, 而不是speakers
     m_cpVoice->SetOutput(cpWavStream, TRUE);
   }
```

```
//开始朗读
   m cpVoice->Speak( wChar, SPF ASYNC |SPF IS NOT XML, 0 );
  //等待朗读结束
   m_cpVoice->WaitUntilDone( INFINITE );
    cpWavStream.Release();
  //把输出重新定位到原来的流
   m_cpVoice->SetOutput( cpOldStream, FALSE );
      • 8 设置朗读音量和速度
            m_cpVoice->SetVolume((USHORT)hpos); //设置音量, 范围是 0 -100
            m_cpVoice->SetRate(hpos); //设置速度, 范围是 -10 - 10
      • 9 设置SAPI通知消息。
     SAPI在朗读的过程中,会给指定窗口发送消息,窗口收到消息后,可以主动获取SAPI的事件,根据事件的不
同,用户可以得到当前SAPI的一些信息,比如正在朗读的单词的位置,当前的朗读口型值(用于显示动画口
型,中文语音的情况下并不提供这个事件)等等。要获取SAPI的通知,首先要注册一个消息:
  m_cpVoice->SetNotifyWindowMessage( hWnd,WM_TTSAPPCUSTOMEVENT, 0, 0 );
   这个代码一般是在主窗口初始化的时候调用,hWnd是主窗口(或者接收消息的窗口)句柄。
WM_TTSAPPCUSTOMEVENT是用户自定义消息。在窗口响应WM_TTSAPPCUSTOMEVENT消息的函数中,
通过如下代码获取sapi的通知事件:
                     event; // 使用这个类,比用 SPEVENT结构更方便
  CSpEvent
   while(event.GetFrom(m_cpVoice) == S_OK)
       switch( event.eEventId )
       {
       }
  eEventID有很多种,比如SPEI_START_INPUT_STREAM表示开始朗读,SPEI_END_INPUT_STREAM表示
朗读结束等。
  可以根据需要进行判断使用。
7: 语音识别
利用微软Speech SDK 5.1在MFC中进行语音识别开发时的主要步骤,以Speech API 5.1+VC6为例:
1、初始化COM端口
一般在CWinApp的子类中,调用ColnitializeEx函数进行COM初始化,代码如下:
::ColnitializeEx(NULL,COINIT_APARTMENTTHREADED); // 初始化COM
注意:调用这个函数时,要在工程设置(project settings)->C/C++标签,Category中选Preprocessor,在
Preprocessor definitions:下的文本框中加上",_WIN32_DCOM"。否则编译不能通过。
2、创建识别引擎
微软Speech SDK 5.1 支持两种模式的:共享(Share)和独享(InProc)。一般情况下可以使用共享型,大的服
务型程序使用InProc。如下:
hr = m\_cpRecognizer.CoCreateInstance(CLSID\_SpSharedRecognizer);//Share
hr = m\_cpRecognizer.CoCreateInstance(CLSID\_SpInprocRecognizer); //InProcRecognizer) = m\_cpRecognizer = m\_c
如果是Share型,可直接进到步骤3;如果是InProc型,必须使用 ISpRecognizer::SetInput 设置语音输入。如
CComPtr<ISpObjectToken> cpAudioToken; //定义一个token
hr = SpGetDefaultTokenFromCategoryId(SPCAT_AUDIOIN, &cpAudioToken); //建立默认的音频输入对象
if \ (SUCCEEDED(hr)) \ \{ \ hr = m\_cpRecognizer-> SetInput(cpAudioToken, \ TRUE); \} \\
CComPtr<ISpAudio> cpAudio; //定义一个音频对象
hr = SpCreateDefaultObjectFromCategoryId(SPCAT AUDIOIN, &cpAudio);//建立默认的音频输入对象
hr = m_cpRecoEngine->SetInput(cpAudio, TRUE);//设置识别引擎输入源
```

```
3、创建识别上下文接口
调用 ISpRecognizer::CreateRecoContext 创建识别上下文接口(ISpRecoContext),如下:
hr = m_cpRecoEngine->CreateRecoContext( &m_cpRecoCtxt );
4、设置识别消息
调用 SetNotifyWindowMessage 告诉Windows哪个是我们的识别消息,需要进行处理。如下:
hr = m_cpRecoCtxt->SetNotifyWindowMessage(m_hWnd, WM_RECOEVENT, 0, 0);
SetNotifyWindowMessage 定义在 ISpNotifySource 中。
5、设置我们感兴趣的事件
其中最重要的事件是"SPEI_RECOGNITION"。参照 SPEVENTENUM。代码如下:
const ULONGLONG ullInterest = SPFEI(SPEI_SOUND_START) | SPFEI(SPEI_SOUND_END) | SPFEI
(SPEI_RECOGNITION);
hr = m_cpRecoCtxt->SetInterest(ullInterest, ullInterest);
6、创建语法规则
语法规则是识别的灵魂,必须要设置。分为两种,一种是听说式(dictation),一种是命令式(command and
control---C&C)。首先利用ISpRecoContext::CreateGrammar 创建语法对象,然后加载不同的语法规则,如
//dictation
hr = m_cpRecoCtxt->CreateGrammar( GIDDICTATION, &m_cpDictationGrammar );
if (SUCCEEDED(hr))
{
hr = m_cpDictationGrammar->LoadDictation(NULL, SPLO_STATIC);//加载词典
//C&C
hr = m_cpRecoCtxt->CreateGrammar( GIDCMDCTRL, &m_cpCmdGrammar);
然后利用ISpRecoGrammar::LoadCmdxxx 加载语法,例如从CmdCtrl.xml中加载:
WCHAR wszXMLFile[20]=L"";
MultiByteToWideChar(CP_ACP, 0, (LPCSTR)"CmdCtrl.xml", -1, wszXMLFile, 256);//ANSI转UNINCODE
hr = m_cpCmdGrammar->LoadCmdFromFile(wszXMLFile,SPLO_DYNAMIC);
注意: C&C时,语法文件使用xml格式,参见Speech SDK 5.1 中的 Designing Grammar Rules。简单例子:
< GRAMMAR LANGID="804">
 <DFFINE>
  <ID NAME="CMD" VAL="10"/>
 <RULE NAME="COMMAND" ID="CMD" TOPLEVEL="ACTIVE">
  <L>
尹成</P>
山东大学
中科院
  </L>
 </RULE>
</GRAMMAR>
LANGI*="804"代表简体中文,在<*>...</*>中增加命令。
7、在开始识别时,激活语法进行识别
hr = m\_cpDictationGrammar->SetDictationState(\ SPRS\_ACTIVE\ );//dictation
hr = m_cpCmdGrammar->SetRuleState( NULL,NULL,SPRS_ACTIVE );//C&C
8、获取识别消息,进行处理
截获识别消息(WM_RECOEVENT),然后处理。识别的结果放在CSpEvent的ISpRecoResult 中。如下:
USES_CONVERSION;
CSpEvent event;
switch (event.eEventId)
   case SPEI_RECOGNITION:
{
```

//识别出了语音输入

```
m_bGotReco = TRUE;
  static const WCHAR wszUnrecognized[] = L"<Unrecognized>";
 CSpDynamicString dstrText;
 //取得识别结果
 if (FAILED (event.RecoResult() -> GetText(SP\_GETWHOLEPHRASE, SP\_GETWHOLEPHRASE, SP\_GTWHOLEPHRASE, SP\_GTWHOLEPHR
 TRUE ,&dstrText, NULL)))
 dstrText = wszUnrecognized;
}
BSTR SRout;
 dstrText.CopyToBSTR(&SRout);
  CString Recstring;
 Recstring.Empty();
 Recstring = SRout;
 //进一步处理
}
 break;
 9、释放创建的引擎、识别上下文对象、语法等。调用相应的Release函数即可。
  参考:
```

http://blog.csdn.net/yincheng01/archive/2010/02/20/5313204.aspx http://blog.csdn.net/buaalei/archive/2010/03/12/5372544.aspx

# 顶 踩 。。。

上一篇 如何通过VC的 CHttpFile 抓取网页内容

下一篇 oracle 循环更新表的数据

## 我的同类文章

## vc/c++ (36)

 ・ VC获取系统空闲时间
 2012-05-29 阅读 657
 ・ vc 屏蔽esc键和回车键
 2012-05-29 阅读 866

 ・ VC++ INI文件修改,读取键值
 2012-04-12 阅读 837
 ・ Unicode Cstring转char互转
 2012-04-09 阅读 718

 ・ vc 遍历文件夹,删除文件...
 2012-03-31 阅读 515
 ・ 如何学才能迅速掌握C++?
 2012-03-19 阅读 284

 ・ VC SQLite 中文转换
 2012-03-12 阅读 867
 ・ c#进程间通信-接收消息
 2012-03-08 阅读 663

 ・ 基于ADO的VC++数据库操...
 2012-03-07 阅读 454
 ・ 五种vc++数据库访问技术的...
 2012-03-06 阅读 368

## 猜你在找

iOS8-Swift开发教程 VC++游戏开发基础系列从入门到精通 MOS认证考试——PPT2013(考试代码77-422) 最适合自学的C++基础视频 HTML 5移动开发从入门到精通

关于使用微软Speech Application SDK实现中文语音 VC++如何编写一个语音识别的类for Micosoft speech microsoft speech语音识别技术语音朗读技术概述 C#基于Microsoft Speech SDK Version 51 的开发语 Android开发学习45使用google语音识别引擎Google



公司简介 | 招贤纳士 | 广告服务 | 银行汇款帐号 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-600-2320 | 北京创新乐知信息技术有限公司 版权所有 | 江苏乐知网络技术有限公司 提供商务支持 京 ICP 证 09002463 号 | Copyright © 1999-2014, CSDN.NET, All Rights Reserved 😲