

Re: WideCharToMultiByte returns default character when input language for non-Unicode programs set to English

## Re: WideCharToMultiByte returns default character when input language for non-Unicode programs set to English

---

*Source:*

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.gdi/2006-06/msg00095.html>

---

- *From:* "John Carson" <jcarson\_n o sp am @xxxxxxxxxxxxxxxxxx>
  - *Date:* Sat, 17 Jun 2006 23:57:07 +1000
- 

"David Liebtag" <liebtag@xxxxxxxxxx> wrote in message  
<news:eHba2KXkGHA.2200@xxxxxxxxxxxxxxxxxxxxxxxx>

I'm having a problem with WideCharToMultiByte. Maybe I just don't understand how it is supposed to work.

I have support for Japanese installed on my machine. If I set the language for non-Unicode programs to Japanese, and call WideCharToMultiByte to translate a Unicode string to Multibyte and use codepage CP\_ACP or 932, it works fine.

However, if I set the language for non-Unicode programs to English (United States), reboot, and do the same thing, the Japanese characters are replaced by the default character (which is question mark in 932.)

How are you determining this? From the screen display of the text string or by examining the numerical value of the characters in the text string?

GetCPIInfo still reports that 932 is installed and available.

Can anyone explain why WideCharToMultiByte is not returning the Shift-JIS representation of the Japanese characters?

What WideCharToMultiByte does has nothing to do with the currently active code page on your machine. It simply translates a buffer of WCHARS into a buffer of CHARS based on the arguments you use when you call it.

Of course to actually see those CHARS displayed properly on screen, you need to have the code page for your system set appropriately.

The following program snippet illustrates how it works.

Re: WideCharToMultiByte returns default character when input language for non-Unicode program\$ set to E

Re: WideCharToMultiByte returns default character when input language for non-Unicode programs set to English

We start with the Unicode character 0x547D, which is the Shift-JIS multibyte character 0x96BD. The program converts the Unicode string to the multibyte string, which means that the multibyte string contains 0x96BD. This can be verified by converting the characters to integers and outputting the integers.

```
// do the conversion
WCHAR wch[] = {(WCHAR)0x547D, 0};
CHAR ch[10];
int count = WideCharToMultiByte(932, 0, wch, -1, ch, sizeof ch, 0, 0);

// show it worked by outputting characters as integers
int charNumber[] = {(unsigned char)ch[0], (unsigned char)ch[1]};
TCHAR buffer[50];
_stprintf(buffer, _T("multibyte character number is 0x%X%X"),
charNumber[0],charNumber[1]);
TextOut(hdc, 0, 0, buffer, _tcslen(buffer));

// Now we display both the Unicode and multibyte strings. The
// unicode string's display is independent of the code page, but the
// multibyte string will only display correctly if the appropriate
// code page is in effect
TextOutW(hdc, 0, 30, wch, wcslen(wch));
TextOutA(hdc, 0, 60, ch, count-1);
```

If there is a better news group for this question, please tell me which one to use.

microsoft.public.win32.programmer.international

--  
John Carson

.

Re: WideCharToMultiByte returns default character when input language for non-Unicode program set to English