
Subject: Re: How to display Traditional Chinese (Big-5)?

Posted by [mirek](#) on Fri, 23 Jun 2006 17:57:49 GMT

[View Forum Message](#) <> [Reply to Message](#)

yoco wrote on Fri, 23 June 2006 13:35No, I didn't.

I didn't know that I can set default charset by this function.

Thank you for tell me that

Is it on the manual already? (I mean user can set default charset through this function.)

I have another problem.

Since the upp does not support Big5,

so I decide to use UNICODE in my application.

I set default charset to CHARSET_UNICODE in the beginning of the program.

And save my text file in UNICODE,

But it display my UNICODE text fall.

=====

My code..

```
class test : public WithtestLayout<TopWindow>
{
public:
typedef test CLASSNAME;
String s ;
test()
{
FileIn fin ( "test.txt" ) ; // In unicode format
s = fin.GetLine() ;
}
virtual void Paint(Draw& w)
{
w.DrawRect(GetSize(), SWhite);
w.DrawText( 0, 0, s, Arial(16), Black);
}
};
```

=====

I found that the definition of CHARSET_UNICODE and CHARSET_UTF8 are both 255,
so in the function

```
WString ToUnicode(const char *src, int l, byte charset){
charset = ResolveCharset(charset);
if(charset == CHARSET_UTF8)
return FromUtf8(src, l);
WStringBuffer result(l);
```

```
ToUnicode(result, src, l, charset);  
return result;  
}
```

it always pass the string to the function FromUtf8(),
but the original string read from the file is UNICODE already.

Do I must save my text file in UTF-8?

Thanks.

Files in U++ are considered to be the stream of bytes.

To read 16-bit unicode file, you should read individual words. Use Get16le (or Get16be for big-endian files) to read individual characters.

Of course, UTF-8 is possible and good alternative. However, while UTF-8 is great for latin alphabets, it is less ideal for chienne - in latin languages, UTF-8 can reduce the size of file (because most characters are from basic ASCII set and therefore represented by single), while for chienne you end with 3-byte combos.

Mirek
