Subject: Re: Building & using U++ without TheIDE
Posted by sergei on Fri, 14 Sep 2007 00:21:15 GMT
View Forum Message <> Reply to Message

luzr wrote on Fri, 14 September 2007 00:58Quote:
3) I don't understand how unicode is implemented. There is String, AString, WString, but there is no TString, or whatever the name, like there is TCHAR that expands into char or wchar_t, depending on whether UNICODE/_UNICODE is defined. How do I define whether I'm in unicode or not?

This is sort of irrelevant. There is no UNICODE mode. All the time you have 8-bit and 16-bit String/WString.

Recommended approach is to use UTF-8 encoding. In that case, both strings can contain unicode and there is simple conversion between them. (You can however use one of 15 WIN/ISO encodings as 8-bit default.

Quote:
I mean, MessageBox will expand into MessageBoxA or MessageBoxW?

Always into MassageBoxA. However, in U++ you rather use Prompt, which can work with UTF8.

Quote:
And why path handling routines use char - can I handle unicode filenames with U++?

Unfortunately, there is drawback caused by fact that we still have to support win98, so we cannot use W variants . In practive, this is really minor trouble, but nothing to be happy about it.

Anyway, when you are using only functions from U++, there is automatic conversion between U++ default encoding and 8-bit encoding of Windows.

Mirek

That would be one pretty serious drawback IMHO. I'm using Russian and Hebrew, and I've used so many apps with bad unicode support that I really want to avoid making another one. Who uses Win98 nowadays (and there's unicode layer for these)

I've tried entering non-English chars in TheIDE source editor, it didn't let me. So I made a UTF-8 file containing a name of a file (with both Hebrew and Russian characters). U++ was able to read the name but unable to load the file. I guess this is the reason:
handle = CreateFile(ToSystemCharset(name),

Since this is CreateFileA, it can't open files with unicode filenames.

I tried adding #define UNICODE and #define _UNICODE. After commenting some stuff (cAlternateFileName didn't exist) it built the program but didn't work. CreateFileW can't accept UTF-8, it needs unicode.

Could this be fixed by some creative editing of Util.h? And what about window title, and controls? I mean, any chance to bring this to the TCHAR style, so the program can be compiled in ANSI, and in Unicode (supporting several languages simultaneously)? This should be a matter of some defines and a function to convert UTF8 to Win32 unicode. Or are there other parts of U++ that rely on a specific String/WString encoding?