Subject: Unicode support is now pretty seamless, right?
Posted by xrysf03 on Tue, 18 Dec 2018 21:54:40 GMT
View Forum Message <> Reply to Message

Hello everybody,
another day, another noob observation...

I live in Windows 7 at the moment, and using the Mingw tool chain.

While trying to set the window title text in an early "hello world" proggie, I quickly found the Title() method of the TopWindow class... and I've noticed that it supposedly takes a WString input. I went on to look for the header file to include, to be able to instantiate WString, I was searching old forum posts for functions to convert between char* and Unicode strings, I considered using #define UNICODE at the top of my source files etc. I also tried making a std::vector<std::string> and then use c_str() to get my hands on the internal char* ... And I kept getting stuck in various ways.

Then I realized that I can just create a Vector<String> (within namespace Upp) and I can use the String instances contained in the Vector directly as input to Title() - e.g. if I index into the Vector using operator[], even though the Assist does not mention such a version of the Title() prototype, i.e. taking Upp::String as input :) Does some automatic conversion from String to WString take place? An automatic intermediate WString object construction or some such?

Also, I can simply assign text string literals written in "eastern latin" to String variables, and I don't need to care about encoding... Does TheIDE use UTF-8 as a default encoding of the source code files? It would seem so, if I'm reading the hints from VIM and Notepad correctly...

It all feels pretty seamless - I feel like I can forget about encodings altogether and just "do my thing", in an ignorant autopilot mode... Is it really that simple? I mean - using Unicode in U++ :)
I would expect to meet some trouble as soon as I deviate from the U++ universe (stuff living in the Upp:: namespace) and start dealing with STL std:: stuff or syscalls and external DLL functions using plain old ASCII char*.

Any futher hints/ideas are welcome...

Frank