
Subject: Re: WinAPI UNICODE question
Posted by [cbpporter](#) on Tue, 09 Dec 2008 08:41:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

I am going to give now a concrete example with attached modified files. I've chosen Color.h and Color.cpp to illustrate some points because they are so easy and universally used throughout my modified files.

What I done here is added inline copies of the function GetRValue to RGB with different names and used them. GetRValue and friends are declared as macros in windows.h and as inline functions under POSIX so it would seem that we are covered in regards to cross-platform compatibility. Unfortunately, this is not the case.

The biggest problem is that you can not write code without having using namespace Upp somewhere before. If I try to use Upp::GetRValue, it will compile and work fine under Linux, but not under Windows. I haven't encountered this problem with exactly this function, but I did with other functions which I'm going to touch in later posts (I'm looking at GetTickCount especially).

Another problem is that under Linux, you can get the adress of such functions, while under Windows the same code will not compile again.

This is why I've chosen to introduce new names. If you don't like the solution, GetRValue and friends could be undefed and defined as macros under all platforms and in Upp namespace, not just under POSIX. But this could lead to name clashes if someone later includes windows.h again.

File Attachments

1) [Core.zip](#), downloaded 485 times
