
Subject: Re: How to determine if U++ is being utilized...

Posted by [ptkacz](#) on Sat, 04 Apr 2020 00:50:11 GMT

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

Thanks Novo, to both your answers. I'm familiar with defining macros and making use of the logic directives (i.e. #if, etc). I was looking for something that could avoid having to manually make a macro definition, or prior inclusion, and then have to always make sure that things are defined or not when hopping from one application (project) to another.

After a little bit more digging on the web, it looks like C++17 has added to it for use with the marco pre-processor, a `__has_include()` function that checks if a header reference exists. Yesterday after updating my system, I tested out the following:

In TheIDE:

```
#if __has_include("CtrlLib/CtrlLib.h")
    CtrlLayout(*this, "U++ Window title");
#else
    CtrlLayout(*this, "U++ not here");
#endif
```

and even,

In Code::Blocks:

```
#if __has_include("iostream")
    cout << "iostream here." << endl;
#else
    cout << "iostream not here" << endl;
#endif

#if __has_include("CtrlLib/CtrlLib.h")
    cout << "U++ Window here." << endl;
#else
    cout << "U++ not here" << endl;
#endif
```

`__has_include()` worked in all circumstances, so it looks like I'll have to be making C++17 for just this one feature.

Some additional information on it can be found here,
<https://en.cppreference.com/w/cpp/preprocessor/include>

Again, thanks for your input!

Peter
