Subject: Re: RE: Job package: A scope-bound worker thread for non-blocking operations.
Posted by mirek on Wed, 11 Oct 2017 07:23:30 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Tue, 10 October 2017 23:48Hello Mirek,

tested Async/AsyncWork with MSC 2017, MinGW on windows, and with GCC on linux.

- It compiles on MSC without any hiccup.
- It does not compile on GCC (7.2) or MinGW unless the nested classes are moved out, (That's why I wrote my prototype that way.) and "Ret" is changed to some other parameter name. Here are the error codes I get:

After a bit of wrestling with C++11, it now compiles.

Quote:
- More importantly there seems to be something wrong with the exception propagation mechanism. For,

1) Sometimes it fails to catch the exception, and the application crashes with that exception.
2) When it catches the exception the application hangs at the end (after the "exception caught" message is printed.)
3) Sometimes the application simply hangs.

I got this erratic behaviour both on windows and on linux, on a single machine, so it maybe a local hardware problem, I need to investigate it further...

I would like to investigate, but need a testcase.

Quote:
Sure, but can this really be attributed to a design flaw?
Recursion is potentially tricky by nature, and requires the developer to be extra cautious with his/her assumptions.

Well, recursion aside, I think that in general, we want the mechanism reentrant. I mean, it should not be a part of contract/function documentation whether is it using IsFinished internally.

I can actually fix the issue, but then IsFinished will not be doing the same thing.

BTW, the very same issue is true for CoWork. So something to think about..

Mirek