Subject: BETA status handling
Posted by [koldo](#) on Wed, 18 Sep 2019 13:18:26 GMT
View Forum Message <> Reply to Message

Hello all

When having an application in beta status, there is a big risk of having ASSERTs called in Release mode binaries.

I propose you to add a new flag like BETA, so the ASSERT could be reconverted into throw Exc() or other way so that the software can handle the error, instead of just crashing abruptly the software. Last is a terrible user experience.

What do you think?

Subject: Re: BETA status handling
Posted by [mirek](#) on Wed, 18 Sep 2019 13:40:15 GMT
View Forum Message <> Reply to Message

ASSERTs are not active in release mode.

Mirek

Subject: Re: BETA status handling
Posted by [koldo](#) on Wed, 18 Sep 2019 15:26:07 GMT
View Forum Message <> Reply to Message

This is the reason I propose an intermediate BETA status, where ASSERTs could be handled by software trying to catch the problems.
In beta, after an ASSERT, a warning window is ugly, but a software crash is much worse.

Subject: Re: BETA status handling
Posted by [mirek](#) on Wed, 18 Sep 2019 16:24:54 GMT
View Forum Message <> Reply to Message

koldo wrote on Wed, 18 September 2019 17:26This is the reason I propose an intermediate BETA status, where ASSERTs could be handled by software trying to catch the problems.
In beta, after an ASSERT, a warning window is ugly, but a software crash is much worse.

Maybe we could introduce Panic hook so that you can install throw routine there?

Subject: Re: BETA status handling

Posted by koldo on Wed, 18 Sep 2019 16:47:28 GMT

That could be great.
It could catch all ASSERT and panic conditions that cost time to be checked (and are void in Release mode), to throw something to be catch at higher level.
This could be set by a flag, so normal Release binaries would not suffer the drawback of, for example, having to check the bounds of any access to a container.

Subject: Re: BETA status handling
Posted by mirek on Thu, 03 Oct 2019 14:10:17 GMT

koldo wrote on Wed, 18 September 2019 18:47That could be great.
It could catch all ASSERT and panic conditions that cost time to be checked (and are void in Release mode), to throw something to be catch at higher level.
This could be set by a flag, so normal Release binaries would not suffer the drawback of, for example, having to check the bounds of any access to a container.

Actually, it is there for ages:

InstallPanicMessageBox

That one gets called on Panic, so putting throw there (instead of displaying the error, or in addition to doing so) should do exactly what you need...

Mirek

Subject: Re: BETA status handling
Posted by koldo on Sun, 06 Oct 2019 11:56:24 GMT

Thank you Mirek. However that is not exactly the same.

InstallPanicMessageBox() and SetAssertFailedHook() let to install panic handlers.

However, in case of the ASSERTs, this only works in DEBUG mode. Please check this code in Defs.h:

```
#ifdef _DEBUG
 #define ASSERT_(x, msg)  ((x) ? (void)0 : ::Upp ::AssertFailed(__FILE__, __LINE__, msg))
 #define ASSERT(x)      ASSERT_(x, #x)
#else
 inline  void LOGNOP__() {}
 #define LOG_NOP       Upp::LOGNOP__()
 #define ASSERT_(x, msg)  LOG_NOP
```

```
#define ASSERT(x)        LOG_NOP
#endif
```
Out of DEBUG, ASSERTs seem to do nothing, as are disconnected to AssertFailed().

Subject: Re: BETA status handling
Posted by mirek on Mon, 07 Oct 2019 08:23:50 GMT
View Forum Message <> Reply to Message

Yes, that is the principal difference between DEBUG and RELEASE.

What you are asking for here is to make RELEASE behave as DEBUG. Why not use DEBUG then?

Mirek

Subject: Re: BETA status handling
Posted by koldo on Tue, 08 Oct 2019 07:19:54 GMT
View Forum Message <> Reply to Message

Hello Mirek

Release version is faster and smaller.
However, do not worry. This is not an important issue.

Subject: Re: BETA status handling
Posted by mirek on Tue, 08 Oct 2019 08:30:30 GMT
View Forum Message <> Reply to Message

koldo wrote on Tue, 08 October 2019 09:19Hello Mirek

Release version is faster and smaller.
However, do not worry. This is not an important issue.

Release version is faster and smaller exactly because all those ASSERTs are not being checked :)

Other than that, you can configure things to optimize debug in exactly same way as release.

Mirek

Subject: Re: BETA status handling
Posted by koldo on Tue, 08 Oct 2019 14:54:02 GMT
View Forum Message <> Reply to Message

I am sorry to disagree with you. I bet you the difference in speed between a RELEASE version with ASSERTS with throws, and a DEBUG version, is already very significant :)

However, do not worry. This is not an important issue.