
Subject: Debugger feature request

Posted by [unodgs](#) on Mon, 20 Feb 2012 09:29:02 GMT

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

Would be possible to add the option "don't step into Core"? I mean very often during debugging I'm taken to the body of for example `Vector::GetCount()` or `Rect` class' constructor. That takes time and from my application point of view is only time wasting.

In IntelliJ idea if someone really wants to debug standard library he presses another debugger shortcut (it could be shift-f11 for instance in theide)

Subject: Re: Debugger feature request

Posted by [cbpporter](#) on Mon, 20 Feb 2012 11:04:56 GMT

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

unodgs wrote on Mon, 20 February 2012 11:29 Would be possible to add the option "don't step into Core"? I mean very often during debugging I'm taken to the body of for example `Vector::GetCount()` or `Rect` class' constructor. That takes time and from my application point of view is only time wasting.

In IntelliJ idea if someone really wants to debug standard library he presses another debugger shortcut (it could be shift-f11 for instance in theide)

I second that! In fact I suggested that a long time ago. I think my simple solution was to automatically step over things in the Upp namespace, but I'm not sure this can be done easily.

An another think I would greatly appreciate: currently you can't have two variables with the same name in different block. The debugger gets confused by this. This is really annoying because every for loop variable needs a different name.

And debugging inspecting global variables and static class members would be great too.

I sincerely love U++ and use it to great results on a day by day basis, but TheIDE is clearly the weakest link. Never mind that it somehow gets confused and start misinterpreting key presses at least once every two days.

Subject: Re: Debugger feature request

Posted by [unodgs](#) on Mon, 20 Feb 2012 11:24:02 GMT

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

cbpporter wrote on Mon, 20 February 2012 06:04

An another think I would greatly appreciate: currently you can't have two variables with the same name in different block. The debugger gets confused by this. This is really annoying because every for loop variable needs a different name.

And debugging inspecting global variables and static class members would be great too.

I'd love to see it fixed as well. Another missing thing is showing container elements. Currently only

data from first element is visible.

Subject: Re: Debugger feature request
Posted by [281264](#) on Sat, 25 Feb 2012 09:28:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

I agree with you guys. Certainly, any improvement in the debugger would be very welcome. For example, the debugger is not able to inspect the STL containers (vector, etc,...); although in U++ we have our NTL, sometimes it is convenient to use the STL.

Best regards,

Javier

Subject: Re: Debugger feature request
Posted by [mdelfede](#) on Sat, 25 Feb 2012 13:10:56 GMT
[View Forum Message](#) <> [Reply to Message](#)

For Linux part, using the new MI2 interface, it's quite easy to add STL support... The python code it's even ready to download somewhere and it's enough to add it to PrettyPrinters.py code.

About the windows part, my knowledges are too limited....

Max

Subject: Re: Debugger feature request
Posted by [mirek](#) on Wed, 29 Feb 2012 07:03:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

I think there should and will be Windows refactoring, adding win64 support and matching MI2 capabilities.

But I guess it is gona happen in next cycle (or somebody else has to do it . This one is dedicated to server related improvements (Core, Web, Skylark).

Subject: Re: Debugger feature request
Posted by [unodgs](#) on Wed, 29 Feb 2012 10:14:23 GMT
[View Forum Message](#) <> [Reply to Message](#)

mirek wrote on Wed, 29 February 2012 02:03I think there should and will be Windows refactoring, adding win64 support and matching MI2 capabilities.

What about using gdb both on windows and linux. It seems that gdb is getting better and better.

Subject: Re: Debugger feature request

Posted by [mirek](#) on Wed, 29 Feb 2012 10:35:11 GMT

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

unodgs wrote on Wed, 29 February 2012 05:14mirek wrote on Wed, 29 February 2012 02:03I think there should and will be Windows refactoring, adding win64 support and matching MI2 capabilities.

What about using gdb both on windows and linux. It seems that gdb is getting better and better.

Can it debug msc produced .exe?

Mirek

Subject: Re: Debugger feature request

Posted by [unodgs](#) on Wed, 29 Feb 2012 10:55:14 GMT

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

Sorry. I don't know what I was thinking about VC uses completely different debug format... and for sure gdb doesn't support it.