Subject: Problem with gcc compilers and -Wall
Posted by koldo on Wed, 08 Sep 2010 07:03:19 GMT

View Forum Message <> Reply to Message

Hello Mirek

In some recent posts it has been oberrved that code compiled with gcc/minqw had non void functions without a return.

This problem is easy to have and produce strange errors.

It would be great if you could add "-Wreturn-type" parameter when using gcc/mingw compiler.

In the same way, the "-Wall" could be considered. Perhaps it is pedantic, but it make sense as many little warnings can produce weird errors.

Subject: Re: Problem with gcc compilers and -Wall
Posted by dolik.rce on Wed, 08 Sep 2010 08:58:27 GMT

View Forum Message <> Reply to Message

Hi Koldo!

I believe that -Wall is overkill. Unless you want to add all those missing parentheses around && within || (which is the most common warning I think) to make the output at least a little bit readable Also there is a lot of warnings in 3rd party code in plugin directory, which obscure the output (even without -Wall).

I think that it should be up to the user to set up how pedantic should the compiler be. It is just a matter of writing the options into the build method setting. At least for debug mode, it is more difficult for release mode, but possible too and seeing the warnings in debug mode should be enough anyway.

Best regards,
Honza

Subject: Re: Problem with gcc compilers and -Wall
Posted by koldo on Wed, 08 Sep 2010 09:12:16 GMT

View Forum Message <> Reply to Message

Hello Honza

Perhaps "-Wall" is pedantic. However I have seen this week two errors because not using "-Wreturn-type" in Forum and at home .

I think "-Wreturn-type" (and perhaps other) should be included as it is more than a warning. It

causes weird errors.

In fact, I captured this error because MSC detects it (and I always test the apps with two compilers just in case).

Subject: Re: Problem with gcc compilers and -Wall
Posted by mr_ped on Thu, 09 Sep 2010 07:33:00 GMT
View Forum Message <> Reply to Message

-Wall is the first thing I set up before creating new project.
It's never an overkill. (ok, I know 1 or 2 type of warnings which are really not helpful because I never made such error they warn against and use that deliberately, but 99% of -Wall are valid objections and code can be improved)

edit:
"I think that it should be up to the user to set up how pedantic should the compiler be."

Of course, but that means that the upp core should be very shy when compiled with -Wall to not disturb.

Subject: Re: Problem with gcc compilers and -Wall
Posted by jerson on Thu, 09 Sep 2010 08:13:18 GMT
View Forum Message <> Reply to Message

How do I set this -Wall option or the -Wreturn-type in my IDE? Perhaps it is wise to do this as a newbie.

Subject: Re: Problem with gcc compilers and -Wall
Posted by mr_ped on Thu, 09 Sep 2010 08:15:55 GMT
View Forum Message <> Reply to Message

I'm using Setup/Build methods - Debug options.

Subject: Re: Problem with gcc compilers and -Wall
Posted by dolik.rce on Thu, 09 Sep 2010 08:22:44 GMT
View Forum Message <> Reply to Message

mr_ped wrote on Thu, 09 September 2010 09:33Of course, but that means that the upp core should be very shy when compiled with -Wall to not disturb.
You are right about this Ped. Maybe if the Core didn't put out so many useless warning I would use the option sometimes too. I guess we should fill in those missing brackets and parenthesis...

Honza

Subject: Re: Problem with gcc compilers and -Wall
Posted by koldo on Thu, 09 Sep 2010 08:51:52 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Thu, 09 September 2010 10:22mr_ped wrote on Thu, 09 September 2010 09:33Of course, but that means that the upp core should be very shy when compiled with -Wall to not disturb.
You are right about this Ped. Maybe if the Core didn't put out so many useless warning I would use the option sometimes too. I guess we should fill in those missing brackets and parenthesis...

Honza
!

Subject: Re: Problem with gcc compilers and -Wall
Posted by mr_ped on Thu, 09 Sep 2010 16:58:41 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Thu, 09 September 2010 10:22mr_ped wrote on Thu, 09 September 2010 09:33Of course, but that means that the upp core should be very shy when compiled with -Wall to not disturb.
You are right about this Ped. Maybe if the Core didn't put out so many useless warning I would use the option sometimes too. I guess we should fill in those missing brackets and parenthesis...

Honza

Mirek was against that. (I did push him into fixing lot of warning a 1-1.5 year ago, but he didn't like the additional () ... well, after all, he *is* C++ developer, so he does remember operators priority. I put () everywhere and don't care, I know I will never learn C++ fully.)

Subject: Re: Problem with gcc compilers and -Wall
Posted by dolik.rce on Thu, 09 Sep 2010 19:45:50 GMT
View Forum Message <> Reply to Message

Well, then maybe -Wall -Wno-parentheses? BTW: I always thought that those extra parentheses are suggested to prevent trouble in case someone writes non-conforming compiler

Also -Wno-reorder would be nice, but that would have to be set only for C++, for C files it prints out warning.

Looking at the warnings, I observed following:

There is a lot of unused-variable warnings and few unused-function. Those could be fixed easily

---

and it might improve readability.

There are some warnings that sound pretty dangerous. Like uninitialized variables. Or even more

that is bit messy, but probably OK (as far as I can tell):q = q > 0 ? d.name.ReverseFind(':', q) : q = d.name.ReverseFind(':');

have to be C++ wizard to know that

Honza

Subject: Re: Problem with gcc compilers and -Wall
Posted by jerson on Fri, 10 Sep 2010 03:11:27 GMT
View Forum Message <> Reply to Message

IMO, the warnings should be eliminated altogether if they can be. It gets messy for newbies to sift through the warnings and know which are caused by their own follies.

If it doesn't break the IDE functionality, I'd say, fix it please.

Subject: Re: Problem with gcc compilers and -Wall
Posted by dolik.rce on Fri, 10 Sep 2010 10:45:07 GMT
View Forum Message <> Reply to Message

Maybe we could (optionally) silence (-Wno-xxx) the most common errors in U++ code (that is anything that is in uppsrc nest) and let the user control the verbosity for his portion of code, defaulting to -Wall. But IMHO that is far too fine grained and I would rather prefer adding those parentheses etc. into the code...

Honza

Subject: Re: Problem with gcc compilers and -Wall
Posted by mdelfede on Tue, 14 Sep 2010 09:04:53 GMT
View Forum Message <> Reply to Message

I agree too with -Wall.... It helps writing bugfree code many times.
I also had an error of function with missing return at end, giving weird errors and I could locate it just compiling with MSC which has the warning enabled.
I think that the -Wall with -Wno-parentheses would be the best choice... and I think it would not require too many "fixes" in Upp sources.

Max