# Lasso - Bug #24771

## 2.6.0 build fails if --enable-debugging is passed to configure script

25 juin 2018 23:59 - John Dennis

| | | | | |
|---|---|---|---|---|
| **Statut:** | Fermé | | **Début:** | 25 juin 2018 |
| **Priorité:** | Normal | | **Echéance:** | |
| **Assigné à:** | Benjamin Dauvergne | | **% réalisé:** | 100% |
| **Catégorie:** | | | **Temps estimé:** | 0:00 heure |
| **Version cible:** | 2.6.1 | | | |
| **Patch proposed:** | Oui | | **Planning:** | Non |

**Description**

On line 703 of configure.ac -Werror is added to CFLAGS if enable-debugging is on.

-Werror causes the compilation to fail if any warnings are emitted during compilation.

However lasso version 2.6.0 when built with gcc 8.1.1 has numerous compiler warnings, both in the core and in the bindings. Thus with --enable-debugging the build fails almost immediately (as soon as a warning is emitted). This prevents lasso from being built with debugging support.

**Révisions associées**

**Révision 62d2f4bf - 24 juillet 2018 12:39 - Benjamin Dauvergne**

Move AC_SUBST declaration for AM_CFLAGS with alike (#24771)

Just to reorder things properly in configure.ac.

**Révision 50b5cdac - 24 juillet 2018 16:29 - Benjamin Dauvergne**

Remove -Werror from --enable-debugging (fixes #24771)

GCC 8 has better warnings and it breaks the build on platform already
using it and wanting debugging symbols.

**Historique**

**#1 - 26 juin 2018 10:23 - Benjamin Dauvergne**

I'm ok to remove it, and create a --pedantic option instead.

**#2 - 26 juin 2018 16:21 - John Dennis**

Adding --pedantic to enable -Werror sounds like a good approach to me, it separates two concepts (debugging and strictness) that got co-mingled. At some point the warnings should be looked at, I assume since -Werror has been a part of --debugging for a while it implies there was a recent point in time where there were no warnings so it shouldn't be too hard.

On an almost unrelated note, I've got a number patches prepared to support using Python3 during the build. It appears as if although the generated Python binding had been ported to Py3 other build scripts and the automake files were not ready to handle Py3. It was while I was working on getting the Py3 build working that I tried to turn on debugging and discovered it failed. My quick and dirty interim solution was to just temporarily remove -Werror from the CFLAGS and I was able to move on.

**#3 - 28 juin 2018 10:34 - Benjamin Dauvergne**

*- Version cible mis à 2.6.1*

**#4 - 28 juin 2018 10:35 - Benjamin Dauvergne**

*- Assigné à mis à John Dennis*

**#5 - 24 juillet 2018 12:30 - Benjamin Dauvergne**

*- Assigné à changé de John Dennis à Benjamin Dauvergne*

**#6 - 24 juillet 2018 12:40 - Benjamin Dauvergne**

*- Fichier 0001-Move-AC_SUBST-declaration-for-AM_CFLAGS-with-alike-2.patch ajouté*

*- Fichier 0002-Add-an-enable-pedantic-option-to-prevent-enable-debu.patch ajouté*

*- Statut changé de Nouveau à Solution proposée*

*- Patch proposed changé de Non à Oui*

John could you review my patch please ?

**#7 - 24 juillet 2018 16:31 - Benjamin Dauvergne**

*- Fichier 0002-Remove-Werror-from-enable-debugging-fixes-24771.patch ajouté*

*- Fichier 0001-Move-AC_SUBST-declaration-for-AM_CFLAGS-with-alike-2.patch ajouté*

I forgot there was already a --enable-pedantic option, so I just removed -Werror in these new patches.

**#8 - 24 juillet 2018 16:35 - Benjamin Dauvergne**

*- Statut changé de Solution proposée à Résolu (à déployer)*

*- % réalisé changé de 0 à 100*

Appliqué par commit [50b5cdac871e9af45b10cf8a548c1d1db290dfeb](#).

**#9 - 24 juillet 2018 16:59 - John Dennis**

Hi Benjamin:

I think there are a couple of issues with the patch or perhaps I'm not in sync with what you were trying to achieve, so here's my questions:

Why is there any connection between debugging and the pedantic option? The two seem independent to me. In one case I want to produce a build I can run under the debugger and in the other case I want to see if the compiler can help me diagnose coding mistakes. I don't see any correlation between the two.

There was already an enable pedantic option and now it looks like you're processing that arg twice.

if test "z$enable_pedantic" = "zyes" ; then
enable_pedantic=yes

Isn't the assignment of 'yes' to enable_pedantic a redundant no-op here? The assignment is inside a conditional block that has confirmed that enable_pedantic is already 'yes'.

```
+   if test "z$enable_debugging" "!=" "zyes" ; then
+       LASSO_DEFINES="$LASSO_DEFINES -DLASSO_DEBUG"
+       CFLAGS="$CFLAGS -O0 -g -Wall -Wextra"
+   fi
```

This seems to enforce the requirement that if you enable pedantic you must compile with the option to produce a debuggable build. Is this something gcc requires? If not why are we enforcing this?

I'm not even sure -Wall and -Wextra should be turned on by saying I want a build that I can run under gdb, but I'm OK with it, I can see the value in it but on the other hand I prefer options that do not have unrelated side-effects.

**#10 - 24 juillet 2018 18:10 - Benjamin Dauvergne**

John Dennis a écrit :

> Hi Benjamin:
>
> I think there are a couple of issues with the patch or perhaps I'm not in sync with what you were trying to achieve, so here's my questions:
>
> Why is there any connection between debugging and the pedantic option? The two seem independent to me. In one case I want to produce a build I can run under the debugger and in the other case I want to see if the compiler can help me diagnose coding mistakes. I don't see any correlation between the two.
>
> There was already an enable pedantic option and now it looks like you're processing that arg twice.
>
> if test "z$enable_pedantic" = "zyes" ; then
> enable_pedantic=yes
>
> Isn't the assignment of 'yes' to enable_pedantic a redundant no-op here? The assignment is inside a conditional block that has confirmed that enable_pedantic is already 'yes'.
>
> ```
> +   if test "z$enable_debugging" "!=" "zyes" ; then
> +       LASSO_DEFINES="$LASSO_DEFINES -DLASSO_DEBUG"
> +       CFLAGS="$CFLAGS -O0 -g -Wall -Wextra"
> +   fi
> ```

This seems to enforce the requirement that if you enable pedantic you must compile with the option to produce a debuggable build. Is this something gcc requires? If not why are we enforcing this?

I'm not even sure -Wall and -Wextra should be turned on by saying I want a build that I can run under gdb, but I'm OK with it, I can see the value in it but on the other hand I prefer options that do not have unrelated side-effects.

Finally I choosed to simply remove -Werror from the debugging CFLAGS, so all is well.

**#11 - 06 septembre 2019 14:40 - Benjamin Dauvergne**

*- Statut changé de Résolu (à déployer) à Solution déployée*

**Fichiers**

| | | | |
|---|---|---|---|
| 0001-Move-AC_SUBST-declaration-for-AM_CFLAGS-with-alike-2.patch | 912 octets | 24 juillet 2018 | Benjamin Dauvergne |
| 0002-Add-an-enable-pedantic-option-to-prevent-enable-debu.patch | 1,56 ko | 24 juillet 2018 | Benjamin Dauvergne |
| 0002-Remove-Werror-from-enable-debugging-fixes-24771.patch | 900 octets | 24 juillet 2018 | Benjamin Dauvergne |
| 0001-Move-AC_SUBST-declaration-for-AM_CFLAGS-with-alike-2.patch | 912 octets | 24 juillet 2018 | Benjamin Dauvergne |