

QGIS Application - Feature request #1239

Compilation strict mode and warnings cleanup for GCC 4.1

2008-08-22 05:20 AM - Mateusz Loskot -

Status:	Closed	
Priority:	Low	
Assignee:	nobody -	
Category:	Build/Install	
Pull Request or Patch supplied:		Resolution: fixed
Easy fix?:	No	Copied to github as #: 11299
Description		
<p>I've prepared a patch for <i>trunk/qgis</i> subtree that enables more strict compilation with GCC 4.1 and cleans number of warnings.</p> <p>Here is cmake comand I used to test compilation after the cleanup:</p> <pre>cmake \\ -D CMAKE_INSTALL_PREFIX=\$HOME/usr \\ -D CMAKE_BUILD_TYPE=Debug \\ -D CMAKE_CXX_FLAGS:String="-pedantic -Wall -Wno-long-long -fstrict-aliasing -Wstrict-aliasing=1" \\ -D CMAKE_VERBOSE_MAKEFILE=TRUE ..</pre> <p>I would suggest to use the CMAKE_CXX_FLAGS above as default compilation flags. It really helps to find subtle bugs. During the cleanup work, with help of the stricter compilation mode I've identified following bugs: #1229, #1233, #1234, #1235, #1236</p> <p>The only warning that hasn't been cleaned is</p> <pre>warning: ISO C++ forbids casting between pointer-to-function and pointer-to-object</pre> <p>but this is a well known issue of 'dlsym_ call used with C++ compiler (see #234).</p> <p>The patch was generated against revision r9120 from inside the trunk using <code>_svn diff</code> command.</p>		

Associated revisions

Revision 2e5506a2 - 2008-08-23 04:06 PM - Jürgen Fischer

various code cleanups by mloskot for GCC 4.1 (fixes #1239, #1240).

Inspired by it I also enabled pedantic mode for MSVC and cleaned up some more warnings found by GCC 4.3 and MSVC.

git-svn-id: <http://svn.osgeo.org/qgis/trunk/qgis@9133> c8812cc2-4d05-0410-92ff-de0c093fc19c

Revision 120a61ca - 2008-08-23 04:06 PM - Jürgen Fischer

various code cleanups by mloskot for GCC 4.1 (fixes #1239, #1240).

Inspired by it I also enabled pedantic mode for MSVC and cleaned up some more warnings found by GCC 4.3 and MSVC.

History

#1 - 2008-08-23 07:11 AM - Jürgen Fischer

- Resolution set to fixed
- Status changed from Open to Closed

Replying to [mloskot](#):

| I've prepared a patch for trunk/qgis subtree that enables more strict compilation with GCC 4.1 and cleans number of warnings.

Thanks for the patch I applied it in commit:120a61ca (SVN r9134). I didn't change the default compile, but the PEDANTIC mode. PEDANTIC also enables more warnings for the MSVC build. GCC 4.3 and MSVC identified some more problems. And the debian package is now compiled in PEDANTIC mode.

I got a bunch of warnings in Qt headers with strict aliasing (GCC 4.3.1 and Qt 4.4.0). So I didn't enable that.

| The only warning that hasn't been cleaned is

> warning: ISO C++ forbids casting between pointer-to-function and pointer-to-object

| but this is a well known issue of 'dlsym_ call used with C++ compiler (see #234).

I made up a cast_to_fptr function with the union approach there. Although that's not a (perfect) solution.

#2 - 2008-08-23 07:44 AM - Mateusz Loskot -

Jef,

Thanks for applying it. A few additional comments.

- You may also add [/Gz](#) and [/Za](#) for Visual C++ compiler.
 - cast_to_fptr does not make much sense. It cleans warning but it doesn't solve the problem and dlsym/QLibrary::resolve [still seriously breaks C++ rules](#)
: It's illegal, period.. Unless, functions being loaded are exported as C functions using *extern "C"*.
- I would suggest to enable this warning back, so it's not forgotten. C++ standarization committee has noticed this problem and [is working to find a solution](#). POSIX gurus have noticed that problem too and probably new version of dlsym will be released in future (see [Rationale](#))

#3 - 2008-08-23 07:51 AM - Mateusz Loskot -

One more note regarding the strict aliasing. Yes, I'd expect Qt breaks strict aliasing rules, however it's a good idea to remember it in case strange and undefined behavior occurs.

Recently, we have experienced such problem in GDAL. It was extremely hard to find out what's going on without enabled strict aliasing warnings. See the

story here: [XPlane causes undefined behavior with GCC 4.3.x](#)

If I'd be asked to decide, I did keep these warnings enabled, even if they occur in 3rd party libraries. We can't be sure Qt is free of bugs :-)

Files

compilation-warnings-and-cleanup-r9120.patch	45.5 KB	2008-08-22	Mateusz Loskot -
--	---------	------------	------------------