Audacious - OLD, PLEASE USE GITHUB DISCUSSIONS/ISSUES - Bug #509

GCC 5.0.0 build failure

February 15, 2015 18:21 - Michael Schwendt

Status:	Rejected	**Start date:**	February 15, 2015
Priority:	Minor	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
Affects version:	3.6		

Description

https://copr-be.cloud.fedoraproject.org/results/mschwendt/audacious-next/fedora-rawhide-x86_64/audacious-3.6-0.3.beta1.fc21/build.log

Haven't examined it yet, but new in Fedora Rawhide is GCC 5.0.0.

History

#1 - February 15, 2015 22:27 - John Lindgren

- Status changed from New to Rejected

Looks like a bug in GCC 5. The error is:

```
../libaudcore/templates.h:250:1: error: body of constexpr function 'constexpr
void (* aud::fill_func())(void*, int) [with T = String; aud::FillFunc = void
(*)(void*, int)]' not a return-statement
```

Look at templates.h and you will see that the body of aud::fill_func is most definitely a return statement.

Additionally, older GCC (4.7 through 4.9) and Clang compile the same code just fine.

#2 - February 17, 2015 16:09 - Michael Schwendt

The compiler seems to believe it's C++14, because a tiny test case I've created compiles with -std=c++14. Investigation ongoing.

And yes, Audacious 3.6-beta1 compiles for Fedora 21 (GCC 4.9.2 patched a lot).

#3 - February 17, 2015 20:11 - Michael Schwendt

https://gcc.gnu.org/bugzilla/show_bug.cgi?id=65075

#4 - February 18, 2015 20:25 - John Lindgren

C++14 loosens the restrictions on constexpr functions, but our code complies with the stricter C++11 rules, as far as I can tell. That GCC bug report does look like the same issue.

#5 - February 18, 2015 22:08 - Michael Schwendt

> That GCC bug report does look like the same issue.

Ack. It's a ticket opened by a Red Hat/Fedora compiler dev after I had asked about the issue.

#6 - February 19, 2015 04:50 - John Lindgren

I envy your contacts. I don't suppose you could convince this same GCC developer to take a look at #63707? We're working around it at the moment, but declaring ten separate variables when we ought to be using an array feels rather silly.

#7 - February 19, 2015 22:44 - Michael Schwendt

No "special" contacts. It's just that they have done a mass-rebuild of the entire dist with GCC 5 in search of regressions.

One issue with PR63707 could be that it's "Known to fail: 4.8.2, 4.9.1, 5.0", so it's broken for a longer time already.