

art - Bug #7006

ServiceCache/ServiceCacheEntry cause "incomplete type" errors on Clang

09/15/2014 08:06 AM - Ben Morgan

Status:	Closed	Start date:	09/15/2014
Priority:	Normal	Due date:	
Assignee:	Christopher Green	% Done:	100%
Category:	Infrastructure	Estimated time:	0.00 hour
Target version:	1.13.02	Spent time:	0.75 hour
Occurs In:		Experiment:	LBNE
Scope:	Internal	SSI Package:	art

Description

When building art using Clang (Apple 5.1, clang 3.4), an error is encountered in compiling ServiceRegistry:

```
art/Framework/Services/Registry/detail/ServiceCacheEntry.h:28:20: note:
    definition of 'art::detail::ServiceCacheEntry' is not complete until the
    closing '}'
class art::detail::ServiceCacheEntry {
```

This has been traced to ServiceCacheEntry having a data member of type ServiceCache::iterator. ServiceCache is a typedef to std::map<TypeID, detail::ServiceCacheEntry>, so even with forward declarations, ServiceCache::iterator is incomplete when used inside the definition of ServiceCacheEntry.

I'm aware that GCC allows it, and that this behaviour is compiler dependent, see for example

<http://stackoverflow.com/questions/10993720/forward-declaration-of-class-used-in-template-function-is-not-compiled-by-clang> .

I'm also aware that clang isn't officially supported, so I'm happy to patch and test if there is a preference for a suitable workaround (pointers/refs or reordering definitions).

Related issues:

Blocks art - Feature #8011: Support for Clang on OS X (Xcode/CLT) and Linux

Closed

03/03/2015

Associated revisions

Revision 5fc22fe0 - 03/23/2015 01:18 PM - Christopher Green

Proposed fix for issue #7006.

Revision a25e5691 - 04/13/2015 01:38 PM - Christopher Green

Fix for issue #7006.

History

#1 - 09/16/2014 05:10 PM - Christopher Green

- Category set to Infrastructure
- Status changed from New to Assigned
- Assignee set to Ben Morgan
- Target version set to 1.13.00
- Experiment LBNE added
- Experiment deleted (-)
- SSI Package art added
- SSI Package deleted ()

This certainly seems to be a case where using multiple compilers is a help to producing clean and correct code, notwithstanding the fact that, as you say, GCC does produce the answer one would want.

After having looked at the code and the implications of the clang diagnostic and the underlying standard violation, we understand that the iterator must be removed from the interface and data of ServiceCacheEntry. I think we would prefer the constructor which currently takes an iterator to take a ServiceCacheEntry & (caller would replace (e.g.) it with it->second). The private data member interface_impl would be a bare pointer, initialized to nullptr in the other constructors. No checks for pointer nullity would be necessary because is_interface() being true implies that the correct constructor

was called, as is the case currently.

If you were able to make these changes and verify them and the tests with clang, we'd happily review and take a patch.

In addition however, we would urge you to put in an issue requesting formal multi-platform support for clang compilation. That way, we can have the stakeholders assign a relative priority and then move toward providing a consistent version of clang across GNU/Linux and Mac OS X, and start providing the full suite as compiled for LLVM/Clang as a matter of course. We believe the language support is there now, and the optimizer is only going to get better.

Thanks for your work on this.

#2 - 01/26/2015 12:17 PM - Christopher Green

Since we have not yet received a patch, is this still a priority for the next release? We also have yet to receive a formal request for clang support.

#3 - 02/03/2015 01:07 PM - Christopher Green

- Target version changed from 1.13.00 to 1.14.00

#4 - 03/11/2015 02:32 PM - Kyle Knoepfel

- Status changed from Assigned to Accepted

- Target version deleted (1.14.00)

Since an official request has been made to add clang support for art, we will defer fixing this bug until issue [#8011](#) is addressed.

#5 - 03/11/2015 03:34 PM - Ben Morgan

- File 0001-Fix-type-recursion-error-on-Clang.patch added

Kyle Knoepfel wrote:

Since an official request has been made to add clang support for art, we will defer fixing this bug until issue [#8011](#) is addressed.

No problem, though the attached patch will do the job for this part of art (note that there are further Clang/C++ issues downstream of this, but one step at a time). The patch requires use of a Boost header-only component (http://www.boost.org/doc/libs/1_57_0/doc/html/container.html) but since art already depends on Boost I felt this was an acceptable solution. Boost.Container is also designed to workaround this issue in the standard (http://www.boost.org/doc/libs/1_57_0/doc/html/container/main_features.html#container.main_features.containers_of_incomplete_types).

As I'm not yet able to fully compile Art with Clang, I have been able to build the tests for this component. However, it does pass tests on Linux (SUSE gcc4.9 and RHEL devtoolset-3)

#6 - 03/12/2015 12:31 PM - Kyle Knoepfel

- Blocks Feature #8011: Support for Clang on OS X (Xcode/CLT) and Linux added

#7 - 03/23/2015 01:33 PM - Christopher Green

- File 0001-Proposed-fix-for-issue-7006.patch added

- Status changed from Accepted to Resolved

- Assignee changed from Ben Morgan to Christopher Green

- Target version set to 1.14.00

- % Done changed from 0 to 100

Due to your proposed fix being labelled as temporary, and the fact that it introduces a new dependence on code in Boost and a non-standard container that we would rather avoid for the long term, I went ahead and spent the time necessary to produce a fix along the lines of the one I proposed in an earlier comment. This fix has been committed as [5fc22fe](#) and attached as a formatted patch below. We'd be grateful if you could apply this locally and verify that the proposed fix resolves this reported issue with the Clang compiler. All code compiles and all tests pass on SLF6/e7.

#8 - 03/23/2015 02:25 PM - Christopher Green

- File 0002-Reverse-overeager-edit-that-resulted-in-unnecessary-.patch added

Reverse accidentally-introduced unconditional creation of service provider during interface handling (per [8b08b23](#)). See also attached patch.

#9 - 04/09/2015 03:29 PM - Christopher Green

- Target version changed from 1.14.00 to 1.13.02

#10 - 05/22/2015 05:53 PM - Christopher Green

- Status changed from Resolved to Closed

Files

0001-Fix-type-recursion-error-on-Clang.patch	1.46 KB	03/11/2015	Ben Morgan
0001-Proposed-fix-for-issue-7006.patch	5.37 KB	03/23/2015	Christopher Green
0002-Reverse-overeager-edit-that-resulted-in-unnecessary-.patch	1.73 KB	03/23/2015	Christopher Green