MAUS - Bug #1026

Installation issue - DataBaseAPI.cc and read/write

18 June 2012 20:20 - Pradhan, Neetish

Status:	Closed		**Start date:**	21 June 2012
Priority:	Normal		**Due date:**	
Assignee:	Rogers, Chris		**% Done:**	100%
Category:	Build System		**Estimated time:**	0.00 hour
Target version:	MAUS-v0.3.0			
Workflow:	New Issue			

Description

I tried installing MAUS on an Arch-Linux 64-bit system, and there seems to be an issue of read, write and close not being declared:

g++ -o src/legacy/Interface/DataBaseAPI.os -c -Wall -Dlong32='int' "-Dlong64='long long'" -O3 -m64 -pthread -rdynamic -fPIC -Ithird_party/install/include -Ithird_party/install/include/python2.7 -Ithird_party/install/include/root -Isrc/legacy -Isrc/common_cpp -Ithird_party/build/root_v5.30.03/include -Ithird_party/build/geant4.9.2.p04/include -I. src/legacy/Interface/DataBaseAPI.cc
src/legacy/Interface/DataBaseAPI.cc: In member function 'XMLMessage DataBaseAPI::sendQuery(const XMLMessage&)':
src/legacy/Interface/DataBaseAPI.cc:87:56: error: 'write' was not declared in this scope
src/legacy/Interface/DataBaseAPI.cc:101:43: error: 'read' was not declared in this scope
src/legacy/Interface/DataBaseAPI.cc:120:18: error: 'close' was not declared in this scope
src/legacy/Interface/DataBaseAPI.cc: In member function 'bool DataBaseAPI::openConnection()':
src/legacy/Interface/DataBaseAPI.cc:179:20: error: 'close' was not declared in this scope

Now I am not sure where C libraries store their functions, but the file "include"d sstream as a header file. sstream was in /third_party/build/root_v5.30.03/cint/cint/include/ which seemed to link to iostream.h in the same folder, in which I couldn't find any read or write functions. Any suggestions for a work-around?

Subtasks:

Bug # 1028: Incorrect calling of base functions fails to compile in gcc 4.8   **Closed**

History

#1 - 19 June 2012 08:42 - Rogers, Chris

- Category set to Build System

- Assignee set to Rogers, Chris

- Target version set to Future MAUS release

What gcc version do you have?

#2 - 19 June 2012 09:24 - Rogers, Chris

Looks like you can delete DataBaseAPI and DataBaseReader - they are no longer called (we now have the cdb python module to deal with this). I will remove them in the trunk.

#3 - 20 June 2012 05:26 - Pradhan, Neetish

Thanks, will try it without them and see how it goes. I've got gcc 4.7.

#4 - 20 June 2012 06:29 - Rogers, Chris

Latest we've tried so far is 4.6... let me know if there are problems.

#5 - 20 June 2012 10:43 - Rogers, Chris

- Status changed from Open to Closed

- % Done changed from 0 to 100

Committed in r747

Let me know if you find any further issues.

#6 - 20 June 2012 18:49 - Pradhan, Neetish

- File install_log_std added

Now it's produced a more obscure error, can't make anything of it - attaching the log. It seems to start with this:

./src/common_cpp/JsonCppProcessors/ArrayProcessors-inl.hh: In instantiation of 'std::vector<ArrayContents*>*
MAUS::PointerArrayProcessor<ArrayContents>::JsonToCpp(const Json::Value&) [with ArrayContents = MAUS::Trigger]':
./src/common_cpp/JsonCppProcessors/SpillProcessor.hh:42:7:   required from here
./src/common_cpp/JsonCppProcessors/ArrayProcessors-inl.hh:63:21: warning: deleting object of polymorphic class type 'MAUS::Trigger' which has non-virtual destructor might cause undefined behaviour [-Wdelete-non-virtual-dtor]

#7 - 21 June 2012 07:42 - Rogers, Chris

The warnings are actually not the problem. What happened here - when we run ROOT macro to make classes ROOT-aware (Rtypes ClassDef macro) ROOT sneakily adds virtual functions to the classes. So now we need a virtual destructor. My compiler didn't warn me about this, I only found out when I got a complaint. So I fixed it in the latest version.

The error is

```
g++ -o src/common_cpp/JsonCppProcessors/HitProcessor.os -c -Wall -Dlong32='int' "-Dlong64='long long'" -O3 -m6
4 -pthread -rdynamic -fPIC -Ithird_party/install/include -Ithird_party/install/include/python2.7 -Ithird_party
/install/include/root -Isrc/legacy -Isrc/common_cpp -Ithird_party/build/root_v5.30.03/include -Ithird_party/bu
ild/geant4.9.2.p04/include -I. src/common_cpp/JsonCppProcessors/HitProcessor.cc
In file included from src/common_cpp/JsonCppProcessors/HitProcessor.cc:17:0:
./src/common_cpp/JsonCppProcessors/HitProcessor.hh: In instantiation of 'void MAUS::HitProcessor<ChannelId>::R
egisterBranches() [with ChannelId = MAUS::SciFiChannelId]':
src/common_cpp/JsonCppProcessors/HitProcessor.cc:28:5:   required from here
./src/common_cpp/JsonCppProcessors/HitProcessor.hh:54:5: error: 'RegisterValueBranch' was not declared in this
 scope, and no declarations were found by argument-dependent lookup at the point of instantiation [-fpermissiv
e]
./src/common_cpp/JsonCppProcessors/HitProcessor.hh:54:5: note: declarations in dependent base 'MAUS::ObjectPro
cessor<MAUS::Hit<MAUS::SciFiChannelId> >' are not found by unqualified lookup
./src/common_cpp/JsonCppProcessors/HitProcessor.hh:54:5: note: use 'this->RegisterValueBranch' instead
```

Let me have a dig...

#8 - 21 June 2012 08:01 - Rogers, Chris

nb: you can just run source env.sh; scons to build now - the third party libraries have all built okay I think so no need to go through all that again (scons takes ~5 minutes to compile everything, otherwise you have to rebuild G4 and ROOT which takes ~ 1 hour). Run the tests using tests/run_tests.bash

#9 - 21 June 2012 08:05 - Rogers, Chris

- Status changed from Closed to Open

- % Done changed from 100 to 0

#10 - 21 June 2012 14:43 - Rogers, Chris

- Status changed from Open to Closed

- % Done changed from 0 to 100

#11 - 02 July 2012 09:18 - Rogers, Chris

- Target version changed from Future MAUS release to MAUS-v0.3.0

Files

install_log_std	4.24 MB	18 June 2012	Pradhan, Neetish
install_log_std	5.01 MB	20 June 2012	Pradhan, Neetish