

MAUS - Support #433

Code review for MiceModuleConfigDB stuff

03 May 2011 17:24 - Rogers, Chris

Status:	Closed	Start date:	03 May 2011
Priority:	Normal	Due date:	13 May 2011
Assignee:	Rogers, Chris	% Done:	100%
Category:	common_cpp	Estimated time:	0.00 hour
Target version:	MAUS-v0.0.3		
Workflow:			
Description			
Okay, G4MICE not MAUS but I will miss this tracker if it goes into G4MICE			

History

#1 - 10 May 2011 18:19 - Tunnell, Christopher

- Target version changed from MAUS-v0.0.1 to Future MAUS release

#2 - 11 May 2011 08:43 - Rogers, Chris

New files:

G4MICE/Config/include/MiceModuleTreeBuilder.hh
G4MICE/Config/src/MiceModuleTreeBuilder.cc
G4MICE/Config/include/MiceModuleConfigDBIO.hh
G4MICE/Config/src/MiceModuleConfigDBIO.cc
G4MICE/GDML_3_0_0

Also changes in:

G4MICE/Config/include/MiceModule.hh
G4MICE/Config/src/MiceModule.cc
G4MICE/Config/src/dataCards.cc

Documentation:

<http://micewww.pp.rl.ac.uk/projects/g4mice/wiki/GDML>

#3 - 11 May 2011 08:45 - Rogers, Chris

Related issues:

Feature #114
Feature #58

#4 - 11 May 2011 10:02 - Tunnell, Christopher

There's a 'related to' thing at the top of the page. Who do you want to look at this? I can spend an hour if you want peaking at it.

#5 - 11 May 2011 12:47 - Rogers, Chris

Also code in
Applications/Geometry

Nb: makefile doesn't compile any executable here. What is it supposed to do? Perhaps forgot to commit something?

#6 - 11 May 2011 15:55 - Reid, Ivan

Make works for me, within the directory. The link step is included with the compile step to get more than one executable compiled; that may confuse some scripted builds.

Don't forget the code in Applications/Simulation too; see <http://micewww.pp.rl.ac.uk/projects/g4mice/wiki/MiceModuleConfigDBIO>.

#7 - 11 May 2011 16:58 - Rogers, Chris

Maybe I'm being stupid. From \$MICESRC/Applications/Geometry I do

```
cvs -q update -Ad
make
ls
```

and list the following files

```
cards.in  dumpcardsgdml.in  dumpcurrentgdml.in  dumpIDgdml.in  DumpMiceModuleGdml.cc  dumprungdml.in  gdmldump
.in      makefile  NewTestGdmlDump.sh  TestGdmlDump.sh
CVS     dumpcards.in  dumpcurrent.in  dumpID.in  DumpMiceModuleTree.cc  dumprun.in  GetGeome
tryIDs.cc  makefile~  StoreGDMLtoDB.cc
```

No executable there. Looking at the shell scripts, they seem to expect an executable to appear. Looking at the makefile, the exe bit says something like:

```
exe : $(mgrobj)
      #$(CXX) -o /home/mice/MICE//Applications/Geometry/GetGeometryIDs $(mgrobj) $(LDFLAGS) -L${MICE_PATH}/${COM
PILER}/compiler/lib
```

which is just all commented out and doesn't have the appropriate exe names in anyway. Am I doing something wrong?

#8 - 11 May 2011 17:54 - Reid, Ivan

Strange! I do

```
make clean
make <gives pages of compile/link messages>
Applications/Geometry> ls
cards.ckov          dumpID.in          DumpRun.out
cards.in            DumpID.out         DumpRun.out.old
CVS                 DumpID.out.old    gdmldump.in
DumpCards.gdml     DumpMiceModuleGdml  GDMLDump.out
dumpcardsgdml.in  DumpMiceModuleGdml.cc  GetGeometryIDs
dumpcards.in       DumpMiceModuleGdml.d  GetGeometryIDs.cc
DumpCards.out      DumpMiceModuleGdml.o  GetGeometryIDs.d
DumpCards.out.old  DumpMiceModuleTree  GetGeometryIDs.o
DumpCurrent.gdml   DumpMiceModuleTree.cc  makefile
dumpcurrentgdml.in  DumpMiceModuleTree.cc.old  NewTestGdmlDump.sh
dumpcurrent.in     DumpMiceModuleTree.d  StoreGDMLtoDB
DumpCurrent.out    DumpMiceModuleTree.o  StoreGDMLtoDB.cc
DumpCurrent.out.old  DumpRun.gdml       StoreGDMLtoDB.d
DumpID.gdml        dumprungdml.in     StoreGDMLtoDB.o
dumpIDgdml.in     dumprun.in         TestGdmlDump.sh
Applications/Geometry> cvs diff makefile
Applications/Geometry>
```

The link step is combined with the compilation:

```
$(Pkgmgr)/%.o: $(Pkgmgr)/%.cc
      #$(CXX) $(CPPFLAGS) -I$(Pkgmgr) $(CXXFLAGS) -c -o $@ $<
      #$(CXX) -o $(basename $@) $@ $(LDFLAGS) -L${MICE_PATH}/${COMPILER}/compiler/lib
```

#9 - 12 May 2011 08:23 - Rogers, Chris

Ah... found it. You hardcoded line 32

```
Pkgmgr = /home/mice/MICE/Applications/Geeomtry
```

Should be

```
Pkgmgr = $(MICESRC)/Applications/Geeomtry
```

If we were sticking with this build structure I would ask you to change the AppMake script - but as we are moving to scones I think it is a waste of time. I committed the patch to cvs head.

#10 - 12 May 2011 09:52 - Reid, Ivan

Ah! Sorry 'bout that. I find that make script unnecessarily opaque, good to hear you're getting shot of it.

#11 - 12 May 2011 10:12 - Rogers, Chris

GDML_3_0_0

BUG - \$MICESRC/GDML_3_0_0/object does not exist in cvs -> causes make to fail. Otherwise okay. As this is auto generated I'm not going to review it...

The build system will need to be revised when we move to MAUS anyway.

MiceModuleTreeBuilder

Summary: Uses datacards to decide how MiceModules should be built and calls relevant functions from MiceModuleConfigDBIO

Reasonable commenting and clear code. Few quibbles:

- No copyright notice. Forgot or perhaps there is a reason?
- doxygen fails to build this class
- No test...
- Fails cpplint for some stuff that can be trivially fixed

MiceModuleTreeBuilder.cc -

- Line 7-19: do we really need all these includes?
- Line 25: stupid thing, but the geometry source for all the options is MiceModel datacard. So this should be

```
if (GeometrySource.compare("MiceModuleTextFile") == 0) {
```

- Line 26/27: Stupid thing but why do you split the line half way through a word(!)
- Line 35: why use variable name "foo"?
- Line 52: Option should be "GDMLTextFile"

MiceModuleConfigDBIO

Two bugs:

- BUG... If http://service-spi.web.cern.ch/service-spi/app/releases/GDML/GDML_3_0_0/schema/gdml.xsd cannot be reached we can't generate a schema? So we can't take data? In any case URL should be a datacard.
- BUG: for config DB write_URL = read_URL + "SuperMouse"; Should be a datacard (URLs change...) so we want two datacards, one is read url and one is write url (just in case we end up with two separate servers).

Summary: does two things really;

- first manages conversion between MiceModules and GDML (using GDML_3_0_0 library) ~ 350 lines code
- second manages interface with ConfigDB and network communications ~ 300 lines code
- comments aren't thorough enough. Need to explain what each parameter is and what the return value is e.g. GetGeometryIDs(std::string * IOVTime1, std::string * IOVTime2); need to explain that IOVTime2==NULL means "present", and what is the return value (integer ID as string?)
- No copyright notice. Forgot or perhaps there is a reason?
- doxygen fails to build this class
- Fails cpplint for some stuff that can be trivially fixed

The timestamp stuff are general algorithms so should go e.g. in STLUtils (which is supposed to be a dumping ground for exactly these sort of general algorithms). Looks correct but should certainly be unit tested.

Next: look at test coverage

#12 - 12 May 2011 11:59 - Rogers, Chris

- File TestGdmlDump_sh_output added
- File NewTestGdmlDump_sh_output added
- File g4mice_unit_test_output added

All of the tests fail for me with error message

```
terminate called after throwing an instance of 'xsd::cxx::tree::parsing<char>'
what(): instance document parsing failed
```

Full log attached.

I am working inside RAL firewall, perhaps a proxy issue?

#13 - 12 May 2011 14:52 - Rogers, Chris

I think I was doing something stupid - trying to run the scripts from inside tcsh so it didn't see environment variables correctly.

Looks like it works okay. Very nice. I like the robustness to errors (I tried making a few typos etc and it gave informative and correct error messages).

Again though, you know what I'm going to say - there is no regression test for this code. In 2 years time will it still work? Probably not. More to the

point, will we know it still works? Certainly not.

Would be really nice to put some stuff in \$MICETESTS/Integration/ConfigDB area that:

- Checks for read/write to test DB
 - Checks formatting to GDML and MiceModule as appropriate
- Checks for correct error handling (malformed date, malformed run ID)

#14 - 12 May 2011 15:01 - Rogers, Chris

So summary:

1. Very nice. I like the robustness to errors (I tried making a few typos etc and applications gave informative and correct error messages). The code works.
2. Comments need to be improved. Try, for example, defining parameters and return values for each function (unless really trivial) as described in [Coding style](#)
3. Would be good to improve cpplint compliance.
4. Unit test coverage needs to be improved. Unit tests need to be made more reliable.
5. Integration tests required

Do you (Ivan) have time to implement this stuff?

#15 - 21 May 2011 16:06 - Reid, Ivan

Reacting to several of the above:

- make problem has been sorted
- problem with .sh scripts may be because I used them with "bash <script>"; other methods of invocation may give different results
- copyright notices omitted as I wasn't sure what to use; have now used "2011 MICE Collaboration"
- doxygen -- I don't know why doxygen doesn't pick up on my stuff. I've obviously got something "not quite right" but the one time I tried to use doxygen on it it gave no hint of what it found wrong
- cpplint -- I've re-linted the latest versions. I've ignored some constraints I find ludicrous, e.g. no more than one statement per line, where I think there's good reason to do so. The original splitting of a word mid-string in an output statement was due to cpplint's objecting to the line in a previous version being (just) over 80 characters long...
- MiceModuleTreeBuilder is inextricably bound with MiceModule -- its tests are part of the tests for MiceModuleConfigDBIO
- MiceModuleTreeBuilder.cc includes -- turns out the Event wasn't needed
- foo -- sometimes you get tired of thinking up trivial names; code originated elsewhere
- schema -- if schema file is inaccessible/invalid the schema used to build the library is used. To be honest, I've no idea what happens if a valid, but different, schema is specified. This now uses a dataCard, which moves the cpplint error from MiceModuleConfigDBIO.cc to dataCards.cc...
- Changed to using dataCards for the ConfigDB URLs, so now the write DB can be different from the read-only DB. Note that both instances should use exactly the same interface except for the write routine! Note also that getCDBsource.bash still uses an environment variable to specify the URL of the WSDL
- added some comments to clarify the arguments to GetGeometryIDs() method; the API only has one method, with two arguments, so I overloaded it with methods also having zero or one arguments, which call the two-argument method appropriately
- using dataCards for the URLs necessitates a prompt for a dataset in the GetGeometryIDs app; if a blank line is entered the test DB is used by default
- moved the TimeStamp routines to STLUtils; added simple tests to the unit testing
- added a few more tests, including writing to the test DB. Now have a couple of tests for read failures as well, due to a slight glitch by the DB manager...

And, really, that's about it. My time is filling up with my new responsibilities and I need to concentrate 100% on them. Still available for consultation and guidance of grad students but otherwise: So long, and thanks for all the fish!

ivan

#16 - 27 May 2011 11:49 - Rogers, Chris

- Status changed from Open to Closed

- % Done changed from 0 to 100

Well I think this is closed out... thanks for all the hard work ivan.

#17 - 23 June 2011 17:27 - Rogers, Chris

- Target version changed from Future MAUS release to MAUS-v0.0.3

Files

TestGdmlDump_sh_output	3.14 KB	12 May 2011	Rogers, Chris
NewTestGdmlDump_sh_output	3.11 KB	12 May 2011	Rogers, Chris
g4mice_unit_test_output	7.89 KB	12 May 2011	Rogers, Chris