MAUS - Bug #1749

Can't run C++ reducers in the same instance as mappers

18 September 2015 16:11 - Greis, Jan

Status:	Open	**Start date:**	18 September 2015
Priority:	Normal	**Due date:**	
Assignee:	Dobbs, Adam	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
Workflow:	New Issue		

Description

When trying to run a reducer directly after mappers, the following error pops up:

```
def process(self, *args): return _ReduceCppReconTesting.ReduceCppReconTesting_process(self, *args)
TypeError: in method 'ReduceCppReconTesting_process', argument 2 of type 'std::string'
```

Reducers still work when processing data from a run that went straight from mapper to outputter. On Durga's suggestion I removed the .i file from the reducer, this changed the error message to which can't interpret.

```
ImportError: dynamic module does not define init function (init_ReduceCppGlobalReconEfficiency)
```

History

#1 - 21 September 2015 09:05 - Rogers, Chris

Jan, you need to add to your ReduceCppGlobalReconEfficiency.cc file:

```
PyMODINIT_FUNC init_ReduceCppGlobalReconEfficiency(void) {
  PyWrapReduceBase<MAUS::ReduceCppGlobalReconEfficiency>::PyWrapReduceBaseModInit
                              ("ReduceCppGlobalReconEfficiency",
                               class_docstring,
                               birth_docstring,
                               process_docstring,
                               death_docstring);
}
```

When you do import my_module in python, the python interpreter looks for a file like my_module.so and calls the function init_my_module(void). MAUS handles

- building my_module.so (done by the compile scripts)
- module and class initialisation (done by PyWrapReduceBase<>::PyWrapReduceBaseModInit); in PyWrapReduceBase we tell python interpreter what all of the functions are in the Reducer

But you just need to join those bits together as above.

#2 - 21 September 2015 09:05 - Rogers, Chris

Also, make sure there are no .py files in your build directory. If you are at RAL I can swing by and have a look.

#3 - 21 September 2015 11:06 - Greis, Jan

Thanks, Chris. I've added it in (adjusted to ReduceCppReconTesting which I'm working with right now), but now I get this error:

```
src/reduce/ReduceCppReconTesting/ReduceCppReconTesting.cc: In function `void MAUS::init_ReduceCppReconTesting(
)':
src/reduce/ReduceCppReconTesting/ReduceCppReconTesting.cc:57:32: error: `class_docstring' was not declared in
this scope
                               class_docstring,
                               ^
src/reduce/ReduceCppReconTesting/ReduceCppReconTesting.cc:58:32: error: `birth_docstring' was not declared in
this scope
                               birth_docstring,
                               ^
src/reduce/ReduceCppReconTesting/ReduceCppReconTesting.cc:59:32: error: `process_docstring' was not declared i
n this scope
                               process_docstring,
```

```
                                ^
src/reduce/ReduceCppReconTesting/ReduceCppReconTesting.cc:60:32: error: `death_docstring' was not declared in
this scope
                                death_docstring);
```

Sorry, not at RAL at the moment.

#4 - 21 September 2015 11:28 - Rogers, Chris

Sorry, I should have pointed out that you need to define the docstrings. I think there is a noddy default if you just put in "".