

## MAUS - Bug #1776

### Sample reducer ReduceCppTOFPlot segfaults in MAUS 1.2.0

15 October 2015 18:18 - Greis, Jan

<b>Status:</b>	Open	<b>Start date:</b>	15 October 2015
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Karadzhov, Yordan	<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			
<b>Workflow:</b>	New Issue		

#### Description

Error confirmed by Adam.  
output:

```
#5  0x00007fde0eaa4454 in MAUS::ReduceCppTOFPlot::_birth(std::string const&) () from /home/jan/MICE/MAUS/maus-globalrecon/build/_ReduceCppTOFPlot.so
#6  0x00007fde0eaa5b01 in MAUS::PyWrapReduceBase<MAUS::ReduceCppTOFPlot>::_birth(_object*, _object*, _object*) () from /home/jan/MICE/MAUS/maus-globalrecon/build/_ReduceCppTOFPlot.so
#7  0x00007fde2d7ccc2c in call_function (oparg=<optimised out>, pp_stack=0x7ffd86508d10) at Python/ceval.c:4013
#8  PyEval_EvalFrameEx (f=f
entry=0x4f2b5e0, throwflag=throwflag
entry=0) at Python/ceval.c:2666
#9  0x00007fde2d7ccb1 in fast_function (nk=<optimised out>, na=2, n=2, pp_stack=0x7ffd86508e90, f
unc=0x2f00f50) at Python/ceval.c:4099
#10 call_function (oparg=<optimised out>, pp_stack=0x7ffd86508e90) at Python/ceval.c:4034
```

#### History

##### #1 - 16 October 2015 10:51 - Rogers, Chris

- Assignee changed from Rogers, Chris to Karadzhov, Yordan

Actually this is Yordan's code. But reading it, I wonder whether it is safe running over multiple runs. In particular, the TH1\* is deleted but the TCanvas\* is not (that I can see, I didn't read in detail). Does TCanvas know not to try to print a deleted TH1? Does CanvasWrappersPushBack() delete any existing CanvasWrappers (probably not). So on the second run, I guess whatever outputter tries to write out TCanvas from the first run, but writes out free'd data and dies. Maybe this is handled in the base class (maybe there is an output->GetCanvasWrappers()->resize() called in base class birth?)

I didn't read in detail, so apologies if I am incorrectly criticising.