ctools - Bug #1254

Problem in ctbin.save() for multiple pointings

07/09/2014 11:25 AM - Martin Pierrick

Status:	Closed		**Start date:**	07/09/2014
Priority:	Normal		**Due date:**	
Assigned To:	Martin Pierrick		**% Done:**	50%
Category:			**Estimated time:**	1.00 hour
Target version:	3rd coding sprint			

Description

Saving data cubes in the case of multiple observations in an observations container does not seem to work when working from python. The ctbin.run() process stops without files being written and without error or warning.

Possible problem is: the file names are built from:
std::string outfile = set_outfile_name(m_infiles[i]);
std::vector<std::string> elements = gammalib::split(filename, "/");
std::string outname = m_prefix + elements[elements.size()-1];
The problem is that when working from python, there are not necessarily input file names and so variable m_infiles[] is empty. So the output file name should consist in only the prefix plus content of memory in elements[-1], which can be anything.

So not obvious why it does not work since the outname variable should contain a string. In any case, code should be adapted to deal properly with the case where no input file names exist.

History

#1 - 07/09/2014 01:53 PM - Mayer Michael

I agree with that. I recently encountered the same problem. As a quick and dirty solution, the problem could be circumvented by adding something like

m_infiles[i] = "events_"+obs->id()+".fits";

at line 853 of ctbin.cpp. If you have e.g. 200 observation, this will however lead to 200 individual files. These files (cubes) should be be stacked and removed afterwards.
With the operator+= of GSkymap, this should be easily possible I guess. At some point, the stacking should be done internally.

#2 - 07/09/2014 06:01 PM - Martin Pierrick

- % Done changed from 0 to 50

I have created a modified version that seems to fix the problem:

In ctbin::save_xml
std::string outfile = set_outfile_name(i);

In ctbin::set_outfile_name:
std::string outname="";
if (elements.size() > 0) {
outname = m_prefix + elements[elements.size()-1];
} else {
outname = m_prefix + m_obs[index]->id() + ".fits";
}

I also now only call save_xml() from save() so that an obs.xml file is created even when there is only one observation in the container...

If we agree on that, I can create a branch, add that, and request for pull...

#3 - 07/09/2014 11:05 PM - Knödlseder Jürgen

Mayer Michael wrote:

> I agree with that. I recently encountered the same problem. As a quick and dirty solution, the problem could be circumvented by adding something like
> [...]
> at line 853 of ctbin.cpp. If you have e.g. 200 observation, this will however lead to 200 individual files. These files (cubes) should be be stacked and removed afterwards.
> With the operator+= of GSkymap, this should be easily possible I guess. At some point, the stacking should be done internally.

I agree, there should be an option to stack the counts maps, which is part of implementing a cube-style analysis. Chia-Chun is actually working on that. I created action #1260 to keep track of the request.

#4 - 07/09/2014 11:16 PM - Knödlseder Jürgen

- Assigned To set to Martin Pierrick

Martin Pierrick wrote:

> I have created a modified version that seems to fix the problem:
>
> In ctbin::save_xml
> std::string outfile = set_outfile_name(i);
>
> In ctbin::set_outfile_name:
> std::string outname="";
> if (elements.size() > 0) {
> outname = m_prefix + elements[elements.size()-1];
> } else {
> outname = m_prefix + m_obs[index]->id() + ".fits";
> }
>
> I also now only call save_xml() from save() so that an obs.xml file is created even when there is only one observation in the container...
>
> If we agree on that, I can create a branch, add that, and request for pull...

Can you make an issue branch with the modifications?

Concerning writing always an XML file and event lists, I was wondering whether in the long run we need this. I would expect that in the long run we will sum events from multiple observations into a single cube to take benefit from binning. Then, ctbin would only have a single event file as output.

Maybe it would be better to add a specific parameter to specify the name of the output descriptor. We could have

outodf, f, a, "cntmap.xml",,, "Output observation definition file"
outfile, f, a, "cntmap.fits",,, "Output counts map"

and ctbin would then use whatever it needs.

#5 - 07/10/2014 06:12 PM - Martin Pierrick

- Status changed from New to Pull request

- Priority changed from High to Normal

I created a branch for that and uploaded the small modifications that solve the problem.
Can be merged to the code since we were at least two to be annoyed by that problem. On a longer term, I agree that we may have only a single summed cube for a typical analysis, but we may want to keep the separate file option which can be useful for debugging.

#6 - 07/10/2014 11:21 PM - Knödlseder Jürgen

- Status changed from Pull request to Resolved

- Target version set to 3rd coding sprint

Merged into trunk.

#7 - 07/20/2014 11:16 PM - Knödlseder Jürgen

- Status changed from Resolved to Closed