ctools - Bug #1836

ctobssim doesn't write obs_id attribute into FITS header

08/03/2016 11:15 AM - Mayer Michael

Status:	Closed	**Start date:**	08/03/2016
Priority:	Normal	**Due date:**	
Assigned To:	Knödlseder Jürgen	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:	1.5.0		

Description

When running a sequence of csobsdef and ctobssim using several observations, the simulated observation XML file contains incremental observation IDs (e.g "00001", "00002", etc...). However, looking into the actual FITS event lists (sim_events_1.fits, sim_events_2.fits, ...), the header keyword "OBS_ID" is zero in every case.

I have looked through the code and think it might be an issue on the GammaLib side:
in GObservations::read, we read the XML file and set the GObservation::id() with what is found in the XML file. This information is however not ported to GCTAObservation::obs_id. The latter is only loaded when reading an actual FITS file. But it is not load when setting up the observation with an XML file in the format like this:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<observation_list title="observation list">
  <observation name="None" id="000001" instrument="CTA">
   <parameter name="Pointing" ra="83.63" dec="22.01" />
   <parameter name="EnergyBoundaries" emin="200000" emax="3e+07" />
   <parameter name="GoodTimeIntervals" tmin="0" tmax="20" />
   <parameter name="TimeReference" mjdrefi="51544" mjdreff="0.5" timeunit="s" timesys="TT" timeref="LOCAL" />
   <parameter name="Deadtime" deadc="0.95" />
  </observation>
  <observation name="None" id="000002" instrument="CTA">
   <parameter name="Pointing" ra="83.63" dec="21.01" />
   <parameter name="EnergyBoundaries" emin="400000" emax="2e+07" />
   <parameter name="GoodTimeIntervals" tmin="20" tmax="40" />
   <parameter name="TimeReference" mjdrefi="51544" mjdreff="0.5" timeunit="s" timesys="TT" timeref="LOCAL" />
   <parameter name="Deadtime" deadc="0.95" />
  </observation>
</observation_list>
```

I would expect the following code returning not 0 but 1:

```
>>> obs = gammalib.GObservations("obs.xml") # obs.xml contains the observation definition above
>>> obs[0].obs_id()
0
```

I see it is difficult since GObservation::m_id is a string while GCTAObservation::m_obs_id is an integer. Maybe this is a feature that could also be taken care of on the ctools side in ctobssim, setting the obs_id explicitly.

Any thoughts?

History

#1 - 08/03/2016 09:52 PM - Knödlseder Jürgen

We had this already some time I ago (see #1445). These are two distinct informations that have not necessarily to do with each other. The purpose of the XML ID is to link an observation to a model. The OBS_ID is an OGIP standard (see https://heasarc.gsfc.nasa.gov/docs/heasarc/ofwg/docs/general/ogip_94_001/ogip_94_001.html) and is a sequence number.

But by reading the OGIP document again, I see that it says it's a string. In Karl's event format document this is an integer, and also on https://gamma-astro-data-formats.readthedocs.io/en/latest/events/events.html it's an integer (well, I wrote that part of the document).

We could (and probably should) transform it to a string. But there is still the problem that you could image that two event files have the same OBS_ID value, but in the XML file the id's must be unique. I see no good way to make both things identical.

#2 - 08/04/2016 09:51 AM - Mayer Michael

Thanks for pointing out the previous discussion - I forgot about this.
Nevertheless, just from a user perspective, I think it would be good to increment the m_obs_id attribute for each observation in ctobssim (unless it is already provided in an input observation container). Having all obs_id as zeros in the simulated FITS files is a bit confusing, right?
In case no obs_id is given e.g. like in the above XML example, we could start from obs_od=1 and increment by one for each observation.
Alternatively, we could leave it as it is and just expand the above format by an optional XML element:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<observation_list title="observation list">
  <observation name="None" id="000001" instrument="CTA">
    <parameter name="Pointing" ra="83.63" dec="22.01" />
    <parameter name="EnergyBoundaries" emin="200000" emax="3e+07" />
    <parameter name="GoodTimeIntervals" tmin="0" tmax="20" />
    <parameter name="TimeReference" mjdrefi="51544" mjdreff="0.5" timeunit="s" timesys="TT" timeref="LOCAL" />
    <parameter name="Deadtime" deadc="0.95" />
    <parameter name="ObsID" obs_id="1" />
  </observation>
  <observation name="None" id="000002" instrument="CTA">
    <parameter name="Pointing" ra="83.63" dec="21.01" />
    <parameter name="EnergyBoundaries" emin="400000" emax="2e+07" />
    <parameter name="GoodTimeIntervals" tmin="20" tmax="40" />
    <parameter name="TimeReference" mjdrefi="51544" mjdreff="0.5" timeunit="s" timesys="TT" timeref="LOCAL" />
    <parameter name="Deadtime" deadc="0.95" />
    <parameter name="ObsID" obs_id="2" />
  </observation>
</observation_list>
```

Accordingly, csobsdef input ASCII files could contain a column with obs_id, too (of course optional, we could leave it zero if none is provided).

#3 - 08/04/2016 11:05 AM - Knödlseder Jürgen

The point is that OBS_ID is set by the provider of the data, we formally don't have control over that. The purpose of the obs_id() method is to return simply the attribute that was set by the data provider.

But I agree that for ctobssim, which is in fact a data provider, we should increment the m_obs_id as you suggest. I simply did not care so far about the content of that attribute smile.png

Would this be sufficient, or are there cases where you want to specify this in the XML file?

#4 - 08/04/2016 11:10 AM - Mayer Michael

Would this be sufficient, or are there cases where you want to specify this in the XML file?

I think it would be sufficient to just increment by one. However, adding the option in the XML file probably allows for more flexibility.
The reason I stumbled upon this was because I was creating a script to simulate a IACT database with more realistic obs_id values (e.g. OBS_ID=28001). So I had to set the OBS_ID manually in the script (which is not a big deal). I am not sure if there is indeed another use case why we should have it in the XML format.

#5 - 08/04/2016 10:54 PM - Knödlseder Jürgen

Then I would prefer to implement the ctobssim incrementation of that value. Probably we want to add a hidden obsid parameter that would be taken as value of the first observation. This will provide full control over the obsid assignment, for example in case that multiple ctobssim runs will be combined.

There is still the other pending question on whether to use an integer or a string. I would suggest a string. But before changing this for GammaLib I'd like to discuss this with the data formats group. I just created an issue at
https://github.com/open-gamma-ray-astro/gamma-astro-data-formats/issues/64.

We may also decide to keep for the moment the integer interface (which is more natural for a value that will be incremented), but use a string when writing to the FITS file.

#6 - 10/06/2016 04:40 PM - Knödlseder Jürgen

- Target version set to 1.2.0

#7 - 03/03/2017 10:32 AM - Knödlseder Jürgen

- Target version changed from 1.2.0 to 1.3.0

#8 - 06/07/2017 05:44 PM - Knödlseder Jürgen

- Target version changed from 1.3.0 to 1.4.0

#9 - 08/01/2017 09:48 AM - Knödlseder Jürgen

- Target version changed from 1.4.0 to 1.5.0

#10 - 01/22/2018 02:54 PM - Knödlseder Jürgen

- Status changed from New to In Progress

- Assigned To set to Knödlseder Jürgen

- % Done changed from 0 to 10

The GCTAObservation::obs_id() methods were removed since the observation identifier is now stored in the generic GObservation::id() attribute.

#11 - 01/22/2018 04:34 PM - Knödlseder Jürgen

- % Done changed from 10 to 50

ctobssim now sets the observation identifier for each observation that has an empty observation identifier string.

#12 - 01/23/2018 12:21 AM - Knödlseder Jürgen

- Status changed from In Progress to Closed

- % Done changed from 50 to 100

The code was merged into devel.