

GammaLib - Change request #749

Move instrument, ontime, livetime and deadc attributes to GObservation base class

02/06/2013 09:43 PM - Knödlseider Jürgen

Status:	New	Start date:	02/06/2013
Priority:	Normal	Due date:	
Assigned To:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			

Description

All classes that are derived from GObservation implement the attributes

- instrument name
- ontime
- livetime
- deadtime correction factor

It is therefore reasonable to move these attributes in the base class and to add methods for setting and retrieving these attributes. They can then be removed from the instrument specific observation classes.

History

#1 - 01/08/2014 12:06 PM - Deil Christoph

While doing this, would it be possible to clarify what deadc is (and maybe choose a more explicit name)?

Is it a time? Or a correction factor?

In HESS we call this the "deadtime correction factor":

$$t_live = t_obs * (1 - deadtime_correction_factor)$$

and this the "deadtime":

$$t_dead = deadtime_correction_factor * t_obs$$

In GCTAObservation I see deadc twice:

```
virtual double      deadc(const GTime& time) const;
void                deadc(const double& deadc);
```

So deadc is used as "deadtime correction factor" and "deadtime" in different places?

#2 - 01/08/2014 06:04 PM - Knödlseider Jürgen

Deil Christoph wrote:

While doing this, would it be possible to clarify what deadc is (and maybe choose a more explicit name)?

deadc is the OGIP name, and indeed, it's the deadtime correction factor.

Is it a time? Or a correction factor?

In HESS we call this the "deadtime correction factor":

$t_{\text{live}} = t_{\text{obs}} * (1 - \text{deadtime_correction_factor})$

and this the "deadtime":

$t_{\text{dead}} = \text{deadtime_correction_factor} * t_{\text{obs}}$

It's exactly this.

In GCTAObservation I see deadc twice:

[...]

So deadc is used as "deadtime correction factor" and "deadtime" in different places?

The first method is to retrieve the value as function of time, the second is to set it as a constant. This is not very consistent, but only the first is imposed by the interface (i.e. in general, deadc may be time dependent) while the second is CTA specific, where deadc is a constant for a given observation (deadc is actually a keyword in Karl's event list format).