GammaLib - Bug #2776

event time bug from gammalib 1.4.0 to 1.5.2

12/14/2018 09:11 PM - Kelley-Hoskins Nathan

Status:	Closed	**Start date:**	12/14/2018
Priority:	Normal	**Due date:**	
Assigned To:	Knödlseder Jürgen	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			

Description

Event times (GCTAEventAtom::time()) are bugged in gammalib 1.5.2. They were correct in 1.4.0 .

The code to reproduce this is:

```
from gammalib import GObservations, GCTAObservation

obs = GObservations()
ob = GCTAObservation()
ob.load('VR50990.fits')
ob.id('VR50990')
obs.append(ob)

for ob in obs :
  for ev in ob.events() :
    print(ev.time().utc())
```

On gammalib 1.4.0, when I read in the Veritas datafile VR50990.fits, it prints:

```
2010-04-14T10:14:47
2010-04-14T10:14:48
2010-04-14T10:14:48
2010-04-14T10:14:50
2010-04-14T10:15:00
2010-04-14T10:15:01
2010-04-14T10:15:04
```

which are the correct times for those events.

When I take the same file, and run that script on gammalib 1.5.2, the same events are at times:

```
2003-12-15T22:14:49
2003-12-15T22:14:50
2003-12-15T22:14:50
2003-12-15T22:14:52
2003-12-15T22:15:02
2003-12-15T22:15:03
2003-12-15T22:15:06
```

which are a few years before VERITAS was built :p .

Related issues:

Related to ctools - Change request # 2054: MJD reference date in the output f...	**Closed**	**04/28/2017**

History

#1 - 12/21/2018 11:58 AM - Knödlseder Jürgen

Does the FITS file contain the reference time key words?

#2 - 12/28/2018 01:30 AM - Kelley-Hoskins Nathan

- File VR81417.chunk3.clean.fits added

The EVENTS table has the time-related header keywords:

```
MJDREFI     53856
MJDREFF      0.0
TIMEUNIT    s
TIMESYS     TT
TIMEREF     local
```

And events have columns like:

```
OBS_ID EVENT_ID TIME
81417  247243   313216622.2168968
81417  247649   313216623.7063196
81417  248645   313216627.4837822
81417  248931   313216628.5783541
81417  249323   313216630.0244226
```

I've added the fits file these came from, which should have events on 2016-04-03, from 4:21-4:51 am UTC.

#3 - 01/09/2019 03:46 PM - Knödlseder Jürgen

- Status changed from New to Feedback

- Assigned To set to Knödlseder Jürgen

Running your example file using the current code I get:

```
>>> import gammalib
>>> obs=gammalib.GCTAObservation('VR81417.chunk3.clean.fits')
>>> print(obs)
=== GCTAObservation ===
 Name .......................:
 Identifier .................: 81417
 Instrument .................: VERITAS
 Event file .................: VR81417.chunk3.clean.fits
 Event type .................: EventList
 Statistic ..................: cstat
 Ontime .....................: 480 s
 Livetime ...................: 439.374101047894 s
```

```
 Deadtime correction .......: 0.915362710516446
 User energy range .........: undefined
=== GCTAPointing ===
 Pointing direction ........: (RA,Dec)=(83.6342417597327,21.5144715979806)
 Response function .........: undefined
=== GCTAEventList ===
 Number of events ..........: 262 (disposed in "VR81417.chunk3.clean.fits")
 Time interval .............: 57481.1812731481 - 57481.2021180556 days
 Energy interval ...........: 0.02 - 500 TeV
 Region of interest ........: RA=83.634242, DEC=21.514472 [0,0] Radius=2.25 deg
=== GSkyRegions ===
 Number of regions .........: 0
>>> print(obs.gti())
=== GGti ===
 Number of intervals .......: 1
 Ontime ....................: 1801.00000101328 sec
 Elapsed time ..............: 1801.00000101328 sec
 MJD range .................: 57481.1812731481 - 57481.2021180556 s (TT)
 UTC range .................: 2016-04-03T04:19:54 - 2016-04-03T04:49:55 s (TT)
 Reference MJD .............: 53856
>>> for ev in obs.events():
...     print(ev.time().utc())
...
2016-04-03T04:35:54
2016-04-03T04:35:56
2016-04-03T04:35:59
2016-04-03T04:36:00
2016-04-03T04:36:02
2016-04-03T04:36:03
2016-04-03T04:36:03
2016-04-03T04:36:03
2016-04-03T04:36:05
2016-04-03T04:36:05
```

Everything seems okay here. I get exactly the same result with ctools/gammalib version 1.5.2.

#4 - 01/09/2019 03:56 PM - Knödlseder Jürgen

Interesting: when running the code as you did I get the following with gammalib 1.5.2:

```
>>> from gammalib import GObservations, GCTAObservation
>>> obs = GObservations()
>>> ob = GCTAObservation()
>>> ob.load('VR81417.chunk3.clean.fits')
>>> ob.id('VR50990')
>>> obs.append(ob)
<gammalib.cta.GCTAObservation; proxy of <Swig Object of type 'GCTAObservation *' at 0x10b925840> >
>>> for ob in obs :
...     for ev in ob.events() :
...         print(ev.time().utc())
...
2009-12-04T16:35:56
2009-12-04T16:35:58
2009-12-04T16:36:01
2009-12-04T16:36:02
2009-12-04T16:36:04
```

Appending a CTA observation to the container seems to change the time reference (after appending it is 51544.5 while before it was 53856):

```
>>> print(obs[0].gti())
=== GGti ===
 Number of intervals .......: 1
 Ontime ....................: 1801.00000101328 sec
 Elapsed time ..............: 1801.00000101328 sec
 MJD range .................: 57481.1812731481 - 57481.2021180556 s (TT)
 UTC range .................: 2016-04-03T04:19:54 - 2016-04-03T04:49:55 s (TT)
 Reference MJD .............: 51544.5
```

This is however fixed in the devel version of the code which gives

```
>>> from gammalib import GObservations, GCTAObservation
>>> obs = GObservations()
>>> ob = GCTAObservation()
>>> ob.load('VR81417.chunk3.clean.fits')
>>> ob.id('VR50990')
>>> obs.append(ob)
<gammalib.cta.GCTAObservation; proxy of <Swig Object of type 'GCTAObservation *' at 0x10b925840> >
>>> for ob in obs :
...     for ev in ob.events() :
...         print(ev.time().utc())
...
2016-04-03T04:35:54
2016-04-03T04:35:56
2016-04-03T04:35:59
2016-04-03T04:36:00
2016-04-03T04:36:02
2016-04-03T04:36:03
2016-04-03T04:36:03
2016-04-03T04:36:03
2016-04-03T04:36:05
2016-04-03T04:36:05
>>> print(obs[0].gti())
=== GGti ===
 Number of intervals .......: 1
 Ontime ....................: 1801.00000101328 sec
 Elapsed time ..............: 1801.00000101328 sec
 MJD range .................: 57481.1812731481 - 57481.2021180556 s (TT)
 UTC range .................: 2016-04-03T04:19:54 - 2016-04-03T04:49:55 s (TT)
 Reference MJD .............: 53856
```

I guess this relates to #2054. Can you check with the devel version of the code?

#5 - 01/09/2019 03:57 PM - Knödlseder Jürgen

- Related to Change request #2054: MJD reference date in the output file of the ctbin task added

#6 - 01/10/2019 07:08 PM - Kelley-Hoskins Nathan

I'm trying to build it, but getting some python-related error:

```
$ ./configure --prefix=$PWD LDFLAGS=-L/gpfs/pace2/project/pc1/nkelleyh3/software/cfitsio/lib
 CPPFLAGS=-I/gpfs/pace2/project/pc1/nkelleyh3/software/cfitsio/include/ --without-mwl --without-lat --without-com
$ make -j 4

...

/nv/hp11/nkelleyh3/data/software/miniconda3//bin/python setup.py build_ext
Python extension module building:
Link cfitsio into Python module.
Link readline into Python module.
Include CTA instrument interface in gammalib Python module.
running build_ext
building '_app' extension
creating build
creating build/temp.linux-x86_64-3.6
creating build/temp.linux-x86_64-3.6/gammalib
gcc -Wsign-compare -DNDEBUG -g -fwrapv -O3 -Wall -Wstrict-prototypes -fPIC -I../include -I/gpfs/pace2/project/pc1/nkelleyh3/softw
are/cfitsio/include/ -I../inst/cta/include -I/nv/hp11/nkelleyh3/data/software/miniconda3/include/python3.6m -c gammalib/app_wrap.
cpp -o build/temp.linux-x86_64-3.6/gammalib/app_wrap.o -fopenmp
cc1plus: warning: command line option '-Wstrict-prototypes' is valid for C/ObjC but not for C++
gammalib/app_wrap.cpp: In function 'swig_module_info* SWIG_Python_GetModule()':
gammalib/app_wrap.cpp:2452:51: error: 'PyCObject_Import' was not declared in this scope
        (char*)"type_pointer" SWIG_TYPE_TABLE_NAME);
                                                   ^
gammalib/app_wrap.cpp: In function 'void SWIG_Python_SetModule(swig_module_info*)':
gammalib/app_wrap.cpp:2521:92: error: 'PyCObject_FromVoidPtr' was not declared in this scope
   PyObject *pointer = PyCObject_FromVoidPtr((void *) swig_module, SWIG_Python_DestroyModule);
                                                                                            ^
gammalib/app_wrap.cpp:2512:22: warning: unused variable 'swig_empty_runtime_method_table' [-Wunused-variable]
   static PyMethodDef swig_empty_runtime_method_table[] = { {NULL, NULL, 0, NULL} };/* Sentinel */
                     ^
gammalib/app_wrap.cpp: In function 'swig_type_info* SWIG_Python_TypeQuery(const char*)':
gammalib/app_wrap.cpp:2544:60: error: 'PyCObject_AsVoidPtr' was not declared in this scope
     descriptor = (swig_type_info *) PyCObject_AsVoidPtr(obj);
                                                         ^
gammalib/app_wrap.cpp:2549:51: error: 'PyCObject_FromVoidPtr' was not declared in this scope
       obj = PyCObject_FromVoidPtr(descriptor, NULL);
                                                   ^
gammalib/app_wrap.cpp: In function 'PyObject* _wrap_new_GLog__SWIG_1(PyObject*, PyObject*)':
gammalib/app_wrap.cpp:7460:28: error: 'PyString_Check' was not declared in this scope
     if (PyString_Check(obj0)) {
                        ^

...

gammalib/app_wrap.cpp:12075:106: error: 'PyString_AsString' was not declared in this scope
       temp2 = GFilename(std::string(PyString_AsString(PyUnicode_AsEncodedString(obj1, "utf-8", "Error ~"))));
                                                                                                          ^
gammalib/app_wrap.cpp: In function 'PyObject* _wrap_GApplicationPars_load__SWIG_1(PyObject*, PyObject*)':
gammalib/app_wrap.cpp:12142:28: error: 'PyString_Check' was not declared in this scope
     if (PyString_Check(obj1)) {
                        ^
gammalib/app_wrap.cpp:12143:59: error: 'PyString_AsString' was not declared in this scope
       temp2 = GFilename(std::string(PyString_AsString(obj1)));
                                                         ^
gammalib/app_wrap.cpp:12147:106: error: 'PyString_AsString' was not declared in this scope
       temp2 = GFilename(std::string(PyString_AsString(PyUnicode_AsEncodedString(obj1, "utf-8", "Error ~"))));
                                                                                                          ^
gammalib/app_wrap.cpp: In function 'PyObject* _wrap_GApplicationPars_load(PyObject*, PyObject*)':
gammalib/app_wrap.cpp:12218:35: error: 'PyString_Check' was not declared in this scope
         if (PyString_Check(argv[1]) ||
                            ^
gammalib/app_wrap.cpp:12242:35: error: 'PyString_Check' was not declared in this scope
         if (PyString_Check(argv[1]) ||
                            ^
gammalib/app_wrap.cpp: In function 'PyObject* _wrap_GApplicationPars_save(PyObject*, PyObject*)':
gammalib/app_wrap.cpp:12290:28: error: 'PyString_Check' was not declared in this scope
```

```
    if (PyString_Check(obj1)) {
                      ^
gammalib/app_wrap.cpp:12291:59: error: ‘PyString_AsString’ was not declared in this scope
      temp2 = GFilename(std::string(PyString_AsString(obj1)));
                                                ^
gammalib/app_wrap.cpp:12295:106: error: ‘PyString_AsString’ was not declared in this scope
      temp2 = GFilename(std::string(PyString_AsString(PyUnicode_AsEncodedString(obj1, "utf-8", "Error ~"))));
                                                                                                ^
gammalib/app_wrap.cpp: In function ‘PyObject* _wrap_GApplicationPar_filename__SWIG_0(PyObject*, PyObject*)’:
gammalib/app_wrap.cpp:13579:28: error: ‘PyString_Check’ was not declared in this scope
    if (PyString_Check(obj1)) {
                      ^
gammalib/app_wrap.cpp:13580:59: error: ‘PyString_AsString’ was not declared in this scope
      temp2 = GFilename(std::string(PyString_AsString(obj1)));
                                                ^
gammalib/app_wrap.cpp:13584:106: error: ‘PyString_AsString’ was not declared in this scope
      temp2 = GFilename(std::string(PyString_AsString(PyUnicode_AsEncodedString(obj1, "utf-8", "Error ~"))));
                                                                                                ^
gammalib/app_wrap.cpp: In function ‘PyObject* _wrap_GApplicationPar_filename(PyObject*, PyObject*)’:
gammalib/app_wrap.cpp:14277:35: error: ‘PyString_Check’ was not declared in this scope
       if (PyString_Check(argv[1]) ||
                                  ^
error: command 'gcc' failed with exit status 1
make[3]: *** [build] Error 1
make[3]: Leaving directory `/gpfs/pace2/project/pc1/nkelleyh3/software/gammalib-timebug/pyext'
make[2]: *** [all] Error 2
make[2]: Leaving directory `/gpfs/pace2/project/pc1/nkelleyh3/software/gammalib-timebug/pyext'
make[1]: *** [all-recursive] Error 1
make[1]: Leaving directory `/gpfs/pace2/project/pc1/nkelleyh3/software/gammalib-timebug'
make: *** [all] Error 2
```

My system:

```
$ uname -a
Linux iw-h31-20.pace.gatech.edu 2.6.32-573.12.1.el6.x86_64 #1 SMP Mon Nov 23 12:55:32 EST 2015 x86_64 x86_64 x86_64 GNU/Linux
```

My python is through anaconda:

```
$ python --version
Python 3.6.7 :: Anaconda, Inc.
```

#7 - 01/10/2019 10:27 PM - Knödlseder Jürgen

Compiling against Anaconda Python is not a good idea. It may or may not work. Just use the Vanilla Python on your machine.

#8 - 01/11/2019 03:55 PM - Kelley-Hoskins Nathan

Ok, I rebuilt it with regular python, and printed out the event times using devel, they come out correct now.

#9 - 03/06/2019 12:45 PM - Knödlseder Jürgen

- Status changed from Feedback to Closed

Files

VR81417.chunk3.clean.fits	39.4 KB	12/28/2018	Kelley-Hoskins Nathan