

ctools - Feature #1982

Phase information in event file

04/04/2017 10:07 AM - Cardenzana Josh

| | | | |
|---|--------------------|------------------------|------------|
| Status: | Closed | Start date: | 04/11/2017 |
| Priority: | Normal | Due date: | |
| Assigned To: | Di Venere Leonardo | % Done: | 100% |
| Category: | | Estimated time: | 3.00 hours |
| Target version: | 1.3.0 | | |
| Description | | | |
| Add a new tool 'ctphase' for adding phase information to event files. Allow both information from an xml file and parameters from a 'ctphase.par' file. | | | |

History

#1 - 04/06/2017 12:06 PM - Cardenzana Josh

- File MJD-59000_P0-0_F0-1_F1-1e-8_F2-0.png added
- File MJD-59000_P0-0_F0-1_F1-1e-8_F2-1e-8.png added
- File Period-3minutes.png added
- Status changed from New to In Progress
- % Done changed from 0 to 50

The tool has been developed and somewhat tested. Phase information is derived using the GModelTemporalPhaseCurve class and the user supplied reference values for:

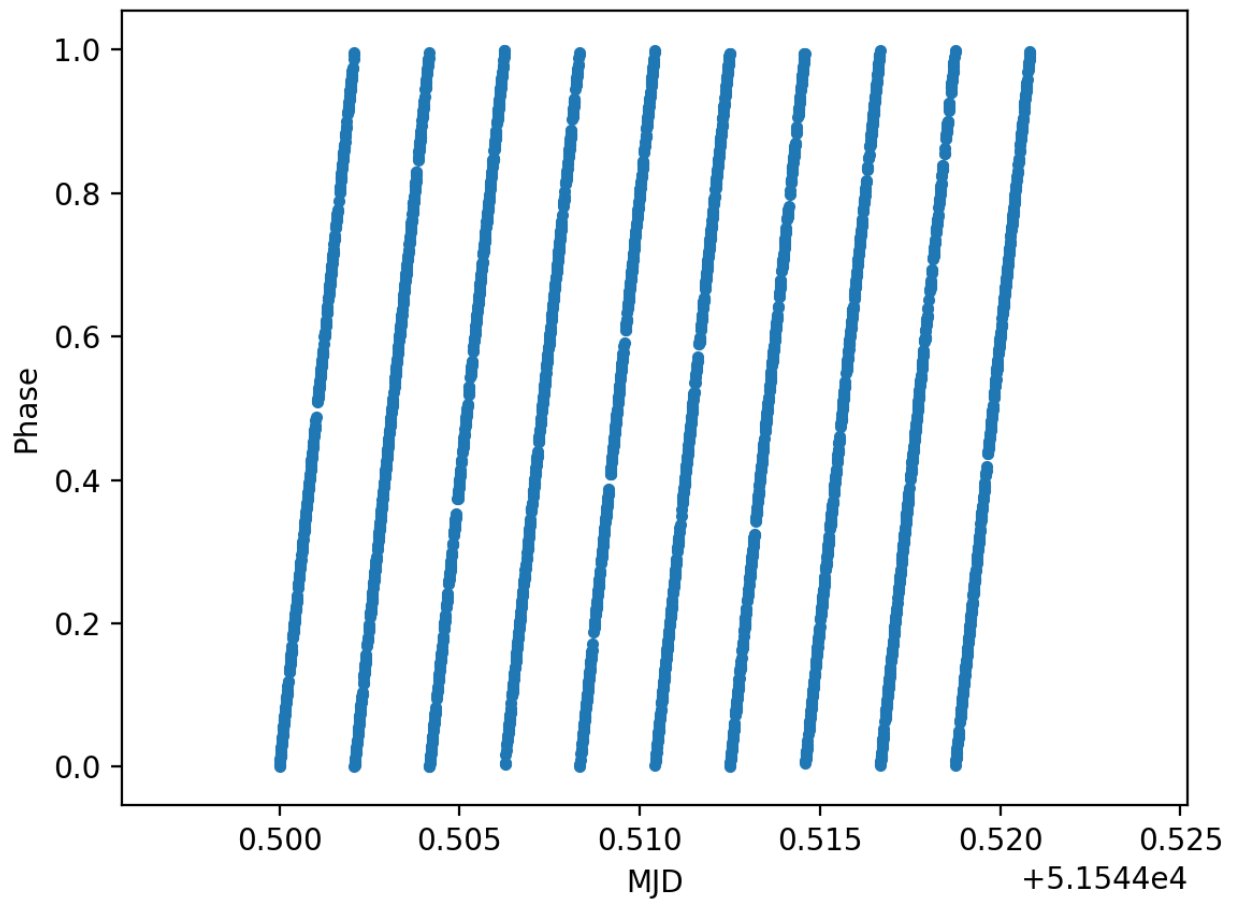
- "MJD" (reference MJD)
- "P0" (phase at reference MJD)
- "F0" (frequency at reference MJD in Hz)
- "F1" (first frequency derivative at reference MJD in sec[^](-1))
- "F2" (second frequency derivative at reference MJD in sec[^](-2))

The phase information is successfully appended to the events file, however precision issues can become a problem. These issues appear to only occur when a reference date far from the MJD of the data is chosen (more than a few years difference in the dates) or when a large value for the first/second derivate are chosen. For the time being, a warning is printed into the ctphase log file when either (F1 * dt) or (F2 * dt * dt) is greater than 10^8 (dt is seconds between the reference MJD and the MJD of the events). When this occurs the user should check that the distribution of derived phases makes sense. The method should be stable for most sensible choices of the parameters.

Tests of the above issue have been done simulating events with ctobssim and then passing the resulting events file through ctphase. These results are documented below:

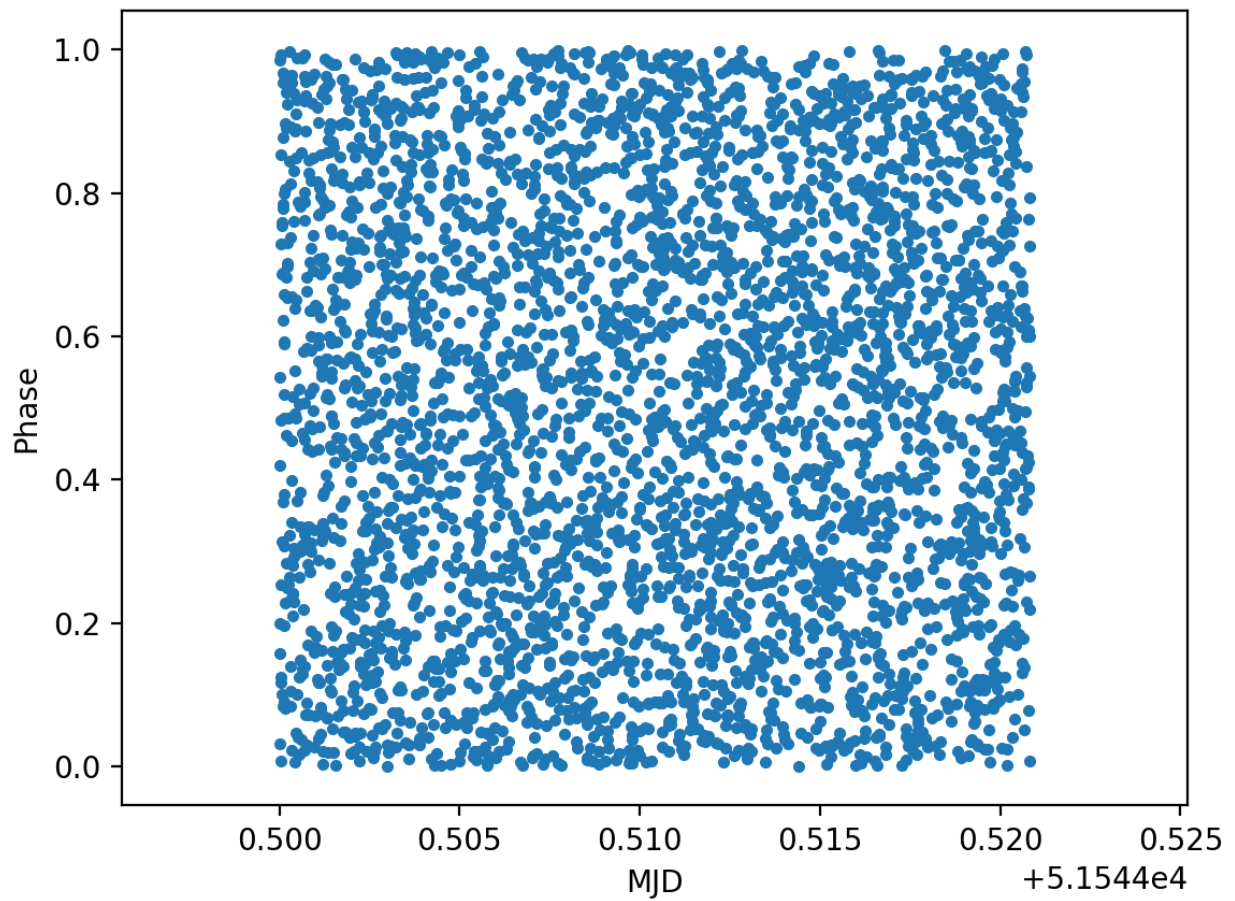
Example plot for 3 minute period:

- Data MJD: 51544.5
- Reference MJD: 59000
- P0: 0.0
- F0: 0.00555556 (1/180)
- F1: 0.0
- F2: 0.0



Example plot for sensible, uniform distribution of phases as a function of event time:

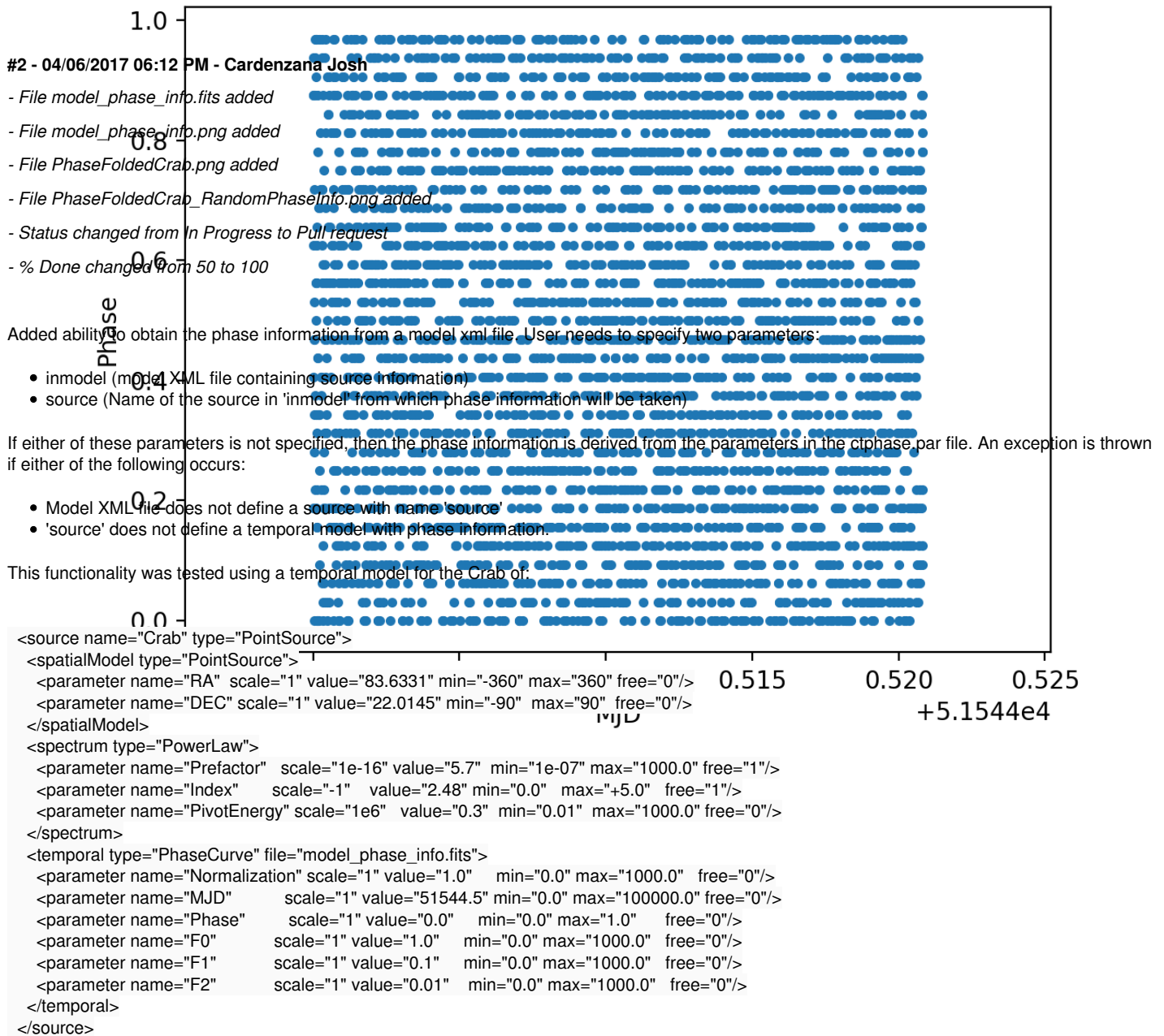
- Data MJD: 51544.5
- Reference MJD: 59000
- P_0 : 0.0
- F_0 : 1.0
- F_1 : 1×10^{-8}
- F_2 : 0



Example plot showing precision issues (generated phases do not appear uniformly distributed)

- Data MJD: 51544.5
- Reference MJD: 59000

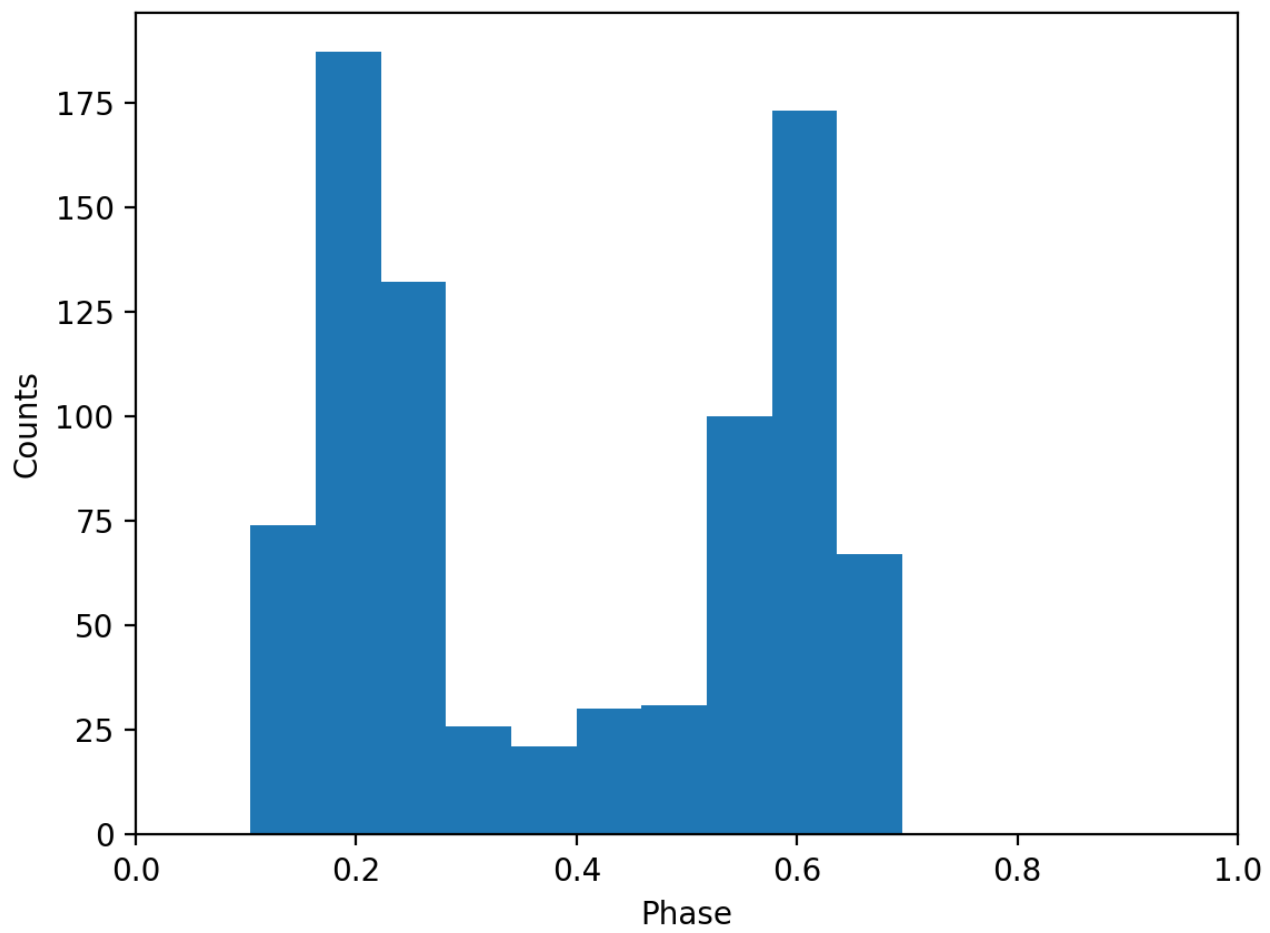
- P0: 0.0
- F0: 1.0
- F1: 1.0e-8
- F2: 1.0e-8



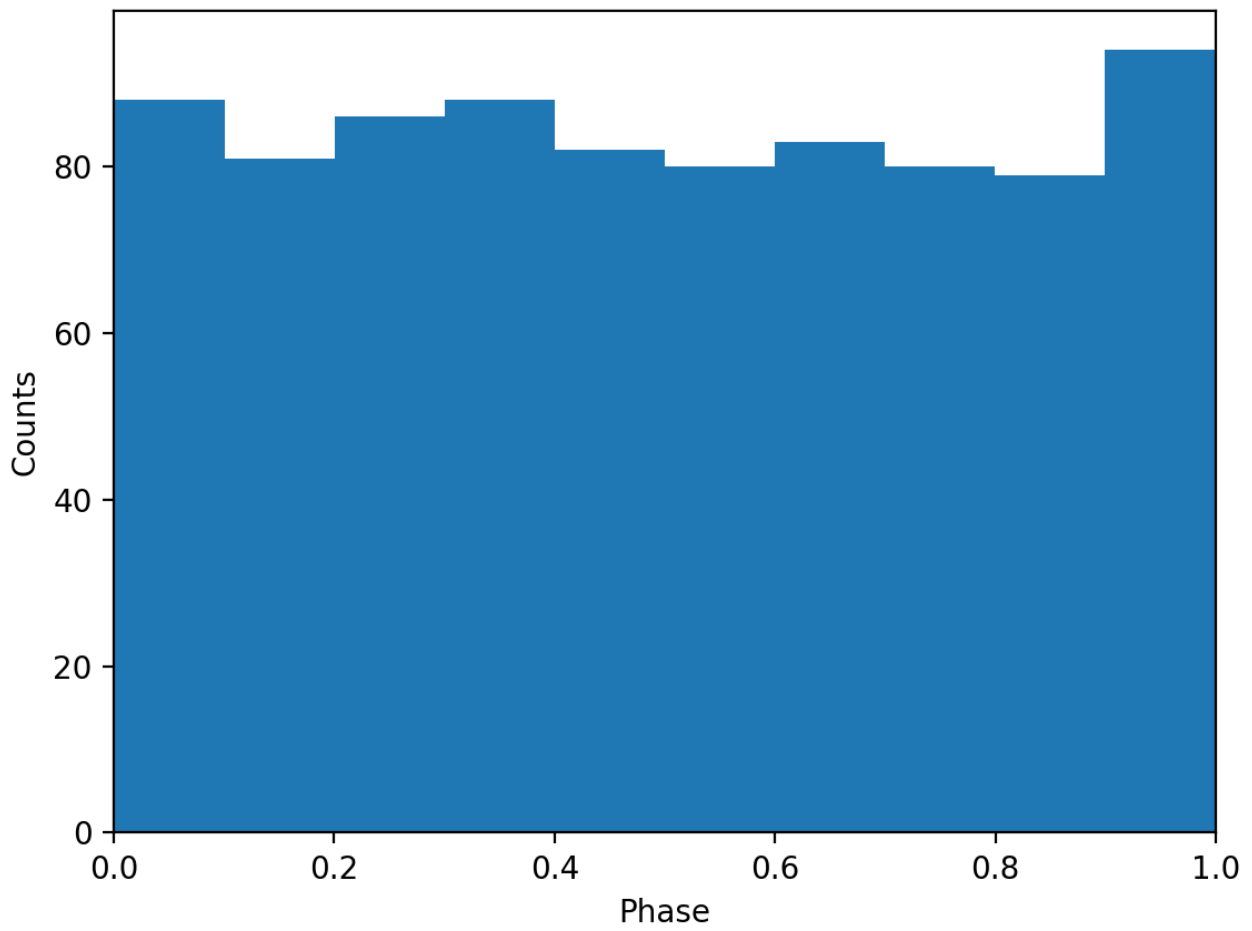
'model_phase_info.fits' is attached and defines node information in the following way:

| <input type="checkbox"/> PHASE Select E | | <input type="checkbox"/> NORM E | |
|--|---------------|------------------------------------|--|
| <input type="checkbox"/> All Invert | Modify | Modify | |
| 1 | 1.0000000E-03 | 0.0000000E+00 | |
| 2 | 1.0000000E-01 | 0.0000000E+00 | |
| 3 | 2.0000000E-01 | 1.0000000E+00 | |
| 4 | 3.0000000E-01 | 1.0000000E-01 | |
| 5 | 4.0000000E-01 | 1.0000000E-01 | |
| 6 | 5.0000000E-01 | 1.0000000E-01 | |
| 7 | 6.0000000E-01 | 9.0000000E-01 | |
| 8 | 7.0000000E-01 | 0.0000000E+00 | |
| 9 | 8.0000000E-01 | 0.0000000E+00 | |
| 10 | 9.0000000E-01 | 0.0000000E+00 | |

Generating events from only the above Crab model and processing through ctpphase produces the following distribution of events (as a function of phase):



Processing the same data through ctpase, but using random values for the phase information produces the following (seemingly random) distribution of events. This is the expected behavior.



I presume that the features requested are now implemented and a pull of this code can be done. Note, a pull of the associated gammaLib branch is also necessary for appropriate functionality, otherwise errors will be produced by GModelTemporalPhaseCurve.

Relevant repositories for pull request:

- GammaLib: [git@cta-gitlab.irap.omp.eu:jcardenzana/gammalib.git](https://git.cta-gitlab.irap.omp.eu/jcardenzana/gammalib.git)
- GammaLib branch: 1982-add-phase-info
- CTools: [git@cta-gitlab.irap.omp.eu:jcardenzana/ctools.git](https://git.cta-gitlab.irap.omp.eu/jcardenzana/ctools.git)
- CTools branch: 1982-add-phase-info-ctphase

#3 - 04/11/2017 03:04 PM - Knödlseider Jürgen

- Target version set to 1.3.0
- Start date set to 04/11/2017

I integrated the GammaLib branch 1982-add-phase-info.

I've seen that you allowed an empty filename in the `GModelTemporalPhaseCurve::read()` method, however I'm worried about the possible side effects of that. In particular, a phase object without any nodes does not really make sense. What is the Use Case of this functionality? For the moment I have not merged in that part.

#4 - 04/11/2017 04:04 PM - Cardenzana Josh

We originally were going to just set the parameters individually using the dedicated `::phase()`, `::f0()`, etc... methods, but ran into issues where any parameter values passed that were outside the range defined in `"init_members()"` resulted in an exception (this was an issue for negative values of `F1` and `F2`).

However, we could control the Min/Max values if we pass the parameters inside a `"GXmlElement"` object. But then there's an error that gets produced in the `"GModelTemporalPhaseCurve::read(const GXmlElement& xml)"` method when the value in the `"file"` attribute doesn't exist. When computing the phases (to the best of my knowledge) we only need to know the time of an event, thus the node information from the file is never used. An empty file attribute still produces an error in an analysis where the node information is needed. But on second thought the original `"invalid file"` error is probably more useful for the user than a seg-fault later on during execution.

Alternative approach:

Expand the ranges of the `f1/f2` parameters in `"GModelTemporalPhaseCurve::init_members()"` to be `[-1000, 1000]` instead of `[0,1000]` and then set all phase/frequency parameters via their dedicated call methods (doesn't require an additional constructor and the read method can be reset to requiring a legitimate file attribute). This should probably be done anyway, since the frequency of a pulsar will most likely decrease over time (i.e. `f1` will most likely be negative).

Unless you can think of a reason not to, I will make this change, rerun the tests to make sure everything still works as expected, and post once I've verified the results haven't changed.

#5 - 04/11/2017 05:14 PM - Knödlseider Jürgen

Indeed, I had not recognized that the phase file is not needed wink.png

I prefer expanding the default range of `F1` and `F2` to `[-1000,1000]` so that the coherence of the XML format is always checked. Otherwise a user may eventually omit the file attribute and does not understand why the model is not working.

#6 - 04/12/2017 12:03 AM - Knödlseider Jürgen

- Status changed from Pull request to Closed

I merged the ctpase tool into devel.

I also added a page to the reference manual.

#7 - 04/12/2017 01:09 AM - Cardenzana Josh

- File test_model.xml added

- File test.py added

I've made the above change and verified that the results are consistent with those presented previously. I've attached my test python script (with supplemental files) that produces the events, then stores the phase information and plots the distribution of events in "phase". Note that the 'model_phase_info.fits' file attached in a previous post is also necessary to get the node information in the pulse profile.

Relevant repositories for pull request:

- GammaLib: [git@cta-github.irap.omp.eu:jcardenzana/gammalib.git](https://git.cta-github.irap.omp.eu/jcardenzana/gammalib.git)
- GammaLib branch: 1982-add-phase-info
- CTools: [git@cta-github.irap.omp.eu:jcardenzana/ctools.git](https://git.cta-github.irap.omp.eu/jcardenzana/ctools.git)
- CTools branch: 1982-add-phase-info-ctphase

#8 - 04/12/2017 09:36 AM - Knödlseider Jürgen

- File test.py added

Thanks. Note that I modified two parameter names so that they are compliant with the names of other tools:

source -> srcname
p0 -> phase

I modified a bit the script to avoid using astropy (which I don't have installed): attachment:test.py

| Files | | | |
|---|---------|------------|-----------------|
| MJD-59000_P0-0_F0-1_F1-1e-8_F2-0.png | 210 KB | 04/06/2017 | Cardenzana Josh |
| MJD-59000_P0-0_F0-1_F1-1e-8_F2-1e-8.png | 128 KB | 04/06/2017 | Cardenzana Josh |
| Period-3minutes.png | 92.1 KB | 04/06/2017 | Cardenzana Josh |
| model_phase_info.fits | 8.44 KB | 04/06/2017 | Cardenzana Josh |
| model_phase_info.png | 35.2 KB | 04/06/2017 | Cardenzana Josh |
| PhaseFoldedCrab.png | 25.9 KB | 04/06/2017 | Cardenzana Josh |
| PhaseFoldedCrab_RandomPhaseInfo.png | 23.1 KB | 04/06/2017 | Cardenzana Josh |
| test_model.xml | 1.8 KB | 04/11/2017 | Cardenzana Josh |
| test.py | 2.29 KB | 04/11/2017 | Cardenzana Josh |

