GammaLib - Bug #1502

Memory leak in GCTAObservation on Mac OS X dependent on m_object member placement

07/02/2015 04:58 PM - Knödlseder Jürgen

Status:	In Progress	**Start date:**	07/02/2015
Priority:	High	**Due date:**	
Assigned To:	Knödlseder Jürgen	**% Done:**	30%
Category:		**Estimated time:**	0.00 hour
Target version:			

Description
There is a memory leak in GCTAObservation that appears on Mac OS X systems and that relates to the placement of the m_object data member. I could so far not find any reason for this memory leak.

This feature is to keep track of the problem, needing a deeper investigation using for example valgrind.

For the moment, the code compiles well by moving m_object down in the list. This is of course only a kluge and not at all satisfactory.

History

#1 - 07/02/2015 08:55 PM - Knödlseder Jürgen

- Status changed from New to In Progress

- Assigned To set to Knödlseder Jürgen

- % Done changed from 0 to 10

A valgrind analysis of test_CTA gave the following result:

```
*****************************************
* CTA instrument specific class testing *
*****************************************
==66396== Invalid read of size 1
==66396==    at 0x1007ADA11: __findenv (in /usr/lib/libSystem.B.dylib)
==66396==    by 0x1007AD9BD: getenv (in /usr/lib/libSystem.B.dylib)
==66396==    by 0x1007FFB80: __detect_path_locale (in /usr/lib/libSystem.B.dylib)
==66396==    by 0x1007FFA02: loadlocale (in /usr/lib/libSystem.B.dylib)
==66396==    by 0x1007E92C0: setlocale (in /usr/lib/libSystem.B.dylib)
==66396==    by 0x100A42CBA: void std::__convert_to_v<double>(char const*, double&, std::_Ios_Iostate&, int* const&) (in
/usr/lib/libstdc++.6.0.9.dylib)
==66396==    by 0x100A253DF: std::num_get<char, std::istreambuf_iterator<char, std::char_traits<char> > >::do_get(std::istreambuf_iterator<char,
std::char_traits<char> >, std::istreambuf_iterator<char, std::char_traits<char> >, std::ios_base&, std::_Ios_Iostate&, double&) const (in
/usr/lib/libstdc++.6.0.9.dylib)
==66396==    by 0x100A15BC6: std::istream& std::istream::_M_extract<double>(double&) (in /usr/lib/libstdc++.6.0.9.dylib)
==66396==    by 0x100061568: gammalib::todouble(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==66396==    by 0x100251B8B: GCTAAeffPerfTable::load(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==66396==    by 0x10025219C: GCTAAeffPerfTable::GCTAAeffPerfTable(std::string const&) (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==66396==    by 0x10022EDAF: GCTAResponseIrf::load_aeff(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==66396== Address 0x100b3dec8 is 24 bytes inside a block of size 72 free'd
==66396==    at 0x10003095E: free (vg_replace_malloc.c:450)
==66396==    by 0x1000031D7: main (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==66396==
Test response: ..... ok
Test effective area: .. ok
Test PSF: ...................................................................... ok
Test King profile PSF: ........ ok
Test integrated PSF: ..... ok
Test energy dispersion: .............. ok
Test energy dispersion Performance Table computation: ........... ok
Test energy dispersion RMF computation: ........... ok
Test energy dispersion 2D computation: .......... ok
Test diffuse IRF: F. NOK
Test diffuse IRF integration: F. NOK
Test exposure cube: ... ok
Test PSF cube: ... ok
```

Test background cube: .. ok
Test CTA cube background model: .... ok
Test CTA IRF background model: .... ok
Test unbinned optimizer: .... ok
Test binned optimizer: .... ok
Test cube-style optimizer: .... ok
Test unbinned observations: ..... ok
Test binned observation: ... ok
Test cube-style observation: ... ok
Test load pointing from table: . ok
Test alt/az interpolation given a time: .. ok
==66396==
==66396== HEAP SUMMARY:
==66396==   in use at exit: 16,478 bytes in 32 blocks
==66396==   total heap usage: 32,893,282 allocs, 32,893,250 frees, 2,448,107,367 bytes allocated
==66396==
==66396== 16 bytes in 1 blocks are definitely lost in loss record 1 of 25
==66396==   at 0x100030C16: malloc (vg_replace_malloc.c:274)
==66396==   by 0x100A480CF: __cxa_get_globals (in /usr/lib/libstdc++.6.0.9.dylib)
==66396==   by 0x100A47DCD: __cxa_allocate_exception (in /usr/lib/libstdc++.6.0.9.dylib)
==66396==   by 0x1000CC8EF: GFits::open(std::string const&, bool const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==66396==   by 0x1000CD578: GFits::GFits(std::string const&, bool const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==66396==   by 0x10022E802: GCTAResponseIrf::load_aeff(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==66396==   by 0x100233B62: GCTAResponseIrf::load(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==66396==   by 0x100006C3D: TestGCTAResponse::test_response() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==66396==   by 0x1001EB736: GTestSuite::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==66396==   by 0x1001F2B9E: GTestSuites::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==66396==   by 0x1000030E4: main (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==66396==
==66396== LEAK SUMMARY:
==66396==   definitely lost: 16 bytes in 1 blocks
==66396==   indirectly lost: 0 bytes in 0 blocks
==66396==   possibly lost: 0 bytes in 0 blocks
==66396==   still reachable: 16,374 bytes in 30 blocks
==66396==   suppressed: 88 bytes in 1 blocks
==66396== Reachable blocks (those to which a pointer was found) are not shown.
==66396== To see them, rerun with: --leak-check=full --show-reachable=yes
==66396==
==66396== For counts of detected and suppressed errors, rerun with: -v
==66396== ERROR SUMMARY: 2 errors from 2 contexts (suppressed: 0 from 0)

After changing


```
double gammalib::todouble(const std::string& arg)
{
    /*
    std::istringstream iss(arg);
    double          result;
    iss >> std::dec >> result;
    return result;
    */
    double result = strtod(arg.c_str(), NULL);
    return result;
}
```


the first problem got away:

Test response: ..... ok
Test effective area: .. ok
Test PSF: .... ok
Test King profile PSF: .... ok
Test integrated PSF: ..... ok
Test energy dispersion: .... ok
Test energy dispersion Performance Table computation: .... ok
Test energy dispersion RMF computation: .... ok
Test energy dispersion 2D computation: .... ok
Test diffuse IRF: F. NOK
Test diffuse IRF integration: F. NOK
Test exposure cube: ... ok
Test PSF cube: ... ok
Test background cube: .. ok

Test CTA cube background model: .... ok
Test CTA IRF background model: .... ok
Test unbinned optimizer: .... ok
Test binned optimizer: .... ok
Test cube-style optimizer: .... ok
Test unbinned observations: ..... ok
Test binned observation: ... ok
Test cube-style observation: ... ok
Test load pointing from table: . ok
Test alt/az interpolation given a time: .. ok
==28975==
==28975== HEAP SUMMARY:
==28975==   in use at exit: 16,478 bytes in 32 blocks
==28975==   total heap usage: 32,879,575 allocs, 32,879,543 frees, 2,447,694,543 bytes allocated
==28975==
==28975== 16 bytes in 1 blocks are definitely lost in loss record 1 of 25
==28975==   at 0x1000315CF: malloc (vg_replace_malloc.c:266)
==28975==   by 0x100A470CF: __cxa_get_globals (in /usr/lib/libstdc++.6.0.9.dylib)
==28975==   by 0x100A46DCD: __cxa_allocate_exception (in /usr/lib/libstdc++.6.0.9.dylib)
==28975==   by 0x1000CB93F: GFits::open(std::string const&, bool const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==28975==   by 0x1000CC5C8: GFits::GFits(std::string const&, bool const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==28975==   by 0x10022D852: GCTAResponseIrf::load_aeff(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==28975==   by 0x100232BB2: GCTAResponseIrf::load(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==28975==   by 0x100006C3D: TestGCTAResponse::test_response() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==28975==   by 0x1001EA786: GTestSuite::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==28975==   by 0x1001F1BEE: GTestSuites::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==28975==   by 0x1000030E4: main (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==28975==
==28975== LEAK SUMMARY:
==28975==   definitely lost: 16 bytes in 1 blocks
==28975==   indirectly lost: 0 bytes in 0 blocks
==28975==   possibly lost: 0 bytes in 0 blocks
==28975==   still reachable: 16,374 bytes in 30 blocks
==28975==   suppressed: 88 bytes in 1 blocks
==28975== Reachable blocks (those to which a pointer was found) are not shown.
==28975== To see them, rerun with: --leak-check=full --show-reachable=yes
==28975==
==28975== For counts of detected and suppressed errors, rerun with: -v
==28975== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)

#2 - 07/02/2015 09:59 PM - Knödlseder Jürgen

Using the gammalib::is_fits function just moves the problem to another place. The problem is with forcing an exception:


```
Test response: ..... ok
Test effective area: .. ok
Test PSF: .................................................................. ok
Test King profile PSF: ........ ok
Test integrated PSF: ..... ok
Test energy dispersion: .............. ok
Test energy dispersion Performance Table computation: ........... ok
Test energy dispersion RMF computation: ........... ok
Test energy dispersion 2D computation: .......... ok
Test diffuse IRF: F. NOK
Test diffuse IRF integration: F. NOK
Test exposure cube: ... ok
Test PSF cube: ... ok
Test background cube: .. ok
Test CTA cube background model: ................... ok
Test CTA IRF background model: .................. ok
Test unbinned optimizer: ....................... ok
Test binned optimizer: ....................... ok
Test cube-style optimizer: ....................... ok
Test unbinned observations: ..... ok
Test binned observation: ... ok
Test cube-style observation: ... ok
Test load pointing from table: . ok
Test alt/az interpolation given a time: .. ok
==36024==
==36024== HEAP SUMMARY:
==36024==     in use at exit: 16,478 bytes in 32 blocks
==36024==   total heap usage: 32,890,078 allocs, 32,890,046 frees, 2,448,676,773 bytes allocated
==36024==
==36024== 16 bytes in 1 blocks are definitely lost in loss record 1 of 25
==36024==    at 0x1000315CF: malloc (vg_replace_malloc.c:266)
==36024==    by 0x100A480CF: __cxa_get_globals (in /usr/lib/libstdc++.6.0.9.dylib)
==36024==    by 0x100A47DCD: __cxa_allocate_exception (in /usr/lib/libstdc++.6.0.9.dylib)
==36024==    by 0x1000CBB4F: GFits::open(std::string const&, bool const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==36024==    by 0x1000CC7D8: GFits::GFits(std::string const&, bool const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==36024==    by 0x1000604CD: gammalib::is_fits(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==36024==    by 0x10022D984: GCTAResponseIrf::load_aeff(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==36024==    by 0x100232CB2: GCTAResponseIrf::load(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==36024==    by 0x100006C3D: TestGCTAResponse::test_response() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==36024==    by 0x1001EA996: GTestSuite::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==36024==    by 0x1001F1DFE: GTestSuites::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==36024==    by 0x1000030E4: main (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==36024==
==36024== LEAK SUMMARY:
==36024==    definitely lost: 16 bytes in 1 blocks
==36024==    indirectly lost: 0 bytes in 0 blocks
==36024==      possibly lost: 0 bytes in 0 blocks
==36024==    still reachable: 16,374 bytes in 30 blocks
==36024==         suppressed: 88 bytes in 1 blocks
==36024== Reachable blocks (those to which a pointer was found) are not shown.
==36024== To see them, rerun with: --leak-check=full --show-reachable=yes
==36024==
==36024== For counts of detected and suppressed errors, rerun with: -v
==36024== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
```

#3 - 07/02/2015 10:57 PM - Knödlseder Jürgen

After correcting the gammalib::is_fits() method I now get the following error:

```
==45489==
==45489== HEAP SUMMARY:
==45489==     in use at exit: 260,698 bytes in 92 blocks
==45489==   total heap usage: 32,878,686 allocs, 32,878,594 frees, 2,440,903,327 bytes allocated
==45489==
==45489== 16 bytes in 1 blocks are definitely lost in loss record 1 of 85
==45489==    at 0x1000315CF: malloc (vg_replace_malloc.c:266)
==45489==    by 0x100A470CF: __cxa_get_globals (in /usr/lib/libstdc++.6.0.9.dylib)
==45489==    by 0x100A46DCD: __cxa_allocate_exception (in /usr/lib/libstdc++.6.0.9.dylib)
==45489==    by 0x1000D7B54: GXmlNode::element(std::string const&, int const&) const (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==45489==    by 0x10029E00C: GCTAModelRadialAcceptance::read(GXmlElement const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==45489==    by 0x100167FFE: GModels::read(GXml const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==45489==    by 0x10016856F: GModels::load(std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==45489==    by 0x10000777A: TestGCTAOptimize::test_unbinned_optimizer() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==45489==    by 0x1001EAA66: GTestSuite::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==45489==    by 0x1001F1ECE: GTestSuites::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==45489==    by 0x1000030E4: main (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==45489==
==45489== LEAK SUMMARY:
==45489==    definitely lost: 16 bytes in 1 blocks
==45489==    indirectly lost: 0 bytes in 0 blocks
==45489==      possibly lost: 0 bytes in 0 blocks
==45489==    still reachable: 260,594 bytes in 90 blocks
==45489==         suppressed: 88 bytes in 1 blocks
==45489== Reachable blocks (those to which a pointer was found) are not shown.
==45489== To see them, rerun with: --leak-check=full --show-reachable=yes
==45489==
==45489== For counts of detected and suppressed errors, rerun with: -v
==45489== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
```

#4 - 07/02/2015 11:45 PM - Knödlseder Jürgen

Having solved the previous one I now obtained the following:

```
==52004==
==52004== HEAP SUMMARY:
==52004==     in use at exit: 260,698 bytes in 92 blocks
==52004==   total heap usage: 32,878,679 allocs, 32,878,587 frees, 2,440,902,494 bytes allocated
==52004==
==52004== 16 bytes in 1 blocks are definitely lost in loss record 1 of 85
==52004==    at 0x1000315CF: malloc (vg_replace_malloc.c:266)
==52004==    by 0x100A470CF: __cxa_get_globals (in /usr/lib/libstdc++.6.0.9.dylib)
==52004==    by 0x100A46DCD: __cxa_allocate_exception (in /usr/lib/libstdc++.6.0.9.dylib)
==52004==    by 0x100280369: GCTAPointing::dir_horiz(GTime const&) const (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==52004==    by 0x1000057C7: TestGCTAPointing::test_interpolate_altaz() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==52004==    by 0x1001EA906: GTestSuite::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==52004==    by 0x1001F1D6E: GTestSuites::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==52004==    by 0x1000030E4: main (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==52004==
==52004== LEAK SUMMARY:
==52004==    definitely lost: 16 bytes in 1 blocks
==52004==    indirectly lost: 0 bytes in 0 blocks
==52004==      possibly lost: 0 bytes in 0 blocks
==52004==    still reachable: 260,594 bytes in 90 blocks
==52004==         suppressed: 88 bytes in 1 blocks
==52004== Reachable blocks (those to which a pointer was found) are not shown.
==52004== To see them, rerun with: --leak-check=full --show-reachable=yes
==52004==
==52004== For counts of detected and suppressed errors, rerun with: -v
==52004== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
```

This one turns out to be a try-catch construction for an out of bound test.

#5 - 07/02/2015 11:53 PM - Knödlseder Jürgen

- % Done changed from 10 to 20

At this point I changed again the order of the members in GCTAObservation to see whether the modifications had any effect. However they did not, and I still get the memory leak problem.

I therefore apply now valgrind on the code with the problematic member order.

#6 - 07/03/2015 09:05 AM - Knödlseder Jürgen

There are now many errors of the following type:

```
==61308== Conditional jump or move depends on uninitialised value(s)
==61308==    at 0x100095E74: gammalib::acos(double const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x1000FCBFE: GSkyDir::dist(GSkyDir const&) const (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10022FA15: GCTAResponseIrf::irf_diffuse(GEvent const&, GSource const&, GObservation const&) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10015D601: GResponse::eval_prob(GModelSky const&, GEvent const&, GEnergy const&, GTime const&, GObservation const&,
bool const&) const (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10015E19E: GResponse::convolve(GModelSky const&, GEvent const&, GObservation const&, bool const&) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x100171232: GModelSky::eval_gradients(GEvent const&, GObservation const&) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x100151EC0: GObservation::model(GModels const&, GEvent const&, GVector*) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10000A7A6: TestGCTAResponse::test_response_irf_diffuse() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==61308==    by 0x1001EA926: GTestSuite::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x1001F1D8E: GTestSuites::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x1000030E4: main (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==61308==  Uninitialised value was created by a stack allocation
==61308==    at 0x10000A33D: TestGCTAResponse::test_response_irf_diffuse() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
...
==61308== Conditional jump or move depends on uninitialised value(s)
==61308==    at 0x1007D9055: sin$fenv_access_off (in /usr/lib/libSystem.B.dylib)
==61308==    by 0x10022FC46: GCTAResponseIrf::irf_diffuse(GEvent const&, GSource const&, GObservation const&) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10015D601: GResponse::eval_prob(GModelSky const&, GEvent const&, GEnergy const&, GTime const&, GObservation const&,
bool const&) const (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10015E19E: GResponse::convolve(GModelSky const&, GEvent const&, GObservation const&, bool const&) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x100171232: GModelSky::eval_gradients(GEvent const&, GObservation const&) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x100151EC0: GObservation::model(GModels const&, GEvent const&, GVector*) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10000A7A6: TestGCTAResponse::test_response_irf_diffuse() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==61308==    by 0x1001EA926: GTestSuite::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x1001F1D8E: GTestSuites::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x1000030E4: main (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==61308==  Uninitialised value was created by a stack allocation
==61308==    at 0x10000A33D: TestGCTAResponse::test_response_irf_diffuse() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
...
==61308== Conditional jump or move depends on uninitialised value(s)
==61308==    at 0x1007DDD44: acos$fenv_access_off (in /usr/lib/libSystem.B.dylib)
==61308==    by 0x1002430A0: cta_irf_diffuse_kern_phi::eval(double const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10008AC54: GIntegral::trapzd(double const&, double const&, int const&, double) (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10008DDF3: GIntegral::romberg(double const&, double const&, int const&) (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x100243531: cta_irf_diffuse_kern_theta::eval(double const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10008ACE6: GIntegral::trapzd(double const&, double const&, int const&, double) (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10008DDF3: GIntegral::romberg(double const&, double const&, int const&) (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10022FCB9: GCTAResponseIrf::irf_diffuse(GEvent const&, GSource const&, GObservation const&) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10015D601: GResponse::eval_prob(GModelSky const&, GEvent const&, GEnergy const&, GTime const&, GObservation const&,
bool const&) const (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x10015E19E: GResponse::convolve(GModelSky const&, GEvent const&, GObservation const&, bool const&) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x100171232: GModelSky::eval_gradients(GEvent const&, GObservation const&) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==    by 0x100151EC0: GObservation::model(GModels const&, GEvent const&, GVector*) const (in
/Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==  Uninitialised value was created by a stack allocation
==61308==    at 0x10000A33D: TestGCTAResponse::test_response_irf_diffuse() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
...
==61308== Conditional jump or move depends on uninitialised value(s)
==61308==    at 0x1007E2A3E: __d2b_D2A (in /usr/lib/libSystem.B.dylib)
==61308==    by 0x1007F6D47: __dtoa (in /usr/lib/libSystem.B.dylib)
==61308==    by 0x1007B899F: __vfprintf (in /usr/lib/libSystem.B.dylib)
==61308==    by 0x1007EC5B6: vsnprintf (in /usr/lib/libSystem.B.dylib)
==61308==    by 0x100A3CB0A: std::__convert_from_v(int* const&, char*, int, char const*, ...) (in /usr/lib/libstdc++.6.0.9.dylib)
```

==61308==   by 0x100A1F99A: std::ostreambuf_iterator<char, std::char_traits<char> > std::num_put<char, std::ostreambuf_iterator<char, std::char_traits<char> > >::_M_insert_float<double>(std::ostreambuf_iterator<char, std::char_traits<char> >, std::ios_base&, char, char, double) const (in /usr/lib/libstdc++.6.0.9.dylib)
==61308==   by 0x100A1FBBF: std::num_put<char, std::ostreambuf_iterator<char, std::char_traits<char> > >::do_put(std::ostreambuf_iterator<char, std::char_traits<char> >, std::ios_base&, char, double) const (in /usr/lib/libstdc++.6.0.9.dylib)
==61308==   by 0x100A276A1: std::ostream& std::ostream::_M_insert<double>(double) (in /usr/lib/libstdc++.6.0.9.dylib)
==61308==   by 0x10003FFC8: gammalib::str(double const&, int const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==   by 0x1001ECBBF: GTestSuite::test_value(double const&, double const&, double const&, std::string const&, std::string const&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==   by 0x10000A859: TestGCTAResponse::test_response_irf_diffuse() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==61308==   by 0x1001EA926: GTestSuite::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308== Uninitialised value was created by a stack allocation
==61308==   at 0x10000A33D: TestGCTAResponse::test_response_irf_diffuse() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
...
==61308==
==61308== HEAP SUMMARY:
==61308==   in use at exit: 261,082 bytes in 96 blocks
==61308==   total heap usage: 27,903,267 allocs, 27,903,171 frees, 1,779,975,250 bytes allocated
==61308==
==61308== 16 bytes in 1 blocks are definitely lost in loss record 1 of 89
==61308==   at 0x1000315CF: malloc (vg_replace_malloc.c:266)
==61308==   by 0x100A470CF: __cxa_get_globals (in /usr/lib/libstdc++.6.0.9.dylib)
==61308==   by 0x100A46DCD: __cxa_allocate_exception (in /usr/lib/libstdc++.6.0.9.dylib)
==61308==   by 0x100079B82: GMatrixSparse::remove_zero_row_col() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==   by 0x100079CAA: GMatrixSparse::cholesky_decompose(bool const&) const (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==   by 0x10007AEE3: GMatrixSparse::solve(GVector const&) const (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==   by 0x10012F1F7: GOptimizerLM::iteration(GOptimizerFunction&, GOptimizerPars&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==   by 0x100130D39: GOptimizerLM::optimize(GOptimizerFunction&, GOptimizerPars&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==   by 0x10014891A: GObservations::optimize(GOptimizer&) (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==   by 0x10000780A: TestGCTAOptimize::test_unbinned_optimizer() (in /Users/jurgen/git/gammalib/test/.libs/test_CTA)
==61308==   by 0x1001EA926: GTestSuite::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)
==61308==   by 0x1001F1D8E: GTestSuites::run() (in /Users/jurgen/git/gammalib/src/.libs/libgamma.0.dylib)

I'm wondering whether the Conditional jump or move depends on uninitialised value(s) is related to the GCTAPointing?

#7 - 07/03/2015 09:24 AM - Knödlseder Jürgen

- % Done changed from 20 to 30

Tried to disable diffuse model caching without any impact.