GammaLib - Action #3324

Remove implicit conversion warnings

08/20/2020 11:56 AM - Knödlseder Jürgen

Status:	Closed	**Start date:**	08/20/2020
Priority:	Normal	**Due date:**	
Assigned To:	Knödlseder Jürgen	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:	2.0.0		

Description

There are a number of implicit conversion warnings in the generation of the Python interface on Mac OS X 10.14 (and alike) that should be removed by adding explicit conversions. Here a list of occurrences on Mac OS X 10.14 :

```
ammalib/fits_wrap.cpp:5989:53: warning: implicit conversion loses integer precision: 'long long' to 'long' [-Wshorten-64-to-32]
        PyList_SetItem(list, i, PyInt_FromLong((static_cast<long long*>(self->pixels())[i])));
                                ~~~~~~~~~~~~~~  ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1 warning generated.
gammalib/fits_wrap.cpp:5423:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int'
 [-Wshorten-64-to-32]
    int size = PyObject_Length(input);
        ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
gammalib/linalg_wrap.cpp:3543:12: warning: unused function 'var_tuple_to_index' [-Wunused-function]
static int var_tuple_to_index(PyObject *input, int *ptr, int dim) {
           ^
1 warning generated.
gammalib/linalg_wrap.cpp:3545:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int'
 [-Wshorten-64-to-32]
    int size = PyObject_Length(input);
        ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro
 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/linalg_wrap.cpp:3543:12: warning: unused function 'var_tuple_to_index' [-Wunused-function]
static int var_tuple_to_index(PyObject *input, int *ptr, int dim) {
           ^
2 warnings generated.
In file included from gammalib/model_wrap.cpp:5578:
../include/GModelSpectralMultiplicative.hpp:180:24: warning: implicit conversion loses integer precision:
 'std::__1::vector<GModelSpectral *, std::__1::allocator<GModelSpectral *> >::size_type' (aka 'unsigned long') to 'int'
 [-Wshorten-64-to-32]
   return (m_spectral.size());
   ~~~~~~  ~~~~~~~~~~~~^~~~~~
1 warning generated.
gammalib/numerics_wrap.cpp:4906:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int'
 [-Wshorten-64-to-32]
    int size = PyObject_Length(input);
        ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro
 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
1 warning generated.
gammalib/sky_wrap.cpp:5252:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int'
 [-Wshorten-64-to-32]
    int size = PyObject_Length(input);
        ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro
 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
```

```
1 warning generated.
gammalib/support_wrap.cpp:5243:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int'
 [-Wshorten-64-to-32]
        int size = PyObject_Length(input);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro
 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
1 warning generated.
gammalib/xspec_wrap.cpp:5064:12: warning: unused function 'var_tuple_to_index' [-Wunused-function]
static int var_tuple_to_index(PyObject *input, int *ptr, int dim) {
           ^
1 warning generated.
gammalib/xspec_wrap.cpp:5066:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int'
 [-Wshorten-64-to-32]
        int size = PyObject_Length(input);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro
 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/xspec_wrap.cpp:5064:12: warning: unused function 'var_tuple_to_index' [-Wunused-function]
static int var_tuple_to_index(PyObject *input, int *ptr, int dim) {
           ^
2 warnings generated.
gammalib/cta_wrap.cpp:5278:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int'
 [-Wshorten-64-to-32]
        int size = PyObject_Length(input);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro
 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
In file included from gammalib/cta_wrap.cpp:5539:
../inst/cta/include/GCTAModelSpatialMultiplicative.hpp:135:23: warning: implicit conversion loses integer precision:
 'std::__1::vector<GCTAModelSpatial *, std::__1::allocator<GCTAModelSpatial *> >::size_type' (aka 'unsigned long') to 'int'
 [-Wshorten-64-to-32]
    return (m_spatial.size());
    ~~~~~~  ~~~~~~~~~~~^~~~~~
2 warnings generated.
```

History

#1 - 08/20/2020 03:45 PM - Knödlseder Jürgen

There are also some warnings at ctools level:

```
/Users/jenkins/jenkins/install/integrate/gammalib/include/gammalib/GCTAModelSpatialMultiplicative.hpp:135:23: warning: implicit conversion loses
 integer precision: 'std::__1::vector<GCTAModelSpatial *, std::__1::allocator<GCTAModelSpatial *> >::size_type' (aka 'unsigned long') to 'int'
 [-Wshorten-64-to-32]
    return (m_spatial.size());
    ~~~~~~  ~~~~~~~~~~~^~~~~~
In file included from ctools/tools_wrap.cpp:5573:
/Users/jenkins/jenkins/workspace/ctools-integrate-os/label/macosx15/src/ctbin/ctbin.hpp:132:20: warning: implicit conversion loses integer precision:
 'std::__1::vector<GCTAEventCube, std::__1::allocator<GCTAEventCube> >::size_type' (aka 'unsigned long') to 'int' [-Wshorten-64-to-32]
    return m_cubes.size();
    ~~~~~~ ~~~~~~~~~^~~~~
3 warnings generated.
```

#2 - 11/17/2020 02:36 PM - Knödlseder Jürgen

- Status changed from New to In Progress

- Assigned To set to Knödlseder Jürgen

- % Done changed from 0 to 20

I added an explicit type conversion to the GCTAModelSpatialMultiplicative::components() method.

Furthermore, I replaced the GTypemaps.i file by the files typemap_GChatter.i and typemap_GTuple.i, including the typemaps only when needed. I also replaced the var_tuple_to_index static function in GTypemaps.i by a direct implementation in typemap_GTuple.i which removes the warnings.

There are still the following warnings

```
gammalib/sky_wrap.cpp:3759:9: warning: destination for this 'memset' call is a pointer to dynamic class 'GSkyDir'; vtable
      pointer will be overwritten [-Wdynamic-class-memaccess]
        memset(v_def,0,sizeof(Type));
        ~~~~~~ ^
gammalib/sky_wrap.cpp:3784:62: note: in instantiation of member function 'swig::traits_as<GSkyDir, swig::pointer_category>::as'
      requested here
    return traits_as<Type, typename traits<Type>::category>::as(obj, te);
                                                             ^
gammalib/sky_wrap.cpp:4347:15: note: in instantiation of function template specialization 'swig::as<GSkyDir>' requested here
        return swig::as<T>(item, true);
                     ^
gammalib/sky_wrap.cpp:4628:42: note: in instantiation of member function 'swig::SwigPySequence_Ref<GSkyDir>::operator GSkyDir'
      requested here
      seq->insert(seq->end(),(value_type)(*it));
                                          ^
gammalib/sky_wrap.cpp:4650:6: note: in instantiation of function template specialization
      'swig::assign<swig::SwigPySequence_Cont<GSkyDir>, std::__1::vector<GSkyDir, std::__1::allocator<GSkyDir> > >' requested
      here
          assign(swigpyseq, pseq);
          ^
gammalib/sky_wrap.cpp:4704:47: note: in instantiation of member function 'swig::traits_asptr_stdseq<std::__1::vector<GSkyDir,
      std::__1::allocator<GSkyDir> >, GSkyDir>::asptr' requested here
        return traits_asptr_stdseq<std::vector<T> >::asptr(obj, vec);
                                                       ^
gammalib/sky_wrap.cpp:3676:32: note: in instantiation of member function 'swig::traits_asptr<std::__1::vector<GSkyDir,
      std::__1::allocator<GSkyDir> > >::asptr' requested here
    return traits_asptr<Type>::asptr(obj, vptr);
                               ^
gammalib/sky_wrap.cpp:10593:18: note: in instantiation of function template specialization
      'swig::asptr<std::__1::vector<GSkyDir, std::__1::allocator<GSkyDir> > >' requested here
    res4 = swig::asptr(obj3, &ptr);
                 ^
gammalib/sky_wrap.cpp:3759:9: note: explicitly cast the pointer to silence this warning
        memset(v_def,0,sizeof(Type));
               ^
               (void*)
1 warning generated.
```

which need the implementation of a GSkyDirs container class for GSkyDir objects, and replacing

```
std::vector<GSkyDir> boundaries(const GSkyPixel& pixel, const int& step = 1) const;
```

by

```
GSkyDirs boundaries(const GSkyPixel& pixel, const int& step = 1) const;
```

#3 - 11/17/2020 04:45 PM - Knödlseder Jürgen

There is also the following compilation warning:

```
GVOClient.cpp:384:35: warning: 'tmpnam' is deprecated: This function is provided for compatibility reasons only.  Due to security concerns inherent in
 the design of tmpnam(3), it is highly recommended that you use mkstemp(3) instead. [-Wdeprecated-declarations]
    std::string samp_share = std::tmpnam(NULL);
                             ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/stdio.h:186:1: note: 'tmpnam' has been explicitly marked deprecated here
__deprecated_msg("This function is provided for compatibility reasons only.  Due to security concerns inherent in the design of tmpnam(3), it is highly
 recommended that you use mkstemp(3) instead.")
^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/cdefs.h:200:48: note: expanded from macro '__deprecated_msg'
        #define __deprecated_msg(_msg) __attribute__((__deprecated__(_msg)))
                                                   ^
GVOClient.cpp:457:35: warning: 'tmpnam' is deprecated: This function is provided for compatibility reasons only.  Due to security concerns inherent in
 the design of tmpnam(3), it is highly recommended that you use mkstemp(3) instead. [-Wdeprecated-declarations]
    std::string samp_share = std::tmpnam(NULL);
                             ^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/stdio.h:186:1: note: 'tmpnam' has been explicitly marked deprecated here
__deprecated_msg("This function is provided for compatibility reasons only.  Due to security concerns inherent in the design of tmpnam(3), it is highly
 recommended that you use mkstemp(3) instead.")
^
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include/sys/cdefs.h:200:48: note: expanded from macro '__deprecated_msg'
        #define __deprecated_msg(_msg) __attribute__((__deprecated__(_msg)))
                                                   ^
2 warnings generated.
```

#4 - 11/17/2020 05:43 PM - Knödlseder Jürgen

- % Done changed from 20 to 30

I added a gammalib::tmpnam() function to replace the deprecated std::tmpnam() method in GVOClient.

#5 - 11/17/2020 07:03 PM - Knödlseder Jürgen

There is still the following SWIG warning:

```
GResponseVectorCache.i:41: Warning 453: Can't apply (GVector *OUTPUT). No typemaps are defined.
```

and the following

```
gammalib/fits_wrap.cpp:5427:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
    int size = PyObject_Length(input);
        ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/fits_wrap.cpp:6144:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
    int size = PyObject_Length(input);
        ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/fits_wrap.cpp:22618:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:22752:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[2]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:22906:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[3]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:23080:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[4]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:23237:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:24607:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:24741:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[2]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:24895:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[3]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:25069:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[4]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:25226:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:26596:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:26730:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[2]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:26884:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[3]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:27058:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[4]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:27215:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:28585:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:28719:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[2]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:28873:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[3]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:29047:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[4]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:29204:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~
```

```
gammalib/fits_wrap.cpp:30574:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:30708:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[2]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:30862:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[3]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:31036:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[4]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:31193:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:32563:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:32697:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[2]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:32851:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[3]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:33025:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[4]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:33182:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:34552:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:34686:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[2]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:34840:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[3]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:35014:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[4]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:35171:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:36541:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:36675:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[2]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:36829:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[3]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:37003:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[4]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:37160:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:38530:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:38664:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[2]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:38818:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[3]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:38992:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[4]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
gammalib/fits_wrap.cpp:39149:15: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
  int num = PyList_Size(swig_obj[1]);
      ~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
```

```
gammalib/fits_wrap.cpp:46059:22: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
        int size = PyObject_Length(argv[1]);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/fits_wrap.cpp:46094:22: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
        int size = PyObject_Length(argv[1]);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/fits_wrap.cpp:46129:22: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
        int size = PyObject_Length(argv[1]);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/fits_wrap.cpp:52437:22: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
        int size = PyObject_Length(argv[1]);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/fits_wrap.cpp:52470:22: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
        int size = PyObject_Length(argv[1]);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
52 warnings generated.


gammalib/numerics_wrap.cpp:17067:18: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
      int size = PyObject_Length(swig_obj[1]);
          ~~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/numerics_wrap.cpp:17149:18: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
      int size = PyObject_Length(swig_obj[1]);
          ~~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/numerics_wrap.cpp:20059:18: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
      int size = PyObject_Length(swig_obj[1]);
          ~~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
gammalib/numerics_wrap.cpp:20141:18: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
      int size = PyObject_Length(swig_obj[1]);
          ~~~~   ^~~~~~~~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
4 warnings generated.


../include/GResponseVectorCache.hpp:127:25: warning: implicit conversion loses integer precision: 'std::__1::vector<std::__1::basic_string<char>,
 std::__1::allocator<std::__1::basic_string<char> > >::size_type' (aka 'unsigned long') to 'int' [-Wshorten-64-to-32]
    return (m_cache_ids.size());
    ~~~~~~  ~~~~~~~~~~~~~^~~~~~
1 warning generated.


gammalib/sky_wrap.cpp:5256:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
        int size = PyObject_Length(input);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
1 warning generated.
```

```
gammalib/support_wrap.cpp:5247:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
        int size = PyObject_Length(input);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
1 warning generated.


../include/GResponseVectorCache.hpp:127:25: warning: implicit conversion loses integer precision: 'std::__1::vector<std::__1::basic_string<char>,
 std::__1::allocator<std::__1::basic_string<char> > >::size_type' (aka 'unsigned long') to 'int' [-Wshorten-64-to-32]
    return (m_cache_ids.size());
    ~~~~~~  ~~~~~~~~~~~~~^~~~~~
1 warning generated.


../include/GResponseVectorCache.hpp:127:25: warning: implicit conversion loses integer precision: 'std::__1::vector<std::__1::basic_string<char>,
 std::__1::allocator<std::__1::basic_string<char> > >::size_type' (aka 'unsigned long') to 'int' [-Wshorten-64-to-32]
    return (m_cache_ids.size());
    ~~~~~~  ~~~~~~~~~~~~~^~~~~~
gammalib/cta_wrap.cpp:5361:20: warning: implicit conversion loses integer precision: 'Py_ssize_t' (aka 'long') to 'int' [-Wshorten-64-to-32]
        int size = PyObject_Length(input);
            ~~~~   ^~~~~~~~~~~~~~~~~~~~~~
/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7/abstract.h:434:25: note: expanded from macro 'PyObject_Length'
#define PyObject_Length PyObject_Size
                        ^
2 warnings generated.


../include/GResponseVectorCache.hpp:127:25: warning: implicit conversion loses integer precision: 'std::__1::vector<std::__1::basic_string<char>,
 std::__1::allocator<std::__1::basic_string<char> > >::size_type' (aka 'unsigned long') to 'int' [-Wshorten-64-to-32]
    return (m_cache_ids.size());
    ~~~~~~  ~~~~~~~~~~~~~^~~~~~
1 warning generated.
```

#6 - 11/18/2020 07:59 AM - Knödlseder Jürgen

- Tracker changed from Bug to Action

- % Done changed from 30 to 50

Except of the conversion warning in HealPix.i, all conversion warnings were successfully removed, also one that occurred in the ctbin.hpp interface.

The next step is to add a GSkyDirs class.

#7 - 11/18/2020 10:33 PM - Knödlseder Jürgen

- Status changed from In Progress to Pull request

- % Done changed from 50 to 90

There are no more compile warnings in GammaLib and ctools.

#8 - 11/18/2020 10:59 PM - Knödlseder Jürgen

- Status changed from Pull request to Closed

- % Done changed from 90 to 100

Merged into devel.

#9 - 11/20/2020 03:28 PM - Knödlseder Jürgen

- Status changed from Closed to In Progress

- % Done changed from 100 to 90

I found another one after an Homebrew upgrade:

```
gammalib/fits_wrap.cpp:6033:53: warning: implicit conversion loses integer precision: 'long long' to 'long' [-Wshorten-64-to-32]
        PyList_SetItem(list, i, PyInt_FromLong((static_cast<long long*>(self->pixels())[i])));
                                ~~~~~~~~~~~~~~  ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1 warning generated.
```

#10 - 12/02/2020 10:45 AM - Knödlseder Jürgen

- Status changed from In Progress to Closed

- % Done changed from 90 to 100

I also fixed this warning, code merged into devel.