GammaLib - Bug #2697

gammalib make checks fails on MacOS

10/10/2018 05:39 PM - Bonnefoy Simon

Status:	New	**Start date:**	10/10/2018
Priority:	Normal	**Due date:**	
Assigned To:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			

Description

Make check fails for test_python.sh and test_examples.sh on Mac OS.
I am running on El Capitan version 10.11.6.
It seems that there is a mismatch between the architecture of some libraries (see attached log files).
I could not find options in the config file to set a different architecture.

Has someone already seen that?

History

#1 - 10/10/2018 05:44 PM - Tibaldo Luigi

Did you have a former installation on the same computer? You ma try to do 'make clean' and restart from scratch.
It's not always clear which python wrapper is used during check (local or system), it depends on your path configuration. You may also try to do 'sudo make install' before running the test suite.

#2 - 10/11/2018 11:10 AM - Bonnefoy Simon

Thanks for the advices, Luigi.
I tried both, and it does not work. I still get the same error.
BTW, in the config output, it says that python wrapper(s) is missing, Requires swig for wrapper generation.
But swing is found on my computer.

#3 - 10/11/2018 11:35 AM - Tibaldo Luigi

Thanks Simon. The config log says that swig is found, the WARNING just means Python wrappers are not already present and will need to be generated.
Could you please check the content of 'pyext/build' to make sure the wrappers and libraries have been generated?
Could you also please type 'dev/testreport.py' and send the output?

#4 - 10/11/2018 03:45 PM - Bonnefoy Simon

Hi Luigi,

Ok, thanks for the info!

here is the containt of the pyext folder

```
pyext/build/gammalib:
total 24336
drwxr-xr-x  20 bonnefoy  staff      680 Oct 11 10:48 ./
drwxr-xr-x   5 bonnefoy  staff      170 Oct 11 10:48 ../
-rwxr-xr-x   1 bonnefoy  staff   441272 Oct 11 11:04 _app.so*
-rwxr-xr-x   1 bonnefoy  staff    63672 Oct 11 11:04 _base.so*
-rwxr-xr-x   1 bonnefoy  staff   712740 Oct 11 11:04 _com.so*
-rwxr-xr-x   1 bonnefoy  staff  2253924 Oct 11 11:04 _cta.so*
-rwxr-xr-x   1 bonnefoy  staff  1483392 Oct 11 11:04 _fits.so*
-rwxr-xr-x   1 bonnefoy  staff   482976 Oct 11 11:04 _lat.so*
```

```
-rwxr-xr-x   1 bonnefoy  staff   469180 Oct 11 11:04 _linalg.so*
-rwxr-xr-x   1 bonnefoy  staff  2067092 Oct 11 11:04 _model.so*
-rwxr-xr-x   1 bonnefoy  staff   164084 Oct 11 11:04 _mwl.so*
-rwxr-xr-x   1 bonnefoy  staff   482408 Oct 11 11:04 _numerics.so*
-rwxr-xr-x   1 bonnefoy  staff   970748 Oct 11 11:04 _obs.so*
-rwxr-xr-x   1 bonnefoy  staff   226544 Oct 11 11:04 _opt.so*
-rwxr-xr-x   1 bonnefoy  staff   977264 Oct 11 11:04 _sky.so*
-rwxr-xr-x   1 bonnefoy  staff   606284 Oct 11 11:04 _support.so*
-rwxr-xr-x   1 bonnefoy  staff   229408 Oct 11 11:04 _test.so*
-rwxr-xr-x   1 bonnefoy  staff   109308 Oct 11 11:04 _vo.so*
-rwxr-xr-x   1 bonnefoy  staff   313244 Oct 11 11:04 _xml.so*
-rwxr-xr-x   1 bonnefoy  staff   364808 Oct 11 11:04 _xspec.so*

pyext/build/lib.macosx-10.11-intel-2.7:
total 0
drwxr-xr-x   3 bonnefoy  staff   102 Oct 11 10:44 ./
drwxr-xr-x   5 bonnefoy  staff   170 Oct 11 10:48 ../
drwxr-xr-x  39 bonnefoy  staff  1326 Oct 11 11:04 gammalib/

pyext/build/temp.macosx-10.11-intel-2.7:
total 0
drwxr-xr-x   3 bonnefoy  staff  102 Oct 11 10:44 ./
drwxr-xr-x   5 bonnefoy  staff  170 Oct 11 10:48 ../
drwxr-xr-x  20 bonnefoy  staff  680 Oct 11 10:48 gammalib/
```

And here is the output of the dev/testreport.py

```
bonnefoy@s68dyn20:~/software/ctools/devel/gammalib$ dev/testreport.py
Traceback (most recent call last):
  File "dev/testreport.py", line 22, in <module>
    import gammalib
  File "/usr/local/gamma/lib/python2.7/site-packages/gammalib/__init__.py", line 21, in <module>
    from gammalib.app import *
  File "/usr/local/gamma/lib/python2.7/site-packages/gammalib/app.py", line 17, in <module>
    _app = swig_import_helper()
  File "/usr/local/gamma/lib/python2.7/site-packages/gammalib/app.py", line 16, in swig_import_helper
    return importlib.import_module('_app')
  File "/System/Library/Frameworks/Python.framework/Versions/2.7/lib/python2.7/importlib/__init__.py", line 37, in import_module
    __import__(name)
ImportError: No module named _app
```

#5 - 10/12/2018 08:28 AM - Tibaldo Luigi

Hi Simon,
it looks like the Python wrappers are not generated. Could you please post the output from './configure' and 'make'?

#6 - 10/12/2018 01:28 PM - Bonnefoy Simon

- File configure_output.log added

Hi Luigi,

I attach both outputs here.
The strange thing is that if I re-run ./configure after issuing a "sudo make install" command,
the python wrappers seem to be present (see configure_make_install_output.log).
But re-compiling the code does not change anything.

#7 - 10/12/2018 01:35 PM - Bonnefoy Simon

- File make_output.log added

- File configure_output.log added

- File configure_make_install_output.log added

Sorry, all the files did not make it in the previous message.

#8 - 10/31/2018 01:53 PM - Knödlseder Jürgen

- Project changed from ctools to GammaLib

#9 - 10/31/2018 03:14 PM - Knödlseder Jürgen

Sorry for coming in so late in the discussion.

The Python wrappers are correctly built, you see this in the make_output.log file:

```
cc -fno-strict-aliasing -fno-common -dynamic -arch i386 -arch x86_64 -g -Os -pipe -fno-common -fno-strict-aliasing -fwrapv -DENABLE_DTRACE
 -DMACOSX -DNDEBUG -Wall -Wstrict-prototypes -Wshorten-64-to-32 -DNDEBUG -g -fwrapv -Os -Wall -Wstrict-prototypes -DENABLE_DTRACE
 -arch i386 -arch x86_64 -pipe -I../include -I../inst/mwl/include -I../inst/cta/include -I../inst/lat/include -I../inst/com/include
 -I/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7 -c gammalib/com_wrap.cpp -o
 build/temp.macosx-10.11-intel-2.7/gammalib/com_wrap.o
c++ -bundle -undefined dynamic_lookup -arch i386 -arch x86_64 -Wl,-F. build/temp.macosx-10.11-intel-2.7/gammalib/com_wrap.o -L../src/.libs
 -L../src/.libs -L/usr/local/gamma/lib -lgamma -lcfitsio -ledit -lcurses -o build/lib.macosx-10.11-intel-2.7/gammalib/_com.so
 -headerpad_max_install_names
Created "build/gammalib" directory.
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_app.so -> build/gammalib/_app.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_base.so -> build/gammalib/_base.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_com.so -> build/gammalib/_com.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_cta.so -> build/gammalib/_cta.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_fits.so -> build/gammalib/_fits.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_lat.so -> build/gammalib/_lat.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_linalg.so -> build/gammalib/_linalg.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_model.so -> build/gammalib/_model.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_mwl.so -> build/gammalib/_mwl.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_numerics.so -> build/gammalib/_numerics.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_obs.so -> build/gammalib/_obs.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_opt.so -> build/gammalib/_opt.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_sky.so -> build/gammalib/_sky.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_support.so -> build/gammalib/_support.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_test.so -> build/gammalib/_test.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_vo.so -> build/gammalib/_vo.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_xml.so -> build/gammalib/_xml.so
Copied build/lib.macosx-10.11-intel-2.7/gammalib/_xspec.so -> build/gammalib/_xspec.so
```

The object files are there, but something goes wrong with the architecture:

```
ImportError: dlopen(/Users/bonnefoy/software/ctools/devel/gammalib/pyext/build/gammalib/_app.so, 2): no suitable image found.  Did find:
    /Users/bonnefoy/software/ctools/devel/gammalib/pyext/build/gammalib/_app.so: mach-o, but wrong architecture
```

I noticed that you have intel as architecture in your build/temp.macosx-10.11-intel-2.7/ filename, while I have x86_64 in build/temp.macosx-10.11-x86_64-2.7. As you see in the make_output.log file, the wrappers are compiled for

```
-arch i386 -arch x86_64
```

What do you see when you type

```
lipo -info build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o
```

#10 - 10/31/2018 03:18 PM - Knödlseder Jürgen

You may also try:

```
$ python
>>> from distutils import util
>>> print(util.get_platform())
```

I get macosx-10.11-x86_64, I suppose you get macosx-10.11-intel.

#11 - 10/31/2018 03:19 PM - Knödlseder Jürgen

And finally you may try

```
$ lipo -info `which python`
```

which should give you the architecture of your Python executable.

#12 - 10/31/2018 03:23 PM - Knödlseder Jürgen

Just read this on https://python.readthedocs.io/fr/stable/distutils/apiref.html:

For universal binary builds on Mac OS X the architecture value reflects the universal binary status instead of the architecture of the current processor. For 32-bit universal binaries the architecture is fat, for 64-bit universal binaries the architecture is fat64, and for 4-way universal binaries the architecture is universal. Starting from Python 2.7 and Python 3.2 the architecture fat3 is used for a 3-way universal build (ppc, i386, x86_64) and intel is used for a universal build with the i386 and x86_64 architectures

Examples of returned values on Mac OS X:

macosx-10.3-ppc
macosx-10.3-fat
macosx-10.5-universal
macosx-10.6-intel

You may try

$ ./configure --with-universal-archs=intel --enable-universalsdk

#13 - 11/01/2018 04:37 PM - Bonnefoy Simon

Hi Jürgen,

thanks for the help.

Here are the outputs of the commands you requested:

bonnefoy@znb44:~/software/ctools/devel/gammalib/pyext$ lipo -info build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o
Architectures in the fat file: build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o are: i386 x86_64

From python:

bonnefoy@znb44:~/software/ctools/devel/gammalib/pyext$ python
Python 2.7.10 (default, Oct 23 2015, 19:19:21)
[GCC 4.2.1 Compatible Apple LLVM 7.0.0 (clang-700.0.59.5)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> from distutils import util
>>> print(util.get_platform())
macosx-10.11-intel

and finally:

bonnefoy@znb44:~/software/ctools/devel/gammalib/pyext$ lipo -info `which python`

Architectures in the fat file: /usr/bin/python are: i386 x86_64

I tried to re-compile with the ./configure --with-universal-archs=intel --enable-universalsdk
but it seems that the architecture configuration has remained the same.

#14 - 11/01/2018 09:11 PM - Knödlseder Jürgen

Forgot one: what do you see when you type

file pyext/build/lib.macosx-10.11-intel-2.7/gammalib/_app.so

or

file /Users/bonnefoy/software/ctools/devel/gammalib/pyext/build/gammalib/_app.so

You may also try

```
$ python
>>> import platform
>>> platform.architecture()
```

to see what Python architecture starts up when you launch Python.

#15 - 11/02/2018 05:18 PM - Bonnefoy Simon

Hi Jürgen,

here it is:

bonnefoy@znb44:~/software/ctools/devel/gammalib$ file pyext/build/lib.macosx-10.11-intel-2.7/gammalib/_app.so
pyext/build/lib.macosx-10.11-intel-2.7/gammalib/_app.so: Mach-O bundle i386

Also, searching on internet, I found that this problem might arise if some software available in macports were not properly installed, i.e., not using macport. I don't know whether this can have an impact on the gammalib installation.
I will take a look.

#16 - 11/02/2018 06:22 PM - Knödlseder Jürgen

That's interesting. While app_wrap.o is i386 x86_64, _app.so is only i386. I assume that

```
$ python
>>> import platform
>>> platform.architecture()
```

will show that the 64 Bit version of Python is started.

Maybe the issue is cfitsio. I think when _app.so is built it checks the architecture of the dependency libraries. If your cfitsio is only 32 Bit this may explain the issue. Or any other of the dependency libraries (-lgamma -lcfitsio -ledit -lcurses).

In gammalib/doc/dev/tn there is a Technote gammalib_tn0001_macosx.tex that I wrote some time ago, and that explains the issues with the different architectures. It is certainly a bit outdated. But it may help to solve your issue.

You should check using the file command the architecture of all dependency libraries, and see whether you have a combination that cannot work.

#17 - 11/05/2018 05:30 PM - Knödlseder Jürgen

Here is what is written in the man ld documentation:

The linker accepts universal (multiple-architecture) input files, but always creates a "thin" (single-architecture), standard Mach-O output file. The architecture for the output file is specified using the -arch option. If this option is not used, ld attempts to determine the output architecture by examining the object files in command line order. The first "thin" architecture determines that of the output file. If no input object file is a "thin" file, the native 32-bit architecture for the host is used.

Don't think this is relevant, since c++ seems to call the linker for each architecture.

#18 - 11/05/2018 05:36 PM - Knödlseder Jürgen

Here is what I see on Mac OS X 10.11:

```
building '_app' extension
creating build
creating build/temp.macosx-10.11-intel-2.7
creating build/temp.macosx-10.11-intel-2.7/gammalib
cc -fno-strict-aliasing -fno-common -dynamic -arch i386 -arch x86_64 -g -Os -pipe -fno-common -fno-strict-aliasing -fwrapv -DENABLE_DTRACE
 -DMACOSX -DNDEBUG -Wall -Wstrict-prototypes -Wshorten-64-to-32 -DNDEBUG -g -fwrapv -Os -Wall -Wstrict-prototypes -DENABLE_DTRACE
 -arch i386 -arch x86_64 -pipe -I../include -I../inst/mwl/include -I../inst/cta/include -I../inst/lat/include -I../inst/com/include
 -I/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7 -c gammalib/app_wrap.cpp -o
 build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o
In file included from gammalib/app_wrap.cpp:3102:
In file included from ../include/GException.hpp:36:
../include/GXmlElement.hpp:152:13: warning: implicit conversion loses integer precision: 'size_type' (aka 'unsigned long') to 'int' [-Wshorten-64-to-32]
    return (m_attr.size());
    ~~~~~~  ^~~~~~~~~~~~~~
1 warning generated.
```

```
creating build/lib.macosx-10.11-intel-2.7
creating build/lib.macosx-10.11-intel-2.7/gammalib
c++ -bundle -undefined dynamic_lookup -arch i386 -arch x86_64 -Wl,-F. build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o -L../src/.libs
 -L../src/.libs -lgamma -lcfitsio -ledit -lcurses -o build/lib.macosx-10.11-intel-2.7/gammalib/_app.so -headerpad_max_install_names
ld: warning: ignoring file ../src/.libs/libgamma.dylib, file was built for x86_64 which is not the architecture being linked (i386): ../src/.libs/libgamma.dylib
ld: warning: ignoring file /usr/local/lib/libcfitsio.dylib, file was built for x86_64 which is not the architecture being linked (i386):
 /usr/local/lib/libcfitsio.dylib
```

Here I see warnings about missing architecture. Do you see the same kind of errors on your side?

#19 - 11/05/2018 05:52 PM - Knödlseder Jürgen

Here is what I get in terms of architecture:

```
$ file pyext/build/gammalib/_app.so
pyext/build/gammalib/_app.so: Mach-O universal binary with 2 architectures
pyext/build/gammalib/_app.so (for architecture i386):    Mach-O bundle i386
pyext/build/gammalib/_app.so (for architecture x86_64):    Mach-O 64-bit bundle x86_64
$ file pyext/build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o
pyext/build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o: Mach-O universal binary with 2 architectures
pyext/build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o (for architecture i386):    Mach-O object i386
pyext/build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o (for architecture x86_64):    Mach-O 64-bit object x86_64
$ file src/.libs/libgamma.dylib
src/.libs/libgamma.dylib: Mach-O 64-bit dynamically linked shared library x86_64
$ file /usr/local/lib/libcfitsio.dylib
/usr/local/lib/libcfitsio.dylib: Mach-O 64-bit dynamically linked shared library x86_64
```

So I looks like the only difference is that you do not get the x86_64 in the _app.so bundle.

#20 - 11/06/2018 09:48 AM - Bonnefoy Simon

I don't get the warning for the missing architecture.
Here is what I get:

```
building '_app' extension
creating build
creating build/temp.macosx-10.11-intel-2.7
creating build/temp.macosx-10.11-intel-2.7/gammalib
cc -fno-strict-aliasing -fno-common -dynamic -arch i386 -arch x86_64 -g -Os -pipe -fno-common -fno-strict-aliasing -fwrapv -DENABLE_DTRACE
 -DMACOSX -DNDEBUG -Wall -Wstrict-prototypes -Wshorten-64-to-32 -DNDEBUG -g -fwrapv -Os -Wall -Wstrict-prototypes -DENABLE_DTRACE
 -arch i386 -arch x86_64 -pipe -I../include -I../inst/mwl/include -I../inst/cta/include -I../inst/lat/include -I../inst/com/include
 -I/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7 -c gammalib/app_wrap.cpp -o
 build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o
creating build/lib.macosx-10.11-intel-2.7
```

```
creating build/lib.macosx-10.11-intel-2.7/gammalib
c++ -bundle -undefined dynamic_lookup -arch i386 -arch x86_64 -Wl,-F. build/temp.macosx-10.11-intel-2.7/gammalib/app_wrap.o -L../src/.libs
 -L../src/.libs -L/usr/local/gamma/lib -lgamma -lcfitsio -ledit -lcurses -o build/lib.macosx-10.11-intel-2.7/gammalib/_app.so
 -headerpad_max_install_names
```

All the libraries involved in the building of _app.so are available for both x86_64 and i386.

Files

test_examples.sh.log	961 Bytes	10/10/2018	Bonnefoy Simon
test_python.sh.log	957 Bytes	10/10/2018	Bonnefoy Simon
config.log	55.9 KB	10/10/2018	Bonnefoy Simon
configure_output.log	12.2 KB	10/12/2018	Bonnefoy Simon
make_output.log	229 KB	10/12/2018	Bonnefoy Simon
configure_output.log	12.2 KB	10/12/2018	Bonnefoy Simon
configure_make_install_output.log	12.3 KB	10/12/2018	Bonnefoy Simon