

GammaLib - Bug #2724

Error in csh when sourcing gammalib-init.csh and ctools-init.csh

11/08/2018 11:23 AM - Specovius Andreas

Status:	Closed	Start date:	11/08/2018
Priority:	High	Due date:	
Assigned To:	Knödseder Jürgen	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	1.6.1		
Description When sourcing gammalib-init.csh and ctools-init.csh for initialising the gammalib/ctools module an error occurs for csh-users. Obviously the internally sourced <i>gammalib-setup</i> and <i>ctools-setup</i> scripts are not coded appropriately for the csh. This error arose from the modifications done in issue #2420 when the following line was removed: #!/bin/sh			
Related issues: Related to GammaLib - Bug # 2420: Fix MANPATH on Mac OS X			
		Closed	03/26/2018

History

- #1 - 11/08/2018 11:24 AM - Specovius Andreas**
- Related to Bug #2420: Fix MANPATH on Mac OS X added
- #2 - 06/05/2019 03:44 AM - Knödseder Jürgen**
- Priority changed from Normal to High

Pat Romano encountered the same issue for the new release 1.6:

I'm having issues with the new 1.6.0 installation. I dowloaded the dmg and unpacked it. When I initialize it with

```
setenv GAMMALIB /usr/local/gamma
source $GAMMALIB/bin/gammalib-init.csh
```

I get:

this_script=gammalib-setup: Command not found.
Bad : modifier in \$ (").

I checked the old (v1.5.2) gammalib-init.csh and it's the very same (with diff and substituting).

```
I get the same error when I initialize CTOOLS with
setenv CTOOLS /usr/local/gamma
source $CTOOLS/bin/ctools-init.csh
```

I'm using A Mac OS 10.10.5 ad had no issues before.
Any suggestions?

Easy solution is to put back

```
#!/bin/sh
```

in the first line of

```
$GAMMALIB/bin/gammalib-setup
```

\$CTOOLS/bin/ctools-setup

which however will lead to the issues reported in #2420. It needs to be checked whether a solution can be found that solves both issues.

#3 - 06/05/2019 05:01 PM - Knödseder Jürgen

- Status changed from New to In Progress
- Assigned To set to Knödseder Jürgen
- % Done changed from 0 to 10

I put back #!/bin/sh in the \$GAMMALIB/bin/gammlib-setup file and also removed the setting of the (DY)LD_LIBRARY_PATH environment variables.

The release check pipeline test work well, except for the clang31 compiler test that ends with the following error in the

make distcheck

step

Making installcheck in test

make[2]: Entering directory

`/home/jenkins/jenkins/workspace/gammlib-release-compilers/CONF_GCC/clang31/arch/i386/gammlib-1.6.0/_build/test'

Traceback (most recent call last):

File "<string>", line 1, in <module>

File

```
from gammlib.app import *
```

File

```
_app = swig_import_helper()
```

File

```
_mod = imp.load_module('_app', fp, pathname, description)
```

ImportError: libgamma.so.7: cannot open shared object file: No such file or directory

*** gammlib unit test failure!

Note that this happens in the following code section of the test/Makefile.am file:

```
# Tests to be done on installed version
```

```
if WITH_PYTHON
```

```
installcheck-local:
```

```
  @export GAMMALIB=$(DESTDIR)$ (prefix); \
```

```
  . $(DESTDIR)$ (prefix)/bin/gammlib-init.sh; \
```

```
  python -c 'import gammlib; gammlib.test()'; \
```

```
  if test "x$$?" != "x0"; then \
```

```
    echo "**** gammlib unit test failure!"; \
```

```
    exit 1; \
```

```
  fi
```

```
endif
```

#4 - 06/11/2019 08:25 PM - Knödseder Jürgen

- *Status changed from In Progress to Closed*
- *Target version set to 1.6.1*
- *% Done changed from 10 to 100*

The change was merged into the 1.6.1 release and also in the devel branch.