

ctools - Bug #2305

csiactobs ignoring "datapath" parameter when environment variable "VHEFITS" is set

02/02/2018 12:39 PM - Specovius Andreas

Status:	Closed	Start date:	02/02/2018
Priority:	Low	Due date:	
Assigned To:	Specovius Andreas	% Done:	100%
Category:		Estimated time:	2.00 hours
Target version:	1.6.0		
Description			
If environment variable VHEFITS is set, the user can not select another master.json file via setting datapath to the corresponding path.			
Script csiactdata and csiactobs (others not testet) ignore the user input for the hidden parameter datapath in case VHEFITS is set on the machine.			
I think that the preferred behaviour should be the other way around: datapath should overwrite the VHEFITS path.			

History

#1 - 02/02/2018 01:51 PM - Knödseder Jürgen

Feel free to change the code as needed. Since Michael Meyer has left and I have no access to a VHE data store it's difficult for me to judge how the code should actually behave.

#2 - 02/04/2018 03:02 PM - Specovius Andreas

- Assigned To set to Specovius Andreas

#3 - 02/13/2018 10:34 AM - Specovius Andreas

- Status changed from New to Pull request

- Target version set to 1.6.0

- % Done changed from 0 to 100

- Estimated time changed from 1.00 to 2.00

I finalised an implementation that favours the user input for <datapath> over the environment variable VHEFITS.

In ctools I modified the scripts that have a datapath parameter (namely csiactdata, csiactobs, csfindobs).

In case the parameter <datapath> has been changed by the user its value will be used instead of the VHEFITS value.

As I could not figure out how to find out that an GApplicationPar value has been changed by user after it was loaded from the corresponding p-file I added a tag called "m_deflt" which states whether the parameter value is on its default value (= that one loaded from a p-file) or not. Correspondingly also is_default() and set_default() are implemented. Any call of GApplicationPar :set_value() disables the default tag. Only the GApplicationPars::parse() method enables it.

I testet all three scripts from terminal and at least csiactdata from within python.

The implementations can be found in my gamma lib and ctools repositories on branch 2305-datapath-before-vhefits.

#4 - 02/14/2018 05:42 PM - Knödseder Jürgen

The problem with this implementation is that it's not traceable by the user what is done by the software, since the user may just type <enter> when queried for the datapath, but the script may use the \$VHEFITS variable.

What other tools are doing, and what is clearer for the user, is to set by default the datapath parameter to NONE. This can be easily checked by the tool, and if the parameter is NONE, it can use the \$VHEFITS environment variable.

Would this scheme work for you?

#5 - 02/15/2018 10:30 AM - Specovius Andreas

user#3 wrote:

What other tools are doing, and what is clearer for the user, is to set by default the datapath parameter to NONE. This can be easily checked by the tool, and if the parameter is NONE, it can use the \$VHEFITS environment variable.

Would this scheme work for you?

Yes, this sounds like a good solution!

So if I understood correctly the default for datapath parameter is to set to "NONE" and the learning capability should be removed from its mode to be only "q".

You can take a look at my ctools repo where I just pushed this implementation.

I encountered a small problem: I had to re-set the mode for datapath before querying it because its mode (originally "q") was changed during loading of the p-file via the call of stop_query() called from set_value() to hidden. Hence a query did no longer work. I don't think this behaviour for "q"-mode parameters is intended because if chosen auto mode which then sets the "q" mode the behaviour is different.

#6 - 06/05/2018 04:50 PM - Knödseder Jürgen

The setting of the q capability to h is not intended, I need to change some code in GammaLib to fix this issue. I created an issue for that: #2495.

However, it's not really clear to me why you do not set the parameter to a. Why would you not want that the parameter value is "learned"?

#7 - 06/06/2018 10:36 AM - Knödseder Jürgen

I think the following should do the job:

```
# Query datapath. If the parameter is not NONE then use it, otherwise
# use the datapath from the VHEFITS environment variable
datapath = self['datapath'].string()
```

```
if gammalib.toupper(datapath) != 'NONE':  
    self._datapath = datapath  
else:  
    self._datapath = os.getenv('VHEFITS', '')
```

#8 - 06/06/2018 12:28 PM - Knödseder Jürgen

- *Status changed from Pull request to Closed*

I merged the code into devel, a bit simplified with respect to what you provided. Please let me know if it works and fits your needs.

#9 - 06/18/2018 10:47 AM - Specovius Andreas

It works fine for me!