

example4

May 7, 2018

1 Using custom arguments

By default the recipes expect the following inputs:

- night_name - the string containing the path inside the raw data folder
- files - the list of filenames (at location: `raw data folder/night_name/`)

In some cases different or additional arguments are required. Examples of recipes where this is true are: - cal_BADPIX_spirou.py - cal_CCF_E2DS_spirou.py - cal_DRIFT_E2DS_spirou.py - cal_DRIFTPEAK_E2DS_spirou.py - cal_WAVE_E2DS_spirou.py

Below we describe how to a slightly modified set up routine to deal with any number of custom arguments required.

1.1 Example with cal_BADPIX_spirou

The recipe cal_BADPIX takes two arguments (after the night_name), this could use the default format requiring two and only two arguments pushed into the list of files.

This is what the user is expected to input:

```
cal_BADPIX_spirou.py [night_directory] [flat_flat_*.fits] [dark_dark_*.fits]
```

And below is the code to deal with these two arguments:

```
In [6]: # imports and variable definition
```

```
from __future__ import division
import numpy as np
import os
```

```
from SpirouDRS import spirouCDB
from SpirouDRS import spirouConfig
from SpirouDRS import spirouCore
from SpirouDRS import spirouImage
from SpirouDRS import spirouStartup
```

```
# =====
# Define variables
# =====
```

```

# Name of program
__NAME__ = 'cal_BADPIX_spirou.py'
# Get version and author
__version__ = spirouConfig.Constants.VERSION()
__author__ = spirouConfig.Constants.AUTHORS()
__date__ = spirouConfig.Constants.LATEST_EDIT()
__release__ = spirouConfig.Constants.RELEASE()
# Get Logging function
WLOG = spirouCore.wlog
# Get plotting functions
sPlt = spirouCore.sPlt

# these are the arguments of the main function (taken out to show below)
# they are all set to None by default - and hence will cause an exception
# unless the user enters these required values (all strings)
night_name=None
darkfile=None
flatfile=None

# force the runtime arguments (notebook only)
import sys
sys.argv = ['cal_BADPIX_spirou.py', '20170710', 'flat_flat02f10.fits', 'dark_dark02d406.

```

First (as always) we run the **Begin()** function to load variables from "config.py":

```

In [12]: # -----
# Set up
# -----
# get parameters from config files/run time args/load paths + calibdb
p = spirouStartup.Begin()

15:48:26.0 - || *****
15:48:26.0 - || * SPIROU @(#) Geneva Observatory (0.1.030)
15:48:26.0 - || *****
15:48:26.0 - ||(dir_data_raw)      DRS_DATA_RAW=/scratch/Projects/spirou_py3/data/raw
15:48:26.0 - ||(dir_data_reduc)    DRS_DATA_REDUCE=/scratch/Projects/spirou_py3/data/reduced
15:48:26.0 - ||(dir_calib_db)      DRS_CALIB_DB=/scratch/Projects/spirou_py3/data/calibDB
15:48:26.0 - ||(dir_data_msg)      DRS_DATA_MSG=/scratch/Projects/spirou_py3/data/msg
15:48:26.0 - ||(print_level)      PRINT_LEVEL=all          %(error/warning/info/all)
15:48:26.0 - ||(log_level)         LOG_LEVEL=all          %(error/warning/info/all)
15:48:26.0 - ||(plot_graph)        DRS_PLOT=1            %(def/undef/trigger)
15:48:26.0 - ||(used_date)         DRS_USED_DATE=undefined
15:48:26.0 - ||(working_dir)       DRS_DATA_WORKING=/scratch/Projects/spirou_py3/data/tmp
15:48:26.0 - ||              DRS_INTERACTIVE is not set, running on-line mode

```

Now we define the custom arguments. First we define pos (position), fmt (types), names and call (calls) and then we use **GetCustomFromRuntime()** to create the custom arguments in the correct format.

```
In [13]: # deal with arguments being None (i.e. get from sys.argv)
pos = [0, 1]
fmt = [str, str]
names = ['flatfile', 'darkfile']
call = [darkfile, flatfile]
# now get custom arguments
customargs = spirouStartup.GetCustomFromRuntime(pos, fmt, names, calls=call)
```

This gets the arguments either from runtime (user input - here via sys.argv) or from the arguments defined in main() (hence darkfile and flatfile both set to None).

The result is "customargs" a dictionary containing all the correct parameters with correct formats.

```
In [14]: customargs
```

```
Out[14]: {'darkfile': 'dark_dark02d406.fits', 'flatfile': 'flat_flat02f10.fits'}
```

"customargs" is then used in **LoadArguments()** and a mainfitsfile **HAS** to be defined (this is responsible for setting which file is defined as "fitsfilename" and is subsequently used to load the calibration database file if needed. It can only be left blank if a recipe is not using "fitsfilename", "arg_file_names" or the calibration database.

```
In [15]: # get parameters from configuration files and run time arguments
p = spirouStartup.LoadArguments(p, night_name, customargs=customargs,
                                mainfitsfile='flatfile')
# as we have custom arguments need to load the calibration database
p = spirouStartup.LoadCalibDB(p)
```

```
15:48:30.0 - |cal_BADPIX_spirou|Now running : cal_BADPIX_spirou with:
15:48:30.0 - |cal_BADPIX_spirou|          -- flatfile=flat_flat02f10.fits
15:48:30.0 - |cal_BADPIX_spirou|          -- darkfile=dark_dark02d406.fits
15:48:30.0 - |cal_BADPIX_spirou|ICDP_NAME loaded from: /scratch/Projects/spirou_py3/spirou_py3
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_flat_flat02f10_badpixel.fits already
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_flat_dark02f10_blaze_AB.fits already
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_dark_flat02f10_blaze_C.fits already
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_dark_dark02d406.fits already exists
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_flat_dark02f10_flat_AB.fits already
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_dark_flat02f10_flat_C.fits already
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_flat_dark02f10_loco_AB.fits already
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_dark_flat02f10_loco_C.fits already
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_flat_dark02f10_order_profile_AB.fit
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_dark_flat02f10_order_profile_C.fits
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 20170710_fp_fp02a203_tilt.fits already exist
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: spirou_wave_ini3.fits already exists - not c
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 2017-10-11_21-32-17_hcone_hcone02c406_wave_A
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: spirou_wave_ini3.fits already exists - not c
15:48:30.0 - |cal_BADPIX_spirou|Calibration file: 2017-10-11_21-32-17_hcone_hcone02c406_wave_C
```

Packaged up in the main function the setup routines will look as follows:

```
In [16]: # =====
# Define functions
# =====
def main(night_name=None, darkfile=None, flatfile=None):
    # -----
    # Set up
    # -----
    # get parameters from config files/run time args/load paths + calibdb
    p = spirouStartup.Begin()
    # deal with arguments being None (i.e. get from sys.argv)
    pos = [0, 1]
    fmt = [str, str]
    names = ['flatfile', 'darkfile']
    call = [darkfile, flatfile]
    # now get custom arguments
    customargs = spirouStartup.GetCustomFromRuntime(pos, fmt, names, calls=call)
    # get parameters from configuration files and run time arguments
    p = spirouStartup.LoadArguments(p, night_name, customargs=customargs,
                                    mainfitsfile='flatfile')
    # as we have custom arguments need to load the calibration database
    p = spirouStartup.LoadCalibDB(p)

    # -----
    # Rest of function here
    # -----
```

1.2 Example with cal_CCF_E2DS_spirou

The recipe cal_CCF_E2DS_spirou.py requires 5 additional arguments (after night_name) and input is as follows:

```
cal_CCF_E2DS_spirou.py [night_directory] [E2DSfilename] [mask] [RV] [width] [step]
```

In the recipe this is packaged up as follows:

```
In [21]: def main(night_name=None, e2dsfile=None, mask=None, rv=None, width=None,
                step=None):
    # -----
    # Set up
    # -----
    # get parameters from config files/run time args/load paths + calibdb
    p = spirouStartup.Begin()

    # deal with arguments being None (i.e. get from sys.argv)
    pos = [0, 1, 2, 3, 4]
    fmt = [str, str, float, float, float]
    name = ['e2dsfile', 'ccf_mask', 'target_rv', 'ccf_width', 'ccf_step']
```

```

lname = ['input_file', 'CCF_mask', 'RV', 'CCF_width', 'CCF_step']
req = [True, True, True, False, False]
call = [e2dsfile, mask, rv, width, step]
call_priority = [True, True, True, True, True]
# now get custom arguments
customargs = spirouStartup.GetCustomFromRuntime(pos, fmt, name, req, call,
                                                call_priority, lname)

# get parameters from configuration files and run time arguments
p = spirouStartup.LoadArguments(p, night_name, customargs=customargs,
                                mainfitsfile='e2dsfile',
                                mainfitsdir='reduced')

# as we have custom arguments need to load the calibration database
p = spirouStartup.LoadCalibDB(p)

```

where this time CCF_width and CCF_step are not required (they have a default value) and an additional parameter was added to **LoadArguments()**.

This additional parameter is "mainfitsdir" and can either have the value "reduced" or "raw" "calibDB" or be a valid absolute path to the folder. This allows the "mainfitsfile" to be located in a folder other than the default ("raw") folder (this was the case above for cal_BADPIX_spirou.py).

The content of customargs is shown below and finally how they appear in the parameter dictionary "p" is shown at the end.

In [37]:

```

# these are the arguments of the main function (taken out to show below)
# they are all set to None by default - and hence will cause an exception
# unless the user enters these required values (all strings)
night_name=None
e2dsfile=None
mask=None
rv=None
width=None
step=None

# force the runtime arguments (notebook only)
import sys
sys.argv = ['cal_CCF_E2DS_spirou.py', '20170710', 'fp_fp02a203_e2ds_AB.fits', 'UrNe.ma

# -----
# Set up
# -----
# get parameters from config files/run time args/load paths + calibdb
p = spirouStartup.Begin()
# deal with arguments being None (i.e. get from sys.argv)
pos = [0, 1, 2, 3, 4]
fmt = [str, str, float, float, float]
name = ['e2dsfile', 'ccf_mask', 'target_rv', 'ccf_width', 'ccf_step']
lname = ['input_file', 'CCF_mask', 'RV', 'CCF_width', 'CCF_step']
req = [True, True, True, False, False]

```

```

call = [e2dsfile, mask, rv, width, step]
call_priority = [True, True, True, True, True]
# now get custom arguments
customargs = spirouStartup.GetCustomFromRuntime(pos, fmt, name, req, call,
                                                call_priority, lname)

# get parameters from configuration files and run time arguments
p = spirouStartup.LoadArguments(p, night_name, customargs=customargs,
                                mainfitsfile='e2dsfile',
                                mainfitsdir='reduced')

# as we have custom arguments need to load the calibration database
p = spirouStartup.LoadCalibDB(p)

16:00:12.0 - || *****
16:00:12.0 - || * SPIROU @(#) Geneva Observatory (0.1.030)
16:00:12.0 - || *****
16:00:12.0 - || (dir_data_raw)      DRS_DATA_RAW=/scratch/Projects/spirou_py3/data/raw
16:00:12.0 - || (dir_data_reduc)     DRS_DATA_REDUC=/scratch/Projects/spirou_py3/data/reduced
16:00:12.0 - || (dir_calib_db)       DRS_CALIB_DB=/scratch/Projects/spirou_py3/data/calibDB
16:00:12.0 - || (dir_data_msg)       DRS_DATA_MSG=/scratch/Projects/spirou_py3/data/msg
16:00:12.0 - || (print_level)        PRINT_LEVEL=all          %(error/warning/info/all)
16:00:12.0 - || (log_level)          LOG_LEVEL=all           %(error/warning/info/all)
16:00:12.0 - || (plot_graph)         DRS_PLOT=1             %(def/undef/trigger)
16:00:12.0 - || (used_date)          DRS_USED_DATE=undefined
16:00:12.0 - || (working_dir)        DRS_DATA_WORKING=/scratch/Projects/spirou_py3/data/tmp
16:00:12.0 - ||                  DRS_INTERACTIVE is not set, running on-line mode
16:00:12.0 - |cal_CCF_E2DS_spirou|Now running : cal_CCF_E2DS_spirou with:
16:00:12.0 - |cal_CCF_E2DS_spirou|      -- e2dsfile=fp_fp02a203_e2ds_AB.fits
16:00:12.0 - |cal_CCF_E2DS_spirou|      -- ccf_mask=UrNe.mas
16:00:12.0 - |cal_CCF_E2DS_spirou|      -- target_rv=0.0
16:00:12.0 - |cal_CCF_E2DS_spirou|      -- ccf_width=10.0
16:00:12.0 - |cal_CCF_E2DS_spirou|      -- ccf_step=0.1
16:00:12.0 - |cal_CCF_E2DS_spirou|ICDP_NAME loaded from: /scratch/Projects/spirou_py3/spirou_p
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_flat_flat02f10_badpixel.fits alre
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_flat_dark02f10_blaze_AB.fits alre
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_dark_flat02f10_blaze_C.fits alrea
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_dark_dark02d406.fits already exis
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_flat_dark02f10_flat_AB.fits alrea
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_dark_flat02f10_flat_C.fits alread
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_flat_dark02f10_loco_AB.fits alrea
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_dark_flat02f10_loco_C.fits alread
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_flat_dark02f10_order_profile_AB.f
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_dark_flat02f10_order_profile_C.fi
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 20170710_fp_fp02a203_tilt.fits already exi
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: spirou_wave_ini3.fits already exists - not
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 2017-10-11_21-32-17_hcone_hcone02c406_wave
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: spirou_wave_ini3.fits already exists - not
16:00:12.0 - |cal_CCF_E2DS_spirou|Calibration file: 2017-10-11_21-32-17_hcone_hcone02c406_wave

```

```

In [40]: # display customargs
         customargs

Out[40]: {'ccf_mask': 'UrNe.mas',
         'ccf_step': 0.1,
         'ccf_width': 10.0,
         'e2dsfile': 'fp_fp02a203_e2ds_AB.fits',
         'target_rv': 0.0}

In [39]: # display parameters in p
         keys = ['E2DSFILE', 'CCF_MASK', 'TARGET_RV', 'CCF_WIDTH', 'CCF_STEP']

         for key in keys:
             args = [key, p[key], type(p[key]), p.sources[key]]
             print('\n - key = "{0}"\n\tvalue={1}\n\ttype={2}\n\tsource={3}'.format(*args))

- key = "E2DSFILE"
  value=fp_fp02a203_e2ds_AB.fits
  type=<class 'str'>
  source=From run time arguments (sys.argv)

- key = "CCF_MASK"
  value=UrNe.mas
  type=<class 'str'>
  source=From run time arguments (sys.argv)

- key = "TARGET_RV"
  value=0.0
  type=<class 'float'>
  source=From run time arguments (sys.argv)

- key = "CCF_WIDTH"
  value=10.0
  type=<class 'float'>
  source=From run time arguments (sys.argv)

- key = "CCF_STEP"
  value=0.1
  type=<class 'float'>
  source=From run time arguments (sys.argv)

```