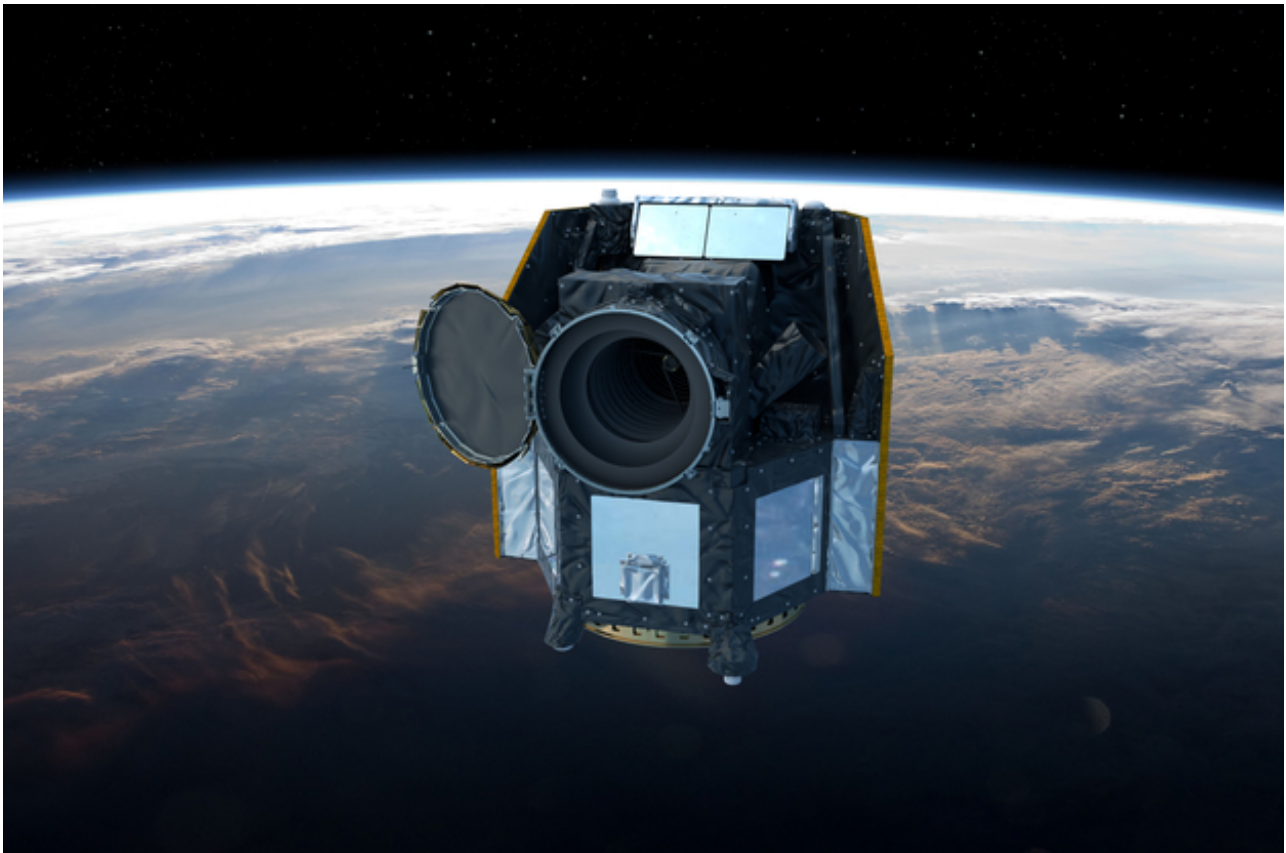

PYCHEOPS make_xml_files

Release notes, version 0.1.4, 13 March 2019

Dr Pierre Maxted, Keele University, UK (p.maxted@keele.ac.uk)



Introduction

This document describes the `make_xml_files` command that is part of the `pycheops` python package. The `pycheops` package is currently under development to provide a library of python functions, classes, constants, etc. for the analysis of data from the ESA CHEOPS satellite.¹

The main purpose of `make_xml_files` is to generate XML files for input into the CHEOPS Feasibility Checker (FC) based on a table of data for multiple observing requests provided by the user. In addition, `make_xml_files` can be used to perform some basic checks on the feasibility of an observation. The XML files produced by `make_xml_files` can also be imported into the CHEOPS Phase 2 Proposal Handling Tool (PHT2).

Use of `make_xml_files` is not mandatory. Template XML files have been provided as part of CHEOPS open time AO-1 and can be edited directly to specify individual observing requests using a text editor.

Caveats and warnings

- `pycheops` and `make_xml_files` are released under the GNU General Public License. They are provided “as-is” in the hope that they will be useful, but without any warranty.
- No guarantee is provided that the output of these programs is correct and no commitment is made to update, fix or maintain this software.
- The PI of the observing programme is responsible for checking that target Gaia DR2 identifier submitted to the Phase 2 Proposal Handling Tool is correct.
- Information provided by `make_xml_files` is based on the best available information about the instrument and spacecraft available prior to launch. Estimates of the optimum exposure time, aperture contamination and target visibility will all change once updated information is available from the in-orbit commissioning phase.
- The visibility estimates provided by `make_xml_files` are not a replacement for the detailed feasibility information provided by the CHEOPS Feasibility Checker (FC).
- Questions, bug reports and requests for new features can be sent to Dr Pierre Maxted (p.maxted@keele.ac.uk).

¹ <http://sci.esa.int/cheops/>

Installation

Installation of `pycheops` requires python version 3.6 or later. The anaconda distribution² of python 3.6 is recommended because it already includes many of the packages required for `pycheops` and it can be installed by any user without interfering with other python distributions on the system. The procedure below should automatically install any additional packages that are required for any python distribution, but this has only been tested for the anaconda python distribution.

To install `pycheops` use the `pip` command as follows (“\$” indicates a unix command line prompt)

```
$ pip install pycheops
```

If this step fails you may need to use `conda` to upgrade `scipy` first, i.e.,

```
$ conda update scipy
$ pip install pycheops
```

This will install the `pycheops` package, including any prerequisites. The `make_xml_files` command should now be available³ and give the following output if entered without any arguments.

```
$ make_xml_files
usage: make_xml_files [-h] [-p PROGRAMME_ID] [-r MATCH_RADIUS]
                    [-g GAIA_MAG_TOLERANCE] [--ignore-gaia-id-check] [-a]
                    [-e] [-f] [-x SUFFIX]
                    [--proprietary_last PROPRIETARY_LAST]
                    [--proprietary_first PROPRIETARY_FIRST]
                    [table]
```

² <https://www.anaconda.com/download/>

³ You may need to run the command “`rehash`” to make the command accessible in the first instance.

Usage

- Copy the example input table files and `ReadMe.txt` file to your current working directory, as follows.

```
$ make_xml_files -e
```

- Use the descriptions in the file `ReadMe.txt` to identify the example input file that most closely matches your requirements in terms of observing requests and whether or not you want `make_xml_files` to suggest a Gaia DR2 source identifier for your targets.
- Using a text editor, spreadsheet program or other table-handling tool (e.g. `topcat`⁴) create a CSV file that contains the observing request information for your targets in the same format as your example input table file. The columns in the input table file are described in the following section. This information can also be seen using the built-in command help, i.e.,

```
$ make_xml_files -h
```

- Generate the XML files for each observing request in your input table file, e.g.,

```
$ make_xml_files MyFirstCHEOPS0bservingRequests.csv
```

An example of the output from `make_xml_files` is shown at the end of this document. The columns in the output table are described below. This information is also available from the built-in command help, i.e.,

```
$ make_xml_files -h
```

Gaia_DR2_ID

Gaia DR2 ID from Gaia data archive. This must match the value of `Gaia_DR2` in the input file unless the flag `--ignore-gaia-id-check` is specified. **N.B.** *The PI is responsible to check the DR2 ID is correct.*

Gmag

The target mean G-band magnitude from Gaia DR2 catalogue.

Contam

Estimate of the contamination of a 30 arcsec photometric aperture by nearby stars relative to the target flux.

Vis

Estimate of the percentage of the orbit for which the target is observable by CHEOPS. This estimate is not a substitute for the detailed scheduling information provided by the CHEOPS Feasibility Checker.

Texp

the exposure time used in the output XML file.

_RAJ2000, _DEJ2000

ICRS position of matching Gaia source in degrees.

Flags

sum of the error/warnings flags listed below.

⁴ <http://www.star.bris.ac.uk/~mbt/topcat/>

List of Error/warning flags

- 32768 = Gaia ID error - input/output IDs do not match
- 16384 = Acquisition error, brighter star within 51"
- 8192 = Acquisition warning, brighter star within 51"-180"
- 4096 = Contamination error, Contam > 1
- 2048 = Contamination warning, Contam = 0.1 - 1
- 1024 = No spectral type match, assuming G-V = 0
- 512 = Visibility error, efficiency = 0
- 256 = Visibility warning, efficiency < 50%
- 128 = Exposure time error - target will be saturated
- 64 = Exposure time warning - below recommended minimum time
- 32 = Exposure time error - magnitude out of range, not set⁵
- 16 = Exposure time warning - magnitude out of range, not checked

⁵ Automatic exposure times only reliable for the magnitude range G=5.847 to G=12.847

Input table column descriptions

The following columns must be defined in the table.

ObsReqName	– unique observing request identifier
Target	– target name
_RAJ2000	– right ascension, ICRS epoch J2000.0, hh:mm:ss.ss
_DEJ2000	– declination, ICRS epoch J2000.0, +dd:mm:ss.s
SpTy	– spectral type (any string starting [AFGKM][0-9])
Vmag	– V-band magnitude
e_Vmag	– error in V-band magnitude
BJD_early	– earliest start date (BJD)
BJD_late	– latest start date (BJD)
T_visit	– visit duration in seconds
N_Visits	– number of requested visits
Priority	– 1, 2 or 3
MinEffDur	– minimum on-source time, percentage of T_visit (integer)

If the flag `--ignore-gaia-id-check` is not specified on the command line then the following column is also required.

Gaia_DR2	– Gaia DR2 identification number (integer)
----------	--

If the flag `--auto-expose (-a)` is not specified on the command line then the following column is also required.

T_exp	– exposure time (seconds)
-------	---------------------------

In addition, for time-critical observations the following columns must also be defined.

BJD_0	– reference time for 0 phase (e.g., mid-transit), BJD
Period	– period in days
Ph_early	– earliest allowable start phase for visit
Ph_late	– latest allowable start phase for visit

The following columns will also be used if available.

N_Ranges	– number of phase ranges with extra efficiency constraints
BegPh1	– start of phase range 1
EndPh1	– end of phase range 1
Effic1	– minimum observing efficiency (%), phase range 1 (integer)
BegPh2	– start of phase range 1
EndPh2	– end of phase range 1
Effic2	– minimum observing efficiency (%), phase range 2 (integer)

N.B. If you have 2 phase ranges with extra efficiency constraints but only require one of them to be satisfied then use `N_Ranges = -2`

Example output

```
$ make_xml_files NonTimeCriticalOnly.csv
# Output from: make_xml_files version 0.1.4
# Run started: 2019-03-13 14:49:44
# Input file: NonTimeCriticalOnly.csv
# Gaia match radius: 1.0 arcsec
# Gmag tolerance: 0.5 mag
# Output file suffix: .xml
#
#ObsReqName  Gaia_DR2_ID          Gmag  _RAJ2000  _DEJ2000  Contam  Vis  Texp  Flags
HD189733_1hr 1827242816201846144  7.41  300.1821  +22.7108  0.01   67  0.5   0
HD209458_1hr 1779546757669063552  7.51  330.7949  +18.8843  0.00   79  0.5   0
HD80606_2hr  1019003226022657920  8.82  140.6566  +50.6037  0.86   56  5.0  2048
HD80607_2hr  1019003329101872896  8.95  140.6656  +50.6039  1.11   56  5.0  20480
DON972_1hr   6684454247844778624 10.20  298.8595  -43.7381  0.03   61  5.0  8192
HD550_30m   4635430079405227520  8.99   2.3966  -78.8399  0.00   28  5.0  256
SA01730_15m 1146439200662722816  9.28  158.8508  +83.1023  0.00    0 10.0  512
18_Sco_1hr  4345775217221821312  5.30  243.9053  -8.3695  0.00   92  0.1  16
```