

The Swift Reduction Package - REM - Users' Manual

by [Stefano Covino](#), 24 Aug 2013, v. **1.4.0**.

Background

The Swift Reduction Package (hereafter [SRP](#)) is a packet of tools supposed to make everyday astronomer's life easier.

For any specific comment the main documentation for **SRP** is the reference source. Here we refer to a sub-package, **SRPAstro.REM**, devoted to the management of the [REM](#) telescope and data.

Installation

If you are just updating **SRPAstro.REM** you just need to download the package from the [PyPI](#) archive with:

```
sudo easy_install -U --script-dir=/usr/local/bin SRPAstro.REM
```

provided of course you are connected to the web, and that you want your executable files in “/usr/local/bin”.

If you, instead, are installing **SRPAstro.REM** for the first time or maybe you are upgrading to a new **Python** release, it is likely you need to install many different libraries **SRP** and the related sub-packges relies on.

In principle the command:

```
sudo easy_install --script-dir=/usr/local/bin SRPAstro.REM
```

should again do the job. You might also consider to install the package in a [virtual python environment](#) if you do not want to interfere with the system python installation.

However, some of the required libraries can (will) require more concerned actions to allow their installation. In essentially all cases, browsing the web you can quickly find the solution to any problem.

An alternative and strongly advised procedure is to make a smart use of the various package managers available on many platforms ([macports](#), yum, apt-get, etc.). A possible sequence of operations on [Mac OSX](#) is the following:

- i) Install SRPAstro (i.e. check the SRP users' manual)
- ii) `sudo port install py27-mysql`
- iii) `sudo easy_install --script-dir=/usr/local/bin -U SRPAstro.REM`

while, on other Linux platforms, using yum or apt-get, an analogous sequence should work.

For instance, on a linux-PC running [Fedora](#):

- i) Install SRPAstro (i.e. check the SRP users' manual)
- ii) `sudo yum install MySQL-python`
- iii) `sudo easy_install --script-dir=/usr/local/bin -U SRPAstro.REM`

Step by step “how to”

These are just examples of what you can do with **SRPAstro.REM**.

REM data analysis

1. Frame astrometry
 - REM frames are produced already with an astrometric solution. In case the on-line system did not work properly, it is possible to derive the astrometric solution with **SRPAstrometry -i input.fits -o input_astro.fits**.
 - You can also determine the offsets between the pointed direction and the actual one with **SRPREMOffsets -f inpput.fits**.
2. Frame photometry
 - You can perform aperture photometry for a selected list of objects with a tool developed as an exercise: **SRPMyPhotometry -i obj_list.txt -f FITS_file**. The input file has a simple format: Id X Y for objects to be measured. And Id X Y Mag eMag for objects to be used as reference. These last objects are analyzed and the magnitudes are used to calibrate the other listed sources. The output is:
 - Id X Y MaxFlux Mag eMag CalibratedMag eCalibratedMag Comment MJD Exptime
 - In case no reference stars are supplied you can simply provide a zero point. You can also obtain a list of sources to be used with **SRPMyPhotometry** in a field with **SRPSourceFinder -f frame.fits -e**.
 - Using **SRPImageMapping** and **SRPMyPhotometry** one can perform photometry of the same set of stars in multiple frames independently of dithering and rotation. This tool applies the photometry by **SRPMyPhotometry** to the objects reported in the photometry file. The coordinates of the objects are updated according to the roto-translation parameters output of **SRPImageMapping** and calibrated magnitudes are then reported based on the reference stars possibly added to the configuration file. Finally, files with all the magnitudes, 1σ errors, observation time and exposure length are created for each object photometrized. The syntax is the same as for **SRPMyPhotometry** but the input FITS file is substituted by the output of **SRPImageMapping**: **SRPREMPhotometry -f file_map.txt -i obj_list.txt**. This tool has been developed for a quick reduction of massive [REM](#) telescope datasets.
3. REM archive query
 - REM archive can be browsed by **SRPREMDataBase --remir -s “date_obs > ‘2012-01-15’”**.

REM data maintainance

1. Pointing model computation
 - A pointing model for the REM telescope (or actually for any other althazimuthal telescope) can be derived with **SRPPointingModel -d data.dat -o pointmod.pm**.

Miscellanea

1. **SRP.REM** running version
 - You can know the present **SRP.REM** version with **SRPREMVersion**

List of commands

1. **SRPMyPhotometry**

- Its purpose is to perform aperture photometry for selected source in a frame.
- **SRPMyPhotometry [-a arg1 arg2] [-e arg3] -f arg4 [-g arg5] [-h] [-H arg6, arg7] -i arg8 [-n arg9 [-r arg10 arg11 arg12] [-s arg13] [-t] [-S] [-v] [-z arg14 arg15]**
 - a Observation airmass and coefficient
 - f Input FITS file
 - g Gain (e-/ADU) for error estimate in photometry
 - i Input file
 - s Saturation level (ADU) for frame(s)
 - e Exposure time (sec) for frame(s)
 - n Readout noise (e-)
 - S ESO-Skycat output
 - r Radius (pixel) for aperture photometry (r is os)
 - t Do not fit centroid position
 - z Zero point and error for photometry
 - H FITS file header for exposure time and duration
[default: MJD-OBS, EXPTIME]

2. **SRPREMAstrometry**

- **SRPREMAstrometry [-h] -i file -o file [-v] [--version]**
 - i Input FITS file.
 - o Output FITS file.

3. **SRPREMDataBase**

- Its purpose is to browse the REM observation archive.
- **SRPREMDataBase [-a arg1] -n arg2/--remir/--ross [-o arg3] -s arg5 [-t arg4] [-v]**
 - a Database internet address
 - n Database name/--remir REMIR database/--ross ROSS database

- o Output file
- s SQL query string
- t Database table name

4. SRPREMOffsets

- Its purpose is to compute offsets between the pointed and actual coordinates for a REM frame.
- SRPREMOffsets [-h] [-c x y] -f file [-v] [--version]
 - c x y, Coordinate center
 - f file, REM FITS file or list of files

5. SRPREMPhotometry

- Its purpose is to perform aperture photometry for selected source in many frames automatically.
- SRPREMPhotometry [-e arg1] [-g arg2] [-h] [-H arg4 arg5] -i arg6 [-n arg7] [-r arg8 arg9 arg10] [-s arg11] [-S] [-t] [-v] [-w] [-z arg9 arg10]
 - f Input FITS file list
 - i Input photometry file
 - g Gain (e-/ADU) for error estimate in photometry
 - H FITS file header for exposure time and duration
[default: MJD-OBS, EXPTIME]
 - n Readout noise (e-)
 - r Radius (pixel) for aperture photometry
 - s Saturation level (ADU) for frame(s)
 - t Do not fit centroid position
 - w Force re-write of object position files
 - z Zero point and error for photometry

6. SRPREMPointingModel

- Its purpose is to compute the pointing model for the REM (or any other) telescope. Parameters are as defined in the “pillar” software.
- SRPREMPointingModel [-h] -d file [-D pos1 pos2 pos3 pos4] [-g] [-G par ...] [-m model] -o file [-p file] [-v] [--version]
 - d file, --data file Data input file (degrees)
 - D pos1 pos2 pos3 pos4, --datapos pos1 pos2 pos3 pos4
Data input file column positions: AZ AZOff ALT ALTOff
(e.g. 1 2 3 4)
 - g show plot
 - G PM parameter guess for fitting
 - m model, --model model
Model flavour
 - o file, --outfile file
Output file with new model parameters (pillar format)
 - n file, --nofile file
File with input data with no PM
 - p file, --presfile file

File with present model parameters

7. **SRPREMVersion**

- its purpose is to show the running **SRP.REM** version.
- SRPVersion [-h] [-v]

Bugs, comments, etc.

Of course, as already stated, any contribution from anyone is welcome. In case you find bugs, have improvements to suggest, would like to contribute to the code, etc. Please send an e-mail to Stefano Covino, stefano.covino@brera.inaf.it. We can not promise to take into account all your comments, but we will anyway try to improve the package to meet your needs.

Evolution

- **From 1.0.0 to 1.1.0:**
 - Command to browse the REM archive, **SRPREMDataBase**, was added.
- **From 1.1.0 to 1.2.0:**
 - Command to derive frame astrometry, **SRPREMAstrometry**, was added.

From 1.2.0 to 1.4.0:

- Command to compute a telescope pointing model, **SRPREMPointingModel**, was added. Minor bugs corrected. Data for the ROs2 camera Better minimization in **SRPREMPointingModel**. **SRPREMOffsets** added. Many improvements to **SRPREMAstrometry** and **SRPREMPointingModel**.