Req 5.4.4

Title:

Night Sky flat

Objective:

Create Night Sky flat frame.

The flatfield that most closely reproduces the actual gain variation of the science and standard observation, can be obtained by averaging a large number of *flat-fielded* science and standard observations, taking care of properly masking the contaminating objects. While such a night-sky flat (aka superflat) could, in principle, improve on the quality of the twilight flat, the procedure to obtain this flat can be computationally very expensive. On the other hand night-sky flats may also be suitable for fringe removal.

It remains to be seen to what extent the problem of sky concentration, i.e. nonuniform illumination due to stray light/reflection affects the quality of the night-sky flat.

Because these night-sky flats can only be obtained from actual observations, we cannot guarantee their availability for bands for which only standards were obtained. It is, therefore, not clear how routine building of the skyframes should be incorporated into our photometric calibration scheme.

A minimum of 5 images in a night in a given band is required to optimally fulfill this requirement. Images are stacked and a median average is calculated, intended to remove any non-systematic effects (objects, cosmic rays, satellite tracks, etc.).

Check reproducibility of artifacts on flat field images in order to inspect filter exchange mechanism reproducibility and flexure of telescope.

Fulfilling or fulfilled by:

Data reduction of raw data from science and photometric standard observations
The night sky flat is used in **req.**546 Master flat.

When performed/frequency:

daytime, daily (flexure, reproducibility during Comm)

Sources, observations, instrument configurations:

Science and standard observations

Inputs:

Raw science images

CalFile– 541 *Master Bias frame*
CalFile– 546 *Master flatfield*
CalFile– 522 *Hot pixel map*
CalFile– 535 *Cold pixel map*

Outputs:

CalFile– 544 *Nightsky flat frame*

Required accuracy, constraints:

This procedure would benefit from a prior detection and masking of bright objects. If no masks of bright objects are available, then a minimum of 15 frames should be included.

Estimated time needed:

Observation: None. Reduction: 5 min./CCD/filter for 15 science frames.

Priority:

very desirable

TSF:

Raw data of science and standard star observations in jitter and dither mode is used. Avoid using multiple exposures with the same pointing in Stare mode.

This procedure benefits from first flatfielding the data, and determining the night-sky flat as a modification of this flatfield. First de-bias the science/standard frames and apply the flats, then normalize the frames and build a median average frame.

Recipe:

```
Nightsky_Flat -i <raw_science_frames> -b bias -f flat [-c cold]
[-h hot]
              [-oc OVERSCAN_CORRECTION]

raw_science_frames  : the raw science images
bias                : the master bias image
flat                : the master flat image
cold                : the cold pixel map
hot                 : the hot pixel map
OVERSCAN_CORRECTION : overscan correction mode (integer).
                      Description of allowed values:
                        0: apply no overscan correction (default)
                        1: use median of the prescan in the
```

```
                           x-direction
                        2: use median of the overscan in the
                           x-direction
                        3: use median of the prescan in the
                           y-direction
                        4: use median of the overscan in the
                           y-direction
                        5: use the per-row value of the prescan
in
                           the x-direction
                        6: use the per-row value of the overscan
in
                           the x-direction
```

Before applying this recipe, use **Recipe– Split**—which is documented in **seq.–631**—with the `-t science` option to split the raw multi-extension FITS input files.

Needed functionality:

image - processing (eclipse.trim_and_overscan)
image - arithmetic (eclipse.image_sub_local, eclipse.image_div_local)
image - statistics (eclipse.stat, eclipse.iter_stat)
image - stacking (eclipse.cube)
cube - median average (eclipse.cube_avg_median)
pixelmap - binary AND (eclipse.pixelmap_binary_AND)

CA:

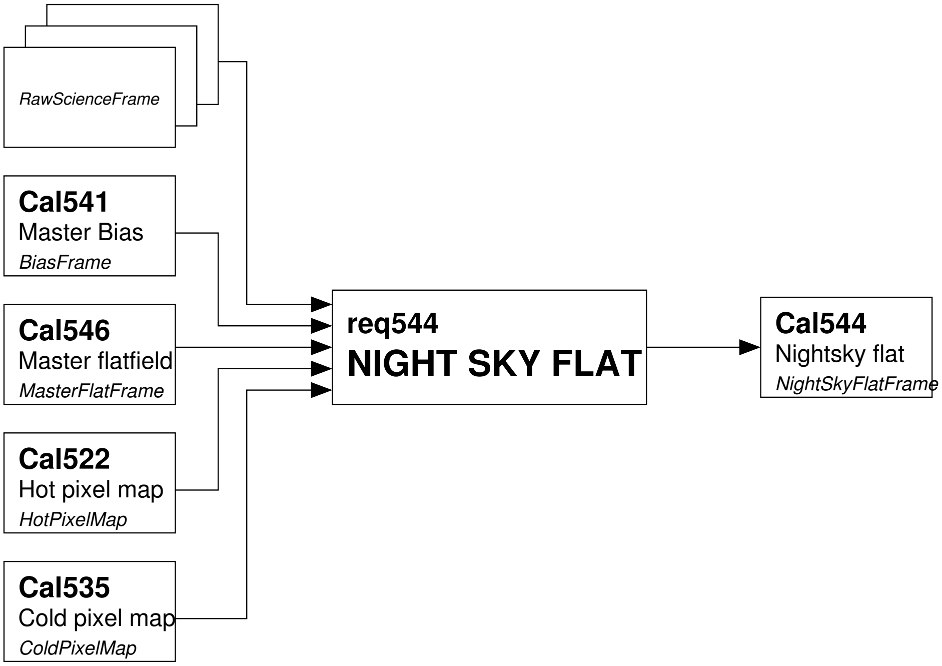

Fig 5.4.4 Dataflow and object class names (in small italic font) for req544

Process (make):

1. Check that at least three input science images are given; this is necessary to calculate a median average of a cube of these.
2. Construct a mask from input hot and cold pixel maps.
3. Trim, overscan correct, de-bias and flat-field the input science images.
4. Iteratively estimate the statistics of the resulting science images.
5. Normalize each image by dividing by its median pixel value as determined above.
6. Stack these science images in a cube.
7. Determine the median average of the cube (this is an image).
8. Normalize the night sky flat obtained in 7. (use mask obtained in 2.).

Verification (verify):
TBD

CAP:

```
mask = eclipse.pixelmap_binary_AND(hot, cold)

cube = []
```

```
for frame in raw_science_frames:
    frame = eclipse.trim_and_overscan(frame)
    frame = eclipse.image_sub(frame, bias)
    frame = eclipse.image_div(frame, flat)
    stats = eclipse.iter_stat(frame)
    frame = eclipse.image_div(frame, stats.median)
    cube.append(frame)

nightsky = eclipse.cube_avg_median(nightsky)
stats = eclipse.stat(nightsky, pixmap=mask)
nightsky = eclipse.image_div(nightsky, stats.avg_pix)
```