ORAC-DR – LCOGT pipeline data reduction
0.4
User Guide

Abstract

ORAC-DR is a general-purpose automatic data-reduction pipeline environment. This document describes its use to reduce optical imaging data collected on the Las Cumbres Observatory Global Telescope Network (LCOGT).

Contents

1   Introduction

ORAC-DR is a data-reduction pipeline operating at UKIRT, JCMT, the AAT and forms the basis of the operational pipeline at LCOGT. The pipeline reduces and displays multi-frame observations soon after they are read from the detector. This allows observers to assess the quality and suitability of their data in near real time but ORAC-DR is also used to produce the final archive-delivered products using the same code base.

SUN/230 presents an overview of ORAC-DR, general facilities like its display system, and it explains the differences between a pipeline and a traditional reduction package. Put briefly, ORAC-DR uses a few data headers to direct the data reduction. Amongst these headers is the name of a *recipe*. A recipe is a series of high-level instructions such as "make a mosaic" or "divide by a flat" that reduces an *observation* comprising one or more data frames. The implementation of each of these instructions is through a Perl script—called a primitive—which calls Starlink packages such as CCDPACK and KAPPA, to actually do the processing of the bulk data.

This document describes how to use ORAC-DR software on Starlink to reduce optical imaging data from the Las Cumbres Observatory Global Telescope Network (hereafter LCOGT; [1]) consists of data taken from one of the LCOGT instruments; LCOSBIG, LCOCC, LCOSINISTRO, LCOSPECTRAL or LCOFLI.

1.1   Document conventions

In an attempt to make this document clearer to read, different fonts are used for specific structures.

Observing modes are denoted by all upper case body text (e.g. FLATFIELD).

Starlink package names are shown in small caps (e.g. SMURF); individual task names are shown in sans-serif (e.g. makebias). ORAC-DR recipes and primitives are also shown in sans-serif and are always upper case (e.g. REDUCE_BIAS).

Content listings are shown in fixed-width type (sometimes called 'typewriter'). Extensions and components within NDF (SUN/33) data files are shown in upper case fixed-width type (e.g. `HISTORY`).

Text relating to filenames (including suffices for data products), key presses or entries typed at the command line are also denoted by fixed-width type (e.g. `% kappa`), as are parameters for tasks which are displayed in upper case (e.g. `METHOD`).

References to Starlink documents, i.e., Starlink User Notes (SUN), Starlink General documents (SG) and Starlink Cookbooks (SC), are given in the text using the document type and the corresponding number (e.g. SUN/95). Non-Starlink documents are cited in the text and listed in the bibliography.

File name suffices represent the text between the final underscore character and the three-letter `.sdf` extension. For example, a file named `lsc1m005-fl07-20140717-0173-d02_bp.sdf` has the suffix `_bp`.

2 LCOGT Pipeline Variants

There are two variants of the LCOGT pipeline, one of which is designed to run in real time as data comes in. Most users will only need to run the second form, known as the offline pipeline.

- The quick-look (QL) pipeline is primarily designed to perform quality-assurance analysis of the incoming data for real-time assessment of the instrument performance.

- The offline pipeline has access to all the data observed for a given night/camera and adopts a best-possible reduction approach. Images are made for each complete observation which are combined to create the final image.

2.1 Requirements for running the LCOGT pipeline

The LCOGT pipeline requires a recent Starlink installation. The latest version may be obtained from `http://starlink.eao.hawaii.edu/starlink`. Since development of the pipeline is an ongoing process, it is recommended that the newest builds be used to access the full capabilities of the pipeline. These builds can be obtained from `https://github.com/Starlink/ORAC-DR/` and may be kept up-to-date with git.

The Starlink Perl installation (Starperl) must be used to run the pipeline due to the module requirements. The Starlink environment should be initialized as usual before running ORAC-DR.

The pipeline uses the following Starlink applications:

- KAPPA

- CCDPACK

- SEXTRACTOR

2.2 Important environment variables

The pipeline uses a number of environment variables to determine where data should be read from and written to. Some are set automatically when the pipeline is initialized, but they can be overridden manually and, with the `-honour` flag may be left unchanged between successive runs of the pipeline. The variables that must be defined in order for the pipeline to run are denoted as 'Mandatory' in the list below.

- `STARLINK_DIR`: location of the user's Starlink installation. [Mandatory]

- `ORAC_DATA_IN`: the location where data are read from. If running with `-loop flag`, this is the location of the flag files, rather than the data files. [Mandatory]

- `ORAC_DATA_OUT`: location where intermediate pipeline data products are written. [Mandatory]

- `FINAL_DATA_OUT`: final location for pipeline data products. At SBA, this is `/mfs-sba/engineering/<site>/<camera>`. If not defined, the current directory is assumed. [Optional]

- `ORAC_DATA_CAL`: a user-specified location for the library calibration files. [Mandatory]

- `ORAC_CAL_ROOT`: a user-specified location for pipeline configuration files. [Mandatory]

- `ORAC_INSTRUMENT`: the type of LCOGT instrument class in use e.g. LCOSINISTRO. [Mandatory]

- `ORAC_CAMERACODE`: the type of LCOGT instrument instance in use e.g. fl07. [Mandatory]

- `ORAC_SITECODE`: the LCOGT site code that the data is from e.g. LSC. [Mandatory]

3   Running ORAC-DR

This is a very brief introduction to running ORAC-DR. More detailed information can be found in SUN/230. SUN/232 also includes a description of how to set up and run ORAC-DR.

These instructions assume that you are using the `tcsh` shell as this is what the pipeline wrappers have been written in and have the Starlink software already setup (this is normally done in the users' setup files e.g. in `/.tcshrc`) but if this has not be done, you can get things setup by issuing the following commands (this assumes the Starlink software has been installed in `/git-star`):

```
$ tcsh
% setenv STARLINK_DIR /git-star
% source $STARLINK_DIR/etc/cshrc
% source $STARLINK_DIR/etc/login
```

You must first initialise ORAC-DR using `source ∼eng/Setup_Night_ORAC.csh <UTdate> <camera>`. This will prepare ORAC-DR to reduce data taken that night. If necessary, you should set the `$ORAC_DATA_IN` and `$ORAC_DATA_OUT` environment variables to the names of the directories from which the raw data should be read and to which reduced data should be written.

For example:

```
% source ~eng/Setup_Night_ORAC.csh 20140511 kb74
```

This will setup the necessary environment variables (as detailed in Section 2.2) to enable the pipeline to run properly. It is also possible to redefine the enviroment variables to read and/or write from different locations or to use a different calibration library for example. (To see a list of all the enviroment variables that were setup, you could issue the following command:

```
% printenv | grep "ORAC\|FINAL"
FINAL_DATA_OUT=/mnt/images/daydirs/kb74/20140511/oracproc
ORAC_CAL_ROOT=/data/tlister/SVN/LCO_Pipeline/trunk/code/oracdr/cal
ORAC_CAMERACODE=kb74
ORAC_DATA_CAL=/data/tlister/SVN/LCO_Pipeline/trunk/code/oracdr/cal/lcosbig/kb74
ORAC_DATA_IN=/mfs-sba/engineering/elp/kb74/20140511/raw
ORAC_DATA_OUT=/speedy/eng/elp/proc/kb74/20140511
ORAC_DIR=/data/tlister/SVN/LCO_Pipeline/trunk/code/oracdr/src/
```

```
ORAC_INSTRUMENT=LCOSBIG
ORAC_LOOP=flag -skip
ORAC_NFS_OK=1
ORAC_PERL5LIB=/data/tlister/SVN/LCO_Pipeline/trunk/code/oracdr/src/lib/perl5/
ORAC_PERSON=tlister@lcogt.net
ORAC_SITECODE=elp
```

To reduce all data for a given night you should run

```
oracdr -log f -verbose -file <filename of frames>
```

This will not normally produce much (if any) output to the screen as the pipeline is designed to be run non-interactively. The pipeline will reduce the data in the order they are specified in the file given after the `-file` option, using the recipe name given in the image header. Details of the processing will be written to a logfile called `.oracdr_NNNN.log` where `NNNN` is the current process ID. It is written to `$ORAC_DATA_OUT` and is a hidden file (use e.g. `% ls -ltrA .oracdr*` to see them with the most recent files at the bottom).

Additional information about the processing can be seen by starting `oracdr` with the `-log fx` option which will open a X window for the log which has the advantage that warnings and errors are written to different, independently scrollable windows and the reduction can be paused and restarted.

You may choose to reduce your data with a recipe other than the one specified in the file headers. If you wished to test data reduction with a new SCAMP-based astrometric solver you had written, you could specify the `OFFLINE_REDUCTION_SCAMP` recipe on the command line, for example:

```
oracdr -log fx -verbose -debug -file OFFLINE_REDUCTION_SCAMP
```

(The `-debug` option will cause additional debug information such as each primitive call and timing information to be written into `.oracdr_NNNN.log` file and additionally into a `ORACDR.DEBUG` file in `$ORAC_DATA_OUT`).

To exit (or abort) ORAC-DR click on 'Exit' in the text log window, or type `[ctrl]-c` in the xterm. The command `oracdr_nuke` can be used to kill all DR-related processes, should you be having serious problems.

4   An overview of the reduction

The data reduction for LCOGT depends on the type of data being reduced. Calibration observations (bias, dark, flat) are reduced differently from science observations.

4.1   Bias observations

Bias observations are used to remove the instrumental DC offset applied to the CCD detector to ensure positive values at the ADC.

Masterbiases are created if there are enough ($>$ 2) raw bias frames in a contiguous group (frames having the same 'molecule number', MOLNUM) when the final frame of the group is reached (i.e. when MOLFRNUM == FRMTOTAL). The masterbias is created using the REDUCE_BIAS recipe which in turn calls the _MAKE_BIAS_FROM_GROUP_ primitive. This uses CCDPACK's makebias command using the method=median combination method without zero offsetting (zero=false) to combine the raw bias frames.

4.2 Dark observations

Dark observations are used to remove the thermal noise from the detector.

Masterdarks are created if there are enough ($>$ 2) raw dark frames in a contiguous group (frames having the same 'molecule number', MOLNUM) when the final frame of the group is reached (i.e. when MOLFRNUM == FRMTOTAL). The masterdark is created using the REDUCE_DARK recipe which in turn calls the _MAKE_DARK_FROM_GROUP_ primitive. This uses CCDPACK's makecal command using the method=median combination method with the expose= parameter set to a (created temporary) file of the dark exposure time in order to scale the darks before combining the raw dark frames.

4.3 Flatfield observations

Flat observations are used to remove the illumination variation across the field of view and to correct for pixel to pixel sensitivity variations.

Masterflats are created if there are enough ($>$ 3) raw flat frames in a contiguous group (frames having the same filter and 'molecule number', MOLNUM) when the final frame of the group is reached (i.e. when MOLFRNUM == FRMTOTAL). The masterflat is created using the SKY_FLAT recipe which in turn calls the _MAKE_FLAT_FROM_GROUP_ primitive. This uses CCDPACK's makeflat command to combine the raw flat frames. The combination method and whether cleaning of bad pixels is desired are passed in from the SKY_FLAT recipe in the call to the _MAKE_FLAT_FROM_GROUP_ primitive and is used to set the method= combination method and the clean= parameters, with the normal default being method=broadened and clean=f (as it was found that cleaning introduced clumps of bad pixels, particularly in the sharp edges between Sinistro quadrants). The resulting master flat is normalised to unity.

As masterflats are constructed, the _FLAT_QC_ primitive will check to see if it has found a matching pair of flats (masterflats having the same filter and binning factors, taken from the same night (which is checked via DAYOBS in the header)). Once a suitable pair is found, a difference frame is formed (using KAPPA's sub) and the sigma is computed. If the sigma is within tolerance, the two flats are filed with calibration system for future use (including making copies into the master calibration library location pointed to by $ORAC_DATA_CAL), otherwise the flats are removed from the calibration system flat index.

4.4 Imaging science observations

This section describes the processing that happens for the science frames i.e. the vast majority of the frames. Note that currently 'standard' (-s00) frames are treated the same as science (-e00) frames. The standard frames could (and there is some support within other ORAC-DR instrument recipes) be treated differently if desired. This processing is controlled by the OFFLINE_REDUCTION recipe.

4.4.1 Bias subtraction

Bias subtraction and bad pixel masking are actually handled by the generic _PREPARE_SINGLE_FRAME_ primitive which is called from within the recipe-specific _OFFLINE_REDUCTION_HELLO_ primitive. The appropriate bias and bad pixel mask to use are selected based on the normal nearest-in-time suitable calibration frame that matches the frame's binning.

4.4.2 Dark scaling and subtraction

Dark scaling (the master dark has previously been scaled to an exposure time of unity) to the science frames' exposure time and subtraction is performed by the _SUBTRACT_DARK_NO_THRESH_ primitive. (The normal _SUBTRACT_DARK_ primitve's additional thresholding behavior is not appropriate for the LCOGT optical imaging data which is much less suspectible to extra hot pixels). The appropriate dark frame to use is selected based on the normal nearest-in-time suitable calibration frame that matches the frame's binning.

4.4.3 Flatfield normalization

Flatfield division is performed by the _DIVIDE_BY_FLAT_ primitive and makes use of the CCDPACK flatcor task to perform the actual correction and trimming to the trim limits. The appropriate flatfield frame to use is selected based on the normal nearest-in-time suitable calibration frame that matches the frame's binning and filter.

4.4.4 Astrometric solution

The current astrometric solver makes use of the AUTOASTROM wrapper around the ASTROM task and is controlled from within the _ADD_AUTO_ASTROMETRY_ primitive. This handles the detection and extraction of objects on the image using SEXTRACTOR and the download of a reference catalog to match against. The reference catalogs are consulted in the order listed in the $ORAC_DATA_CAL/index.skycat_catalogue and the default is UCAC-3 [7] with a fallback of 2MASS [8] (both with two possible server sources at CDS, France and CADC, Canada to add redundancy) and a final fallback of the USNO-B [9] catalog (The Context Cameras (LCOCC) have a different version of $ORAC_DATA_CAL/index.skycat_catalogue which only makes use of the Tycho-2 catalog [10] due to the larger field of view and shallower depth of these cameras).

The ASTROM task can fit 4, 6, 7, 8, or 9 coefficients. The 4 coefficient is never used (assumes same scale in both axes & no shear terms), the 6 coefficient is the normal default for most cameras, the 7th term adds fitting for the radial distortion (barrel/pincushion), the 8 coefficient fit adds fitting for the plate center (but not distortion) and finally the 9 coefficient fit adds fit includes fitting for the plate center along with the distortion and is the default for the Context Cameras (LCOCC) due to their large field of view.

After a putative fit, the results are sanity checked on the X and Y rms, the number of stars used in the fit (must be ≥ 6) and the fitted platescale is within 5% of the expected value (which can cause false rejection if the nominal platescale (PIXSCALE in the FITS header) placed by the Instrument Agent hasn't been properly configured). The cases where the astrometric solution fails or is reset after the sanity check are distinguished by setting WCSERR in the header to '3' or '4' respectively (with '0' indicating success).

4.4.5   Calculation of image statistics

This occurs within the _CALCULATE_SEEING_STATS_ primitive and makes use of SEXTRAC-
TOR to perform the extraction and measurement of sources. A relatively high threshold of 5.0
sigma above the background mesh fit with a minimum source size of 12 pixels is used to avoid
pushing down too far into the noise and producing a biased result. Objects that have zero
size or SEXTRACTOR FLAGS!=0 are removed from the calculation and then the mean FWHM,
ellipticity and orientation angle are calculated, reported in the log and filed with the calibration
system, appearing in `index.dqc`. These are subsequently added to the header of the products as
described in Section 4.4.8.

4.4.6   Production of image catalog

This occurs within the _CREATE_IMAGE_CATALOGUE_ primitive and makes use of SEXTRAC-
TOR to perform the source detection and extraction and produce a LCOGT standard source
catalog in the format as defined in the LCOGT Pipeline/Science Archive Interface Control
Document (LCOGT-SA-ICD Version 0.10.0). This extraction occurs at a lower threshold of 1.5
sigma above the background mesh fit to give a reasonably complete source catalog without
overwhelming users or the Science Archive with spurious detections.

The SEXTRACTOR config file is in `$ORAC_DATA_CAL/extractor_catalogue.sex` but some ele-
ments are dynamically modified within the primitive as the LCOGT-SA-ICD Version 0.10.0
specification calls for aperture photometry in fixed aperture of $1''$, $3''$, $5''$ and $7''$ apertures which
translates to a variable camera-specific number of pixels, based on the platescale. The output
catalog is renamed to `<root name>10_cat.fits` or `<root name>90_cat.fits` depending on
whether QuickLook (indicated by `$Frm->uhdr( "QUICK_LOOK" ) == 1`) or offline processing is
occurring. The output catalog is subsequently modified by the `timecorrect.py` code as further
detailed in Section 4.4.9.

4.4.7   Calculation of zeropoint

This makes use of the catalog of detected sources produced by the previous _CREATE_IMAGE_CATALOGUE_
primitive and performs a cross-match to the combined standard star catalog produced by AJP's
findassm program. The cross match filters on distance, area (object size), ellipticity and quality
flags and surviving good matches are used to calculate the mean axial ratio of bright sources,the
sky brightness, the limiting magnitude, a transparency estimate and the instrumental zero
point. These results are reported in the log and filed with the calibration system, appearing in
`index.zeropoint`. These are subsequently added to the header of the products as described in
Section 4.4.8.

4.4.8   Output of derived products

The _DERIVED_PRODUCTS_ primitive is used to create the derived products from the pro-
cessed frames. The following items are performed:

- the headers are updated with the details of the master calibration frames and correction
  steps used,

- the saturation and trimmed section are updated with the values used in the pipeline
  processing,

- A clipped mean, median and sigma are calculated (using _CLIPPED_STATS_MEDIAN_) and added to the header,

- the QC module (the _SET_QC_FLAGS_ primitive) is run to produce the four QC flags in the headers,

- incorrect values of the TAGID and/or PROPID are flagged and corrected and the PROPID is checked against the lists of known proposals,

- access rights and state of the data are set,

- a 512×512 PNG thumbnail of the processed image is created,

- the frame is converted to FITS format,

- the `timecorrect.py` code (see Section 4.4.9) is called to compute per-star barycentric time correction and airmass and updates the header of the BCD image and source catalog,

- if a final destination is defined by the enviroment variable `$FINAL_DATA_OUT`, the derived products are moved there.

4.4.9   Airmass and barycentric time correction

In order to provide the most precise absolute timestamp that is free from ambiguities and limited by the properties of the target system, while allowing for improved corrections in the future, we quote the site arrival time (and timesystem) along with the correction to the Barycentric Julian Date in the Barycentric Dynamical Time standard (BJD$_{\mathrm{TDB}}$; see [2] for further details). Although Barycentric Co-ordinate Time, TCB, is formally recommended in the IAU framework (see [3]), it differs from the formerly and widely used time systems such as TDB and TT by $\sim 0.5\,\mathrm{s\,yr^{-1}}$ with a current difference of $\sim 17\,\mathrm{s}$ which is likely to lead to confusion for little benefit.

The per-star airmass and barycentric time correction is performed by timecorrect.py which uses a Python-wrapped copy of the SLALIB library to perform the time and reference system transformations. This code calculates a `TCORR` column which is added to the source catalogs, before being ingested into the archive.

This `TCORR` is the per-object complete "time correction" to go from UTC start at the observatory to TDB (Barycentric Dynamical Time) at the midpoint of the exposure at the Solar System Barycenter (SSB). This `TCORR` should be added to the UTC start time to get a proper BJD$_{\mathrm{TDB}}$. `TCORR` is a collection of terms:

$$\mathrm{TCORR} = t_{\mathrm{mid}} + t_{\mathrm{clock}} + t_{\mathrm{romer}} + t_{\mathrm{einstein}} \tag{1}$$

where t_{mid} is half the exposure time (to get to the midpoint), t_{clock} contains the UTC $\rightarrow$ TDB correction (which is time-dependent at the few ms level as it involves knowing the Earth's orientation and the position of the observatory within the Solar System potential), t_{romer} is the Røemer delay (light travel time) from the observatory to the SSB and $t_{einstein}$ is the Einstein delay due to the fact the observatory is moving (with the Earth) and displaced from the geocenter.

The aim of the code was that code should not contribute more than 1% error on the shortest exposure times likely on the LCOGT Network, which was taken to be 30 Hz ($= 33.3\,\mathrm{ms}$). The errors due to missing terms or limitations withing the time correction code are believed to be at the few $\times \sim 10\mu s$ level due to:

- Missing from the above equation is the Shapiro delay (due to the light-bending by the Sun's gravitational potential). This effect is of the order $\sim -20\,\mu$s when observing an object 30 degrees from the Sun's limb, with the effect decreasing as the Sun→object angle increases.

- The Earth's barycentric position is calculated using the SLALIB sla_epv routine which in turn uses a simplified version of the `VSOP2000` planetary theory [4]. Comparison by TAL with the `INPOP10a` ephemeris [5] over the range 2005–2031 produced an RMS error of 3.83 km or 12.8 μs.

- The accuracy on the mean to apparent place transformation (used for airmass) is sub-milliarcsecond, limited by the precession-nutation model (IAU 1976 precession, Shirai & Fukushima 2001 forced nutation and precession corrections [6]), which is also used to transform the Earth's position to J2000.0 from the true equator and equinox of date.

- The UTC $\rightarrow$ TT correction needs to know the correct number of leapseconds which relies on the SLALIB sla_dat routine being updated and the code recompiled when a new leap second is announced by the IERS.

The error on the time correction is most likely to be dominated by 1) uncertainty on the NTP synchronization of the instrument control computer ($\sim$1–5 ms), 2) uncertainty in the time-stamping of the exposure start ($\sim$100 μs), and 3) uncertainty in the photon-weighted midtime of the exposure.

4.4.10   Photometric calibration

Stefano's code...

5   Data files

5.1   Filenames and locations

Raw LCOGT data conforms to the following format:
`<site><tel. class><tel. #>-<inst>-YYYYMMDD-<nnnn>-<type><red. level>.fits`
where:

- `<site>` is the LCOGT Network Site (one of {ogg, coj, bpl, lsc, sqa, cpt, elp, tfn}),

- `<tel. class>` is the telescope class (one of {2m0, 1m0, 0m8, 0m4}),

- `<tel. #>` is the telescope number,

- `<inst>` is the (unique) CCD descriptor,

- `YYYYMMDD` is the UTC date at start of night (defined to be 16:00 local),

- `<nnnn>` is the running number, restarting at 0001 when UTC date increments,

- `<type>` is the type of file (from {a=arc, b=bias, d=dark, e=exposure, f=sky flat, l=lambert flat, s=standard, x=experimental})

- `<red. level>` is the reduction level (00-99, currently one of {00 (raw data), 01 (flash processed), 02 (pre-processed), 10 (quicklook processed) 90 (pipeline processed)})

For ease of tracking data flow, the pipeline products will follow the same policy, with the `<red. level>` altered appropriately to 90 in the case of normally processed data. Catalogs derived from frames will inherit the same rootname with a `_cat.fits` suffix. Bitmap versions of the frame will use the same rootname, replacing the `.fits` extension with `.png`.

5.2 File suffixes

Processing Frame suffixes

Suffix	Kept	Description
`_raw`	N	The raw frame
`_bp`	N	Bad pixels masked
`_db`	N	Bias frame removed
`_bp_db`	N	Bad pixels masked and bias frame removed
`_ff`	N	Flat fielded

Group suffixes

Not used.

FITS Processed Frame suffixes

Suffix	Kept	Description
`_{a,b,d,e,f,g,w}00`	Y	The raw frame
`_{a,b,d,e,f,g,w}01`	N	Pre-processed frame
`_{a,b,d,e,f,g,w}02`	N	Flash reduced
`_{e,s}10`	Y	QuickLook processed
`_{e,s}90`	Y	Offline processed

References

[1] Brown, T. M. et al., 2013, *Las Cumbres Observatory Global Telescope Network*, PASP, 125, 1031–1055 1

[2] Eastman, J., Siverd, R. & Gaudi, B. S., 2010, *Achieving Better Than 1 Minute Accuracy in the Heliocentric and Barycentric Julian Dates*, PASP, 122, 935 4.4.9

[3] Petit, G. & Luzum, B. (eds), 2010, *IERS Conventions (2010)*, IERS Technical Note No. 36, Chap. 10 4.4.9

[4] Moisson, X. & Bretagnon, P., 2001, Celestial Mechanics and Dynamical Astronomy, Volume 80, 3/4, 205–213 4.4.9

[5] Fienga, A., Laskar, J., Kuchynka, P., Manche, H., Desvignes, G., Gastineau, M., Cognard, I., Theureau, G., 2011, *The INPOP10a planetary ephemeris and its applications in fundamental physics*, Celestial Mechanics and Dynamical Astronomy, Volume 111, Issue 3, pp.363-385 DOI: 10.1007/s10569-011-9377-8 4.4.9

[6] Shirai, T. & Fukushima, T., 2001, *Construction of a new forced nutation theory of the nonrigid Earth* , AJ, 121, 3270–3283. 4.4.9

[7] Zacharias, N., Finch, C., Girard, T., Hambly, N., Wycoff, G., Zacharias, M. I., Castillo, D., Corbin, T., DiVittorio, M., Dutta, S., Gaume, R., Gauss, S., Germain, M., Hall, D., Hartkopf, W., Hsu, D., Holdenried, E., Makarov, V., Martinez, M., Mason, B., Monet, D., Rafferty, T., Rhodes, A., Siemers, T., Smith, D., Tilleman, T., Urban, S., Wieder, G., Winter, L., Young, A., 2010, *The Third US Naval Observatory CCD Astrograph Catalog (UCAC3)*, AJ, 139, 2184–2199. 4.4.4

[8] Skrutskie, M. F., Cutri, R. M., Stiening, R., Weinberg, M. D., Schneider, S., Carpenter, J. M., Beichman, C., Capps, R., Chester, T., Elias, J., Huchra, J., Liebert, J., Lonsdale, C., Monet, D. G., Price, S., Seitzer, P., Jarrett, T., Kirkpatrick, J. D., Gizis, J. E., Howard, E., Evans, T., Fowler, J., Fullmer, L., Hurt, R., Light, R., Kopan, E. L., Marsh, K. A., McCallon, H. L., Tam, R., Van Dyk, S., Wheelock, S., *The Two Micron All Sky Survey (2MASS)*, 2006, AJ, 131, 1163–1183. 4.4.4

[9] Monet, D. G., Levine, S. E., Canzian, B., Ables, H. D., Bird, A. R., Dahn, C. C., Guetter, H. H., Harris, H. C., Henden, A. A., Leggett, S. K., Levison, H. F., Luginbuhl, C. B., Martini, J., Monet, A. K. B., Munn, J. A., Pier, J. R., *The USNO-B Catalog*, 2003, AJ, 125, 984–993. 4.4.4

[10] Høg, E., Fabricius, C., Makarov, V. V., Bastian, U., Schwekendiek, P., Wicenec, A., Urban, S., Corbin, T., Wycoff, G., *Construction and verification of the Tycho-2 Catalogue*, 2000, A&A, 357,367–386. 4.4.4

A   Alphabetical list of LCOGT recipes

OFFLINE_REDUCTION Recipe for processing imaging science data

QUICK_LOOK QuickLook processing recipe for imaging science data

REDUCE_BIAS Process bias observations and form a masterbias

REDUCE_DARK Process dark observations and form a masterdark

SKY_FLAT Process sky flat observations and form a masterflat

B   Classified list of LCOGT primitives

LCOGT primitives may be classified in terms of their camera class and observing mode as follows. Note that a lot of the **LCOSBIG** primitives are common to the other instrument classes unless it has been overridden by an instrument-specific version. The instrument-specific primitives are used in preference to the generic ones when reducing that class of instrument data i.e. the **LCOSINISTRO** version of _MAKE_BIAS_FROM_GROUP_ will be used instead of the **LCOSBIG** one when reducing **LCOSINISTRO** data.

SBIG Camera (LCOSBIG) imaging data:

Bias observations

_FILE_LIBRARY_BIAS_ File a copy of the masterbias SDF, PNG and calibration index files in the calibration library

_MAKE_BIAS_FROM_GROUP_ Makes a masterbias from the current group of frames.

_REDUCE_BIAS_HELLO_ Sets up data-reduction tasks and data for REDUCE_BIAS recipes.

_REDUCE_BIAS_STEER_ Steers processing for REDUCE_BIAS recipe.

_REDUCE_BIAS_TIDY_ Removes unwanted intermediate files for the REDUCE_BIAS recipe.

Dark observations

_FILE_LIBRARY_DARK_ File a copy of the masterdark SDF, PNG and calibration index files in the calibration library

_MAKE_DARK_FROM_GROUP_ Makes a masterdark from the current group of frames.

_REDUCE_DARK_STEER_ Steers processing for REDUCE_DARK recipe.

_REDUCE_DARK_TIDY_ Removes unwanted intermediate files for the REDUCE_DARK recipe.

_REMOVE_BIAS_ Subtracts a bias frame.

Flat observations

_FILE_LIBRARY_FLAT_ File a copy of the masterflat SDF, PNG and calibration index files in the calibration library

_FLAT_QC_ To perform Quality Control on flat fields produced by this pipeline.

_MAKE_FLAT_FROM_GROUP_ Makes a masterflat from the current group of frames.

_REMOVE_BIAS_ Subtracts a bias frame.

_SKY_FLAT_STEER_ Steers processing for SKY_FLAT recipes.

_SUBTRACT_DARK_ Subtracts a dark frame.

_SUBTRACT_DARK_NO_THRESH_ Subtracts a dark frame without thresholding of remaining hot pixels.

Science observations

_ADD_AUTO_ASTROMETRY_ Performs automated astrometric corrections.

_CALCULATE_IMAGE_STATISTICS_ Calculate various image quality statistics based on an input catalog(ue).

_CALCULATE_SEEING_STATS_ Extract objects and determine the average FWHM, ellipticity, and position angle across a field.

_CALCULATE_ZEROPOINT_ Calculate various image quality statistics based on an input catalogue.

_CREATE_IMAGE_CATALOGUE_ Performs source extraction and photometry on all sources.

_DATA_QC_TEST_ Evaluates the image data quality control flag and sets the relevant bitmask keywords in the data catalogue product.

_DERIVED_PRODUCTS_ Create derived products from the processed frames.

_DERIVED_PRODUCTS_STEER_ Steers processing for DERIVED_PRODUCTS recipe.

_DIVIDE_BY_FLAT_ Divides an object frame by a flatfield frame.

_IMG_QC_READOUT_TEST_ Primitive description

_REMOVE_BIAS_ Subtracts a bias frame.

_SUBTRACT_DARK_ Subtracts a dark frame.

_SUBTRACT_DARK_NO_THRESH_ Subtracts a dark frame without thresholding of remaining hot pixels.

_SET_QC_FLAGS_ Evaluate and set Quality Control flags.

Miscellaneous

_ADD_PIPELINE_VERSION_ Adds the current pipeline version to the file.

_CHECK_PROPID_ Checks whether the proposal id in the frame header is known to the science archive.

_CLIPPED_STATS_MEDIAN_ Finds the clipped mean, median and standard deviation of a frame.

_CONVERT_TO_FITS_ Converts current observation to FITS.

_CREATE_GRAPHIC_FROM_FILE_ Create a PNG, GIF, or JPG graphic from a given file.

_CREATE_RAW_FRAME_ Creates a raw frame in ORAC_DATA_OUT.

_DELETE_RAW_FRAME_ Remove the raw frame file.

_GET_FILTER_PARAMETERS_ Returns LCOGT optical filter characteristics.

_GET_GAIN_ Finds the LCOGT instrument gain in electrons per ADU for the current Frame.

_GET_READNOISE_ Finds the LCOGT instrument readnoise in electrons for the current Frame.

_GET_SATURATION_LEVEL_ Finds the LCOSBIG saturation level in ADU the current Frame.

_IMBOX_STATS_ Primitive description

_INSTRUMENT_HELLO_ Performs the LCOSBIG-specific imaging setup.

_LCOGT_STANDARD_MAGNITUDE_ Obtains the catalogue magnitude of a LCOGT standard.

_NIGHT_LOG_ Produces a text file log of a night's imaging observations.

_OFFLINE_REDUCTION_HELLO_ Sets up data-reduction tasks and data for OFFLINE_REDUCTION recipes.

_QUICK_LOOK_HELLO_ Sets up data-reduction tasks and data for QUICK_LOOK recipes.

_SET_ORIGIN_ Sets the origin of an observation.

_STANDARD_MAGNITUDE_ Obtains the catalogue magnitude of a standard.

_TURN_ON_HISTORY_ Switches on history recording.

Context Camera (LCOCC) imaging data:

_CALCULATE_ZEROPOINT_NULL_ Null primitive for Context Camera

_GET_FILTER_PARAMETERS_ Return central wavelength, extinction and zeropoint for a given filter for the ContextCamera.

Sinistro Camera (LCOSINISTRO) imaging data:

_CALCULATE_ZEROPOINT_ alculate various image quality statistics based on an input catalogue.

_CREATE_IMAGE_CATALOGUE_ source extraction and photometry on all sources.

_DERIVED_PRODUCTS_ Create derived products from the processed frames.

_FLAT_QC_ To perform Quality Control on flat fields produced by this pipeline.

_GET_FILTER_PARAMETERS_ Returns LCOGT optical filter characteristics.

_GET_GAIN_ Finds the LCOGT Sinistro instrument gain in electrons per ADU for the current Frame.

_GET_READNOISE_ Finds the LCOGT Sinistro instrument readnoise in electrons for the current Frame.

_GET_SATURATION_LEVEL_ Finds the LCOSINISTRO saturation level in ADU the current Frame.

_MAKE_BIAS_FROM_GROUP_ Makes a masterbias from the current group of frames.

_MAKE_DARK_FROM_GROUP_ Makes a masterdark from the current group of frames.

_MAKE_FLAT_FROM_GROUP_ Makes a masterflat from the current group of frames.

Spectral Instruments Camera (LCOSPECTRAL) imaging data:

_CREATE_RAW_FRAME_ Creates a raw frame in ORAC_DATA_OUT, removing legacy header items if present.

_GET_GAIN_ Finds the LCOGT Spectral instrument gain in electrons per ADU for the current Frame.

_GET_READNOISE_ Finds the LCOGT Spectral instrument readnoise in electrons for the current Frame.

_INSTRUMENT_HELLO_ Performs the LCOSPECTRAL-specific imaging setup.

FLI Autoguider Camera (LCOFLI) imaging data:

_ADD_PIPELINE_VERSION_ Adds the current pipeline version to the file.

_CREATE_RAW_FRAME_ Creates a raw frame in ORAC_DATA_OUT, removing legacy header items if present.

_MAKE_DARK_FROM_GROUP_ Makes a masterdark from the current group of frames.

FLOYDS Andor Camera (LCOFLOYDS) spectroscopy data:

_ADD_PIPELINE_VERSION_ Adds the current pipeline version to the file.

_DETERMINE_NREADS_ Determine the number of array reads per exposure.

_DETERMINE_WAVEBAND_ Determine the waveband for flux calibration and blackbody calculations.

_FIX_EXTRA_HEADERS_ Null primitive for LCOFLOYDS.

_MAKE_BIAS_FROM_GROUP_ Makes a masterbias from the current group of frames.

_MAKE_RAW_FILE_ Copies raw data to a _mraw file.

_REDUCE_BIAS_HELLO_ Sets up data-reduction tasks and data for the REDUCE_BIAS recipe.

_REDUCE_BIAS_STEER_ Steers processing for the REDUCE_BIAS recipe.

_REDUCE_BIAS_TIDY_ Removes unwanted intermediate files for the REDUCE_BIAS recipe.

C   Specifications of LCOGT recipes

The following pages describe the LCOGT recipes in detail. Recipes in general live in sub-directories under `$ORAC_DIR/recipes` (this is the general location for all recipes for imaging, spectroscopy, IFU's and general purposes). For LCOGT use, the majority of recipes live under `$ORAC_DIR/recipes/imaging/LCOSBIG` or a camera-class specific subdirectory (e.g. `$ORAC_DIR/recipes/imaging/LCOSINISTRO`). ORAC-DR searches for recipes via a 'recipe search path' (controlled by the `orac_determine_recipe_search_path()` subroutine in `ORAC::Inst::Defn`) which is ordered in decreasing order of specificity e.g. for LCOSINISTRO the order will be:

- `$ORAC_DIR/recipes/LCOSINISTRO/`

- `$ORAC_DIR/recipes/imaging/LCOSINISTRO/`

- `$ORAC_DIR/recipes/LCOSBIG/`

- `$ORAC_DIR/recipes/imaging/LCOSBIG/`

- `$ORAC_DIR/recipes/imaging/`

The recipe search path can also be overridden with an environment variable if needed or a specific recipe to be run can be specified on the command line call to `oracdr`.

OFFLINE_REDUCTION
Reduce a normal imaging EXPOSE observation

Description:

It performs a debiasing, bad-pixel masking, dark subtraction, flat-field division, automatic astrometry, object detection and catalogue creation and converts to a PNG graphic and FITS version of the reduced frame. See the "Notes" for further information.

Notes:

All intermediate frames are deleted.

QUICK_LOOK
Perform a basic reduction of EXPOSE observations to give a "quick look" at the data

Description:

It performs a debiasing, bad-pixel masking, dark subtraction, flat-field division, automatic astrometry, object detection and catalogue creation and converts to a PNG graphic and FITS version of the reduced frame. See the "Notes" for further information.

Notes:

All intermediate frames are deleted. The resulting output PNG graphic and FITS frames and catalog files have -{e,s}10.fits suffixes, rather than the normal -{e,s}90.fits

REDUCE_BIAS
Reduce a imaging BIAS observation

Description:

Reduces a imaging BIAS observation, including coadding multiple integrations. Files the reduced bias frame for use by subsequent reduction of STARE and CHOP mode data.

Notes:

Creates a variance array for the bias frame, determined from the variance of the multiple integrations in the bias observation. There should be at least 3 integrations in a BIAS observation.

See Also :

The _REDUCE_BIAS_ imaging primitive.

REDUCE_DARK
Averages and files observations as the current dark

Description:
> This recipe reduces dark-frame observations from optical imaging data. Individual darks are scaled by the exposure time before being averaged. It files the single or averaged dark in the dark index file. Other reduction steps comprise bad-pixel masking, optional creation of data errors.

Notes:

- The bad-pixel mask applied is the binning-appropriate match taken from the $ORAC_DATA_CAL/index.mask calibration index.
- Intermediate frames are deleted.

Configurable Steering Parameters

USEVAR = LOGICAL
> Whether or not to create and propagate variance arrays. [0]

Output Data :

- The dark called dark_<instrument>_<UT night>_<Xbin>_<Ybin> where <instrument> is the LCOGT instrument identifier and <Xbin>, <Ybin> are the binning factors in X and Y.
- The dark is filed in $ORAC_DATA_OUT/index.dark after checking whether it's mean is within the allowed tolerance.
- If $FINAL_DATA_OUT is set, the masterdark is converted to FITS and copied there and a PNG version of the masterdark is created and moved to $FINAL_DATA_OUT.

Implementation Status:

- The processing engines are from the Starlink packages KAPPA and CCDPACK.
- Uses the Starlink NDF format.
- History is recorded within the data files.
- The title of the data is propagated through the intermediate file to the dark.

SKY_FLAT
Creates and files a flat field derived from dithered frames

Description:

This recipe makes a sky flat for LCOGT imaging from a series of sky or Lambert flat frames which are combined using one of a selection of statistics. It expects a series of dithered sky frames in different filters.

It performs debiasing, bad-pixel masking, and dark subtraction before combining normalised frames pixel by pixel using the median. Details of the flat are filed in the index of flats for future selection and use of the flat. See the "NOTES" for further details.

For best results the field observed should contain few stars and no bright ones. Ideally the telescope should have been pointed to the "flat spot" - usually 5 to 10 degrees away from the zenith in the anti-solar direction to minimize gradients in the resultant flat.

Notes:

- The bad-pixel mask applied is the binning-appropriate match taken from the $ORAC_DATA_CAL/index.mask calibration index.
- Intermediate frames are deleted.

Configurable Steering Parameters

NUMBER = INTEGER

The number of frames in the jitter. If absent, the number of offsets, as given by header NOFFSETS, minus one is used. If neither is available, 5 is used. An error state arises if the number of dithered frames is fewer than 3. []

USEVAR = LOGICAL

Whether or not to create and propagate variance arrays. [0]

Output Data :

- The created flat fields are of the form
  flat_<instrument>_<UT night>_<flattype>_bin<Xbin>x<Ybin>_<filter>[_c<cycle_number>]
  where <instrument> is the LCOGT instrument identifier, <flattype> is the type of flat (either 'SKYFLAT' or 'FAKEFLAT' (a unity flat), <filter> is the common name of the filter (e.g. 'R') and <Xbin>, <Ybin> are the binning factors in X and Y. If a flat of the same name already exists, a <cycle_number> will be added, counting from one.
- The flats are filed in $ORAC_DATA_OUT/index.flat.

Implementation Status:

- The processing engines are from the Starlink packages: CCDPACK, KAPPA, and FIGARO.
- Uses the Starlink NDF format.
- History is recorded within the data files.
- The title of the data is propagated through the intermediate file to the dark.
- Error propagation is controlled by the USEVAR parameter.

D   Specifications of LCOGT primitives

The following pages describe the LCOGT primitives in detail. Primitives in general live in subdirectories under `$ORAC_DIR/primitives` (this is the general location for all primitives for imaging, spectroscopy, IFU's and general purposes). For LCOGT use, the majority of primitives live under `$ORAC_DIR/primitives/imaging/LCOSBIG` or a camera-class specific subdirectory (e.g. `$ORAC_DIR/primitives/imaging/LCOSINISTRO`). ORAC-DR searches for primitives via a 'primitive search path' (controlled by the `orac_determine_primitive_search_path()` subroutine in `ORAC::Inst::Defn`) which is ordered in decreasing order of specificity e.g. for LCOSINISTRO the order will be:

- `$ORAC_DIR/primitives/LCOSINISTRO/`

- `$ORAC_DIR/primitives/imaging/LCOSINISTRO/`

- `$ORAC_DIR/primitives/LCOSBIG/`

- `$ORAC_DIR/primitives/imaging/LCOSBIG/`

- `$ORAC_DIR/primitives/imaging/`

- `$ORAC_DIR/primitives/general/`

The primitive search path can also be overridden with an environment variable if needed.

_ADD_AUTO_ASTROMETRY_
Performs automated astrometric corrections

Description:

This primitive automatically corrects astrometry for a given observation. It does so by downloading a catalogue from a source (typically UCAC-3), detects objects in the observation, correlates between the two catalogues, and calculates an astrometric solution.

Notes:

- This primitive is suitable for optical imaging instruments.
- Processing only occurs when the steering header ADD_ASTROMETRY is true.
- Processing occurs on the current Group object.
- Astrometric correction will probably fail if fewer than five objects are detected in the frame. Should this occur, the original WCS will be retained.
- Should an astrometric solution be found, the WCS in the current Group will be overwritten with the solution.

Required Perl Modules :

Starlink::Autoastrom.

_ADD_PIPELINE_VERSION_
Adds the current pipeline version to the file

Description:

This primitive adds the current pipeline version to the file

Notes:

- This primitive is suitable for LCOGT optical imaging instruments.
- This primitive needs to have SVN version properties set to update properly

Tasks :

KAPPA FITSMOD

_CALCULATE_IMAGE_STATISTICS_
Calculate various image quality statistics based on an input catalogue

Description:

This primitive calculates image quality statistics based on an input catalogue. This catalogue is typically the output from either Starlink::Extractor or Starlink::Autoastrom, so that object morphology information is available for calculations.

This primitive calculates the mean axial ratio of bright sources, the average object diameter converted to K-band equivalent, the sky brightness, the limiting magnitude, and the instrumental zero point.

Notes:

- Files results with the Calibration system. FWHM is obtained with the fwhm() method, axial ratio is obtained with the axial_ratio() method, sky brightness is obtained with the sky_brightness() method, limiting magnitude is obtained with the limiting_mag() method, and the zero point is obtained with the zeropoint() method.

Output Data :

None.

Required Perl Modules :

Astro::Catalog.

_CALCULATE_SEEING_STATS_
Extract objects and determine the average FWHM, ellipticity, and position angle across a field

Description:

For the group file, this primitive finds good sources in the field and calculates the average FWHM, ellipticity, and position angle. It then displays these averages along with errors.

Notes:

- Currently uses SEXTRACTOR for source extraction.
- The SEXTRACTOR configuration file is found in $ORAC_DATA_CAL/extractor_seeing_stats.sex.
- This primitive runs only when the CALCULATE_SEEING_STATS internal header is true.
- This primitive only operates on the current Group file.

_CALCULATE_ZEROPOINT_
Calculate various image quality statistics based on an input catalogue

Description:

This primitive calculates image quality statistics based on an input catalogue. This catalogue is typically the output from either Starlink::Extractor or Starlink::Autoastrom, so that object morphology information is available for calculations.

This primitive calculates the mean axial ratio of bright sources, the sky brightness, the limiting magnitude, a transparency estimate and the instrumental zero point.

Notes:

- Files results with the Calibration system. FWHM is obtained with the fwhm() method, axial ratio is obtained with the axial_ratio() method, sky brightness is obtained with the sky_brightness() method, limiting magnitude is obtained with the limiting_mag() method, and the zero point is obtained with the zeropoint() method.

Output Data :

None.

Required Perl Modules :

Astro::Catalog.

_CALCULATE_ZEROPOINT_NULL_
Files a null set of zeropoint parameters with the Calibration system.

Description:

    Null primitive for LCOCC.

Notes:

- This primitive is suitable for LCOCC.

_CHECK_PROPID_
Checks whether the proposal id in the frame header is known to the science archive

Description:

The primitive checks whether the proposal id in the frame header is known to the IPAC Science Archive or the LCOGT Proposal DB, depending on the setting of $test_proposaldb.

Notes:

- This primitive is suitable for imaging cameras.
- Processing only occurs for object frames.
- Email sending requires Mail::Sendmail to be installed
- The files to be checked are $ORAC_CAL_ROOT/lcogt_propid.txt or $ORAC_CAL_ROOT/proposal_list depending on whether the IPAC Science Archive or the LCOGT Proposal DB is being checked.

Required Perl Modules :

Mail::Sendmail

_CLIPPED_STATS_MEDIAN_
Finds the clipped mean, median and standard deviation of a frame

Description:

Use progressive sigma-clipping to find a representative mean, median and standard deviation of a frame. The default clipping thresholds give a reasonable approximation to the mode.

Notes:

- This primitive is suitable for all instruments.

Tasks :

KAPPA: STATS.

_CONVERT_TO_FITS_
Converts current observation to FITS

Description:

  Converts current observation to FITS.

Notes:

- Bad pixels are replaced with 65535 as the previous default FITS NaN caused problems for some other tools.
- The FITS-WCS encoding is set to "FITS-WCS(CD)" to describe the rotation and scaling as a CD matrix rather than as PC matrix with CDELT values.
- The ORIGIN is set to 'LCOGT' .
- Only the Data component is converted, along with the FITS extension of the NDF and any NDF history records - any other NDF extensions (such as CCDPACK's) are not converted.

Tasks :

  CONVERT, NDF2FITS; KAPPA: NOMAGIC

_CREATE_GRAPHIC_FROM_FILE_
Create a PNG, GIF, or JPG graphic from a given file

Description:

This primitive creates a PNG, GIF, or JPG graphic from the supplied file. It currently only supports 1-D and 2-D files.

Notes:

- This primitive is suitable only for 1-D and 2-D input files.
- The graphic file will have the same root filename as the input file, with the appropriate file extension.

_CREATE_IMAGE_CATALOGUE_
source extraction and photometry on all sources

Description:

For the group file, find all the sources and calculate the flux of each detected source. Write the results to a catalogue file. The resulting catalogue file is in FITS_1.0 format with columns defined to produce an LCOGT Standard Catalogue Product.

Notes:

- Currently uses SEXTRACTOR for source extraction and for photometry.
- The output catalogue is not completely in LCOGT Standard Catalogue Product form and requires post-processing by timecorrect.py in order to propogate the image headers, calculate the missing AIRMASS and TCORR columns and splitting the FLUX_APER and FLUXERR_APER 2D arrays into 8 separate columns.

Required Perl Modules :

Starlink::Extractor.

Author :

Brad Cavanagh <b.cavanagh@jach.hawaii.edu> Tim Lister <tlister@lcogt.net>

_CREATE_RAW_FRAME_
Creates a raw frame in ORAC_DATA_OUT

Description:

Null primitive for LCOSBIG.

Notes:

- This primitive is suitable for LCOSBIG.

_DATA_QC_TEST_
Evaluates the image data quality control flag and sets the relevant bitmask keywords in the data catalogue product

Description:

Evaluates the image data quality control flag and sets the relevant bitmask keywords in the data catalogue product.

Notes:

- This primitive is suitable for imaging cameras.
- Processing only occurs for object frames.

Output Data :

The results of the comparison are returned in $_DATA_QC_TEST_{QCPARAM} and will contain either 'T' if the comparison passed, 'F' if the comparison failed or 'U' if the comparison couldn't be made.

Output Files :

None.

_DELETE_RAW_FRAME_
Remove the raw frame file

Description:

Generic primitive to remove the actual raw frame file.

Notes:

In rare cases you do not care about the Frame data product (maybe because the group product is the only product of interest) and do not want it to remain on disk. This primitive will erase the current frame file.

_DERIVED_PRODUCTS_STEER_
Steers processing for DERIVED_PRODUCTS recipe

Description:

This primitive control processing for DERIVED_PRODUCTS recipe through steering headers listed below.

Notes:

- This primitive is suitable for LCOGT imaging CCD cameras.
- Processing only occurs for object frames.

Steering Headers :

ADD_ASTROMETRY = LOGICAL Whether or not automated astrometry is to occur. This equates to argument DOASTROM. CREATE_CATALOGUE = LOGICAL Whether or not catalogue creation is to occur. This equates to argument DOIMGCAT. CALCULATE_SEEING_STATS = LOGICAL Whether or not seeing statistics should be calculated. This equates to argument DOSEEING. QUICK_LOOK = LOGICAL Whether or not to perform quicklook processing. This equates to argument DOQUICKLOOK.

_DERIVED_PRODUCTS_
Create derived products from the processed frames

Description:

This primitive creates the derived products from the processed frames. The following items are performed:

- headers are updated with the details of the master calibration frames and correction steps used,
- the saturation and trimmed section are updated with the values used in the pipeline processing,
- A clipped mean, median and sigma are calculated and added to the header,
- the QC modules is run to produce the four QC flags in the headers,
- incorrect values of the TAGID and/or PROPID are flagged and corrected and the PROPID is checked against the lists of known proposals,
- access rights and state of the data are set,
- a 512x512 PNG thumbnail of the processed image is created,
- the frame is converted to FITS format
- the pytimecorrect.py code is called to compute per-star barycentric time correction and airmass and updates the header of the BCD image and source catalog,
- if a final destination is defined by the enviroment variable FINAL_DATA_OUT, the derived products are moved there.

Notes:

- This primitive is suitable for imaging cameras.
- Processing only occurs for object frames.
- Various incorrect/invalid values of the PROPID and TAGID are corrected and over-written.
- Data is normally set to private with a publication date of 1 year from the frames' date. Data with a PROPID beginning with "ENG" or Engineering is set to private indefinitely. EPO with a PROPID or TAGID beginning with LCOEPO|FTPEPO|HAWEPO is set to public immediate.

External Tasks :

The following external tasks are used: KAPPA FITSMOD ORAC-DR PRIMITIVES _ADD_PIPELINE_VERSIO _CHECK_PROPID_, _CLIPPED_STATS_MEDIAN_, _CONVERT_TO_FITS_, _CREATE_GRAPHIC_FROM_ _GET_SATURATION_LEVEL_, _SET_FILE_FITS_ITEM_, _SET_QC_FLAGS_

Output Files :

Depending on whether the Frame uhdr entry QUICK_LOOK is set to 1 or not, the 00_bp_ff
suffix of the processed SDF frame is replaced with a 10.fits or 90.fits when converting to
the output FITS frame. The processed SDF frame is also converted to a PNG image with a
10.png or 90.png suffix, depending on whether the Frame uhdr entry QUICK_LOOK.

_DIVIDE_BY_FLAT_
Flat-fields a frame

Description:

This primitive divides the current frame by the most-recent and matching flat-field frame from $Cal->flat method.

Notes:

- This primitive is suitable for UFTI, IRCAM, and Michelle in imaging mode.
- Processing only occurs for object and sky frames.
- The flat-fielded image can be displayed.
- The frame title is propagated.

Output Data :

- Flat-fielded frame inheriting the frame's name but with the _ff suffix.

Tasks :

CCDPACK: FLATCOR.

_FILE_LIBRARY_BIAS_
Makes a copy of the masterbias and index file in the calibration library

Description:

This primitive makes a copy of the masterbias and index.bias from the current directory to calibration library (pointed to by the ORAC_DATA_CAL environment variable.

Notes:

- This primitive is suitable for optical imaging instruments.
- Processing only occurs for bias frames, and when the steering header MAKE_BIAS is true.

Output Data :

The masterbias. It is called " bias_<instrument>_<UT night>_bin<Xbin>x<Ybin>" where <instrument> is the LCOGT instrument identifier and <Xbin>, <Ybin> are the binning factors in X and Y.

Tasks :

KAPPA: NDFCOPY.

_FILE_LIBRARY_DARK_
Makes a copy of the masterdark and index file in the calibration library

Description:

This primitive makes a copy of the masterdark and index.dark from the current directory to calibration library (pointed to by the ORAC_DATA_CAL environment variable.

Notes:

- This primitive is suitable for optical imaging instruments.
- Processing only occurs for dark frames, and when the steering header MAKE_DARK is true.

Output Data :

The masterdark. It is called " dark_<instrument>_<UT night>_bin<Xbin>x<Ybin>" where <instrument> is the LCOGT instrument identifier and <Xbin>, <Ybin> are the binning factors in X and Y.

Tasks :

KAPPA: NDFCOPY.

_FILE_LIBRARY_FLAT_
Makes a copy of the masterflat and index file in the calibration library

Description:

This primitive makes a copy of the masterflat and index.flat from the current directory to calibration library (pointed to by the ORAC_DATA_CAL environment variable.

Notes:

- This primitive is suitable for optical imaging instruments.
- Processing only occurs for flat frames, and when the steering header MAKE_FLAT is true.

Output Data :

The masterflat. It is called " flat_<instrument>_<UTnight>_SKYFLAT_bin<Xbin>x<Ybin>_<filter>" where <instrument> is the LCOGT instrument identifier, <Xbin>, <Ybin> are the binning factors in X and Y and <filter> is the filter.

Tasks :

KAPPA: NDFCOPY.

_FLAT_QC_
To perform Quality Control on flat fields produced by this pipeline

Description:

   This primitive looks for the separate master flat fields produces in each filter at both the start and the end of a single night of data. It performs the following tests:

   - If morning and evening twilight master flats were produced:
   - Compare the am/pm masters and check the RMS of the residuals is within bounds (implementation of J. Eastman's IDL algorithm).

Notes:

   This is an ORAC-DR/Perl implementation of J. Eastman's IDL algorithm.

External Tasks :

   KAPPA: STATS, SUB ORAC-DR PRIMITIVES _DELETE_A_FRAME_

Output Data :

   None.

_GET_FILTER_PARAMETERS_
Returns LCOGT optical filter characteristics

Description:

This primitive using a switch structure to return two characteristics or relating to the LCOGT optical imaging filters through arguments. Default values are returned if the filter is not recognised.

Notes:

- This primitive is suitable for LCOGT optical photometry.
- The filter name comes from the user header ORAC_FILTER.
- The recognised filters have names ending with U, B, V, R, I, up, gp, rp, ip, zp, zs and Y.
- The mean extinction coefficients are: U: 0.53, B: 0.27, V: 0.14, R: 0.10, I: 0.05, u': 0.56, g': 0.20, r':0.11, i': 0.05, z': 0.04 and Y: 0.03. Zero extinction applies to any other filter. Effective wavelengths are from Bessell (1998); extinction is from AJP's loc_m1_assm.c code as of 2012/05/08 for ELP; M1 inst. zeropoints are from AJP's loc_m1_assm.c code as of 2012/05/08 for 1m0 kb72.

_GET_GAIN_
Finds the LCO instrument gain in electrons per ADU for the current Frame

---

Description:

This primitive obtains the gain in electrons per ADU for the current frame. It first attempts to find a value from the ORAC_GAIN header. If this is null or less than 3.0, the primitive uses a time-dependent default value, and it reports the use of the default.

The gain is returned through an argument.

Notes:

- This primitive is suitable for LCO optical instruments.

_GET_READNOISE_
Finds the LCO instrument readnoise in electrons for the current Frame

Description:

This primitive obtains the readnoise in electrons for the current frame. It first attempts to find a value for the calibration system. Where there is no value, it tries a header for the value, and if that's not defined, the primitive assigns a default.

The readnoise is returned through an argument.

Notes:

- This primitive is suitable for LCO optical instruments.
- The read noise comes from the readnoise calibration, or failing that the header RDNOISE.

_GET_SATURATION_LEVEL_
Finds the LCOSBIG saturation level in ADU the current Frame

Description:

This primitive obtains the LCOSBIG saturation level in ADU for the current frame.

The saturation level is returned through an argument.

Notes:

- This primitive is only suitable for LCOSBIG.
- The values are estimated from fullwell and gain on the instrument's Web page, viz. http://www.sbig.com/large_format/6303E_specs.htm.

_IMBOX_STATS_
Evaluates basic statistics for a given pixel range of an image

Description:

Evaluates basic statistics (sigma-clipped mean and sigma) for a given pixel range of an image

Notes:

- This primitive is suitable for imaging cameras.
- Processing only occurs for object frames.

Output Data :

- The mean and sigma are returned

_IMG_QC_READOUT_TEST_
Test the frame for readout errors

Description:

To test the readout of the frame data for common symptoms of readout errors.

Notes:

- This primitive is suitable for imaging cameras.
- Processing only occurs for object frames.

Output Data :

The following computed values are returned as primitive variables:

- The image quality control flag (set to 2048 in the case of readout problems).
- Mean and sigma of the whole frame.
- Mean and sigma of the four individual quadrants of the image.

_INSTRUMENT_HELLO_
Performs the LCOSBIG-specific imaging setup

Description:

This primitive is performs the instrument specific setup for imaging. It's needed for the generic _IMAGING_HELLO_. In this case it merely reports that the set-up operations are complete.

Notes:

- This primitive is suitable for LCOSBIG.

_LCOGT_STANDARD_MAGNITUDE_
Obtains the catalogue magnitude of a LCOGT standard

Description:

> This primitive reads the faint-standard catalogue or its predecessor. A case- and space-insensitive comparison of the supplied object name with the entries in the table provides a catalogue magnitude in U, B, V, R, or I for a standard star.

Notes:

- This primitive is suitable for LCOGT optical imagers.
- Processing only occurs when it is time to perform photometry, i.e. when the steering header DO_APHOT is true.
- An error occurs when the filter is not one of U, B, V, R, or I.
- The standard-star catalogue used is $ORAC_DATA_CAL/landolt_ubvri.dat. An error results when the catalogue cannot be opened.

_MAKE_BIAS_FROM_GROUP_
Makes a masterbias from the current group of frames

Description:

This primitive makes a master bias from the current group. The primitive files the resultant bias in its calibration index.

Notes:

- This primitive is suitable for optical imaging instruments.
- Processing only occurs for bias frames, and when the steering header MAKE_BIAS is true.
- The bias is displayed.

Output Data :

The masterbias. It is called "bias_<instrument>_<UT night>_bin<Xbin>x<Ybin>" where <instrument> is the LCOGT instrument identifier and <Xbin>, <Ybin> are the binning factors in X and Y.

Tasks :

CCDPACK: MAKEBIAS; KAPPA: FITSMOD, NDFCOPY.

_MAKE_DARK_FROM_GROUP_
Makes a masterdark from the current group of frames

Description:

This primitive makes a master dark from the current group. The primitive files the resultant dark in its calibration index.

Notes:

- This primitive is suitable for optical imaging instruments.
- Processing only occurs for dark frames, and when the steering header MAKE_DARK is true.
- The dark is displayed.

Output Data :

The masterdark. It is called "dark_<instrument>_<UT night>_bin<Xbin>x<Ybin>" where <instrument> is the LCOGT instrument identifier and <Xbin>, <Ybin> are the binning factors in X and Y.

Tasks :

CCDPACK: MAKEDARK; KAPPA: FITSMOD, NDFCOPY.

_MAKE_FLAT_FROM_GROUP_
Makes flats from the current group of frames

Description:

This primitive makes self flats from the current group, one for each distinct observation filter. For eacg flat it uses a median to combine the frames pixel by pixel, and then divides the resultant image by its mean form the flat field. The primitive files the resultant flat in its calibration index.

Notes:

- This primitive is suitable for infrared imaging instruments.
- Processing only occurs for object, sky, or calibration-lamp frames, and when the steering header MAKE_FLAT is true.
- The list of filters present in the group is listed in an array stored by reference in the group user header FILTER_LIST. If this is undefined, only a single flat is made for filter stored in the current Frame's user header ORAC_FILTER.
- There is special behaviour for a combined polarimetry flat (see "OUTPUT DATA"). The string "pol" in the filter name is used to indicate polarimetry data.
- The flat is displayed.

Output Data :

The flat field. It is called is "flat_<filter>_<groupnumber>" for the first cycle, and "flat_<filter>_<groupnumber>_c<cyclenumber>" for subsequent cycles, where <groupnumber> is the frame number of the group, <filter> is the filter name, and <cyclenumber> is the cycle number derived from steering header CYCLE_NUMBER. An exception is for polarimetric data, where the name becomes flat_<filter>_pol<waveplate_angle>_<groupnumber>. The <waveplate_angle> is the integer part of the angle, e.g. 22, 67, from internal header ORAC_WAVEPLATE_ANGLE. Subsequent cycles for polarimetry also have the "_c<cyclenumber>" suffix, but the cycle comes from steering header POL_CYCLE_NUMBER. When steering header WAVEPLATE_FLAT is false (0), copies of the flat are made, one for each angle, using the above nomenclature. Each has its waveplate angle set to its nominal angle. This allows a single ORAC_WAVEPLATE_ANGLE rule entry irrespective of whether all waveplate angles were combined to make a flat or not.

Tasks :

CCDPACK: MAKEFLAT; KAPPA: FITSMOD, NDFCOPY.

_NIGHT_LOG_
Produces a text file log of a night's imaging observations

Description:

    This recipe takes a night's imaging observations, and creates a text file containing a headed tabulation of parameters for each frame.

    The parameters are: observation number, group number, object name, observation type, UT start time, exposure time, number of coadds, read mode and speed, filter, start airmass, frame dimensions in pixels, base equatorial co-ordinates, and data-reduction recipe name.

Notes:

- The <date> comes from the internal header keyword ORAC_UTDATE.
- The logfile created by this primitive does not follow the standard ORAC-DR naming convention (log.xxxx) since it can be used to write log files to directories other than $ORAC_DATA_OUT and unique file names are required.
- Fudges missing or old headers.
- Uses user header ORAC_INSTRUMENT to specify the file name.
- Specification provided by Sandy Leggett.

Output Data :

- The text log file $ORAC_DATA_IN/<date<.nightlog, where <date> is the UT date, unless the OUT argument is set, whereupon the log is in $ORAC_DATA_OUT. This enables a separate on-the-fly log. For the multi-mode instruments UIST, Michelle, IRIS2, and ISAAC the file is $ORAC_DATA_IN/<date>_im.nightlog. For LCOGT data, we make use of the DAY-OBS header keyword to use as the observation date (and hence log root) to prevent writing two logs after we go over the UTC midnight boundary. The on-the-fly log in $ORAC_DATA_OUT is always appended to, being created only if it doesn't exist. Thus multiple entries for the same observation may exist in the on-the-fly log if the pipeline is rerun. The "clean" log file in F$<ORAC_DATA_IN> is re-started if the observation number equals 1 and is appended to otherwise, being created as necessary.

_OFFLINE_REDUCTION_HELLO_
Sets up data-reduction tasks and data for OFFLINE_REDUCTION recipes

Description:

Sets up CCDPACK-related global parameters for OFFLINE_REDUCTION recipes. The settings are as follows.

- The readout bounds in the internal headers ORAC_X_LOWER_BOUND, ORAC_Y_LOWER_BOUND, ORAC_X_UPPER_BOUND, ORAC_Y_UPPER_BOUND define the pixel limits for processing, i.e. there are no bias strips and interpolation direction.
- Error processing is disabled so the readout noise and analogue-to-digital conversions are not specified.
- There is no deferred charge.
- Position list processing tasks expect to find the names of lists stored within NDFs.
- Logging is to the terminal.
- The data type of NDF arrays is preserved.
- Does not detect saturated pixels.
- Parameters are neither saved from or to a 'restoration' file.

The script also performs the following tasks.

- Calls the steering primitive to set steering headers.
-

Notes:

- This primitive is suitable for imaging instruments.

Tasks :

CCDPACK: CCDSETUP.

_QUICK_LOOK_HELLO_
Sets up data-reduction tasks and data for QUICK_LOOK recipes

Description:

Sets up CCDPACK-related global parameters for QUICK_LOOK recipes. The settings are as follows.

- The readout bounds in the internal headers ORAC_X_LOWER_BOUND, ORAC_Y_LOWER_BOUND, ORAC_X_UPPER_BOUND, ORAC_Y_UPPER_BOUND define the pixel limits for processing, i.e. there are no bias strips and interpolation direction.
- Error processing is disabled so the readout noise and analogue-to-digital conversions are not specified.
- There is no deferred charge.
- Position list processing tasks expect to find the names of lists stored within NDFs.
- Logging is to the terminal.
- The data type of NDF arrays is preserved.
- Does not detect saturated pixels.
- Parameters are neither saved from or to a 'restoration' file.

The script also performs the following tasks.

- Calls the steering primitive to set steering headers.
-

Notes:

- This primitive is suitable for imaging instruments.

Tasks :

CCDPACK: CCDSETUP.

_REDUCE_BIAS_HELLO_
Sets up data-reduction tasks and data for REDUCE_BIAS recipes

Description:

This primitive sets up CCDPACK-related global parameters for the REDUCE_BIAS recipe and performs preliminary data reduction.

The CCDPACK settings are as follows.

- The readout bounds in the internal headers ORAC_X_LOWER_BOUND, ORAC_Y_LOWER_BOUND, ORAC_X_UPPER_BOUND, ORAC_Y_UPPER_BOUND define the pixel limits for processing, i.e. there are no bias strips and interpolation direction.
- Error processing is disabled so the readout noise and analogue-to-digital conversions are not specified.
- There is no deferred charge.
- Position list processing tasks expect to find the names of lists stored within NDFs.
- Logging is to the terminal.
- The data type of NDF arrays is preserved.
- Does not detect saturated pixels.
- Parameters are neither saved from or to a 'restoration' file.

The script also performs the following tasks.

- Calls the steering primitive to set steering headers.
- Calls _MASK_BAD_PIXELS_ primitive to mask bad pixels.

Notes:

- This primitive is suitable for imaging instruments.

Steering Headers :

USE_VARIANCE = LOGICAL Whether or not variance processing is to occur. This equates to argument USEVAR.

_REDUCE_BIAS_STEER_
Steers processing for REDUCE_BIAS recipe

Description:

This primitive control processing for REDUCE_BIAS recipe through steering headers listed below.

Notes:

- This primitive is suitable for imaging optical cameras.
- Processing only occurs for bias frames.

Steering Headers :

BIAS_FRAMES = HASH The name of the bias for a given exposure time, and the corresponding number of frames used to create it. USE_VARIANCE = LOGICAL Whether or not variance processing is to occur. This equates to argument USEVAR.

_REDUCE_BIAS_TIDY_
Removes unwanted intermediate files for the REDUCE_BIAS recipe

---

Description:

Removes all intermediate frames. Files are only removed when they are no longer needed, as guided by the steering header MAKE_BIAS.

_REDUCE_DARK_STEER_
Steers processing for REDUCE_DARK recipe

Description:

This primitive control processing for REDUCE_DARK recipe through steering headers listed below.

Notes:

- This primitive is suitable for imaging optical cameras.
- Processing only occurs for dark frames.

Steering Headers :

USE_VARIANCE = LOGICAL Whether or not variance processing is to occur. This equates to argument USEVAR.

_REDUCE_DARK_TIDY_
Removes unwanted intermediate files for the REDUCE_DARK recipe

Description:

Removes all intermediate frames. Files are only removed when they are no longer needed, as guided by the steering header MAKE_DARK.

_REMOVE_BIAS_
Subtracts a bias frame

Description:

This primitive subtracts a zero bias from the current frame; unless the data have variance information and were taken using a non-ND mode (i.e. where the bias has not already been subtracted), whereupon a bias frame, if available, is subtracted.

For most instruments there is no bias to subtract so it is something of a placeholder primitive. Its main purpose is to set up CCDPACK for subsequent processing. For instance, CCDPACK will complain if debiassing is not performed before say flat-fielding. The primitive reports a successful bias subtraction and the frames concerned.

Notes:

- This primitive is suitable for UFTI, IRCAM, INGRID, and Michelle in imaging mode.
- Processing occurs for all frames, and sub-frames therein.
- Where a bias frame is used, it is the most-recent and matching given by $Cal->bias method.
- The observing mode (read type) comes from user header ORAC_DETECTOR_READ_TYPE.
- The bias-subtracted image can be displayed.
- The frame title is propagated.

Output Data :

- Bias-subtracted frame inheriting the frame's name but with the _db suffix.

Tasks :

CCDPACK: DEBIAS.

_SET_ORIGIN_
Sets the origin of an observation

Description:

This primitive sets the origin of an observation, including all integrations, using the ORAC_X_LOWER_BOUND and ORAC_Y_LOWER_BOUND user headers in the frame. It is needed to correct the raw data from a sub-array for which the origin is still at the default. If either header is undefined, the primitive creates a default origin (1,1). The origin is not set if ORAC_X_LOWER_BOUND does not exist.

Notes:

- This primitive is suitable for UFTI, IRCAM, and Michelle and UIST in imaging mode.

Tasks :

KAPPA: SETORIGIN.

_SET_QC_FLAGS_
Evaluates quality control parameters and sets the relevant bitmask keywords in the data catalogue product

Description:

Evaluates quality control parameters and sets the relevant bitmask keywords in the data catalogue product.

Notes:

- The definitions of the respective bitmasks for the Quality Control values are also given in the LCOGT Pipeline/Science Archive Interface Control Document (LCOGT-SA-ICD Version 0.10.0)
- This primitive is suitable for imaging cameras.
- Processing only occurs for object frames.

Output Data :

- The four Quality Control values (QC_OBCON (observing constraints satisfied), QC_IMGST (processed image status>), QC_CATST (source catalogue production status), QC_PHTST (photometric calibration status) along with the ellipticity, FWHM and orientation are filed with Calibration system in index.dqc
- The four Quality Control values are also written into the frame header as L1QOBCON, L1QIMGST, L1QCATST, L1QPHTST keywords.

Steering Headers :

ASTROMETRY_ADDED = INTEGER Whether or not astrometry fitting has been performed on the frame. SEXNDET = INTEGER The number of SExtractor detections found in the image.

_SKY_FLAT_STEER_
Steers processing for SKY_FLAT recipes

Description:

This primitive control processing for SKY_FLAT recipes through steering headers listed below.

Notes:

- This primitive is suitable for imaging infrared cameras.
- Processing only occurs for object, sky, or calibration lamp frames.
- The data are deemed to be polarimetry if the frame internal header ORAC_FILTER contains the string "pol".
- A list of the distinct filters within the group is stored in an array stored by reference in the group user header FILTER_LIST.

Steering Headers :

CYCLE_NUMBER = INTEGER Number of the cycle, a cycle being a set of frames to complete a pass through the recipe. The first cycle is 0. JITTER_NUMBER = INTEGER The number of frames in the jitter. MAKE_FLAT = LOGICAL Whether or not to make the flat. The flat is made once all the jittered target frames in a cycle are available. MASK_OBJECTS = LOGICAL Whether or not to mask the objects. Masking occurs when all the jittered frames in a cycle are available. POL_CYCLE_NUMBER = INTEGER Number of the polarimetry cycle, a cycle being a set of frames to complete a pass through the recipe for all waveplate angles. The first cycle is 0. USE_VARIANCE = LOGICAL Whether or not variance processing is to occur. This equates to argument USEVAR. WAVEPLATE_FLAT = LOGICAL See the argument of the same name. This header merely propagates the value of the argument.

_STANDARD_MAGNITUDE_
Obtains the catalogue magnitude of a standard

Description:

This primitive reads the faint-standard catalogue or its predecessor. A case- and space-insensitive comparison of the supplied object name with the entries in the table provides a catalogue magnitude in U, B, V, R, or I for a standard star.

Notes:

- This primitive is suitable for LCOGT optical imagers.
- Processing only occurs when it is time to perform photometry, i.e. when the steering header DO_APHOT is true.
- An error occurs when the filter is not one of U, B, V, R, or I.
- Invokes _LCOGT_STANDARD_MAGNITUDE_ to obtain the magnitude and catalogue name.

_SUBTRACT_DARK_NO_THRESH_
Subtracts a dark frame

Description:

This primitive subtracts from the current frame the most-recent and matching dark frame given by $Cal->dark method. It reports a successful dark subtraction and the frames concerned.

Notes:

- This primitive is suitable for UFTI, IRCAM, and Michelle in imaging mode.
- Processing only occurs for object and sky frames.
- The dark-subtracted image can be displayed.
- The subtraction assumes the same exposure time for the dark and object frame. That validation should be done by the $Cal->dark method.
- The frame title is propagated.

Output Data :

- Dark-subtracted frame inheriting the frame's name but with the _dk suffix.

Tasks :

CCDPACK: CALCOR.

_SUBTRACT_DARK_
Subtracts a dark frame

Description:

This primitive subtracts from the current frame the most-recent and matching dark frame given by $Cal->dark method. It reports a successful dark subtraction and the frames concerned.

Since transient 'hot' and 'cold' pixels can be present despite the application of a bad-pixel mask, the primitive also thresholds the dark-subtracted frame, setting values beyond the limits to be bad (i.e. undefined), to remove these non-physical values. Such values can lead to problems later in the pipeline. In a sense this processing step augments the bad-pixel mask.

Notes:

- This primitive is suitable for infrared imaging.
- Processing only occurs for object, sky, and flat frames.
- The dark-subtracted image can be displayed.
- The subtraction assumes the same exposure time for the dark and object frame. That validation should be done by the $Cal->dark method.
- The lower threshold limit is the clipped mean (mode) minus five standard deviations, subject to the constraint that the limit lies between -100 and 1. The upper limit is 1000 above the nominal saturation level for the instrument and its mode.
- The primitive issues a warning if the dark-subtracted frame's mode is negative, allowing for the error of the mode. It aborts with an error message if the modal dark-subtracted signal is more than one standard deviation negative.
- The frame title is propagated.

Output Data :

- Dark-subtracted frame inheriting the frame's name but with the _dk suffix.
- An array with bad-value substitution beyond thresholds and inheriting the frame's name but with the _th suffix.

Tasks :

CCDPACK: CALCOR; KAPPA: THRESH.

_TURN_ON_HISTORY_
Switches on history recording

Description:

This primitive enables NDF history recording for each integration in an observation.

Notes:

- This primitive is suitable for UFTI, IRCAM, and Michelle and UIST in imaging mode.
- If the ORAC_HISTORY_OFF environment variable is set, then history will be disabled.

Tasks :

KAPPA: HISSET.

E   Specification of LCOGT Quality Control index files

The following pages describe the LCOGT Quality Control index files in detail. These files are part of the calibration scheme therefore they are controlled by LCOGT-specific subclasses of `ORAC::Calib` (found in `<ORAC-DR root>/src/lib/perl5/ORAC/Calib/`) namely `ORAC::Calib::LCOSBIG` (used for the LCOSBIG, LCOSBIG_0M8 (Sedgwick), LCOSINISTRO, LCOFLI, LCOM-EROPE and LCOSPECTRAL instruments), `ORAC::Calib::LCOSBIG_0M4` (used for LCOS-BIG_0M4 (0.4m) instruments) and `ORAC::Calib::LCOCC` (used for Context Camera instruments). In addition there is the basis of low-resolution spectrographic calibration support through `ORAC::Calib:LCOFLOYDS`.

The columns to appear in the index file are controlled by a list of column names within a subroutine of the same name (e.g. the format of the `index.dqc` file is controlled by a 'dqc' subroutine) and extra columns can be added as desired. Once defined, the ORAC-DR calibration infrastructure handles the location, creation, indexing, updating, retrieval, consistency checking and verification of the index files. In addition to the declared columns, columns for the frame filename and the time are automatically created.

E.1   Specification of the `index.astromqc` file

This index is used to store the result of the astrometric calibration i.e. the WCS fit. The format of the file is a header of the column names (with a hash character in the first column) on the first line followed by a line entry for each frame consisting of multiple colums with the form of `<FILENAME> OFFSETDEC OFFSETRA <ORACTIME> SECPIX XRMS YRMS` where:

- `<FILENAME>` is the name of the reduced frame,

- `NSTARS` is the number of stars used in the astrometric fit (set to 0 if no satisfactory fit was determined),

- `OFFSETDEC` is the offset in Dec in arcseconds between the fitted frame center and the Dec of the telescope (set to 0 if no satisfactory fit was determined),

- `OFFSETRA` is the offset in RA in arcseconds between the fitted frame center and the RA of the telescope (set to 0 if no satisfactory fit was determined),

- `<ORACTIME>` is the auto-inserted time of frame in standard ORAC YYYYMMDD.ddddd format,

- `SECPIX` is the fitted pixel scale in arcseconds/pixel (set to the nominal pixel scale if no satisfactory fit was determined)

- `XRMS` is the RMS of the astrometric fit in the X direction in arcseconds (set to '-99' if no satisfactory fit was determined)

- `YRMS` is the RMS of the astrometric fit in the X direction in arcseconds (set to '-99' if no satisfactory fit was determined)

A section of an example file is shown below:

```
#NSTARS OFFSETDEC OFFSETRA ORACTIME SECPIX XRMS YRMS
lsc1m005-kb78-20150124-0065-s00_bp_ff 0 0 0 20150125.0169907 0.467 -99 -99
lsc1m005-kb78-20150124-0066-s00_bp_ff 0 0 0 20150125.0178704 0.467 -99 -99
lsc1m005-kb78-20150124-0067-s00_bp_ff 1 94.7442281467999 -65.0030048559793 20150125.0189352 0.4
lsc1m005-kb78-20150124-0068-s00_bp_ff 1 93.4636426392003 -64.9209369759612 20150125.0209722 0.4
lsc1m005-kb78-20150124-0069-s00_bp_ff 12 92.9154980000001 -64.8774346599973 20150125.0231713 0.
lsc1m005-kb78-20150124-0070-s00_bp_ff 12 92.8168515195999 -64.9570614880085 20150125.0243056 0.
```

The index file is updated by filing new astrometric fit results with the calibration system in the
_ADD_AUTO_ASTROMETRY_ primitive.

E.2   Specification of the `index.dqc` file

This index is used to store the result of the data quality control. The format of the file is a header
of the column names (with a hash character in the first column) on the first line followed by a
line entry for each frame consisting of multiple colums with the form of <FILENAME> AIRMASS
ELLIPTICITY FWHM <ORACTIME> ORIENT QC_OBCON QC_IMGST QC_CATST QC_PHTST where:

- <FILENAME> is the name of the reduced frame,

- AIRMASS is the mean airmass at the middle of the exposure,

- ELLIPTICITY is the mean ellipticity (0..1) of the sources in the frame determined by SEX-
  TRACTOR (set to -99 if no seeing statistics were determined),

- FWHM is the mean Full Width Half Maximum in arcseconds of the sources in the frame
  determined by SEXTRACTOR (set to -99 if no seeing statistics were determined),

- <ORACTIME> is the auto-inserted time of frame in standard ORAC YYYYMMDD.ddddd
  format,

- ORIENT is the mean orientation in degrees of the sources in the frame fitted by SEXTRACTOR
  (set to -99' if no seeing statistics were determined)

- QC_CATST is the bitmask of flags set by the QC module for the catalog extraction (see the
  LCOGT-SA-ICD Version 0.10.0 or the Science Archive data release notes for the full list of
  flags),

- QC_IMGST is the bitmask of flags set by the QC module for the processed image (see the
  LCOGT-SA-ICD Version 0.10.0 or the Science Archive data release notes for the full list of
  flags),

- QC_OBCON is the string concatanation of tests set by the QC module for the observing
  constraints (see the LCOGT-SA-ICD Version 0.10.0 or the Science Archive data release
  notes for the full list of flags),

- QC_PHTST is the bitmask of flags set by the QC module for the photometric calibration (see
  the LCOGT-SA-ICD Version 0.10.0 or the Science Archive data release notes for the full list
  of flags),

A section of an example file is shown below:

```
#AIRMASS ELLIPTICITY FWHM ORACTIME ORIENT QC_CATST QC_IMGST QC_OBCON QC_PHTST
lsc1m005-kb78-20150124-0061-s00_bp_ff 1.2029843 0.19747 2.746 20150125.0125926 23.140 1 0 UUUUU
lsc1m005-kb78-20150124-0062-s00_bp_ff 1.2049946 0.22837 2.723 20150125.0137037 33.649 1 0 UUUUU
lsc1m005-kb78-20150124-0063-s00_bp_ff 1.2072453 0.21732 2.737 20150125.0150231 25.667 1 0 UUUUU
lsc1m005-kb78-20150124-0064-s00_bp_ff 1.2089885 0.20205 2.628 20150125.0159259 19.913 1 0 UUUUU
```

The index file is updated by filing new seeing statistic results with the calibration system in the
_CALCULATE_SEEING_STATS_ primitive and the index entry is updated to set the QC flags
by the ._SET_QC_FLAGS_ primitive

E.3   Specification of the `index.zeropoint` file

This index is used to store the result of the per-frame photometric zeropoint determination.
The format of the file is a header of the column names (with a hash character in the first col-
umn) on the first line followed by a line entry for each frame consisting of multiple colums
with the form of <FILENAME> AIRMASS EXTINCTION FILTER MAG_LIMIT NCALOBJS <ORACTIME>
SKY_VALUE SKY_VALUE_ERROR SKY_VALUE_MAG TRANSPARENCY Z EROPOINT ZEROPOINT_ERROR ZEROPOINT_SRC
where:

- <FILENAME> is the name of the reduced frame,

- AIRMASS is the mean airmass at the middle of the exposure,

- EXTINCTION is the extinction in magnitudes/airmass in the specific filter (set to '-99' if no
  photometric statistics were determined),

- FILTER is the filter used,

- MAG_LIMIT is the limiting magnitude of the frame (set to '-99' if no photometric statistics
  were determined),

- NCALOBJS is the number of objects used in the zeropoint determination (set to '-99' if no
  photometric statistics were determined),

- <ORACTIME> is the auto-inserted time of frame in standard ORAC YYYYMMDD.ddddd
  format,

- SKY_VALUE is the sky value of the frame in ADU (set to '-99' if no photometric statistics
  were determined),

- SKY_VALUE_ERROR is the variance on sky value of the frame in ADU (set to '-99' if no
  photometric statistics were determined),

- SKY_VALUE_MAG is the sky value of the frame in magnitudes (set to '-99' if no photometric
  statistics were determined),

- TRANSPARENCY is the estimated transparency (0..1) of the frame (set to '-99" if no photomet-
  ric statistics were determined),

- ZEROPOINT is the zeropoint of the frame in magnitudes (set to '-99" if no photometric
  statistics were determined),

- ZEROPOINT_ERROR is the error on the zeropoint of the frame in magnitudes (set to '-99' if no photometric statistics were determined),

- ZEROPOINT_SRC is the source of the zeropoint determination (normally `<primitive name>-version`; set to 'N/A' if no photometric statistics were determined),

A section of an example file is shown below:

```
#AIRMASS EXTINCTION FILTER MAG_LIMIT NCALOBJS ORACTIME SKY_VALUE SKY_VALUE_ERROR SKY_VALUE_MAG
lsc1m005-kb78-20150124-0351-e00_bp_ff 1.9601564 -99 gp -99 0 20150125.3628819 -99 -99 -99 -99 -
lsc1m005-kb78-20150124-0352-e00_bp_ff 1.90434185 -99 gp -99 0 20150125.3665509 -99 -99 -99 -99
lsc1m005-kb78-20150124-0364-e00_bp_ff 1.6940621 0.076346831 rp 15.87 7 20150125.3884028 49.1304
```

The index file is updated by filing new per-frame zeropoint determination results with the calibration system in the _CALCULATE_ZEROPOINT_ primitive.