



# *Tests of bias corrections using the stack*

Thibault Guillemin

With help from Dominique Boutigny

IN2P3-commissioning meeting

June 29, 2022



# Context

- Strategy for bias correction in deep discussion since a few months  
Presentations by Thibault, Jim, Aaron, Pierre
- Need (at minimum) to understand what is implemented in DM and how the corrections are precisely applied
  - ➔ Useful to be able to test alternatives directly in DM
- My first step goals:
  - Understand ISR
  - Be able to run modified ISR on BOT data (using gen3 butler)
  - ➔ master bias and overscan correction methods

Concrete goals:

- 1) 2D overscan correction
- 2) Build and apply a master dark

# Instrument Signature Removal steps in DM Stack

- ISR steps in DM as of July 2020
  - integer-to-float conversion
  - saturation and suspect pixel masking
  - overscan subtraction
    - Optional: Apply crosstalk correction here before CCD assembly, and before trimming
  - CCD assembly of individual amplifiers
  - bias subtraction
    - Note: Calibration products construction (master bias, master dark, master flat).
  - variance image construction
  - linearization of nonlinear response
  - crosstalk correction
  - mask defects, edges, nan's, etc.
  - brighter-fatter correction
  - dark subtraction
  - fringe correction
  - stray light subtraction
  - flat correction
  - apply gains
    - Optional: Fringe Correction after flat
  - vignette calculation
  - attach transmission curve
  - illumination correction

**Overscan subtraction**

**Master bias subtraction  
(mean over n bias images)**

# Butler gen3

- Very useful documentation from Dominique
- Using SQLite3 for now
- PostgreSQL: account created but not yet used

1) Ingestion OK

Preparing full ingestion of Run 5 data (Slurm)

2) Calibrations added OK

3) Check (by hand) of corrected .FITS files OK

```
LSSTCam/calib
LSSTCam/calib/unbounded
LSSTCam/raw/all
u/tguillem/DM-30000/biasGen.21062022a
u/tguillem/DM-30000/biasGen.21062022a/20220621T085649Z
u/tguillem/DM-30000/biasGen.21062022a/20220624T093758Z
u/tguillem/master_dark_0
u/tguillem/master_dark_0/20220621T090619Z
u/tguillem/master_dark_0/20220621T090653Z
u/tguillem/master_dark_0/20220621T090913Z
u/tguillem/master_dark_no_bias_0
u/tguillem/master_dark_no_bias_0/20220624T090458Z
u/tguillem/master_dark_no_bias_0/20220624T092733Z
u/tguillem/master_dark_no_bias_2
u/tguillem/master_dark_no_bias_2/20220624T122005Z
u/tguillem/master_dark_no_bias_3
u/tguillem/master_dark_no_bias_3/20220624T123125Z
DatasetType('bias', {instrument, detector}, ExposureF, isCalibration=True)
DatasetType('camera', {instrument}, Camera, isCalibration=True)
```

# Setup

- lsst\_distrib/w\_2022\_19

with local packages:

ip\_isr

cp\_pipe

cp\_verify

Working area: /sps/lsst/users/tguillem/Rubin/stack/w\_2022\_19/tests\_isr

Fork of ip\_isr:

[https://github.com/tguillemLSST/ip\\_isr](https://github.com/tguillemLSST/ip_isr)

Run 5 BOT data ingested with butler gen3:

/sps/lsst/groups/FocalPlane/SLAC/run5/butler/gen3/


# Overscan corrections currently in DM

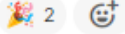
- Only use of the serial overscan region


```
class OverscanCorrectionTaskConfig(pexConfig.Config):
    """Overscan correction options.
    """
    fitType = pexConfig.ChoiceField(
        dtype=str,
        doc="The method for fitting the overscan bias level.",
        default='MEDIAN',
        allowed={
            "POLY": "Fit ordinary polynomial to the longest axis of the overscan region",
            "CHEB": "Fit Chebyshev polynomial to the longest axis of the overscan region",
            "LEG": "Fit Legendre polynomial to the longest axis of the overscan region",
            "NATURAL_SPLINE": "Fit natural spline to the longest axis of the overscan region",
            "CUBIC_SPLINE": "Fit cubic spline to the longest axis of the overscan region",
            "AKIMA_SPLINE": "Fit Akima spline to the longest axis of the overscan region",
            "MEAN": "Correct using the mean of the overscan region",
            "MEANCLIP": "Correct using a clipped mean of the overscan region",
            "MEDIAN": "Correct using the median of the overscan region",
            "MEDIAN_PER_ROW": "Correct using the median per row of the overscan region",
            "ROW_COL": "2D row and col correction from the serial and parallel overscan regions",
        },
    ),
```

➔ Tried to add  
but overscanImage  
has only serial region

Vendredi 24 juin ▾

 **Chris Waters** 2 h 21  
A quick update on the DM serial+parallel overscan correction progress:  
<https://jira.lsstcorp.org/browse/DM-33429> Barring any serious bugs, this should be looking for a reviewer tomorrow.



 **SQuaRE Bot** APPLI 2 h 21  
DM-33429: In Progress Add ability to do both serial and parallel overscan correction  
Christopher Waters | 28 janv.

➔ Will try to  
test this

# Calibration products

- Great page:  
<https://pipelines.lsst.io/v/daily/modules/lst.cp.pipe/constructing-calibrations.html>
- Construct master biases OK
- Certify calibrations OK
- Apply only overscan corrections to images (changing method) OK
- Construct master darks OK
- Construct darks OK

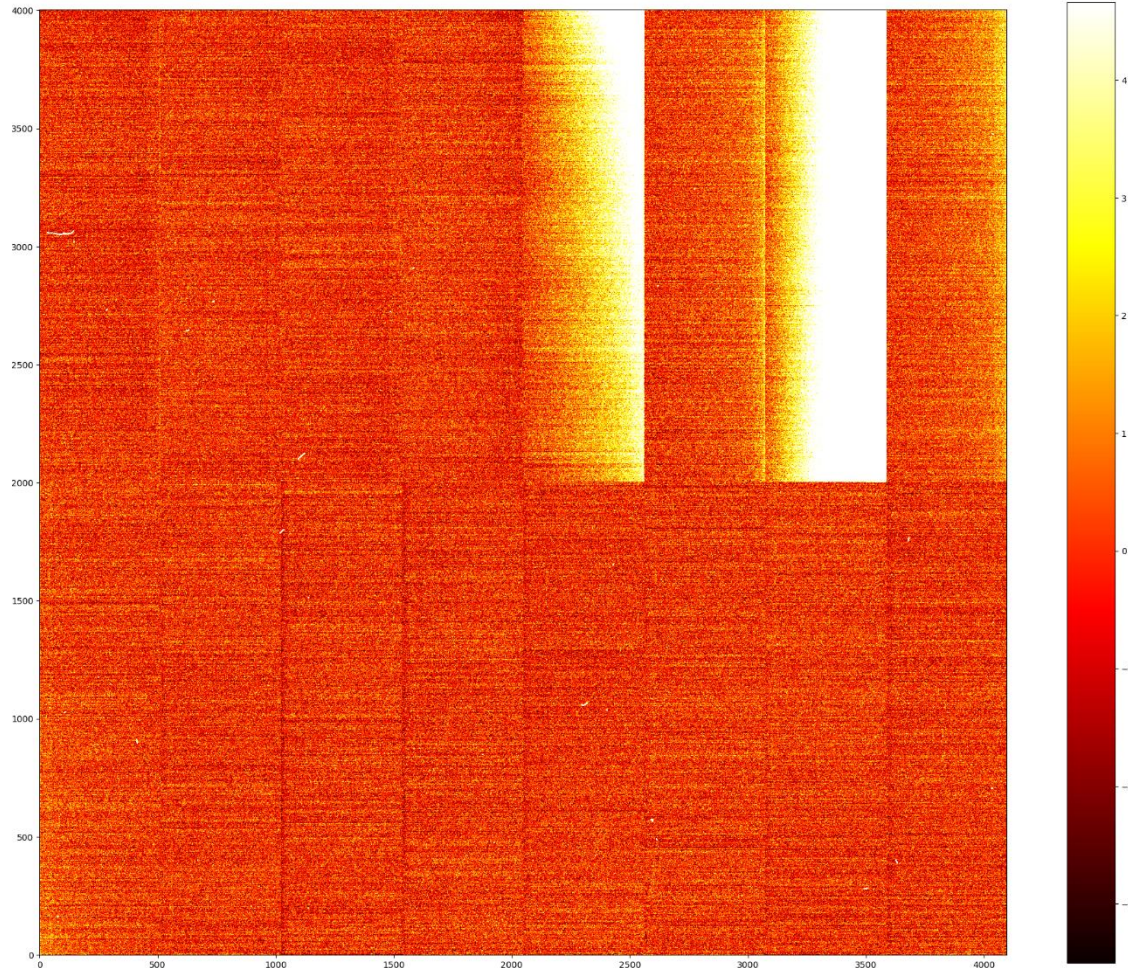
## Task example:

```
71, exposure: 3021121200163, ...}} (isrTask.py:1042) - Constructing linearizer from cameraGeom information.
exposure: 3021121200163, ...}} (isrTask.py:1454) - Converting exposure to floating point values.
exposure: 3021121200163, ...}} (isrTask.py:1521) - Assembling CCD from amplifiers.
71, exposure: 3021121200163, ...}} (isrTask.py:1525) - No WCS found in input exposure.
exposure: 3021121200163, ...}} (isrTask.py:1533) - Applying bias correction.
exposure: 3021121200163, ...}} (isrTask.py:1565) - Applying linearizer.
exposure: 3021121200163, ...}} (isrTask.py:1588) - Masking non-finite (NaN, inf) value pixels.
exposure: 3021121200163, ...}} (isrTask.py:1592) - Widening saturation trails.
exposure: 3021121200163, ...}} (isrTask.py:1736) - Set 4 BAD pixels to 0.173926.
exposure: 3021121200163, ...}} (isrTask.py:1739) - Interpolating masked pixels.
71, exposure: 3021121200163, ...}} (isrTask.py:2648) - No rough magnitude zero point defined for filter unknown.
exposure: 3021121200163, ...}} (isrTask.py:2656) - Setting rough magnitude zero point for filter unknown: 32.445378
```

# Results

Dark image (120 s) corrected via ISR with the 1-D line overscan correction

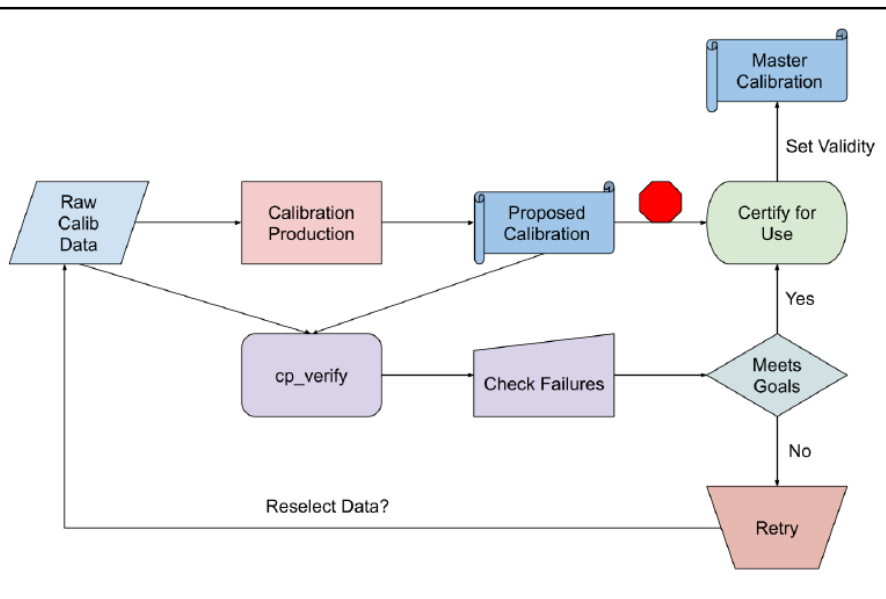
Image per amplifier : (70% percentile)  
cpDarkisr\_LSSTCam\_unknown\_MC\_C\_20211212\_000169\_R14\_S22\_u\_tguillem\_master\_dark\_no\_bias\_3\_20220624T123125Z.fits





# Calibration strategy

Document: <https://dmtn-222.lsst.io/v/u-czw-20220321/index.html>



Calibration Type	Cadence	$N_{\text{exposure}}$
Bias	Daily	15
Dark	Daily	15
Flat	Daily <sup>1</sup>	15
Defects	Weekly	Uses the bias, dark, flat exposures.
Gain	Daily	Uses the flat exposures.
PTC	As needed	N/A
Linearity	As PTC	N/A
Brighter-Fatter Kernel	As PTC	N/A
Fringes	N/A	N/A

Table 1: Recommended cadence and exposure count for daily calibration verification.

# Conclusion

- Started to do bias studies using ISR
- Involvement in ISR activities (DM-team)?  
The key place for all calibration aspects in the coming years...  
Camera-team and DM-team: interaction/collaboration not crystal clear to me.