MDIA

A Difference Imaging Tool for AstroWISE

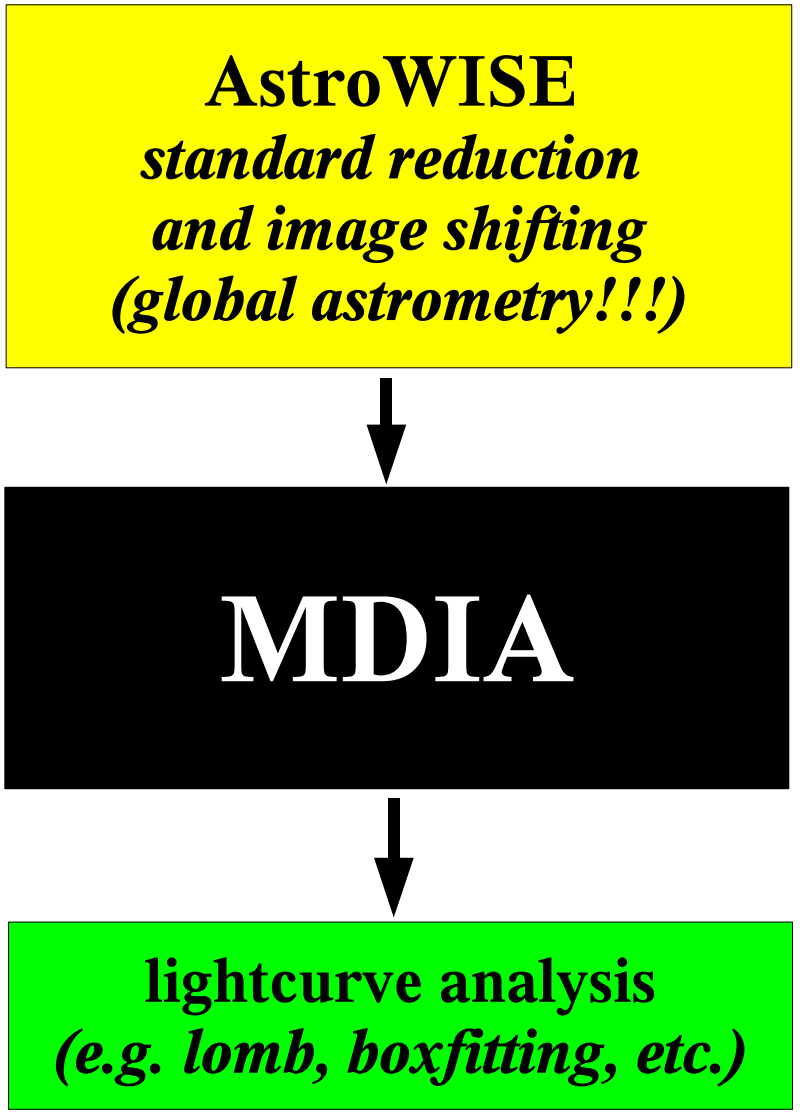
by Johannes Koppenhöfer and Arno Riffeser

AstroWISE workshop, Leiden 11/2006

MDIA - Flowchart

input:

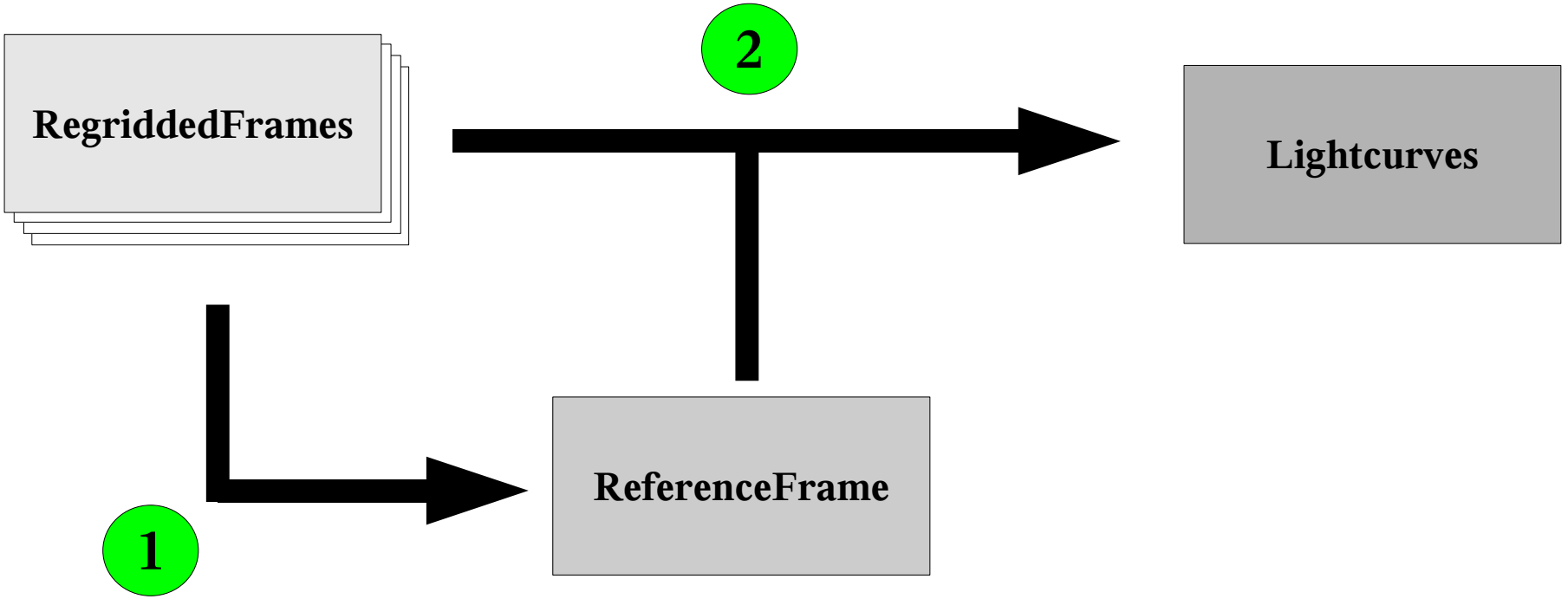
RegriddedFrames,
SourceLists,
Parameters



output:

ReferenceFrame,
Lightcurves

The MDIA Functionality



1 Creating a reference frame

- preparation: creation of error frames, expanding images
- sky-subtract best seeing image (2^{nd} order polynomial)
- photometrically align images (constant absorption, 2^{nd} order sky)
- measure PSF in all images (moffat-fit: x- & y-fwhm, angle, beta)
- replace masked pixel/regions with best fitting other image
- weighted stacking (by seeing, background noise)

class ReferenceFrame

- all properties of RegriddedBaseFrame

- ReferenceFrame.regridded_frames	input RegriddedFrames
- ReferenceFrame.process_params	process parameters
(- ReferenceFrame.sources	SourceList of all sources)
(- ReferenceFrame.variables	SourceList of all variable sources)
- ReferenceFrame.error	associated error frame

How-To create a ReferenceFrame in AstroWISE:

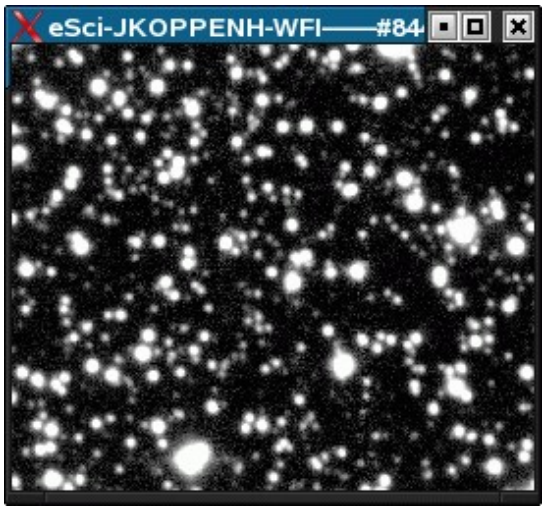
```
awe> ref = ReferenceFrame()
awe> ref.regridded_frames = my_regridded_list
awe> ref.process_params.VERBOSE = 1
awe> ref.make()
awe> ref.store()
awe> ref.commit()
```

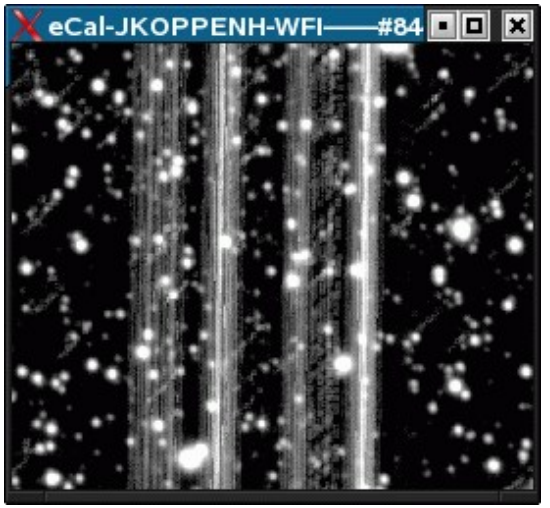
filenames: Sci-JKOPPENH-WFI-------#844---Ref---Sci-54048...
Cal-JKOPPENH-WFI-------#844---Ref---Err-54048...
Cal-JKOPPENH-WFI-------#844---Ref---Wei-54048...

The MDia error frames

created from photon and readout noise (maybe flatfield to be included)

consequent pixel by pixel error propagation in each reduction step





2 Creating lightcurves

- preparation: creation of error frames, expanding images
- photometrically align images (constant absorption, 2nd order sky)
- difference imaging in whole frame or subframes
- photometry in ReferenceFrame
- PSF- or aperture-photometry in difference images
- lightcurve creation (ascii tables)
- *optional: barycentric time correction*

class MDia

- MDia.reference_frames input ReferenceFrame

  or

- *MDia.mdia* *previous MDia run*

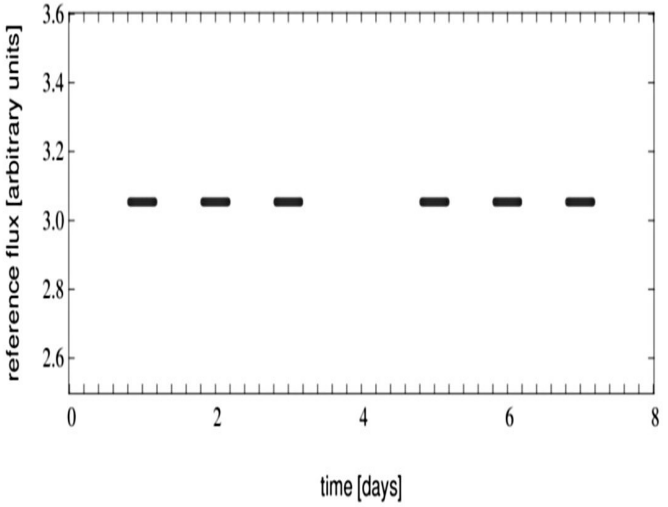
- MDia.regridded_frames input RegriddedFrames
- MDia.process_params process parameters

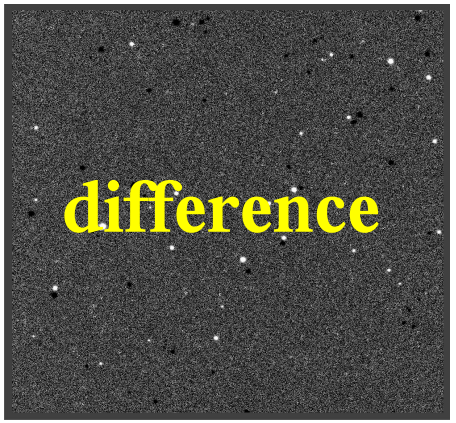
(- MDia.sources SourceList of all sources)
(- MDia.variables SourceList of all variable sources)

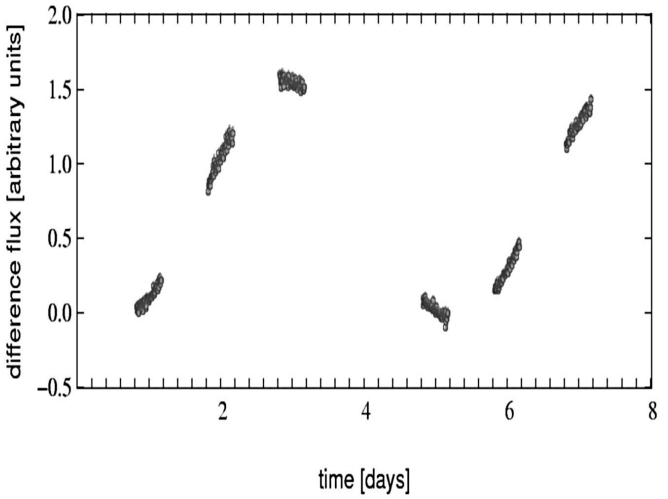
How-To create a Lightcurves in AstroWISE:

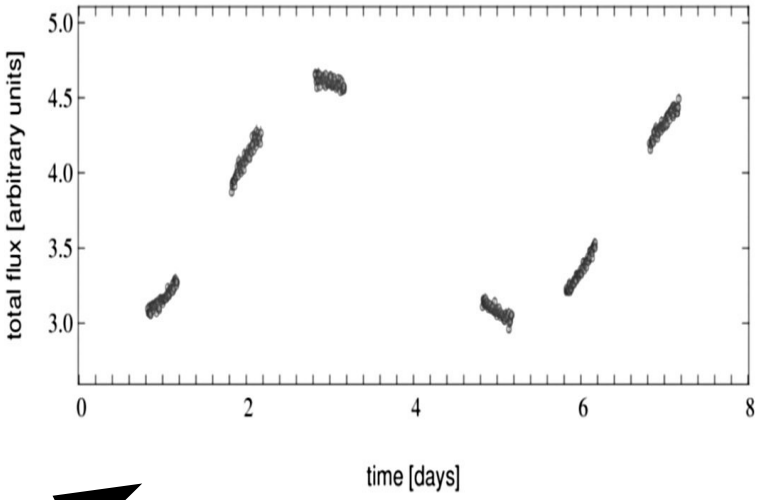
awe> my_mdia = MDia()
awe> my_mdia.regridded_frames = my_regridded_list
awe> my_mdia.process_params.VERBOSE = 1
awe> my_mdia.make()
awe> my_mdia.store()
awe> my_mdia.commit()

The Amplitude of the Variation:

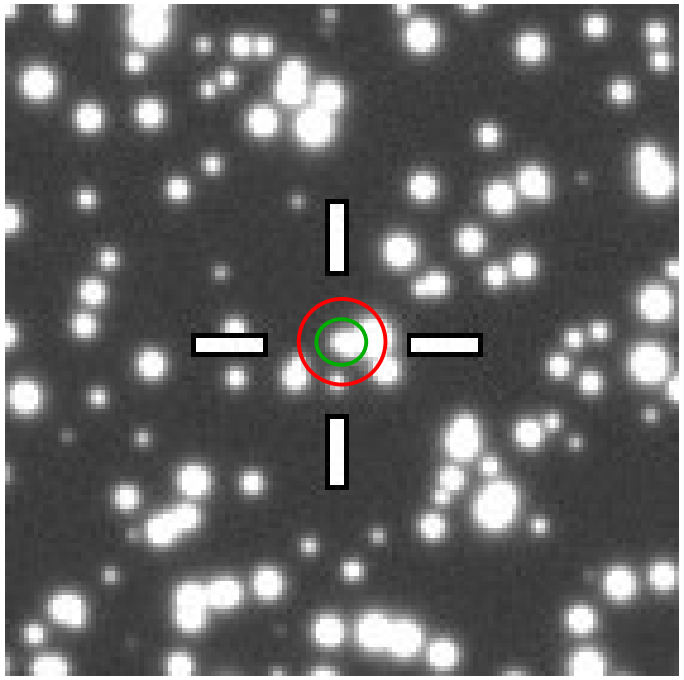


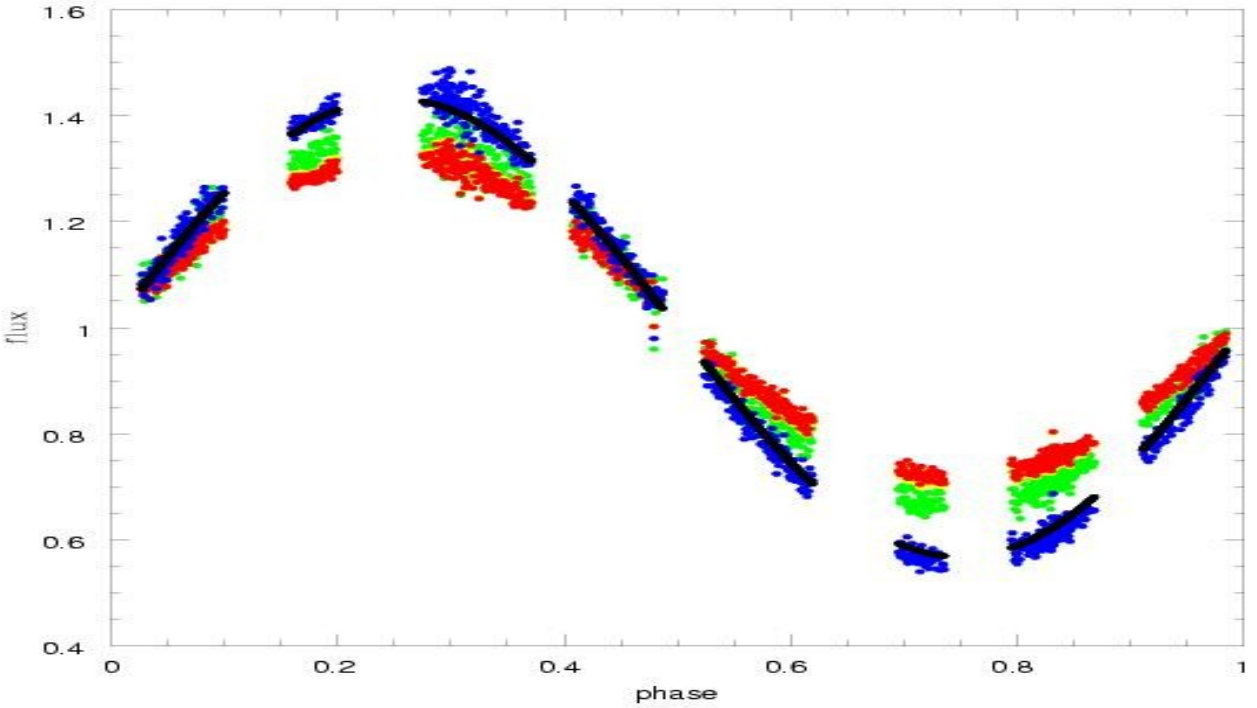






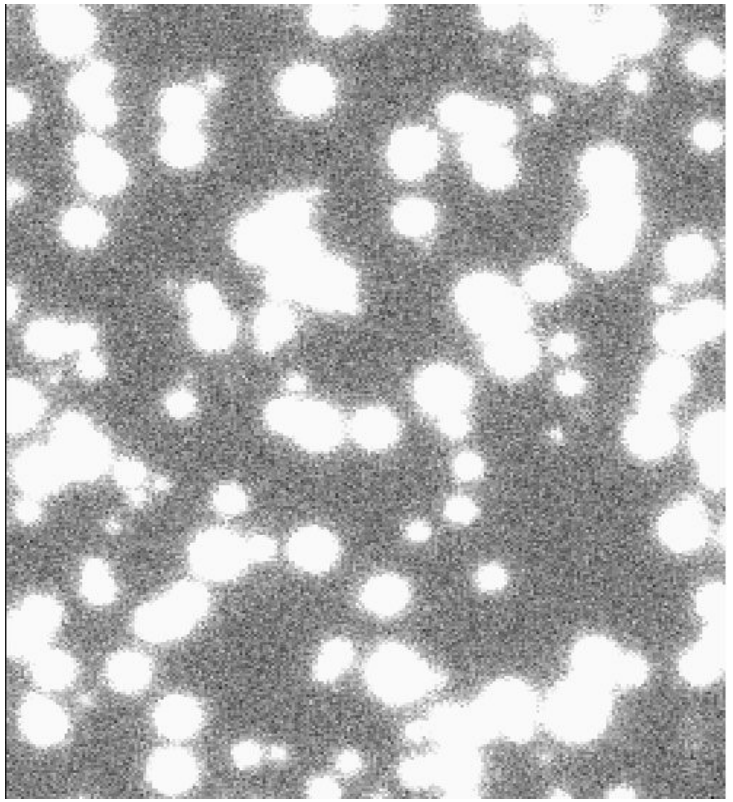
The Fitting-Radius Dependency

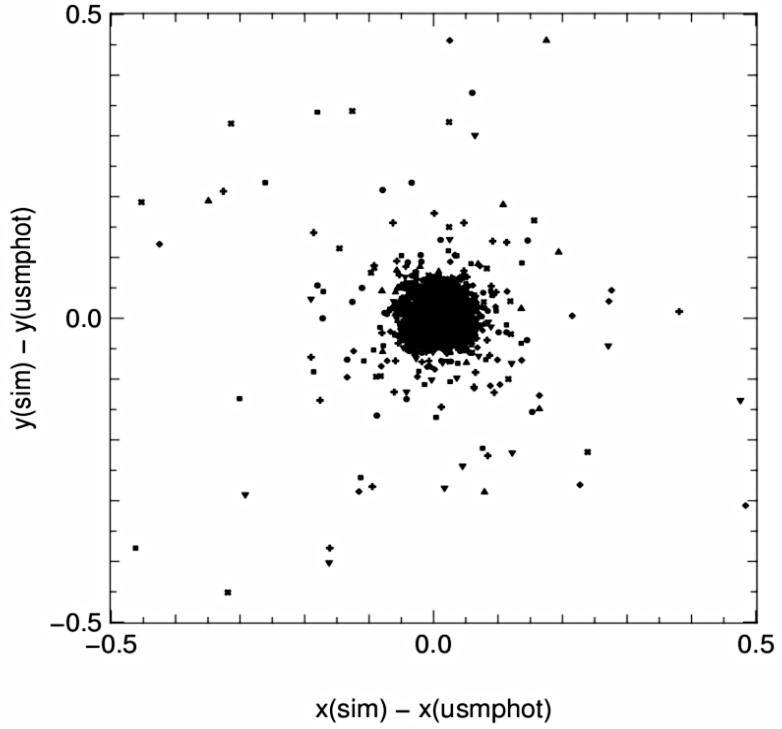


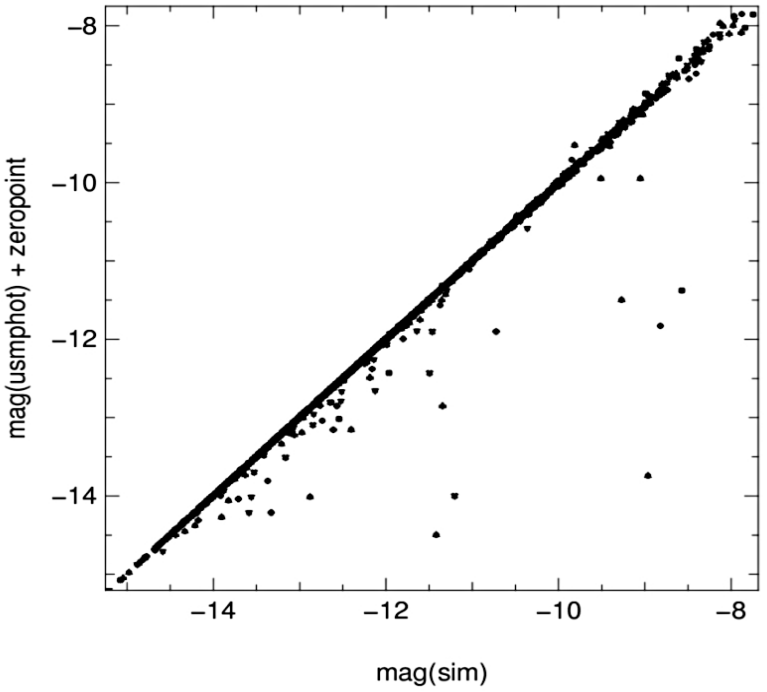


true amplitude = 43.2%
small fitting-radius = 34.6 (80%)
bigger fitting-radius = 29.8 (69%)
USMPHOT = 44.5 (103%)

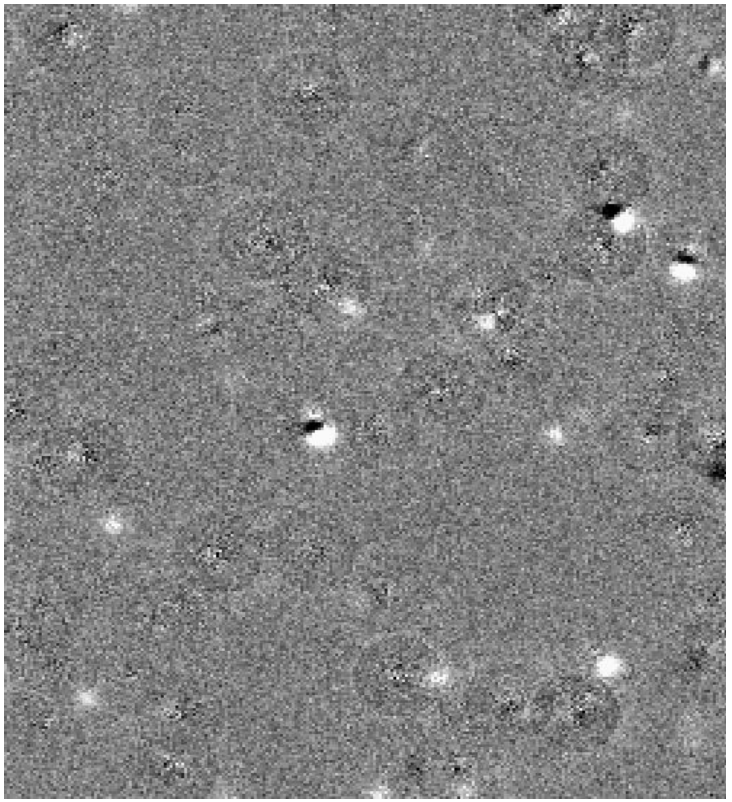
USMPHOT:

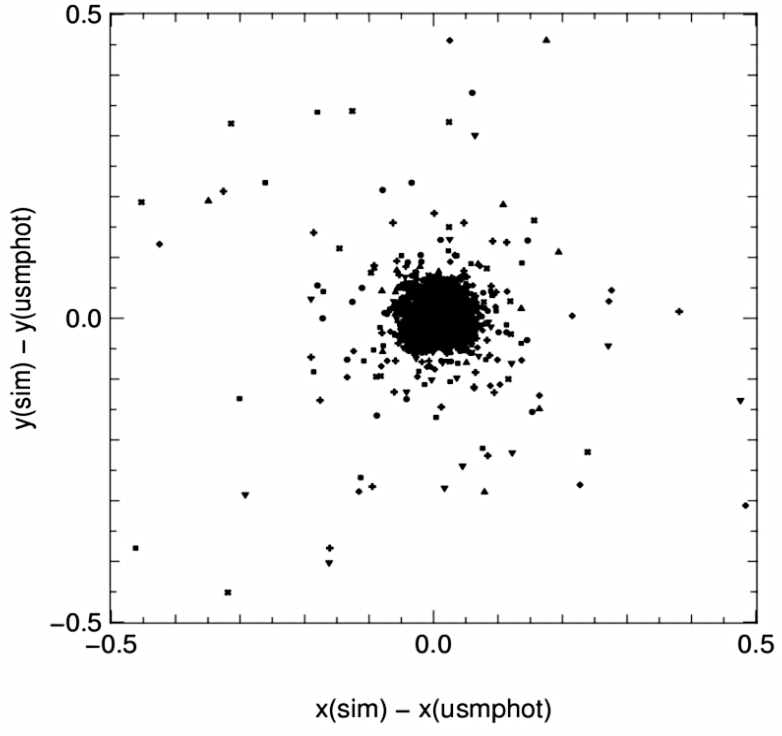


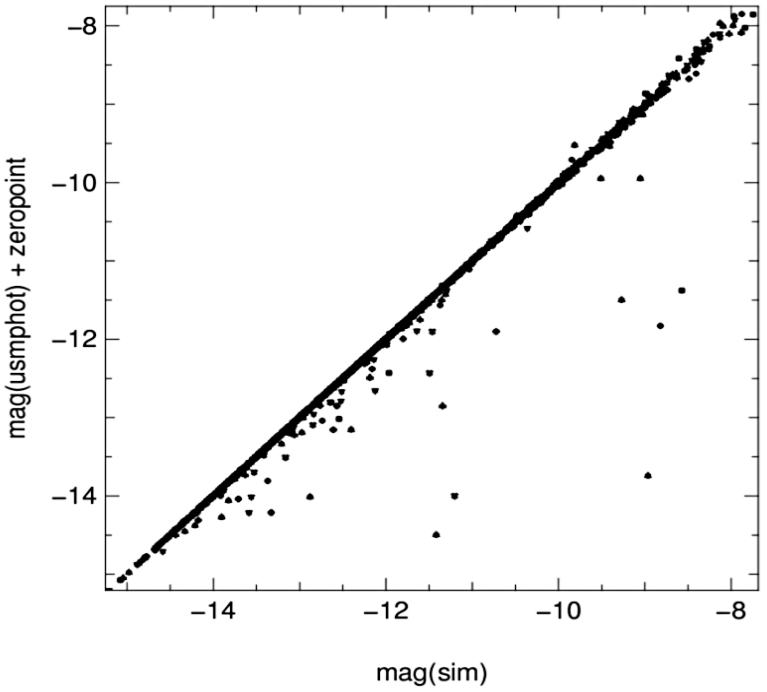

0.5
0.0
−0.5
−0.5
0.0
0.5
y(sim) – y(usmphot)
x(sim) – x(usmphot)


−8
−10
−12
−14
−14
−12
−10
−8
mag(usmphot) + zeropoint
mag(sim)

USMPHOT:




0.5
0.0
−0.5
−0.5
0.0
0.5
y(sim) – y(usmphot)
x(sim) – x(usmphot)


−8
−10
−12
−14
−14
−12
−10
−8
mag(usmphot) + zeropoint
mag(sim)

...to be done:

- decide about DB object design (splinter meeting)
- integrate USMPHOT (psfex?)
- implement dpu-interface
- add barycentric time correction
- provide full documentation

Photometric Alignment

$$f' = \text{absorption} \cdot f + \text{sky}$$

$$\text{sky} = a \cdot x + b \cdot y + c$$

$$\text{absorption} = \text{const.}$$

back