

Astro-WISE  
and  
pre-calibrated data  
and  
external catalogs

For the Astro-WISE consortium

Gijs Verdoes Kleijn

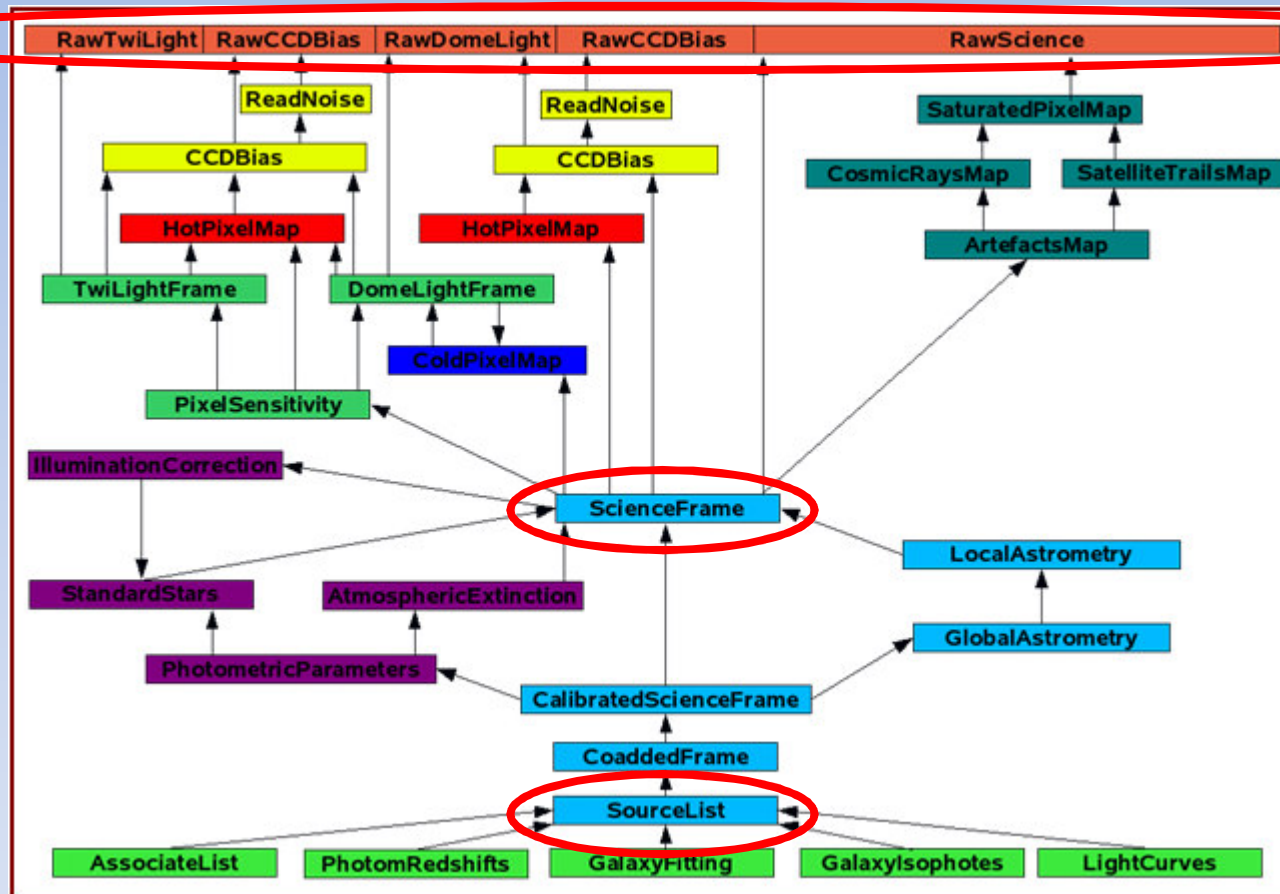
John McFarland

# Why use Astro-WISE?

## Large surveys, Small projects

- **New paradigm : full data lineage (L+s)**
  - Inspect & verify back to raw data
  - automatic workflow
- **Adaptable data reduction pipeline available (L+s)**
- **Share with your team (L+s)**
  - Workload: data reduction, quality flagging
- **Release data from end-product to raw data**
  - share with team+flag=share with world
- **Benefit from other's existing work (L+s)**
  - Calibration data
  - Science data,
  - Catalogs (cross-correlated, also SDSS, 2Mass, UKIDSS, etc)
- **Computing power, storage space (L)**

# Where enter Astro-WISE system?



# Getting in: ingestion

- HeaderTranslator: entry point of an instrument into Astro-WISE
  - makes header keywords uniform and provides information about instrument, detectors, and filters
- For pre-calibrated data:
  - Need to construct calibration objects
    - AstrometricParameters
    - PhotometricParameters
    - WeightFrame

# Using Astro-WISE

1. It's about more than iraf/midas/idl/gipsy
  - Sharing, data-management, survey management
  - Not only fits files and headers
2. You need to make paradigm shift:
  - As clear as possible: documentation
  - As intuitive as possible
3. You need to know the things (without which this shift is not possible)
  - python
  - Object oriented+database approach to scientific analysis and data calibration and reduction
4. 2 & 3 often require close user-developer interaction

# Roadmap for ingesting an instrument

- Email John
- You get a HeaderTranslator template
- User edits:
  - Filter information
  - Chip information
  - Instrument information
- User checks HeaderTranslator on instrument data
- John verifies
- User does test ingestion (MyDB)

```
awe>
awe> RawScienceFrame.get_persistent_properties()
['AIRMEND', 'AIRMSTRT', 'DATE', 'DATE_OBS', 'EXPTIME', 'LST', 'MJD_OBS', 'NAXIS1', 'NAXIS2', 'OBJECT', 'OB
SERVER', 'OVSCX', 'OVSCY', 'PRSCX', 'PRSCY', 'UTC', 'astrom', 'chip', 'extension', 'filename', 'filter', '
globalname', 'imstat', 'instrument', 'is_valid', 'object_id', 'observing_block', 'overscan_x_stat', 'overs
can_y_stat', 'prescan_x_stat', 'prescan_y_stat', 'process_status', 'quality_flags', 'raw_fits_data', 'temp
late']
awe>
awe>
awe>
awe>
awe> ReducedScienceFrame.get_persistent_properties()
['AIRMEND', 'AIRMSTRT', 'DATE', 'DATE_OBS', 'EXPTIME', 'NAXIS1', 'NAXIS2', 'OBJECT', 'astrom', 'bias', 'ch
ip', 'cold', 'creation_date', 'filename', 'filter', 'flat', 'fringe', 'globalname', 'hot', 'illumination',
'imstat', 'instrument', 'is_valid', 'object_id', 'observing_block', 'process_params', 'process_status', '
psf_radius', 'quality_flags', 'raw', 'scale_factor', 'template', 'weight']
awe>
awe> PhotometricParameters.get_persistent_properties()
['chip', 'creation_date', 'extinct', 'filename', 'filter', 'globalname', 'instrument', 'is_valid', 'mag_id
', 'number_of_sources', 'object_id', 'observing_block', 'photcat', 'process_params', 'process_status', 'qu
ality_flags', 'template', 'timestamp_end', 'timestamp_start', 'zeropnt']
awe>
awe>
```

- Show Header Translator
  - MPG-ESO/WFI
  - HST/ACS