rPICARD - A CASA-based Calibration and Imaging Pipeline for VLBI Data

14th EVN Symposium & Users Meeting
Granada, October 11 2018

Michael Janssen
BlackHoleCam* PhD student
Supervisors: Heino Falcke & Ciriaco Goddi,
Radboud University Nijmegen

*BHC is an ERC-funded project and partner of the Event Horizon Telescope Consortium

rPICARD - Radboud PIpeline for the Calibration of high Angular Resolution Data

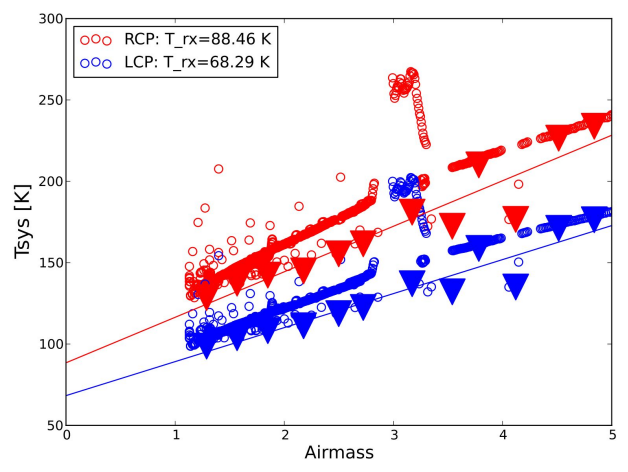
- AIPS
  - Official support discontinued.
  - Steep learning curve for new python generation of astronomers.
  - Limited support for batch processing.
- CASA
  - Secure development future, ipython interface, built for batch processing → pipeline.
  - Widely used and well established (ALMA, VLA, …).
  - MPI scaling to deal with increasing data volumes in the future.
  - VLBI upgrade from JIVE+BlacHoleCam (Ilse's talk)

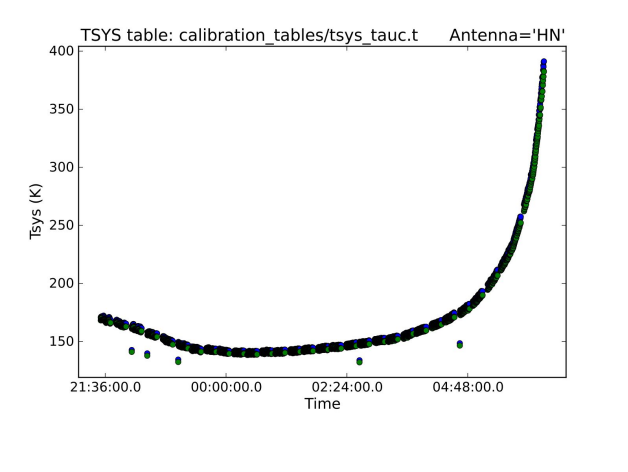
- rPICARD: CASA-based VLBI calibration and imaging pipeline
  - Highly configurable & self-tuning parameters (e.g., fringe-fit solution interval based on SNR).
  - Verbose diagnostics (→ plots), easy to control and re-run + intervene semi-interactively.
  - Used for EHT data processing. Work with Ilse van Bemmel, Kazi Rygl, Elisabetta Liuzzo.
  - And for other arrays: GMVA, VLBA, EVN, … can work with any fits-idi files or measurement sets.
  - MPI speedup (fringe-fit scans in parallel).
  - Documentation/cookbook (40 pages).
  - Open source: https://bitbucket.org/M_Janssen/picard.
  - Science reproducibility.

rPICARD calibration

All plots shown are automatically generated by the pipeline

ANTAB Amplitude calibration and opacity fit

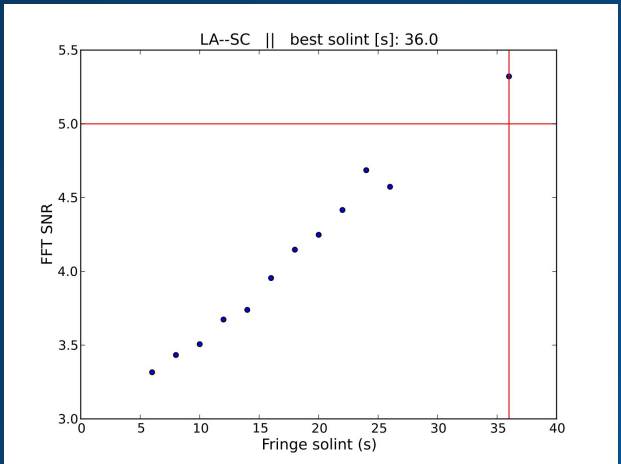


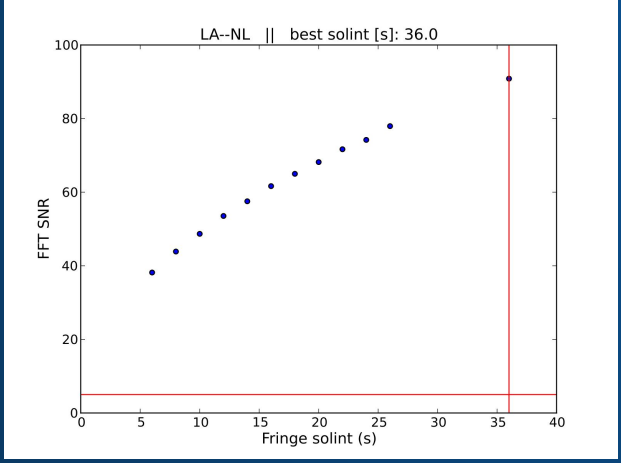


- $T_{sys} \sim T_{rx} + (1 - e^{-\tau})T_{atm}$
- $T^{*}_{sys} = T_{sys} * e^{\tau}$
- Find T_{atm} with Pardo et al. (2001) atmospheric code.
- Find T_{rx} by extrapolating T_{sys} to zero airmass.

7mm VLBA data of M87.
Project code: BW0106.

Fringe-fit solution intervals tuned by SNR



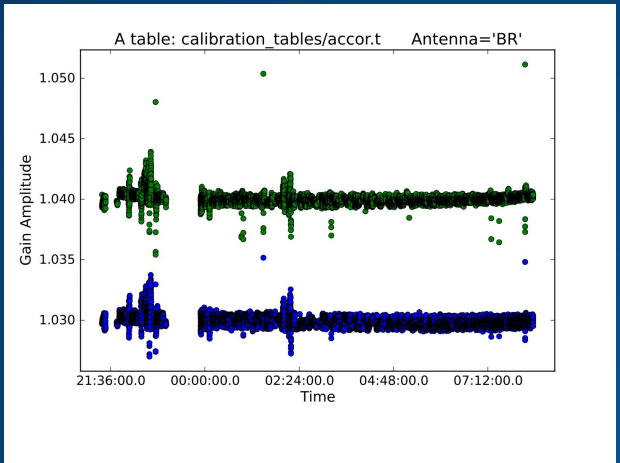


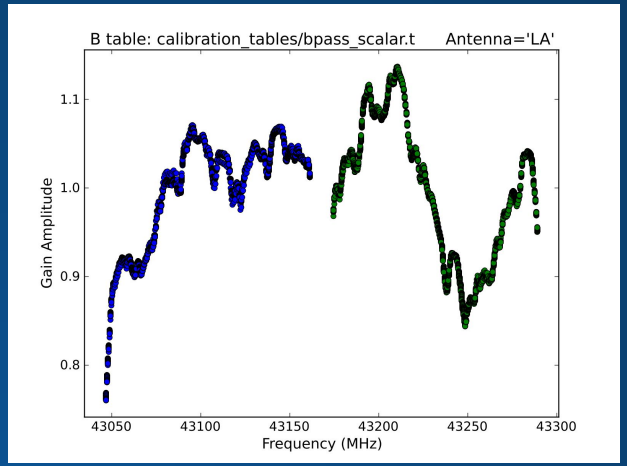
7mm VLBA data of M87 from June 2013.
Project code: BW0106.

- CASA *fringefit* is copy from AIPS FRING task: Schwab and Cotton (1983).
  → FFT with SNR cut for initial guesses and station-based least-squares refinement.
- Skip least-squares for quick solution interval parameter search (smallest solint that yields detections on all possible baselines) per scan.
- Can have different solution intervals per station.

Calibration Solution Examples (7mm VLBA)

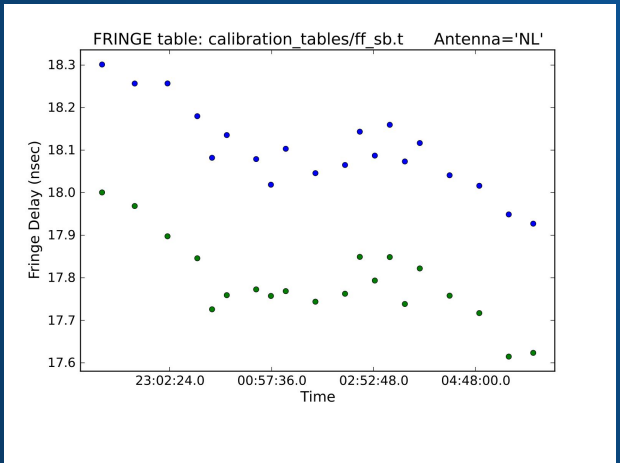
Sampler thresholds correction

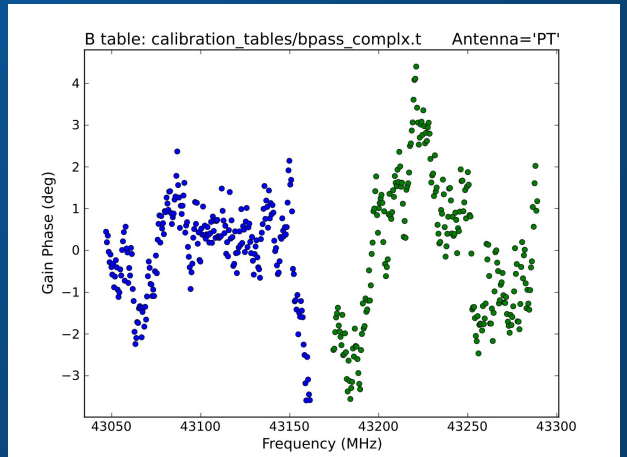




Scalar bandpass

Manual phase calibration

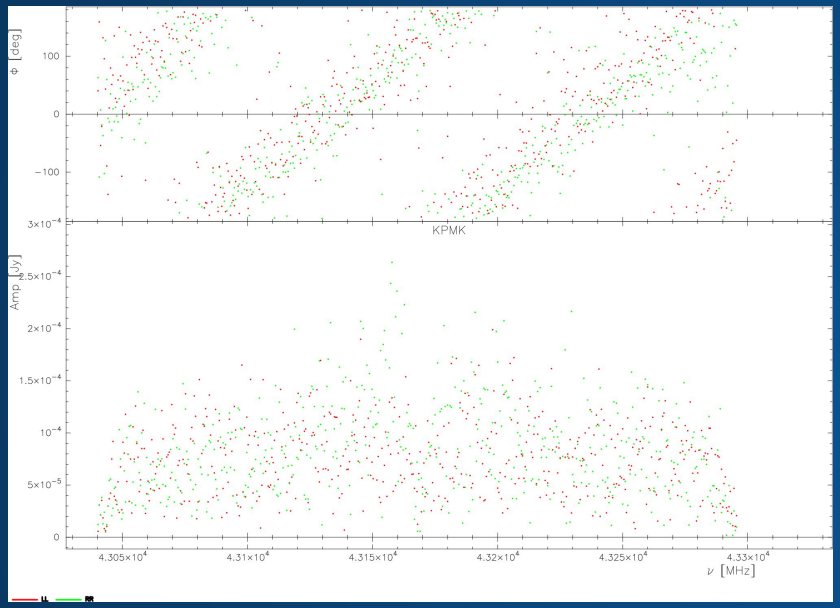




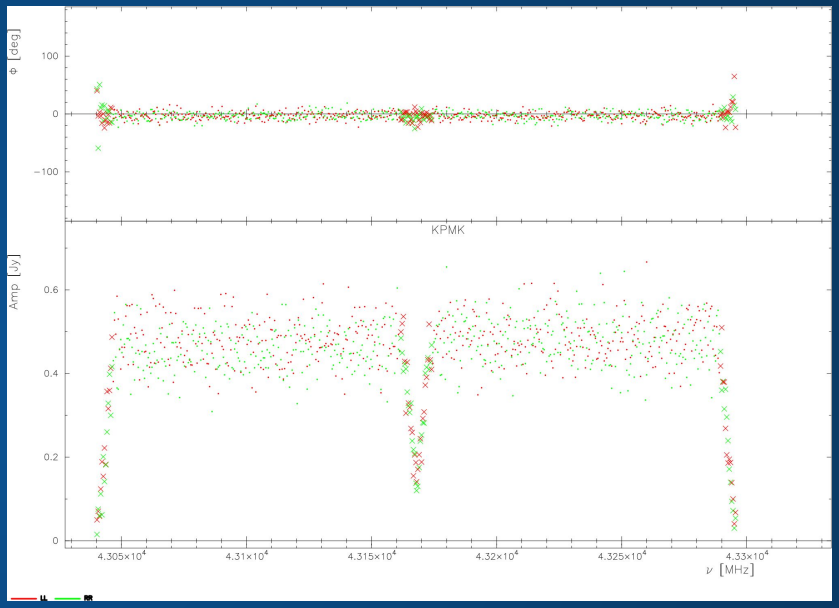
Complex bandpass

7mm VLBA data of M87 from June 2013. Project code: BW0106. Two spectral windows (IFs).

Calibration Solutions applied (7mm VLBA)



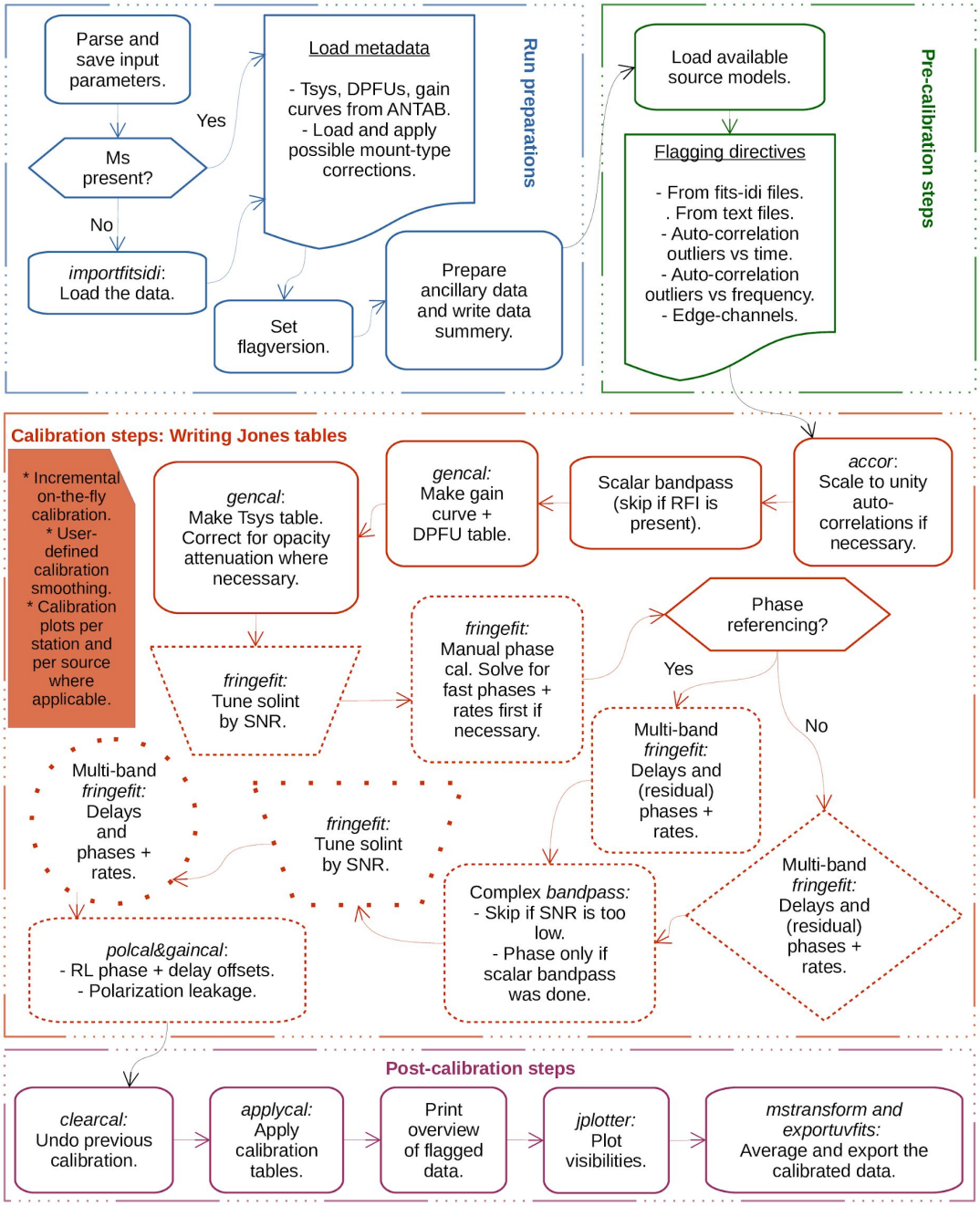
Uncalibrated.



Calibrated. Edge channels flagged (crosses).

Plots made with Harro Verkouter's *jplotter*.

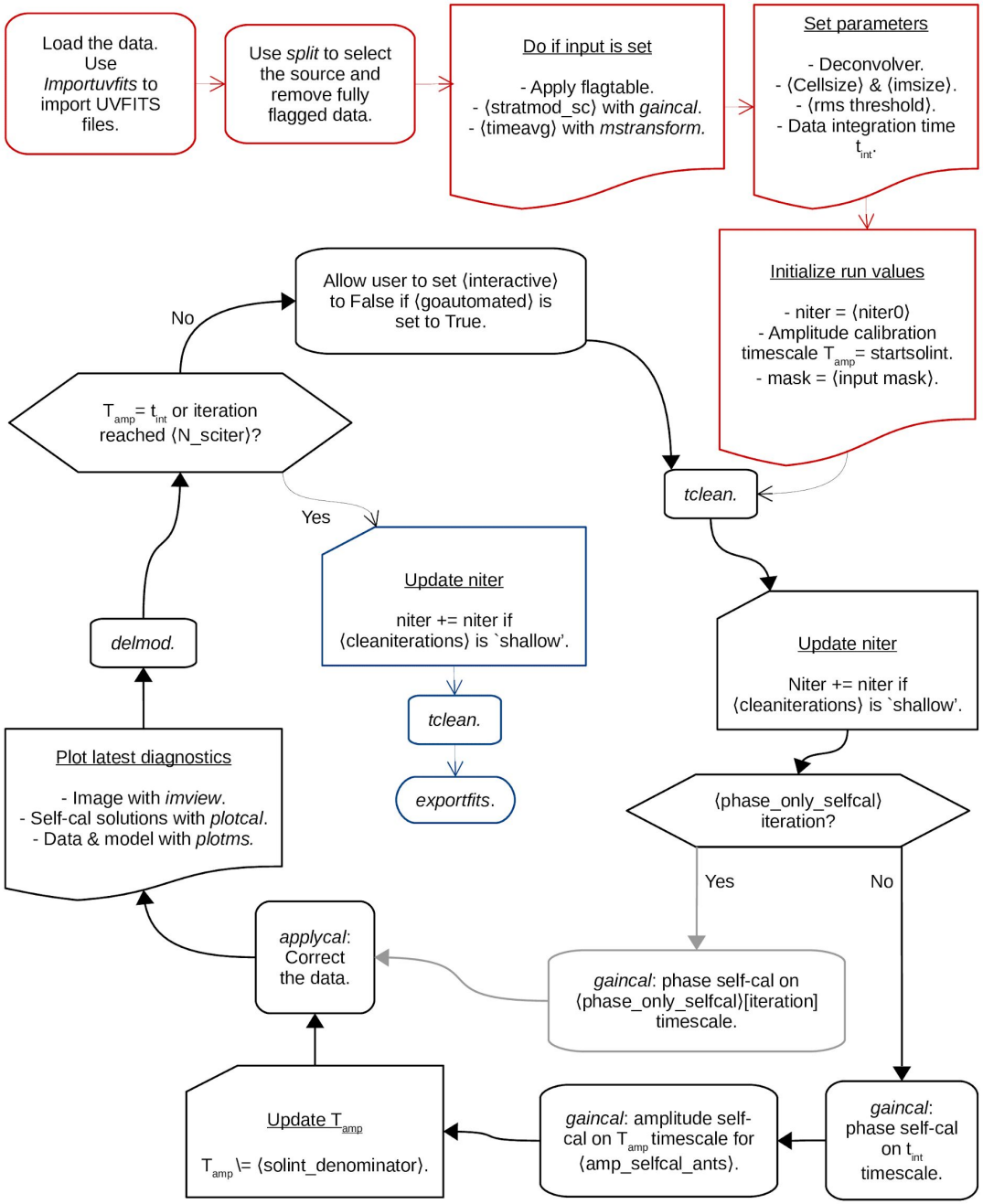
rPICARD Calibration flowchart



- **Solid border:**
  - **All sources used.**
- **Dashed border:**
  - **Calibrators used.**
- **Dotted border:**
  - **Science targets used.**
- **Rectangular boxes:**
  - **Applied to all sources.**
- **Diamonds:**
  - **Applied to calibrators.**
- **Circles:**
  - **Applied to science targets.**

rPICARD imaging

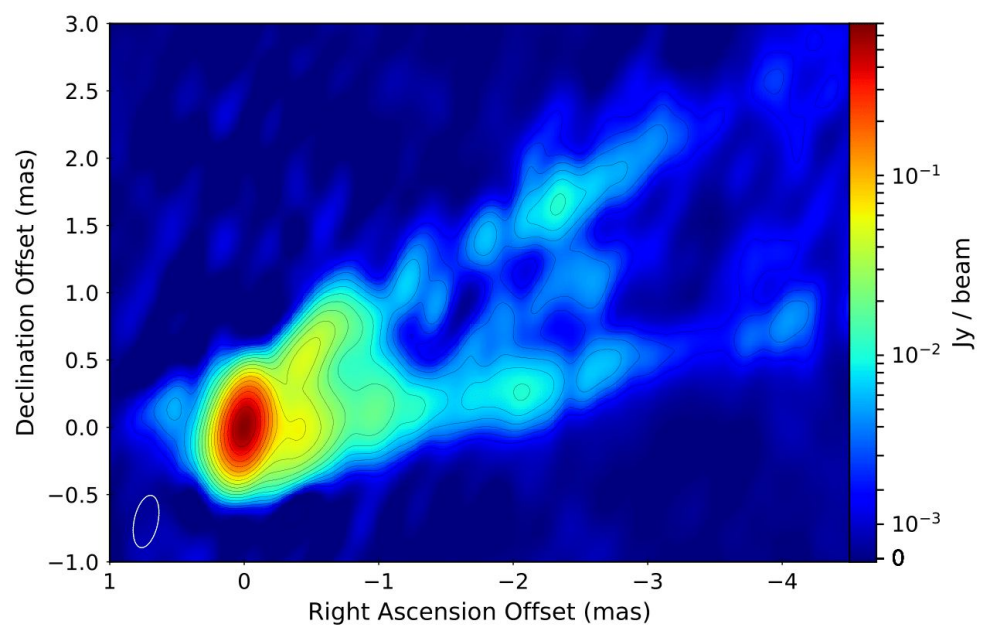
Uses CASA tclean



rPICARD Imager flowchart

- **Loops of multi-scale tclean and self-calibration.**
  - **Phases: accumulation period.**
  - **Amplitudes: start at hours timescale and lower by factor of 2 in each iteration.**
- **Stop after set number of maximum iterations or when calibrating amplitudes on accumulation period timescales.**
- **Promising work from Jose Luis: Using auto-masking for automated imaging without user bias.**

rPICARD image of 7mm VLBA M87 Data



7mm VLBA data of M87 from June 2013.
Project code: BW0106.

- Calibrated and imaged with CASA-based rPICARD pipeline.
- Results agree with Walker et al. (2018):
  - Weak counterjet.
  - Edge-brightening.
  - Large initial opening angle.
  - Re-collimation of upper arm.

Summary

Thank you!

- CASA is ready for VLBI.

- rPICARD, a first general purpose CASA-based calibration+imaging pipeline is available (Janssen et al., in prep.)
  - Promotes reproducibility of scientific results. Pipeline is open source:
    $ git pull https://bitbucket.org/M_Janssen/picard
  - Verbose diagnostics, self-tuning default parameters, highly configurable, interactive mode, option to quickly re-run steps.
  - MPI scalable.
  - Well documented.
  - Used for EHT data processing (cross-validated with AIPS and HOPS) and successfully applied to GMVA, VLBA, EVN, and synthetic data as well. Modularity → easy to add other arrays.
  - Phase referencing and polarization calibration supported (for leakage calibration a sufficiently compact calibrator is needed).
  - Future features
    - Spectral line calibration features (delay solutions from continuum sourced and rate solutions from the bright line).
    - LPCAL-like task to solve for leakage from calibrators with extended polarization structure.

- Coming soon: MeqSilhouette (Blecher et al., 2017) + rPICARD realistic synthetic data generation pipeline (Heino's talk)

```
michael@mjpc:~/JeanLuc/Picard$ ./setup.py -p ~/Software/

*** This script will link your CASA installation to the pipeline. ***
*** It can always be executed again.                               ***


First we will try to find a suitable CASA installation.
The required features for this pipeline are:
['mpi', 'fringefit.py', 'accor.py']

It is highly recommended to use the exact same CASA version as advertised in the README.md file


Looking for CASA executables in /home/michael/Software/

I have found more than one CASA installation.
Please enter the number for the installation you want to use:

0 for /home/michael/Software/CASA/casa-release-4.7.2-el7/bin/casa
      Has mpi:          True
      Has fringefit.py: False
      Has accor.py:     False


1 for /home/michael/Software/CASA/casa-release-5.3.0-143.el7/bin/casa
      Has mpi:          True
      Has fringefit.py: True
      Has accor.py:     True


2 for /home/michael/Software/CASA_builds_from_JIVE/casa-feature-CAS-10684-22.el7/bin/casa
      Has mpi:          True
      Has fringefit.py: True
      Has accor.py:     True


3 for /home/michael/Software/CASA_builds_from_JIVE/casa-feature-CAS-10684-24.el7/bin/casa
      Has mpi:          True
      Has fringefit.py: True
      Has accor.py:     True


4 for /home/michael/Software/old_CASA/casa-release-4.7.0-1-el7/bin/casa
      Has mpi:          True
      Has fringefit.py: False
      Has accor.py:     False


Enter the number corresponding to the installation you want to use and press Enter
 >3

Found
 /home/michael/Software/CASA_builds_from_JIVE/casa-feature-CAS-10684-24.el7/bin/casa
 as your CASA executable.
 Checking this CASA version:
```

Run setup script to link CASA installation to rPICARD.

```
Found
 /home/michael/Software/CASA_builds_from_JIVE/casa-feature-CAS-10684-24.el7/bin/casa
 as your CASA executable.
 Checking this CASA version:

      Has mpi:          True
      Has fringefit.py: True
      Has accor.py:     True

 Press Enter and I will use the absolute path to this executable for picard.sh.
 Write anything else (and then press Enter) to abort.
 >

Writing the CASA executable path to a <your_casapath.txt> file, which will be used by picard.sh.


Making picard.sh executable.


Editing the input/mpi_host_file using the determined name of this computer (mjpc)
and 4 cores. Change this setup manually if desired.

I could put some default values for array.inp depending on which array you inted to use.

  0 for VLBAlo (for low frequencies)

  1 for EHT

  2 for VLBAhi (for high frequencies)

  3 for EVN

  4 for GMVA
Press enter without entering anything else to continue without altering your array.inp file.
Else, enter the number corresponding to the array you want to use and press Enter
 >3


The pipeline should be ready to run now.
If there are issues with mpicasa contact M.Janssen@astro.ru.nl
or look at https://casa.nrao.edu/casadocs/@@search?SearchableText=mpi


If you want to be able to run the pipeline from everywhere,
then you should add the following line to your .bashrc folder:
export PATH=$PATH:/home/michael/JeanLuc/Picard/picard/


Remember set some input parameters in the beginning, before running the pipeline.
At least edit input/observation.inp and input/array.inp


Please read documentation/picard_documentation.pdf and follow the Quick Start Guide chapter to get started.
michael@mjpc:~/JeanLuc/Picard$ cp -r picard/input/ ../testrun/input
```

The setup script can also prepare a default set of input files for different arrays.

The next step is to copy your input files to the working directory.

```
#Can also be set to '$pwd', which will be expanded to the parent directory of the input folder.
workdir = $pwd

##
#  ** If you make changes to any of the sources listed below, you will have to       **
#  ** pass the -m command line argument to picard.sh to re-determine the metadata!   **
##

#Name of the science target(s) of the experiment.
#Write a comma separated list if there are more then one.
#Science targets are typically weak sources which cannot be used for calibration tasks.
#If all observed targets can be set as some type of calibrator below, set science_target = None.
science_target = SGR_A

### Lists for the different types of calibrators.                              ###
### Write a comma separated lists if there are more than one.                  ###
### The same source can of course occur multiple times                         ###
###   (same calibrator for manual phase calibration and bandpass for example). ###
### Put None if a certain calibrator is not needed                             ###
###   (e.g., no phase referencing or no polarization experiment).              ###

#Bright calibrators for manual phase calibration.
calibrators_instrphase = 3C279, NRAO530

#Bright calibrators for complex bandpass calibration.
#If bandtype_cmplx_bandpass = 'BPOLY' is set in array.inp, only a single source can be set here,
#since POLY bandpasses cannot be averaged.
calibrators_bandpass = 3C279

#Bright calibrators for rl delay and phase offsets.
#If set to None, the rl delay calibration will be skipped.
calibrators_rldly = 3C279

#Calibrators for D-term calibration.
#Must be polarized and observed over a wide range of parallactic angles for every antenna.
#If set to None, the D-term calibration will be skipped.
calibrators_dterms = 3C279, NRAO530

#Phase-referencing sources.
#If set, the phase-referencing mode is activated.
#For a list of comma separated science_targets, use a corresponding csv list here.
#Example: science_target = s1, s2, s3 and calibrators_phaseref = p1, p2 means that
#p1 is used as phaseref source for s1, p2 is used for s2, and no phase-referencing is
#done for s3.
calibrators_phaseref = None

#Whether or not to also fringe-fit the science targets themselves in phase-referencing mode.
#Set to False for very weak science targets, then only the fringe solutions from
#calibrators_phaseref are applied to the science targets.
#Set to True if the science targets are strong enough for a residual fringe-fit. In that case,
#after a transfer of fringe solutions from calibrators_phaseref, the science targets themselves
#are fringe-fitted as well, to take out residual phase/delays/rates.
#Generally, this should be False for astrometry. It should be True when the calibrator is far away
#from the science target and/or at higher frequency observations.
phaseref_ff_science = False
```

Typically, only science targets and calibrators have to be specified.

```
3C84.smodel  example.antab  example.flag  input  linkto_example_EVN.IDI1
michael@mjpc:~/JeanLuc/testrun$ picard.sh -p

========================================
The start-up time of CASA may vary
depending on whether the shared libraries
are cached or not.
========================================


========================================
The start-up time of CASA may vary
depending on whether the shared libraries
are cached or not.
========================================


========================================
The start-up time of CASA may vary
depending on whether the shared libraries
are cached or not.
========================================


========================================
The start-up time of CASA may vary
depending on whether the shared libraries
are cached or not.
========================================


========================================
The start-up time of CASA may vary
depending on whether the shared libraries
are cached or not.
========================================

IPython 5.1.0 -- An enhanced Interactive Python.

IPython 5.1.0 -- An enhanced Interactive Python.

IPython 5.1.0 -- An enhanced Interactive Python.

IPython 5.1.0 -- An enhanced Interactive Python.

IPython 5.1.0 -- An enhanced Interactive Python.

CASA 5.3.0-136   -- Common Astronomy Software Applications

CASA 5.3.0-136   -- Common Astronomy Software Applications

CASA 5.3.0-136   -- Common Astronomy Software Applications

CASA 5.3.0-136   -- Common Astronomy Software Applications

CASA 5.3.0-136   -- Common Astronomy Software Applications
```

[Links to] files used in working directory (loaded automatically):

- **Input folder**
- **Metadata**
- **Visibility data**

MPI: start CASA N times

```
The pipeline will execute the following steps for the
EVN array in the given order:
   a : load models of observed sources (if present)
   b : use online flags from idi files (if present)
   c : use flags from metadata (if present)
   d : flag based on outlier detection from auto-correlations vs time
   e : flag based on outlier detection from auto-correlations vs frequency
   f : flag edge channels
   0 : task_tsys
   1 : task_scalar_bandpass
   2 : task_gaincurve
   3 : task_fringefit_solint_cal
   4 : task_fringefit_single
   5 : task_fringefit_multi_cal
   6 : task_complex_bandpass
   7 : task_fringefit_solint_sci
   8 : task_fringefit_multi_sci
   9 : task_rldelay
  10 : task_rlphase
  11 : task_dterms
   g : clear the calibrated data column of the MS from previous applycal runs
   h : apply all existing tables from all_calibration_steps
   i : print overview of flagged data (can be slow)
   j : make diagnostic plots of calibrated visibilities for selected baselines
   k : average and export the calibrated data
Can use quickmode [-q] to execute only a subset of these steps.


         -- Executing step a --

Loading model data for the observed sources...
  Using
    /home/michael/JeanLuc/testrun/input/../3C84.smodel
  as model for 3C84
Done

         -- Executing step b --

Getting flags from fits-idi files...
  No FG table extension found in
    /home/michael/JeanLuc/testrun/input/../linkto_example_EVN.IDI1
  Continuing without generating a flag file.
Done

         -- Executing step c --

Looking for metadata files with flagging information...
  Found the following files with correlator/online flags:
    [/home/michael/JeanLuc/testrun/input/../example.flag]
  Applying flags from metadata files.
  Done applying metadata flags.
Done

         -- Executing step d --

  Skip flagging based on autocorrelations vs time because flag_autocorr_vs_time is not set.
```

rPICARD will print the steps executed for the array to be calibrated (labeled for quick re-runs).

And the steps are executed.