



# The IDIA processMeerKAT pipeline: *Fast CASA processing on a HPC cluster*

Dr Jordan Collier

ilifu Support Astronomer, IDIA, Department of Astronomy, University of Cape Town  
Adjunct Fellow, Western Sydney University

Bradley Frank, Srikrishna Sekhar, Russ Taylor, Joe Bochenek



**UNIVERSITY OF CAPE TOWN**  
IYUNIVESITHI YASEKAPA • UNIVERSITEIT VAN KAAPSTAD



Inter-University Institute  
for Data Intensive Astronomy

**WESTERN SYDNEY**  
UNIVERSITY





# MeerKAT



**UNIVERSITY OF CAPE TOWN**

IYUNIVESITHI YASEKAPA • UNIVERSITEIT VAN KAAPSTAD

**WESTERN SYDNEY  
UNIVERSITY**



Jordan Collier | 18 December 2019 | Socorro

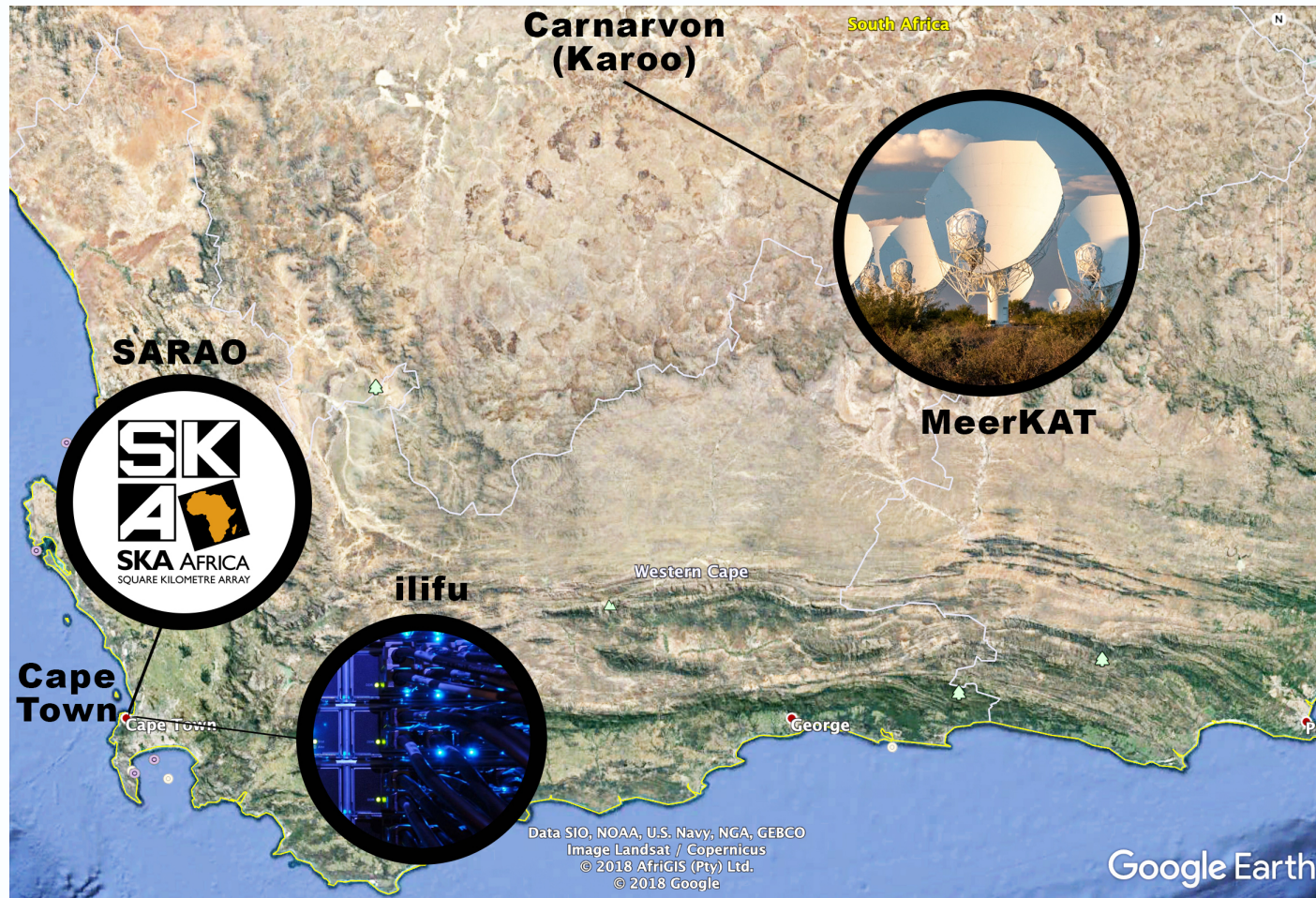
# MeerKAT

Dishes	64
Dish Diameter	13.5 m
Total collecting area	9161 m <sup>2</sup>
Pairs of dishes	2016
Maximum baseline	8 km
Resolution	6" (robust = -0.5)
$A_e / T_{\text{sys}}$ (per dish)	~6 m <sup>2</sup> K <sup>-1</sup> (1.7 over spec!)
Observing frequency	580 – 3500 MHz
Bandwidth	~800 MHz
Spectral Channels	32,768



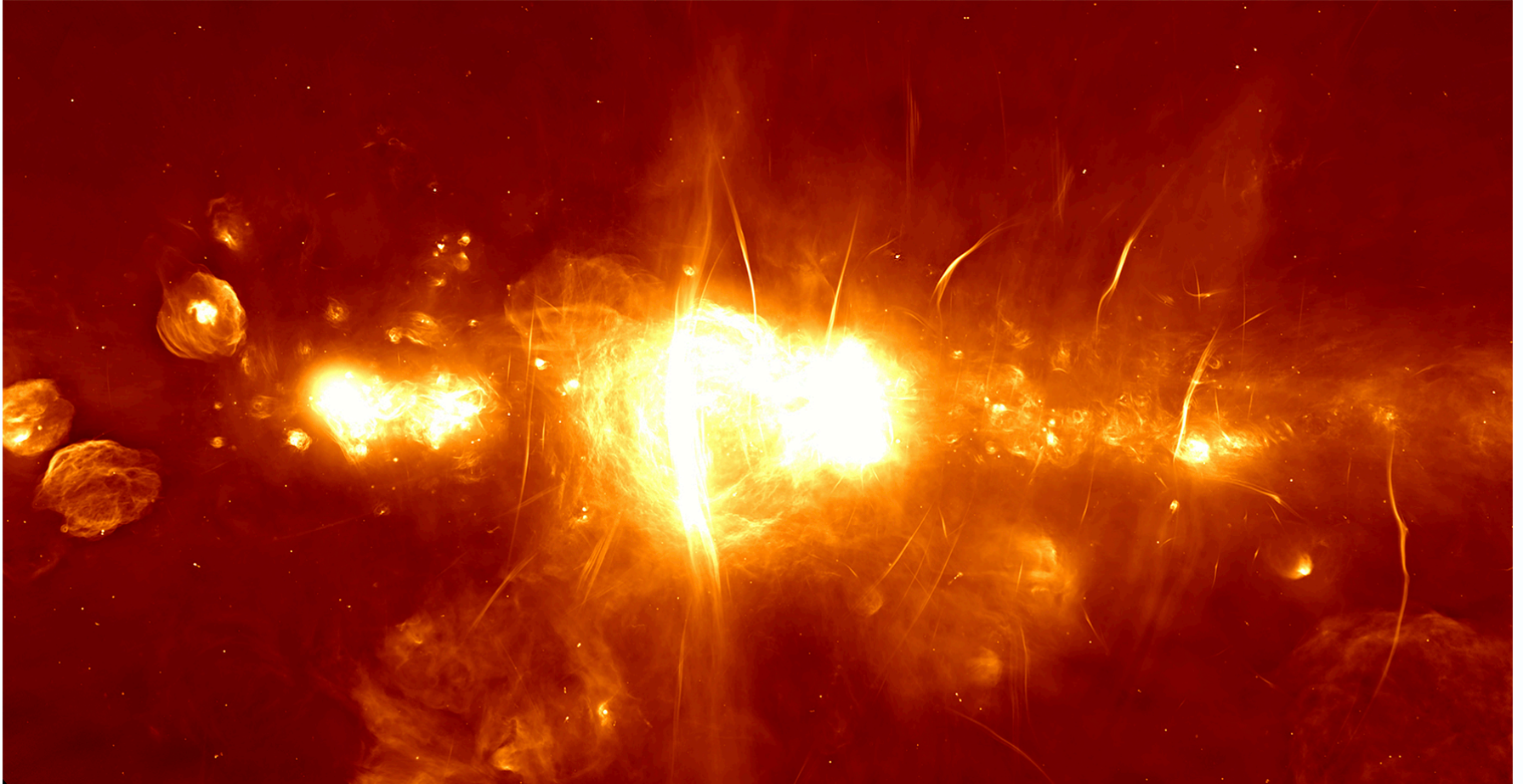


# MeerKAT





# Galactic Centre Image



Radio Continuum Mosaic: Galactic Centre, Ian Heywood, SARA0 (Processed at IDIA)

# MeerKAT Large Survey Projects (LSPs)

- Imaging
  - LADUMA (Deep neutral hydrogen)
  - MIGHTEE (Deep continuum imaging of the early universe)
  - Fornax (Deep HI Survey of the Fornax cluster)
  - MHONGOOSE (targeted nearby galaxies HI)
  - MeerKAT Absorption Line Survey (extragalactic HI absorption)
- Time domain
  - ThunderKAT (exotic phenomena, variables and transients)
  - TRAPUM (pulsar search)
  - Pulsar Timing (MeerTIME)

<http://public.ska.ac.za/meerkat/meerkat-large-survey-projects>





IDIA / ilifu



**UNIVERSITY OF CAPE TOWN**

IYUNIVESITHI YASEKAPA • UNIVERSITEIT VAN KAAPSTAD

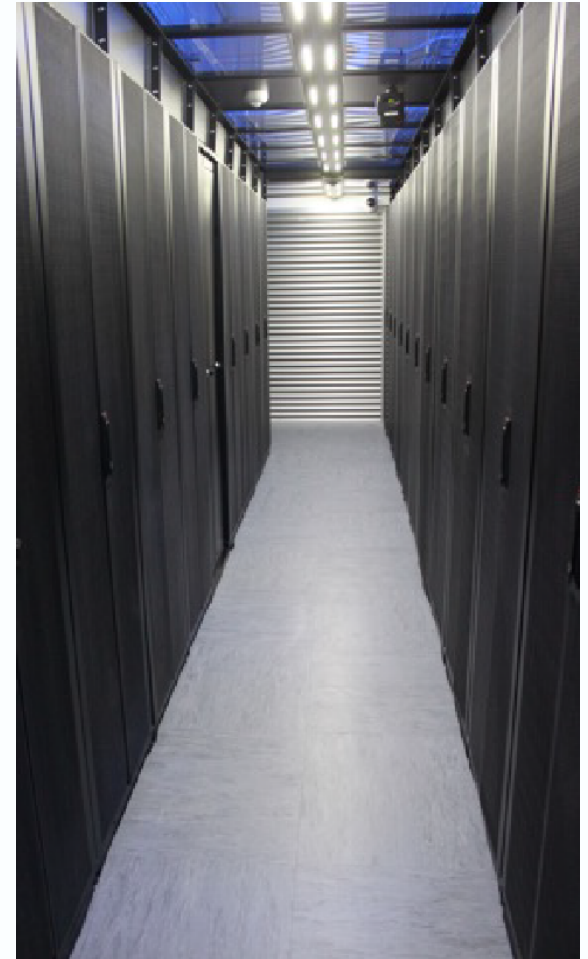
**WESTERN SYDNEY  
UNIVERSITY**



Jordan Collier | 18 December 2019 | Socorro

# ilifu

- ilifu (<http://ilifu.ac.za>)
  - Tier 2 Data Intensive Research Facility
  - Joint Cloud Platform for Astronomy and Bioinformatics (cluster + fat nodes)
  - **Pathfinder Science Regional Data Centre**
  - 60 Nodes (32 cores), 2.6 GHz, 256/512 GB
  - 3.3 PB raw storage (BeeGFS & CEPH)
  - 10 Gb/s network to South African National Research Network (SANReN)





# ilifu and IDIA

- Systems and astronomy support from IDIA
- A Research Service
  - Purpose-built software containers
  - Data processing, analysis, storage, access, transport, etc
- Any member of a supported project (e.g. MeerKAT LSP) can get access — provided they have permission from PIs.



# MeerKAT LSP Use Cases

---

- Full-Stokes continuum images and cubes (MIGHTEE, others)
- Continuum subtracted cubes and moment maps (LADUMA, MHONGOOSE, Fornax, MALS, others)
- Multi-Epoch images (ThunderKAT)





# The IDIA MeerKAT Pipeline



**UNIVERSITY OF CAPE TOWN**

IYUNIVESITHI YASEKAPA • UNIVERSITEIT VAN KAAPSTAD

**WESTERN SYDNEY  
UNIVERSITY**



Jordan Collier | 18 December 2019 | Socorro

# IDIA MeerKAT pipeline (processMeerKAT.py)

---

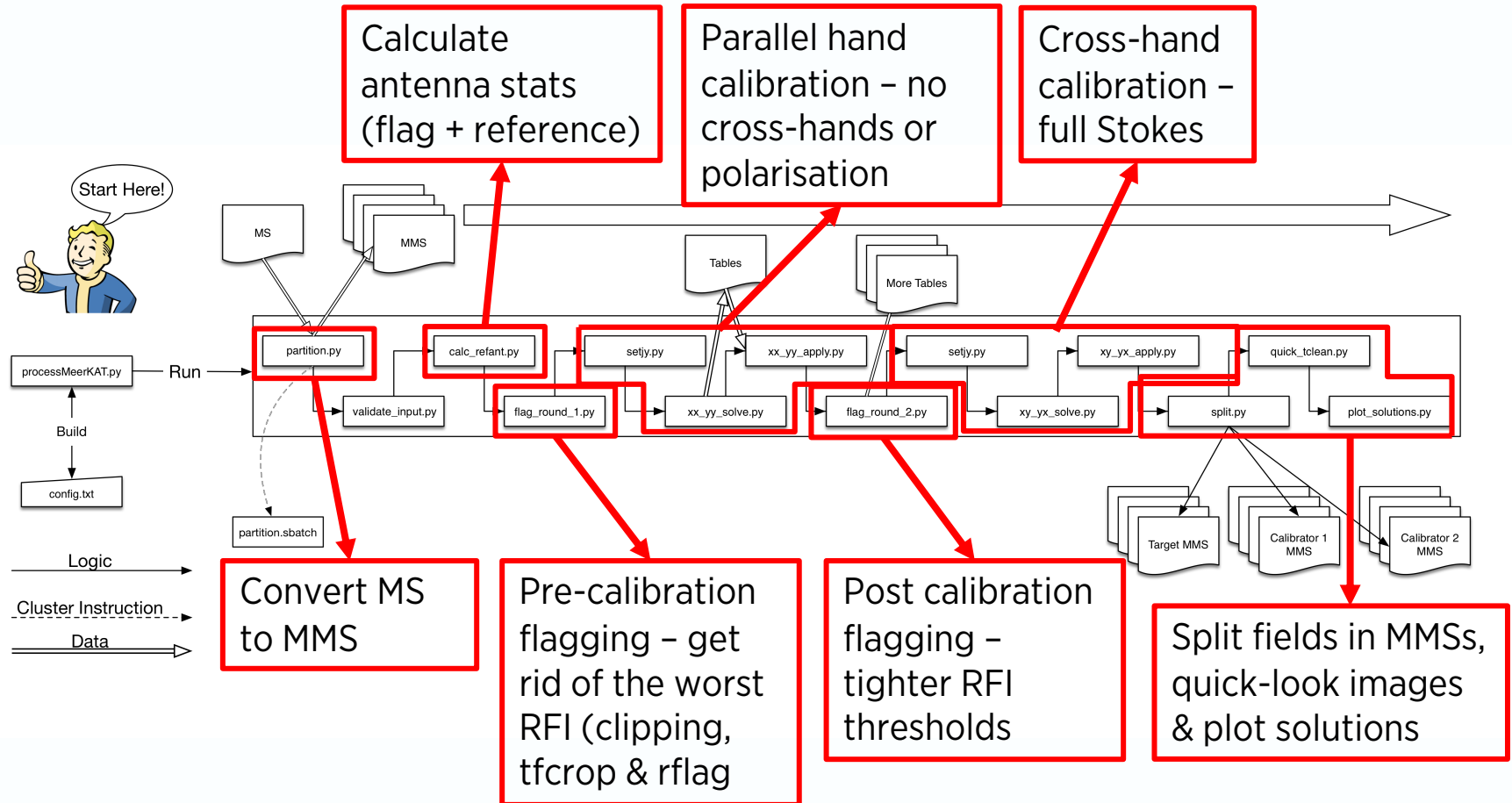
- Brad Frank, Jordan Collier, Srikrishna Sekhar, Russ Taylor
  - V1.0 (released March 2019) performance tested by LSPs
- Full Stokes calibration in CASA
  - Continuum images + polarisation cubes + spectral line cubes
- Parallelised package for HPC processing (SLURM + cluster)
  - Uses multi-measurement sets (MMS) to parallelise across a cluster
- Robust, generic, fast implementation of a priori calibration
- Easy to use, transparent, reproducible
- Builds and submits pipeline jobs to SLURM
  - Input measurement set, build / run your config file, request resources
  - Optionally insert your own scripts, specify containers and MPI wrappers
- Aim:  $T(\text{cal}) \sim T(\text{obs})$

Jordan Collier | 18 Dec 2019 | Socorro





# IDIA MeerKAT pipeline (processMeerKAT.py)

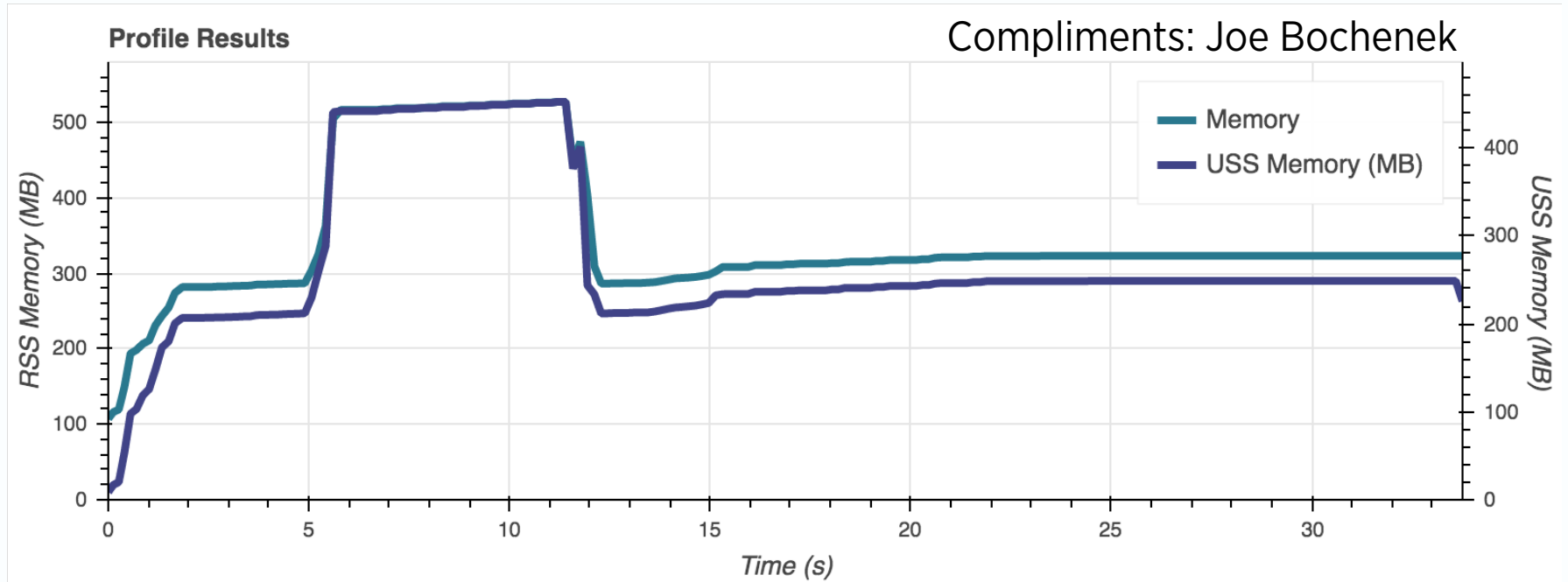


# A Good Framework

---

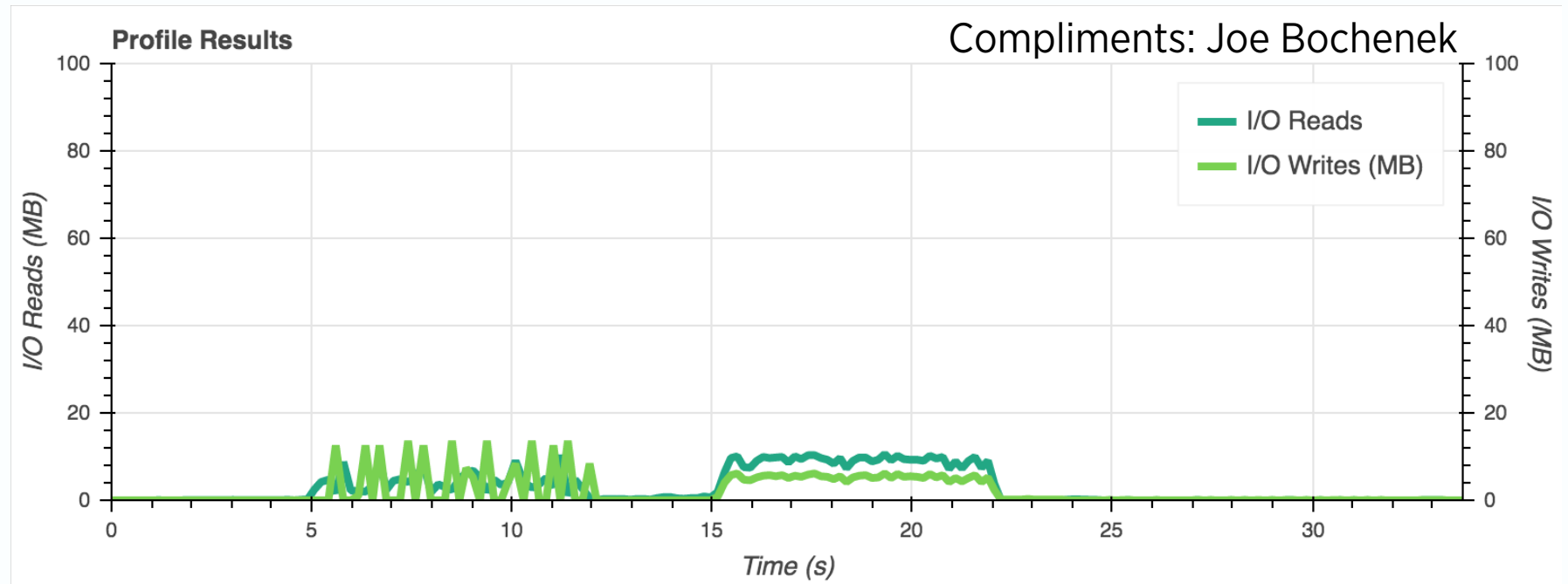
- Outputs calibrated MMS + MSs split by field (push of button!)
- HPC-friendly – dynamically uses resources & submits to queue
- Each job/script is a logical step that does/doesn't use MPI, and optionally uses a different container
- You insert your scripts at start, middle or end (e.g. WSClean)
- Use cases we currently support
  - Full stokes calibration + Stokes I only calibration (minimal speedup)
  - Narrow band (spectral line) calibration, full-band calibration
  - Single MS (speedup for small BW), multi-MS
  - Inserting your own scripts (hard-coded or read config file)
- <https://idia-pipelines.github.io/docs/processMeerKAT>

# Pipeline Diagnostics – Benchmarking

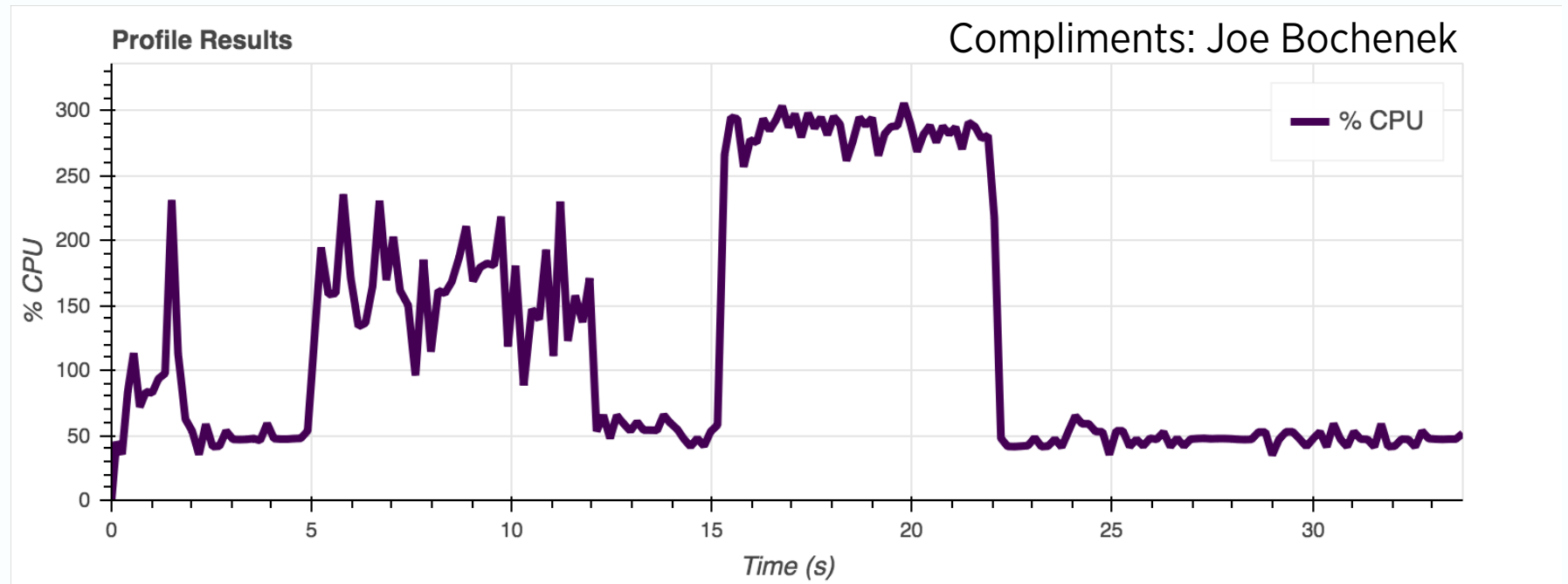




# Pipeline Diagnostics – Benchmarking



# Pipeline Diagnostics – Benchmarking



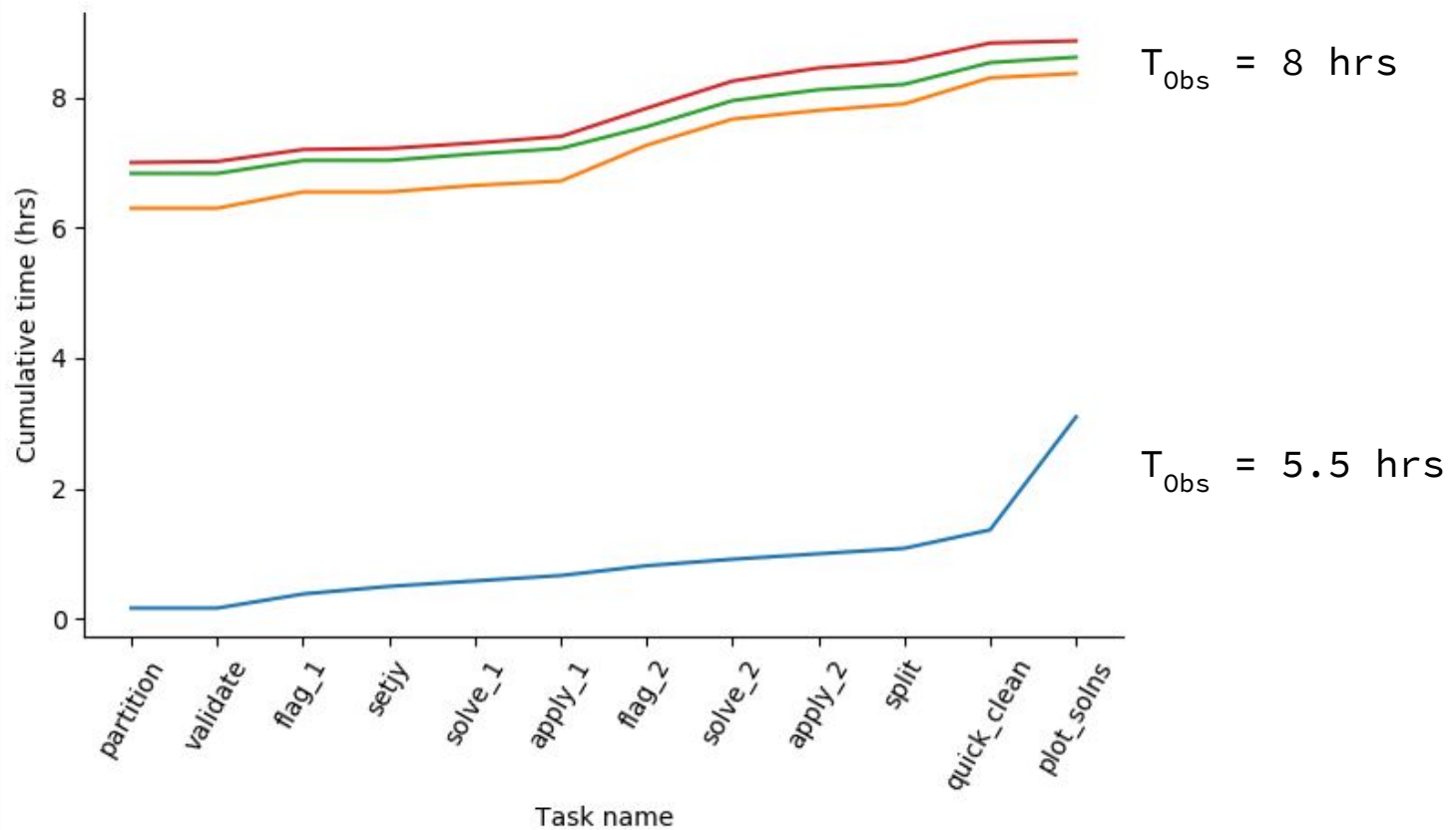
# IDIA MeerKAT pipeline (processMeerKAT.py)

---

- New mode (V1.1) splits into N SPW, to achieve
  1. A speed up due to processing each SPW simultaneously
    - Each launches instance of pipeline, with only partition as serial step
  2. More reliable wideband polarisation & parallelism
    - CASA not “RM aware” over single, wideband SPW
- Takes a few hours per SPW, completed in several hours

# IDIA MeerKAT pipeline (processMeerKAT.py)

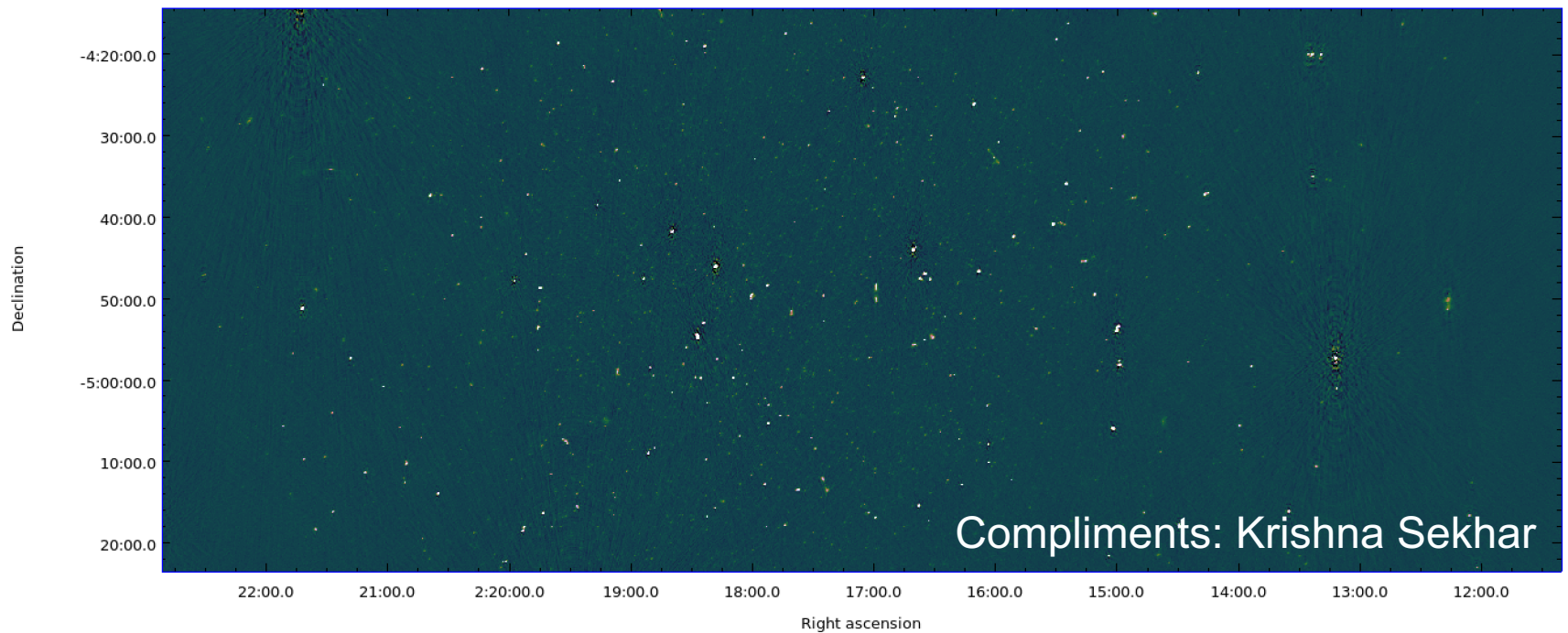
$$T_{\text{Cal}} \sim T_{\text{Obs}}$$





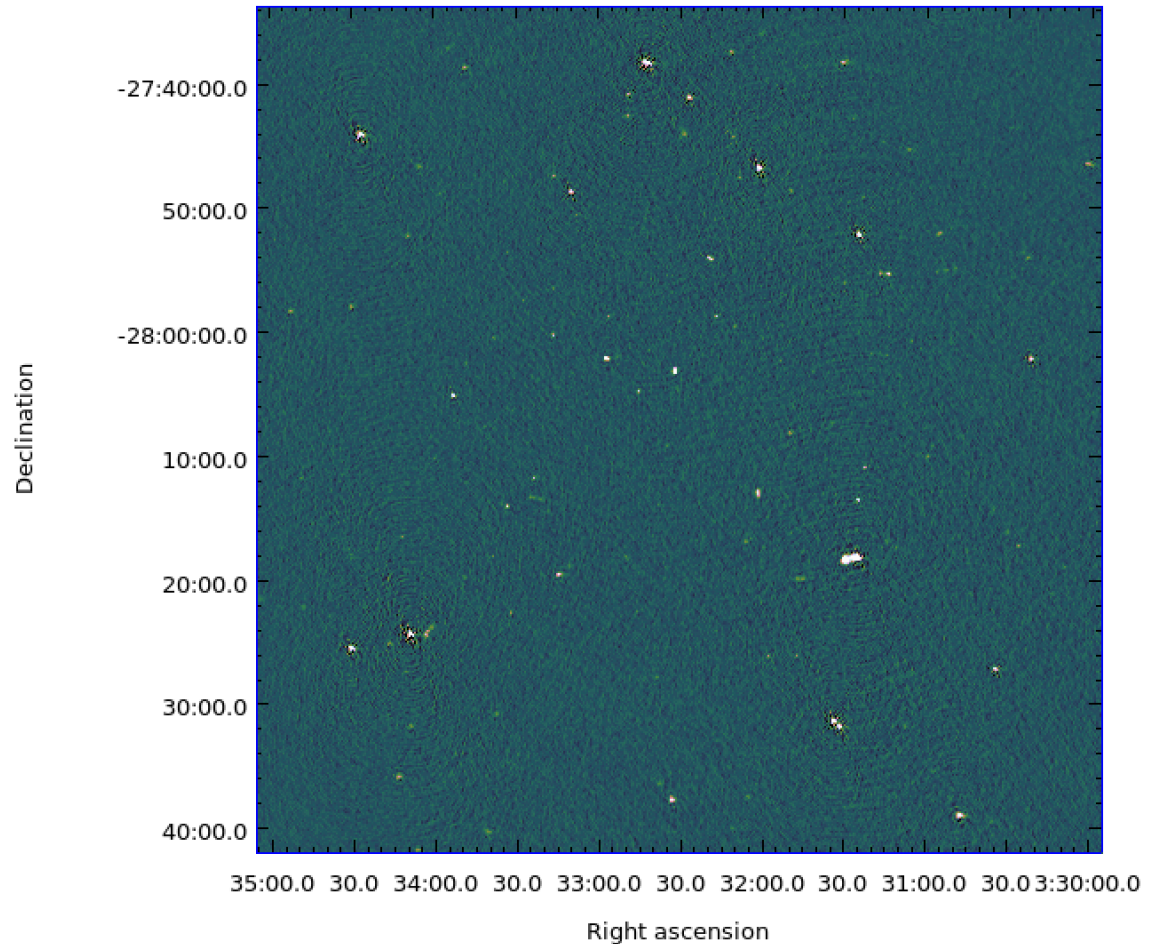
# Initial QA: Quick Look Images

- Very quick and dirty imaging for QA purposes
  - No selfcal, no w-projection, no thresholding, no multi-scale, etc
  - XMM-LSS field: RMS  $\sim 10$   $\mu$ Jy / beam



# Initial QA: Quick Look Images

- LADUMA (CDFS) field
- ~8 hours, 10 MHz spw
- RMS ~80  $\mu$ Jy / beam
- Scales to ~10  $\mu$ Jy over whole BW assuming ~100 MHz flagged out



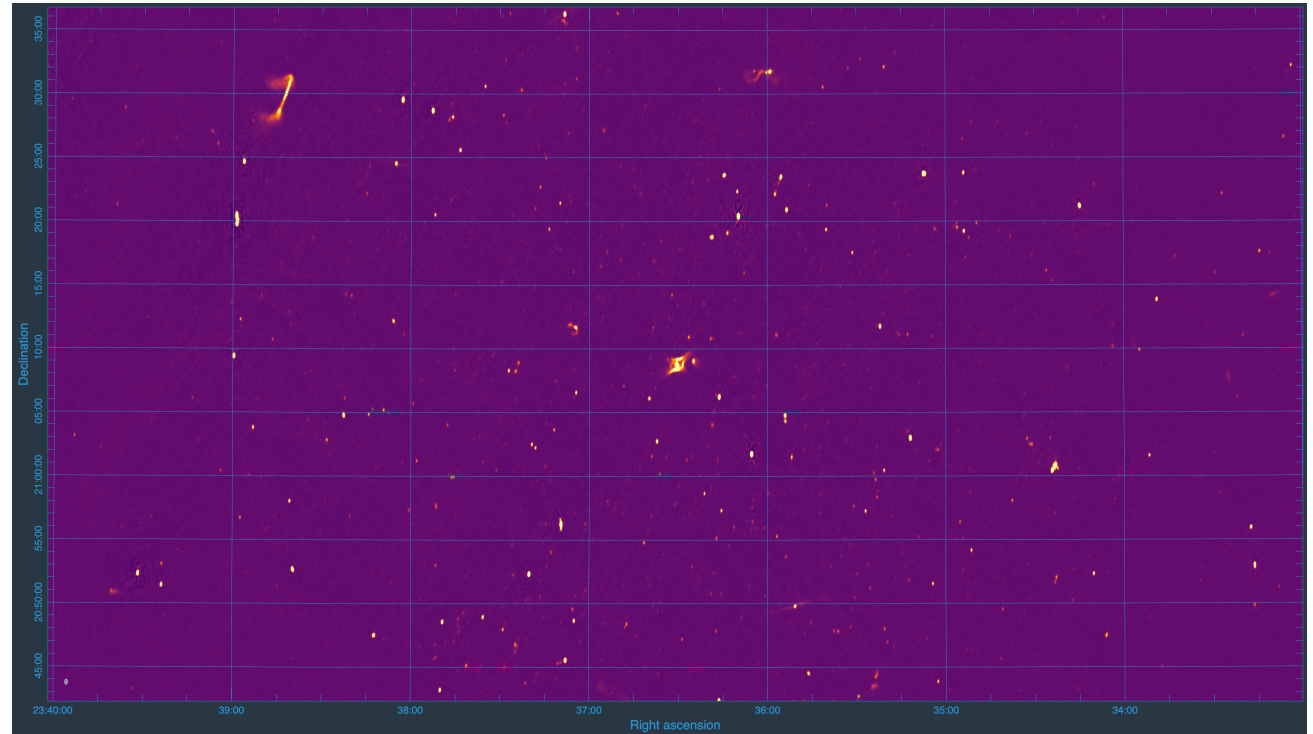
# Current and Future Pipeline Development

---

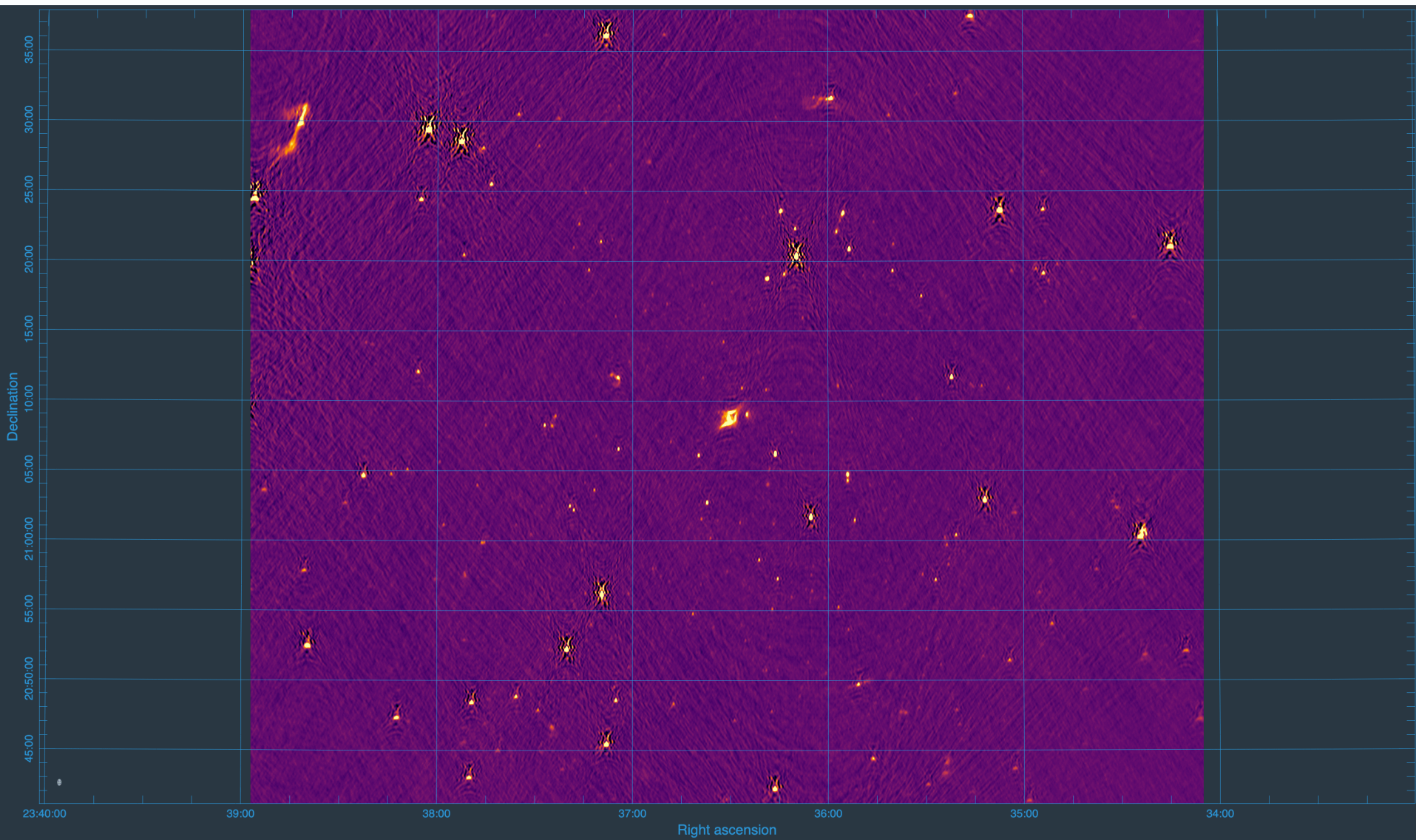
- V1.0 only a 1GC (cross-calibration) pipeline
  - Optimising this to run more than 1 order of magnitude faster (later slide)
- V2.0 (active development): extending to 2GC (self-calibration)
  - Broad algorithm locked in and currently used routinely in single script (PyBDSF, tclean with masking, gaincal, applycal, flagdata on residual)
  - Prototyped single node, multi-CPU, but very slow (days) over whole band
  - Needs to be slightly optimised and then pipelined in 2 steps per loop
- Will soon extend to 3GC (AW-projection: Srikrishna Sekhar)
  - Implemented in tclean, in conjunction with NRAO CASA ARDG group (Preshanth Jagannathan & Sanjay Bhatnagar)
  - Implemented for any telescope, only need holography model
  - Great / successful preliminary results (output images & performance)

# Beyond Quick Look Images

- Abell 2626
- PI: J. Healy
- 100 MHz BW
- 5 hours
- $\sim 7$   $\mu$ Jy / beam RMS
- Up to 2GC, direction-independent selfcal





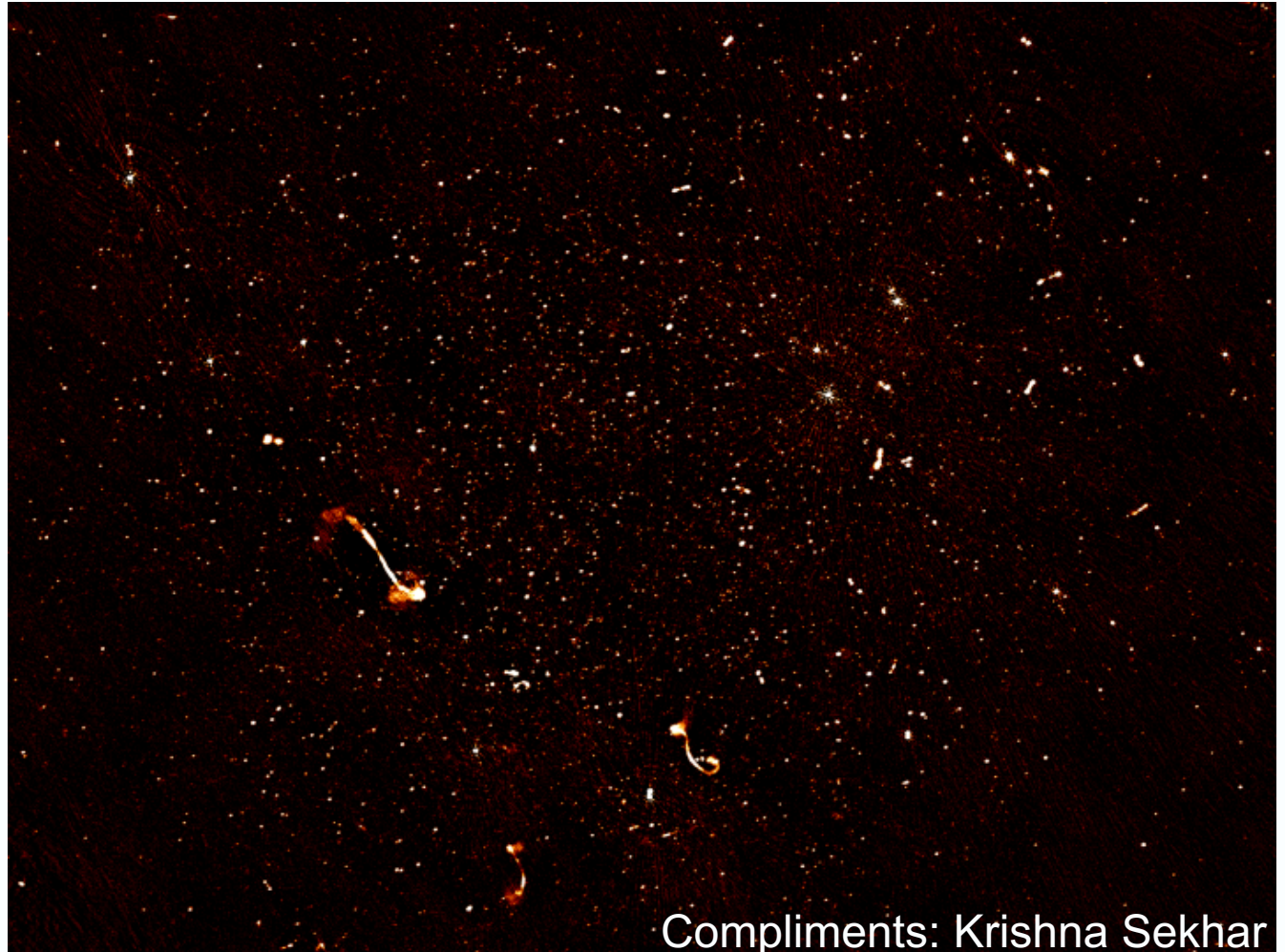


Jordan Collier | 18 Dec 2019 | Socorro



# Beyond Quick Look Images

- MIGHTEE  
CDFS 16
- 55 antennas
- 8 hours
- $\sim 5 \mu\text{Jy}$  /  
beam RMS
- Up to 2GC,  
direction-  
independent  
selfcal



Compliments: Krishna Sekhar

# Beyond Quick Look – Future Pipeline Dev.

---

- Optimisation of resources / performance
  - Currently takes ~1 day to calibrate 64 dish 4k data (~2.5 TB) with nspw = 1
- Split data into separate (calibrator and target) MMSs for during beginning, and simultaneously calibrate / flag
  - Prototyped manually, reducing time calibrate to several hours
  - Also want dynamic use of threads & memory per script per intent (based on benchmarking)
    - Partitioning (IO), flagging (RAM), imaging (CPU)
  - Will see a significant speedup, necessary for arrival of 32k data (now!)



# A focus on tclean

---

- Next step of pipeline development, beyond quick-look images
- Limited by bugs in CASA
  - Runtime of makePSF step when imsize x 1.2 contains large prime factor (see Joe Bochenek's report) – patched on Monday, hopefully for CASA 5.7!
  - CASA crashes when writing to model column with MPICASA
    - tclean with savemodel='modelcolumn' (a known issue)
  - Initial testing of AW-projection algorithm shows good scaling with threads
  - CASA exit codes
- MPI vs. OpenMP
  - CPU-level parallelism (OpenMP) seems to scale linearly for FT/iFT, gridding
  - Task-level parallelism (MPICASA) seems to distribute the work over n scans during major cycle (also minor cycle for cube imaging over n channels)
  - What fraction of time spent in each? How do each scale? Which is better?



# Summary

---

- Many challenges involved in delivering / operating MeerKAT
  - IDIA/ilifu has solved many of the computational challenges
- ilifu a good model / prototype SKA Regional Centre
- The IDIA “processMeerKAT” pipeline is an efficient, user-friendly pipeline, that is widely tested and documented
  - Have successfully imaged some test 32k data with our pipeline!
  - ~8-10 uJy/beam RMS without selfcal, (currently) 4 uJy/beam after selfcal
- It runs on the ilifu cluster, making dynamic use of resources, and containers, and presents a good framework for pipelines
- Many use cases are supported/demonstrated, incl. continuum, spectral line, polarisation, and inserting your own scripts
- Coming soon: selfcal, AW-projection, optimisation/speedup

# THANK YOU

**Dr Jordan Collier**

ilifu Support Astronomer, IDIA  
Department of Astronomy,  
University of Cape Town

Adjunct Fellow, Western Sydney University  
School of Computing, Engineering and  
Mathematics

Jordan@idia.ac.za  
+27 664 343 953 (RSA Mobile)  
+61 414 443 622 (AU Mobile / WhatsApp)



**UNIVERSITY OF CAPE TOWN**  
IYUNIVESITHI YASEKAPA • UNIVERSITEIT VAN KAAPSTAD



**WESTERN SYDNEY**  
UNIVERSITY