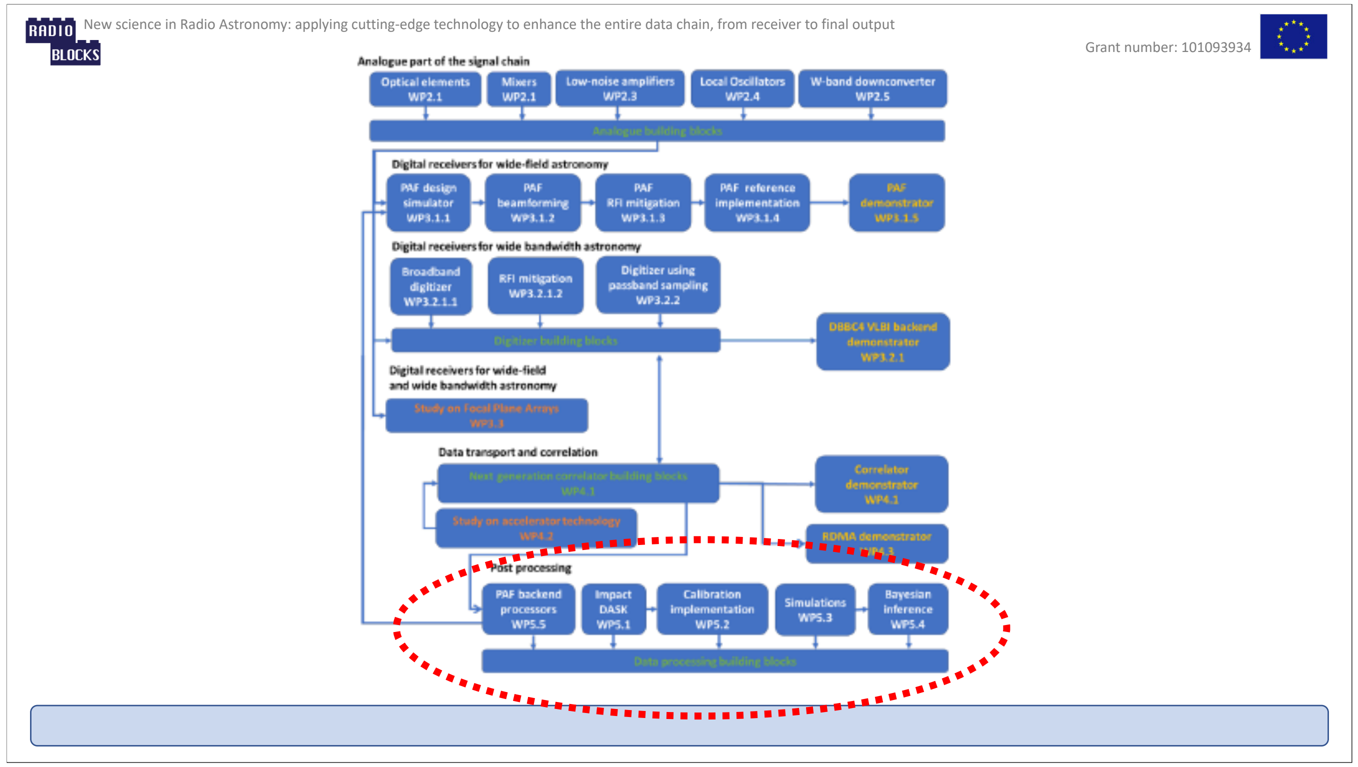
# WP5: Data processing toolkit for Advanced Radio Astronomy

M. Verkouter (JIVE) & R. Beswick (e-MERLIN, UKSRC / JBCA, University of Manchester)

*on behalf of WP5 team:*

ASTRON, JIVE, VIRAC, CSIC, MPG, ULEI, INAF, RadboudU, HeidelbergU,
UMAN, SKAO, EPFL, UP, RATT, CNIG/OAN, ICRAR

This talk is about WP5: data processing toolkits for advanced radio astronomy. The information in this talk represents the work of many people - no less than 16 partner institutes are involved.




I'm going to skip most of the pleasantries - WP5 sits there at the other end of the end-to-end signal chain

# WP5 Overview & objectives

- Main Objectives: Development of ***modular***, ***open-source*** and flexible toolkit components for associated workflows. Components to enable rapid, reproducible and ***scalable*** analysis tools. ***Strong emphasis on translation of knowledge & developments between different but adjacent RIs*** (*pooling, sharing and developing*)
- Supports and integrates with WP3 & WP4 (in particular)
- Essential to maximise outputs from facilities

and summarizing the objectives, the aim is for scalable open source toolkits with a strong emphasis in knowledge-sharing and translation between adjacent research infrastructures. The work in WP5 connects to both WP3 but in particular to WP4.
It should be noted that efficient and state-of-the art post-processing is essential to maximise science output from the facilities

# WP5 structure & tasks:

## 5 key task areas:

- *Task 5.1 : The impact of DASK on automated processing workflows for Radio Astronomy data*
  ***ASTRON***,*VIRAC, UNIMAN, EPFL, SKAOb, RATT*
- Task 5.2 : *Develop a generic and scalable fringe fit calibration implementation in the Dask framework*
  ***JIV-ERIC***
- *Task 5.3 : Simulations for optimising calibration and parameter extraction*
  ***JIV-ERIC***, *UNIMAN, Radboud, CSIC, CNIG/OAN, UP, ICRAR*
- *Task 5.4 : Bayesian inference for sparse visibility data*
  ***ULEI, HeidelbergU***, *INAF*
- *Task 5.5 : Modular PAF Backend Processors toolkit*
  ***MPG***

➢Each task coordinated by ***independent teams***, brought together to share knowledge & expertise.

➢Modular, open-source and flexible components to process interferometry data

Here we briefly bring into recollection the five tasks that make up the full work package.

New science in Radio Astronomy: applying cutting-edge technology to enhance the entire data chain, from receiver to final output

Grant number: 101093934

# WP5 structure & tasks:

## 5 key task areas:

- *Task 5.1 : The impact of DASK on automated processing workflows for Radio Astronomy data*
  ***ASTRON***,*VIRAC, UNIMAN, EPFL, SKAOb, RATT*
- Task 5.2 : *Develop a generic and scalable fringe fit calibration implementation in the Dask framework*
  ***JIV-ERIC***
- *Task 5.3 : Simulations for optimising calibration and parameter extraction*
  ***JIV-ERIC***, *UNIMAN, Radboud, CSIC, CNIG/OAN, UP, ICRAR*
- *Task 5.4 : Bayesian inference for sparse visibility data*
  ***ULEI, HeidelbergU***, *INAF*
- *Task 5.5 : Modular PAF Backend Processors toolkit*
  ***MPG***

➢Each task coordinated by ***independent teams***, brought together to share knowledge & expertise.

➢Modular, open-source and flexible components to process interferometry data

two tasks focus on scalable processing, possibly leveraging GPU acceleration and/or tensor cores

Grant number: 101093934

# WP5 structure & tasks:

## 5 key task areas:

- *Task 5.1 : The impact of DASK on automated processing workflows for Radio Astronomy data*
  ***ASTRON****,VIRAC, UNIMAN, EPFL, SKAOb, RATT*
- Task 5.2 : *Develop a generic and scalable fringe fit calibration implementation in the Dask framework*
  ***JIV-ERIC***
- *Task 5.3 : Simulations for optimising calibration and parameter extraction*
  ***JIV-ERIC****, UNIMAN, Radboud, CSIC, CNIG/OAN, UP, ICRAR*
- *Task 5.4 : Bayesian inference for sparse visibility data*
  ***ULEI, HeidelbergU****, INAF*
- *Task 5.5 : Modular PAF Backend Processors toolkit*
  ***MPG***

➢Each task coordinated by ***independent teams***, brought together to share knowledge & expertise.

➢Modular, open-source and flexible components to process interferometry data

One task that focuses on simulating data sets with realistic errors to allow calibration algorithms to be analysed how well the introduced errors can be recovered

# WP5 structure & tasks:

## 5 key task areas:

- *Task 5.1 : The impact of DASK on automated processing workflows for Radio Astronomy data*
  ***ASTRON****,VIRAC, UNIMAN, EPFL, SKAOb, RATT*
- Task 5.2 : *Develop a generic and scalable fringe fit calibration implementation in the Dask framework*
  ***JIV-ERIC***
- *Task 5.3 : Simulations for optimising calibration and parameter extraction*
  ***JIV-ERIC****, UNIMAN, Radboud, CSIC, CNIG/OAN, UP, ICRAR*
- *Task 5.4 : Bayesian inference for sparse visibility data*
  ***ULEI, HeidelbergU****, INAF*
- *Task 5.5 : Modular PAF Backend Processors toolkit*
  ***MPG***

➢Each task coordinated by ***independent teams***, brought together to share knowledge & expertise.

➢Modular, open-source and flexible components to process interferometry data

A task that focuses on bringing Bayesian inference to sparse visibility data sets from the EHT domain into cm-wavelength, possibly benefiting from tensor-core based hardware

Grant number: 101093934

# WP5 structure & tasks:

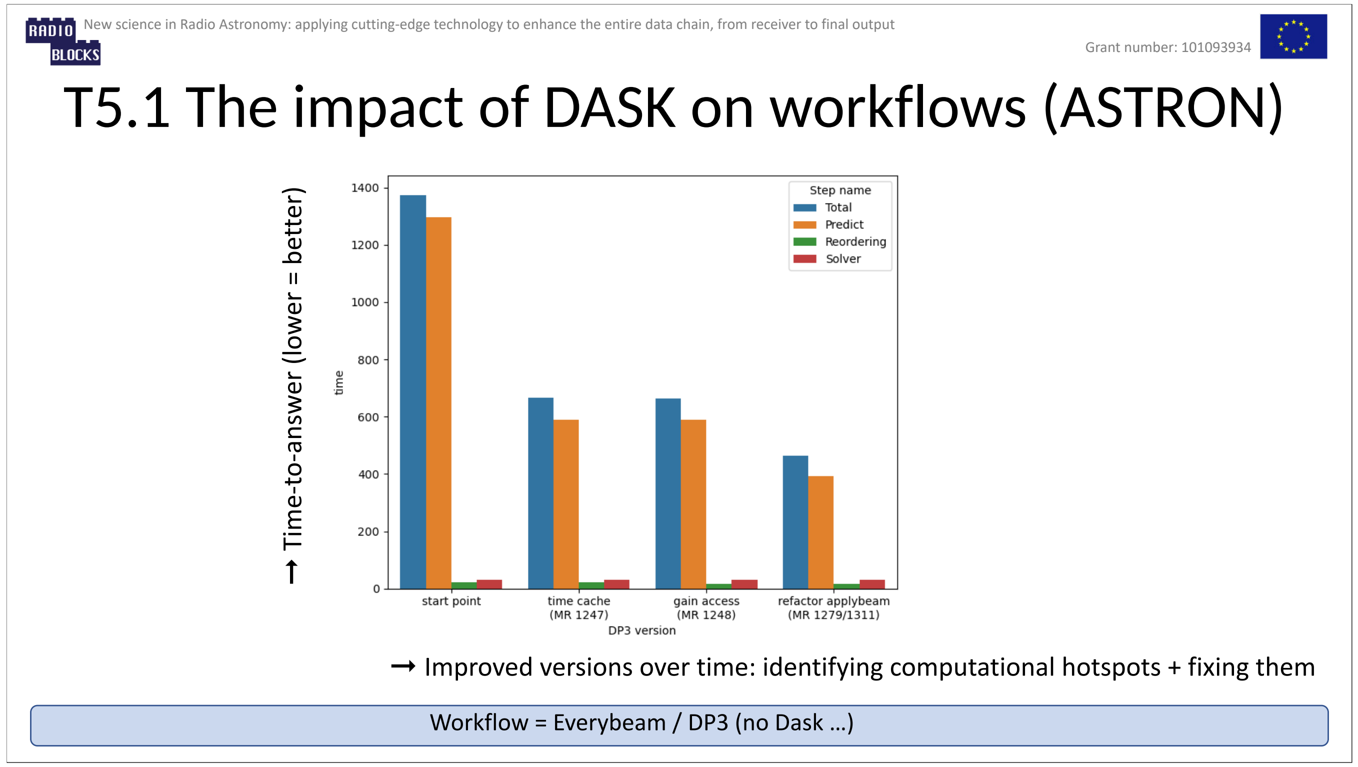
## 5 key task areas:

- *Task 5.1 : The impact of DASK on automated processing workflows for Radio Astronomy data*
  ***ASTRON****,VIRAC, UNIMAN, EPFL, SKAOb, RATT*
- Task 5.2 : *Develop a generic and scalable fringe fit calibration implementation in the Dask framework*
  ***JIV-ERIC***
- *Task 5.3 : Simulations for optimising calibration and parameter extraction*
  ***JIV-ERIC****, UNIMAN, Radboud, CSIC, CNIG/OAN, UP, ICRAR*
- *Task 5.4 : Bayesian inference for sparse visibility data*
  ***ULEI, HeidelbergU****, INAF*
- *Task 5.5 : Modular PAF Backend Processors toolkit*
  ***MPG***

➢Each task coordinated by ***independent teams***, brought together to share knowledge & expertise.

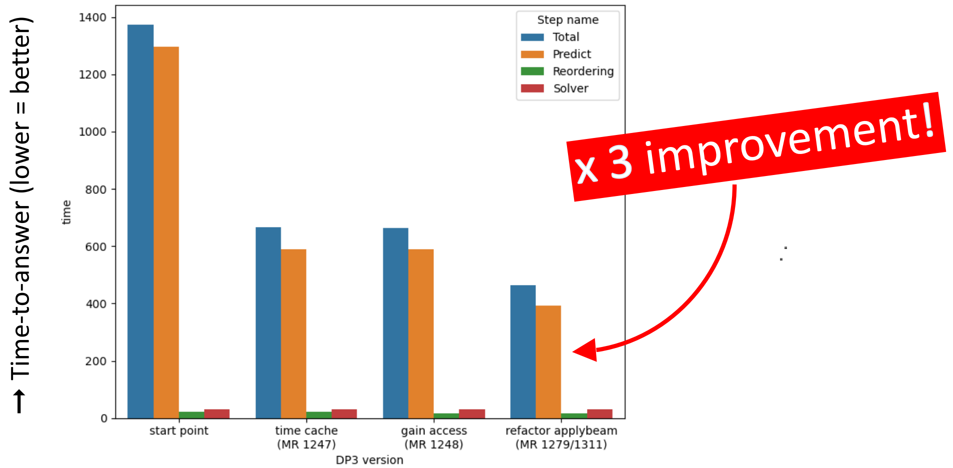
➢Modular, open-source and flexible components to process interferometry data

And finally the modular PAF processing toolkit based around GPU slash tensor-core based acceleration




The team at ASTRON have been working on e.g. the every-beam component in the LOFAR DP3 workflow. Through profiling and analysis computational hotspot(s) could be identified and fixed, leading to

# T5.1 The impact of DASK on workflows (ASTRON)

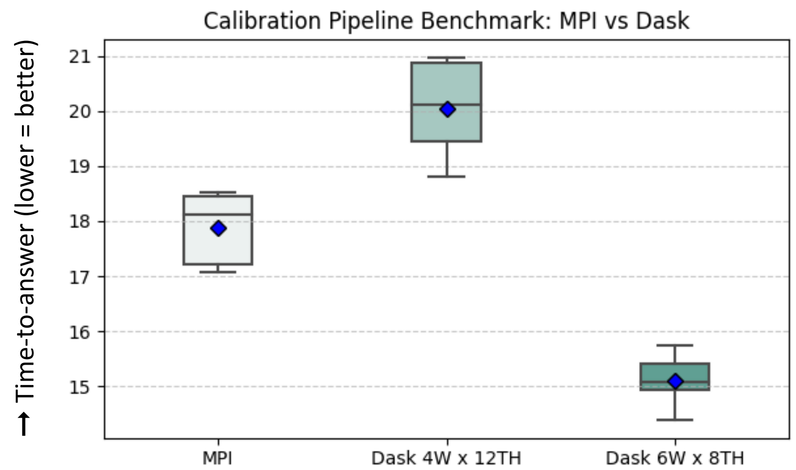



→ Improved versions over time: identifying computational hotspots + fixing them

Workflow = Everybeam / DP3 (no Dask ...)

a times 3 improvement of speed

# T5.1 The impact of DASK on workflows (VIRAC)




→ MPI versus two different Dask configurations

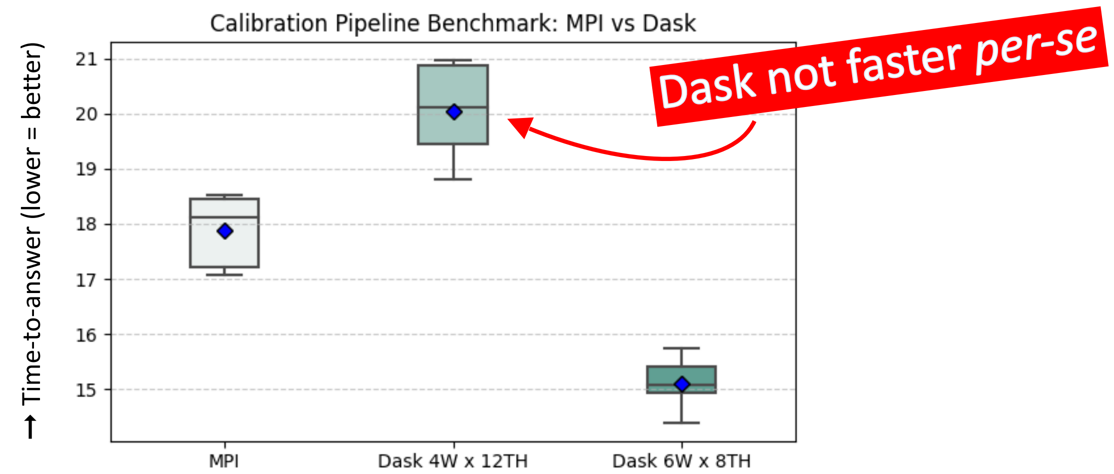
Workflow = Calibration pipeline developed at VENTSPILS

The VENTSPILS team ported their algorithm from the Message Passing Interface (MPI) to Dask and compared the results.

New science in Radio Astronomy: applying cutting-edge technology to enhance the entire data chain, from receiver to final output

Grant number: 101093934

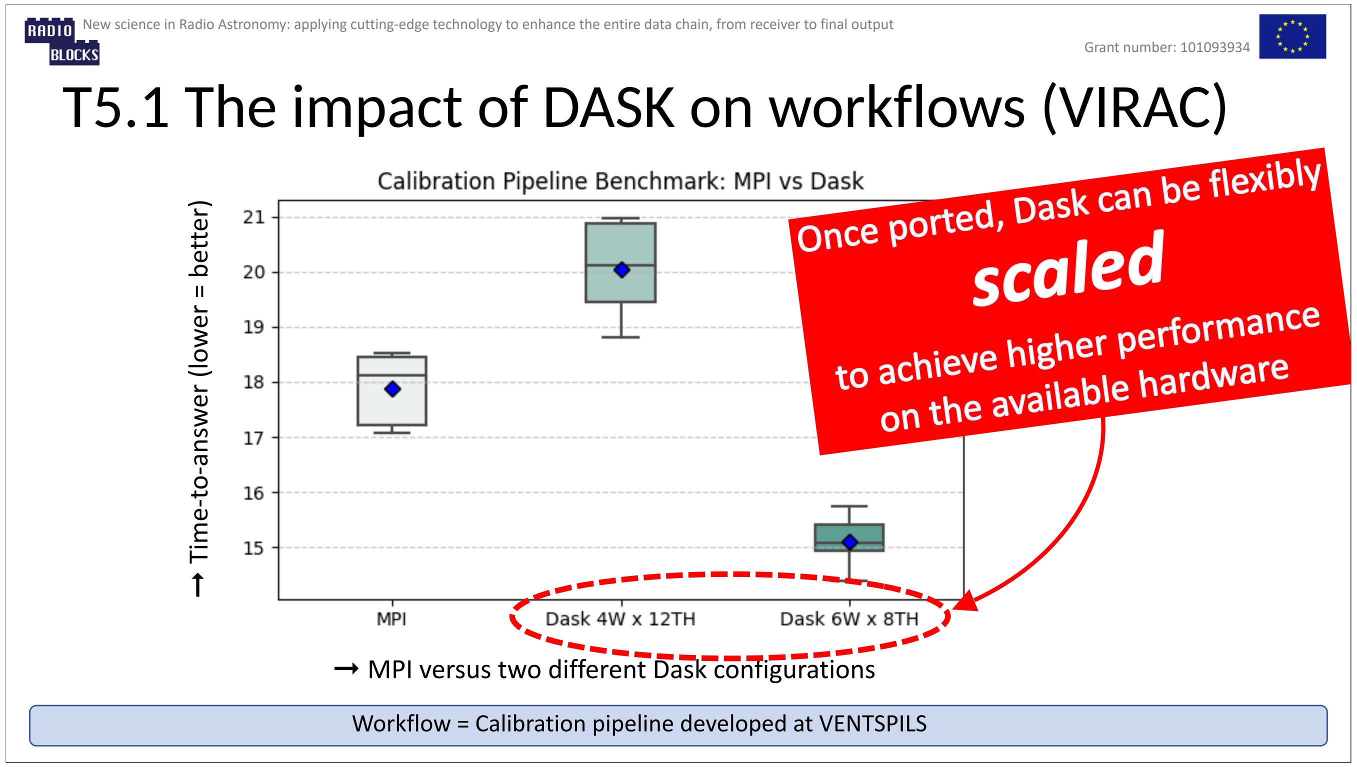
# T5.1 The impact of DASK on workflows (VIRAC)



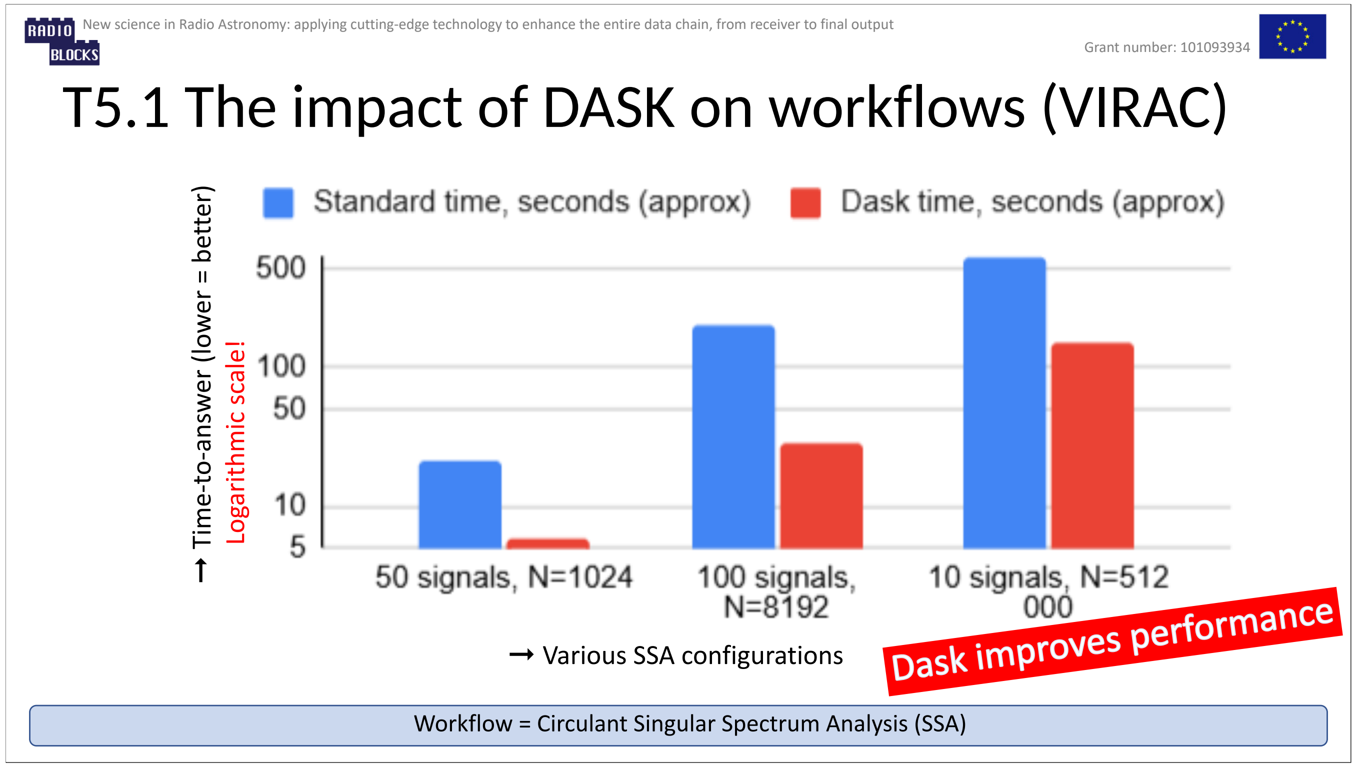

→ MPI versus two different Dask configurations

Workflow = Calibration pipeline developed at VENTSPILS

The IMPORTANT result here is to see that porting an algorithm to Dask by itself does not guarantee it is faster




but IF the algorithm is ported, which is a ONE TIME INVESTMENT, then SCALING the processing over the available hardware does not impact the algorithm and allows tuning to get the maximum performance out of the hardware

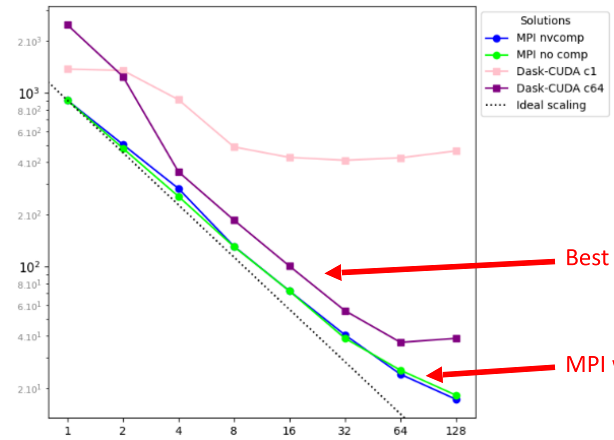



VENTSPILS also investigated MPI versus Dask based implementations of the singular spectrum analysis algorithm and Dask enables better performance across different signal / noise decompositions

RADIO BLOCKS New science in Radio Astronomy: applying cutting-edge technology to enhance the entire data chain, from receiver to final output

Grant number: 101093934

# T5.1 The impact of DASK on workflows (EPFL)




→ Time-to-answer (lower = better)

Logarithmic scale!

Dask can not meet MPI performance?
**Under investigation!**

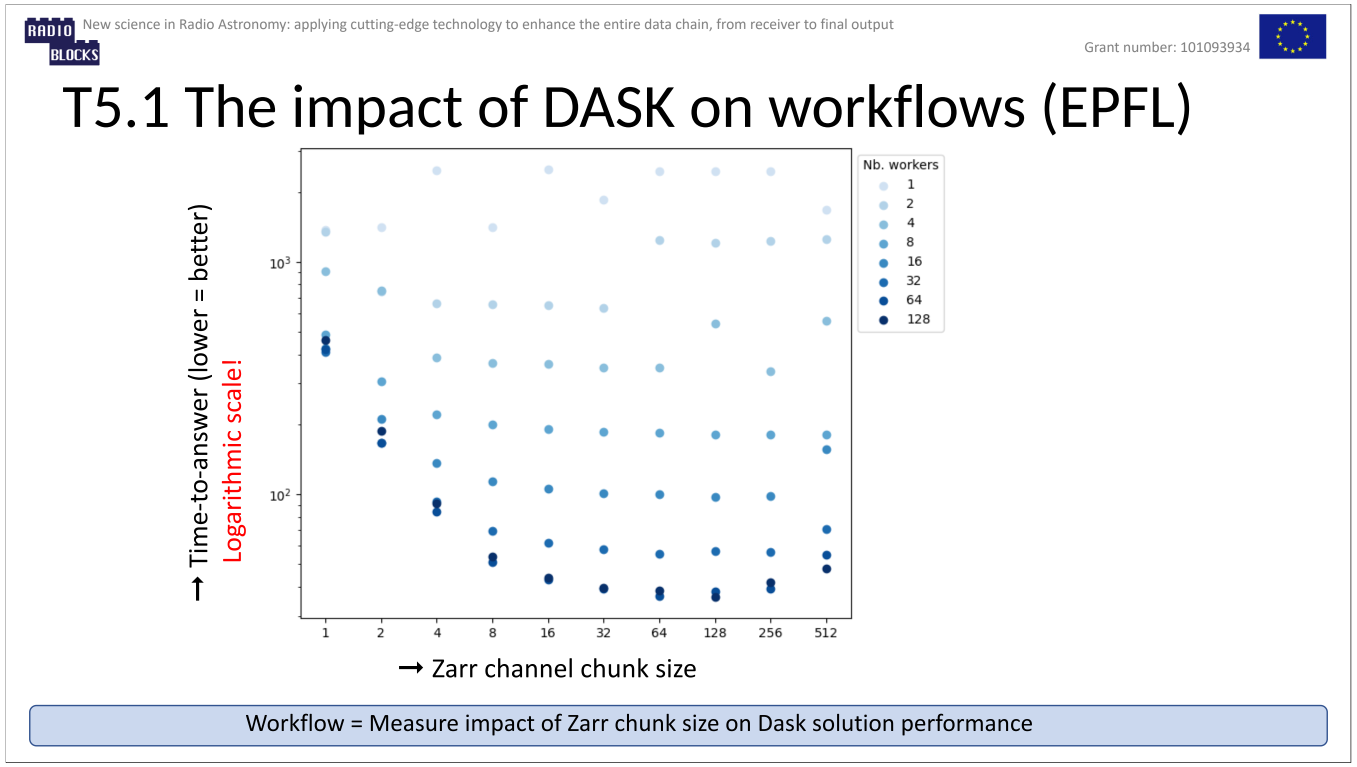
Best Dask performance

MPI w/wo NVIDIA (de)compression lib

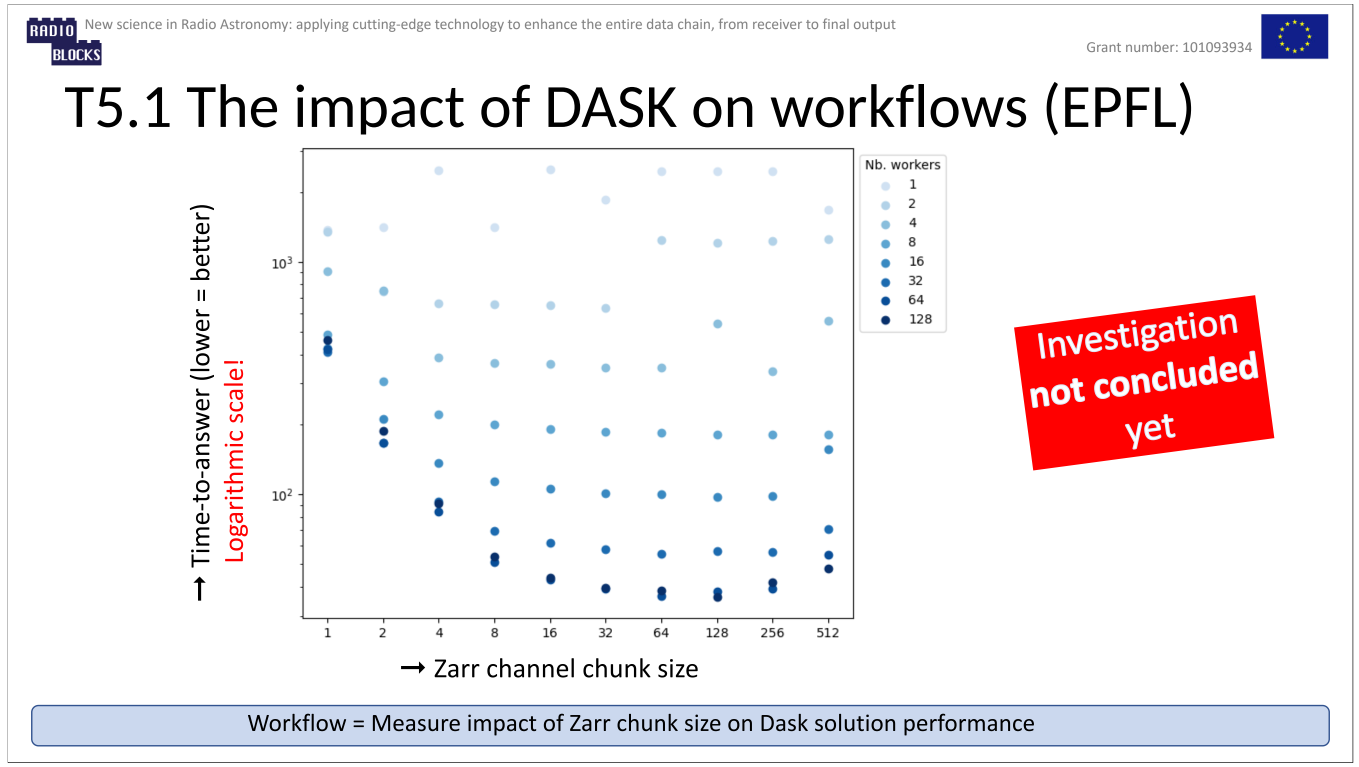
→ Number of MPI ranks or Dask-CUDA workers (on H100 GPU)

Workflow = combined gridding + imaging steps

At EPFL a cuda-based workflow was ported from MPI to Dask and the two were compared, finding different results - specifically, the Dask version could not match the MPI performance.




EPFL then started to investigate the impact of Dask chunk size.to see if there are bottlenecks or sweetspots and yes, the chunk size plays a large part.




EPFL then started to investigate the impact of Dask chunk size.to see if there are bottlenecks or sweetspots and yes, the chunk size plays a large part.

# 5.2 Fringe fit in DASK framework (JIV-ERIC)

Our mission is simple: write code. Progress until now:

- Rudimentary algorithm already by RP1
- Now collaborating at core-level with NRAO’s RADPS/xradio dev team:
    - design & implement RADPS middleware to store / load calibration tables
    - design fringe-fit calibration table layout
    - even more ”core”-level: part of the design team of the actual *calibration framework*
        - years of experience in current CASA framework highly relevant for suggesting improvements

In task 5.2 the focus now is on strong collaboration with NRAO’s RADPS / xradio team on designing and implementing both the middleware that handles storing, retrieving and applying calibration tables. But even closer to the core of the new framework, this involves designing the calibration framework in general. The JIV-ERIC partner expert knowledge built up over the years is extremely useful here.

Grant number: 101093934

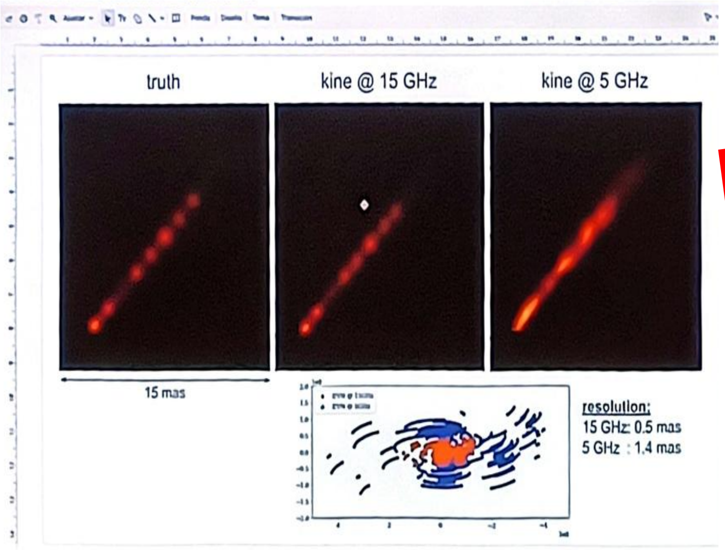
# T5.3 Simulations for optimising calib (various)

- Collaboration w/ ICRAR: more simulation tools accessible:
  - `ARIS` simulation software
- Also: `ngEHTsim` and `eht-imaging`
- Simulated time-variable X-ray binary system
  - 5 + 15 GHz
  - with EVN-like array
- Simulated SKA-LOW data set (heavily affected by ionosphere):
  - Run T5.2 fringe-fit(*) algorithm on that data: looks like *promising results*!

*(*) not the Dask-based, but the advanced/slow/memory inefficient but including dispersive ionospheric fitting algorithm*

Simulations have started to be generated in Task 5.3. Part of this is because of ICRAR and OAN joining the task. So far this have yielded a simulated SKA-LOW dataset and a time-variable X-ray binary source was generated using an EHT tool for an EVN-like array at EVN-like observing frequencies.

# T5.3 Dynamic imaging (IAA-CSIC + CNIG)



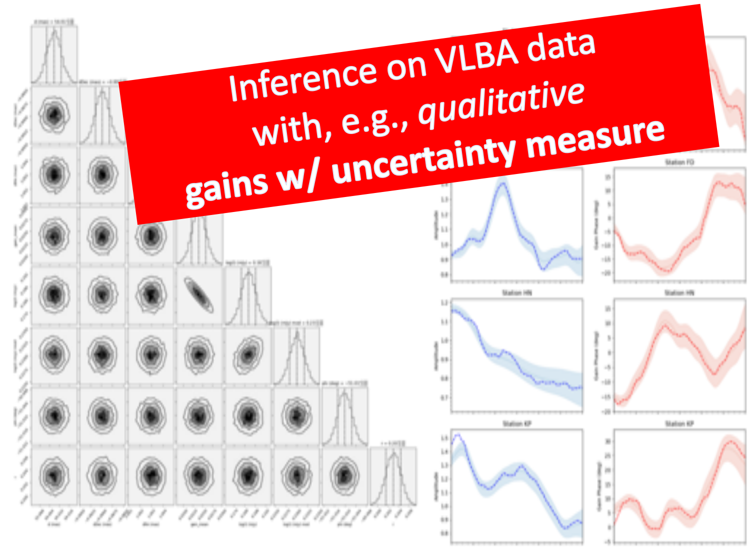

**Applying**
**EHT dynamic imaging algorithm**
**to**
**simulated EVN data!**

The simulated time-variable X-ray binary data was processed by the dynamic imaging software designed for the EHT and that seems to work pretty well!

New science in Radio Astronomy: applying cutting-edge technology to enhance the entire data chain, from receiver to final output

Grant number: 101093934

# T5.4 Bayesian inference (LeidenU)




Using:
- `numpyro`
- JAX's `autodiff` function
- VLBA data

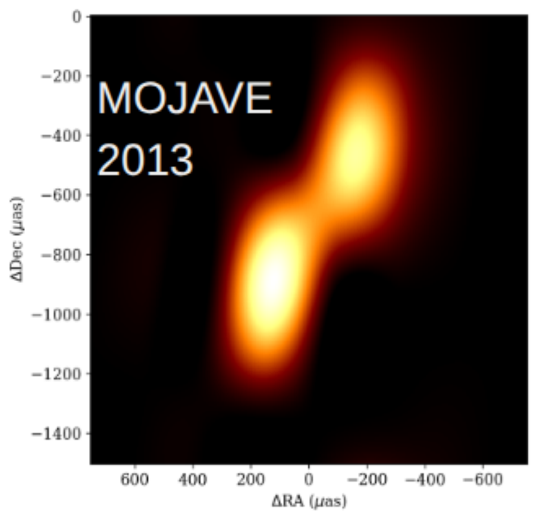
On TensorCore hardware:
- estimating 200 individual uncertain gain parameters simultaneously!

*https://num.pyro.ai/*
*https://github.com/jax-ml/jax*

MCMC sampling, estimating VLBA model parameters

In task 5.4 Leiden University were able to run an inference on VLBA data, estimating many independent parameters, e.g. 200 individual gain parameters, yielding for the first time the time variable gains with their uncertainty measure.

# T5.4 Bayesian inference (HeidelbergU)

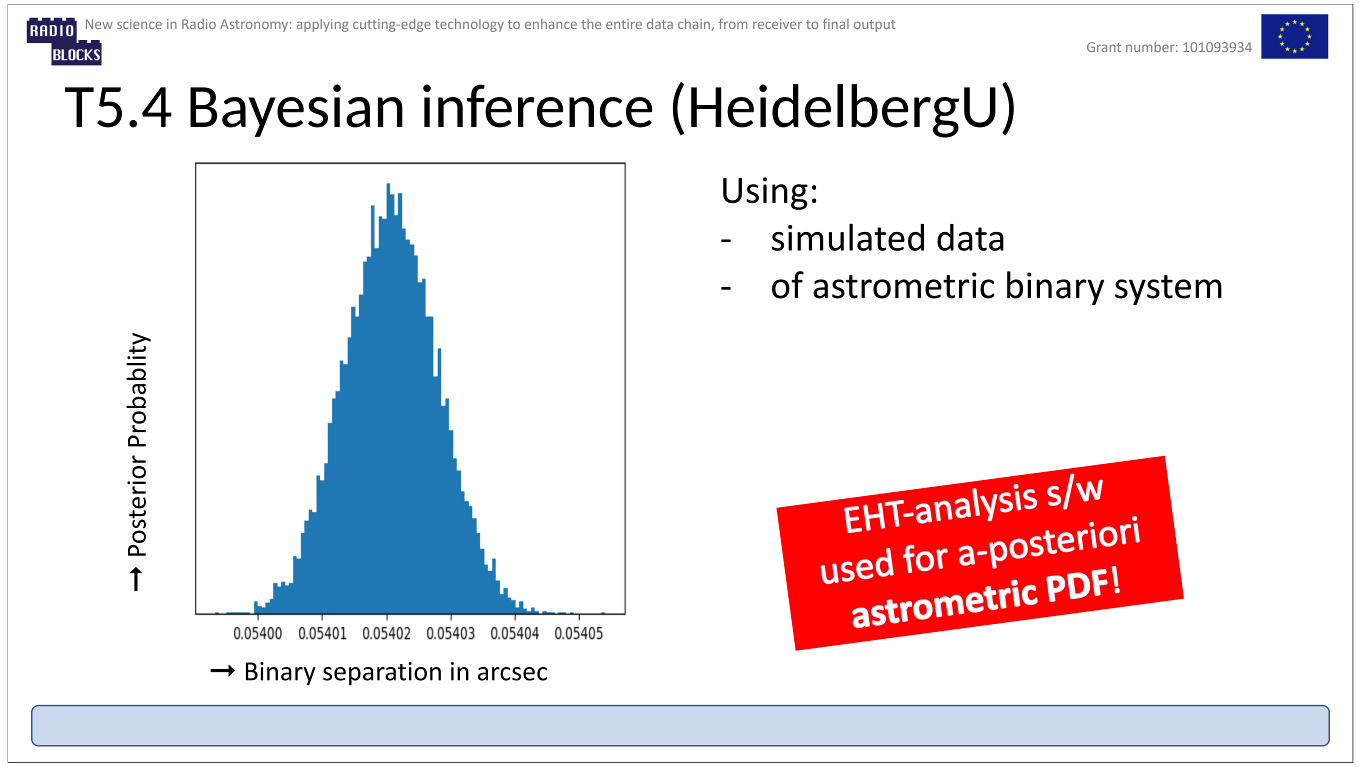



Using:

- MOJAVE data on `J1924-2914`
- `Themis` image reconstruction

**Apply `Themis` EHT modelling s/w to cm-VLBI observations (VLBA)**

HeidelbergU applied the EHT modelling software Themis to MOJAVE data, showing that it is feasible for cm-wavelength VLBI observations to benefit from this software.




In a different test, the separation of an astrometric binary system was extracted from simulated data, yielding astrometric results with a posterior probability distribution.

Grant number: 101093934

# T5.5 Modular PAF Processing toolkit (MPG)

- Effelsberg Direct Digitisation (EDD) system:
  - re-engineered to modular plug-in based system
  - **significantly** improved deployment system
    - time-to-start down **from ½ day to 20 min**!


Modules in the EDD:
capabilities (blue), technologies (orange), tooling (green)

- The DADAFLOW(*) library
  - graph-based (like Dask, but now w/ compiled code!)
  - C++20 goodies
  - type-safe and types carry meta data
  - multithreaded

*(*) https://gitlab.mpcdf.mpg.de/mpifr-bdg/psrdada_cpp/-/tree/devel/psrdada_cpp/dadaflow*

In task 5.5 the Effelsberg Direct Digitisation system was overhauled and re-engineered to a proper plugin-based system, and re-worked the deployment system, bringing the startup time down from half a day to 20 minutes!

Another feat is the publication of the DADAFLOW library: it's a C++20-based library, which provides graph-based execution, but now in compiled C++ code. The graph compute nodes can be TensorCore accelerated - the framework doesn't care.

# Milestones

| | description | Month | Verification |
|---|---|---|---|
| M19 | *Report on granularities suitable for efficient fringe fitting (T5.2)* | 12 | Internal report |
| M20 | *Interface & conversion software (CASA MS format and Python GPU DataFrame library cuDF) (T5.1)* | 12 | *S/W and documentation internal release - TeamValidation* |
| M21 | *Report on generic VLBI simulation Strategies (T5.3)* | 15 | *Internal report* |
| M22 | *Report on modular architecture (T5.5)* | 28 | report |
| **M23** | **Workflow demonstration and validation (T5.1)** | **36** | **Internal team validation** |
| **M24** | ***Assessment of Bayesian package links (T1,2,3) (T5.4)*** | **36** | ***Internal report & joint meeting*** |
| **M25** | ***Advocating package for early-adopters (T5.3)*** | **42** | ***Presentation to external users*** |

There are three milestones left

New science in Radio Astronomy: applying cutting-edge technology to enhance the entire data chain, from receiver to final output

Grant number: 101093934

# Milestones

| | description | Month | Verification |
|---|---|---|---|
| M19 | *Report on granularities suitable for efficient fringe fitting (T5.2)* | 12 | Internal report |
| M20 | *Interface & conversion software (CASA MS format and Python GPU DataFrame library cuDF) (T5.1)* | 12 | *S/W and documentation internal release - TeamValidation* |
| M21 | *Report on generic VLBI simulation Strategies (T5.3)* | 15 | *Internal report* |
| M22 | *Report on modular architecture (T5.5)* | 28 | report |
| **M23** | **Workflow demonstration and validation (T5.1)** | **36** | **Internal team validation** |
| **M24** | ***Assessment of Bayesian package links (T1,2,3) (T5.4)*** | **36** | ***Internal report & joint meeting*** |
| M25 | *Advocating package for early-adopters (T5.3)* | 42 | *Presentation to external users* |

two of them by February 2026, leaving about four months. Given the progress during RP2 we see no showstoppers for the teams passing these milestones in time.

# Milestones

| | description | Month | Verification |
|---|---|---|---|
| M19 | *Report on granularities suitable for efficient fringe fitting (T5.2)* | 12 | Internal report |
| M20 | *Interface & conversion software (CASA MS format and Python GPU DataFrame library cuDF) (T5.1)* | 12 | *S/W and documentation internal release - TeamValidation* |
| M21 | *Report on generic VLBI simulation Strategies (T5.3)* | 15 | *Internal report* |
| M22 | *Report on modular architecture (T5.5)* | 28 | report |
| M23 | Workflow demonstration and validation (T5.1) | 36 | Internal team validation |
| M24 | *Assessment of Bayesian package links (T1,2,3) (T5.4)* | 36 | *Internal report & joint meeting* |
| **M25** | ***Advocating package for early-adopters (T5.3)*** | **42** | ***Presentation to external users*** |

And the final milestone Is by August 2026. JIVE is responsible for this milestone but the person responsible for this has left JIVE. JIVE is trying to coordinate, figuring out which conferences to target, but it it will be the partners doing the presentations, with JIVE not having the manpower nor the expertise anymore.

# Deliverables

- *D5.1: Library of DASK-accelerated interface to analyse radio astronomy data formats (EPFL, Task 5.1, month 12) - **Complete***
- *D5.2: Implementation of a fringe fit algorithm in the DASK framework (JIV-ERIC, Task 5.2, month 48)*
- *D5.3: Prototype processing workflow functionality using software under DASK framework (ASTRON, Task 5.1, month 42)*
- *D5.4: Port of Bayesian package to make use of DASK-like HPC methods (HeidelbergU, Task 5.4, month 48)*
- *D5.5: Demonstrated usage of end-to-end simulation tool for single RI (JIV-ERIC, Task 5.3, month 48)*
- *D5.6: Full-Stokes dynamical imaging algorithm for time-variable VLBI sources (CSIC, Task 5.3, month 48)*
- *D5.7: Software/firmware repositories containing developed code and documentation (MPG, Task 5.5, month 48)*
- *D5.8: DASK accelerated data reading & GPU processing for radio astronomy tools (EPFL, Task 5.1, month 48)*

There are seven deliverables still to de done in WP5. We’re not going to sugarcoat it - the huge risk is this

# Deliverables

- *D5.1: Library of DASK-accelerated interface to analyse radio astronomy data formats (EPFL, Task 5.1, month 12) - **Complete***
- ***D5.2: Implementation of a fringe fit algorithm in the DASK framework (JIV-ERIC, Task 5.2, month 48)***
- *D5.3: Prototype processing workflow functionality using software under DASK framework (ASTRON, Task 5.1, month 42)*
- ***D5.4: Port of Bayesian package to make use of DASK-like HPC methods (HeidelbergU, Task 5.4, month 48)***
- ***D5.5: Demonstrated usage of end-to-end simulation tool for single RI (JIV-ERIC, Task 5.3, month 48)***
- ***D5.6: Full-Stokes dynamical imaging algorithm for time-variable VLBI sources (CSIC, Task 5.3, month 48)***
- ***D5.7: Software/firmware repositories containing developed code and documentation (MPG, Task 5.5, month 48)***
- ***D5.8: DASK accelerated data reading & GPU processing for radio astronomy tools (EPFL, Task 5.1, month 48)***

Only one of them Is NOT exactly at the end of the project

# Deliverables

- *D5.1: Library of DASK-accelerated interface to analyse radio astronomy data formats (EPFL, Task 5.1, month 12) - **Complete***
- ***D5.2: Implementation of a fringe fit algorithm in the DASK framework (JIV-ERIC, Task 5.2, month 48)***
- ***D5.3: Prototype processing workflow functionality using software under DASK framework (ASTRON, Task 5.1, month 42)***
- ***D5.4: Port of Bayesian package to make use of DASK-like HPC methods (HeidelbergU, Task 5.4, month 48)***
- *D5.5: Demonstrated usage of end-to-end simulation tool for single RI (JIV-ERIC, Task 5.3, month 48)*
- ***D5.6: Full-Stokes dynamical imaging algorithm for time-variable VLBI sources (CSIC, Task 5.3, month 48)***
- ***D5.7: Software/firmware repositories containing developed code and documentation (MPG, Task 5.5, month 48)***
- ***D5.8: DASK accelerated data reading & GPU processing for radio astronomy tools (EPFL, Task 5.1, month 48)***

All those deliverables require:

- documented s/w
- code published open access.

These six deliverables share the same requirements, so let's see where we stand

Grant number: 101093934

# Deliverables

- *D5.1: Library of DASK-accelerated interface to analyse radio astronomy data formats (EPFL, Task 5.1, month 12) - **Complete***
- ***D5.2: Implementation of a fringe fit algorithm in the DASK framework (JIV-ERIC, Task 5.2, month 48)***
- ***D5.3: Prototype processing workflow functionality using software under DASK framework (ASTRON, Task 5.1, month 42)***
- ***D5.4: Port of Bayesian package to make use of DASK-like HPC methods (HeidelbergU, Task 5.4, month 48)***
- ***D5.5: Demonstrated usage of end-to-end simulation tool for single RI (JIV-ERIC, Task 5.3, month 48)***
- ***D5.6: Full-Stokes dynamical imaging algorithm for time-variable VLBI sources (CSIC, Task 5.3, month 48)***
- ***D5.7: Software/firmware repositories containing developed code and documentation (MPG, Task 5.5, month 48)***
- ***D5.8: DASK accelerated data reading & GPU processing for radio astronomy tools (EPFL, Task 5.1, month 48)***

For all those a code base already exists

For those deliverables we know that a code base already exists

Grant number: 101093934

# Deliverables

- *D5.1: Library of DASK-accelerated interface to analyse radio astronomy data formats (EPFL, Task 5.1, month 12) -* ***Complete***
- ***D5.2: Implementation of a fringe fit algorithm in the DASK framework (JIV-ERIC, Task 5.2, month 48)***
- ***D5.3: Prototype processing workflow functionality using software under DASK framework (ASTRON, Task 5.1, month 42)***
- ***D5.4: Port of Bayesian package to make use of DASK-like HPC methods (HeidelbergU, Task 5.4, month 48)***
- ***D5.5: Demonstrated usage of end-to-end simulation tool for single RI (JIV-ERIC, Task 5.3, month 48)***
- ***D5.6: Full-Stokes dynamical imaging algorithm for time-variable VLBI sources (CSIC, Task 5.3, month 48)***
- ***D5.7: Software/firmware repositories containing developed code and documentation (MPG, Task 5.5, month 48)***
- ***D5.8: DASK accelerated data reading & GPU processing for radio astronomy tools (EPFL, Task 5.1, month 48)***

The code base is already proven by passing milestones (M20, M22)

For those this was demonstrated through passing milestones

# Deliverables

- *D5.1: Library of DASK-accelerated interface to analyse radio astronomy data formats (EPFL, Task 5.1, month 12) - **Complete***
- ***D5.2: Implementation of a fringe fit algorithm in the DASK framework (JIV-ERIC, Task 5.2, month 48)***
- ***D5.3: Prototype processing workflow functionality using software under DASK framework (ASTRON, Task 5.1, month 42)***
- ***D5.4: Port of Bayesian package to make use of DASK-like HPC methods (HeidelbergU, Task 5.4, month 48)***
- ***D5.5: Demonstrated usage of end-to-end simulation tool for single RI (JIV-ERIC, Task 5.3, month 48)***
- ***D5.6: Full-Stokes dynamical imaging algorithm for time-variable VLBI sources (CSIC, Task 5.3, month 48)***
- *D5.7: Software/firmware repositories containing developed code and documentation (MPG, Task 5.5, month 48)*
- *D5.8: DASK accelerated data reading & GPU processing for radio astronomy tools (EPFL, Task 5.1, month 48)*

The code was already applied to simulated data (see RP2 report)

this we have seen being applied to simulated data

Grant number: 101093934

# Deliverables

- *D5.1: Library of DASK-accelerated interface to analyse radio astronomy data formats (EPFL, Task 5.1, month 12) - **Complete***
- ***D5.2: Implementation of a fringe fit algorithm in the DASK framework (JIV-ERIC, Task 5.2, month 48)***
- ***D5.3: Prototype processing workflow functionality using software under DASK framework (ASTRON, Task 5.1, month 42)***
- ***D5.4: Port of Bayesian package to make use of DASK-like HPC methods (HeidelbergU, Task 5.4, month 48)***
- ***D5.5: Demonstrated usage of end-to-end simulation tool for single RI (JIV-ERIC, Task 5.3, month 48)***
- *D5.6: Full-Stokes dynamical imaging algorithm for time-variable VLBI sources (CSIC, Task 5.3, month 48)*
- *D5.7: Software/firmware repositories containing developed code and documentation (MPG, Task 5.5, month 48)*
- *D5.8: DASK accelerated data reading & GPU processing for radio astronomy tools (EPFL, Task 5.1, month 48)*

- Basic version of the algorithm already exists
- Working on calibration framework
- Advanced algorithm:
  - currently being optimized in other context (CASA)
  - porting new algorithm to Dask = least of anyone's worries (infrastructure is the *hard* part, mostly done)

a basic version of this algorithm already exists, the focus is now on helping design and implement the middleware that allows the extracted calibration information to be stored and applied. In a different context the algorithm is being investigated to be more efficient. Translating such a new approach to Dask is easy once the infrastructure exists (and it mostly does).

# Deliverables

- *D5.1: Library of DASK-accelerated interface to analyse radio astronomy data formats (EPFL, Task 5.1, month 12) - **Complete***
- *D5.2: Implementation of a fringe fit algorithm in the DASK framework (JIV-ERIC, Task 5.2, month 48)*
- ***D5.3: Prototype processing workflow functionality using software under DASK framework (ASTRON, Task 5.1, month 42)***
- ***D5.4: Port of Bayesian package to make use of DASK-like HPC methods (HeidelbergU, Task 5.4, month 48)***
- ***D5.5: Demonstrated usage of end-to-end simulation tool for single RI (JIV-ERIC, Task 5.3, month 48)***
- *D5.6: Full-Stokes dynamical imaging algorithm for time-variable VLBI sources (CSIC, Task 5.3, month 48)*
- *D5.7: Software/firmware repositories containing developed code and documentation (MPG, Task 5.5, month 48)*
- *D5.8: DASK accelerated data reading & GPU processing for radio astronomy tools (EPFL, Task 5.1, month 48)*

A successful migration from MPI-based to DASK-based processing exists (RP2)

For this one we have seen that a workflow was already successfully ported from the Message-Passing-Interface to a Dask-based implementation with very interesting results.

# Deliverables

- *D5.1: Library of DASK-accelerated interface to analyse radio astronomy data formats (EPFL, Task 5.1, month 12) - **Complete***
- *D5.2: Implementation of a fringe fit algorithm in the DASK framework (JIV-ERIC, Task 5.2, month 48)*
- *D5.3: Prototype processing workflow functionality using software under DASK framework (ASTRON, Task 5.1, month 42)*
- ***D5.4: Port of Bayesian package to make use of DASK-like HPC methods (HeidelbergU, Task 5.4, month 48)***
- ***D5.5: Demonstrated usage of end-to-end simulation tool for single RI (JIV-ERIC, Task 5.3, month 48)***
- *D5.6: Full-Stokes dynamical imaging algorithm for time-variable VLBI sources (CSIC, Task 5.3, month 48)*
- *D5.7: Software/firmware repositories containing developed code and documentation (MPG, Task 5.5, month 48)*
- *D5.8: DASK accelerated data reading & GPU processing for radio astronomy tools (EPFL, Task 5.1, month 48)*

ULeiden's implementation using TensorCore is *superfast*, want it into THEMIS!

Which leaves this one in the group of software deliverables. The team would very much like to integrate the *extremely* promising results from the astrometry modelling package from Leiden University, based on TensorCore enabled processing into the Themis modelling package.

# Deliverables

- *D5.1: Library of DASK-accelerated interface to analyse radio astronomy data formats (EPFL, Task 5.1, month 12) - **Complete***
- *D5.2: Implementation of a fringe fit algorithm in the DASK framework (JIV-ERIC, Task 5.2, month 48)*
- *D5.3: Prototype processing workflow functionality using software under DASK framework (ASTRON, Task 5.1, month 42)*
- ***D5.4: Port of Bayesian package to make use of DASK-like HPC methods (HeidelbergU, Task 5.4, month 48)***
- ***D5.5: Demonstrated usage of end-to-end simulation tool for single RI (JIV-ERIC, Task 5.3, month 48)***
- *D5.6: Full-Stokes dynamical imaging algorithm for time-variable VLBI sources (CSIC, Task 5.3, month 48)*
- *D5.7: Software/firmware repositories containing developed code and documentation (MPG, Task 5.5, month 48)*
- *D5.8: DASK accelerated data reading & GPU processing for radio astronomy tools (EPFL, Task 5.1, month 48)*

ULeiden's implementation using TensorCore is *superfast*, want it into THEMIS!

- TensorCores $\perp$ Dask
- RoI of integrating TensorCores $\gg$ RoI porting to DASK

But it should be noted that Enabling Tensor-Core computations is, to some extent, orthogonal to Dask - and it could imply that for this **particular** application that the return on investment of enabling TensorCore-based algorithms is a lot bigger than porting it to the Dask framework.

# Outlook (post-RADIOBLOCKS)

- Many more algorithms could/should be ported to Dask
- Invest effort into investigating:
  - can algorithm(s) be written in matrix-matrix formalism?
  - can algorithm(s) produce results with less internal precision?

**Running algorithm on TensorCore GPUs w/ FP16, FP8 increases performance and energy efficiency by orders of magnitude!**

- See WP4 D4.4 - Comparing efficiencies in `J / bit`
  - WP5 building blocks are guidelines for performance comparison services

For the outlook we see opportunities in these directions:
- Dask is really scalable and many algorithms would benefit
- Investment into trying to rewrite algorithms to matrix-matrix formalism could enable benefiting from the extreme compute power of the TensorCore hardware
- It should be investigated if algorithms can provide the same output quality with lower internal precision: the TensorCore processors are orders of magnitude faster with 8 or 16 bit floating point arithmetic
- In WP4 the radioblocks are compared using the unit joule per (data) bit for fair comparison. WP5 has demonstrated that there also need to be comparison services to measure improvement

T5.3 All hands in Granada 2025

And with that we would like to thank you for your attention