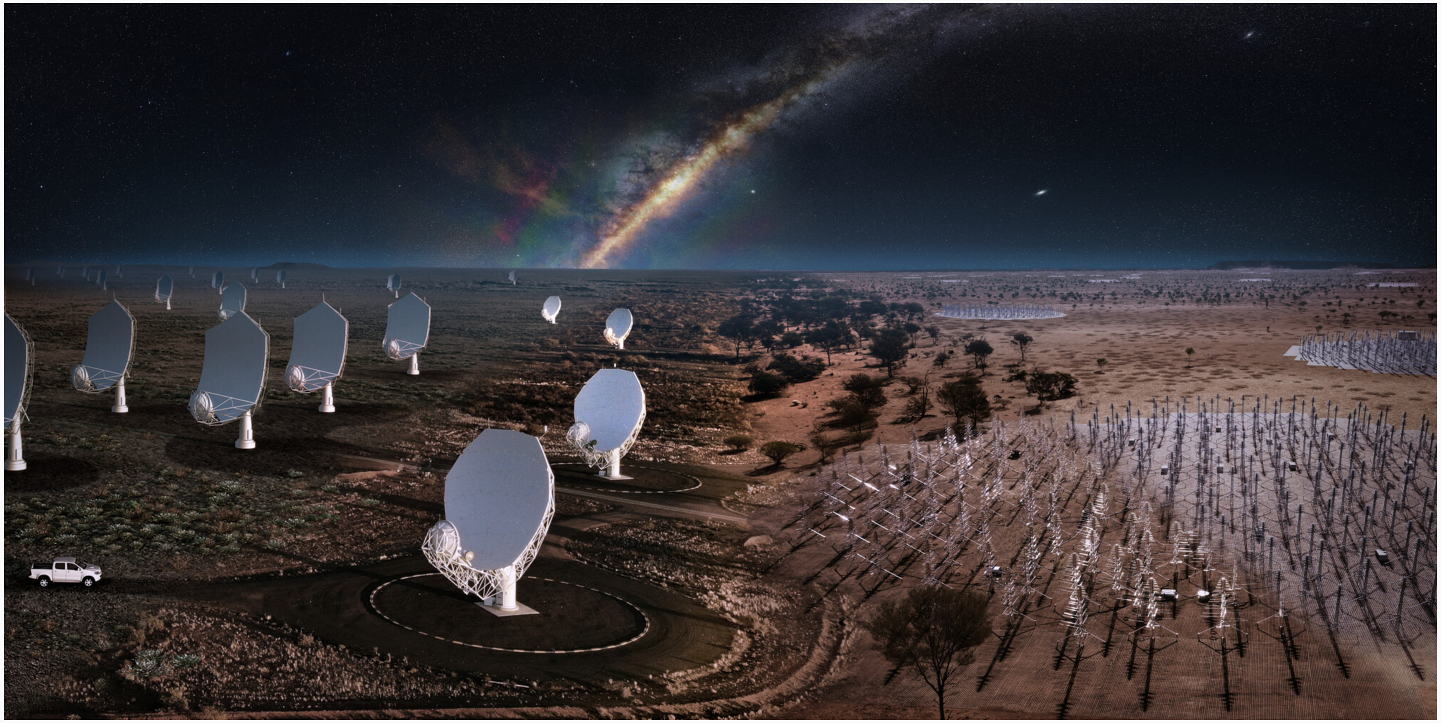


Distributed SKA science-driven workflows at extreme scales : lessons from SKA precursors/pathfinders and next SKA challenges

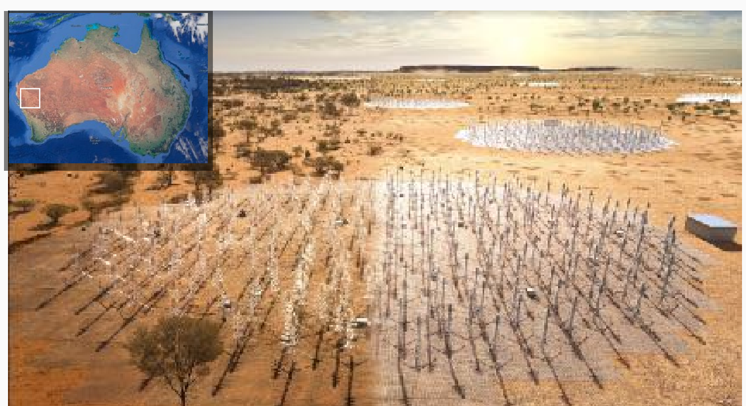
Damien Gratadour

SKAO: the largest (radio-)telescope

1 observatory: 2 telescopes (**Australia & South Africa**)
+ Headquarters (**U.K.**)

A giant software observatory, streaming data globally

50 MHz — 350 MHz — 15 GHz →

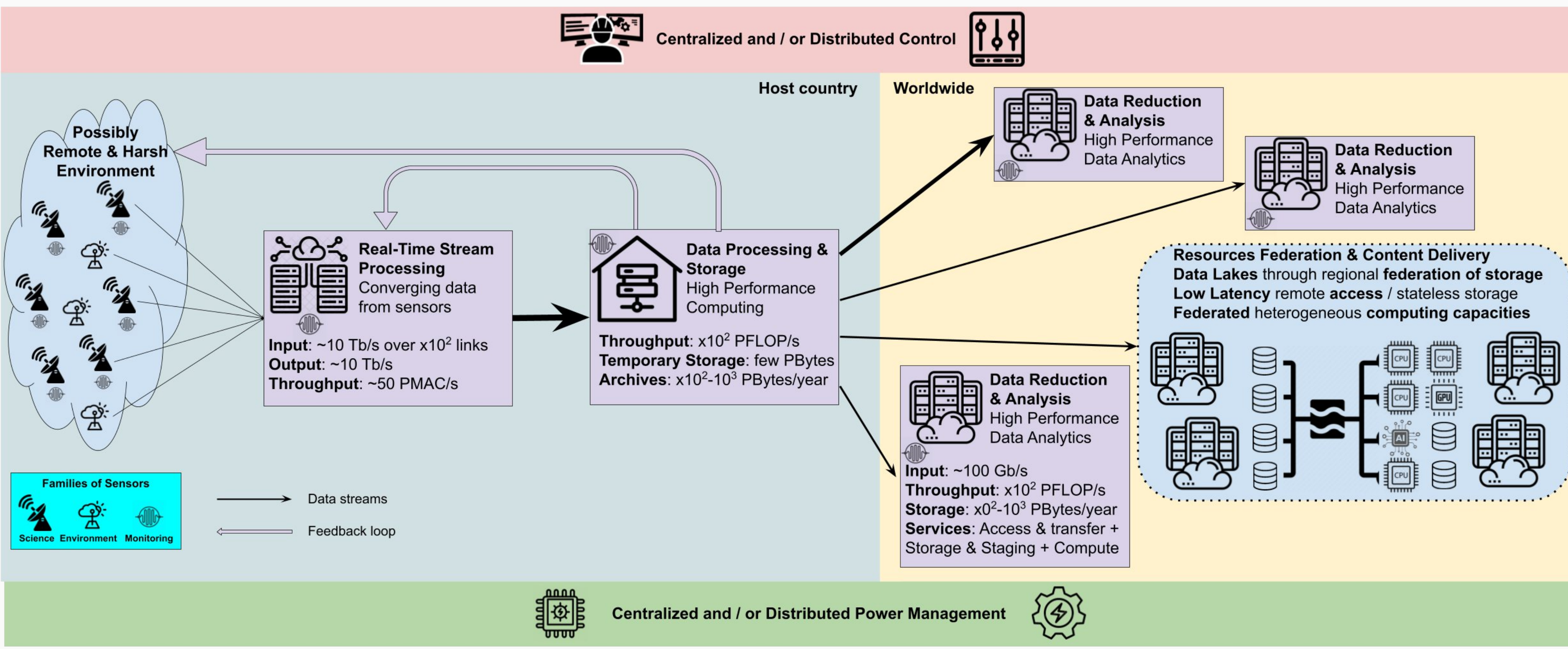


SKAO: unraveling the unknown

Cyber Continuum for SKA

Hierarchical architecture: system of systems

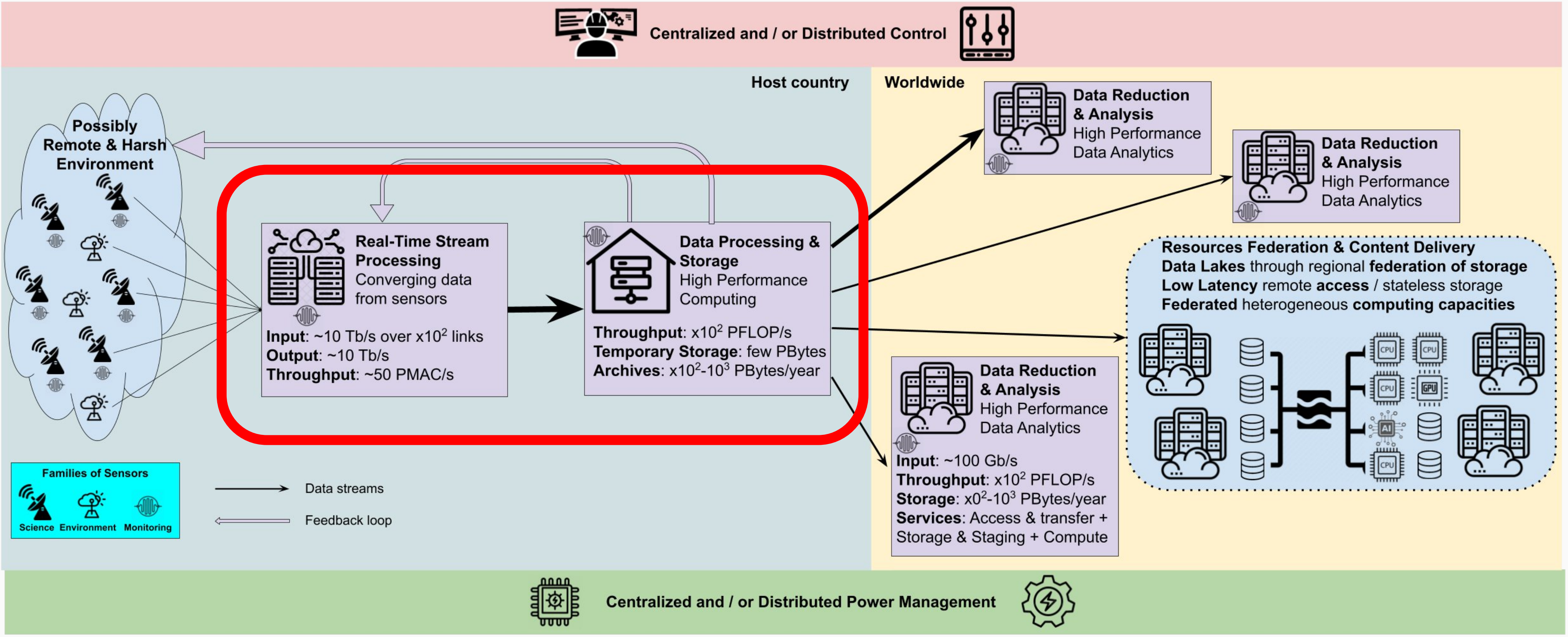
- Large amount of distributed & heterogeneous sensors
- Real-time stream engine for raw data convergence
- State-of-the-art datacenter for processing, storage and distribution
- Distributed network of national HPC facilities for content delivery to the users



Edge-to-HPC computing for SKA

Collect, Converge and Reduce data streams from distributed sensors

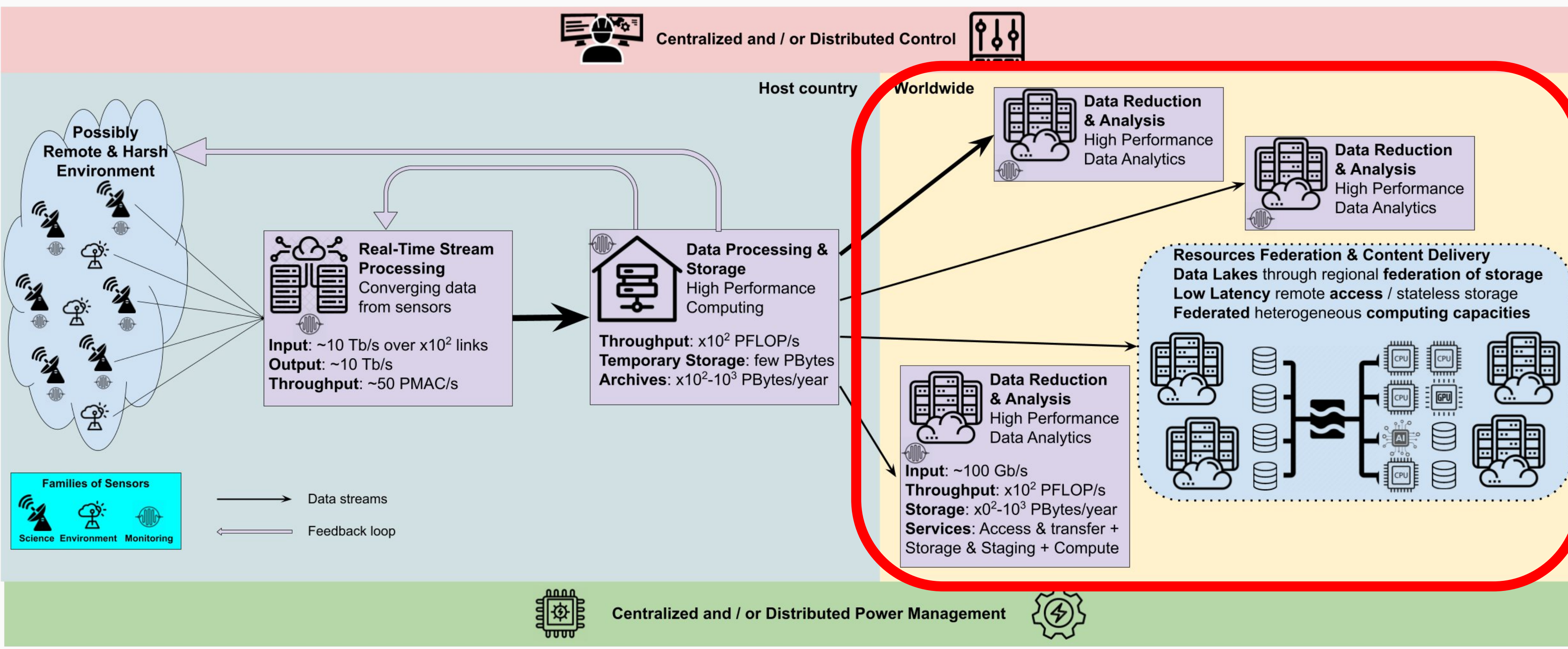
- In situ & Online data processing with centralized HPC systems
- Reduce continuous 10 Tb/s stream to 350 PB/year of data products
- Affordable / Adaptable / Frugal / Resilient
- Duplicated in two host countries (with centralized control in UK)



HPC-to-Cloud computing for SKA

Federate resources to analyze distributed data

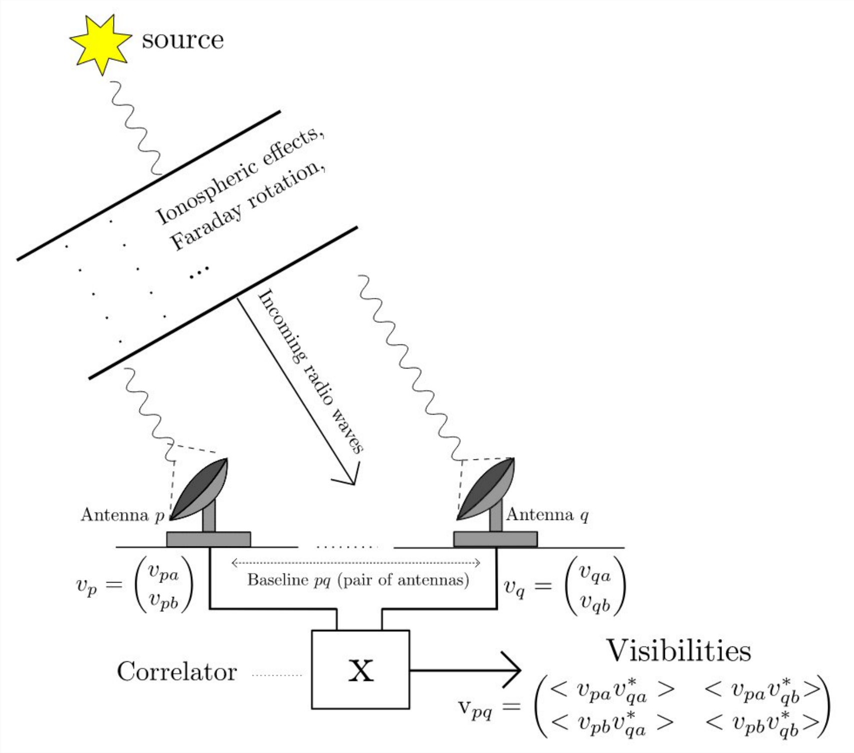
- Rely on external resources (regional centers), possibly at continental level
- Federate: compute, data logistics, storage, wide-area workflows
- Increasing use of AI for many science programs
- Access patterns, provenance, resources accounting, power management



SKA computing challenges: imaging

Typical acquisition and processing pipeline

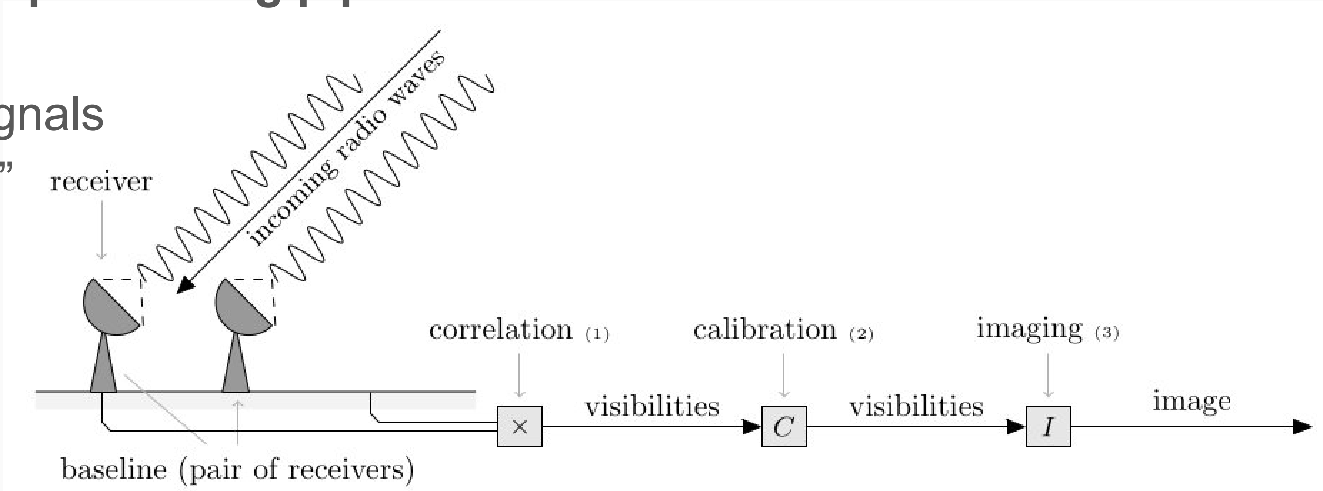
- Combine pairs of signals to create "visibilities"

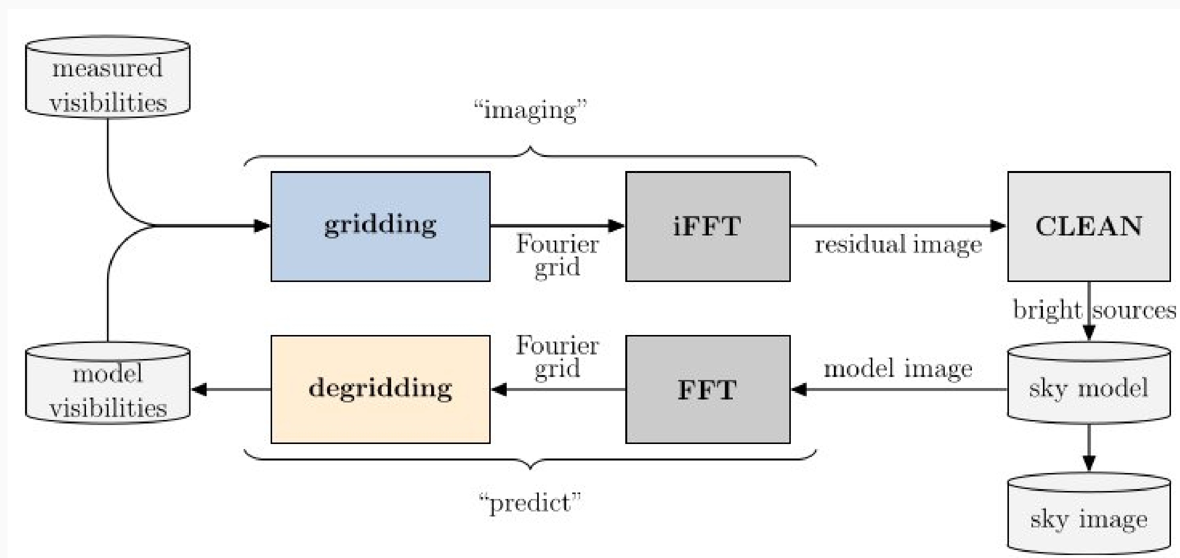


SKA computing challenges: imaging

Typical acquisition and processing pipeline

- Combine pairs of signals to create "visibilities"
- Iterative algorithm to perform image synthesis incl. calibrations + deconvolution

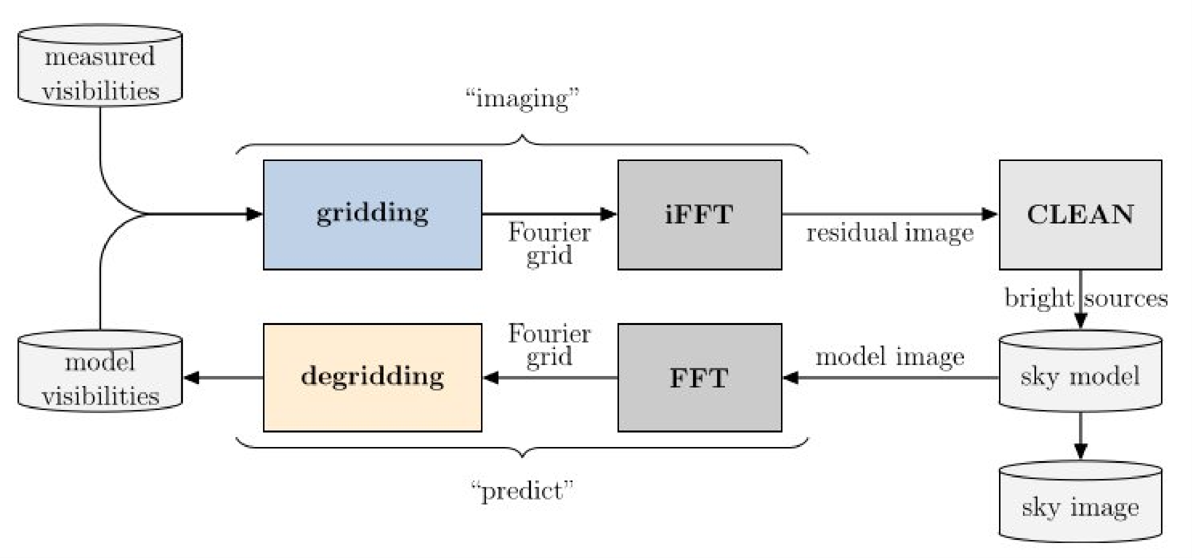


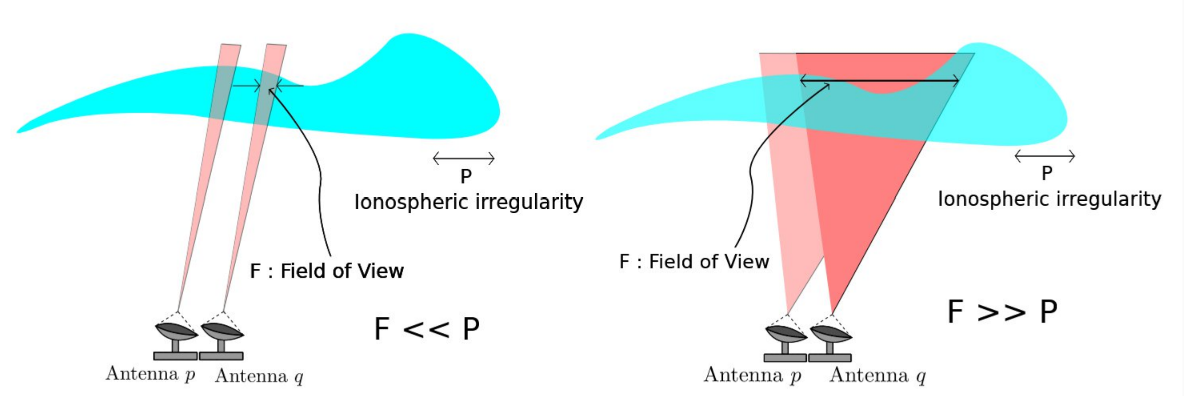


SKA computing challenges: imaging

Typical acquisition and processing pipeline

- Combine pairs of signals to create “visibilities”
- Iterative algorithm to perform image synthesis incl. calibrations + deconvolution
- Multiple targets (planets galaxies, cosmic dawn): variable obs. scenario = variable workload

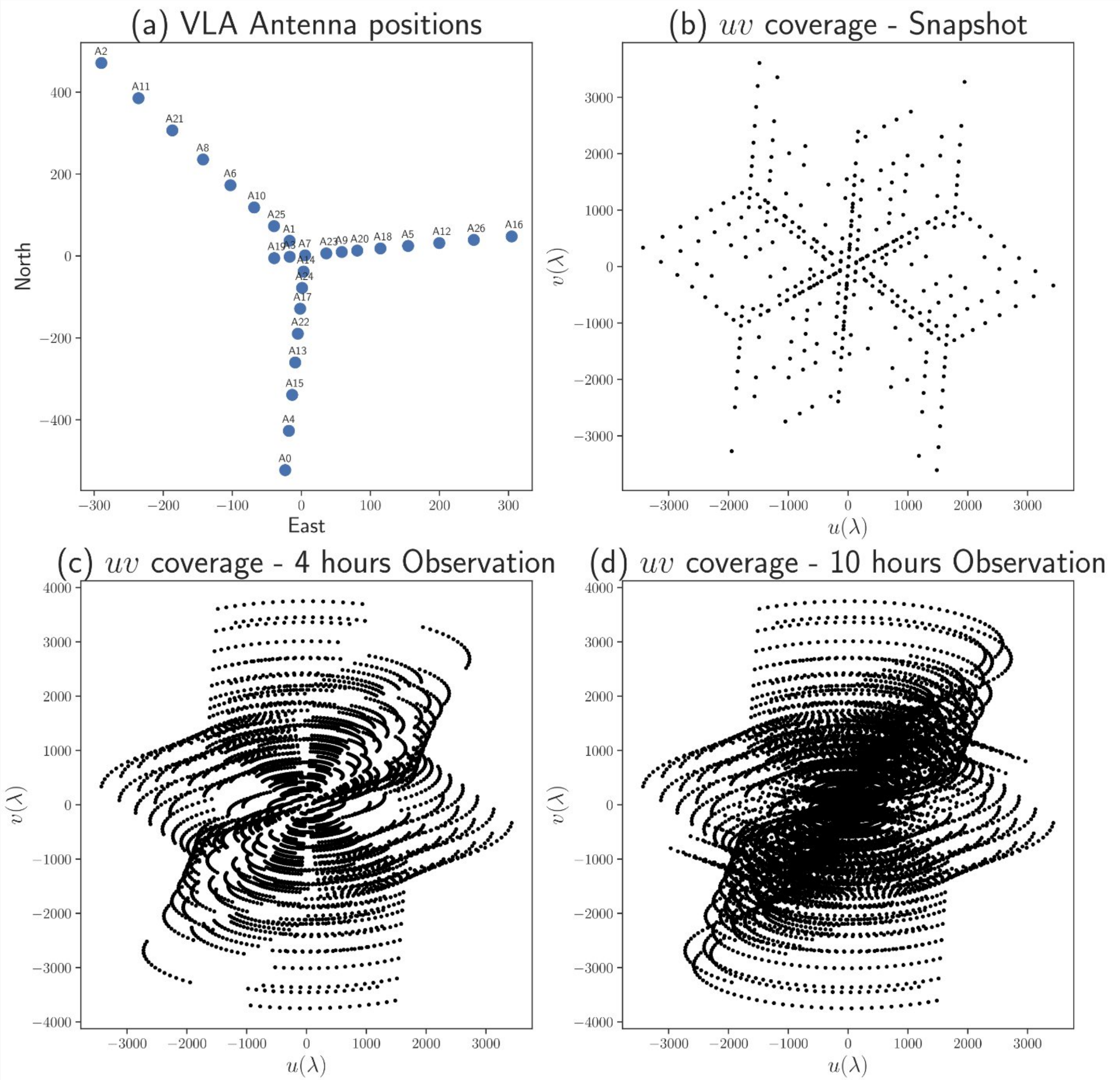




SKA computing challenges: imaging

Imaging:

- Earth rotation synthesis: increasing frequency plane coverage over extended observation time
- Both iterative (deconvolution) and incremental (observing time)
- Trade-off between online processing (buffer based) and batch processing (storage)



Challenges across the continuum

Facilities operations

- Multiscale **system of systems**
- **Intercontinental control** strategies
  - Including "owned" and "shared" facilities
- **x10 years typical lifetime**
  - Continuous integration of emerging & non-conventional technologies
  - Preserve operations

Facilities management

- **Limited power envelope**
  - Access to power grid

- **Cost containment**
  - Mostly relying on taxpayers money
- **Optimized operations**
  - Dynamical cyberinfrastructure, including reconfigurable HPC

Existing implementations (I)

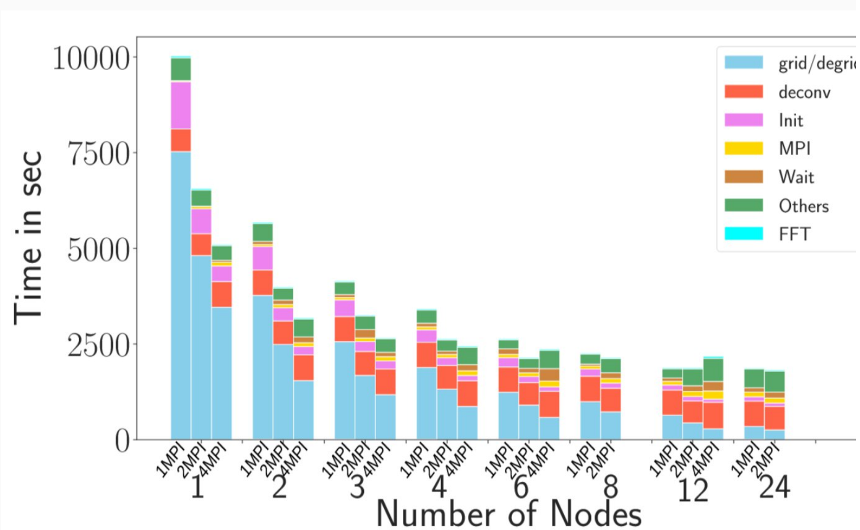
KillMS + DDfacet

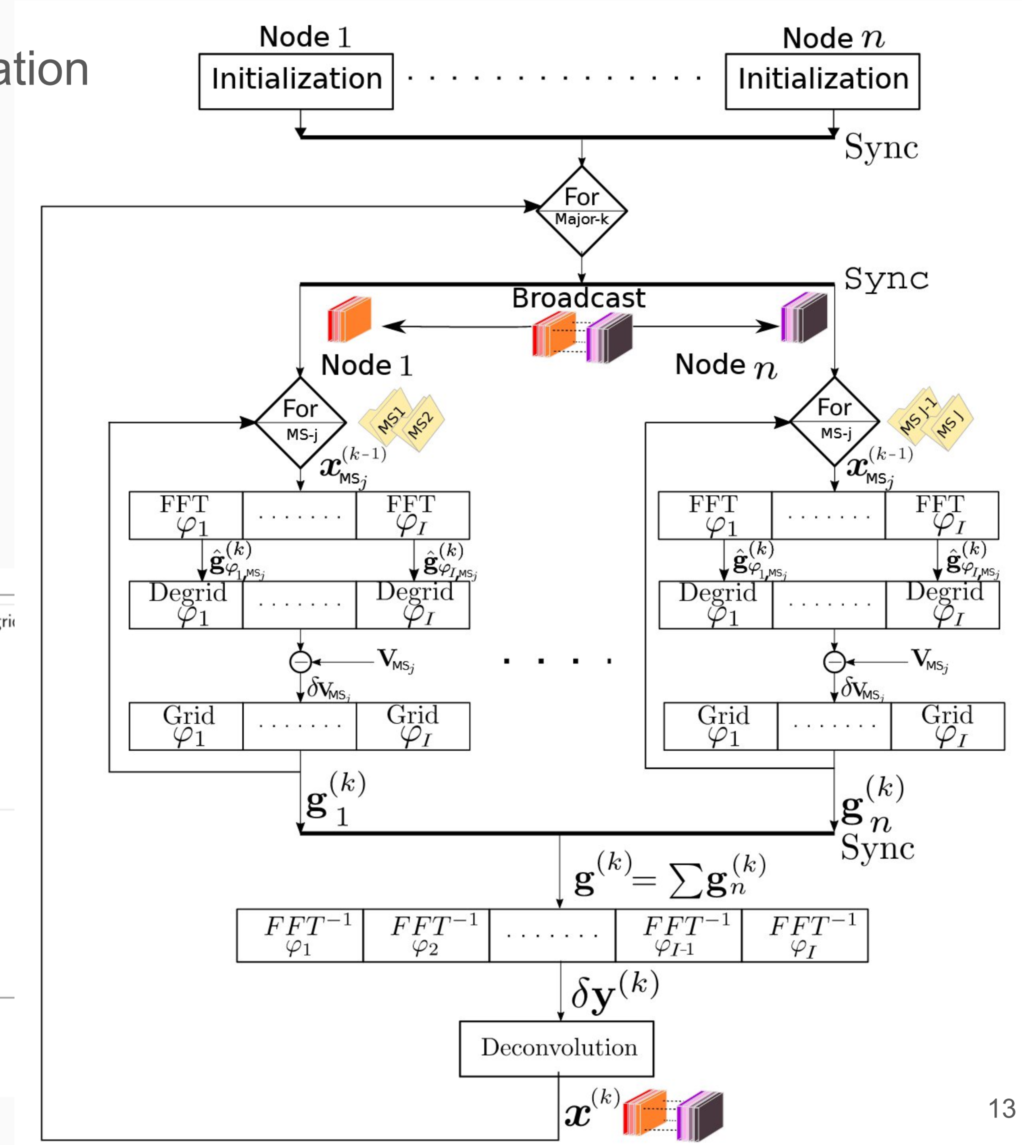
- Developed at OdP (in collab. with others, incl. Atos/Eviden)
- Proven solution: e.g. LoTSS survey / MeerKAT
- Direction dependent calibrations & deconvolution. Probably the most advanced and accurate pipeline
- Parallelization effort: static scheduling, recent results from N. Monnier PhD (collab. OdP + L2S + Atos/Eviden)
- McMn parallelization, up to 24 nodes:
  - meeting specs for “small” LOFAR dataset: 1h observation reduced in less than 1h
- **Needs to scale up**:
  - MeerKAT extension & LOFAR VLBI & SKA-Low/Mid require
    - x40 in number of visibilities
    - x25 in final image size
    - Significant impact
- Today: pure Python software, incl. ConcurrentFutures & MPI for parallelization. https://github.com/saopicc/DDFacet

Existing implementations (I)

KillMS + DDfacet: current parallelization strategy

- From N. Monnier PhD
- Relying on static scheduling & static dispatch of “measurement sets”
- Deconvolution on single node / single core





Existing implementations (II)

Stimela + Dask + Kubernauts

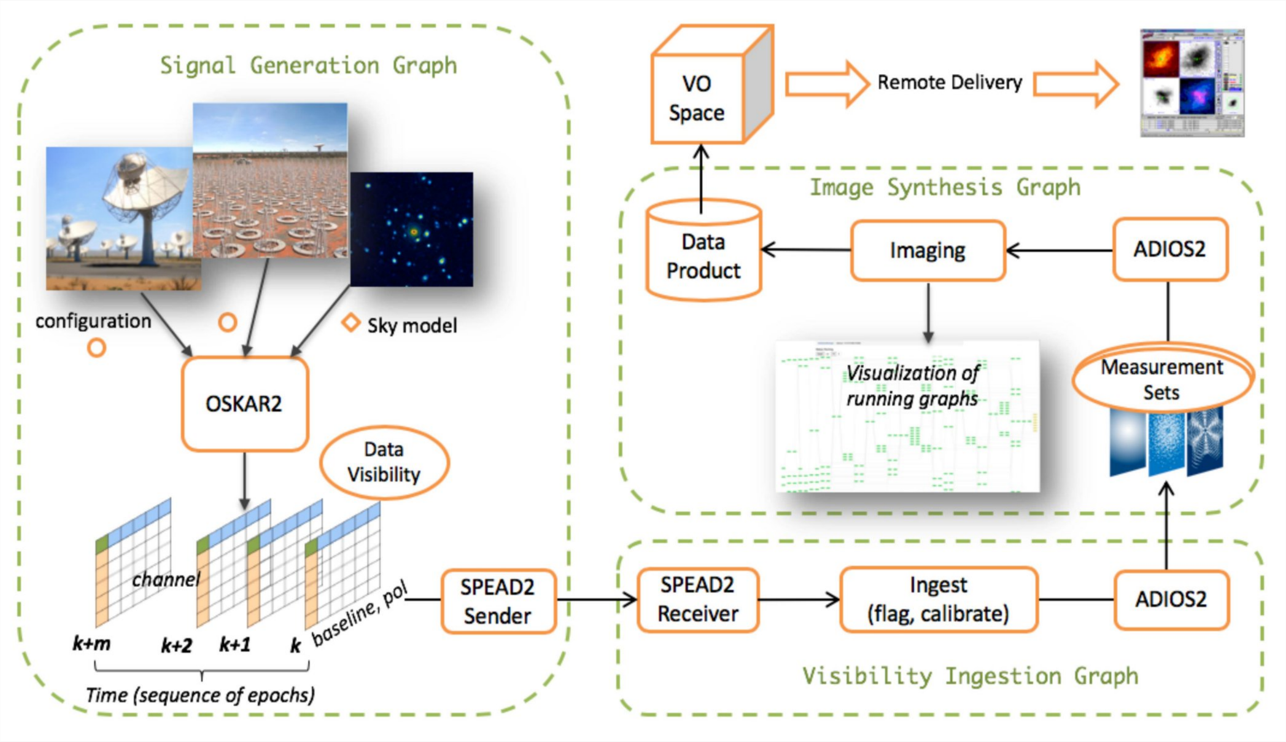
- Developed at SARAO (South Africa, MeerKAT team: RATT)
- Proven solution: new MeerKAT baseline solution
- Designed to run in the cloud: deployability + portability as key requirements
- Still missing features (direction dependent effects), but very versatile (concept of "pipeline framework" to be adaptable to varying observational scenario)
- **Not optimized for performance**:
  - Relying on Dask + "Cloud toolbox" (e.g. Kubernetes, tested on AWS, etc ...)
  - Significant work done to handle (& distribute) "measurements sets": dask-ms
    - https://github.com/ratt-ru/dask-ms
  - South African team very active & happy to collaborate

QuartiCal (J. Kenyon) https://github.com/ratt-ru/QuartiCal: calibration suite
pfb-clean (L. Bester) https://github.com/ratt-ru/pfb-clean: imager
tricolour (B. Hugo, S. Perkins) https://github.com/ratt-ru/tricolour: flagger
xova (S. Perkins) https://github.com/ratt-ru/xova: visibility data averaging, including BDA
crystalball (S. Perkins) https://github.com/caracal-pipeline/crystalball: DFT-based model predict
shadeMS (O. Smirnov, I. Heywood) https://github.com/ratt-ru/shadeMS: plotting & visualization

Towards a SKA demonstrator

Demonstrating the SKA online workflow at full scale

- Addressing Edge-to-HPC component
- Going beyond Gordon Bell finalist study from Wang et al. 2020
- From data generation to Scientific Data Products
- Including **frugality and resilience** as sustainability indicators



Supporting initiatives

ECLAT: ***Laboratoire Commun*** **CNRS - Inria – Eviden - Obs. Côte d'Azur & Obs. Paris** (head: D. Gratadour)

- Support structure for French contribution to SKA
- In kind contributions from partners, incl. **INSU, INS2I, INSIS**, multiple **Inria** teams together with **Atos R&D** and business dev.
- Truly multi-disciplinary and trans-sectoral collaboration

That's it for today !