Efficient Cloud Utilisation on Gaia and PLATO

Patrick Burgess

IRIS Collaboration Meeting Durham July 2nd 2025

PLATO

PLAnetary Transits and Oscillations of stars

- Space based observatory primarily tasked with detection of terrestrial exoplanets around bright, solar type stars.

- PLATO Spacecraft has 26 cameras covering 2132 square degrees in an overlapping field of view.

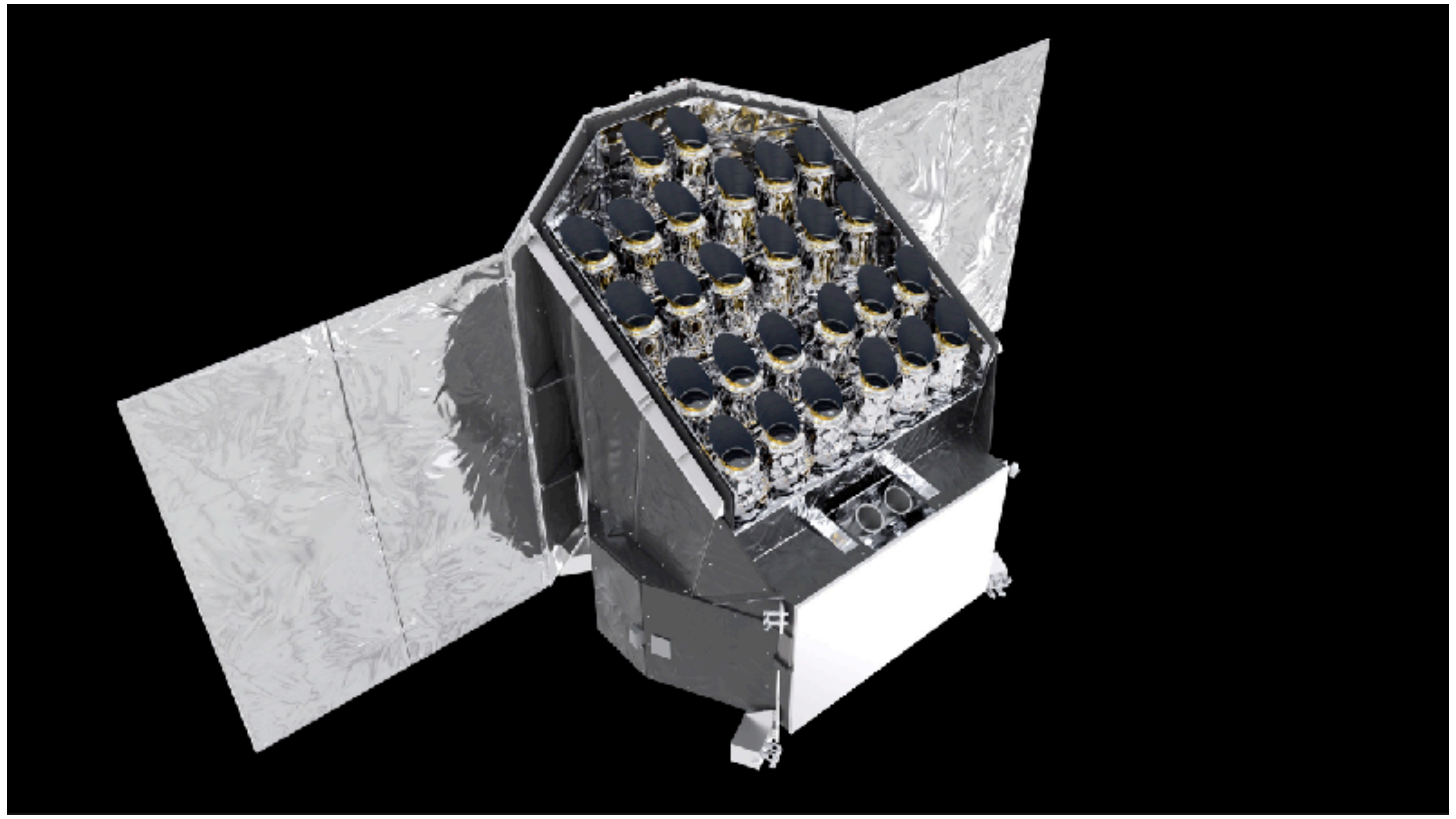
- Stares at selected target stars for long duration - initially 2 years - rotating 90 degrees every 3 months to avoid the sun.

- Over 200,000 target stars.

- Observations are imagettes (6x6 pixels) or light curves with cadences of 25s, 50s or 600s depending on the target.

- Downloading 435Gb per day at 72Mbps.

- Publishing results every 3 months during science operations.



https://www.esa.int/Science_Exploration/Space_Science/Plato

PLATO Mission Timeline

- To be launched at the end of 2026 and start science operations in early 2027 after commissioning.
- With a nominal science mission duration of 4 years (finish early 2031) extendable to 8.5 years (finish mid 2035).
- Cambridge responsible for Exoplanet Analysis System (EAS): TransitPipe, PlanetPipe,…
    - Detection and characterisation of planet signals within calibrated input data.
- Series of test campaigns leading up to launch. Execution performance, science performance, integration of data centres.
- Software development ongoing. Requirements and schedule still subject to a high degree of change.

PLATO IRIS Allocation

- Requested resources for PLATO full operational allocation unlikely to be possible at a single site.
- Planning to be able to run different processes in different locations: Transit detection at one location, Planet characterisation at another location.
- Currently main resources at Cambridge on Arcus cloud with smaller allocations at Imperial and STFC cloud to test distributed execution.
- Revising current allocation requirements to fit into available resources. Scaling down tests where this is possible with low impact, combing tests to run in parallel etc.

Location	vCPU	Disk (TB)	GPU
Cambridge	3552	400	4
Imperial	288	5.4	1
STFC	200	5.6	1
24/25 RSAP	5000	400	5
25/26 RSAP	10000	400	20
25/26 Revised	4000*	400	6*

* Still require 10k vCPU and 20 GPU temporarily in order to run scale tests prior to Launch Readiness Review in Q3 2026

PLATO Deployment

- PLATO operations architecture will make use of various IRIS resources at geographically separate sites. Deployment automated as far as possible with Ansible code.
    - PLATO EAS processing receives data from a central hub based in Germany and delivers results back to the central hub.
    - EAS software deployed on Kubernetes so once that environment is configured, location is not significant.
- All sites are using similar technology: Openstack cloud, Rocky or Redhat. Simplifies use of multiple sites.
    - However there are differences in software versions, security policies, upgrade timetables. Nothing serious so far.
    - Some increase in workload for systems administration but try to minimise this.
- Aim to have a flexible Ansible deployment to different sites so location is transparent as far as possible. Still under development.

Exoplanet detection

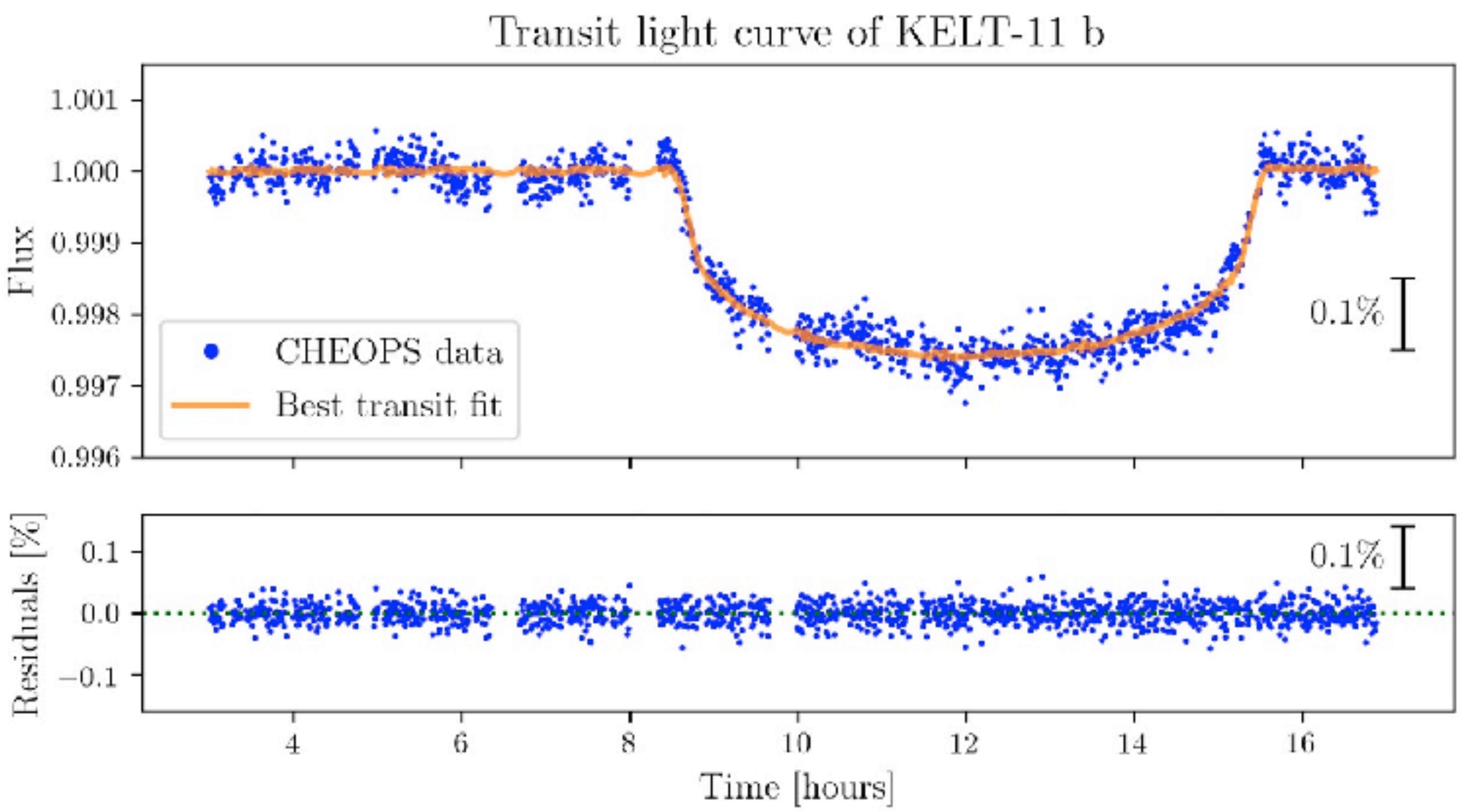
- Transit Photometry: continuous observation of a star to detect reduction in light when a planet passes between the observer and the star.

- Other methods: Radial velocity, astrometry,…

- Frequently used methods for transit photometry:

  - Box-fitting Least Squares algorithm (BLS)

  - Transit Least Squares (TLS) algorithm.



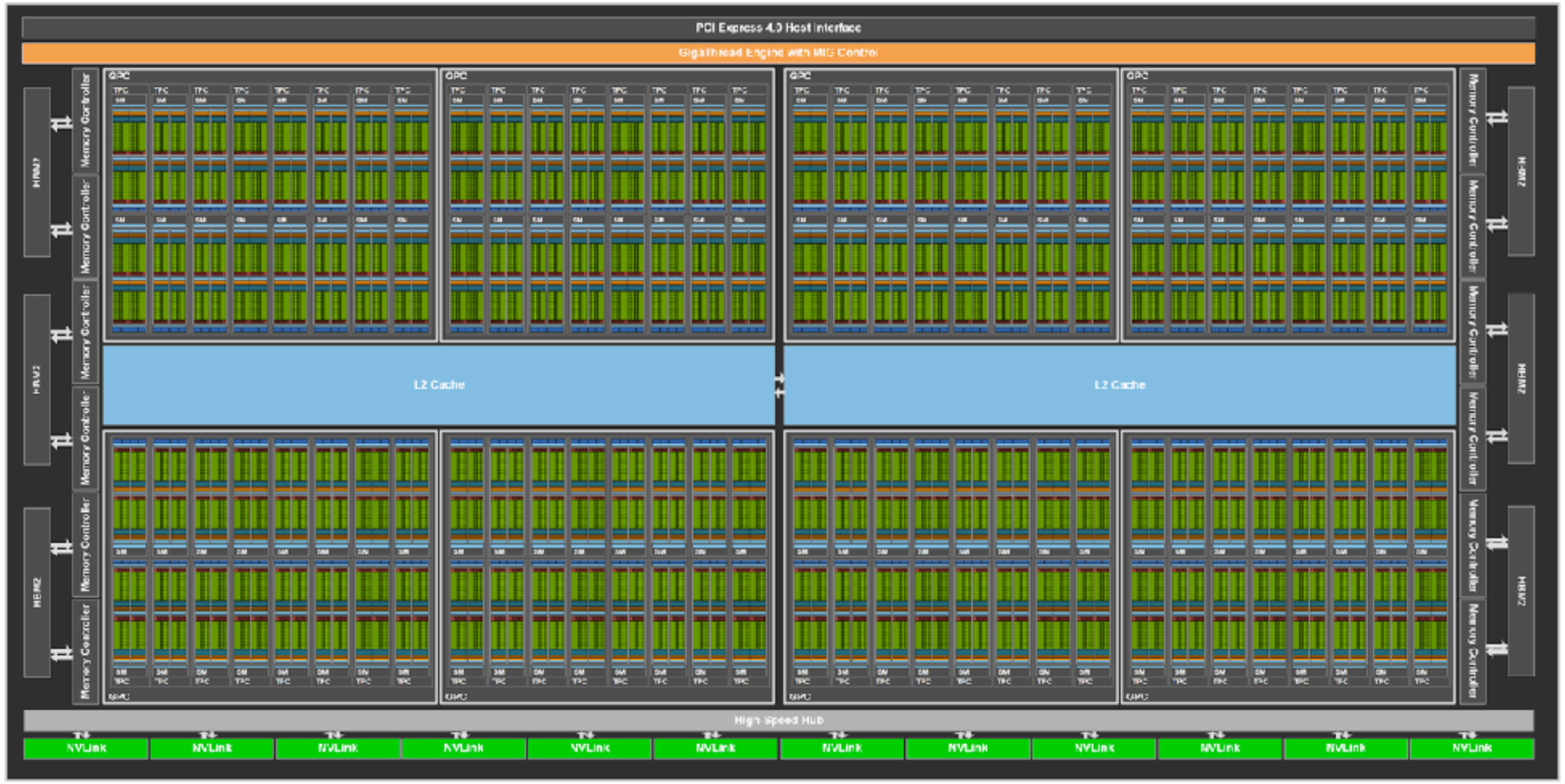
CETRA

Cambridge Exoplanet Transit Recovery Algorithm

- A fast, sensitive exoplanet transit detection algorithm implemented for NVIDIA GPUs (https://arxiv.org/abs/2503.20875)
- Detect possible transits and characterise the periodicity of the signal.
- Implemented using NVIDIA CUDA framework in C by Leigh Smith.
- Provides a Python API supporting straightforward integration into Python software or notebooks.
- Used for EAS TransitPipe transit detection and characterisation.
- Open source : https://github.com/leigh2/cetra

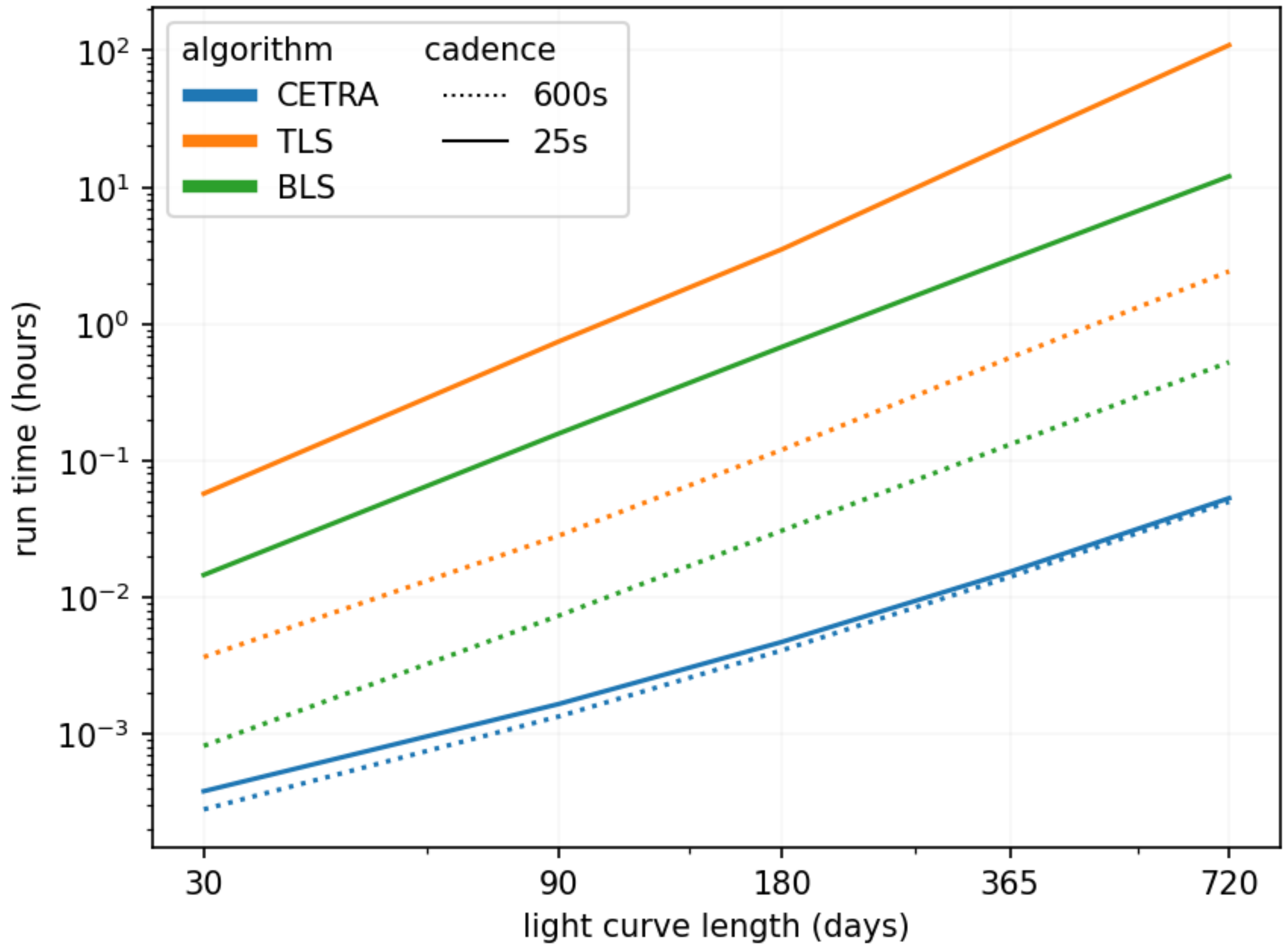
CETRA Overview

- For a light curve e.g. 2,552,880 samples for 2 years observing at 25s cadence.
- Processing stages of CETRA transit detection:
  - Resample to regularise and eliminate gaps.
  - Linear search to detect likely transits.
  - Periodic signal search based on detected transits.
- Stages utilise the GPU to parallelise and improve overall performance.



CETRA Performance

- Initial PLATO stare is 2 years - ~720 days.
- 25s cadence used for the brightest PLATO targets (P1, P2, P4).
- Significant improvements in overall processing time compared to other CPU based existing methods.



https://arxiv.org/abs/2503.20875

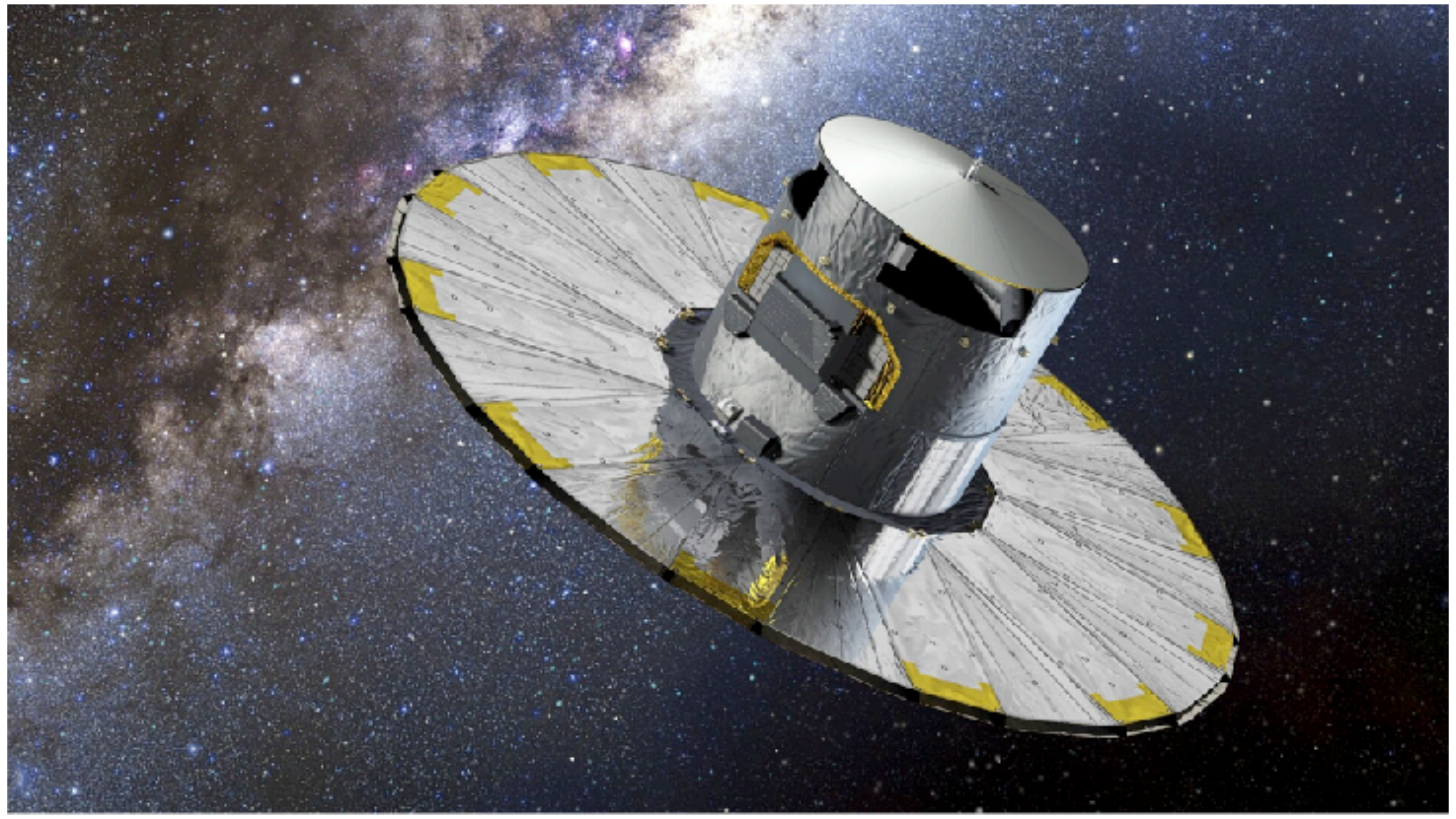
Kubernetes GPU Deployment

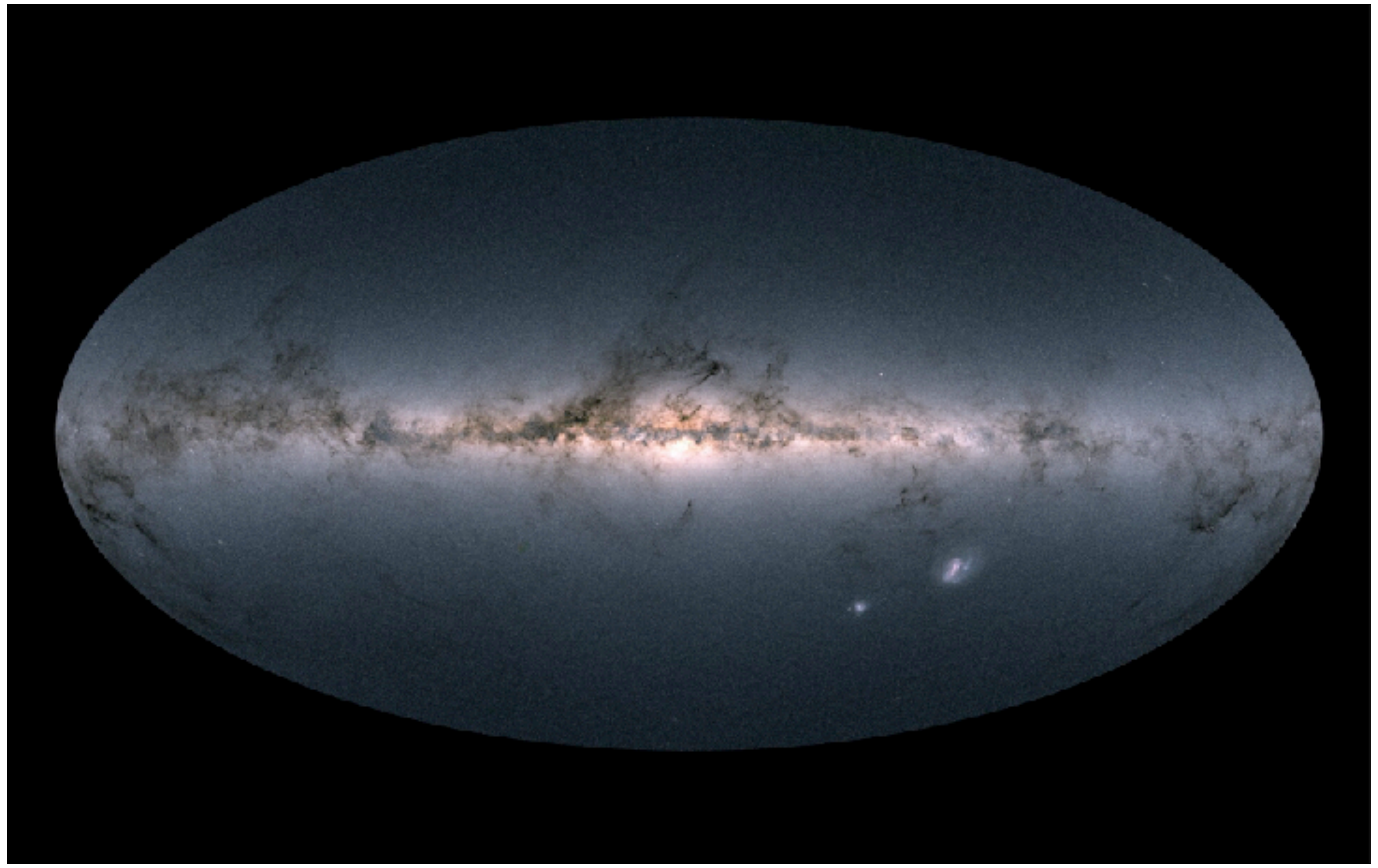
- Create instance(s) in Openstack with GPU enabled flavour.
- Install NVIDIA drivers on GPU enabled instances.
- Enable NVIDIA container support on GPU enabled instances.
- Install NVIDIA GPU operator using helm.
- All part of Ansible deployment code.
- Run test container (yaml to the right)

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: gpu-pod
spec:
  runtimeClassName: nvidia
  restartPolicy: Never
  containers:
    - name: cuda-container
      image: nvcr.io/nvidia/k8s/cuda-sample:vectoradd-cuda12.5.0
      resources:
        limits:
          nvidia.com/gpu: 4
```

Gaia

- Launched in December 2013.
- Build the largest and most precise three dimensional map of our galaxy.
- Three trillion observations of two billion stars and other objects.
- Continuous scanning and taking observations of detected sources brighter than about magnitude 20.





https://www.esa.int/Science_Exploration/Space_Science/Gaia

Gaia Mission Timeline

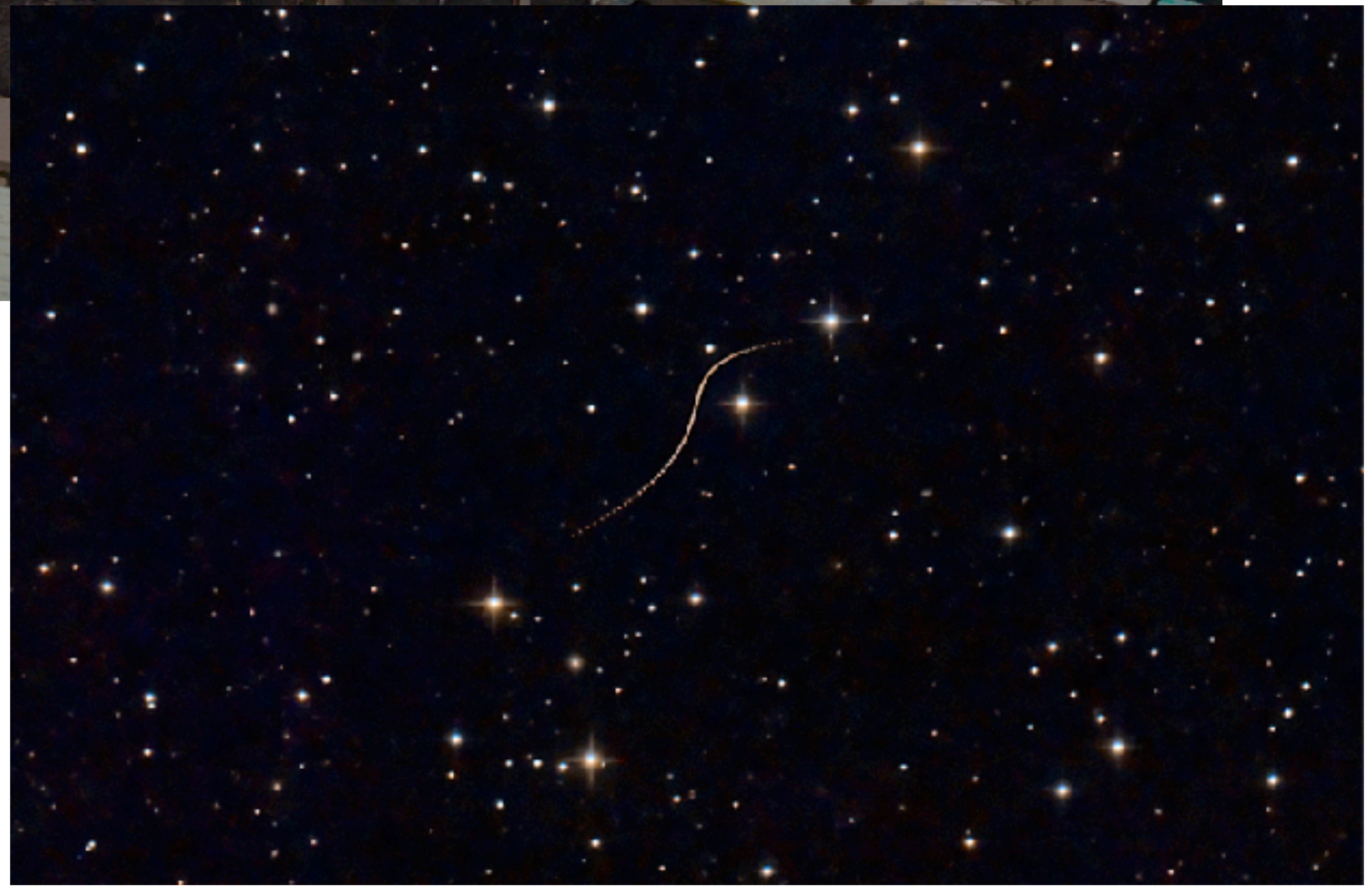
- Spacecraft has been decommissioned and passivated. No fresh observations.

- Cambridge has completed processing for Data Release 4 and delivered data products: 236TB, 433 billion records.

- Preparation for final data processing is underway. Data Release 5 to cover 10 years of observations.

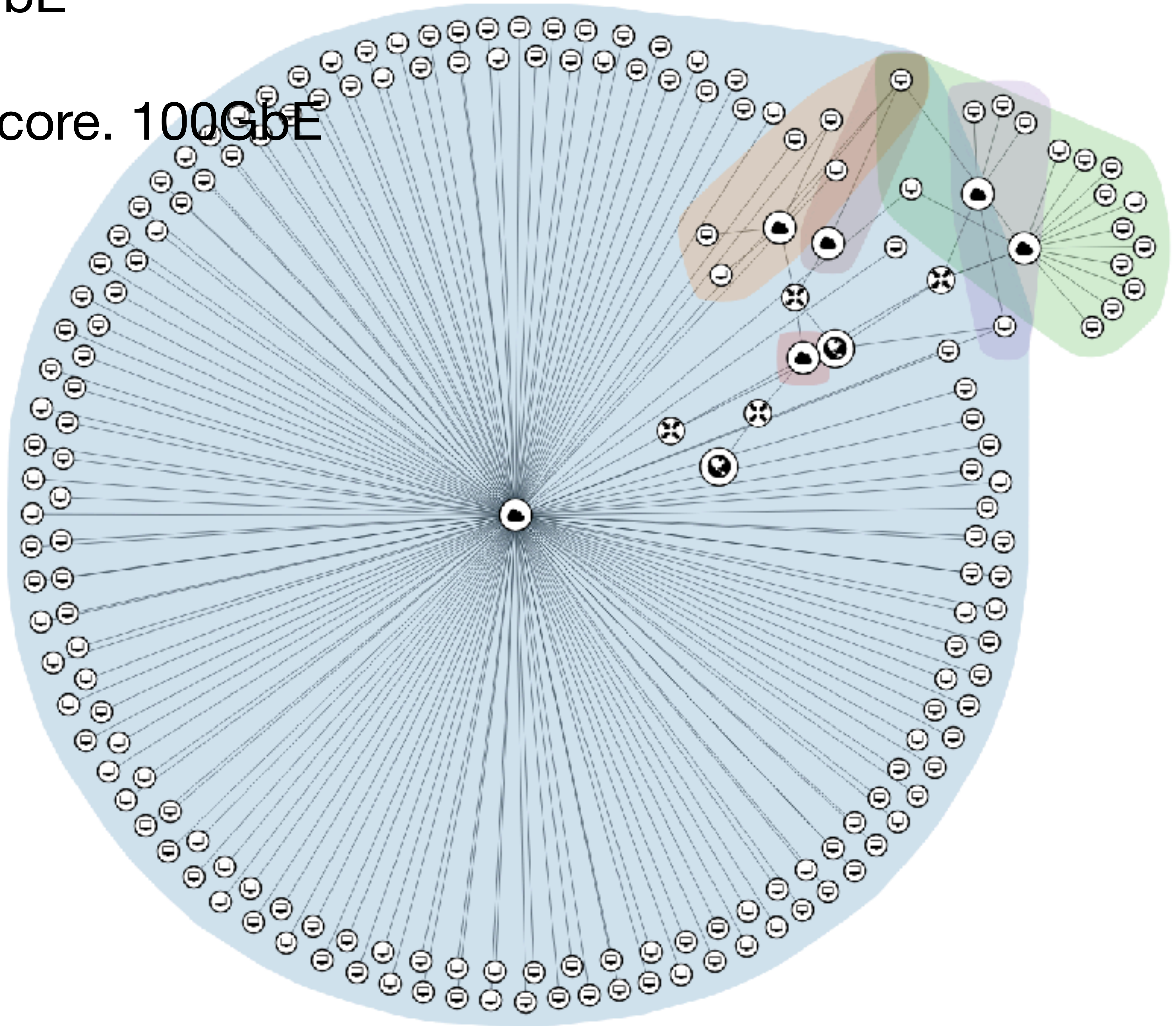
- Data Release 5 is expected “not before the end of 2030”.

    - Testing and development during 2025/26 and then operational processing 2026-2029.

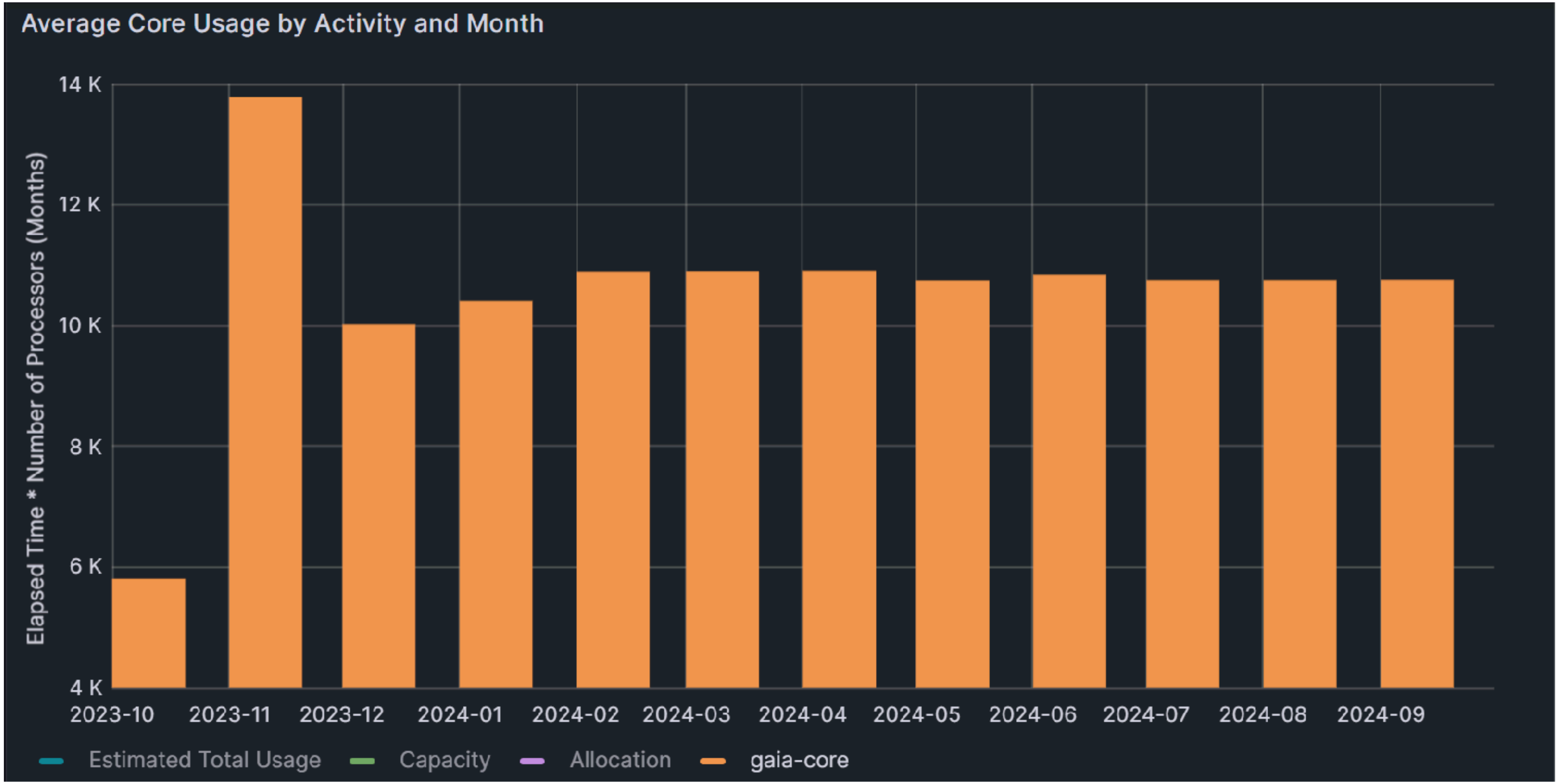


Gaia IRIS Allocation

- Hardware hosting instances:
  - 35 Intel Icelake 76 core, 480G, 3.2G RAM per core. 100GbE
  - 32 Intel Sapphire Rapids 96 core, 1008G, 5.5G RAM per core. 100GbE
- Mixture of systems can impact processing configurations.
- Total is 11,068 vCPU cores (9248 @ 5G RAM per core)
- Total Disk Storage: 4021TB
  - 1,200TB Triple replicated Ceph
  - 1,741TB 8+3 Erasure Coded Ceph
  - 1,080TB Remote S3 storage



Gaia Usage



Average Core Usage by Activity and Month
Elapsed Time * Number of Processors (Months)
14 K
12 K
10 K
8 K
6 K
4 K
2023-10
2023-11
2023-12
2024-01
2024-02
2024-03
2024-04
2024-05
2024-06
2024-07
2024-08
2024-09
Estimated Total Usage
Capacity
Allocation
gaia-core

Gaia Disk Storage

- Gaia processing requires many hundreds of TB inputs, intermediate data products and generated outputs.
- Most disk storage is on CephFS network shares which can be dynamically attached as required to instances.
- Small amount provided as Ceph Volumes - attached block devices backed by Ceph triple replicated storage.
- Significant cost savings using 8+3 erasure coded Ceph vs triple replicated Ceph with no loss of resilience.
- Concerns about performance.

Gaia Disk Storage

- Switch to SRIOV (single root input/output virtualization) ports for Ceph traffic increased performance significantly.
- Tested reading 3.3TB and writing 0.5TB using 50 OpenStack instances and a few hundred cores.
- Numbers reported below are now very old and need to be updated. Major expansion of Ceph hardware has happened since the tests were carried out.
- Triple replicated Ceph:
  - Reading approx 10GB/s Writing approx 4GB/s
- 8+3 Erasure Coded Ceph:
  - Reading approx 3.5GB/s Writing approx 4GB/s
- Conclusion that EC Ceph fine for less frequently accessed datasets.

Software Optimisation

- Development: scaling processes carefully.
  - Designing for performance at scale. It can be more efficient to change how things are processed when running distributed processes compared to small prototype cases.
  - Discover bugs in software before running at large scale. More time testing which is a trade-off.
  - Discover quality issues in processes earlier. Requires more testing which is sometimes difficult to justify. Reduce potential re-execution of expensive processes.
- Think about software and processes.