SKAO on software optimisation

Byte by Byte, Watt by Watt
IRIS Collaboration Meeting

Ugur Yilmaz

02 July 2025

SKAO on software optimisation

Byte by Byte, Watt by Watt
IRIS Collaboration Meeting

Ugur Yilmaz

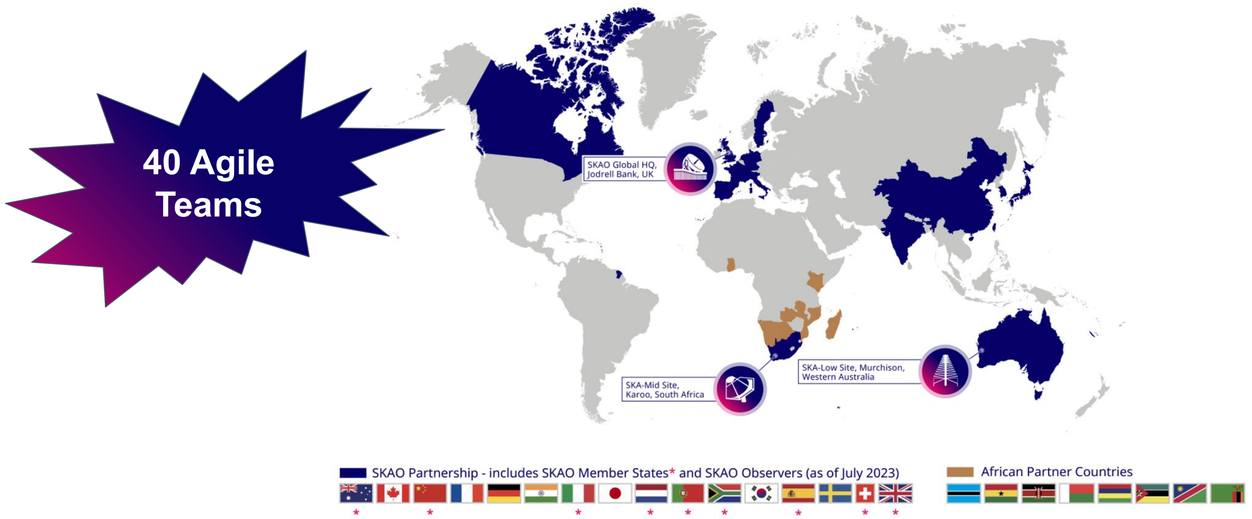
02 July 2025

Agenda

- Background
- STFC Clusters (CI/CD & Integration/Staging)
- SRCNet
- DP HPC Cluster (AWS)

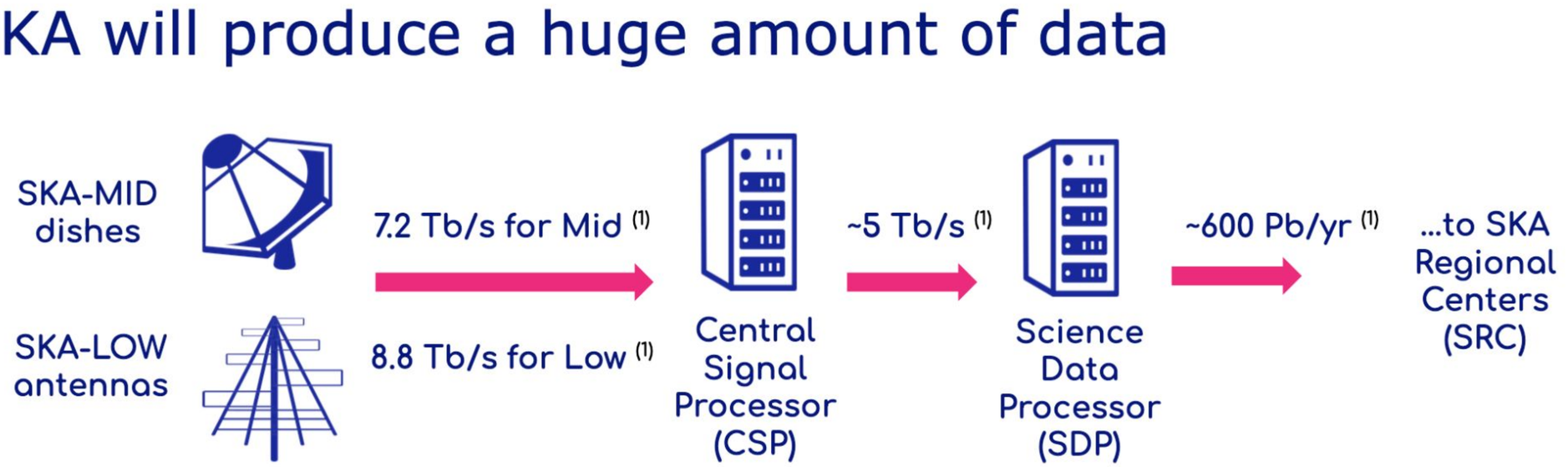
SKAO Construction - Software

- We are a highly distributed and diverse project - different:
  - Time zones
  - Cultures
  - Developer experiences
  - Subject domains
  - Delivery timeframes
  - Large and varied codebase
  - Small central team driving this (supported by a team of 10)



"How do we make sense of this so that we can deliver to a consistent level of quality, and efficiently use resources available, on time, and on budget"

SKA Data Needs



- At this scale every 1 % saving = 120k£(2) per year.

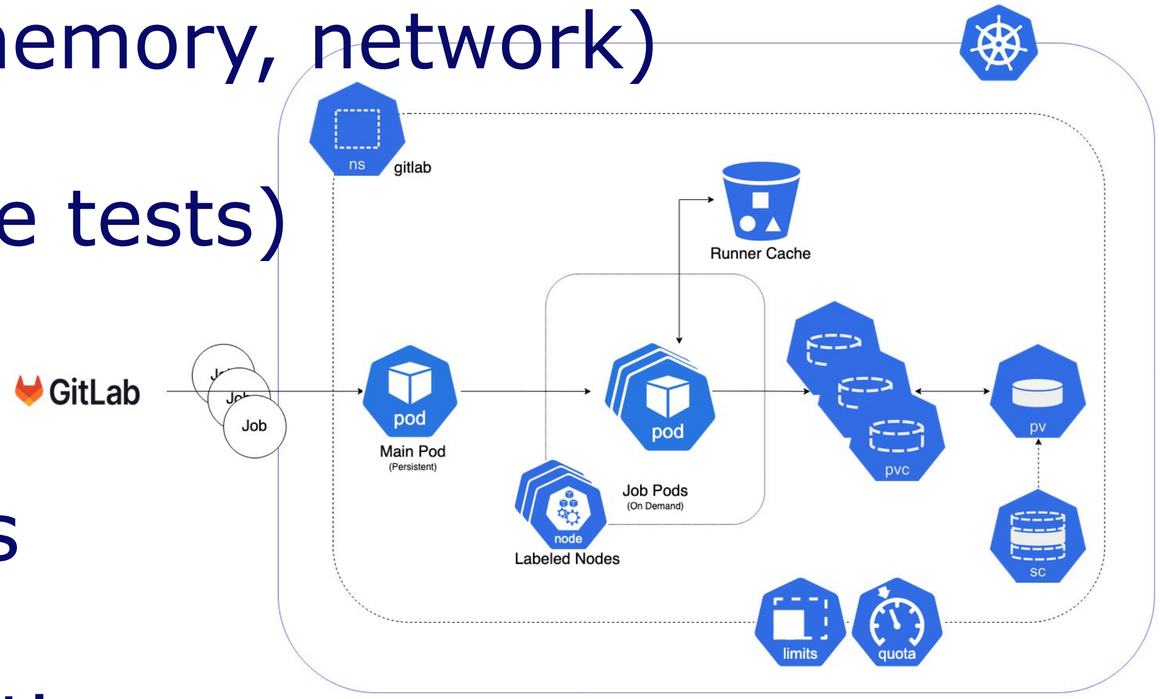
(1) Data for SKA1 implementation
(2) Napkin math based on very rough estimates

STFC - CI/CD

Cluster: 2k vCPU, 8TB RAM, 1 TB Storage

*Daily Average

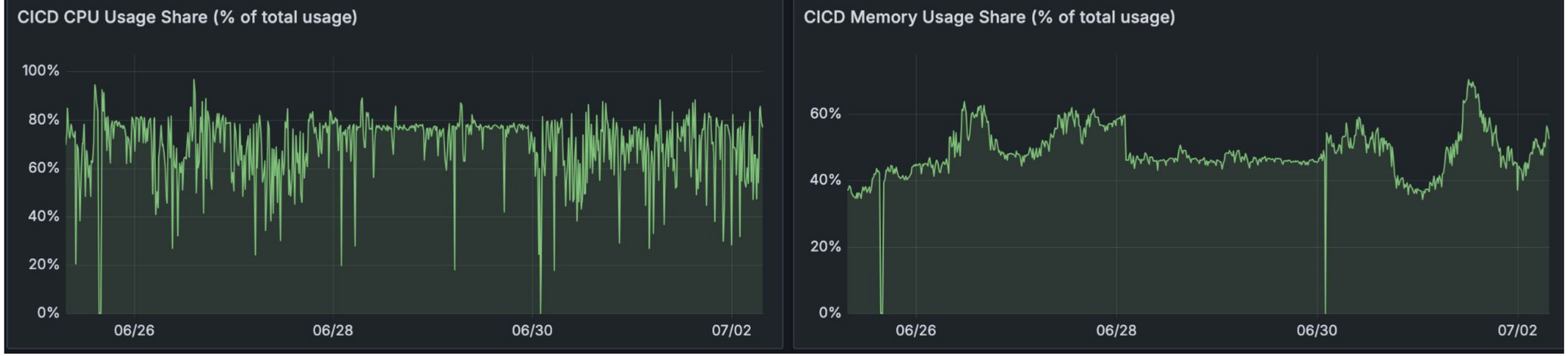
- Main Workload
- Dynamic Job Processing
- **Story:**
  - We set some defaults (cpu, memory, network)
    - They looked good
  - They were bad defaults (broke tests)
  - They didn’t work at all!
    - *Realised many devs do not optimise*
    - *if they think they have the resources*
- Right-sizing containers & pods
- Node heterogeneity
  - Choosing right flavor at right time



Observability

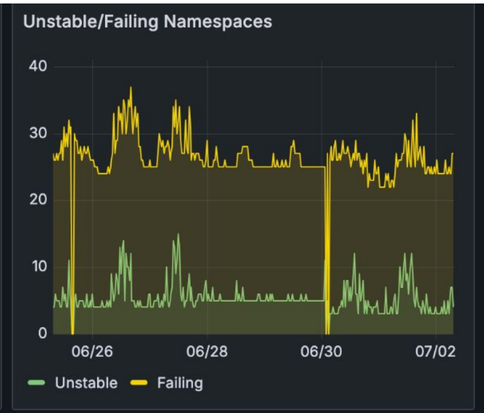
You cannot optimise what you cannot measure

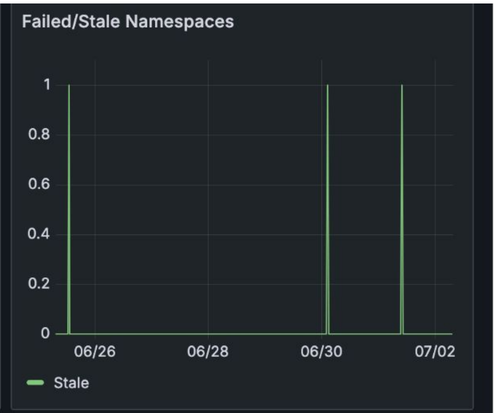
- PromStack to understand, identify and optimise usage
- Enabler for teams to self-optimise
  - *Incentives*
    - Shows progress
    - Recognition
  - *Deterrants*
    - Nudge them in optimising/profiling their workloads
      - *SRE giveth, SRE taketh away*
    - Do not allow usage, eviction policies etc.
- Special Cases
  - One-shot experiments:
    - TMC/CSP requiring 1K vCPU, 5TB Ram for AA2.0 Scale testing

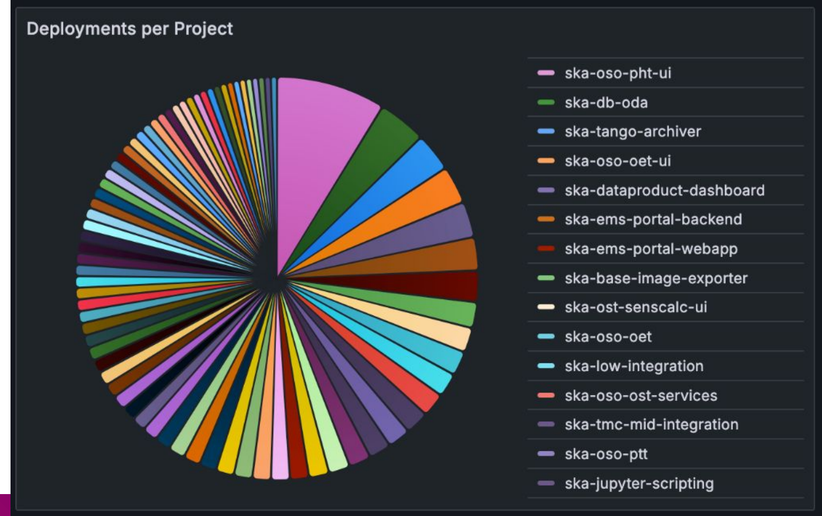

CICD CPU Usage Share (% of total usage)
100%
80%
60%
40%
20%
0%
06/26
06/28
06/30
07/02

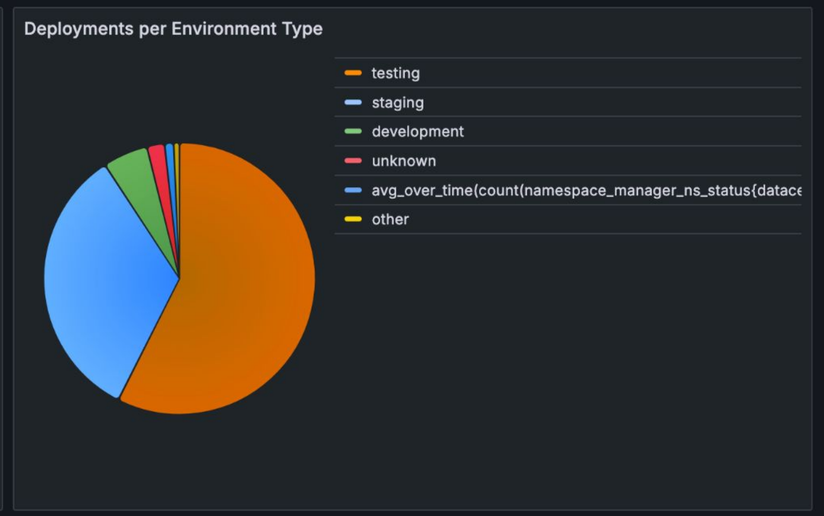
CICD Memory Usage Share (% of total usage)
60%
40%
20%
0%
06/26
06/28
06/30
07/02

Healthy Namespaces
160
140
120
100
80
60
40
20
0
06/26
06/28
06/30
07/02

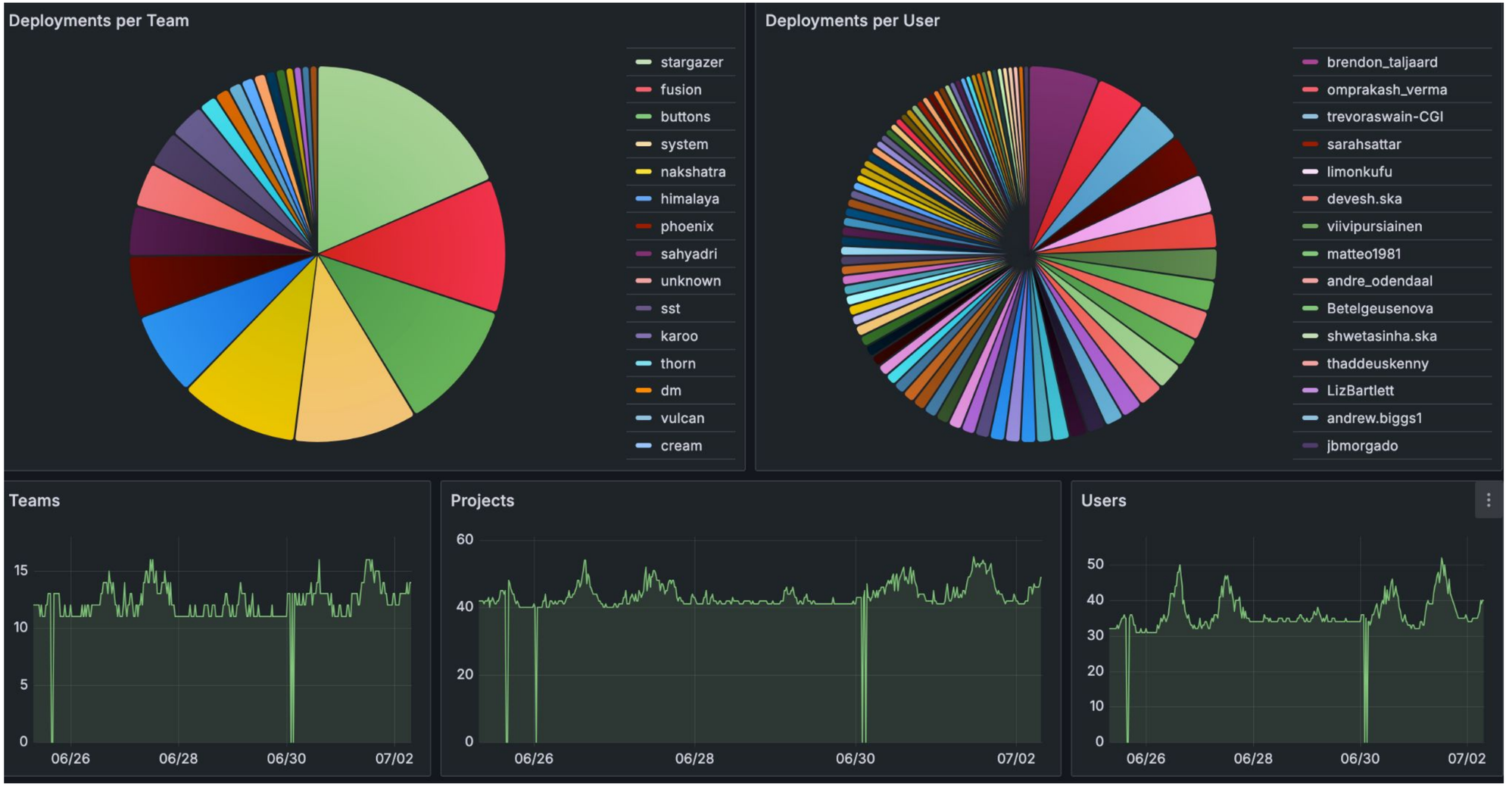

Unstable/Failing Namespaces
40
30
20
10
0
06/26
06/28
06/30
07/02
Unstable
Failing


Failed/Stale Namespaces
1
0.8
0.6
0.4
0.2
0
06/26
06/28
06/30
07/02
Stale


Deployments per Project
ska-oso-pht-ui
ska-db-oda
ska-tango-archiver
ska-oso-oet-ui
ska-dataproduct-dashboard
ska-ems-portal-backend
ska-ems-portal-webapp
ska-base-image-exporter
ska-ost-senscalc-ui
ska-oso-oet
ska-low-integration
ska-oso-ost-services
ska-tmc-mid-integration
ska-oso-ptt
ska-jupyter-scripting


Deployments per Environment Type
testing
staging
development
unknown
avg_over_time(count(namespace_manager_ns_status{datace
other

Observability



STFC - Platform

- Usage of CAPI
  - We scale up&down based on historical data, expected usage
  - Future Work: Autoscaling
- Storage: Future Work
  - *Investigate Performance of different types (next month)*
  - *No plans for tiered storage usage in near future*
- Network
  - Preventing unnecessary traffic with network policies, node affinitions on Calico/Cilium
  - Future: Topology Awareness, QoS Policies
    - Topology Awareness (or simulating it) will allow us to deploy/redeploy and rotate more efficiently

Some funny(funny now!) stories

- Unexpected Cluster Rollouts while learning/using CAPI
  - Cost a lot of time to rotate everything
- Switching to AWS for STFC downtimes/maintenance
  - We have a small AWS CI/CD cluster as backup
  - First time we switched it showed being cloud agnostic, IaC is hard!
    - Hardcoded/Uncoded config values, LB and Network Issues
    - User Migration
- Backlog of CI jobs or deployments/workloads that takes ages for unnecessary usage: *Clean up Policies*
  - unstable deployments get deleted after 2 days
  - All testing/ci deployments gets deleted after 1 day

SRCNet - Services utilising IRIS Resources

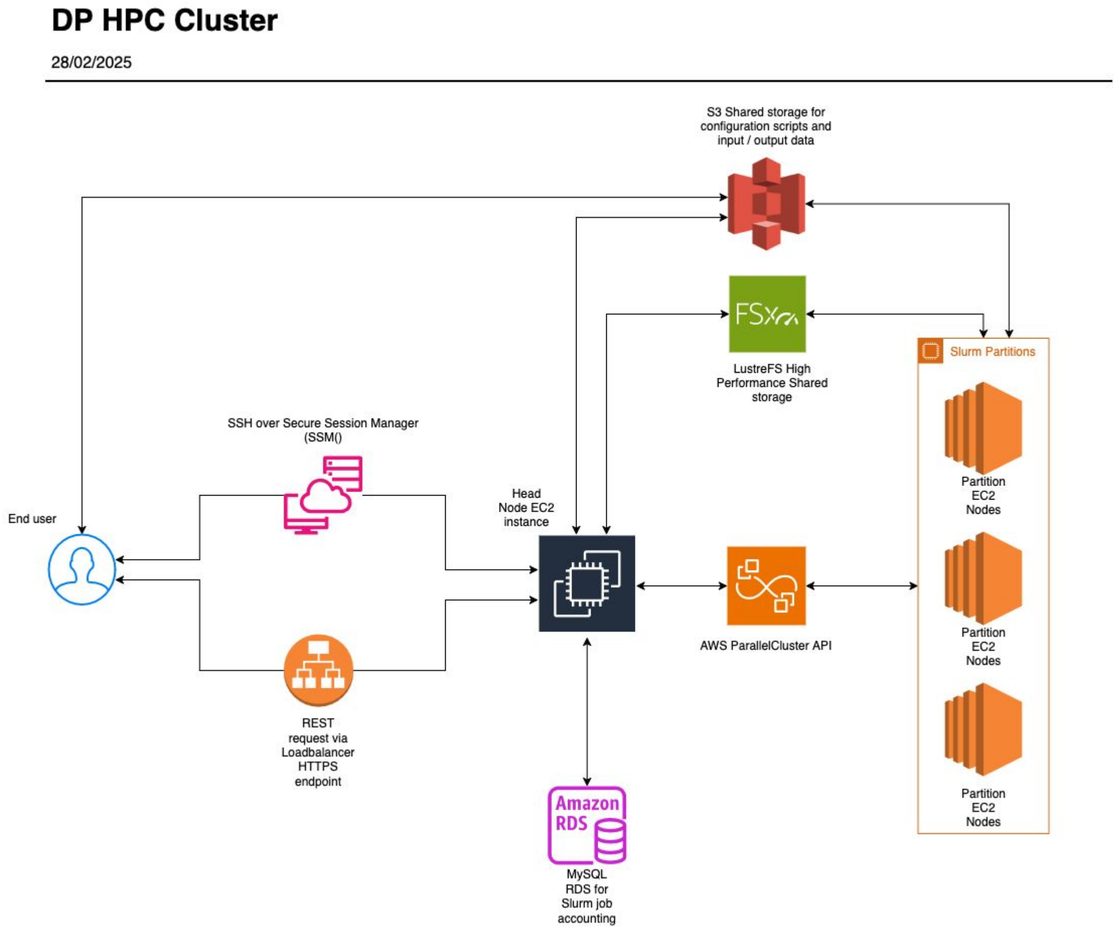
Cluster Resources: 400 CPU, 1.6TB RAM, 40TB

- CAPI management cluster controls workload clusters
- Global services supporting SRCNet running in STFC Cloud
  - Rucio data management system + core API services
  - Metadata management services
  - Indigo IAM + API interface services
- Local ‘SKAOSRC’ test environment
  - StoRM-WebDAV Rucio Storage Element
  - ‘CANFAR’ container-based science platform
  - Data access services

DP HPC Cluster

Started as pilot project
Usage: 3k vCPU peak, (350 average), 229 TB RAM peak (194 avg),30TB Lustre

- Provides a scalable on-demand HPC solution
- Utilises AWS ParallelCluster
- Reduces cost compared to always-on configuration by 80-90%
- Accessible via CLI (users) and REST API (applications)
- Built and operated in house by SKAO to allow rapid / agile configuration changes and reduce 3rd party costs



DP HPC Resource Optimisations

- **Provisioning:** Nodes in a HPC partition are shut down when not in use or needed - only head node stays active
- **Location:** Resides in AWS Stockholm region which is 10% cheaper than London region, while offering more HPC capacity
- **Usage:** Multiple node sizes available to suit use cases
- **Price:** Spot priced nodes have a higher default priority than on-demand. Users are encouraged to use spot nodes wherever possible
- Default partition is spot priced with suitable node type
- Input / output working data stored in S3

WSClean Scaling/Optimisations

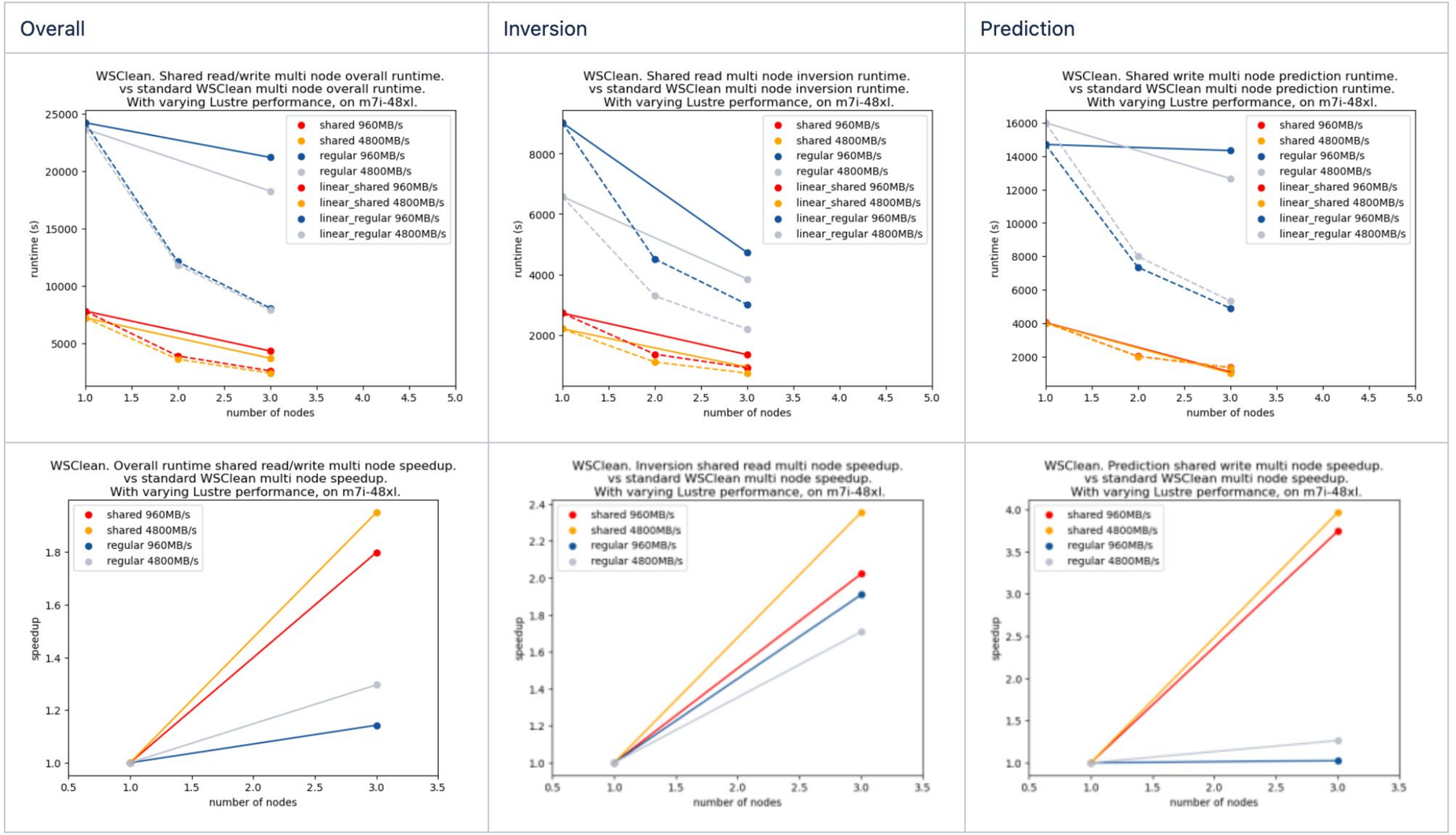
Science Data Processing Workflows

User-wise:

- *From scaled back parameters to full performance scale*
- *Base performance*
  - *New bottlenecks*
- *Feeds back to how actual pipelines should look like*

Infra-wise:

- Allows us to see the end goal and create a roadmap (at a cost) for our own HPC
  - Machines Configs..
  - Storage..
  - Use-cases..



Thank you!

ugur.yilmaz@skao.int

We recognise and acknowledge the Indigenous peoples and cultures that have traditionally lived on the lands on which our facilities are located.

SKAO

www.skao.int