UKRI Science and Technology Facilities Council

Scientific Computing

Welcome

Science and Technology Facilities Council

Scientific Computing

Federated Kubernetes with Geo-Distributed IRIS IAM

Donald Chung (STFC)
IRIS IAM Service Manager

2nd July 2025

Agenda

1 Overview of IRIS IAM

2 Geo-Distributed/Multi-cluster HA-IAM technical overview

3 Performance testing findings for Multi-cluster IAM

4 Multi-cluster Setup beyond IAM

Science and Technology Facilities Council

Scientific Computing

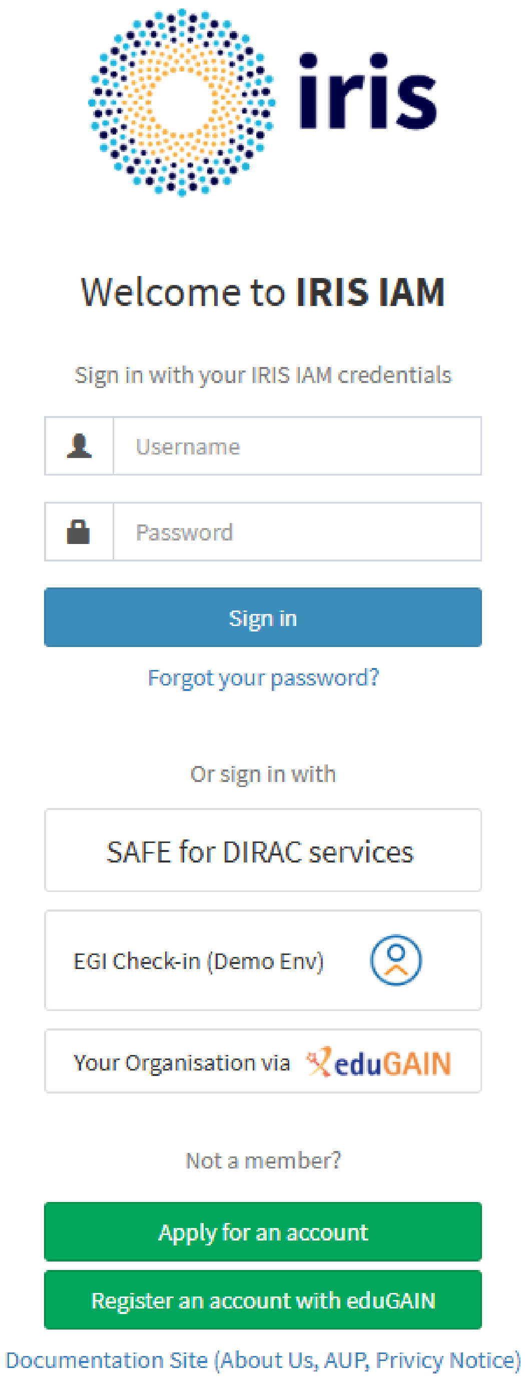
Science and Technology Facilities Council

Scientific Computing

Overview of IAM

What is IRIS IAM?

- IAM (Identity and Access Manager) provides an Authentication and Authorization Infrastructure (AAI) solution to IRIS.
- The IAM acts as a proxy service, allowing IRIS collaborators access to other IRIS services.
  - SCD Cloud
  - IRIS indico
  - SAFE for Dirac
  - FTS & Rucio
  - Many more…



iris

Welcome to **IRIS IAM**

Sign in with your IRIS IAM credentials

Username

Password

Forgot your password?

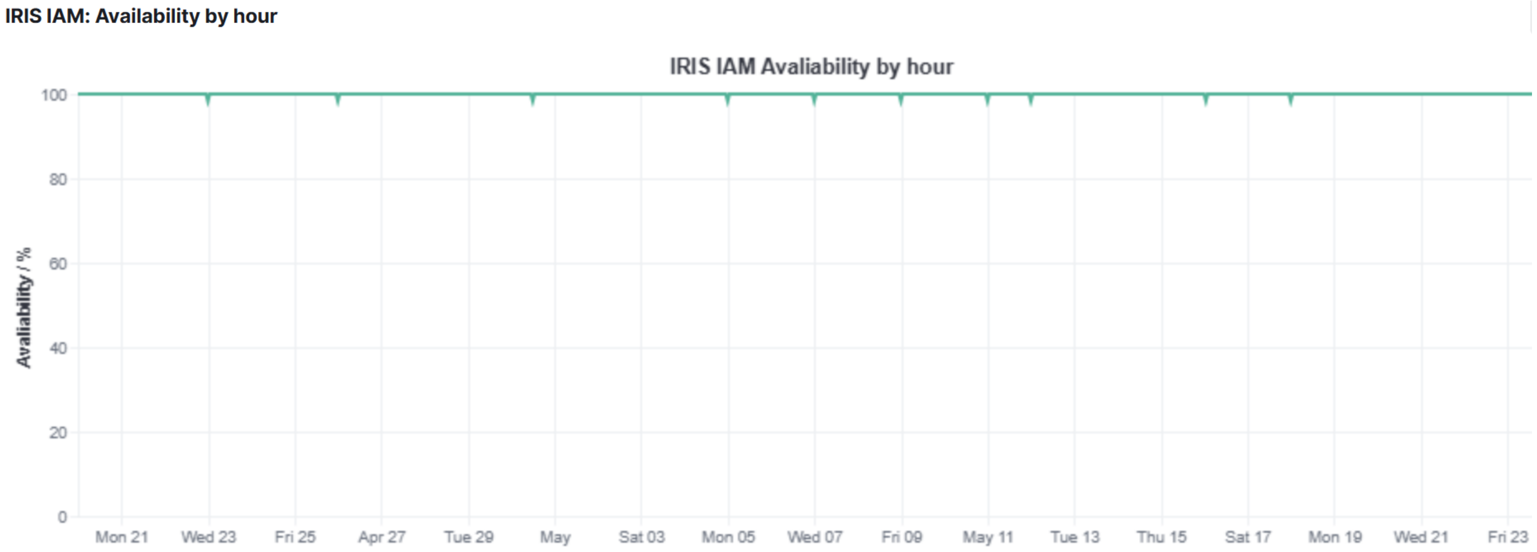
Or sign in with

Not a member?

Documentation Site (About Us, AUP, Privacy Notice)

IRIS IAM – Why HA

- Provide IAM for entire UK
- Good availability
- Reduce risk of
  - Loss of service
  - Provide better grantee for downstream services
- Geographically distributed IAM service



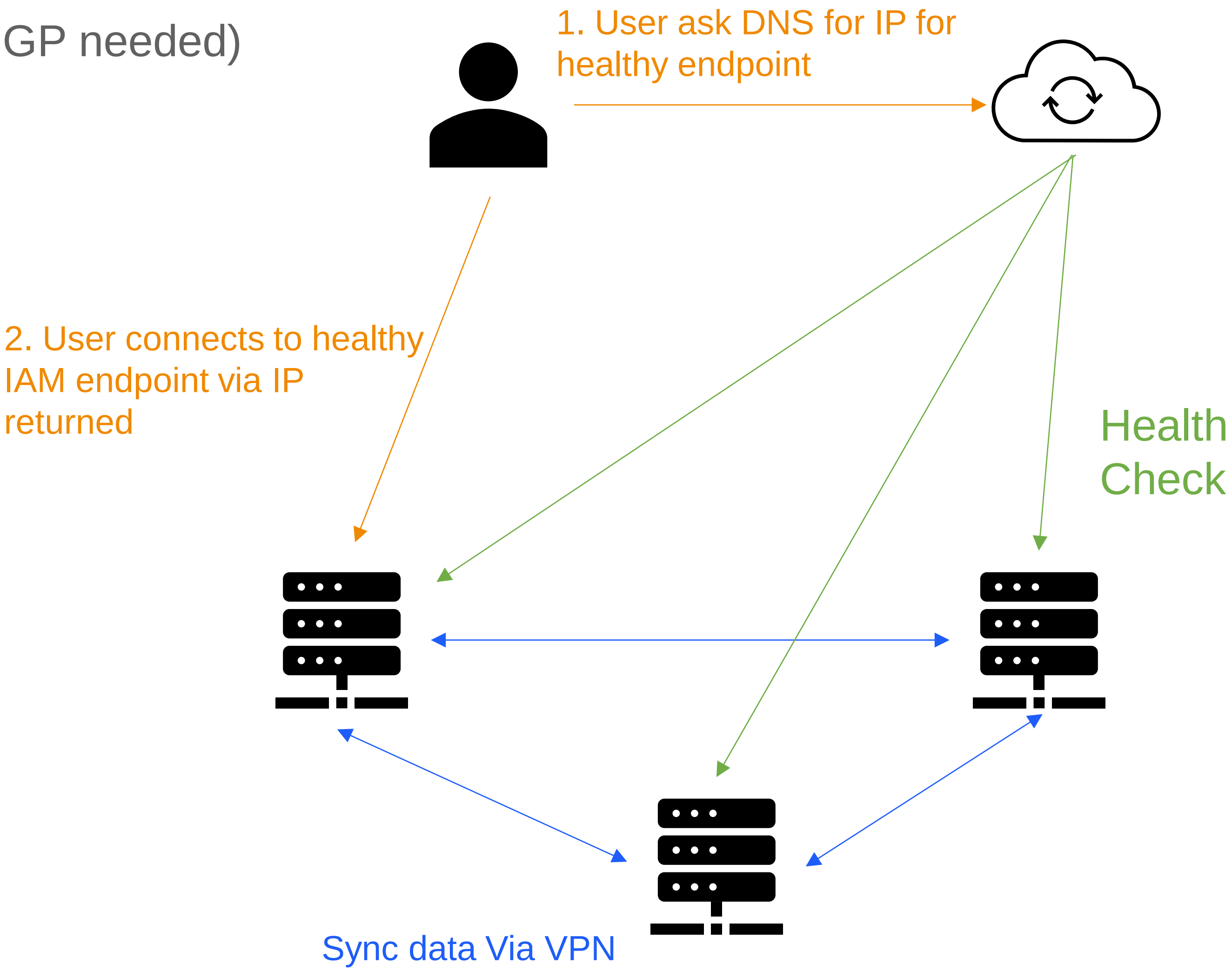
UKRI Science and Technology Facilities Council

Scientific Computing

Geo-Distributed HA-IAM technical overview

Architecture

- DNS load balancer
  - Low infrastructure requirement (No BGP needed)
- Kubernetes
  - Running
    - IAM
    - Database
    - Session Storage
  - Performance advantage
- VPN services
  - Allow synchronization of data
- Needed
  - Data synchronization
  - Orchestration


1. User ask DNS for IP for healthy endpoint
2. User connects to healthy IAM endpoint via IP returned
Health Check
Sync data Via VPN

Liqo

- liqotech/liqo: Enable dynamic and seamless Kubernetes multi-cluster topologies
- Self-negotiated resource and service consumption relationships between cluster
  - VPN configurations
  - Certification authorities
- Workload offloading to remote clusters
  - No modification to K8s
  - Status transparent
- Network Fabric: Native Pod-to-Pod and Pod-to-Service
  - VPN tunnel for secure communication
  - Synchronisation of State
- Storage Fabric:
  - Auto configuration of storage class
  - Storing that data closer to workload

Liqo

- Setup
  - Helm
  - Build-in CLI application
- Peering between cluster
  - Made aware of each other’s configuration
    - E.g. pods and service CIDR
  - Propagation of pod affinity/anti-affinity
  - Reflecting resource
  - Automatic offloading namespace
  - CA
  - Setting up control plane
    - Communication with kubeapi can be done within VPN or outside of VPN

Liqo

- Install with Helm or liqoctl on both cluster
  - Set parameters such as: Pod/Service CIDR, amount of resource to share, resources not to share, gateway network
- Peering the cluster
  - Negotiate Network
  - Create relevant resources such as resources slice, network pods
- Foreign cluster represented as worker virtual nodes ready to schedule workloads from master node

Master
Foreign Worker

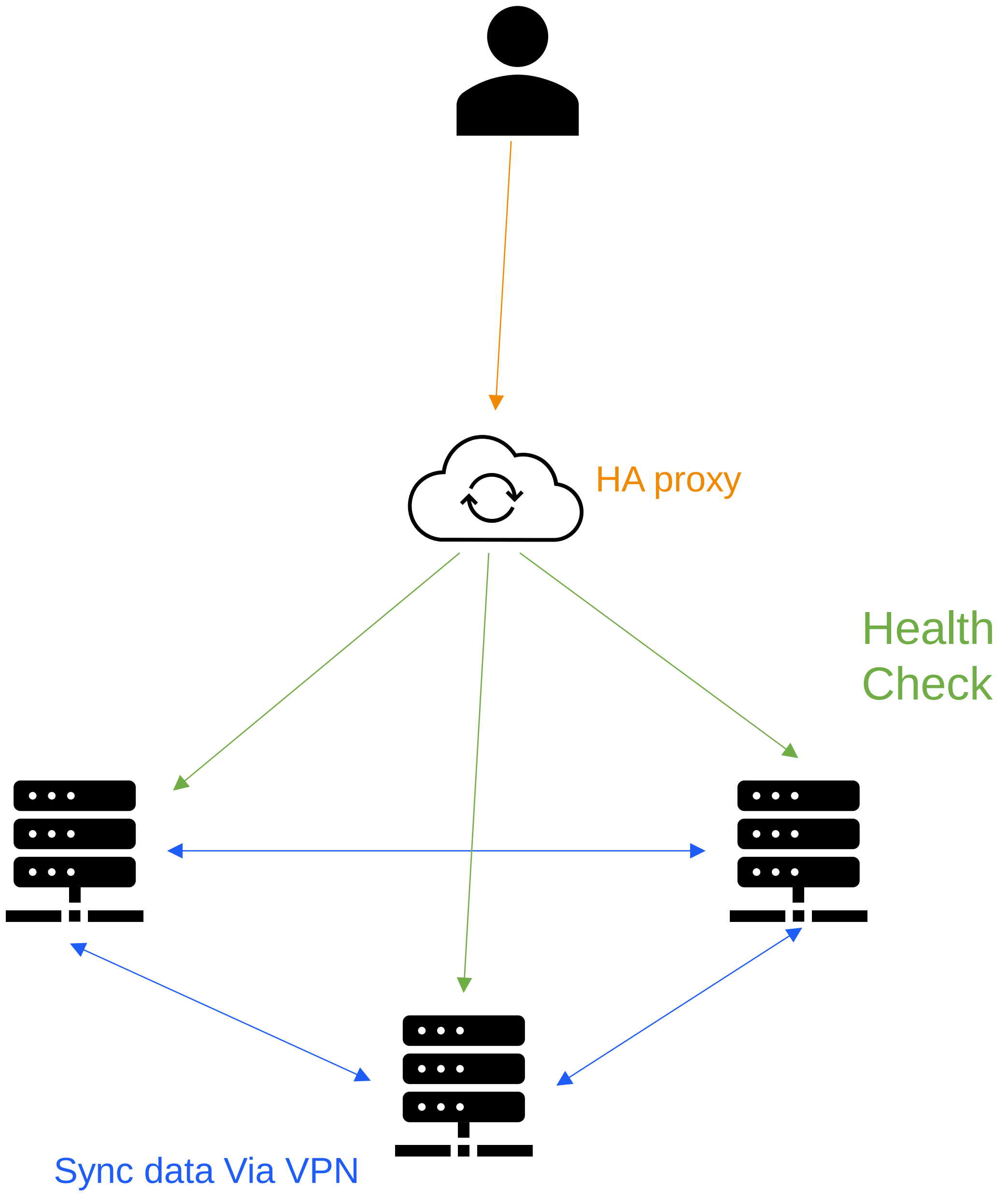
Testing Scopes

- Mainly testing service reachability and performance in the IAM Context
  - With combination of
    - Container Engine
    - Network Environment
    - Backend DB

- Kubernetes Engine performance is out of scope of this investigation
  - Benchmarking Liqo: Kubernetes Multi-Cluster Performance | by Marco Iorio | The Liqo Blog | Medium
  - Minimal at lost at 10k pods and 100ms latency between cluster

Testing Architecture

- HA Proxy load balancer
  - 4x Core 16GB RAM
  - Round Robin
- Kubernetes
  - RKE2
  - Testing Local Cluster
    - 3x 8 Core + 30GB RAM (HA masters)
  - Testing Remote Cluster
    - 2x 8 Core + 30GB RAM (1x Master, 1x Worker)
  - 30ms latency introduced with Linux traffic command, queuing discipline applied on all remote cluster nodes IP
- IAM Setup
  - One container per node
  - Nginx
  - INDIGO IAM
  - Redis Sentinel
  - Persistence Database
    - SCD Galera
    - MariaDB Replication
    - Galera



Science and Technology Facilities Council

Scientific Computing

Testing

- Locust - A modern load testing framework
    - Python based
- Tests
    - Access Token
        - Issue Access Token
    - Refresh Token
        - Issue Access Token → Issue Refresh Token
    - Workflow
        - Issue Access Token → Issue Refresh Token → Token Exchange
- Hardware Setup
    - 10 min x 3 Trial / Setup
    - 8 Worker
        - 500 simulated User
        - 10 users/s ramp up

Science and
Technology
Facilities Council

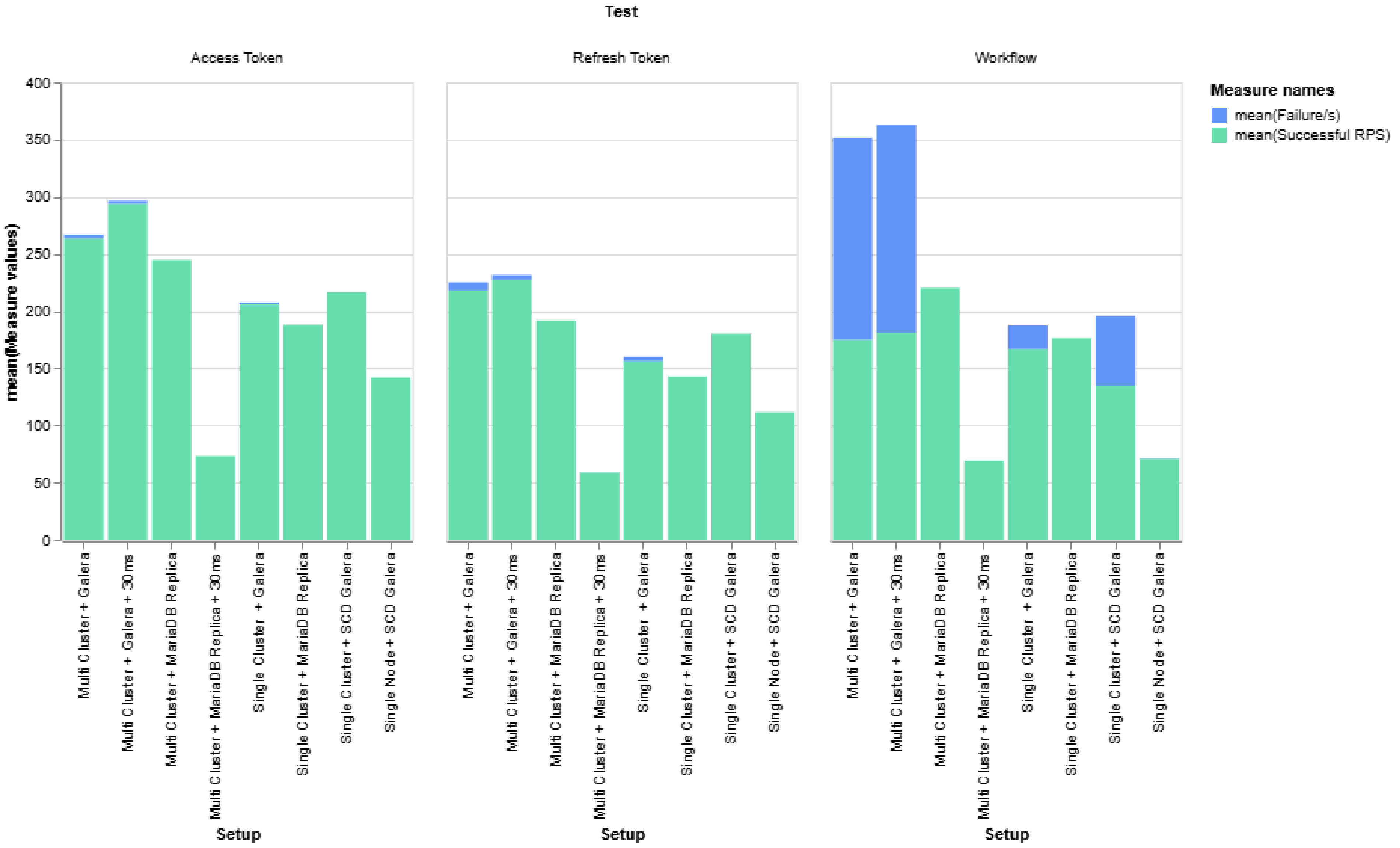
Scientific Computing

Findings

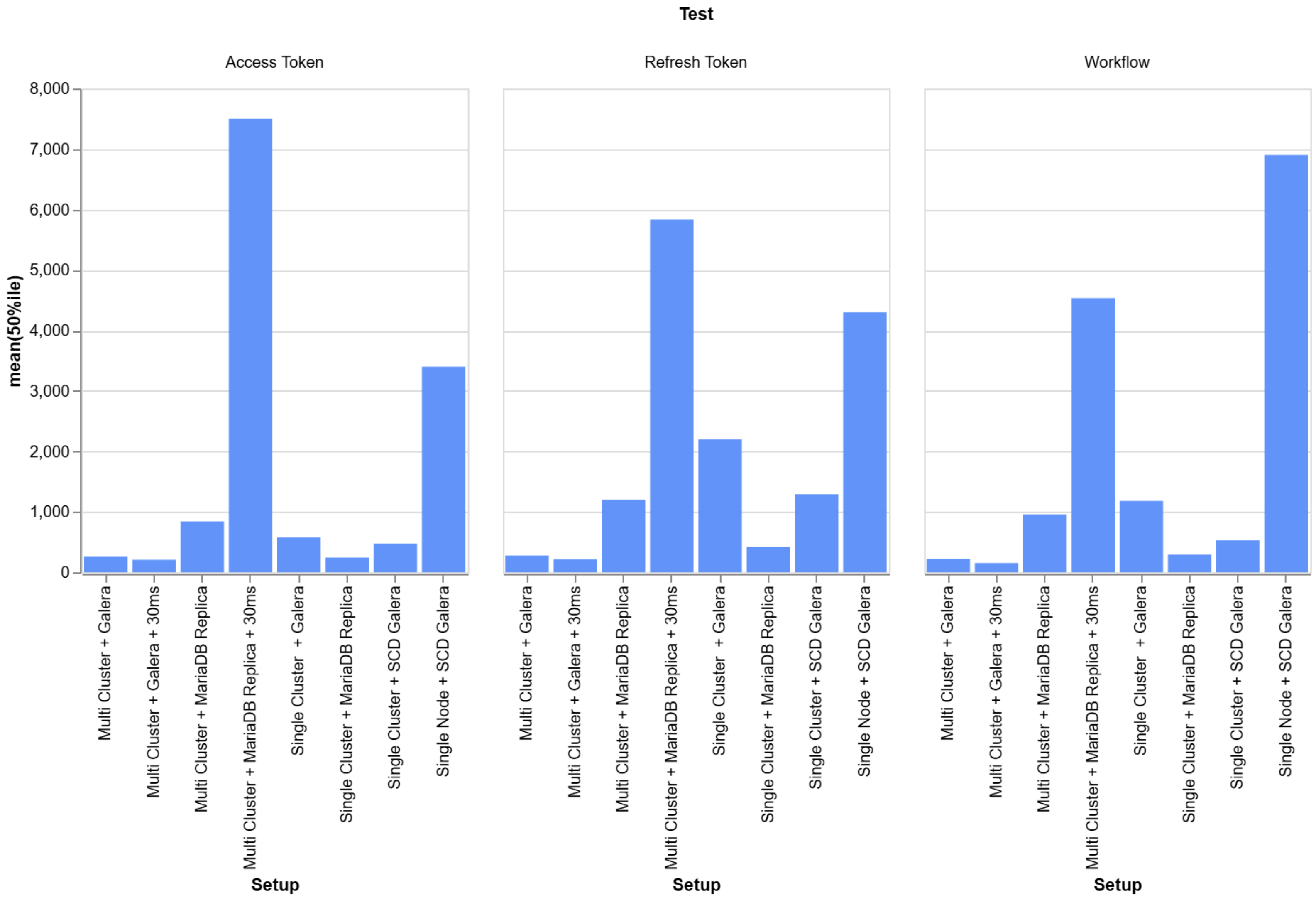
Current IAM

- Usage Level
  - IRIS IAM
    - 10-20 tokens / hr
    - 20-30 logins / hr
    - ~850 users in total
    - ~330 clients in total
  - SKA IAM
    - 1000 – 2000 tokens/ hr
    - 1000 – 2000 logins/ hr
    - ~200 users in total
    - ~ 620 clients in total

Request per Second (Throughput)



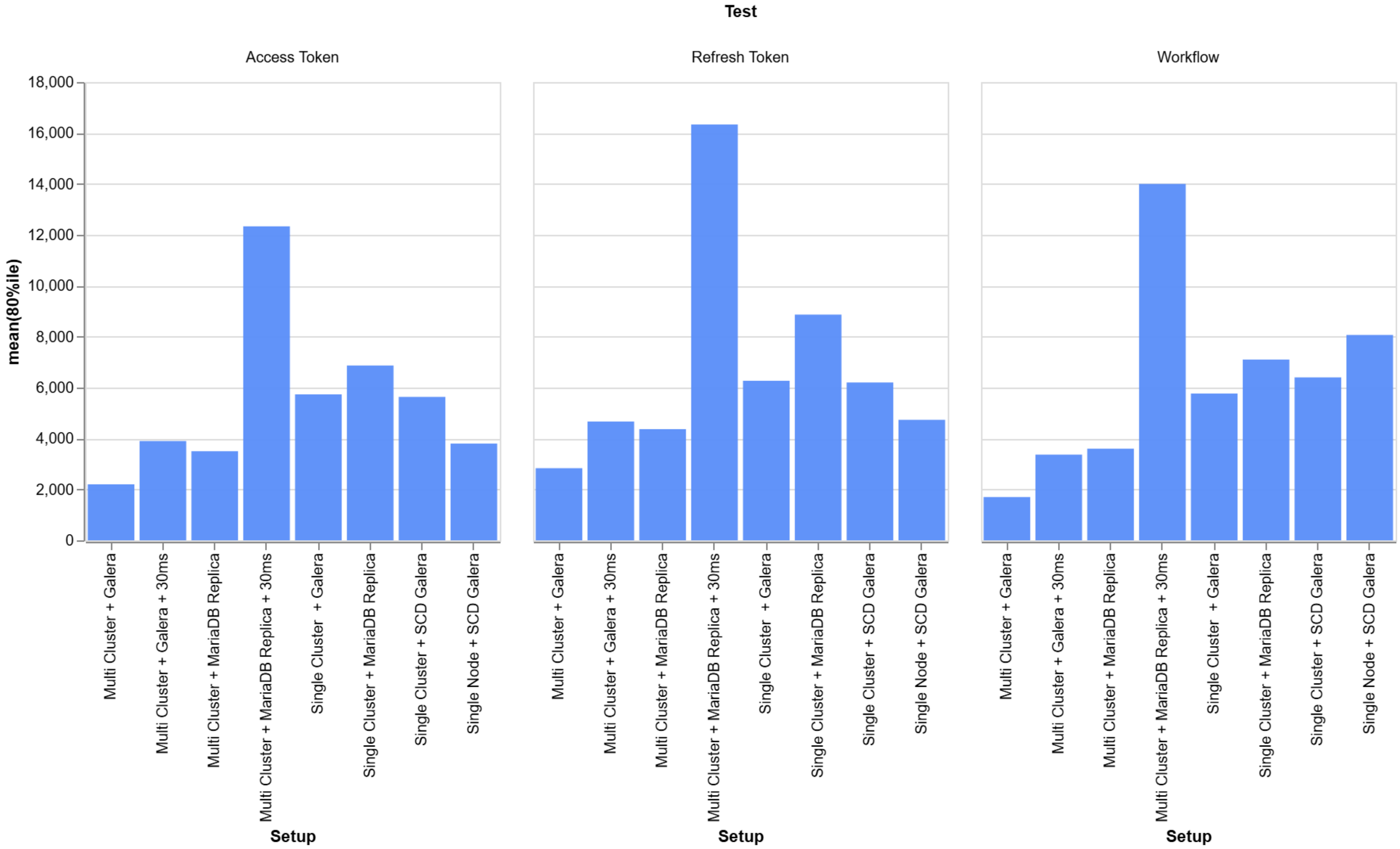
Response Time (50 percentile)



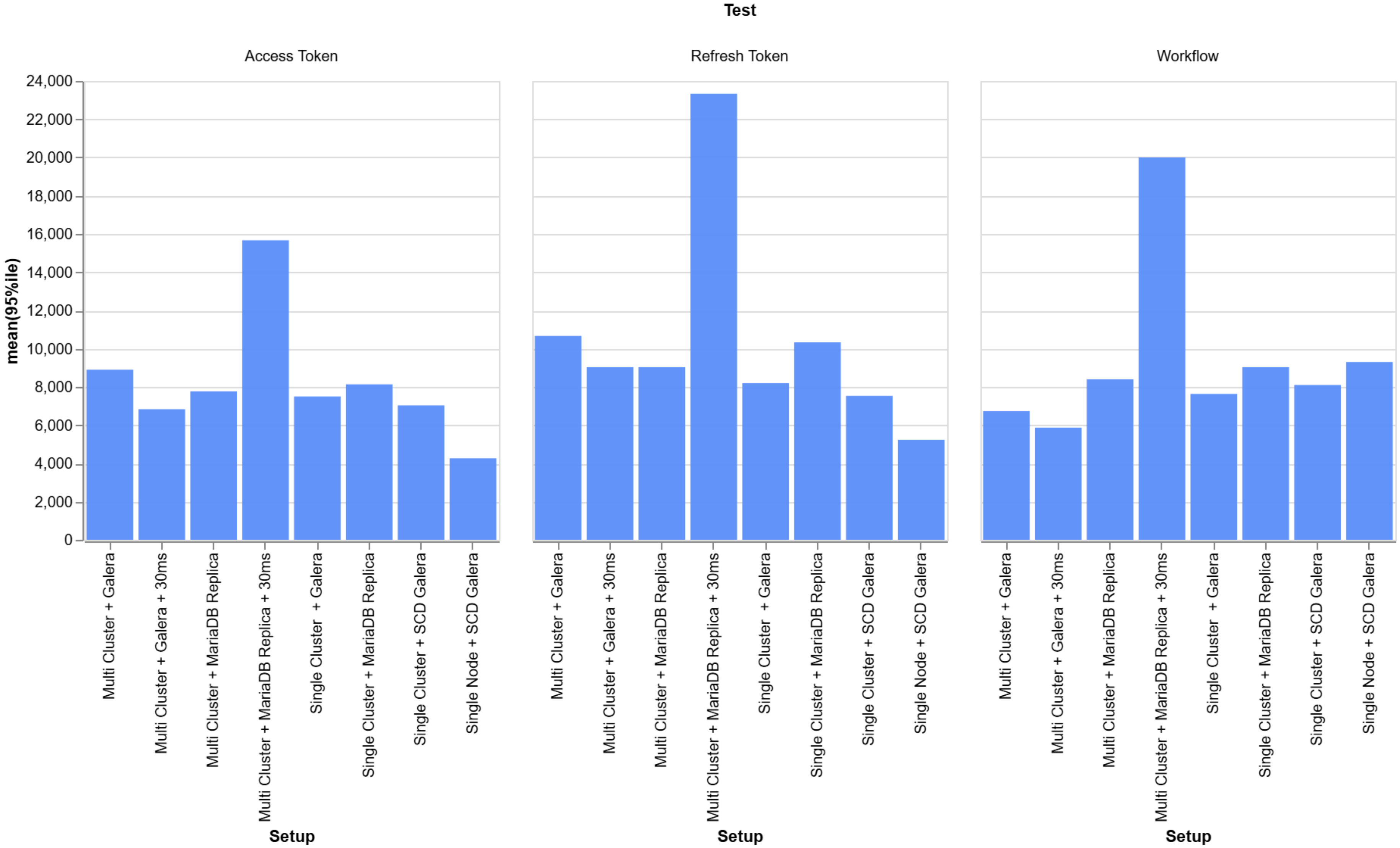
Science and Technology Facilities Council

Scientific Computing

Response Time (80 percentile)



Response Time (95 percentile)

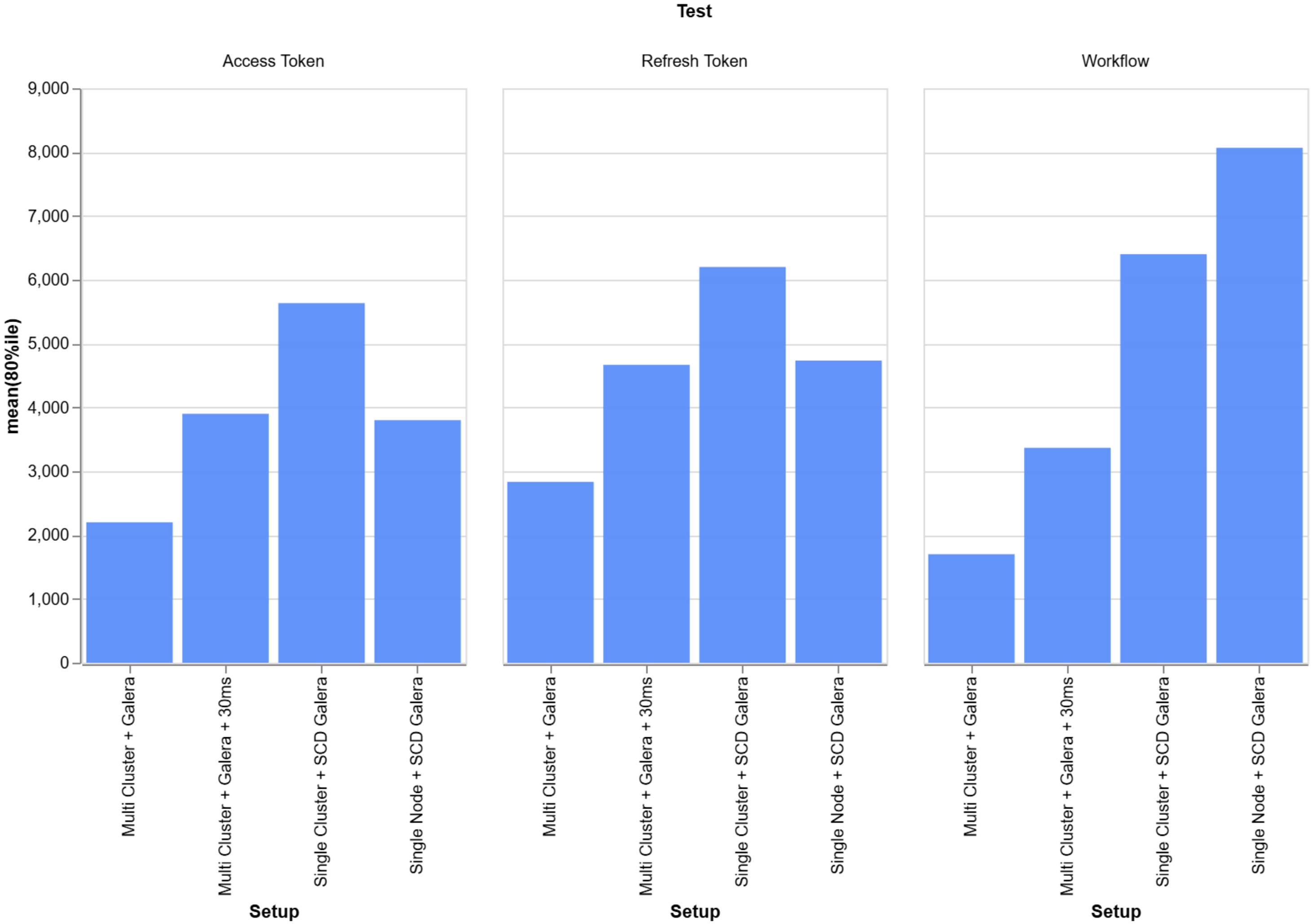


Science and Technology Facilities Council

Scientific Computing

Findings

- More container = Better performance
  - Single node Docker < Single cluster (3x Frontend) < Multi cluster (5x Frontend)



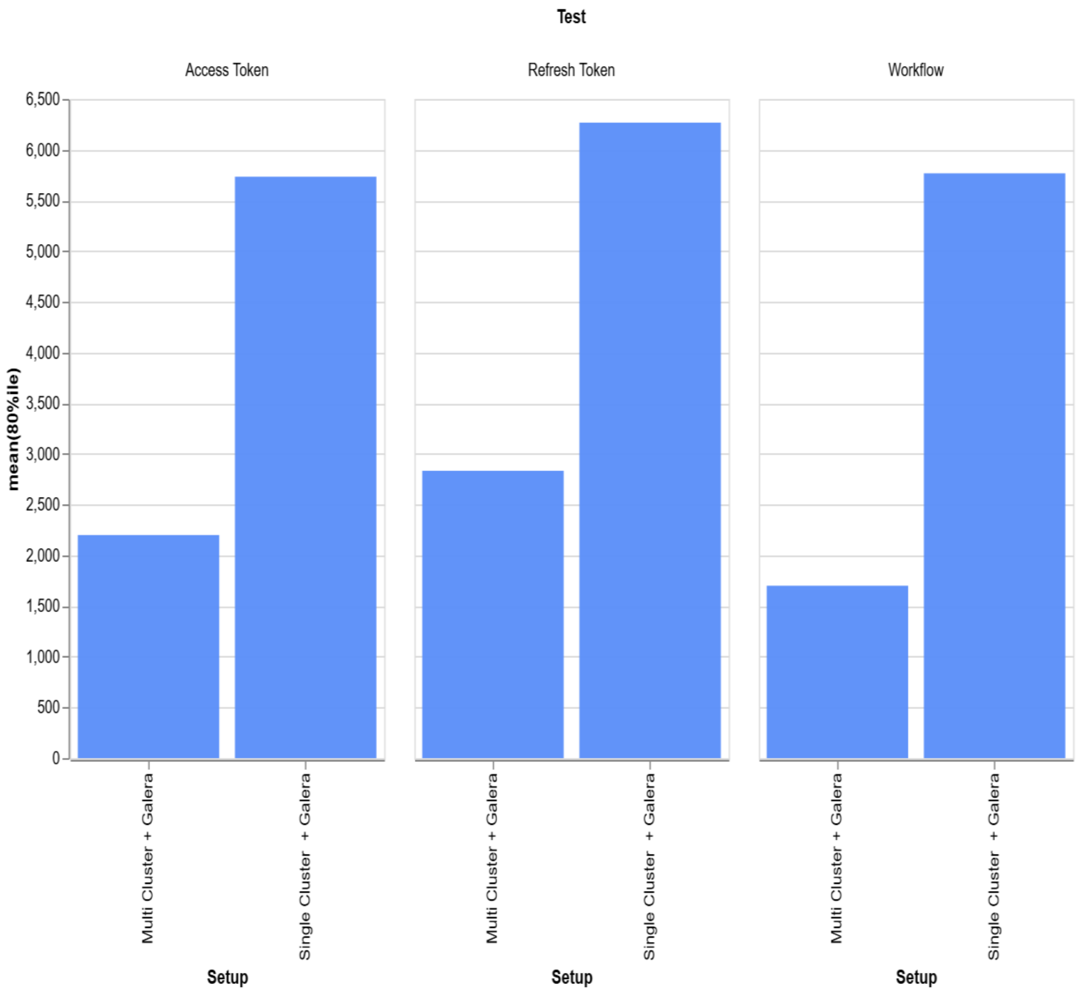
Science and Technology Facilities Council

Scientific Computing

Findings

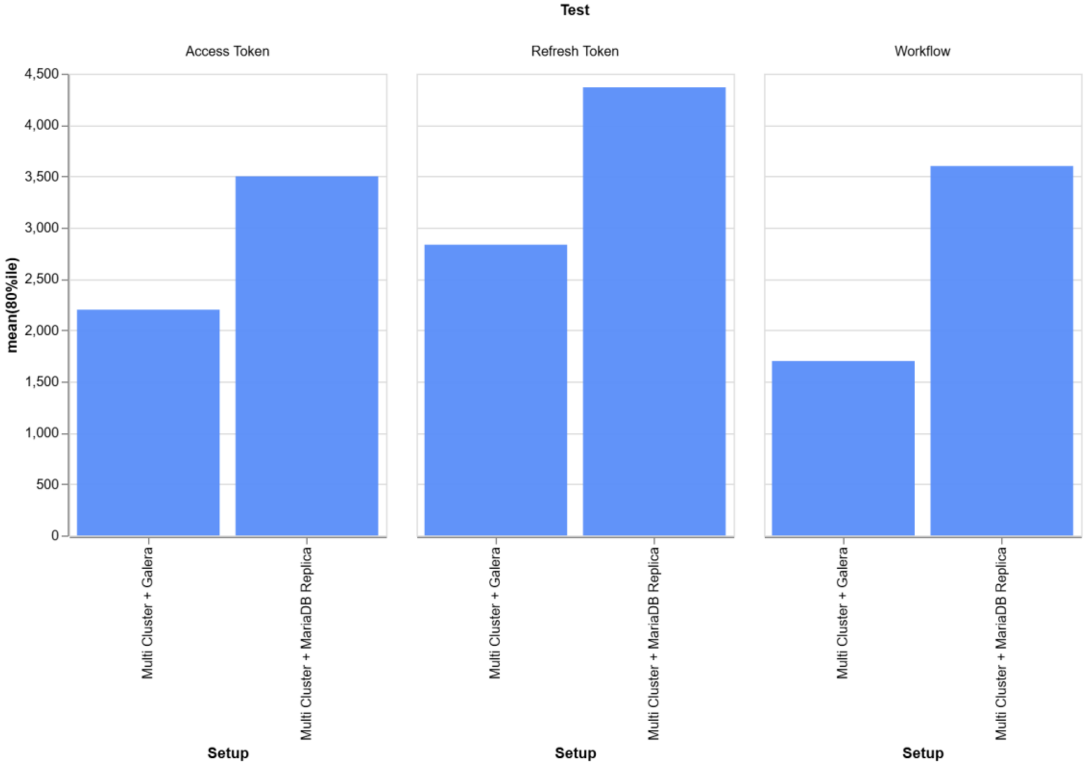
Minimal overhead with the network fabric

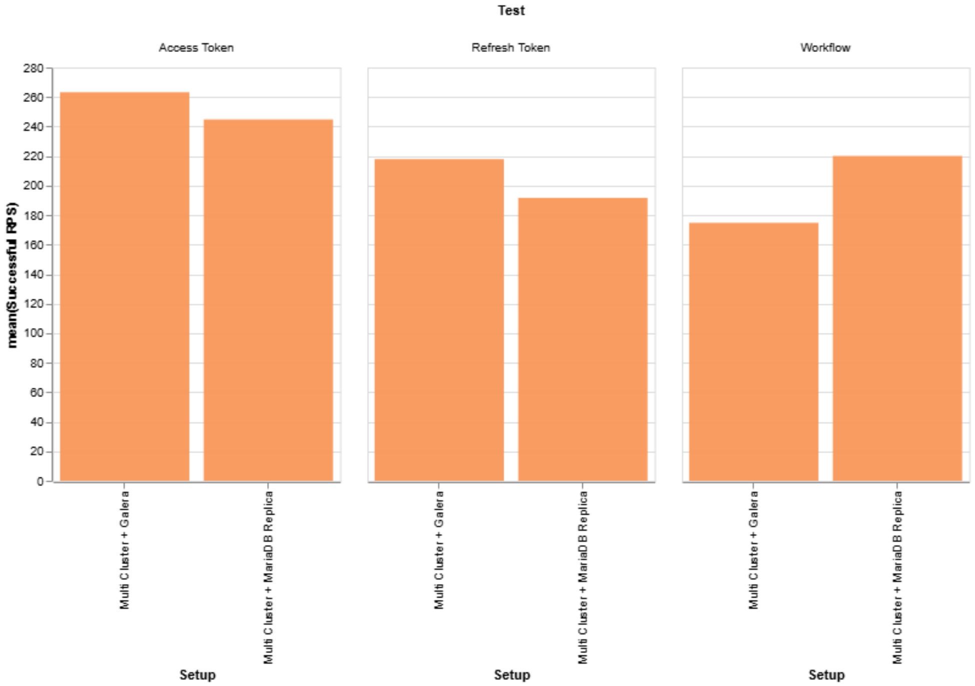
- VPN wireguard, additional overhead for remote monitoring



Findings

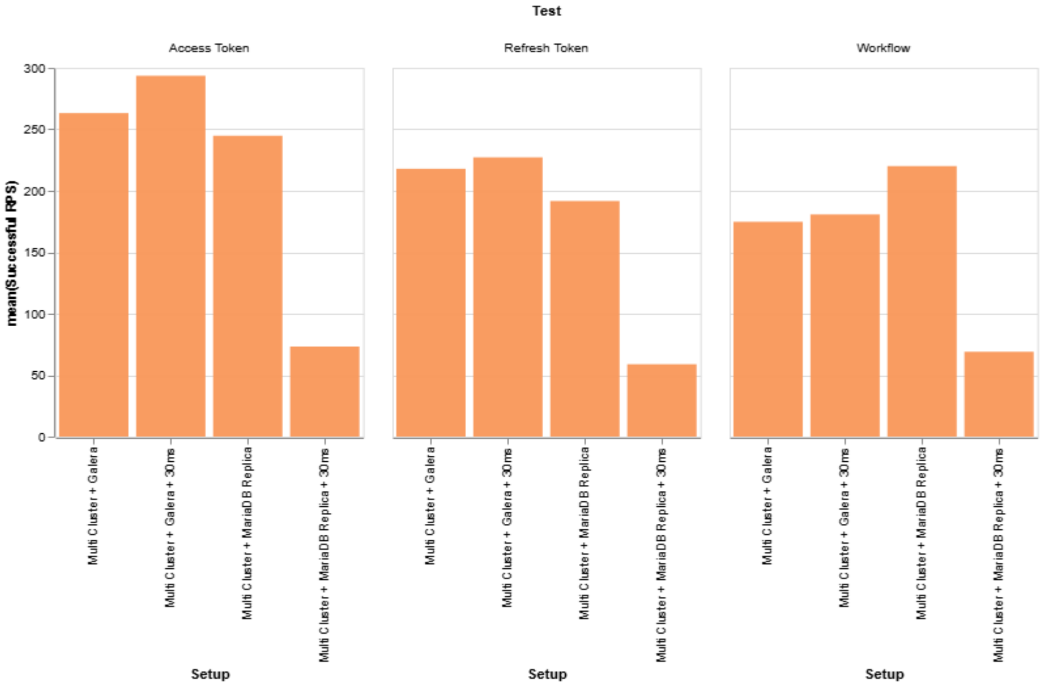
- Galera yields better Throughput and response time
  - Especially for workloads involves higher proportion of DB read





Findings

- Performance drop significantly when
  - Higher latency between DB and INDIGO IAM
    - Replica maria DB ensure worse case scenario most of the time at latency
    - Minimum difference when no latency between cluster, performance degraded significantly when latency is introduced

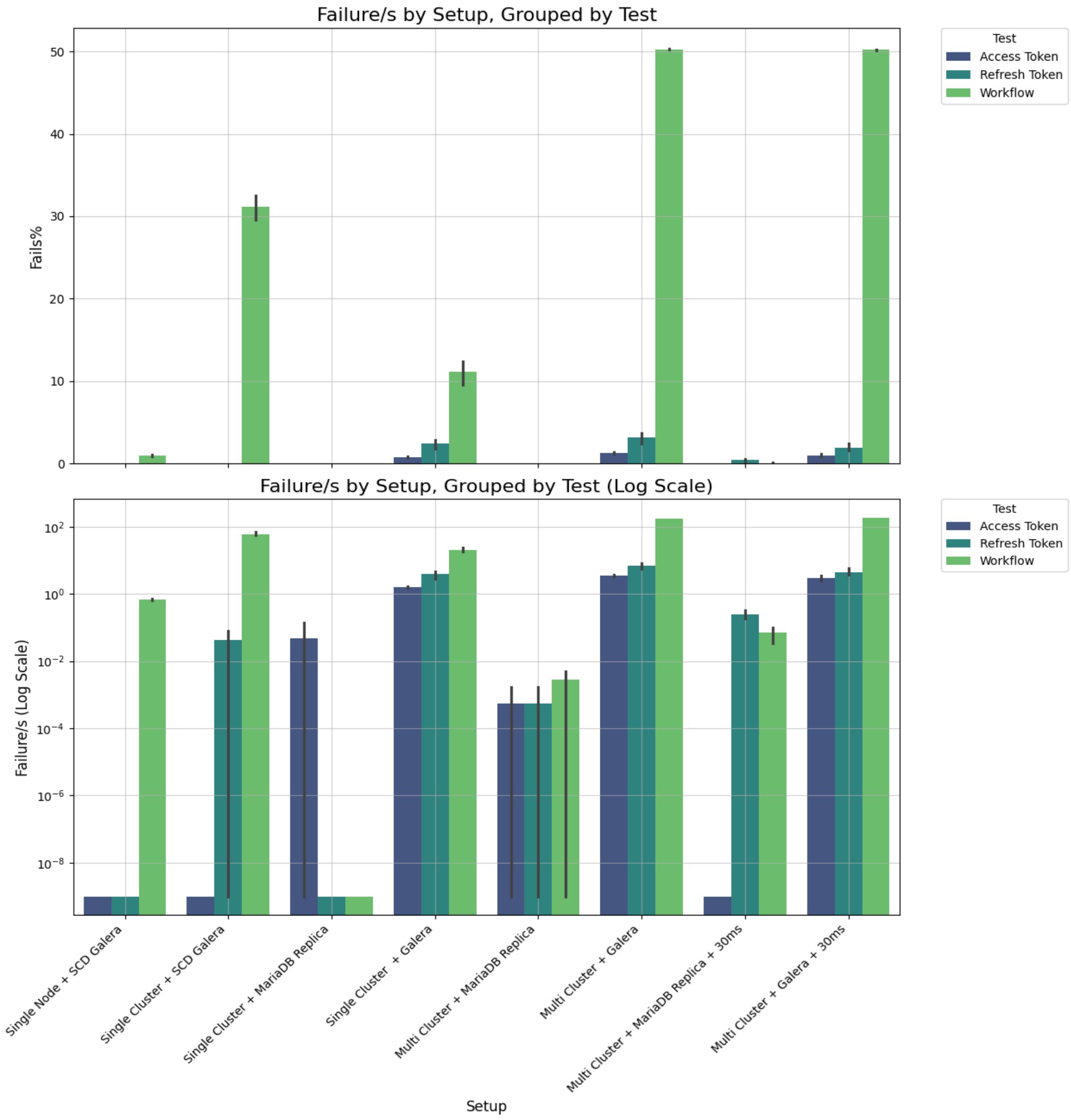


Science and Technology Facilities Council

Scientific Computing

Findings

- Failure rate Galera backend
  - Proportional to
    - number of member galera
    - Number of frontend
  - Potential causes
    - Lack of Global lock for cluster (Error 500)
      - Forcing rollback when conflicting write happens
    - INDIGO IAM no critical read support (Error 400, Error 401)
      - IAM read from a node that is not synced up with the latest write
  - Inconsistency with DB cluster
  - Depends on workload
    - Issuing access token only involve static read + insert
    - Workflow involves additional referencing of data written immediately after



Findings

- Delay in web response when 30ms latency introduced
  - Sentinel is a master replica structure
  - Highly likely that any query is subjected to cross data centre

Conclusion

- Using K8s
    - Improve performance
    - Ease of management
        - Recovery
        - Upgrading
- Multi-cluster geo distributed setup
    - Done securely
    - No performance degradation compared to existing baseline
    - Improve availability
- Request hosting of foreign cluster for IAM.

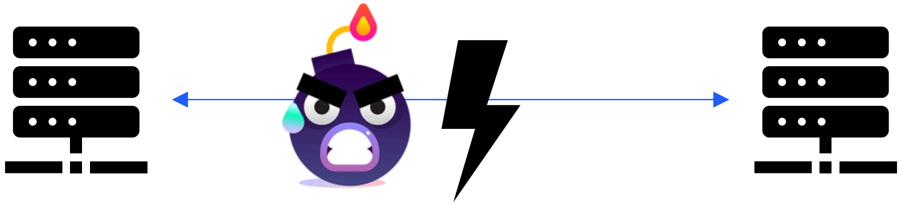
UKRI Science and Technology Facilities Council

Scientific Computing

Multi-cluster Setup beyond IAM

Managing edge cluster

- K8s cluster on edge, placed near field equipment (detector)
- Normal Kubernetes extension
  - Network resources (high latency, low bandwidth)
    - Not plentiful
  - Security
    - Not strictly on site
    - Field equipment may be listened or tampered if communication not encrypted
  - Non uniform cluster setup
    - Different routing and CIDR for pods and services
    - Different storage implementation



Science and Technology Facilities Council

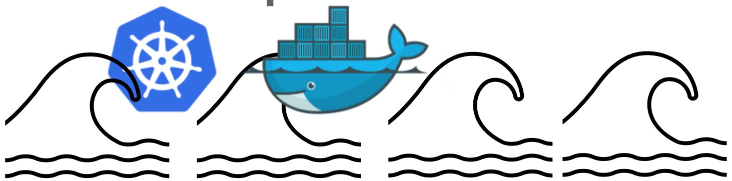
Scientific Computing

Managing edge cluster

- Mult cluster setup mitigates these issue
  - Communication are subject to environmental Hazzard
    - Allows ad-hoc joining of edge cluster
    - Up to 200ms latency between clusters
  - Re-establish communication amongst each other
  - Member cluster are made aware of each others configuration
    - Auto routing or NAT for pods and services between cluster CIDR (*)
      - Some applications don’t work with NAT
    - Mirrored storage class that follows the pods on a cluster
      - Casted to local cluster preferred storage class
      - No need to be aware of all storage engine downstream

Focus on resource-efficiency

- Leverage resources across research partners and public cloud providers



- Focus on higher value-added activity
  - GPU compute which has a high mark-up on GPU but lower mark up for CPU
  - Bursting of CPU workload into public cloud maybe cost efficient
- Not tied to specific vendors
  - Liqo is installed via helm and compatible with K8s compliant cluster
  - In the event of vendor switch, service can be migrated with lower disruption

UKRI Science and Technology Facilities Council

Scientific Computing

Questions?

Science and Technology Facilities Council

Scientific Computing

Thank you

scd.stfc.ac.uk

@SciComp_STFC