Jefferson Lab Scientific Computing Infrastructure Update

CLAS Collaboration Meeting

June 2024

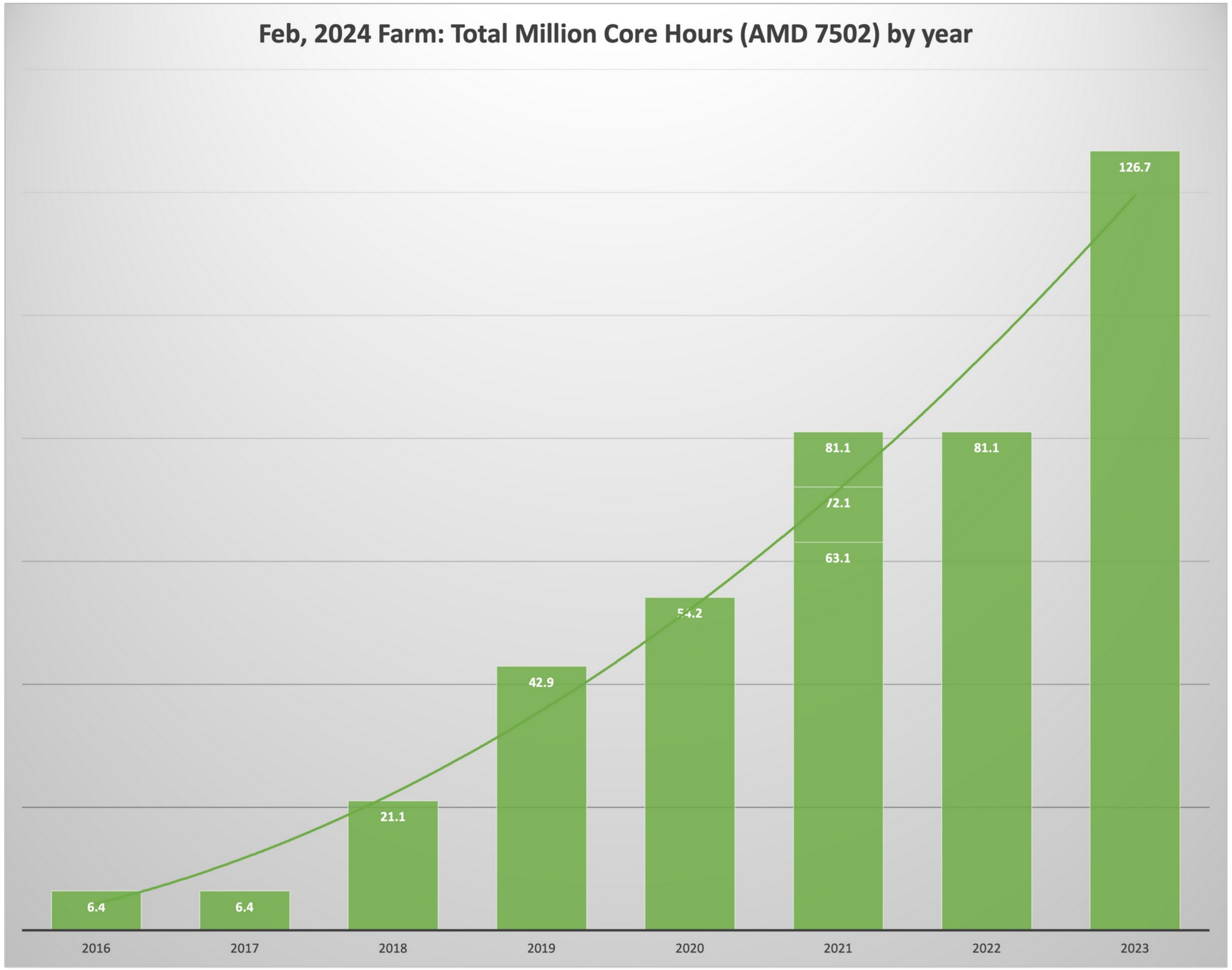
Brad Sawatzky

Tuesday, June 25, 2024

Jefferson Lab’s High Throughput Computing – The Farm

- FY23 3072 EPYC 7763 AMD “Milan” cores added
- FY19 – FY21 Were EPYC 7502 “Rome”
- 140 Million Core Hour/yr capacity
  - Using 7502 as norm.
- Planning for Farm node purchase in FY25

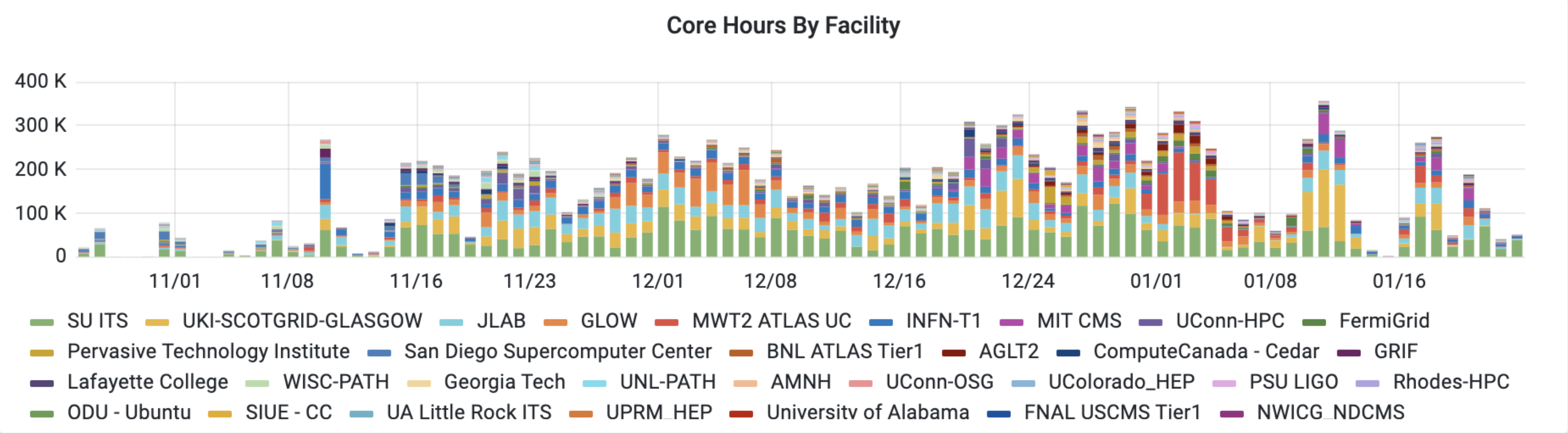
- The farm is routinely busy, Utilization is almost always over 80%
- FairShare allocations are used to balance consumption between the halls
  - B&D ~40% each
  - A&C ~10% each
  - Bursts beyond share when cycles are free
  - CLAS routinely claims free cycles.
  - NB: Hall A data volumes and analysis requirements are rising to Hall B/D levels (SBS program, Moller)
    - Allocations will be rebalanced in Fall 2024
    - A,B,D: ~30% each; C: ~10%



Feb, 2024 Farm: Total Million Core Hours (AMD 7502) by year

Open Science Grid Processing

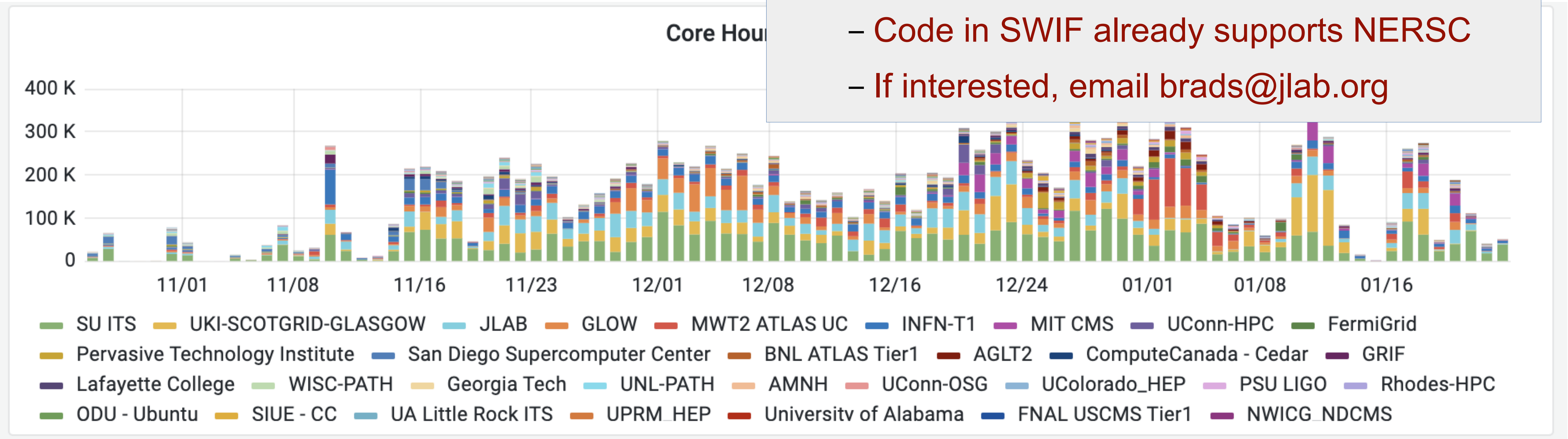
- The Open Science Grid continues to be a significant resource for Monte Carlo Simulation Compute Cycles. GlueX and CLAS12 are significant consumers of CPU cycles.
- Newer additions include EIC and Moller. Hall A/SBS is encouraged to get on-board too



Open Science Grid Processing

- The Open Science Grid continues to be a significant resource for Monte Carlo Simulation Compute Cycles. GlueX and CLAS12 are significant consumers of CPU cycles.
- Newer additions include EIC and Moller.

- NERSC has cycles available on Perlmutter at no cost (but must apply); Lawrencium not ‘free’ but arrangements can be made
  - Code in SWIF already supports NERSC
  - If interested, email brads@jlab.org



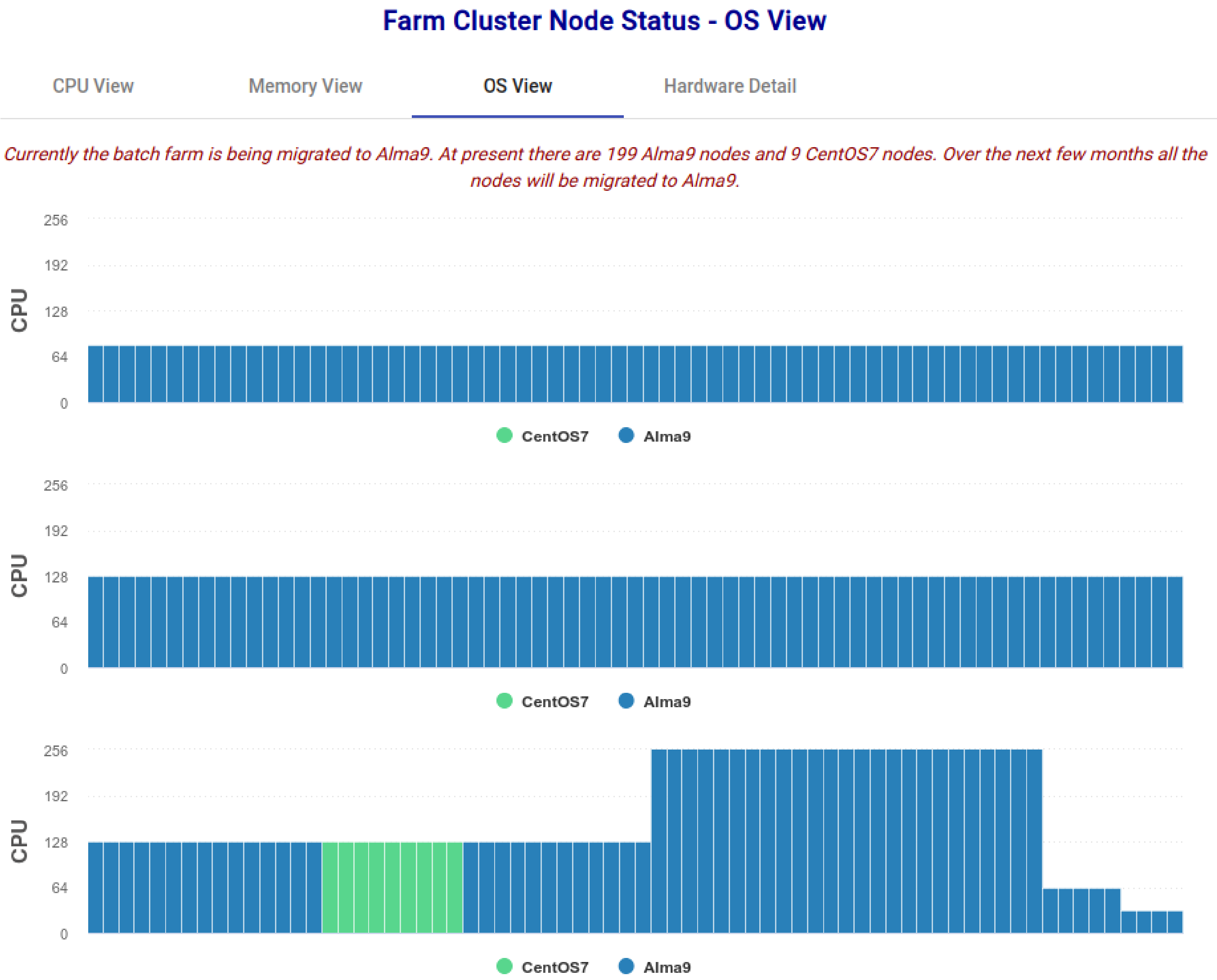
Infrastructure Updates (SW): 2024–25

- Farm transition to Alma9
  - EL7 will disappear in a month
- code.jlab.org
  - CI/CD
  - Container registry
  - JLab GitHub Org will remain while cost-effective

- Kubernetes for workflows that don’t fit Batch model
  - OpenShift ‘enterprise’ K8 platform being rolled out as we speak
    - CI/CD (above) is first target
    - more general (case-by-case) availability over summer
- Building out off-site compute support
  - GlueX/CLAS12 already significant users of OSG
  - Hall A?
- Rucio
  - Distributed (large-file) data management framework
  - “alpha”-testing under way
    - JLab MSS/tape integration in progress
- JLab Research DB
  - “One stop shop” to locate data, publications, workflow information, logbook references, etc...

RHEL7 → RHEL9/Alma9 (Farm Transition)

- Farm OS is transitioning from
  CentOS7 (~RHEL7) → **Alma9** (~RHEL9)
  - → Mostly there; last few nodes to change-over on July maintenance day
- (Much) newer default software, but be mindful of changes
  - → 'ssh ifarm9' for Alma9 interactive node
    - » 'ssh ifarm' will soon point at el9 ifarm
  - → 'default' of el7 changed to el9 recently
    - » Use:
      swif2 add-job -constraint el9 <other arguments>
    - » SWIF notes
    - » Slurm notes
  - → */site, /apps* no longer mounted on farm nodes
    - » use 'environment modules' framework (SW modules under /cvmfs, /group)
      - – run 'modules avail'
    - » If something is missing, contact your Hall Compute Coordinator and/or open a Helpdesk ticket



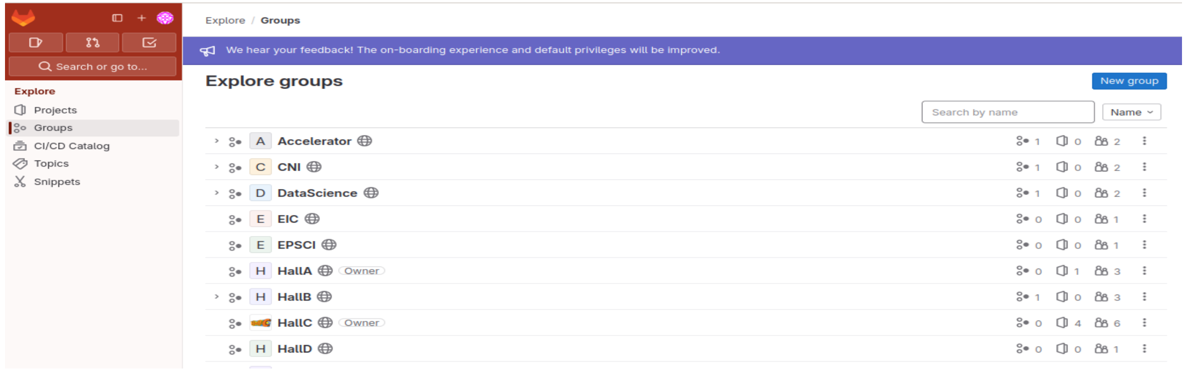
Farm Cluster Node Status - OS View

Currently the batch farm is being migrated to Alma9. At present there are 199 Alma9 nodes and 9 CentOS7 nodes. Over the next few months all the nodes will be migrated to Alma9.

code.jlab.org (GitLab Service)

- GitHub is getting $$$
  - CI/CD, storage, etc are all metered costs
  - JLab is on a ‘legacy’ license model for now but limitations are frustrating
- JeffersonLab GitHub Organization will be maintained as-is
  - BUT goal is for code.jlab.org to be a “value-added” proposition

- code.jlab.org (GitLab instance)
  - JLab run/managed
  - Open / Offsite access
    - Federated logins avail.
  - CI/CD and Storage can leverage our Farm
  - Built-in Container Registry
  - Supports several Data Management requirements important to JLab / PhysDiv



Explore / Groups

We hear your feedback! The on-boarding experience and default privileges will be improved.

Explore groups

A	Accelerator	1	0	2
C	CNI	1	0	2
D	DataScience	1	0	2
E	EIC	0	0	1
E	EPSCI	0	0	1
H	HallA Owner	0	1	3
H	HallB	1	0	3
	HallC Owner	0	4	6
H	HallD	0	0	1

code.jlab.org (GitLab Service) Cont...

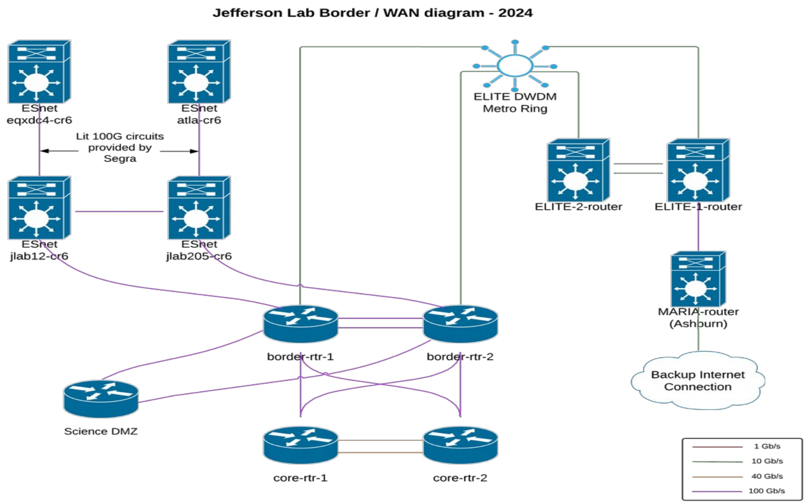
- Early User testing phase for JLab gitlab server: **`code.jlab.org`**
  - *"GitHub style"* repository hosting system
  - Current GitHub license will be maintained for as long as Microsoft lets us
- Features:
  - Standard git repo features, pull-requests, forking, etc
  - Trouble/bug-ticket reporting system, etc.
  - GitLab Pages: website/wiki features
  - Integrated support for Container Registry
  - Login/ access with home institution credentials, including authorization approval step for non-jlab accounts (open from off-site soon)

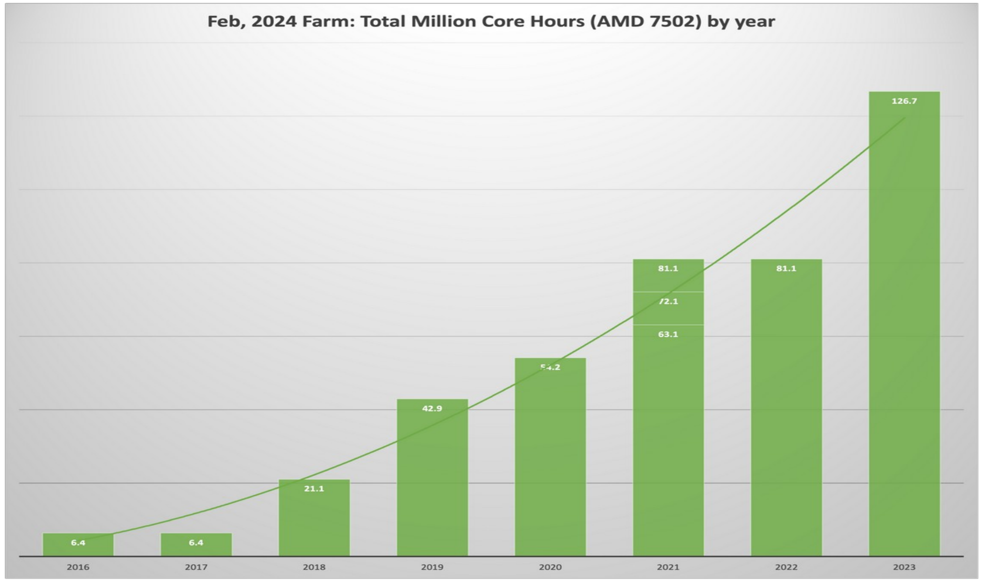
- Coming soon (end of June!):
  - Greatly improved on-boarding for JLab (and other) users
  - CI/CD support using Kubernetes
    - Takes advantage of RH Openshift framework; in deployment now

Deployment was delayed by new LCDQ cluster online + recent Alma9 related Lustre stability issues...

Recent and Near-term Infrastructure Updates (HW) : 2024–25

- JLab WAN connection
  - 2x10 Gbit → 2x100 Gbit
    - → 2x400 Gbit planned (2025/6)
- Significant disk space increases
  - /cache, /volatile will increase by 3–4x (“Lustre24” upgrade)
  - “/work” (and SciComp /group?) → “/sciproject” with upgraded HW (Fall 2024)
- Additional Tape Drives on order
  - increased throughput/capacity
- CPU purchase next year (FY25)
  - Mostly CPUs, but GPUs are an option if they will be used


Jefferson Lab Border / WAN diagram - 2024


Feb, 2024 Farm: Total Million Core Hours (AMD 7502) by year

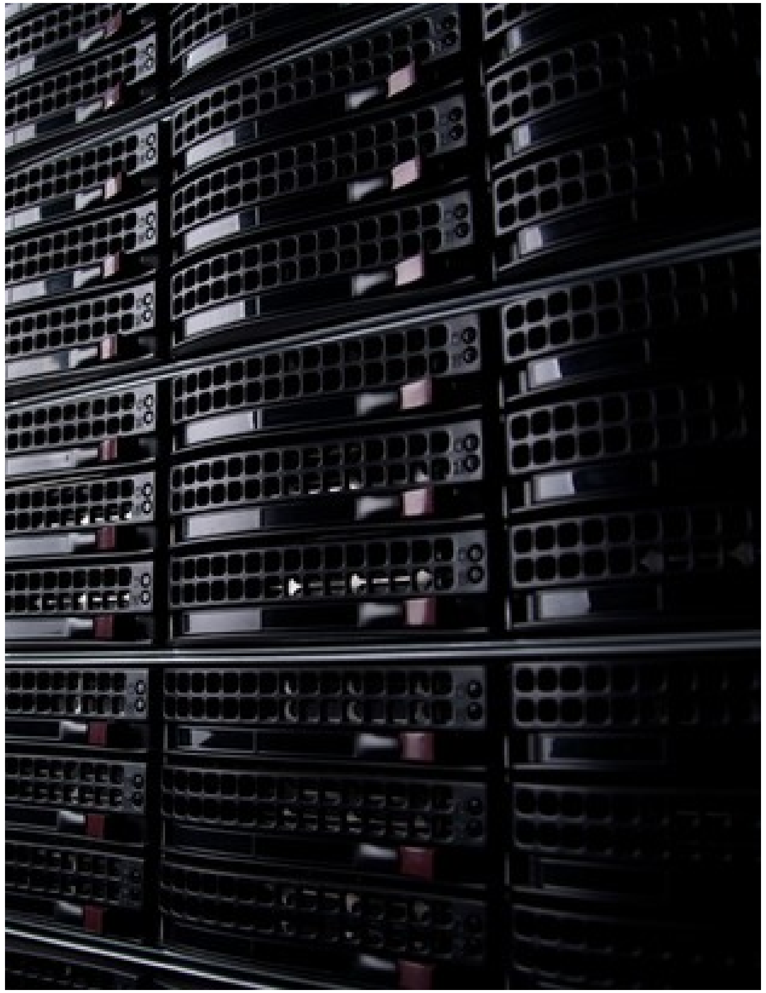
Disk Storage Areas and Their Uses

- *There is strong demand for more disk. This is an explicit priority for for us.*
  - High performance, reliable disk and associated infrastructure is still expensive and has been long lead…
  - We are trying to get "ahead of the curve" on disk with FY23, and FY24 purchases
- Lustre Storage is good for large files, streaming, large block I/O, production farm runs.
- Lustre is *not* good for small files, high IOPS, and frequent metadata operations (worst case: open, write 1kB, close, repeat)
- /work will not scale for large farm campaigns.
- Node-local /scratch is good for jobs with high IOPS to working files.
  - *Note: SWIF-declared MSS files are automatically copied to node-local working directory*
  - *Old GlueX wrappers still doing this manually*
- We are reevaluating the role of /work areas as legacy NFS spinning disk storage.
  - *New "/sciproject" space to merge scicomp /group + /work being evaluated...*
  - *More on this later...*

Path	Best Use	FS Type	Deletion	backup
/cache	Bulk I/O, Migration to tape	Lustre	Once on tape	/mss
/volatile	Bulk I/O Temporary storage	Lustre	auto	NO
/work	Source code, DB files, exe's, etc. User Managed	NFS+ ZFS	manual	NO
/home	Dot files, personal documents, etc	NFS ssd	manual	YES
/farm_out	Farm job stdout/stderr	NFS ssd	auto	NO
/group or /scigroup	Source code Papers, thesis, analysis scripts	NFS ssd	Manual	YES
/scratch	Farm job I/O to node local disk	ssd	auto	NO
/u/scratch	*CUE scratch. Deprecated (Unavailable on el9)*			
/cvmfs	Software stack. Configuration.			

Hardware Deployment Updates

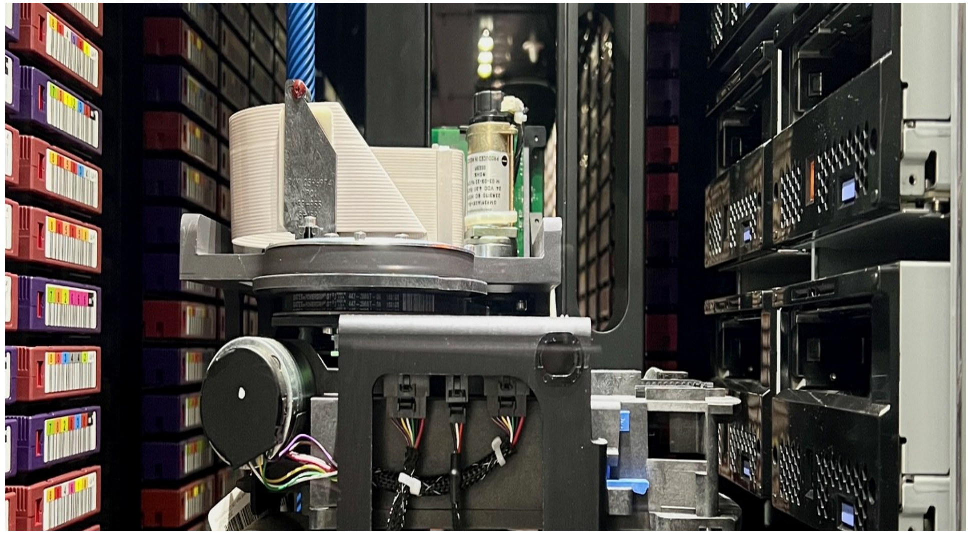
- Lustre 24 Installation (/cache, /volatile)
  - Long delayed by supply-chain issues for several components, but production deployment has (finally!) begun
  - Doubled sustained IO performance
  - Will also double available space on those filesystems → ~10PB
    - Additional disk shelves on order
    - → another +10PB by end of 2025
  - Staged migration of Hall data to new hardware has begun; this will take at least a month to complete.
    - Minor downtimes for selected pathnames under /cache, /volatile during cut-over periods
    - *Hall Compute Coordinators will help schedule any downtimes!*
- NVMe storage evaluation (/work)
  - /work and “/sciproject” (possible replacement for general /group)
    - Expansion → O(1PB) of fast, backed-up storage
  - filesystem evaluation in progress (Weka, CephFS)
    - Got a little stalled behind LQCD cluster roll-out over last few months...



Write-through /cache mechanism going away ...

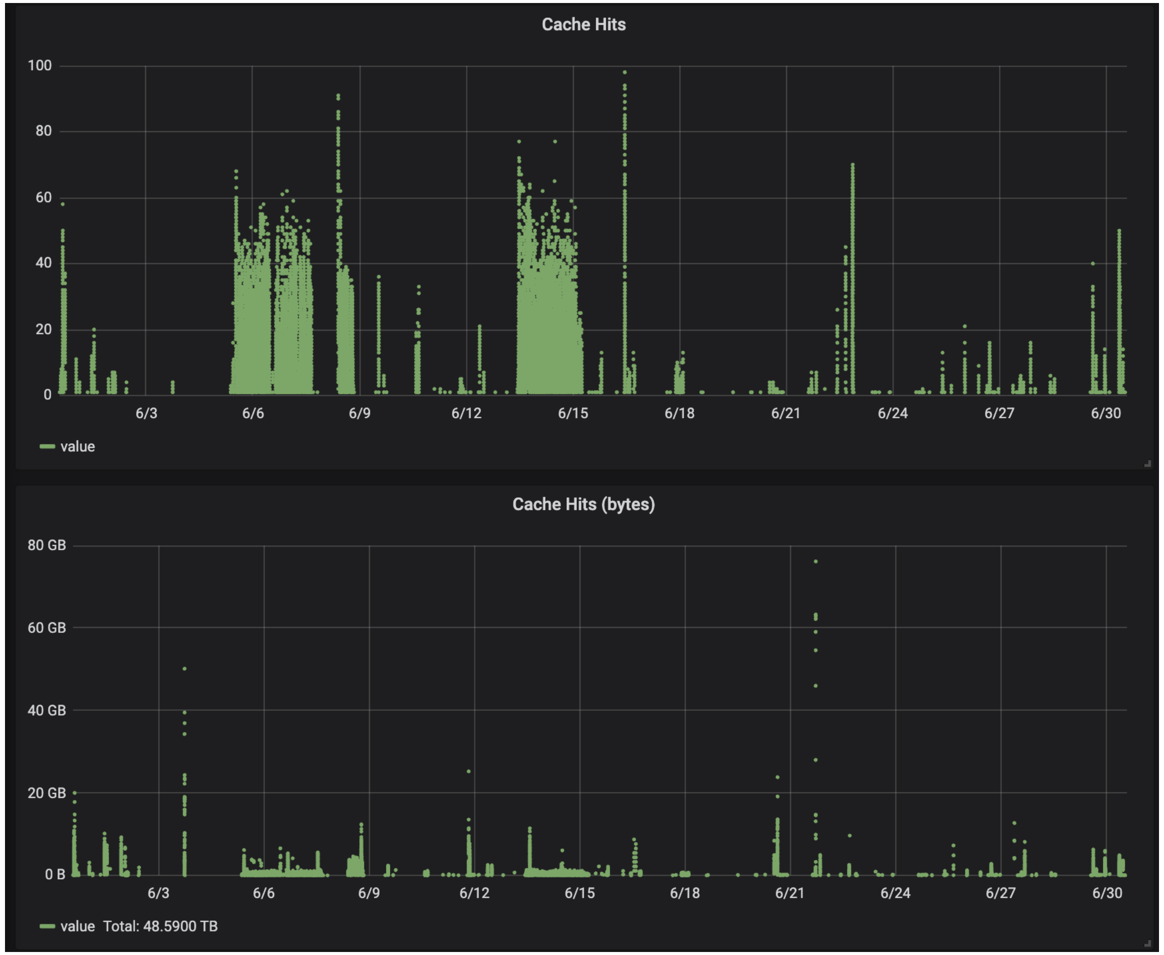
- Originally /cache was a user-facing read-only filesystem to store files located on tape
- In 2014 /cache was made user-writeable to address (in part) challenges with sufficient online storage for analysis campaigns
- However, this has caused a number of complications
  - Small file proliferation: O(10^7)
  - File ownership and permission mismatches
  - ‘Sync’ issues: policy is that items on /cache are backed up to tape *but* there are quite a few corner cases
    - ‘duplicates’ / file name collisions between files on tape and files in /cache
    - delay between file close on disk and file on tape

- *SciComp would like to go back to the read-only model in the fall (TBD!)*
  - Remove user-write permissions
  - Jobs should declare output files in SWIF2
    - System will ensure writes to tape
    - Files will still show up under /cache as soon as job completes



Tape Library Strategy: Write Once, Read Rarely.

- Historically, access to Tape has been a bottleneck; This is no longer true
  - Consolidation of data to LTO8
  - Use of SWIF for data+cpu co-scheduling
  - Expansion of disk storage
- We are at an articulation point where it is becoming possible to cache “hot” data and avoid repeated round-trip churn to tape
- Implementation of a system Read Only Cache (distinct from user-visible cache)
  - Keep all small files disk resident
  - Aggressively cache hot files
  - Use XRootD storage
  - Repurpose EOL storage for RO Cache
  - RO cache failures well-tolerated since storage is not POSIX or user-facing. Worst case, go to tape for the file again.



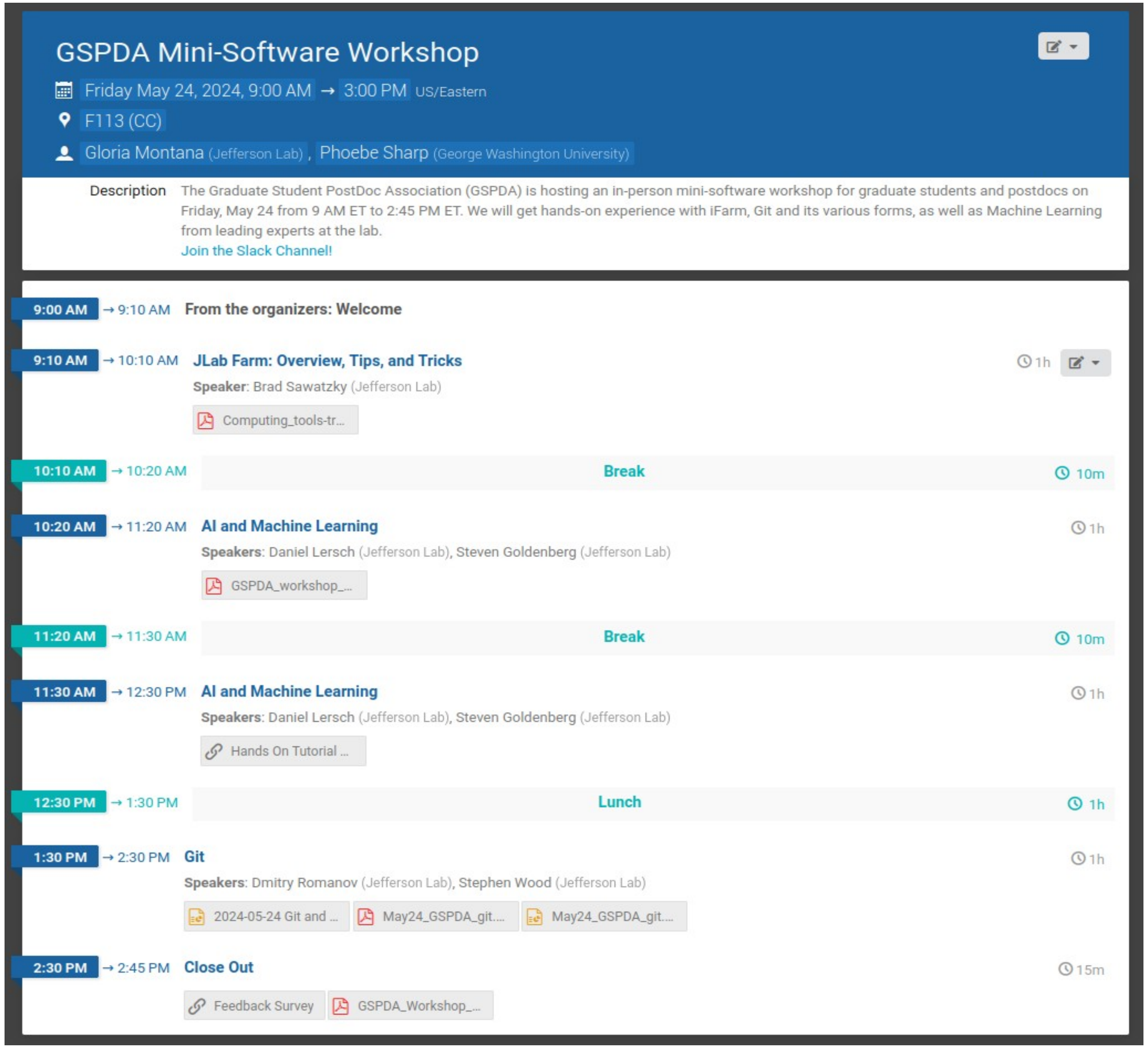
Hall ESX Virtual Machine Cluster

- VMs within the Hall experimental enclave are available
    - High-uptime infrastructure explicitly targeted to support Hall operations, etc.
    - Update policy, downtime scheduling for VMs defined by Hall Compute Coords as usual.
  - Ex: Slow control systems
    - EPICS softIOCs
    - Windows/Rockwell control systems
      - ie. Hall C: cmagnets, skylla10
    - PXE boot hosts/services
  - Ex: Data-base hosts
    - RCDB, CCDB hosts
  - Ex: "Remote CH" support hosts

- VM hosts functionally operate within the Hall subnets (no firewall issues)
  - Direct access within respective subnets
  - 2-factor hop (via. hallgw, etc) as with any existing Hall hosts
- Take advantage of VM flexibility
  - snapshotting / backups
  - auto-failover on HW issues
  - advantages wrt "cloning", load balancing, etc
- Keep this in mind for future deployments, HW upgrades, etc
  - Cheaper / better than HW for many applications

GSPDA Mini-Software Workshop (May 24)

- Thanks to Grad Students PostDoc Assoc. for organizing!
  - Phoebe Sharp (GWU)
  - Gloria Montana (JLab)
    - Indico Page
- This is a *Part 1 of 2.* Not sure when the next one will run (late summer, I think).
- Much smaller than previous year (JSA chose not to provide funding in FY24)
  - Please advocate for funding in FY25!



GSPDA Mini-Software Workshop

Friday May 24, 2024, 9:00 AM → 3:00 PM US/Eastern

F113 (CC)

Gloria Montana (Jefferson Lab) , Phoebe Sharp (George Washington University)

Description The Graduate Student PostDoc Association (GSPDA) is hosting an in-person mini-software workshop for graduate students and postdocs on Friday, May 24 from 9 AM ET to 2:45 PM ET. We will get hands-on experience with iFarm, Git and its various forms, as well as Machine Learning from leading experts at the lab.
Join the Slack Channel!

9:00 AM → 9:10 AM From the organizers: Welcome

9:10 AM → 10:10 AM JLab Farm: Overview, Tips, and Tricks — 1h
Speaker: Brad Sawatzky (Jefferson Lab)
Computing_tools-tr...

10:10 AM → 10:20 AM Break — 10m

10:20 AM → 11:20 AM AI and Machine Learning — 1h
Speakers: Daniel Lersch (Jefferson Lab), Steven Goldenberg (Jefferson Lab)
GSPDA_workshop_...

11:20 AM → 11:30 AM Break — 10m

11:30 AM → 12:30 PM AI and Machine Learning — 1h
Speakers: Daniel Lersch (Jefferson Lab), Steven Goldenberg (Jefferson Lab)
Hands On Tutorial ...

12:30 PM → 1:30 PM Lunch — 1h

1:30 PM → 2:30 PM Git — 1h
Speakers: Dmitry Romanov (Jefferson Lab), Stephen Wood (Jefferson Lab)
2024-05-24 Git and ... May24_GSPDA_git.... May24_GSPDA_git....

2:30 PM → 2:45 PM Close Out — 15m
Feedback Survey GSPDA_Workshop_...

I want to hear from you

- What are the problems / pain-points in your workflows?
- Other question / comments?

Thank you!

Rucio

- Distributed data management system
  - Initially developed for ATLAS
  - Highly scalable & modular
- Features
  - Basic data operations
    - Storage, transfer, deletion
  - Policy based replication (automatic)
  - Designed with distributed storage and ‘protocol agnostic’ data transfer methods in mind
- Can serve as a Replica/File catalog and metadata service
  - Can search its DB and present files matching names, metadata, etc

- “Beta” JLab Rucio Instance under development
  - Initial customers will be EIC group
    - JLab ↔ BNL automated file registration and transport
  - GlueX (modest sub-project)
  - Load testing, solve authentication challenges, develop policy and namespace conventions
  - Integrate relevant metadata from RunDB and other sources
- Goal is progressive, but full fledged roll-out in 2024
  - Transparent Jasmine/tape integration
  - Full offsite DB query and data transport functionality
  - Backfill from existing tape library as needed

Containerization Support

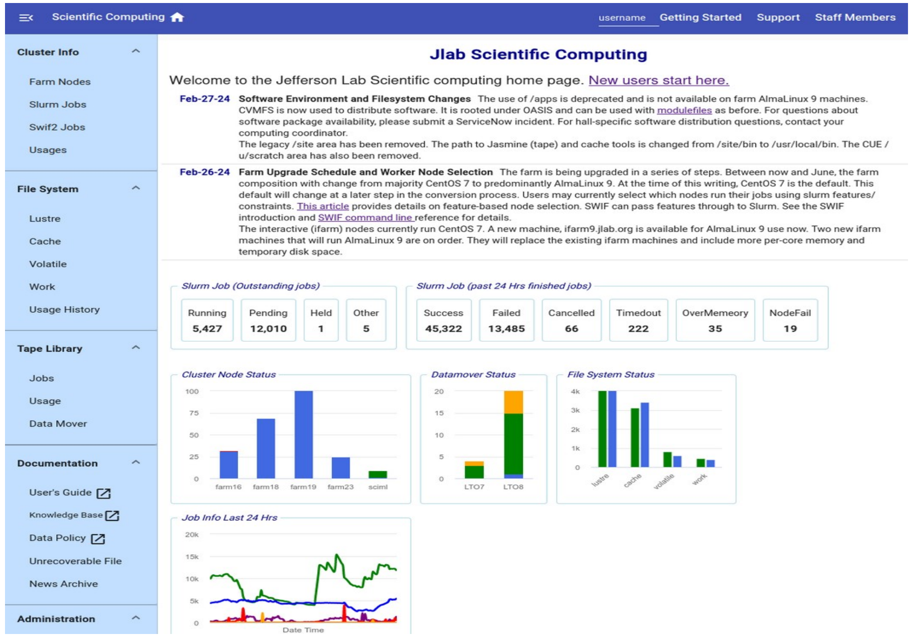
- CST is developing formal containerization support for Users
  - Documentation
    - ie. Easy ‘on-ramp’ / how-to for common use cases
  - “Official” Infrastructure support
    - Apptainer / Singularity
    - Docker, Podman support for image building and deployment
    - JLab GitLab Container Registry
- Among other benefits, Containers can provide
  - ‘Plug and play’ software configurations
  - SW version / configuration snapshots
  - Ability to run ‘custom’ software frameworks on other datacenters, computers, laptops
- Will also support/streamline upcoming Farm transition from RHELX → Alma9

WORK IN PROGRESS

- Apptainer (was Singularity)
  - works on both ifarm and farm
- Podman
  - works on ifarm9 now
  - will work on alma9 farm soon (this summer)
- Docker
  - not happening on compute clusters
  - but podman == docker (pretty much)
  - Note: docker != dockerhub

Information Resources

- scicomp.jlab.org
  - →SciComp web page
- scicomp-briefs
  - →mailing list for JLab Scientific Computing



Jlab Scientific Computing

Welcome to the Jefferson Lab Scientific computing home page. New users start here.

Feb-27-24 **Software Environment and Filesystem Changes** The use of /apps is deprecated and is not available on farm AlmaLinux 9 machines. CVMFS is now used to distribute software. It is rooted under OASIS and can be used with modulefiles as before. For questions about software package availability, please submit a ServiceNow incident. For hall-specific software distribution questions, contact your computing coordinator.
The legacy /site area has been removed. The path to Jasmine (tape) and cache tools is changed from /site/bin to /usr/local/bin. The CUE / u/scratch area has also been removed.

Feb-26-24 **Farm Upgrade Schedule and Worker Node Selection** The farm is being upgraded in a series of steps. Between now and June, the farm composition with change from majority CentOS 7 to predominantly AlmaLinux 9. At the time of this writing, CentOS 7 is the default. This default will change at a later step in the conversion process. Users may currently select which nodes run their jobs using slurm features/ constraints. This article provides details on feature-based node selection. SWIF can pass features through to Slurm. See the SWIF introduction and SWIF command line reference for details.
The interactive (ifarm) nodes currently run CentOS 7. A new machine, ifarm9.jlab.org is available for AlmaLinux 9 use now. Two new ifarm machines that will run AlmaLinux 9 are on order. They will replace the existing ifarm machines and include more per-core memory and temporary disk space.

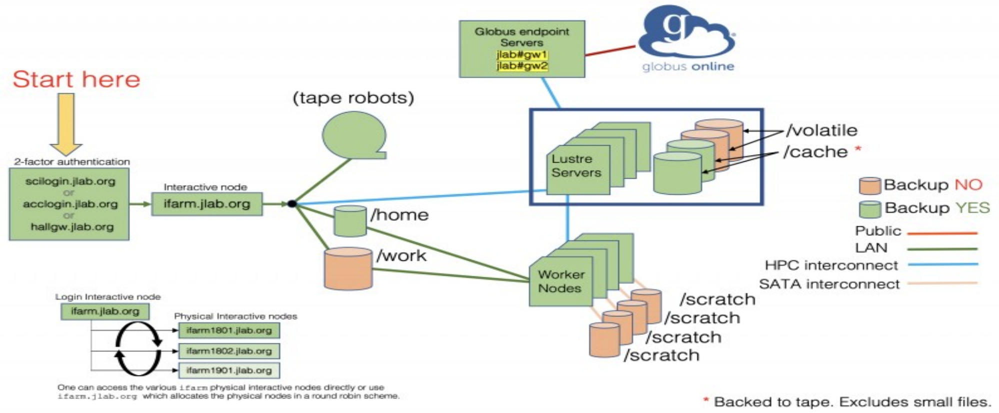
Slurm Job (Outstanding jobs)

Running	Pending	Held	Other
5,427	12,010	1	5

Slurm Job (past 24 Hrs finished jobs)

Success	Failed	Cancelled	Timedout	OverMemeory	NodeFail
45,322	13,485	66	222	35	19

- Documentation links
  - →Getting Started
  - →SciComp Knowledge Base
  - →CST User Portal
  - →JLab Helpdesk
    - » helpdesk@jlab.org
    - » Incident Request



One can access the various ifarm physical interactive nodes directly or use ifarm.jlab.org which allocates the physical nodes in a round robin scheme.

* Backed to tape. Excludes small files.

Improving Data Management at JLab

- There are a number of ongoing challenges with Data Management at JLab (and elsewhere!)
  - Difficulty capturing the analysis workflow
    - Software toolchain, metadata and calibrations, etc
    - Difficult/impossible to revisit prior ‘working’ code for comparison and cross-checks
  - Raw and Processed data locations may be insufficiently documented
  - Experimental metadata is being scattered more broadly as groups develop distributed and cloud-supported workflows (outside historical lab-provided frameworks)
    - Google Workspace/Groups vs. O365/Teams; Instant Messaging (Slack, Discord, SMS, etc); University/Institution provided wikis, document repos, etc.
  - We must to provide the right combination of Training, Policy, *and* appropriate software tools so the Users/Collaborations *want* to “stay in the fold”.

Challenges

Improving Data Management at JLab (2)

- Sustainable software and archival support (Containers)
  - Develop infrastructure to make it easier to capture, snapshot, archive, and restore software and workflows.
  - Improve infrastructure support and documentation for Containerizing workflows incl. “How-to’s”, template containers for existing workflows, etc.
  - Provide a Container registry (w/ history), and supporting git-backed code repos
    - Much of this supported by new User-facing GitLab instance managed by JLab (full CI/CD support as well)
    - VM snapshots of Farm environments as 2nd layer of defense on running old code / containers
- Get a handle on what we do have and make it findable
  - New initiative: JLab Experimental Research DB

Building Blocks for the Solution

Improving Data Management at JLab (3)

- “JLab Experimental Research DB” is in the early stages of development
  - Directly supported by recent hire in PhysDiv (Anil Panta)
  - Provide a ‘1-stop shop’ to store and search information associated with Experiments executed at JLab
    - Provide a searchable database that contains a comprehensive set of information about an experiment.
      - Targets, beam characteristics, kinematics, reaction info
      - Proposals, papers, theses/dissertations, technical documents
        - machine readable “results” databases where available
      - References to raw and processed file locations
      - References to analysis software and workflow Containers
      - References to meta-data sources: RunDBs, Config DBs, Logbooks, Wikis, etc.
- We will take it in bite-sized steps that still provide value to the Lab and User community as a whole. Achievable and still useful will be our guiding principle.
  - Start by cross-referencing existing (but scattered) databases / data sources on and off-site
    - JLab Publication DB, proposal DB, experimental logbooks, wikis, web-sites, MSS URIs/paths (later Rucio datasets), JLab filesystem paths to working environments, etc
    - Reference software snapshots/workflows in the Container Registry
- <u>**This will be a long term project that we will refine as we go.**</u>

Building Blocks for the Solution

Improving Data Management at JLab (4)

- Continue to identify gaps in what we are providing to Users and address them
  - Instant Messaging (Slack, Discord, Teams, SMS, etc) has come to provide very valuable “real-time” support both when an experiment is on the floor and during analysis.
    - Too much “logbook-worthy” information is getting lost here.
    - We’re evaluating our options (Teams for ‘all’, Cloud service licensing, Mattermost, etc.)

Building Blocks for the Solution