CLAS12 Software Working Group

N. Baltzell

11/12/2024

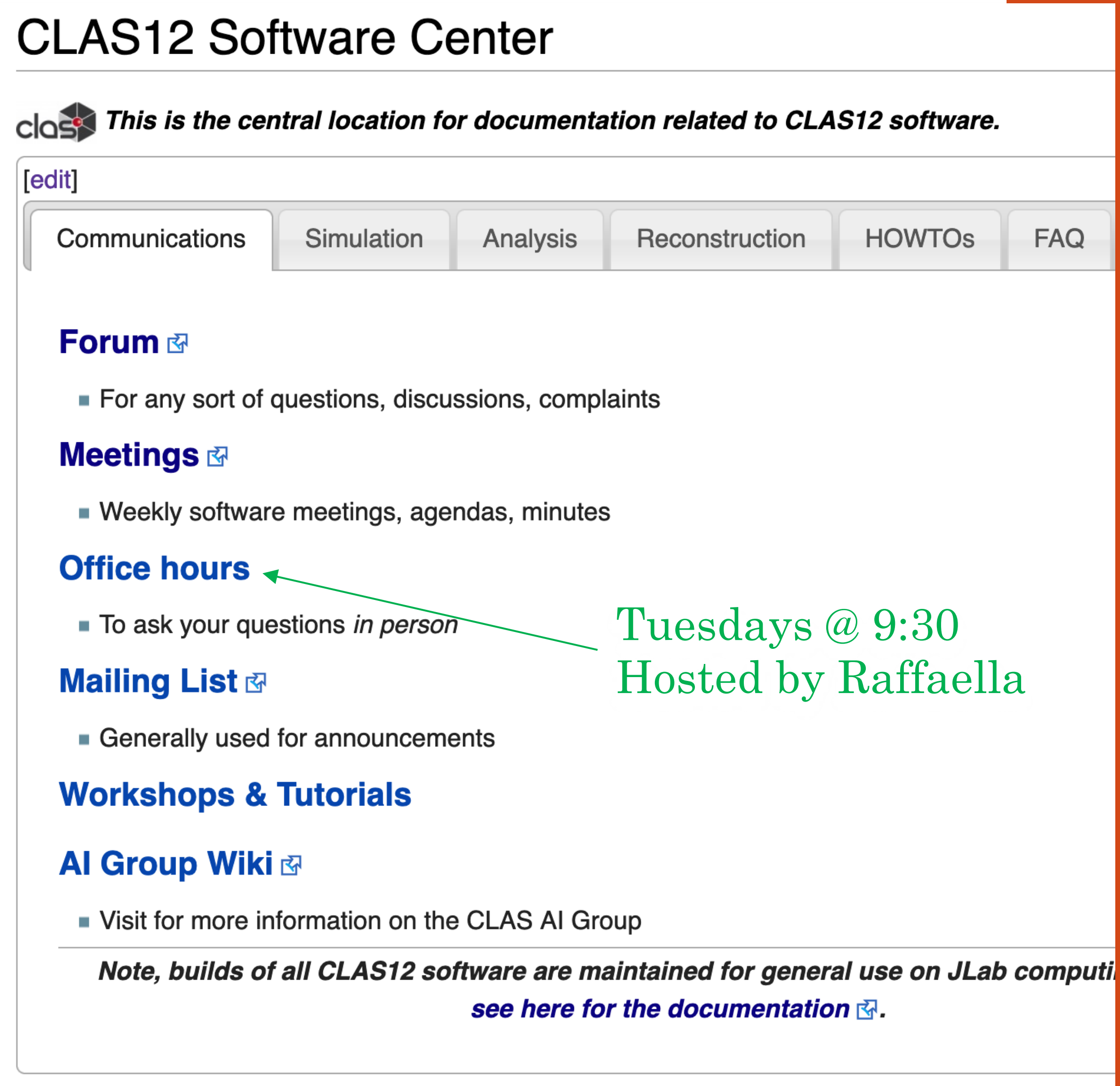
CLAS Collaboration Meeting

Overview

- Getting Started
  - CLAS12 Software Wiki
  - JLab Scicomp Documentation
  - CLAS12 Discourse Forum
  - GITLAB@JLAB
- Ongoing Projects
  - Real-Run-Number Simulations
  - DC V2
  - GEMC 5.11
  - CI/CD/CR with GitLab@JLab
- Best Practices Proposal
- CLAS6

Software Wiki

- Was intended as the main landing page for finding documentation for general-purpose CLAS12 software and computing stuff
- Due for a refresh, reorg, find and fix dead links, …
- Needs an onboarding section? Already exists somewhere?
- Volunteers? Could be service work …



CLAS12 Software Center

This is the central location for documentation related to CLAS12 software.

[edit]

Forum

- For any sort of questions, discussions, complaints

Meetings

- Weekly software meetings, agendas, minutes

Office hours

- To ask your questions *in person*

Mailing List

- Generally used for announcements

Workshops & Tutorials

AI Group Wiki

- Visit for more information on the CLAS AI Group

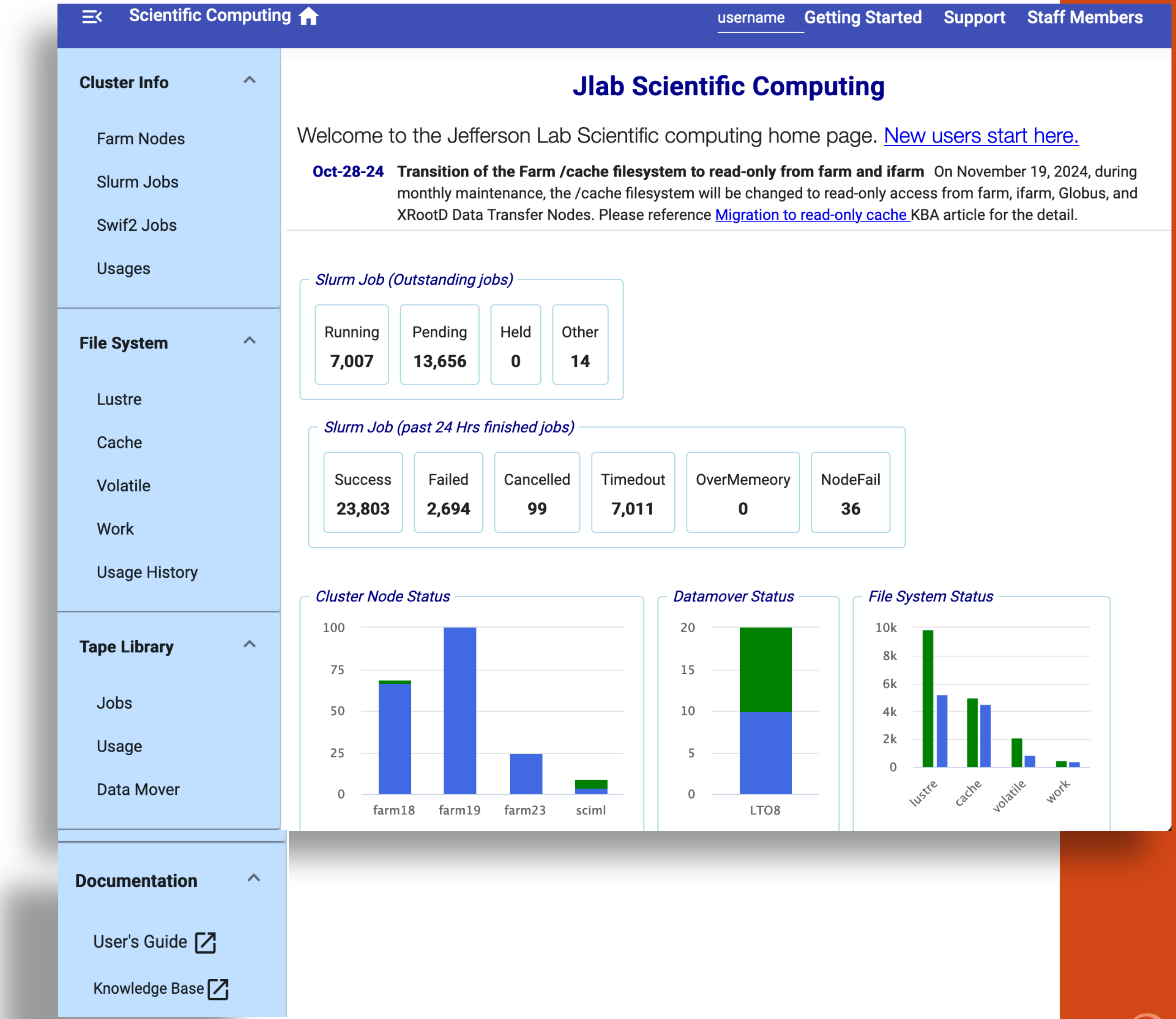
Note, builds of all CLAS12 software are maintained for general use on JLab computi

see here for the documentation.

Tuesdays @ 9:30
Hosted by Raffaella

JLab Scicomp

- Lots of upgrades and improvements in recent years
  - https://scicomp.jlab.org
    - SLURM, SWIF, JupyterHub, …
    - Documentation, documentation …
  - There's also the more "non-scicomp" https://cc.jlab.org/, e.g. /home quotas
- Use ServiceNow for reporting system issues, enrolling in 2-factor, etc:
  - https://jlab.servicenowservices.com
- Read-only /cache and 24-hour job limits coming soon, ongoing projects include GitLab, CI/CD, Rucio, 2025 farm node purchases



Jlab Scientific Computing

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

Oct-28-24 **Transition of the Farm /cache filesystem to read-only from farm and ifarm** On November 19, 2024, during monthly maintenance, the /cache filesystem will be changed to read-only access from farm, ifarm, Globus, and XRootD Data Transfer Nodes. Please reference Migration to read-only cache KBA article for the detail.

Slurm Job (Outstanding jobs)

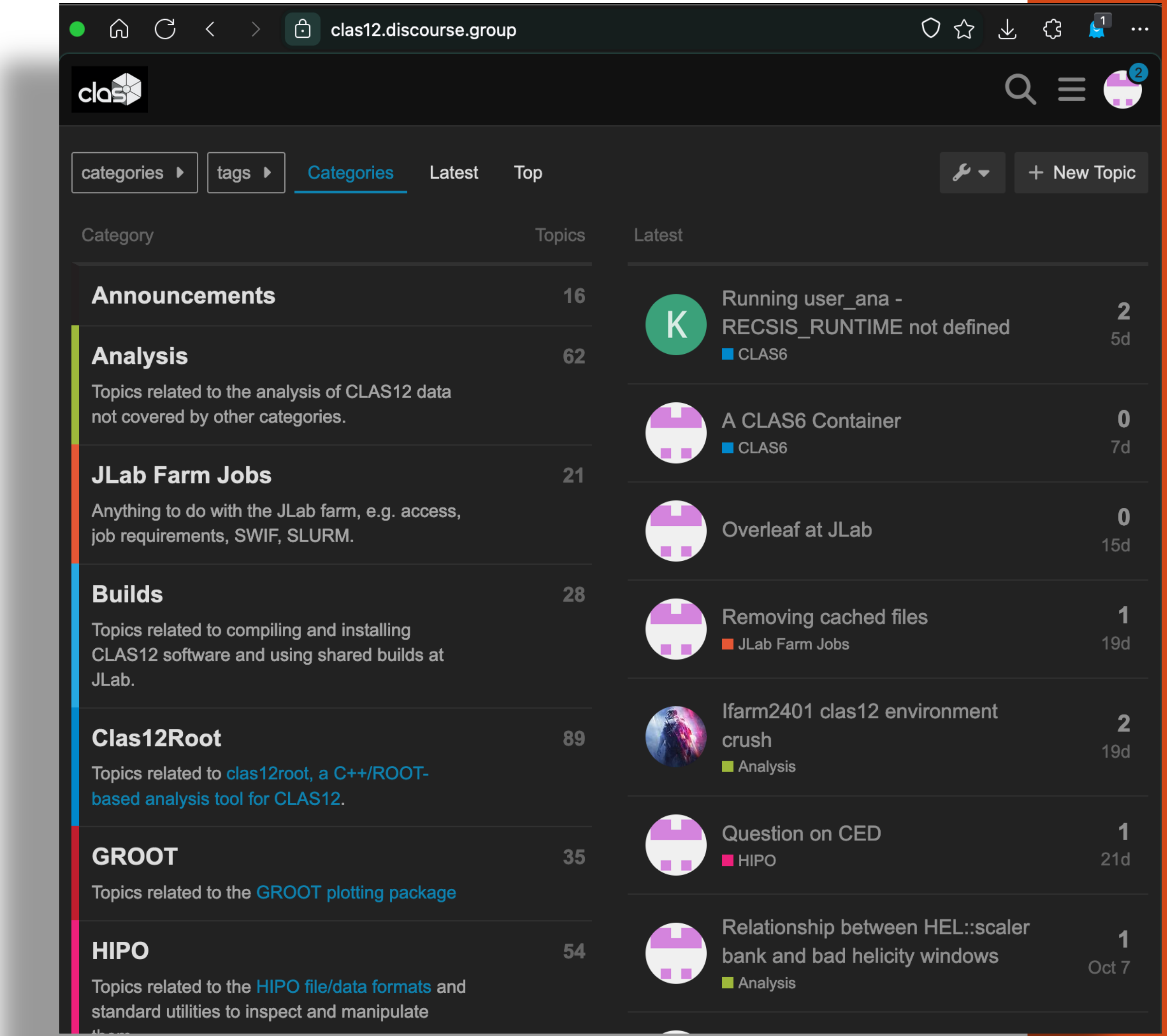
Running	Pending	Held	Other
7,007	13,656	0	14

Slurm Job (past 24 Hrs finished jobs)

Success	Failed	Cancelled	Timedout	OverMemeory	NodeFail
23,803	2,694	99	7,011	0	36

Discourse Forum

- We applied for and received a free one years ago
  - https://clas12.discourse.group
- Use it!
- To avoid bothering with hitting service limits, an account is required (most people use their GitHub account), although we could change that.
- In the past year, per month we average
  - 2 new contributors and users
  - 1k page views
  - 30 posts

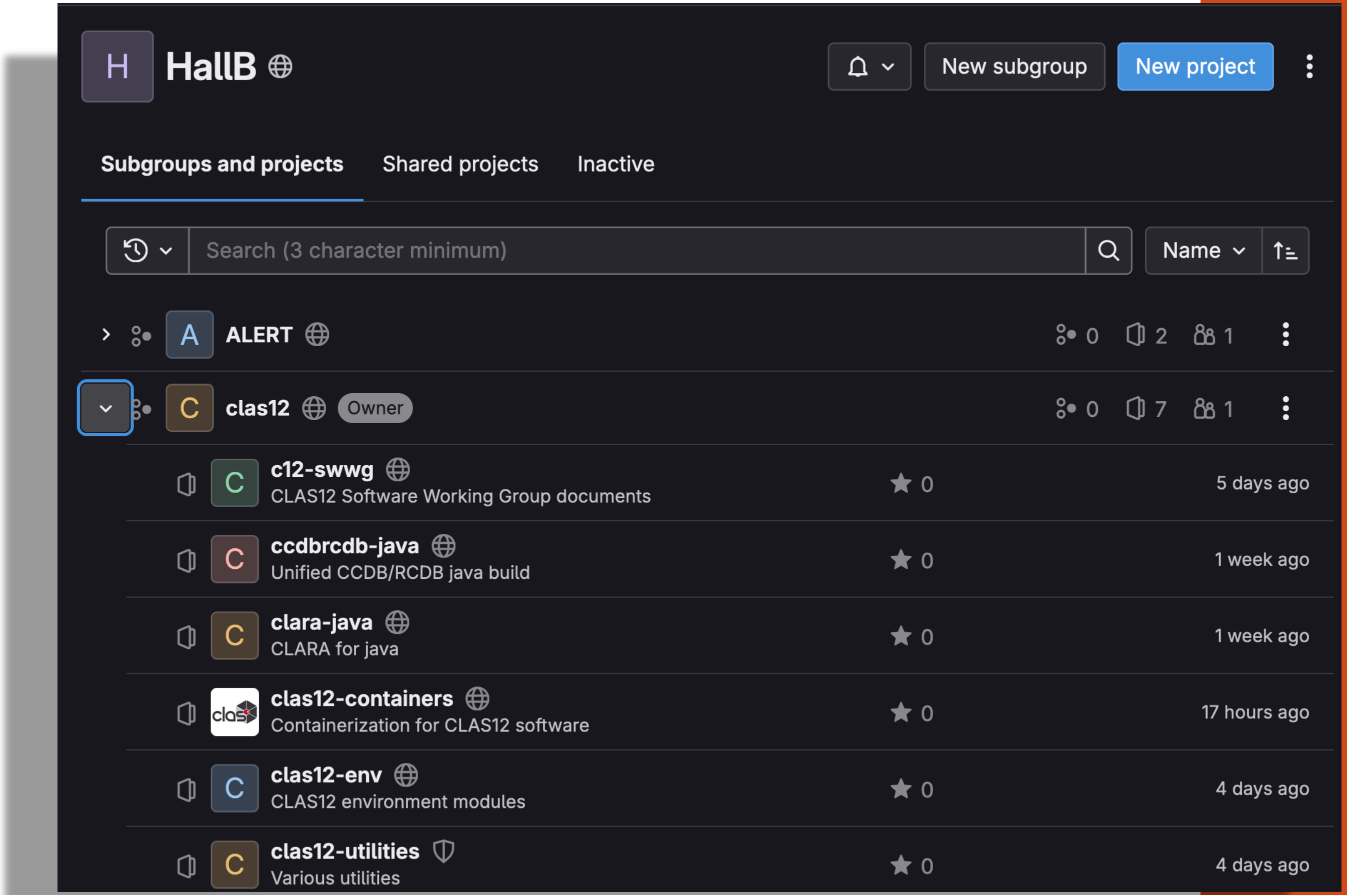


Category	Topics
Announcements	16
Analysis Topics related to the analysis of CLAS12 data not covered by other categories.	62
JLab Farm Jobs Anything to do with the JLab farm, e.g. access, job requirements, SWIF, SLURM.	21
Builds Topics related to compiling and installing CLAS12 software and using shared builds at JLab.	28
Clas12Root Topics related to clas12root, a C++/ROOT-based analysis tool for CLAS12.	89
GROOT Topics related to the GROOT plotting package	35
HIPO Topics related to the HIPO file/data formats and standard utilities to inspect and manipulate them.	54

Latest		
Running user_ana - RECSIS_RUNTIME not defined CLAS6	2	5d
A CLAS6 Container CLAS6	0	7d
Overleaf at JLab	0	15d
Removing cached files JLab Farm Jobs	1	19d
Ifarm2401 clas12 environment crush Analysis	2	19d
Question on CED HIPO	1	21d
Relationship between HEL::scaler bank and bad helicity windows Analysis	1	Oct 7

GIT@JLAB

- https://code.jlab.org
- Everyone with a Jlab account has access
- Jlab is grandfathered into their current GitHub plan, isn’t going away (nor upgraded) for the foreseeable future
- Easy to switch regarding version control (either direction)
- Will live in parallel to the lab’s GitHub organization, but with different restrictions, e.g. number of private repositories and concurrent runners, ***and more accessible/integrated features we can really benefit from***



H HallB

Name	Description		
A ALERT		0 · 2 · 1	
C clas12 Owner		0 · 7 · 1	
c12-swwg	CLAS12 Software Working Group documents	★ 0	5 days ago
ccdbrcdb-java	Unified CCDB/RCDB java build	★ 0	1 week ago
clara-java	CLARA for java	★ 0	1 week ago
clas12-containers	Containerization for CLAS12 software	★ 0	17 hours ago
clas12-env	CLAS12 environment modules	★ 0	4 days ago
clas12-utilities	Various utilities	★ 0	4 days ago

See Chris’s talk

Real Run Number Simulations

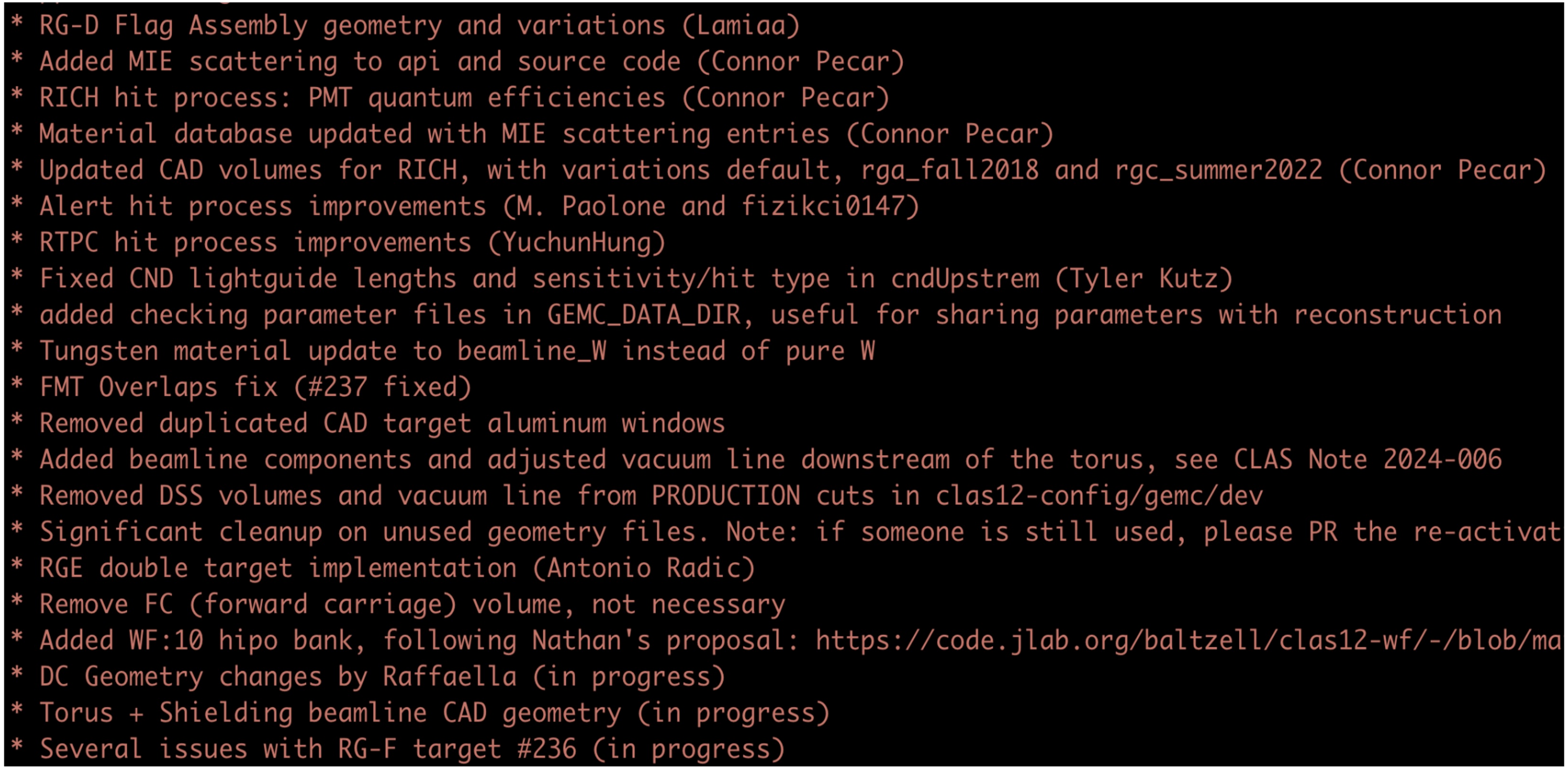
- Due at least in part to the inertia of RUN=11, no automation or caching support for time-dependent geometry parameters, and the flexibility of CCDB …
- CCDB variations were created to manage geometries (and all run-dependent simulation parameters in CCDB) across all run groups, and simulations run with only RUN=11 or similar, which creates a few issues
    - bookkeeping headache, increasingly unmanageable over time
        - Currently 37 run group variations, probably not all used
    - lost the easy and natural link to real data conditions in RCDB and CCDB
    - *most importantly, the natural quantum of simulating experimental conditions becomes the larger run period (variation) instead of the RUN*

Real Run Number Simulations

- What’s required?
  - COATJAVA: get rid of run-number and MC/DATA checks in reconstruction engines, automate run-number-change detection and force each service to initialize its geometry in a new “detectorChanged” method, see https://github.com/JeffersonLab/coatjava/pull/346
  - GEMC: manage the geometries and materials (currently in TXT files) by run number, e.g., in an SQLite database
  - CCDB: extend the generic RUN=11 constants to all runs, copy run-period specific parameters from their variation to their appropriate run ranges in the default variation
  - Test, test, test (must be 100% identical results)

- All the fundamental pieces are in place, expect to finish by end of 2024
  - Then can start to leverage some of the benefits, e.g., sampling across runs

GEMC 5.11 Coming Soon



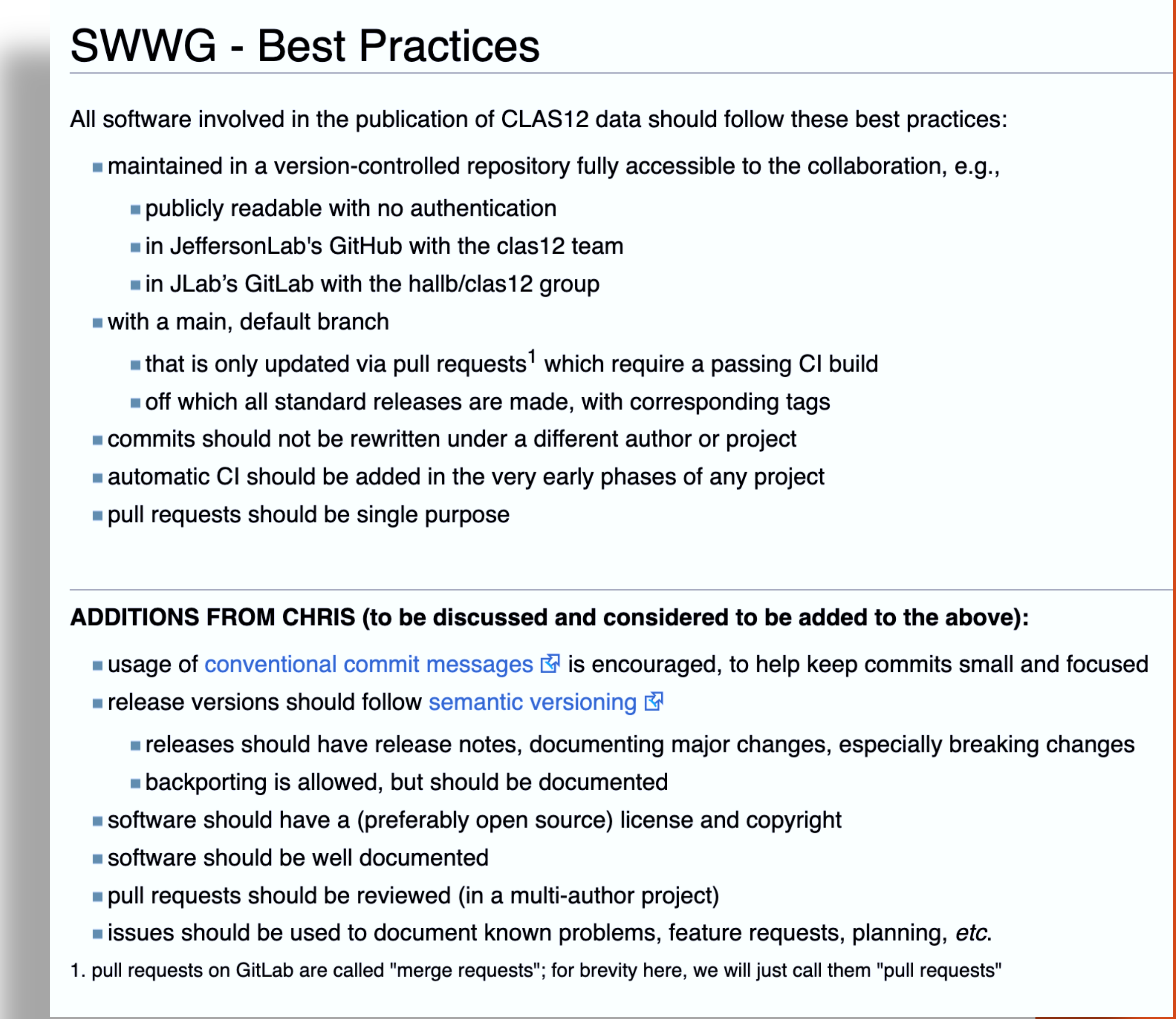
```
* RG-D Flag Assembly geometry and variations (Lamiaa)
* Added MIE scattering to api and source code (Connor Pecar)
* RICH hit process: PMT quantum efficiencies (Connor Pecar)
* Material database updated with MIE scattering entries (Connor Pecar)
* Updated CAD volumes for RICH, with variations default, rga_fall2018 and rgc_summer2022 (Connor Pecar)
* Alert hit process improvements (M. Paolone and fizikci0147)
* RTPC hit process improvements (YuchunHung)
* Fixed CND lightguide lengths and sensitivity/hit type in cndUpstrem (Tyler Kutz)
* added checking parameter files in GEMC_DATA_DIR, useful for sharing parameters with reconstruction
* Tungsten material update to beamline_W instead of pure W
* FMT Overlaps fix (#237 fixed)
* Removed duplicated CAD target aluminum windows
* Added beamline components and adjusted vacuum line downstream of the torus, see CLAS Note 2024-006
* Removed DSS volumes and vacuum line from PRODUCTION cuts in clas12-config/gemc/dev
* Significant cleanup on unused geometry files. Note: if someone is still used, please PR the re-activat
* RGE double target implementation (Antonio Radic)
* Remove FC (forward carriage) volume, not necessary
* Added WF:10 hipo bank, following Nathan's proposal: https://code.jlab.org/baltzell/clas12-wf/-/blob/ma
* DC Geometry changes by Raffaella (in progress)
* Torus + Shielding beamline CAD geometry (in progress)
* Several issues with RG-F target #236 (in progress)
```

More Ongoing Projects

- DC V2
  - Over the past couple years, a few significant inconsistencies around geometry, calibration, and simulation were found.
  - Fixing them reasonably involves breaking compatibility with existing calibration parameters in CCDB, (presumably) requiring a DC recalibration. Hence the “V2”; anyone got a better name?
  - RG-L is the next run group to see beam and will be starting with V2.
  - *See Tongtong’s talk*
- GitLab@Jlab
  - We have a lot to benefit from better CI/CD, container registry, automation, validation, accessibility, preservation …
  - Still new at Jlab, some technical limitations being addressed.
  - *See Chris’s talk*

Best Practices

- Formalizing some best practices by the Software Working Group, was a good suggestion by a couple CLAS collaborators
- It's currently a living document on our main wiki
  - https://clasweb.jlab.org/wiki/index.php/SWWG_-_Best_Practices
- All very common, standard, modern practices
- Feel free to add stuff
- In January we'll consolidate, cleanup, edit, etc.



SWWG - Best Practices

All software involved in the publication of CLAS12 data should follow these best practices:

- maintained in a version-controlled repository fully accessible to the collaboration, e.g.,
  - publicly readable with no authentication
  - in JeffersonLab's GitHub with the clas12 team
  - in JLab's GitLab with the hallb/clas12 group
- with a main, default branch
  - that is only updated via pull requests[1] which require a passing CI build
  - off which all standard releases are made, with corresponding tags
- commits should not be rewritten under a different author or project
- automatic CI should be added in the very early phases of any project
- pull requests should be single purpose

ADDITIONS FROM CHRIS (to be discussed and considered to be added to the above):

- usage of conventional commit messages is encouraged, to help keep commits small and focused
- release versions should follow semantic versioning
  - releases should have release notes, documenting major changes, especially breaking changes
  - backporting is allowed, but should be documented
- software should have a (preferably open source) license and copyright
- software should be well documented
- pull requests should be reviewed (in a multi-author project)
- issues should be used to document known problems, feature requests, planning, *etc.*

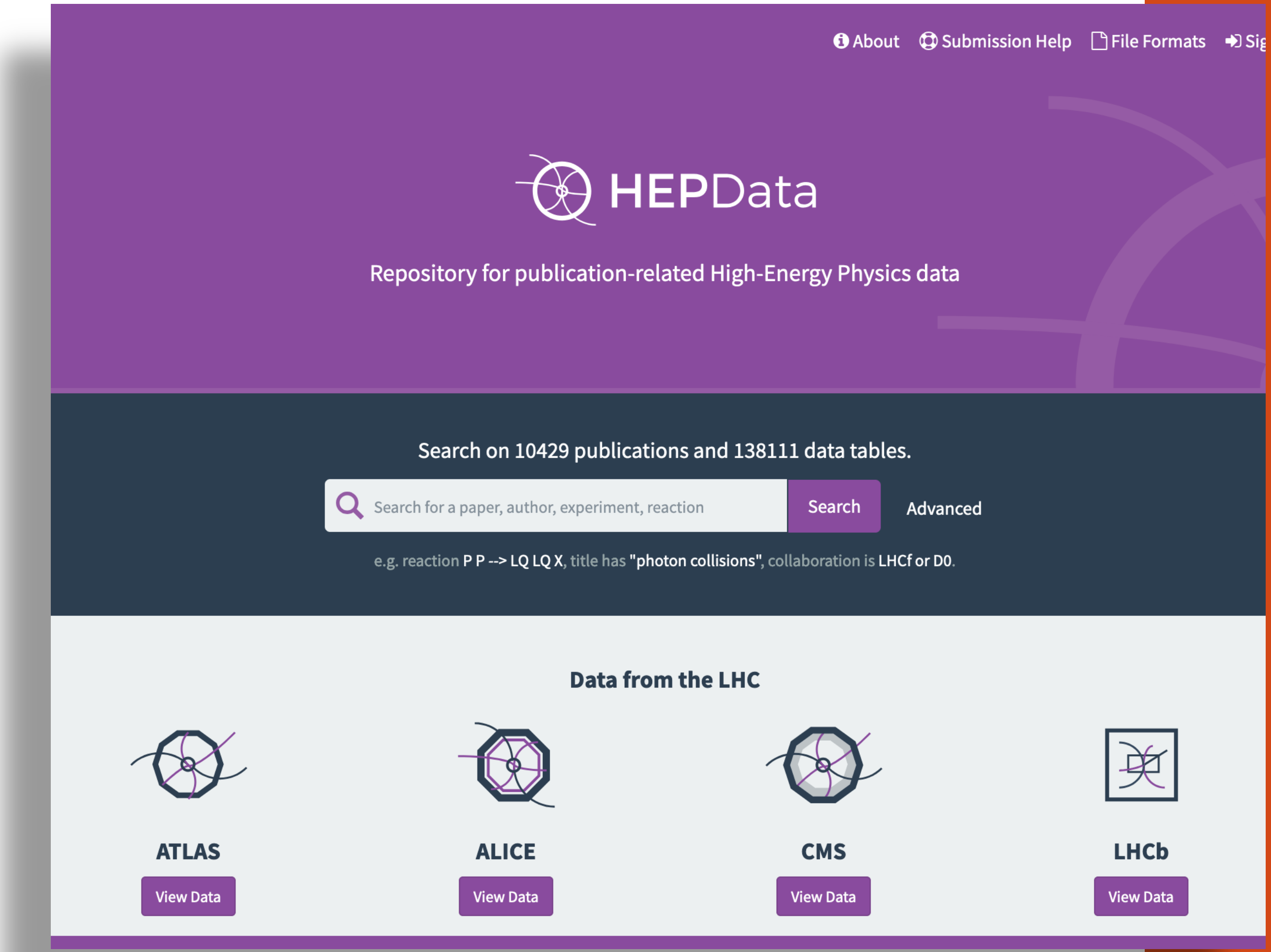
1. pull requests on GitLab are called "merge requests"; for brevity here, we will just call them "pull requests"

CLAS6 Software

- Existing communications channels
  - The original clas_offline@jlab.org listserv mailing list is pretty dormant and new collaborators using clas6 software are often not subscribed
  - A dedicated, new category at the clas12 discourse forum now exists
- The librarian
  - The CLAS collaboration traditionally had a "librarian" role
  - For the past decade, that role was only CLAS6 software builds for local use at JLab
  - With the latest round of system upgrades at JLab, it's now copied and rebuilt and provided in the same place but in a container
    - https://mailman.jlab.org/pipermail/clas_offline/2024-November/001017.html
- Discussion
  - The CLAS6 librarian position is ostensibly vacant, can it be abandoned?
  - CLAS6 software is in SVN, when was the last time anyone used it?
  - CLAS6 software still needs a communication forum. Mailing list, discourse forum, and/or, better, move version control to https://code.jlab.org (svn->git is easy?)

HEPData

- A couple years ago there was an initiative to put all published CLAS12 data in HEPData.
  - An institution account was established, but seems that's about it
- The existing review process is the place to make it happen going forward, if the collaboration wants it to.
- Currently assessing what's involved to register data properly, testing with a user account (Chris)



Repository for publication-related High-Energy Physics data

Search on 10429 publications and 138111 data tables.

e.g. reaction P P --> LQ LQ X, title has "photon collisions", collaboration is LHCf or D0.

Data from the LHC

Questions?