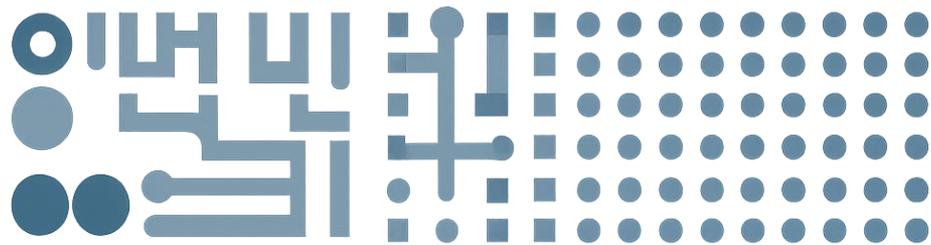


BOF Days

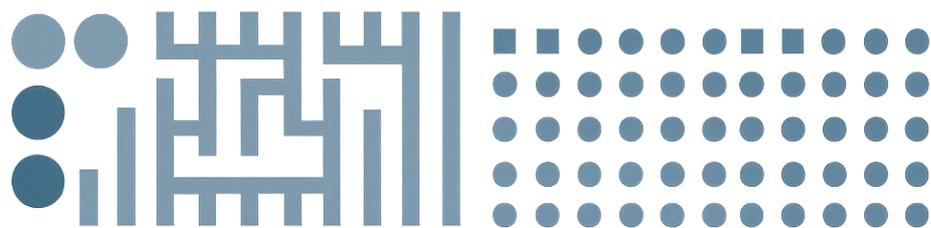
February 10 - 12, 2026

<https://cass.community>

**The
Consortium
for the
Advancement
of Scientific
Software
(CASS)**



CASS



BOF DAYS

<https://cass.community/news/2026-02-10-cass-bof-days.html>

CASS: Stewardship and Advancement of the Scientific Software Ecosystem

- **Inward-facing activities:** Strengthening software products
 - Improve development practices, sustainability, quality, and trustworthiness
 - Enhance user experience and integration within the broader ecosystem
- **Outward-facing activities:** Community engagement and discovery
 - Curate and evolve the software portfolio
 - Help teams connect with and grow their user communities
 - Enable the broader community to discover and adopt useful software

CASS Members

[CORSA](#)
Partnering with foundations to provide sustainable pathways for scientific software

[FASTMATH](#)
Stewardship, advancement, and integration for math and ML/AI packages

[PESO](#)
Stewarding, evolving and integrating a cohesive ecosystem for DOE software

[RAPIDS](#)
Stewardship, advancement, and integration for data, visualization and ML/AI packages

[S4PST](#)
Stewardship, advancement and engagement for programming systems

[STEP](#)
Stewardship, advancement of software tools for understanding performance and behavior

Sponsored by the Department of Energy, Office of Advanced Scientific Computing Research

Engage with CASS!

- Learn about CASS:
 - <https://cass.community/about/>
- Join the CASS Announcement list (low-volume):
 - <http://eepurl.com/iRiSnY>
- Find out more about our **software products**
 - Catalog: <https://cass.community/software/>
 - Collected as part of the [Extreme-Scale Scientific Software Stack](#) (E4S)
- Participate in **CASS Working Groups**
 - Impact Framework, Integration, Metrics, Software Ecosystem, User-Developer Experience, Workforce
 - <https://cass.community/working-groups/>

PESO CI

CASS BoF Feb 2026

The Panel

- **(Host) Vicente Bolea @ Kitware**
- **Prof. Sameer Shende @ University of Oregon**
- **Ryan Krattiger @ Kitware**
- **Daniel Arndt @ ORNL**

Agenda

- ◆ **0'-10' : Introductions**
- ◆ **10'-30' : UO Frank Cluster and Spack CI**
- ◆ **30'-50' : CI in HPC systems (Facility CI)**
- ◆ **50'-70' : The future of CI**
- ◆ **70'-90' : Further discussion**

We will try our best for the last 5 mins of each sections to be open for audience interaction.

UO Frank Cluster and Spack CI

- **Short presentation by Sameer Shende**
- **Conversation**
 - What is the current Frank cluster usage, is it sustainable to keep accepting new projects pipelines? What is the current strategy?
 - We are in need of a standardized way to sync our gitlab/github repos to Spack gitlab? What is the status of Hubcast? Are there any other tools?
 - Would Frank cluster keep acquiring new hardware, if so what can we expect next?
 - What are other resources that we can use in the Frank Cluster apart of their gitlab-runners?
 - Is the Frank cluster willing to host non-Spack gitlab runners? This is, is it willing to host github-runners, or other gitlab-runners that connect to other Gitlab instances such as LLML, Kitware, ORNL...?
 - What other related services (aside of runners) can we host in the Frank cluster?

◆ CI in HPC systems (Facility CI)

- **Short presentation by Vicente Bolea.**
- **Questions:**
 - Would a reusable components to provide these ad-hoc solutions be useful for other team?
 - Do you envision difficulties when extending our work to Perlmutter and Aurora?
 - For most of the PESO projects we have mostly avoided per-project Facility CI since it is difficult to maintain a pipeline per project, What are your thoughts about this?
 - What are your thought for further automate our pipelines?

The future of CI

- **Short presentation by Ryan Krattiger**
- **Conversation:**
 - How could projects support their CI when they face funding reductions/discontinuity?
 - What innovations do you look forward in CI?
 - In a dystopic future where LLMs produces and consume most of the OSS software, is there a place for CI?

Conversation with the audience

- What are your experiences with PESO CI?
- Please share your opinion and experiences
- Thank you!

DAV integration Facility CI

Vicente Bolea @ kitware, inc

Our Goal for CI

- 1. To be the main source from which E4S feeds for CASS/PESO software in Frontier, Aurora, and Perlmutter**
 - a. Produce a Spack binary cache and spack specs for E4S to consume during its release.
- 2. Improve compatibility of CASS/PESO software in target systems**
- 3. Avoid manual processes**

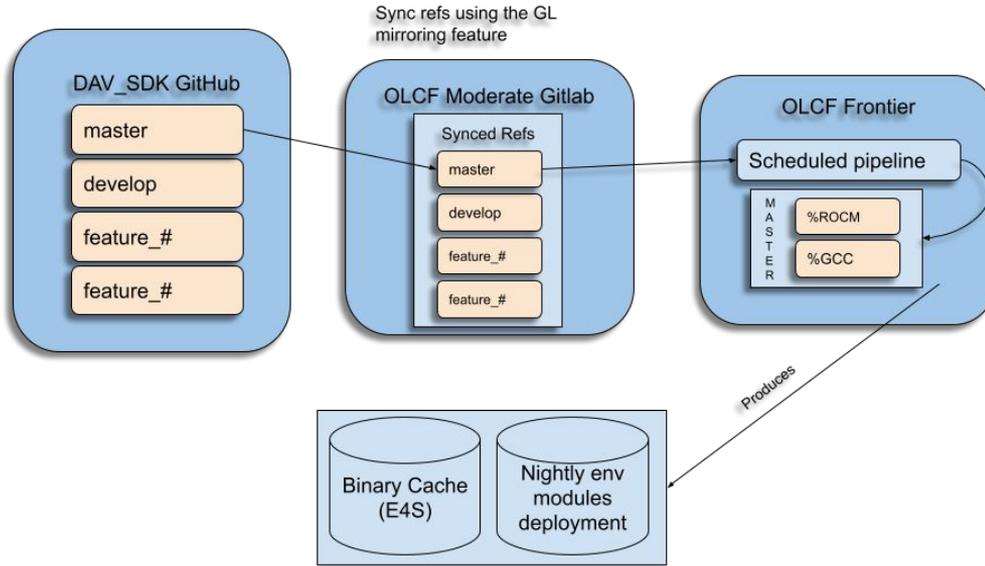
Current status

- **Achieved milestones**
 - Frontier CI
- **Current efforts**
 - Environmental module deployment
 - CDash Dashboard
- **Future works**
 - Perlmutter CI
 - Aurora CI

CI in OLCF infrastructure



The DAV SDK Nightly/Weekly deployment



Some checks were not successful
3 successful and 1 failing checks

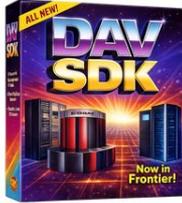
✓	Lint / Shellcheck (pull_request)	Successful in 16s	Details
✓	Security / check_user_allowlist (pull_request_target) ...		Details
✓	Lint / YAML Lint (pull_request)	Successful in 10s	Details
✗	OLCF Frontier — Pipeline failed		Details

Report Status from OLCF Gitlabs back to Github

Nightly deployment

Install on Frontier

[back](#)



****Note:**** These are nightly deployments and are experimental. Pre-built DAV SDK packages are deployed as environment modules on Frontier. To use them, add the module path and load the desired modules.

Setup

Add the DAV SDK module path to your environment:

```
$ module use /lustre/orion/world-shared/ums032/frontier-deployed-env/modules/Core
```

You can add this line to your `~/.bashrc` or job scripts to make it persistent.

Available Modules

List available DAV SDK modules:

```
$ module avail
```

The following modules are available:

CI in OLCF infrastructure

- **Challenge 1: Restrict access to OLCF resources.**
 - Solution: Restricting who can run jobs in Frontier.
 - Needs to be part of the team of the davsdk org and whitelisted
 - Make sure all the allowed users are part of the OCLF project.

Require approval for first-time contributors who are new to GitHub
 Only first-time contributors who recently created a GitHub account will require approval to run workflows.

Require approval for first-time contributors
 Only first-time contributors will require approval to run workflows.

Require approval for all outside collaborators

Save

```

on:
  push:
    branches:
      - master
      - develop
  pull_request_target:

env:
  ALLOWED_USERNAMES: |
    kwryankrattiger
    qtpowell
    vicentebolea

jobs:
  check_user:
  
```

CI in OLCF infrastructure

- **Challenge 2: Sync Pull requests** <https://gitlab.spack.io/dav-sdk/gh-gl-sync>
 - Solution: Delegates into gh-gl-sync reusable component

Explore / CI/CD Catalog / gh-gl-sync

G

dav-sdk/gh-gl-sync

gh-gl-sync v1.0.2

Gitlab to github sync component

Components
Readme

component 🔗 0

```
include:
- component: $CI_SERVER_FQDN/dav-sdk/gh-gl-sync/component@v1.0.2
```

Inputs 🔗

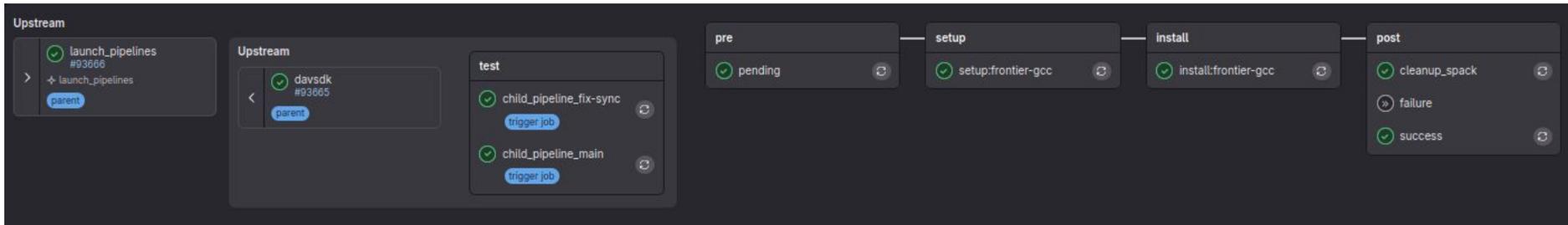
Name	Mandatory	Type

github-sync 🔗 3

```
include:
- component: $CI_SERVER_FQDN/dav-sdk/gh-gl-sync/github-sync@v1.0.2
```

CI in OLCF infrastructure

- Challenge 3: Only scheduled and dynamic builds in Frontier.**
 - Solution: Make use of dynamic pipelines so that each schedule build will trigger other “Child” builds for each vetted ref.





The future of CI

Ryan Krattiger, Kitware Inc.



PESO CI BoF
11 February 2026

Overview

CI is a project process not a project feature

- ◆ **The Maintenance problem**
- ◆ **Expansion of Capabilities**
 - Testing and more testing
 - Continuous Benchmarking

The Maintenance Problem(s)

- ◆ **External/Facility CI**
 - Who has access to debug build/test failures
 - System software updates under CI configuration
 - Synchronization scripts
- ◆ **Pipelines**
 - Who is responsible when the world goes **!!RED!!**



The Maintenance Problem(s): External/Facility CI

- **Who has access to debug build/test failures**
 - External CI needs collaborating stakeholders with access to machines used by CI.
- **System software updates under CI configuration**
 - Establish routine contact with CI infrastructure providers to stay ahead of upcoming changes
- **Synchronization scripts**
 - Utilizing HubCast/GH-GL Bridge to offload maintenance burden



The Maintenance Problem(s): Pipelines

- ◆ Who is responsible when the world goes **!!RED!!**
 - This needs an answer for each project
 - Run pipelines which create the largest impact / cost



Testing and More Testing!

◆ Facility/Deployment tests

- Linking with correct dependencies
- Ensure correct features are enabled

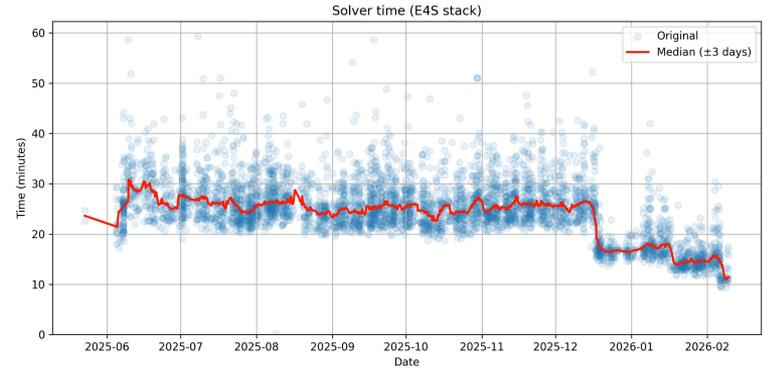
◆ Common test suits

- Benchmarking and tool comparison
- Shared maintenance



Continuous Benchmarking

- ◆ **Detect degradations faster**
 - New features
 - Bug fixes
- ◆ **Track performance improvements**
 - Micro benchmarks can't tell the whole story
- ◆ **Filter CI system noise**
 - CI system usage can affect results



Conversation: The future of CI

- **How could projects support their CI when they face funding reductions/discontinuity?**
- **What innovations do you look forward in CI?**
- **In a dystopic future where LLMs produces and consume most of the OSS software, is there a place for CI?**

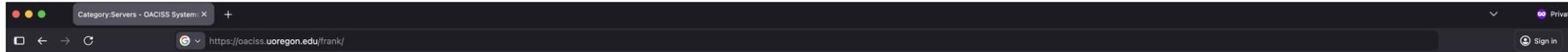


Frank CI @ University of Oregon
<https://oaciss.uoregon.edu/frank>

Prof. Sameer Shende and Luke Peyralans
University of Oregon



GPU Clusters on Frank



<https://oaciss.uoregon.edu/frank>

Some relevant pages:

- [Introduction](#) and about <- click me
- The [NetworkInfrastructure](#) page describes the host naming (dns) conventions, as well as documenting the physical setup and connections within the OACISS racks in the machine room. All OACISS systems automatically search .nic.uoregon.edu for DNS, so only the short hostname is needed for ssh internally.
- The [Service:storage](#) describes available storage for users of OACISS systems. OACISS currently has a total of just under 350TB of online storage available.
- The new [HowtoMPI](#) page describes various tested-working MPI setups and the steps

Click on the server links to access more information about individual machines. Note that only the three machines designated as login gateways (orthus, sphinx, cerberus) are accessible by machines outside of nic.uoregon.edu.

IMPORTANT: Cerberus is in the process of being decommissioned and replaced with a new and much more capable system, Sphinx. Sphinx is the new secondary login node.

Nodes in Computing Center datacenter

Name	Description	OS	Model	Processors	Local Network	Physical location
Compute: Orthus	Primary login gateway	RHEL-8.8	VM	6 x Cascade Lake	10GbE	
Compute: Odyssey	AMD quad MI300A system	RHEL 8.10	SM AS-4145-GT	4 x MI300a, 24cpu	10GbE	R81.U27
Compute: Pinwheel	2xMI210 Debian 12 system	Debian 12	SM AS-2024US-TRT	2 x Epyc Milan 7413	10GbE	R81.U32
Compute: Headroom	Intel Data Center Max 1100 (Ponte Vecchio)	Ubuntu 22	Supermicro	2 x Xeon 4410T @ 3GHz	10GbE	R82.U8
Compute: Hopper1/2	NVIDIA Grace-Hopper GH200	RHEL1 9.3	Quanta S74G-2U	Grace (72c)+GH200 Hopper GPU @ 3.4GHz	10GbE + Connect X-7 200 Gbps (pt2pt w/ hopper1/2.eth)	R81.U33-36
Compute: Grace1/2	NVIDIA Grace-Grace	Ubuntu 22.04 LTS	Supermicro ARS-221GL-NR01 (x2)	Grace-Grace (144c) Superchip @ 3.4GHz	10GbE + EDR 100 Gbps (pt2pt w/grace1/2)	R81.U4-8
Compute: Godzilla	Broadwell + RTX6000 Blackwell Server	RHEL 8.10	Broadwell GPU server	2 x 14c Xeon E5-2680v4 @ 2.3GHz	40GbE + EDR	R85.U6
Compute: Roberta	HPE CPE Epyc Genoa	RHEL 8.10	DL385 gen 11	2 x Epyc 9654 Genoa 96c @ 2.4GHz	10GbE + NDR 400 Gbps (pt2pt w/gary)	R82.U6
Compute: Gary	HPE CPE Epyc Genoa	RHEL 8.10	DL345 gen 11	Epyc 9124 Genoa 16c @ 3GHz	10GbE + NDR 400 Gbps (pt2pt w/roberta))	R82.U4
Compute: Picard	Atipa Sapphire Rapids + 2 A2000	RHEL 8.10	R283-S91	2 x 32c Xeon 6430 @ 3.4GHz	100GbE	R82.U6
Compute: Athena	4xA100 nlink box	RHEL 8.10	Gigabyte RS292	2 x Epyc Milan 7763 5	10GbE	R85.U39
Compute: Illyad	AMD + 1 H100 (80GB) + 2 MI210	RHEL 8.10	Preproduction SuperMicro	2 x 64c Epyc Milan @ 2.8GHz	100GbE + 400G IB	R85.U22
Compute: Gilgamesh	AMD + 1 H100 (80GB) + 2 MI210	RHEL 8.10	Preproduction SuperMicro	2 x 24c Epyc Milan 7413 @ 2.6GHz	100GbE + 400G IB	R85.U26
Compute: Jupiter	Quad Cooper lake + Intel DG1	Ubuntu 22.04.4	Supermicro Sys-240	4 x 24c Xeon Gold 6438 @ 2.3GHz	100GbE + EDR	R86.U10
Compute: Saturn	Quad Ice lake + A100 (80GB) + 2 A100 (40GB)	Ubuntu 22.04.4	Gigabyte RS292-4S1	4 x 26c Xeon Platinum 8367HC @ 3.2GHz	100GbE + EDR	R86.U10



Frank System – Hardware Diversity

macOS/Darwin arm64

Linux arm64 w/ NVIDIA GPU

**Linux x86_64 w/ NVIDIA,
AMD, and Intel GPU**

**Linux power10, power9 w/
NVIDIA GPU**

NVIDIA:

**RTX 6000 Blackwell Server,
RTX5080, GB10, H100,
GH200, A100, V100, P100**

AMD:

MI300A, MI210, MI100

INTEL:

**B580, Data Center Max 1100,
A770, DG1**



Frank Usage

<https://stats.e4s.io>

GPU Type	# Jobs
AMD MI100	163
AMD MI210	25
AMD MI300A	81
AMD MI50	16
INTEL A770	136
INTEL DATA CENTER MAX 1100	236
INTEL DG1	28
NVIDIA A100	74
NVIDIA A2000	14
NVIDIA GH200	71
NVIDIA H100	21
NVIDIA RTX5080	71



Trilinos Nightly Builds – An Example Use Case

<https://gitlab.e4s.io/uo-public/trilinos>
Pipeline from February 10

In **master**

Scheduled latest branch 13 jobs 23 minutes 37 seconds, queued for 2 seconds

Pipeline Jobs 13 Tests 0

Group jobs by Stage Job dependencies

NVIDIA

AMD

INTEL

Prep

✓ Repo-Sync

1-NVIDIA GPU

- ✓ NVIDIA-A100
- ✓ NVIDIA-A2000
- ✓ NVIDIA-GH200
- ✓ NVIDIA-H100
- ✓ NVIDIA-RTX5080

2-AMD GPU

- ✓ AMD-MI50
- ✓ AMD-MI100
- ✓ AMD-MI210
- ✓ AMD-MI300a

3-INTEL GPU

- ✓ INTEL-A770
- ✓ INTEL-DATA-CENTER-MAX-1100
- ✓ INTEL-DG1

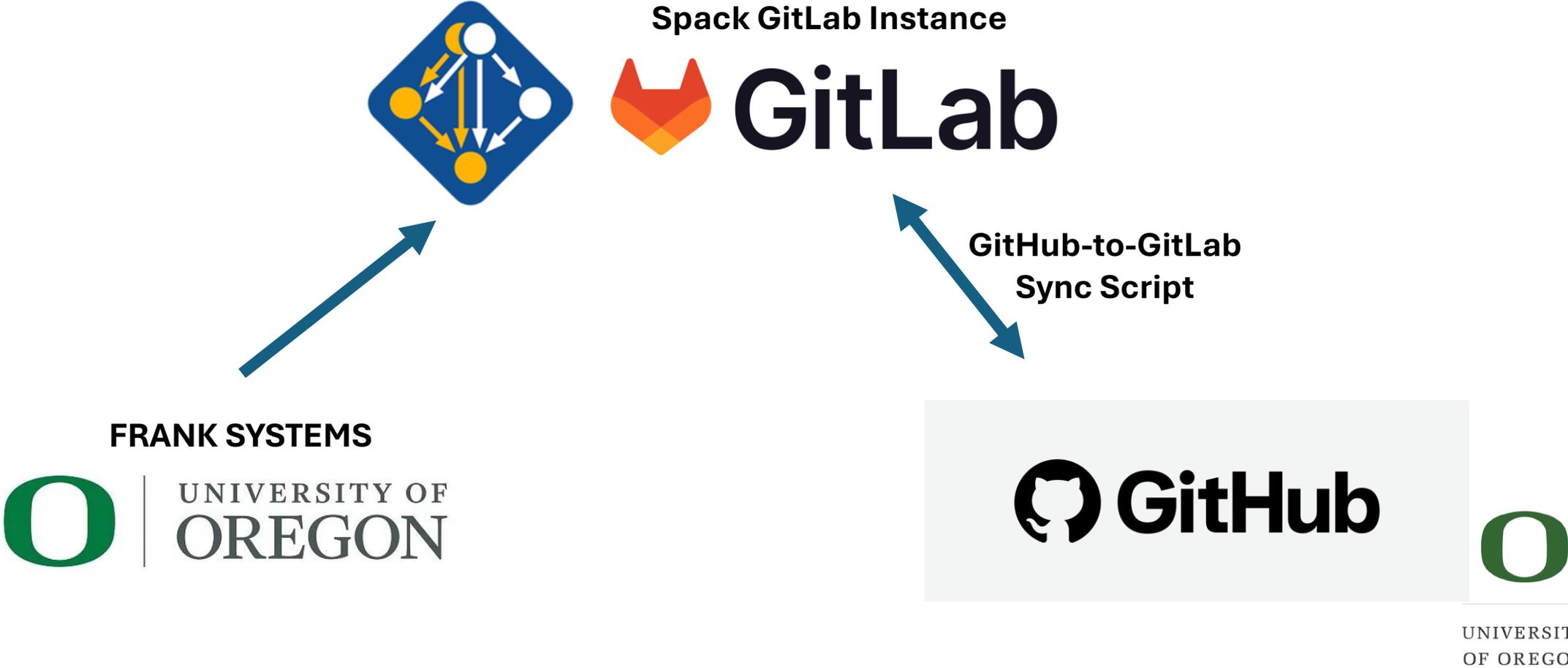


Accessing Frank for CI Jobs – GitLab Integration

- GitLab Runners on individual Frank nodes
- GitLab Runners registered with gitlab.spack.io, gitlab.e4s.io
- Docker Executor for reproducibility and isolation
- GPUs exposed via GitLab `config.toml` settings
- Runners are tagged for appropriate job routing
 - Example tags: `hpsf-gpu`, `nvidia-rtx5080`, `nvidia-h100`, `amd-mi300`, etc...



Accessing Frank for CI Jobs – GitLab Integration



NVIDIA GPU GitLab Runner Snippet

🔄 Tags = hpsf-gpu X Tags = nvidiagpu X 🔍 Created date ↓

Online 9 ● Offline 0 ○ Stale 0 🔄 Upgrade available 0 ↑ Upgrade recommended 9 ⬆

Status	Runner configuration	Owner
<input type="checkbox"/> Online Active	#183835 (jDdbp48C) Instance ↑ Version 18.2.1 · heimdall-nvidia-rtx5080-0 👤 1,000+ ⌚ Last contact: 54 seconds ago 📄 10.0.92.18 📅 Created by luke 11 months ago heimdall uo-gpu x86_64-nvidiagpu linux-6.8 hpsf-gpu nvidia-rtx5080 cuda120 nvidia-575.57.08	Administrator ✎ 🗑
<input type="checkbox"/> Online Idle	#183715 (JgPUjyKm) Instance ↑ Version 18.6.1 · illyad-nvidia-h100-0 👤 221 ⌚ Last contact: 23 minutes ago 📄 10.0.61.181 📅 Created by luke 11 months ago illyad uo-gpu cuda90 x86_64-nvidiagpu nvidia-h100 linux-4.18 hpsf-gpu nvidia-580.65.06	Administrator ✎ 🗑

NVIDIA RTX5080

NVIDIA H100



AMD GPU GitLab Runner Snippet

Filter: Tags = hpsf-gpu X Tags = amdgpu X Search: Created date ▾

Online 3 ● Offline 0 ○ Stale 0 ⚠ Upgrade available 0 ⬆ Upgrade recommended 3 ⬆

Status	Runner configuration	Owner
<input type="checkbox"/> Online Active	#180570 (TVN88wPR) 88 Instance Version 18.6.3 · odyssey-amd-mi300-0 1,000+ Last contact: 56 seconds ago 10.0.78.230 Created by luke Nov 26, 2024 uo-gpu x86_64-amdgpu amdgpu-6.2 odyssey gfx942 amd-mi300 linux-4.18 hpsf-gpu	Administrator [edit] [pause] [delete]
<input type="checkbox"/> Online Active	#84531 (zT1fd1W2) 88 Instance Version 18.7.1 · instinct-amd-mi100-0 1,000+ Last contact: 22 seconds ago 10.0.94.55 Created by luke Sep 30, 2024 instinct uo-gpu x86_64-amdgpu amd-mi100 gfx908 linux-5.15 hpsf-gpu	Administrator [edit] [pause] [delete]

AMD MI300A

AMD MI100



Online **18** ● Offline 0 ○ Stale 0 🔄 Upgrade available 0 ⬆️ Upgrade recommended 0 ⬆️

**macOS arm64 hardware
18 distinct runners online**

<input type="checkbox"/>	Status	Runner configuration ?	Owner ?	
<input type="checkbox"/>	Online Idle	<p>#186872 (m6z2ljCn) Instance</p> <p>Version 18.8.0 · uo-epsilon-3</p> <p>🌐 7 ⌚ Last contact: 28 seconds ago 📄 10.0.0.82</p> <p>📅 Created by luke 18 hours ago</p> <p>public spack small uo shell aarch64-macos m2 epsilon macos-tahoe apple-clang-17 uo-epsilon-3</p>	Administrator	✎ ⏸ 🗑
<input type="checkbox"/>	Online Idle	<p>#186871 (tuW1rOEK) Instance</p> <p>Version 18.8.0 · uo-epsilon-2</p> <p>🌐 9 ⌚ Last contact: 28 seconds ago 📄 10.0.51.253</p> <p>📅 Created by luke 18 hours ago</p> <p>public spack small uo shell aarch64-macos m2 epsilon uo-epsilon-2 macos-tahoe apple-clang-17</p>	Administrator	✎ ⏸ 🗑

macOS / Darwin Hardware

- Variety of of ARM64 macOS / Darwin nodes for CI
- M4 Pro x2 (Eta 32gb, Rho 48gb)
- M2 Pro x3 (Kappa 32gb, Upsilon 32gb, Theta 32gb)
- M1 Ultra (Mu 128gb)

Examples Projects using Frank Hardware for CI

Viskores



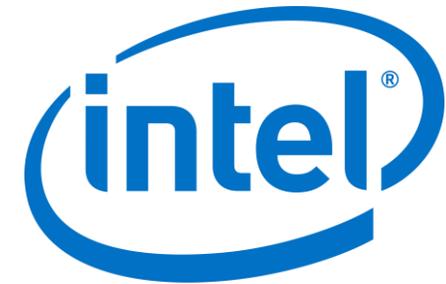
Adios2



Spack



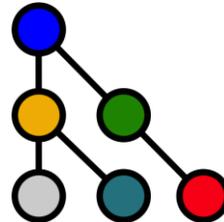
Intel



E4S



HPCToolkit



Kokkos





Viskores CI Usage

- Code
- Issues 20
- Pull requests 10
- Discussions
- Actions
- Projects
- Models
- Security
- Insights

Commits

main

Commits on Feb 3, 2026

Merge pull request #250 from vicentbolea
vicentbolea authored last week · 7 / 9

add security policy file
vicentbolea committed last week · 8 / 9

Commits on Feb 1, 2026

Merge pull request #237 from kmorel
kmorel authored last week · 8 / 9

Commits on Jan 31, 2026

All checks have passed
8 successful and 1 skipped checks

- ✓ Frank CI - Pipeline succeeded [Details](#)
- ✓ OLCF Frontier - Pipeline succeeded [Details](#)
- ✓ cdash - Build and test results available on CDash [Details](#)
- ✓ CI / format (push) Successful in 32s [Details](#)
- ✓ external / generate_statuses (push) Successful in 8s [Details](#)
- ✓ CI / macos (macos-14) (push) Successful in 1m [Details](#)
- ⊘ Doxygen / push (push) Skipped [Details](#)
- ✓ Doxygen / user_guide (push) Successful in 4m [Details](#)

Viskores Frank Pipeline February 1, 2026

- ✓ ubuntu2004_gcc9
- ✓ ubuntu2004_gcc9_cuda118
- ✓ ubuntu2004_kokkos37
- ✓ ubuntu2204_clang11_cuda122
- ✓ ubuntu2204_clang12
- ✓ ubuntu2204_gcc12
- ✓ ubuntu2204_hip_kokkos37
- ✓ **ubuntu2204_hip_kokkos43**
- ✓ ubuntu2204_kokkos37_cuda122
- ✓ ubuntu2404_gcc13
- ✓ ubuntu2404_gcc14
- ✓ ubuntu2404_gcc14_cuda130

Job Output: ubuntu2204_hip_kokkos43 Using Instinct AMD MI100 GPU to run CTests

Log timestamps in UTC. Search visible log output

```
6/Build.xml to https://open.cdash.org/submit.php?project=Viskores&fileName=ubuntu2204_hip_kokkos43___relwithdebinfo%2Bbenchmarks%2Bkokkos%2Bhip%2Bno_rendering%2Bno_testing%2Bccache___20260201-2016-merge-requests___XML___Build.xml&build=relwithdebinfo%2Bbenchmarks%2Bkokkos%2Bhip%2Bno_rendering%2Bno_testing%2Bccache&site=ubuntu2204_hip_kokkos43&stamp=20260201-2016-merge-requests&MD5=190238c078b8dace8f2a9eec38acd290 Size: 1182
1198 20:17:32 Uploaded: /builds/viskores/viskores/build/Testing/20260201-2016/Build.xml
1199 20:17:32 Submission successful
1200 20:17:32 Build submission build_id: 11012004
1201 20:17:32 $ ctest -VV -S .gitlab/ci/ctest_build_notes.cmake
1202 20:17:32 Extra verbosity turned on
1203 20:17:32 Reading Script: /builds/viskores/viskores/.gitlab/ci/ctest_build_notes.cmake
1204 20:17:32 Add coverage exclude regular expressions.
1205 20:17:32 SetCTestConfiguration:SourceDirectory:/builds/viskores/viskores
1206 20:17:32 SetCTestConfiguration:BuildDirectory:/builds/viskores/viskores/build
1207 20:17:32 Run dashboard with to-be-determined model
1208 20:17:32 Source directory: /builds/viskores/viskores
1209 20:17:32 Build directory: /builds/viskores/viskores/build
1210 20:17:32 Reading ctest configuration file: /builds/viskores/viskores/CTestConfiguration.cmake
1211 20:17:32 SetCTestConfigurationFromCMakeVariable:NightlyStartTime:CTEST_NIGHTLY_START_TIME
1212 20:17:32 SetCTestConfiguration:NightlyStartTime:01:00:00 UTC
1213 20:17:32 SetCTestConfigurationFromCMakeVariable:Site:CTEST_SITE
1214 20:17:32 SetCTestConfiguration:Site:ubuntu2204_hip_kokkos43
1215 20:17:32 SetCTestConfigurationFromCMakeVariable:BuildName:CTEST_BUILD_NAME
1216 20:17:32 SetCTestConfiguration:BuildName:relwithdebinfo+benchmarks+kokkos+hip
```

Duration: 2 minutes 13 seconds
Finished: 1 week ago
Queued: 1 minute 59 seconds
Timeout: 2h (from job) ?

Runner: #84531 (zT1fd1W2) instinct-
amd-mi100-0

Source: Push

Tags: gfx908 uo-gpu x86_64-amdgpu
instinct

Commit 08bc14f8 [🔗](#)
Merge pull request #237 from kmorel/heck-kokkos-finalize

Pipeline #1409385 ✓ Passed for main [🔗](#)

build

Related jobs

- ✓ almalinux8
- ✓ almalinux8_sanitizer
- ✓ almalinux8_vtk_types
- ✓ gcc-continuous

HPCToolkit Example Pipeline

February 10, 2026

Tests HPCToolkit / HPCToolkit / Jobs / #20279332

Job output: rocm6.3 amd64:test:hpsf: [mi300]

test		
Failed jobs		
✖	intel amd64:test:hpsf	1
✔	cuda12.0 amd64:test:hpsf	1
✔	cuda12.1 amd64:test:hpsf	1
✔	cuda12.2 amd64:test:hpsf	1
✔	cuda12.4 amd64:test:hpsf	1
✔	cuda12.5 amd64:test:hpsf	1
✔	ppc64le:test:hpsf	1
✔	rocm6.3 amd64:test:hpsf	2
✔	ubuntu22.04 ppc64le:test:hpsf	1
✔	ubuntu24.04 ppc64le:test:hpsf	1

```

3024 [265/278] Compiling C++ object src/hpcstruct/dotgraph.p/intel_GPUBlock.cpp.o
3025 [266/278] Linking target tests/data/meas/testmeas-small
3026 [267/278] Compiling C++ object src/hpcstruct/dotgraph.p/intel_GPUCodeSource.cpp.o
3027 [268/278] Linking target tests/data/meas/testmeas-divergence
3028 [269/278] Linking target tests/data/meas/testmeas-loops
3029 [270/278] Compiling C++ object src/hpcstruct/dotgraph.p/intel_GPUCFG_Intel.cpp.o
3030 [271/278] Compiling C++ object src/hpcstruct/dotgraph.p/Struct.cpp.o
3031 [272/278] Compiling C++ object src/hpcstruct/dotgraph.p/gpu_CudaCFG.cpp.o
3032 [273/278] Linking target tests/data/meas/testmeas-recursion-cuda
3033 [274/278] Linking target tests/data/meas/testmeas-loops-cuda
3034 [275/278] Linking target subprojects/dyninst-13.0.0/parseThat/parseThat
3035 [276/278] Linking target src/hpctracedump/hpctracedump
3036 [277/278] Linking target src/hpcstruct/dotgraph
3037 [278/278] Linking target src/hpcprof/hpcprof-mpi
3038 INFO: autodetecting backend as ninja
3039 INFO: calculating backend command to run: /usr/bin/ninja
> 3040 Saving cache for successful job 00:03
✓ 3049 Uploading artifacts for successful job 00:01
3050 Uploading artifacts...
3051 Uploading artifacts 0 B (0 B/s) tmp/builddir*/meson-logs/testlog.junit.xml: found 1 matching artifact files and directories
3052 Uploading artifacts 74.71 KB (341.7 KB/s)
3053 Uploading artifacts as "junit" to coordinator... 201 Created correlation_id=01KH3XWJ6RW3CAC6R6RP0TN1JA id=20279332 responseStatus=201 Created token=eyJraWQiOiI
✓ 3054 Cleaning up project directory and file based variables 00:01
3055 Job succeeded
    
```

Odyssey AMD MI300A Runner

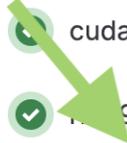
Queued: 1 second
 Timeout: 1h (from project) [?](#)
 Runner: #180570 (TVN88wPR)
 odyssey-amd-mi300-0
 Source: web
 Test summary: 323
 Tags: hpsf-gpu x86_64-amdgpu amd-mi300

Job artifacts [?](#)
 The artifacts will be removed in 2 months [?](#)

Commit 2d62bc89 [📄](#)
 hpcrun: improve error handling for (unmonitored) threads without thread data (hpc/toolkit/hpctoolkit!1433)

Pipeline #1421671 ✖ Failed for development [📄](#)

test ▼



Spack Package Stacks building on Frank

- Frank hardware heavily used for Spack CI
 - Spack core: spack/spack
 - Spack packages: spack/spack-packages
- macOS and Darwin jobs exclusively run on Frank hardware
- CrayPE jobs historically run exclusively on Frank hardware (Cray stacks currently disabled while systems are maintained)
- Other jobs split between AWS and Frank, ~40-50% run on Frank



Search or go to...

spack / spack-packages / Pipelines / #1421762

spack/spack-packages
ml-darwin-aarch64-mps-build
pipeline from February 10, 2026

stage-5

- ✔ nvtx@3.2.1 /646kdre4b7xf7zyep5mgmshtuximgz...
- ✔ nvtx@3.3.0 / ic2onqy7r3iqhjlnud5qnvortxdquqt...
- ✔ py-aiohappyeyeballs@2.6.1 / khwkiraqxmcybpmvgvig6msajkj5yeio...
- ✔ py-astunparse@1.6.3 /577g4sramvluzrnofjonrclzb7kcxgxp...
- ✔ py-astunparse@1.6.3 / xwbpyml7hm24km4knlh2nta5pkwl6h...
- ✔ py-beniget@0.4.2.post1 /37dl7atluzsrhuqqi3lavnokdkey7gwu...
- ✔ py-beniget@0.4.2.post1 / veaa2fgcgijnj3nrwoa5ojsjdiovte3t...
- ✔ py-click@8.2.1 /

stage-6

- ✔ py-build@1.2.2 / qnkf3f5smzkb3sftb6tly2tmgrnxj57y...
- ✔ py-cligj@0.7.2 / k6abqosseup7kkf42cwh7w2dd5fnx2q6...
- ✔ py-grpcio@1.75.0 /3bojxnxtatbsje4hbjaafibqhdg56ri...
- ✔ py-grpcio@1.75.0 / fpr7vo7hgcrou2xcjzmxh6yxuixboho...
- ✔ py-lightning-utilities@0.11.2 / bvkagsn5kteczm42xnqfnmdohsehndb2...
- ✔ py-markdown-it-py@4.0.0 / ayux32wz4ndxaeptsc4mlcx6p3yanghs...
- ✔ py-markdown-it-py@4.0.0 / svebka2a4r6al2zvvg5227wabwlkj3n...
- ✔ py-omegaconf@2.3.0 /

stage-7

- ✔ py-cppy@1.3.1 / for2nrbbmirhdb2pfl5...
- ✔ py-hydra-core@1.3.2 / ggzz6vgvrnon77wb3...
- ✔ py-importlib-resources@6.1.0 / t6mgrprurryu2zgy32j...
- ✔ py-meson-python@0.16.1 / ejsezu4yax7o5ar4c7...
- ✔ py-meson-python@0.16.1 / kexeaambmsngijywal...
- ✔ py-meson-python@0.16.1 / uiak3yaarawmfyfpg...
- ✔ py-pluggy@1.6.0 / /5yhi4nfhl4k54hbdpr...
- ✔ py-pluggy@1.6.0 /

Kappa M2 Ultra Darwin Runner

py-tensorflow-metadata@1.17.2 / dppsqvjbgipjczsozrlhpjsdbp6wace Machine Learning MPS



Passed Started 1 hour ago by Spack Bot

Duration: 5 minutes 20 seconds
Finished: 1 hour ago
Queued: 11 minutes 20 seconds
Timeout: 4h (from job)

Log timestamps in UTC. Search visible log output

```
1 16:06:30 Running with gitlab-runner 18.8.0 (9ffb4aa0)
2 16:06:30 on uo-kappa-2 5iTVzrW3b, system ID: s_000909c387b4
3 16:06:30 Resolving secrets
4 16:06:30 Preparing the "shell" executor 00:00
5 16:06:30 Using Shell (bash) executor...
6 16:06:30 Preparing environment 00:00
7 16:06:30 Running on U0-2007187...
8 16:06:30 Getting source from Git repository 00:06
9 16:06:30 Gitaly correlation ID: 01KH44VR59TKA58JABQRJDB8XF
10 16:06:30 Fetching changes with git depth set to 2...
11 16:06:30 Reinitialized existing Git repository in /Users/gitlab-runner-2/build
s/5iTVzrW3b/0/spack/spack-packages/.git/
12 16:06:31 Checking out c9ef9d1f as detached HEAD (ref is pr2966_ci-enable-darwi
```

Runner: #186866 (5iTVzrW3) uo-kappa-2
Source: Parent Pipeline
Tags: macos-tahoe apple-clang-17 aarch64-macos public

Job artifacts ?
The artifacts will be removed in 2 months ?
Keep Download Browse

Commit c9ef9d1f
Merge 1e3c3a8de91e58a051fa2a4a21f4 d502e5206514 into 2f482e6f00007b000

E4S: Extreme-scale Scientific Software Stack



<https://e4s.io>

About E4S

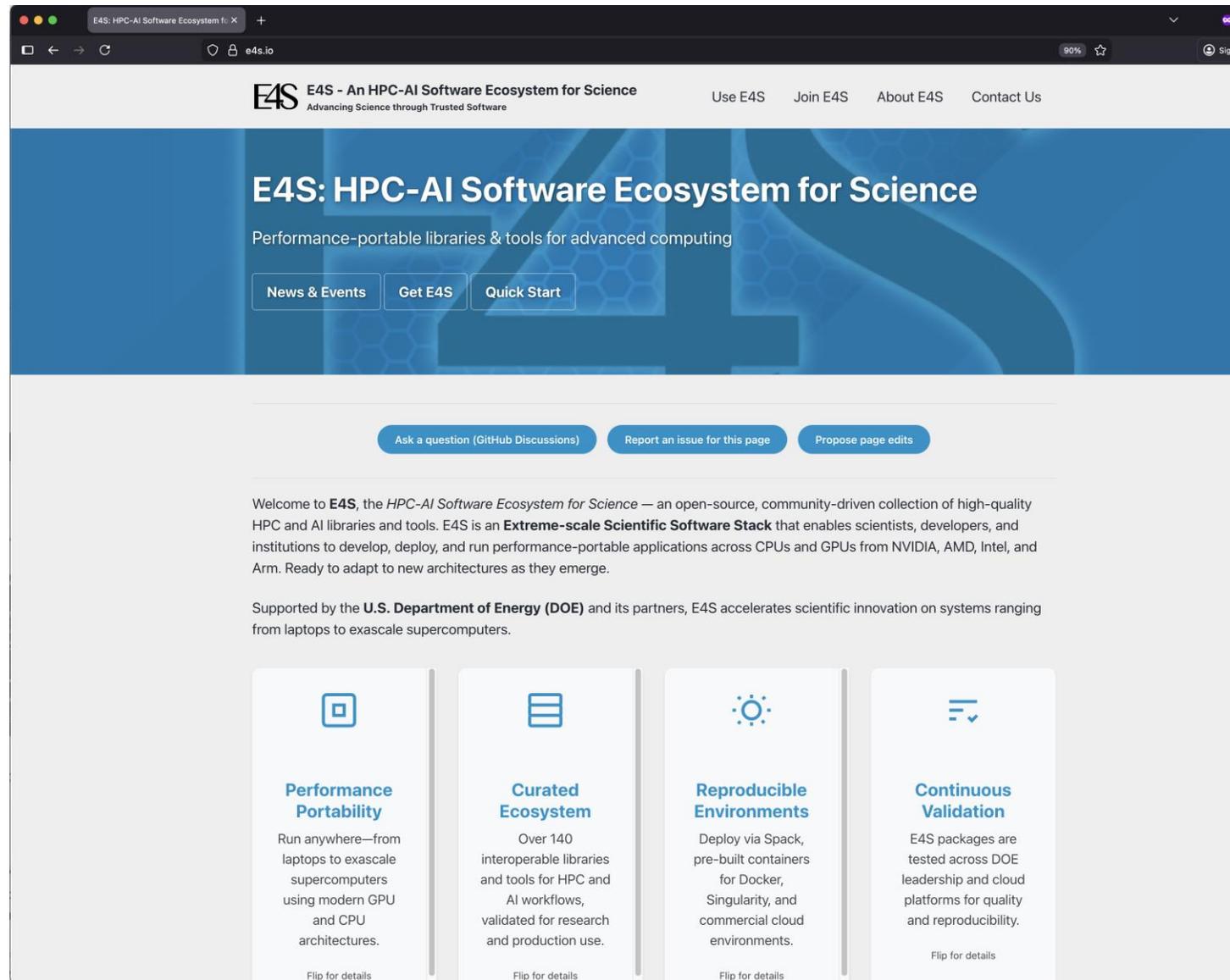
- E4S, an HPSF project, is an **ecosystem for science** and a community effort to provide open-source software packages for developing, deploying and running scientific applications on HPC platforms.
- E4S has built a comprehensive, coherent software stack that enables application developers to productively develop highly parallel applications that effectively target diverse exascale architectures.
- E4S provides a curated, Spack based software distribution of 120+ HPC (OpenFOAM, Gromacs, Nek5000, LAMMPS), EDA (e.g., Xyce), and AI/ML packages (e.g., NVIDIA NeMo™, HuggingFace Hub, TensorFlow, PyTorch, OpenCV, TorchBraid, Scikit-Learn, Pandas, JAX, Horovod, LBANN with support for GPUs).
- Base images and full featured containers (with GPU support) and DOE LLVM containers.
- Commercial support for E4S through ParaTools, Inc. for installation, maintaining an issue tracker, and ECP AD engagement.
- E4S for commercial clouds: Adaptive Computing's ODDC with ParaTools Pro for E4S™ image for **AWS, GCP, Azure, OCI**.
- With E4S Spack binary build caches, E4S supports both bare-metal and containerized deployment for GPU based platforms.
 - X86_64, ppc64le (IBM Power 10), aarch64 (ARM64) with support for GPUs from NVIDIA, AMD, and Intel
 - HPC and AI/ML packages are optimized for GPUs and CPUs.
 - Container images on DockerHub and E4S website of pre-built binaries of ECP ST products.
- e4s-chain-spack.sh to chain two Spack instances allows us to install new packages in home directory and use other tools.
- e4s-cl container launch tool allows binary distribution of applications by swapping MPI in the containerized app w/ system MPI
- e4s-alc is an à la carte tool to customize container images by adding system and Spack packages to an existing image.

CI on Frank@UO Powered by E4S and Spack

- E4S provides a variety of containers that may be used by CI projects.
- Custom Spack based containers with support for GPUs.
- <https://e4s.io>
- <https://spack.io>
- <https://gitlab.e4s.io/uo-public/>



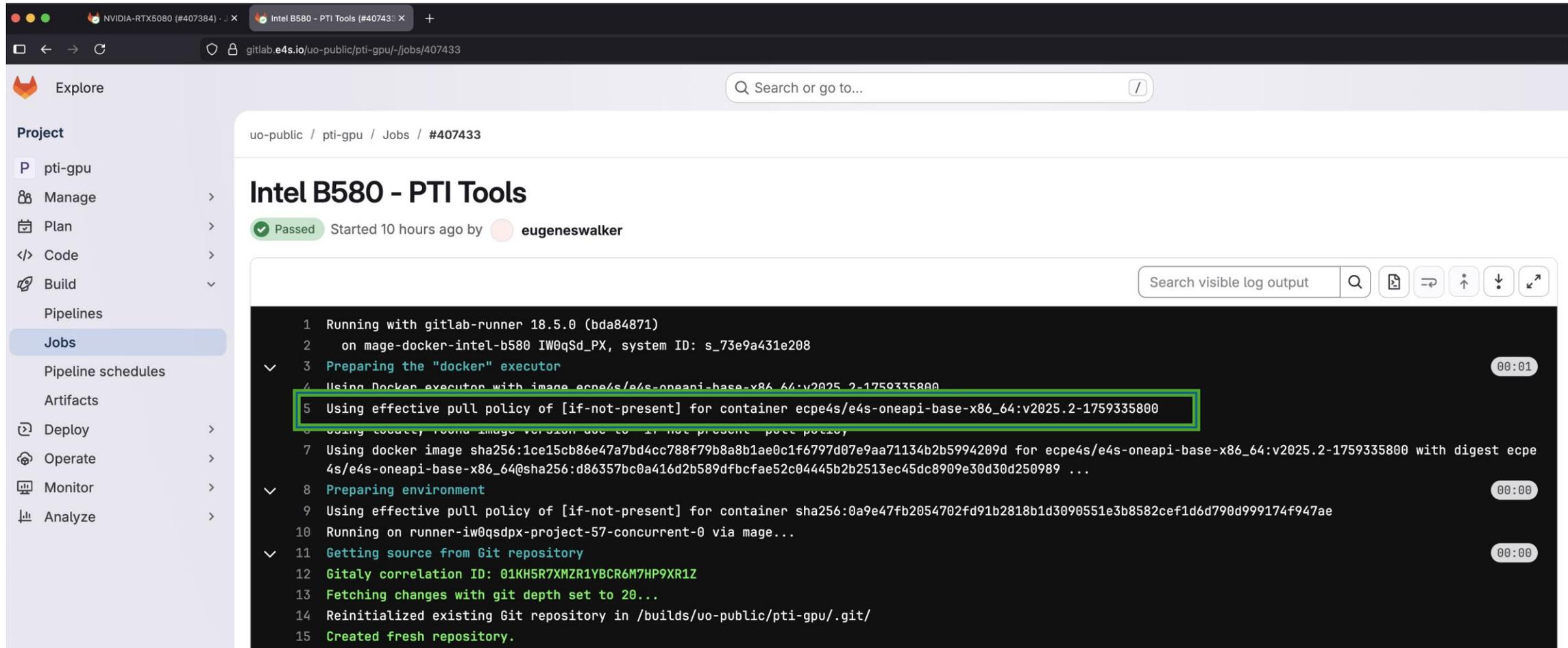
E4S



The screenshot shows the E4S website homepage. At the top, there is a navigation bar with the E4S logo and the text "E4S - An HPC-AI Software Ecosystem for Science" and "Advancing Science through Trusted Software". To the right of the logo are links for "Use E4S", "Join E4S", "About E4S", and "Contact Us". The main header features the title "E4S: HPC-AI Software Ecosystem for Science" and the subtitle "Performance-portable libraries & tools for advanced computing". Below this are three buttons: "News & Events", "Get E4S", and "Quick Start". A secondary navigation bar contains three buttons: "Ask a question (GitHub Discussions)", "Report an issue for this page", and "Propose page edits". The main content area starts with a welcome message: "Welcome to **E4S**, the *HPC-AI Software Ecosystem for Science* — an open-source, community-driven collection of high-quality HPC and AI libraries and tools. E4S is an **Extreme-scale Scientific Software Stack** that enables scientists, developers, and institutions to develop, deploy, and run performance-portable applications across CPUs and GPUs from NVIDIA, AMD, Intel, and Arm. Ready to adapt to new architectures as they emerge." This is followed by a line of support: "Supported by the **U.S. Department of Energy (DOE)** and its partners, E4S accelerates scientific innovation on systems ranging from laptops to exascale supercomputers." Below this are four feature cards, each with an icon and a title: 1. "Performance Portability" with a square icon containing a smaller square, describing running on various architectures. 2. "Curated Ecosystem" with a list icon, mentioning over 140 interoperable libraries. 3. "Reproducible Environments" with a sun icon, describing deployment via Spack and containers. 4. "Continuous Validation" with a checkmark icon, describing testing across DOE platforms. Each card has a "Flip for details" link at the bottom.



Intel PTI-GPU CI at UO using E4S containers



The screenshot shows a GitLab CI job page for 'Intel B580 - PTI Tools'. The job status is 'Passed' and it was started 10 hours ago by 'eugenewalker'. The log output is visible, with the following steps:

- 1 Running with gitlab-runner 18.5.0 (bda84871)
- 2 on mage-docker-intel-b580 IW0qSd_PX, system ID: s_73e9a431e208
- 3 **Preparing the "docker" executor** 00:01
- 4 Using Docker executor with image ecpe4s/e4s-oneapi-base-x86_64:v2025.2-1759335800
- 5 **Using effective pull policy of [if-not-present] for container ecpe4s/e4s-oneapi-base-x86_64:v2025.2-1759335800**
- 6 Using locally found image version due to [if-not-present] pull policy
- 7 Using docker image sha256:1ce15cb86e47a7bd4cc788f79b8a8b1ae0c1f6797d07e9aa71134b2b5994209d for ecpe4s/e4s-oneapi-base-x86_64:v2025.2-1759335800 with digest ecpe4s/e4s-oneapi-base-x86_64@sha256:d86357bc0a416d2b589dfbcfae52c04445b2b2513ec45dc8909e30d30d250989 ...
- 8 **Preparing environment** 00:00
- 9 Using effective pull policy of [if-not-present] for container sha256:0a9e47fb2054702fd91b2818b1d3090551e3b8582cef1d6d790d999174f947ae
- 10 Running on runner-iw0qsdpj-project-57-concurrent-0 via mage...
- 11 **Getting source from Git repository** 00:00
- 12 **Gitaly correlation ID: 01KH5R7XMZR1YBCR6M7HP9XR1Z**
- 13 **Fetching changes with git depth set to 20...**
- 14 Reinitialized existing Git repository in /builds/uo-public/pti-gpu/.git/
- 15 **Created fresh repository.**

<https://gitlab.e4s.io/uo-public/pti-gpu/-/jobs/407433>



Acknowledgment

- *This work was supported by the U.S. Department of Energy, Office of Science, Advanced Computing Research, through the Next-Generation Scientific Software Technologies (NGSST) under contract DE-AC02-AC05-00OR22725 and DOE SBIR DE-SC0022502.*

- <https://science.osti.gov/ascr>
- <https://pesoproject.org>
- <https://ascr-step.org>
- <https://hpsf.io>
- <https://www.energy.gov/technologytransitions/sbirsttr>



U.S. DEPARTMENT OF
ENERGY

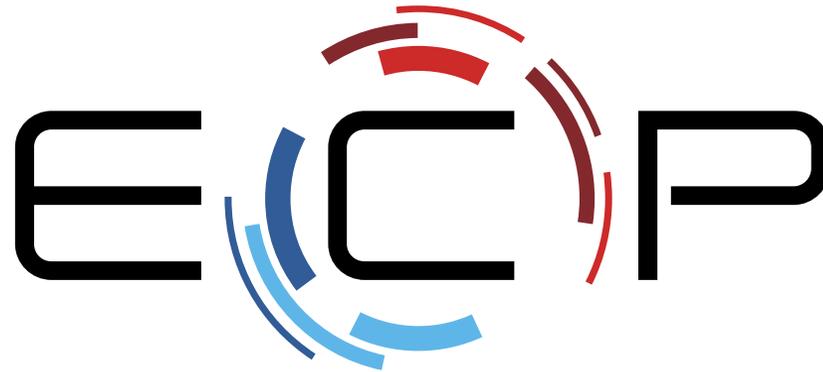
Office of
Science



Thank you

<https://www.exascaleproject.org>

This research was supported by the Exascale Computing Project (17-SC-20-SC), a joint project of the U.S. Department of Energy's Office of Science and National Nuclear Security Administration, responsible for delivering a capable exascale ecosystem, including software, applications, and hardware technology, to support the nation's exascale computing imperative.



EXASCALE COMPUTING PROJECT

Thank you to all collaborators in the ECP and broader computational science communities. The work discussed in this presentation represents creative contributions of many people who are passionately working toward next-generation computational science.



UNIVERSITY
OF OREGON

