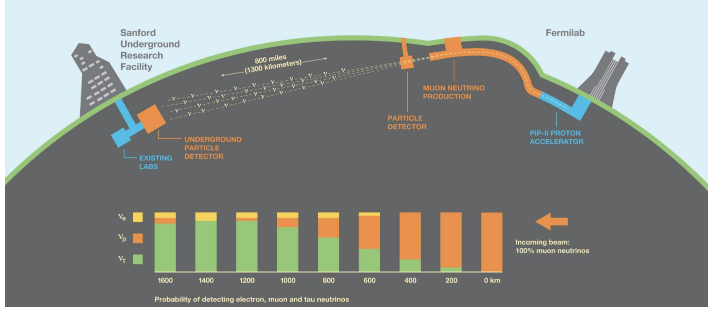
DUNE Computing Tutorials

David DeMuth, Valley City State University
CHEP 2023, Norfolk, Virginia, USA
For the DUNE Collaboration
May 8, 2023

https://bit.ly/chep23-dune-training

DUNE Computing

- The physics realms of the Deep Underground Neutrino Experiment presents a unique challenge for data analysis and data processing software frameworks:
  - Precision neutrino oscillation measurements,
  - Searches for nucleon decay,
  - Sensitivity to nearby Supernova explosions, and more…
- Particle interactions will span timescales of nanoseconds to 100's of seconds, where event energies and associated processing timelines require a **wide arsenal of software tools**.
- Moreover, **data storage is cloud-based and worldwide**, and require failsafe high bandwidth networks with strong security and monitoring protocols.

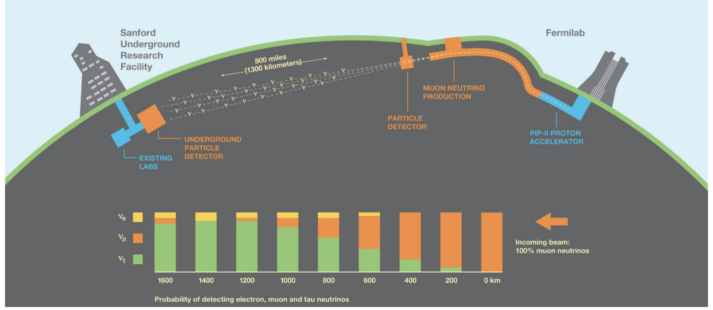

Sanford Underground Research Facility
Fermilab
800 miles (1300 kilometers)
MUON NEUTRINO PRODUCTION
PARTICLE DETECTOR
UNDERGROUND PARTICLE DETECTOR
EXISTING LABS
PIP-II PROTON ACCELERATOR
ν_e
ν_μ
ν_τ
1600
1400
1200
1000
800
600
400
200
0 km
Incoming beam: 100% muon neutrinos
Probability of detecting electron, muon and tau neutrinos

Worldwide and sophisticated, DUNE Computing Tutorials are used regularly to train new colleagues.

DUNE Computing CDR: https://arxiv.org/abs/2210.15665

DUNE Computing

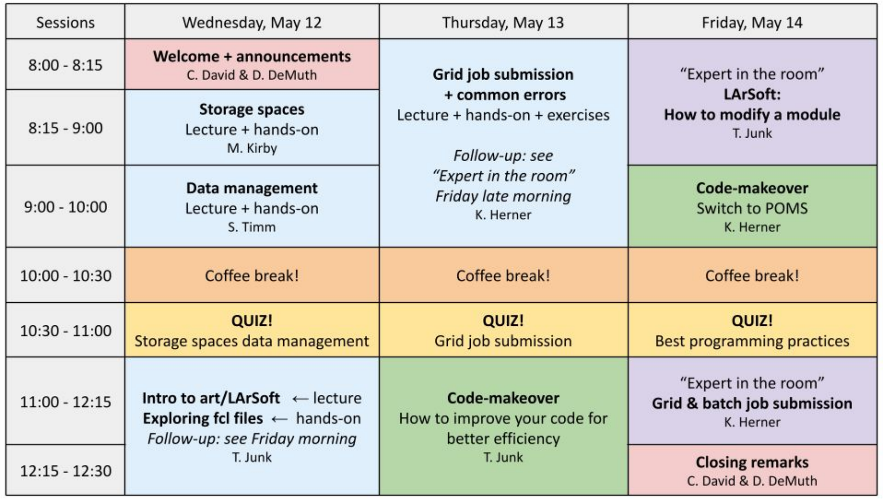
- With the first of four detector designs prototyped, blasting of the deep underground caverns underway at Lead, SD, detector installation anticipated for 2027, and science starting in 2029, computing system development has been vigorous.
- ArgoNeuT, and MicroBooNE are LArTPc experiments that have produced software that DUNE Computing has adopted to ensure efficient liquid argon detector designs, those **modelling techniques are highly numerical**, and sophisticated to use.
- Graduate students, post-docs, young and senior researchers are among the hundreds of collaborators added each year, with most **requiring general and specific software training**.
- While selected training can be outsourced to an evolving curricula provided by the HEP Software Foundation (HSF), some training remains specific to DUNE.



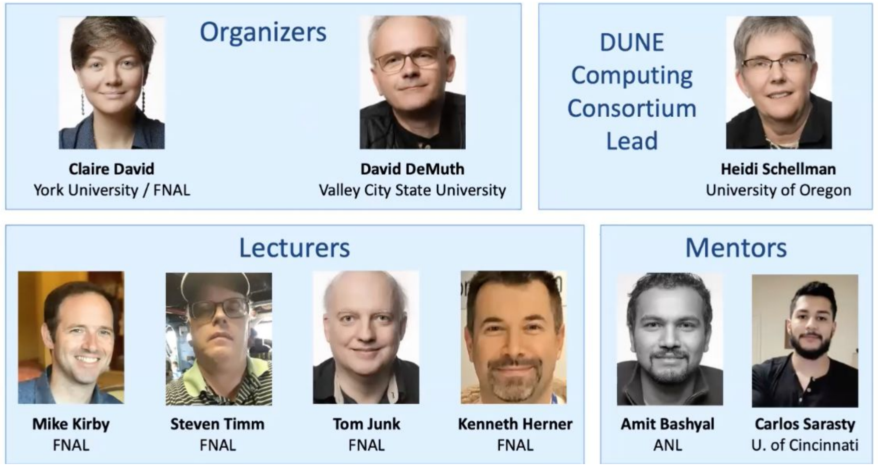
Worldwide and sophisticated, DUNE Computing Tutorials are used regularly to train new colleagues.

DUNE Trainings

- DUNE Computing first sponsored training in 2016, and since coordinated 10 events in various formats, with ~400 participants.
- Tutorials have focused on three topics:
  - Data storage and management,
  - LArSoft (software for LArTPCs),
  - Job submission and monitoring.
- The goal is to certify that new colleagues have access to DUNE computing resources and understand the basics of logging in, storage areas, running applications, code modifications, submitting and monitoring batch jobs.



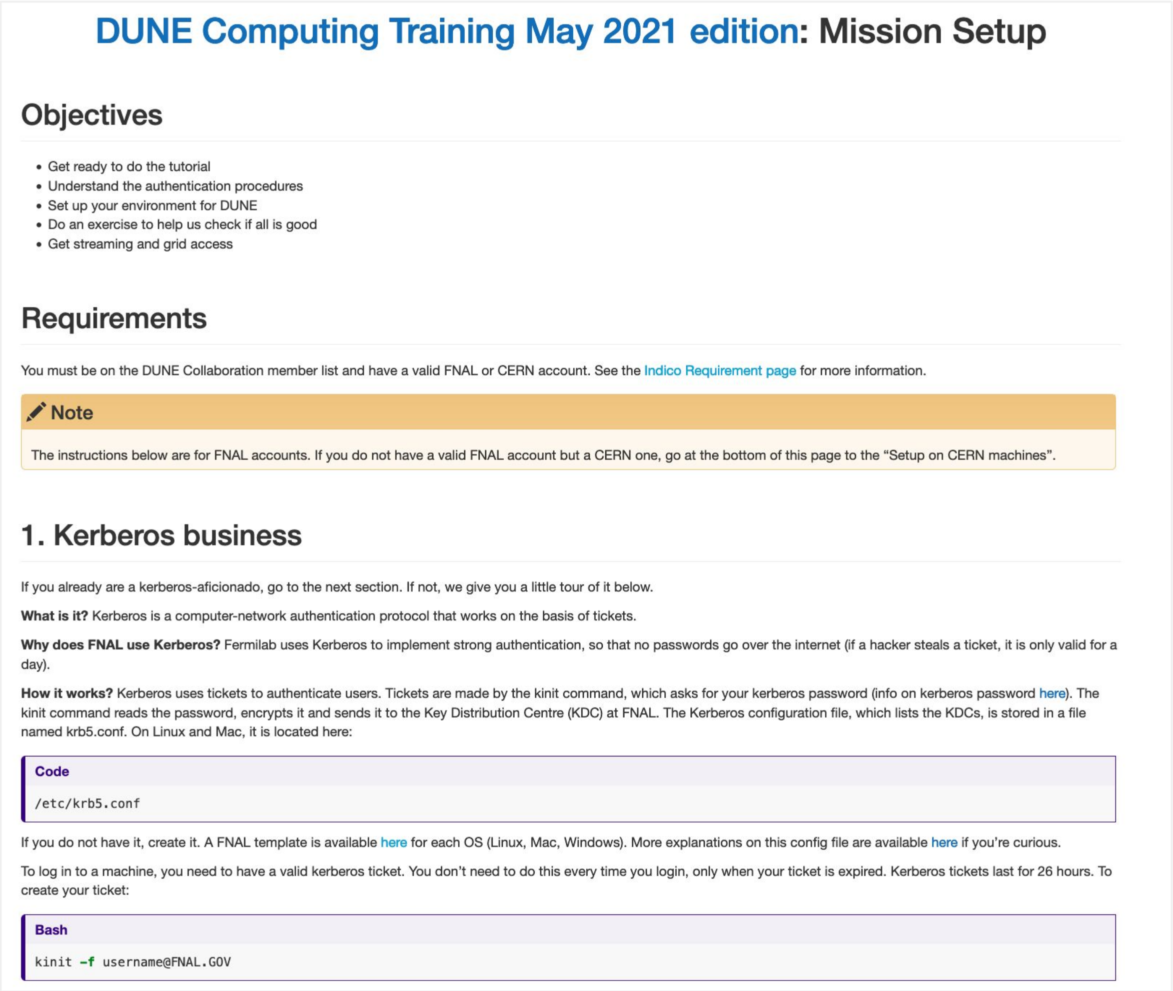
Sessions	Wednesday, May 12	Thursday, May 13	Friday, May 14
8:00 - 8:15	**Welcome + announcements** C. David & D. DeMuth	**Grid job submission + common errors** Lecture + hands-on + exercises *Follow-up: see "Expert in the room" Friday late morning* K. Herner	"Expert in the room" **LArSoft: How to modify a module** T. Junk
8:15 - 9:00	**Storage spaces** Lecture + hands-on M. Kirby		
9:00 - 10:00	**Data management** Lecture + hands-on S. Timm		**Code-makeover** Switch to POMS K. Herner
10:00 - 10:30	Coffee break!	Coffee break!	Coffee break!
10:30 - 11:00	**QUIZ!** Storage spaces data management	**QUIZ!** Grid job submission	**QUIZ!** Best programming practices
11:00 - 12:15	**Intro to art/LArSoft** ← lecture **Exploring fcl files** ← hands-on *Follow-up: see Friday morning* T. Junk	**Code-makeover** How to improve your code for better efficiency T. Junk	"Expert in the room" **Grid & batch job submission** K. Herner
12:15 - 12:30			**Closing remarks** C. David & D. DeMuth



The May 2021 training was offered as a three day event, each of four lecturers and two mentors doing the bulk of the work.

Training Logistics

- Event registration and communications are managed using Indico.
- Participants must verify their ability to use the Unix Shell.
- **Pre-event homework is vital** as it includes verification that participants can access the DUNE general purpose virtual machines, and ideally the corresponding CERN VM's (LX+).
- Livecoding, quizzes, expert in the room sessions, and assigned mentors ensure the event as hands-on.
- Each session is delivered and captured via Zoom, then embedded into the Software Carpentries lesson framework (SWC) which is hosted at DUNE Computing's GitHub site for review.



DUNE Computing Training May 2021 edition: Mission Setup

Objectives

- Get ready to do the tutorial
- Understand the authentication procedures
- Set up your environment for DUNE
- Do an exercise to help us check if all is good
- Get streaming and grid access

Requirements

You must be on the DUNE Collaboration member list and have a valid FNAL or CERN account. See the Indico Requirement page for more information.

Note

The instructions below are for FNAL accounts. If you do not have a valid FNAL account but a CERN one, go at the bottom of this page to the "Setup on CERN machines".

1. Kerberos business

If you already are a kerberos-aficionado, go to the next section. If not, we give you a little tour of it below.

What is it? Kerberos is a computer-network authentication protocol that works on the basis of tickets.

Why does FNAL use Kerberos? Fermilab uses Kerberos to implement strong authentication, so that no passwords go over the internet (if a hacker steals a ticket, it is only valid for a day).

How it works? Kerberos uses tickets to authenticate users. Tickets are made by the kinit command, which asks for your kerberos password (info on kerberos password here). The kinit command reads the password, encrypts it and sends it to the Key Distribution Centre (KDC) at FNAL. The Kerberos configuration file, which lists the KDCs, is stored in a file named krb5.conf. On Linux and Mac, it is located here:

Code

```
/etc/krb5.conf
```

If you do not have it, create it. A FNAL template is available here for each OS (Linux, Mac, Windows). More explanations on this config file are available here if you're curious.

To log in to a machine, you need to have a valid kerberos ticket. You don't need to do this every time you login, only when your ticket is expired. Kerberos tickets last for 26 hours. To create your ticket:

Bash

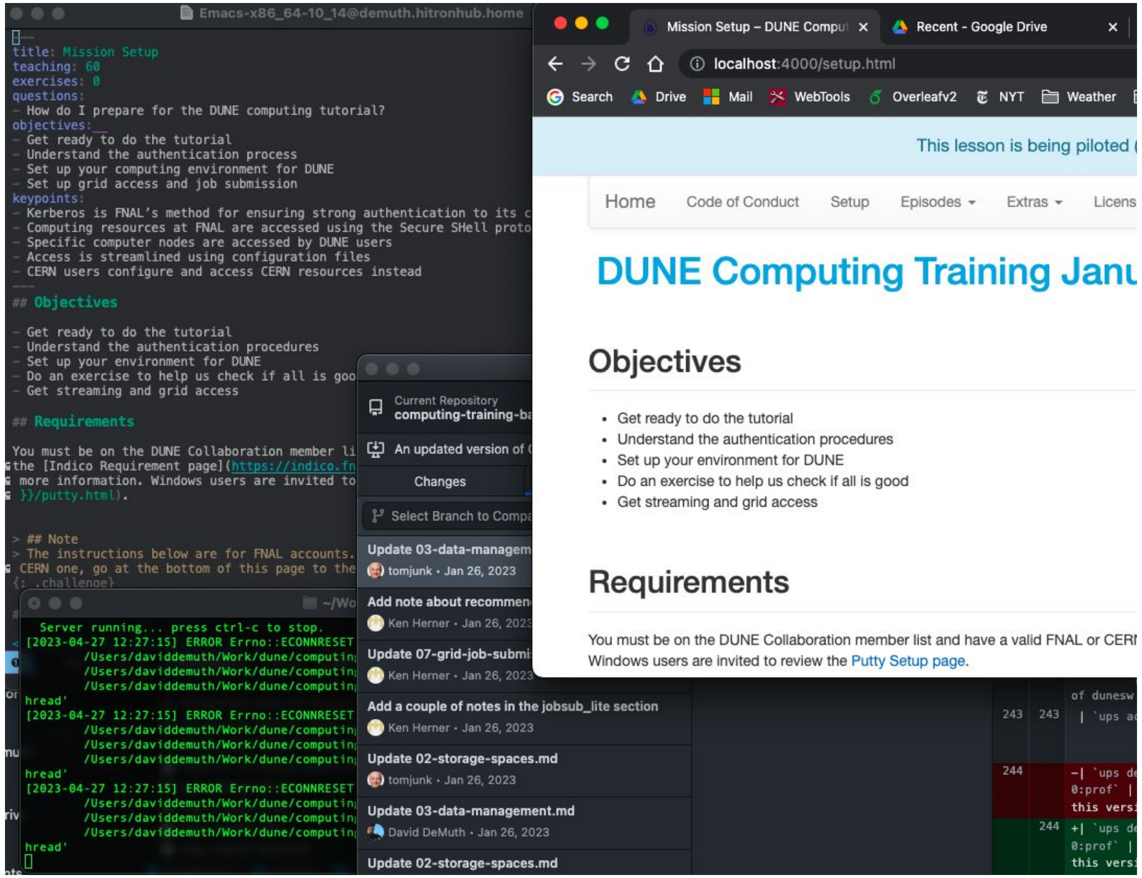
```
kinit -f username@FNAL.GOV
```

In advance of the opening salvo, students must demonstrate an understanding of using the Unix shell to access secure VMs.

Lesson Development

The infrastructure to develop lessons is provided in the Software Carpentries framework:

- A lesson template is imported as a new DUNE GitHub repo, configuration via a _config.yml file, and main lesson content as markdown files (.md) located in _episodes/.
- GitHub Desktop is used to manage the repository locally.
- Raw editing of files on github.com is frowned on, editing .md files locally is encouraged.
- Viewing edits in a localhost browser uses a Ruby/Jekyll rendering engine.
- As edits are verified, lessons are pushed to the site's main branch for access by multiple authors, and the public.
- Lessons are rendered elegantly on the web via GitHub Pages as a free service.



```
---
title: Mission Setup
teaching: 60
exercises: 0
questions:
- How do I prepare for the DUNE computing tutorial?
objectives:
- Get ready to do the tutorial
- Understand the authentication process
- Set up your computing environment for DUNE
- Set up grid access and job submission
keypoints:
- Kerberos is FNAL's method for ensuring strong authentication to its c
- Computing resources at FNAL are accessed using the Secure SHell proto
- Specific computer nodes are accessed by DUNE users
- Access is streamlined using configuration files
- CERN users configure and access CERN resources instead
---
## Objectives

- Get ready to do the tutorial
- Understand the authentication procedures
- Set up your environment for DUNE
- Do an exercise to help us check if all is goo
- Get streaming and grid access

## Requirements

You must be on the DUNE Collaboration member li
the [Indico Requirement page](https://indico.fn
 more information. Windows users are invited to
 }}/putty.html).


> ## Note
> The instructions below are for FNAL accounts.
 CERN one, go at the bottom of this page to the
{: .challenge}
```

```
 Server running... press ctrl-c to stop.
[2023-04-27 12:27:15] ERROR Errno::ECONNRESET
        /Users/daviddemuth/Work/dune/computin
        /Users/daviddemuth/Work/dune/computin
        /Users/daviddemuth/Work/dune/computin
hread'
[2023-04-27 12:27:15] ERROR Errno::ECONNRESET
        /Users/daviddemuth/Work/dune/computin
        /Users/daviddemuth/Work/dune/computin
        /Users/daviddemuth/Work/dune/computin
hread'
[2023-04-27 12:27:15] ERROR Errno::ECONNRESET
        /Users/daviddemuth/Work/dune/computin
        /Users/daviddemuth/Work/dune/computin
        /Users/daviddemuth/Work/dune/computin
hread'
```

Current Repository
computing-training-ba

An updated version of

Changes

Select Branch to Compa

Update 03-data-managem
tomjunk • Jan 26, 2023

Add note about recommen
Ken Herner • Jan 26, 2023

Update 07-grid-job-submi
Ken Herner • Jan 26, 2023

Add a couple of notes in the jobsub_lite section
Ken Herner • Jan 26, 2023

Update 02-storage-spaces.md
tomjunk • Jan 26, 2023

Update 03-data-management.md
David DeMuth • Jan 26, 2023

Update 02-storage-spaces.md

Mission Setup – DUNE Comput | Recent - Google Drive

localhost:4000/setup.html

This lesson is being piloted

Home | Code of Conduct | Setup | Episodes | Extras | Licens

DUNE Computing Training Janu

Objectives

- Get ready to do the tutorial
- Understand the authentication procedures
- Set up your environment for DUNE
- Do an exercise to help us check if all is good
- Get streaming and grid access

Requirements

You must be on the DUNE Collaboration member list and have a valid FNAL or CERI
Windows users are invited to review the Putty Setup page.

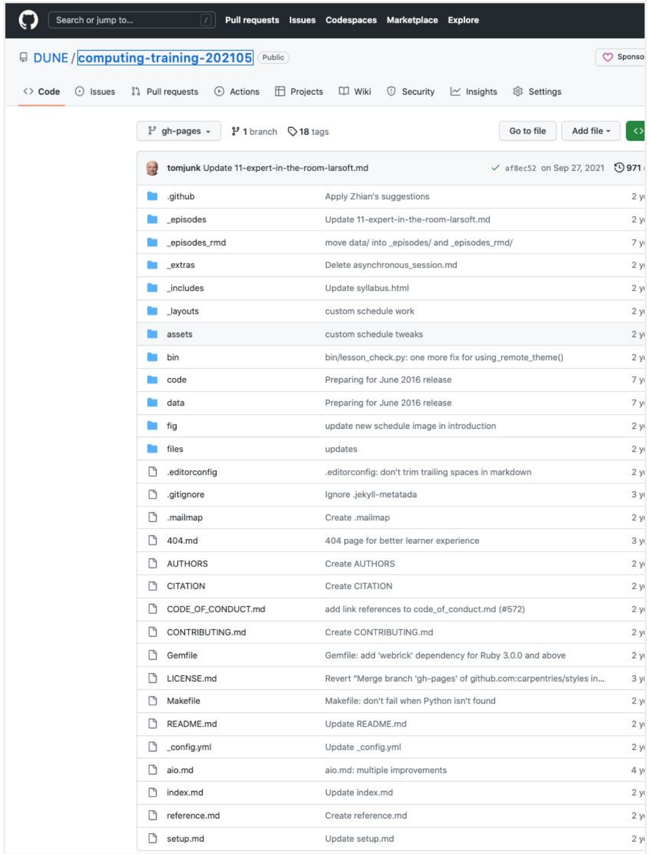
Once installed on a curriculum designers local machine, the lesson production environment is slick.

VALLEY CITY STATE UNIVERSITY

DUNE

Lesson Deployment

End users see:

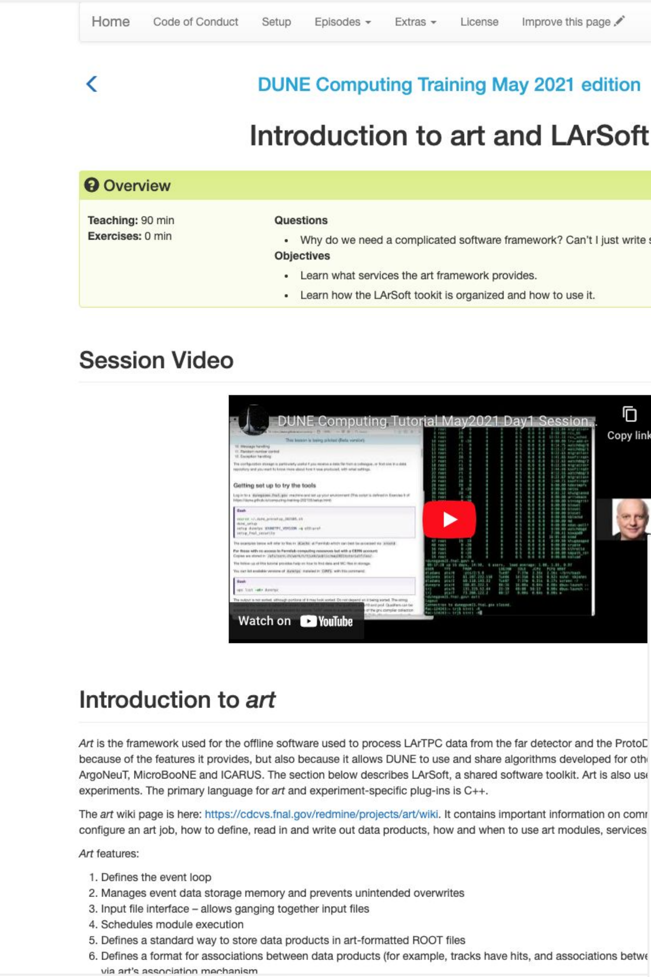


DUNE / computing-training-202105 Public

1 branch 18 tags

tomjunk Update 11-expert-in-the-room-larsoft.md ✓ af8ec52 on Sep 27, 2021 971

.github	Apply Zhian's suggestions	2 y
_episodes	Update 11-expert-in-the-room-larsoft.md	2 y
_episodes_rmd	move data/ into _episodes/ and _episodes_rmd/	7 y
_extras	Delete asynchronous_session.md	2 y
_includes	Update syllabus.html	2 y
_layouts	custom schedule work	2 y
assets	custom schedule tweaks	2 y
bin	bin/lesson_check.py: one more fix for using_remote_theme()	2 y
code	Preparing for June 2016 release	7 y
data	Preparing for June 2016 release	7 y
fig	update new schedule image in introduction	2 y
files	updates	2 y
.editorconfig	.editorconfig: don't trim trailing spaces in markdown	2 y
.gitignore	Ignore .jekyll-metatada	3 y
.mailmap	Create .mailmap	2 y
404.md	404 page for better learner experience	3 y
AUTHORS	Create AUTHORS	2 y
CITATION	Create CITATION	2 y
CODE_OF_CONDUCT.md	add link references to code_of_conduct.md (#572)	2 y
CONTRIBUTING.md	Create CONTRIBUTING.md	2 y
Gemfile	Gemfile: add 'webrick' dependency for Ruby 3.0.0 and above	2 y
LICENSE.md	Revert "Merge branch 'gh-pages' of github.com:carpentries/styles in...	3 y
Makefile	Makefile: don't fail when Python isn't found	2 y
README.md	Update README.md	2 y
_config.yml	Update _config.yml	2 y
aio.md	aio.md: multiple improvements	4 y
index.md	Update index.md	2 y
reference.md	Create reference.md	2 y
setup.md	Update setup.md	2 y



DUNE Computing Training May 2021 edition

Introduction to art and LArSoft

Overview

Teaching: 90 min
Exercises: 0 min

Questions

- Why do we need a complicated software framework? Can't I just write standalone code?

Objectives

- Learn what services the art framework provides.
- Learn how the LArSoft tookit is organized and how to use it.

Session Video

Introduction to *art*

Art is the framework used for the offline software used to process LArTPC data from the far detector and the ProtoD because of the features it provides, but also because it allows DUNE to use and share algorithms developed for oth ArgoNeuT, MicroBooNE and ICARUS. The section below describes LArSoft, a shared software toolkit. Art is also us experiments. The primary language for *art* and experiment-specific plug-ins is C++.

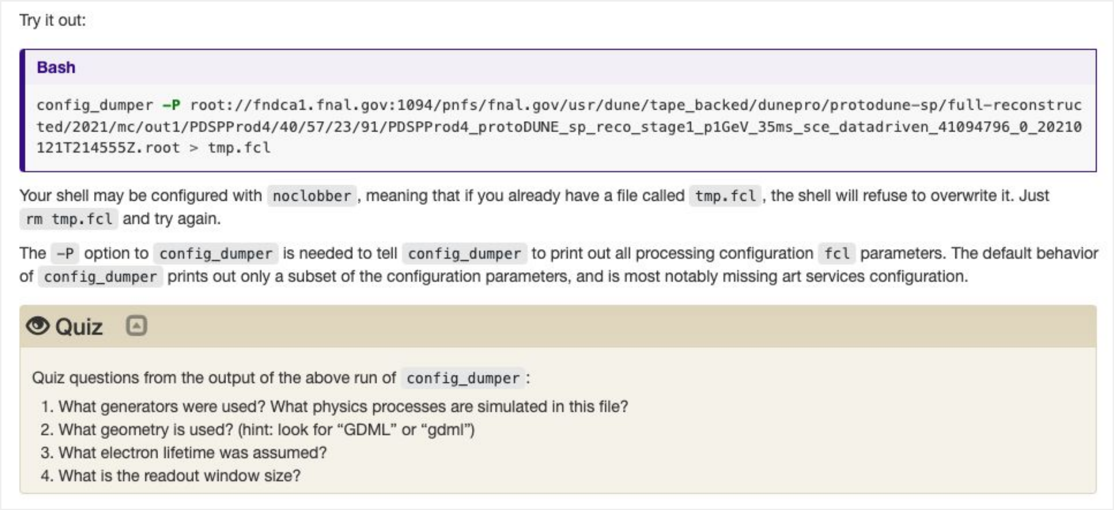
The *art* wiki page is here: https://cdcvs.fnal.gov/redmine/projects/art/wiki. It contains important information on com configure an art job, how to define, read in and write out data products, how and when to use art modules, services

Art features:

1. Defines the event loop
2. Manages event data storage memory and prevents unintended overwrites
3. Input file interface – allows ganging together input files
4. Schedules module execution
5. Defines a standard way to store data products in art-formatted ROOT files
6. Defines a format for associations between data products (for example, tracks have hits, and associations betw via art's association mechanism

Markdown language features:

- Simplified editing structure for web browsers,
- Bash script blocks easily copied and pasted in terminal windows,
- Drop down Quiz blocks that can be opened on demand.



Try it out:

Bash

```
config_dumper -P root://fndca1.fnal.gov:1094/pnfs/fnal.gov/usr/dune/tape_backed/dunepro/protodune-sp/full-reconstruc
ted/2021/mc/out1/PDSPProd4/40/57/23/91/PDSPProd4_protoDUNE_sp_reco_stage1_p1GeV_35ms_sce_datadriven_41094796_0_20210
121T214555Z.root > tmp.fcl
```

Your shell may be configured with `noclobber`, meaning that if you already have a file called `tmp.fcl`, the shell will refuse to overwrite it. Just `rm tmp.fcl` and try again.

The `-P` option to `config_dumper` is needed to tell `config_dumper` to print out all processing configuration `fcl` parameters. The default behavior of `config_dumper` prints out only a subset of the configuration parameters, and is most notably missing art services configuration.

Quiz

Quiz questions from the output of the above run of `config_dumper`:

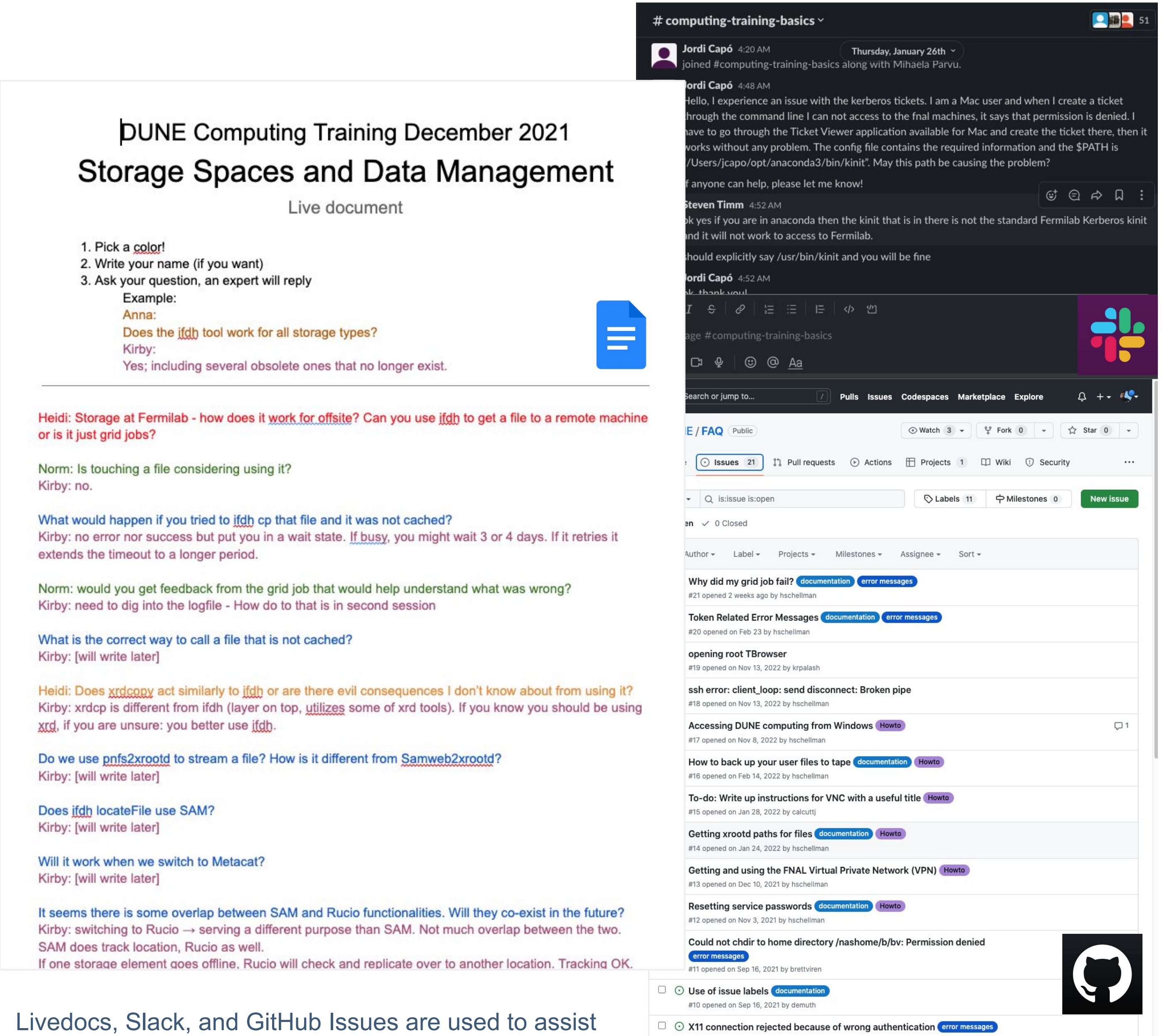
1. What generators were used? What physics processes are simulated in this file?
2. What geometry is used? (hint: look for "GDML" or "gdml")
3. What electron lifetime was assumed?
4. What is the readout window size?

GitHub's gp-pages are rendered seamlessly, and for free.

Getting Help

As expert instructors work through a lesson plan, often live coding in an adjacent window, multiple methods are used to ensure near- and long-term support:

- Instructors encourage participants to pose questions via a shared and world editable Google Doc (**Livedoc**) which is monitored by mentors and other experts.
  - Each participant selects a unique color to improve clarity.
  - As questions are volleyed, experts will form a dialogue with individuals to reason solutions; often a second expert will contribute.
  - Livedocs can be studied by others, solutions can be contrasted, and the Livedoc becomes a permanent educational resource.
- **Slack channels** are set up for each training event, where questions can be posed asynchronously, when questions receive attention from a cadre of experts.
- **GitHub Issues** has proven to be another method to field questions, provide mentorship, and ensure the novice become proficient in DUNE computing.



DUNE Computing Training December 2021

Storage Spaces and Data Management

Live document

1. Pick a color!
2. Write your name (if you want)
3. Ask your question, an expert will reply

Example:
Anna:
Does the ifdh tool work for all storage types?
Kirby:
Yes; including several obsolete ones that no longer exist.

Heidi: Storage at Fermilab - how does it work for offsite? Can you use ifdh to get a file to a remote machine or is it just grid jobs?

Norm: Is touching a file considering using it?
Kirby: no.

What would happen if you tried to ifdh cp that file and it was not cached?
Kirby: no error nor success but put you in a wait state. If busy, you might wait 3 or 4 days. If it retries it extends the timeout to a longer period.

Norm: would you get feedback from the grid job that would help understand what was wrong?
Kirby: need to dig into the logfile - How do to that is in second session

What is the correct way to call a file that is not cached?
Kirby: [will write later]

Heidi: Does xrdcopy act similarly to ifdh or are there evil consequences I don't know about from using it?
Kirby: xrdcp is different from ifdh (layer on top, utilizes some of xrd tools). If you know you should be using xrd, if you are unsure: you better use ifdh.

Do we use pnfs2xrootd to stream a file? How is it different from Samweb2xrootd?
Kirby: [will write later]

Does ifdh locateFile use SAM?
Kirby: [will write later]

Will it work when we switch to Metacat?
Kirby: [will write later]

It seems there is some overlap between SAM and Rucio functionalities. Will they co-exist in the future?
Kirby: switching to Rucio → serving a different purpose than SAM. Not much overlap between the two. SAM does track location, Rucio as well.
If one storage element goes offline, Rucio will check and replicate over to another location. Tracking OK.

computing-training-basics

51

Jordi Capó 4:20 AM
joined #computing-training-basics along with Mihaela Parvu.

Thursday, January 26th

ordi Capó 4:48 AM
Hello, I experience an issue with the kerberos tickets. I am a Mac user and when I create a ticket hrough the command line I can not access to the fnal machines, it says that permission is denied. I ave to go through the Ticket Viewer application available for Mac and create the ticket there, then it orks without any problem. The config file contains the required information and the $PATH is /Users/jcapo/opt/anaconda3/bin/kinit". May this path be causing the problem?

f anyone can help, please let me know!

teven Timm 4:52 AM
k yes if you are in anaconda then the kinit that is in there is not the standard Fermilab Kerberos kinit nd it will not work to access to Fermilab.

hould explicitly say /usr/bin/kinit and you will be fine

ordi Capó 4:52 AM
k, thank you!

age #computing-training-basics

Why did my grid job fail? documentation error messages
#21 opened 2 weeks ago by hschellman

Token Related Error Messages documentation error messages
#20 opened on Feb 23 by hschellman

opening root TBrowser
#19 opened on Nov 13, 2022 by krpalash

ssh error: client_loop: send disconnect: Broken pipe
#18 opened on Nov 13, 2022 by hschellman

Accessing DUNE computing from Windows Howto 1
#17 opened on Nov 8, 2022 by hschellman

How to back up your user files to tape documentation Howto
#16 opened on Feb 14, 2022 by hschellman

To-do: Write up instructions for VNC with a useful title Howto
#15 opened on Jan 28, 2022 by calcuttj

Getting xrootd paths for files documentation Howto
#14 opened on Jan 24, 2022 by hschellman

Getting and using the FNAL Virtual Private Network (VPN) Howto
#13 opened on Dec 10, 2021 by hschellman

Resetting service passwords documentation Howto
#12 opened on Nov 3, 2021 by hschellman

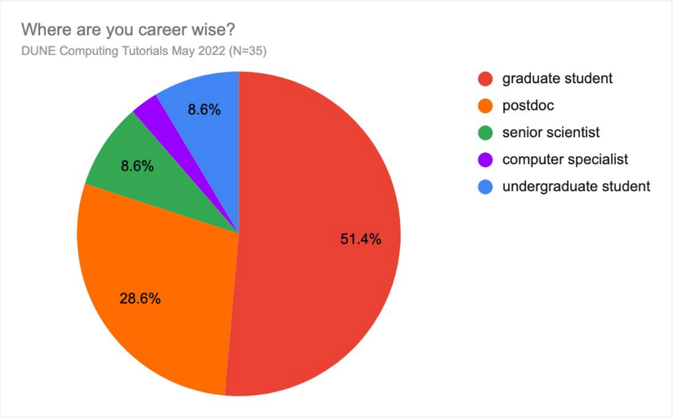
Could not chdir to home directory /nashome/b/bv: Permission denied error messages
#11 opened on Sep 16, 2021 by brettviren

□ Use of issue labels documentation
#10 opened on Sep 16, 2021 by demuth

□ X11 connection rejected because of wrong authentication error messages

Livedocs, Slack, and GitHub Issues are used to assist DUNE colleagues with computing questions.

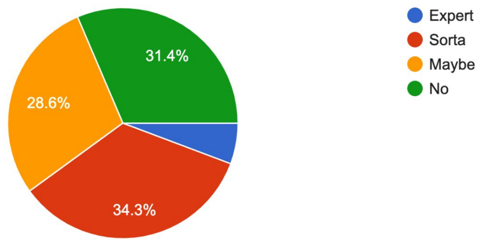
Student Snapshot


Where are you career wise?

DUNE Computing Tutorials May 2022 (N=35)

How familiar are you with running large batch jobs

35 responses



Graduate students and postdocs make up the majority of our training audience. It is exciting to see undergraduates are participating.

Few are familiar with large batch jobs.

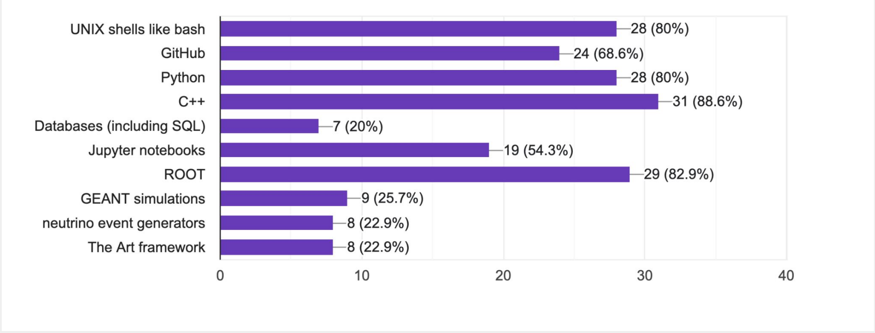
A strong majority know bash, use GitHub, Python, C++, and Root!

Over half have yet to start their physics analyses, and these were who we focused on in the event registration communications.

A healthy number of participants use GitHub for code backups.

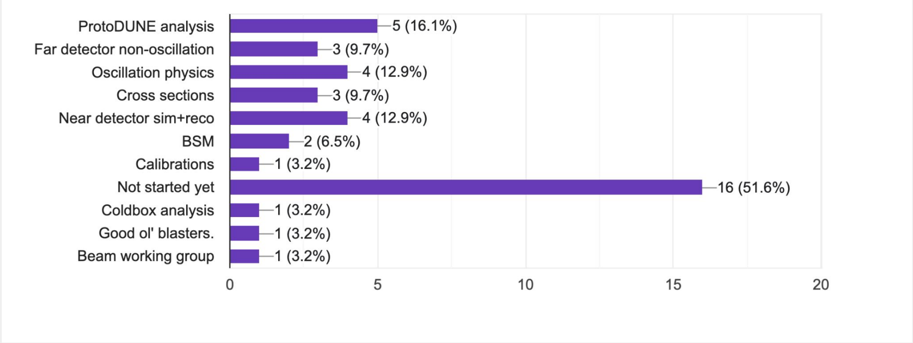
I am familiar with

35 responses



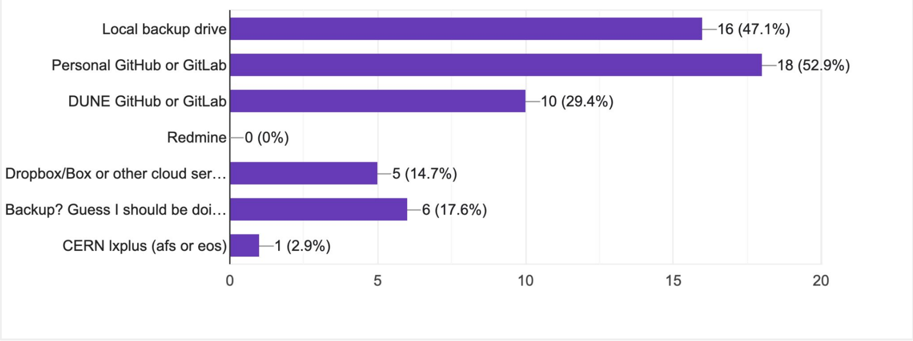
What physics/algorithm group are you working with, so far?

31 responses



Where do you back up your code?

34 responses



Below, one student provides feedback after participating in a training.

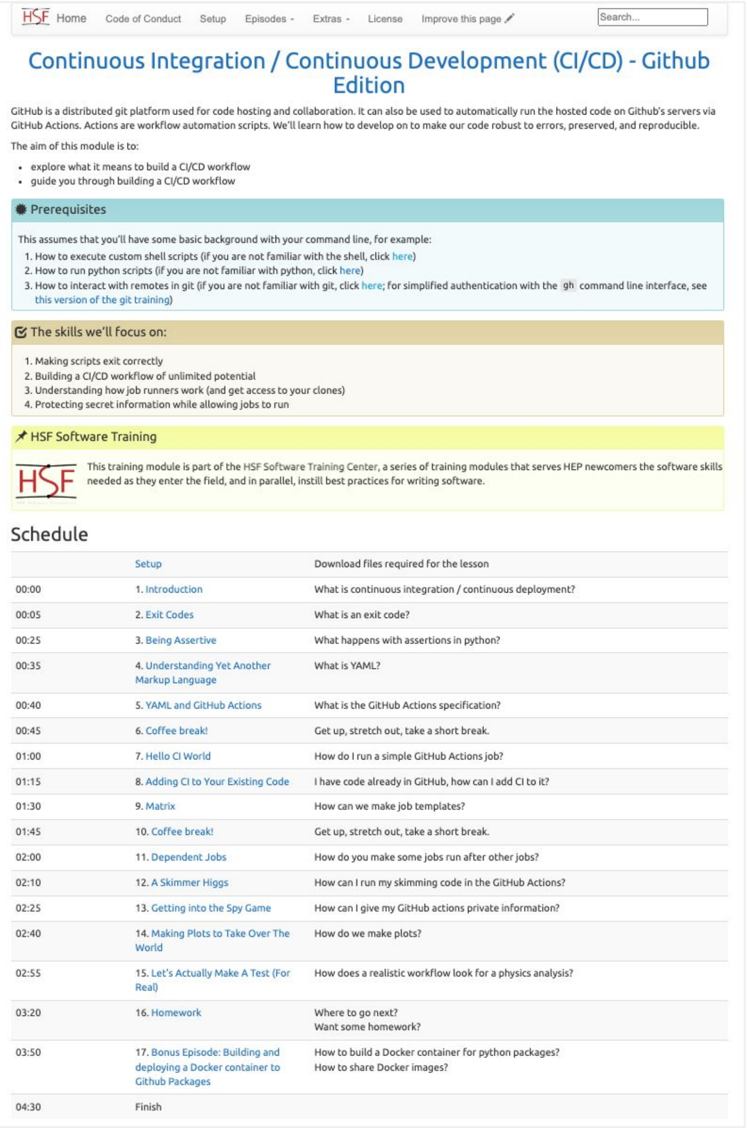
The FNAL grid job submission has always been a pain for us. It seems very cumbersome to run anything on grid. I found this fact from the oscillation analysis I am working on (some inherited job submission scripts) as well as from the past dune computing tutorials. We need set up so many softwares for LArSoft, dune tpc, mrb as well as things like xrootd, IFDH, SAM input, etc. This maybe very necessary and useful for experienced users, but it's very hard for undergraduate/early graduate to learn. Part of these is due to me not wrapping my head around all the softwares so far used at DUNE (and no one is teaching us).

Lessons Learned

- SWC lesson template is elegant, functional, and practical for delivering hands-on learning materials.
- Pre-event homework that includes checking that students can access FNAL servers is a must.
- It’s easy to be ambitious with the material for one half day introductions, two day, or three day events.
- Coffee breaks during trainings are important for assimilation.
- Mentors are essential to ensure skill development and understanding.
- Hybrid synchronous delivery is ideal.
- Zoom captures on YouTube allow asynchronous access, and a record of the event.
- Other approaches to training, such as Hackathons, could be used to take lessons to a next step.

HEP Software Skill Development

- The list of skills that experimental HEP physicists use on a regular basis is lengthy and evolving.
- Mastering data analysis techniques is not specific to one experiment.
- Using HSF training materials such as Unix Shell is effective and efficient.
- DUNE Computing encourages participation in HSF workshops.



HSF Home Code of Conduct Setup Episodes - Extras - License Improve this page Search...

Continuous Integration / Continuous Development (CI/CD) - Github Edition

GitHub is a distributed git platform used for code hosting and collaboration. It can also be used to automatically run the hosted code on Github's servers via GitHub Actions. Actions are workflow automation scripts. We'll learn how to develop on to make our code robust to errors, preserved, and reproducible.

The aim of this module is to:

- explore what it means to build a CI/CD workflow
- guide you through building a CI/CD workflow

Prerequisites

This assumes that you'll have some basic background with your command line, for example:

1. How to execute custom shell scripts (if you are not familiar with the shell, click here)
2. How to run python scripts (if you are not familiar with python, click here)
3. How to interact with remotes in git (if you are not familiar with git, click here; for simplified authentication with the `gh` command line interface, see this version of the git training)

The skills we'll focus on:

1. Making scripts exit correctly
2. Building a CI/CD workflow of unlimited potential
3. Understanding how job runners work (and get access to your clones)
4. Protecting secret information while allowing jobs to run

HSF Software Training

This training module is part of the HSF Software Training Center, a series of training modules that serves HEP newcomers the software skills needed as they enter the field, and in parallel, instill best practices for writing software.

Schedule

	Setup	Download files required for the lesson
00:00	1. Introduction	What is continuous integration / continuous deployment?
00:05	2. Exit Codes	What is an exit code?
00:25	3. Being Assertive	What happens with assertions in python?
00:35	4. Understanding Yet Another Markup Language	What is YAML?
00:40	5. YAML and GitHub Actions	What is the GitHub Actions specification?
00:45	6. Coffee break!	Get up, stretch out, take a short break.
01:00	7. Hello CI World	How do I run a simple GitHub Actions job?
01:15	8. Adding CI to Your Existing Code	I have code already in GitHub, how can I add CI to it?
01:30	9. Matrix	How can we make job templates?
01:45	10. Coffee break!	Get up, stretch out, take a short break.
02:00	11. Dependent Jobs	How do you make some jobs run after other jobs?
02:10	12. A Skimmer Higgs	How can I run my skimming code in the GitHub Actions?
02:25	13. Getting into the Spy Game	How can I give my GitHub actions private information?
02:40	14. Making Plots to Take Over The World	How do we make plots?
02:55	15. Let's Actually Make A Test (For Real)	How does a realistic workflow look for a physics analysis?
03:20	16. Homework	Where to go next? Want some homework?
03:50	17. Bonus Episode: Building and deploying a Docker container to Github Packages	How to build a Docker container for python packages? How to share Docker images?
04:30	Finish	

This excellent instructional model by the HSF introduces concepts of continuous integration and development using GitHub motivated DUNE Computing's use of the SWC lesson templates.

VALLEY CITY STATE UNIVERSITY

HEP Software

Data Management: XrootD, dCache, HDCondorCE, StashCache. CVMFS, Rucio CLI, SAM, MetaCat, Data Dispatcher

Reconstruction: Art, LArSoft, Pandora, Geant4, ROOT, WCT, PyTorch, UPS

Simulations: MARS, g4lbnf, Geant4, GDML, DUNENDGGD, GENIE, NuWRO, GIBUU, NEUT, CORSIKA, MUSUN/MUSIC, BXDECAY0, FLUKA, Geant4Reweight, dk2nu, PPFX, larnd-sim, edep-sim, CRY, garg4, ECAL, STT, GRAIN, OverlayGenie, Garfield, SPICE

Visualization: Bee, WebEVD, TEve

Analysis: CAF, CAFAna, CAFFEA, HighLAnd, Nuisance

Documentation: Wiki, GitHub, Read the Docs, Sphynx

Due to the vast suite of software used by the DUNE collaboration for the various analysis steps and sub-detectors, training is important, with some on an ad-hoc basis by physics and hardware groups, e.g. software systems which monitor QA/QC for installation of the far detector.

Future Work

- HEP experiments host a large number of collaborators, many have limited computer science training.
- The movement is for software to be reused by multiple experiments; common tools are anticipated.
- DUNE Computing's training work thus far has been focused on a few essential topics.
- Use of the Carpentries templates provides a proof of concept for other topics which have wider appeal.

HSF HEP Software Foundation

THE CARPENTRIES

iris hep

SIDIS Software Institute for Data-Intensive Sciences

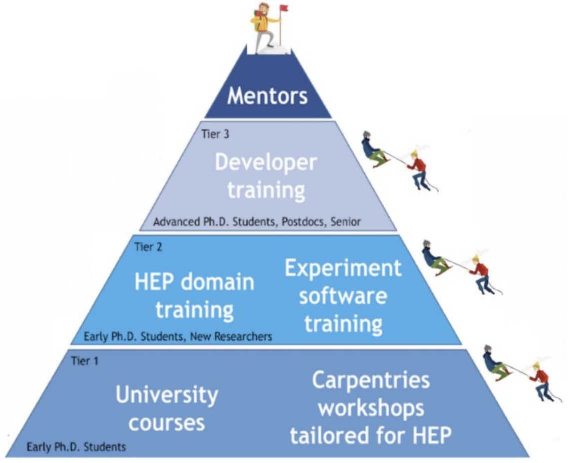

Mentors
Tier 3
Developer training
Advanced Ph.D. Students, Postdocs, Senior
Tier 2
HEP domain training
Experiment software training
Early Ph.D. Students, New Researchers
Tier 1
University courses
Carpentries workshops tailored for HEP
Early Ph.D. Students

Figure 1. Evolution of HEP Education and Training

A rough estimate is that there are ~10K people in the HEP community to train every year, ranging from undergraduates to scientists.

Software Training in High Energy Physics, Michel H. Villanueva, Sudhir Malik, Meirin Oan Evans

Summary

Students and postdocs who join HEP experiments who have no formal software training but become proficient with structured training, mentorship, and peer support.

DUNE Computing has developed multi-day training events to focus on the basics:

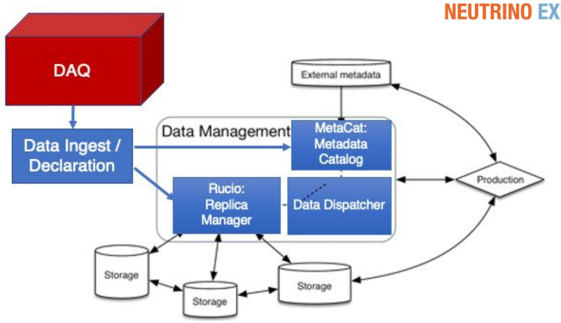
- Understanding data storage and best practices for operating in a cloud infrastructure,
- Learn event reconstruction and analysis techniques in liquid argon experiments,
- Master batch submission practices in a HPC grid environment.

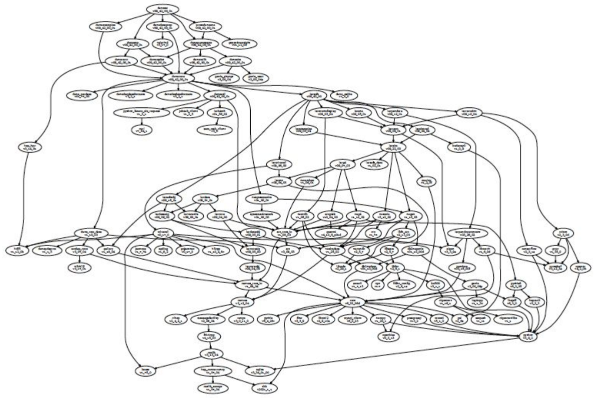
Our **aim is to jump start individual's simulation and reconstruction work**.

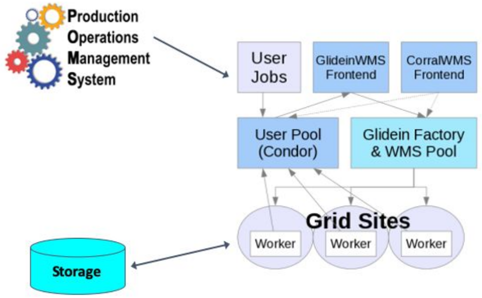
Expert instructors practice clear communication, produce materials that are straightforward, are open and patient with students, offer access for questions though livedocs, and long term support via Slack channels, GitHub, and email.

Rich with multicultural heritages themselves, DUNE collaborators are conscientious when developing curriculum to ensure the innate diversity becomes a feature for learning, not a bug.

DUNE Computing is committed to software training, recognizes that the materials developed for DUNE has pertinence for other experiments, and is working to contribute to the larger open science community.







CHEP 2023 Abstract

Providing computing training to the next generation of physicists is the principal driver for a biannual multi-day workshop hosted by the DUNE Computing Consortium. Materials are cast in the Software Carpentries templates, and to date topics have included storage space, data management, LArSoft, grid job submission and monitoring. Moreover, experts provide extended breakout sessions to demonstrate the intricacies of the unique software used in HEP analysis. Each workshop session uses live documents for real time correspondence, and are captured on Zoom; afterwards, videos are embedded on the corresponding webpages for review. As a GitHub repository, shared editing of the learning modules is straightforward, and provides a trusted framework to extend to other training topics in the future. An overview of the machinery will be provided, post workshop statistics will be discussed, with lessons learned will be the focus of this presentation.