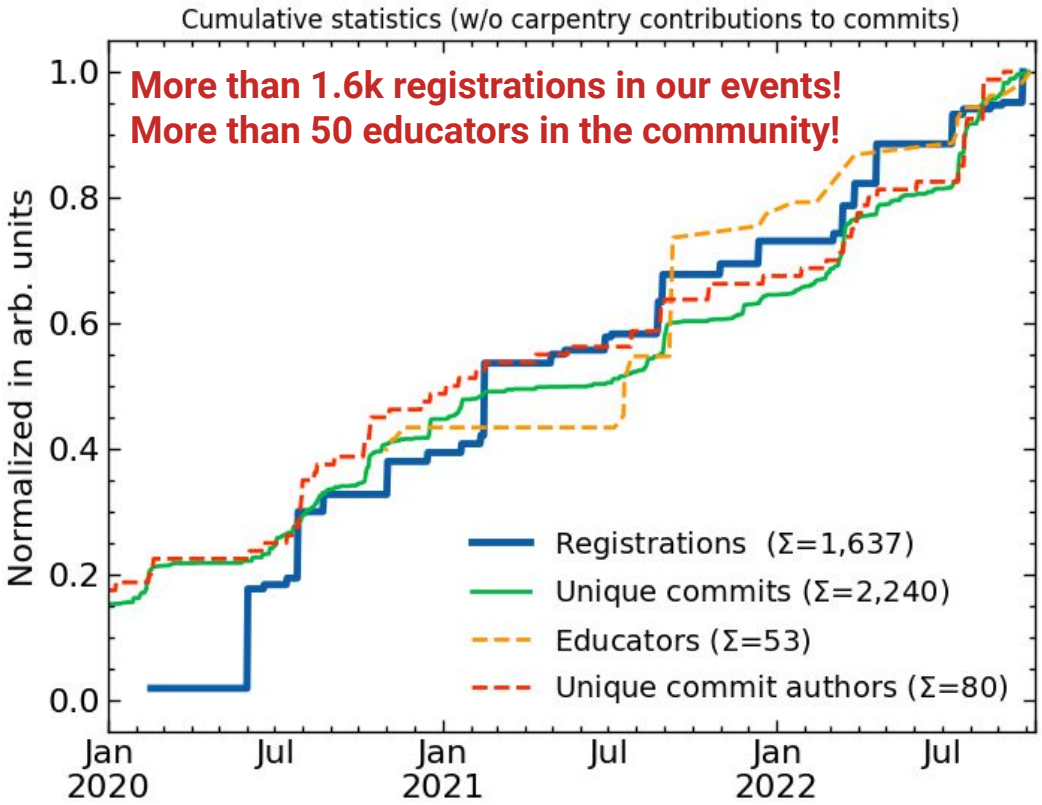
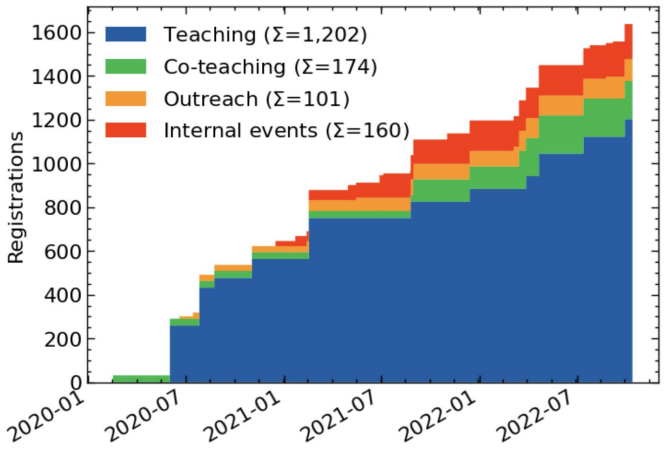
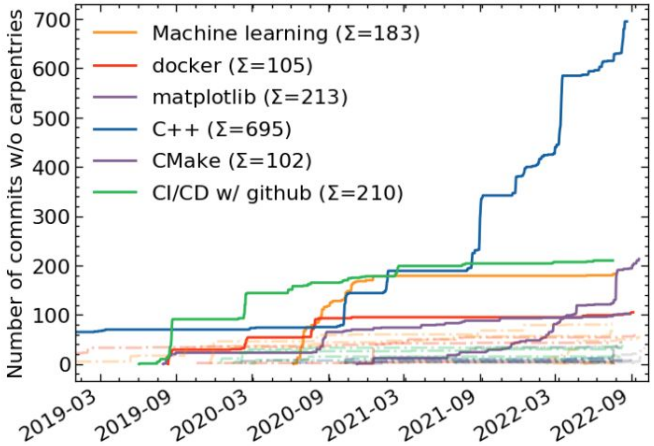
✅ Current status

Consistent output and steady community growth

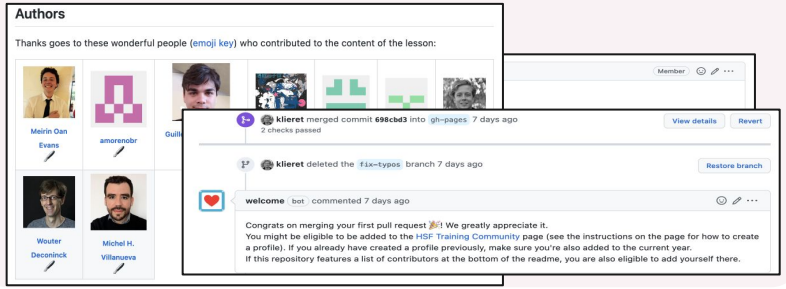




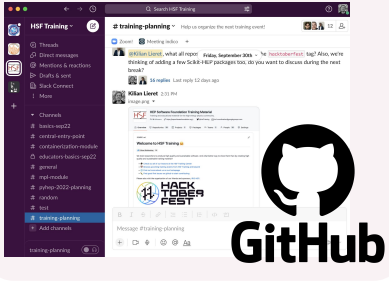


✅ Building a community

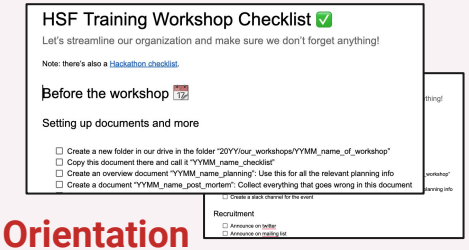
Recognition



Platforms



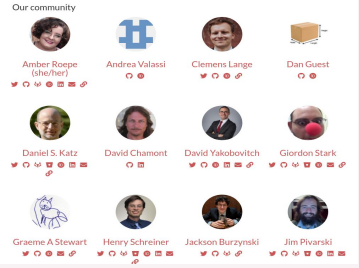
Orientation



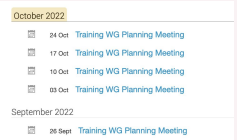
Increasing our reach

@hsftraining

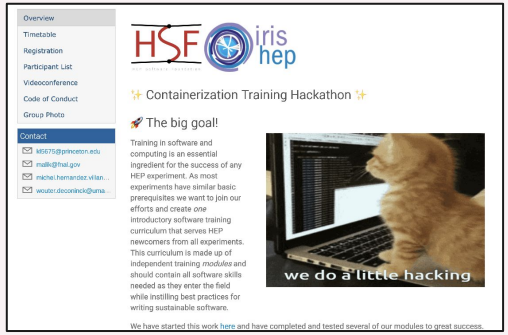
Community pages



Weekly meetings



Monthly Hackathons



Pivoting to intermediate-level training

Recently focused a lot on absolute prerequisites, teaching basic python, git, unix shell. **However:**

- **Only small window of opportunity** for effective training (need to fall within first ~2 weeks of initial HEP enrollment)
- Even basic ROOT/Scikit-HEP training is on different level
- Smaller community activation
- Unclear if we hit our core target group
- Intermediate trainings showed higher number of participants
- **All material is self-study ready with videos** → can set as prerequisite

Conclusions (preliminary):

- **Focus on intermediate target audience and HEP-specific tools**
  - Teach existing modules on containerization, analysis preservation, ML(+GPU)
  - Experiment again with asynchronous training + mentoring
- **Teach more scikit-HEP & IRIS-HEP packages**
  - Active community that is interested in growing its user base
  - Need separate workshops (teaching together with software basics is not optimal and reduces reach)
- Compile **new module wishlist** together with experiments
- Use of **fellowships** central to expanding training material and scaling further

Leveraging fellowships

"Training Fellow"

- **Project:** Organizational aspects; repository maintenance; new training material
- **Timeline:** Variable length, part-time possible
- **Requirement:** Preferably previous fellows if work on new training material
- **Supervision:** Training group + subject-matter expert
- **Start date:** **Soon?**

"Hybrid Fellow"

- **Project:** 80% dev (package used in production); 20% training + teaching
- **Timeline:** 3 month full time; 2 weeks after the initial month to be dedicated to the creation of material suitable for a workshop on X; will teach at least once
- **Supervision:** Training group + subject-matter expert
- **Start date:** Together with other fellowships

"Web Fellow"

- **Project:** Central-entry point project; anything web-technology
- **Timeline:** Short-term
- **Requirement:** Skilled, little technical supervision needed
- **Supervision:** Various
- **Start date:** Independent

Expand Training Center

Current Training Center

Served us well, but becoming crowded

Basics

The UNIX Shell
A guide through the basics of the file systems and the shell.
Start learning now!
Contribute!

Version controlling with git
Track code changes, undo mistakes, collaborate. This module is a must.
Start learning now!
Contribute!

Programming with python
Get started with an incredibly popular programming language.
Start learning now!
Contribute!

SSH
Introduction to the Secure Shell (SSH)
Status: Early development
Start learning now!
Contribute!

Machine learning
Get behind the buzzword and teach machines to work for you intelligently!
Start learning now!
Watch the videos!
Contribute!

Matplotlib for HEP
Make science prettier with beautiful plots!
Status: Beta testing
Start learning now!
Contribute!

Software Development and Deployment

Version controlling with git
Track code changes, undo mistakes, collaborate. This module is a must.
Start learning now!
Contribute!

CI/CD (gitlab)
Continuous integration and deployment with gitlab.
Start learning now!
Watch the videos!
Contribute!

CI/CD (github)
Continuous integration and deployment with github actions.
Start learning now!
Watch the videos!
Contribute!

Docker
Introduction to the docker container image system.
Start learning now!
Watch the videos!
Contribute!

Singularity
Introduction to containerization with Singularity/Apptainer.
Status: Early development
Start learning now!
Contribute!

Unit testing
Unit testing in python.
Status: Beta testing
Start learning now!
Contribute!

"Central Entry Point"

Various Filters → Scales!

Inspiration: learn.astropy.org

Format
- tutorial (25)
- guide (1)

Astropy packages
Search here…
- coordinates (10)
- astroquery (9)
- units (8)
- modeling (4)
- wcs (4)
- spectral cube (3)
- table (3)
- synphot (2)
- constants (1)
- convolution (1)

Show more

Python packages
Search here…
- matplotlib (14)

Search here… Search by algolia

GUIDE **CCD Data Reduction Guide**
The purpose of this text is to walk through image reduction and photometry using Python, especially Astropy and its affiliated packages. It assumes some basic familiarity with astronomical ima
for IRAF, "A Use
Stellar CCD Pho

TUTORIAL **Modeling 2: Create a Us**
In this tutorial,
model and with

TUTORIAL **Extracting and Plotting Position-Velocity Diagrams**
In this tutorial, we will extract position-velocity (PV) diagrams from a spectral cube and plot them.

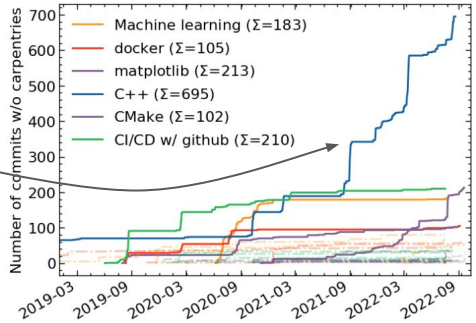
TUTORIAL **Cube Reprojection Tutorial**

Vision: One website as entry point for all HEP-related training material (by anyone)

Filters could include: Experiment, prog. language, package, analysis techniques, packages, level, curriculum, …

Collaborating with experiments

- **Our objective:** Find blind spots and motivated contributors; bring together new teams
- **Previous meetings:** Good overview of different training approaches, but unclear actionable outcomes
- **Need new survey** (building on results of 2019):
  - Explicitly also targeting supervisors/experts: "To which workshop would you send how many students?" / "To which project would you contribute?", etc.
  - Non-anonymous to follow up
  - Concrete training modules



Want to find topics that are

- In use by multiple experiments similarly
- Training neglected but desired by multiple experiments
- Clear target audience
- Motivated contributors available

Timeline:

- **Oct:** Send out survey
- **Nov:** Select most promising topics and search for existing material; reach out to interested experiments & people
- **Dec:** First hackathon on topic: Create repository and ToC; determine further schedule

🧠 Brainstorming 🌪️

We've already collected > 5 pages of suggestions in the last weeks,
but there's always room for more

- Which **IRIS-HEP projects** could benefit from regular training activities?
- How can the training group be better **integrated with the remaining team**?
- What are the **"blind spots"** that could take off like the C++ training given some initial activation energy?
- Which projects are best for the **"hybrid" fellows**? Who wants "a part" of the **web fellow**?

Find us

@hepsoftfound

@hsf-training

hepsoftwarefoundation.org