

# Sustainable Software Training Delivery at the HEP Software Foundation

On behalf of the HSF Training Working Group:

Wouter Deconinck (University of Manitoba),

Michel Hernandez Villanueva (DESY)

Kilian Lieret (Princeton University),

Sudhir Malik (University of Puerto Rico Mayaguez)

**Sustainable HEP 2022**

September 5, 2022



# Gap Between Student Training and Research Needs

HEP (and increasingly also NP, astro, etc) continues to be a **computationally intensive and data driven** field with **innovative data processing techniques**.

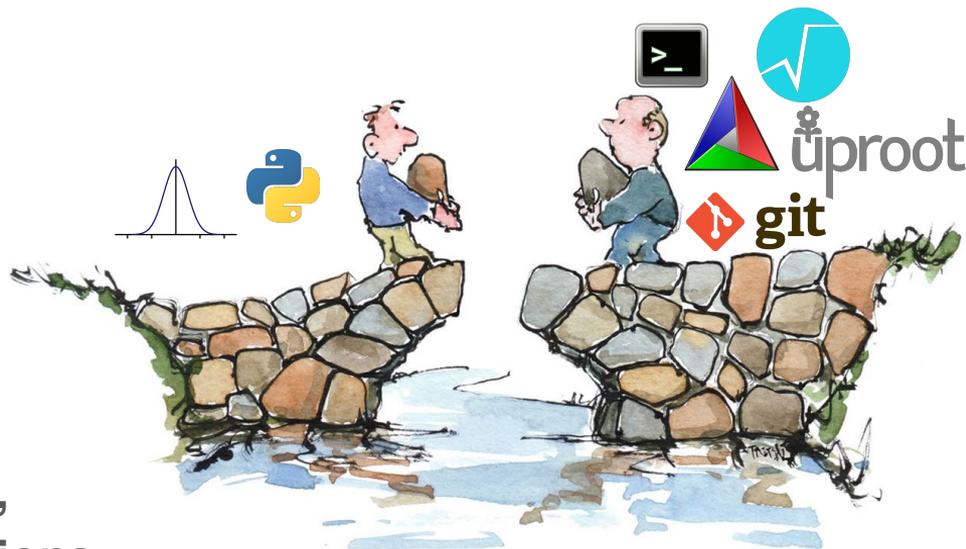
There is a gap between what **students are trained** in when they enter research:

- simple statistical techniques
- introductory programming

... and what **HEP research requires**:

- advanced statistical tools (AI/ML)
- software engineering skills
- programming for HPC and HTC

**True for leading research universities,  
but especially true for smaller institutions.**



# But, How Do We Offer Software Training Scalably?

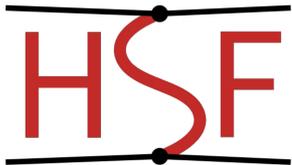
HEP Software Foundation Community White Paper Working Group – Training, Staffing and Careers (2018), [arXiv: 1807.02875 \[physics.ed-ph\]](https://arxiv.org/abs/1807.02875)

- Provide **incentives and encouragement to offer software training**,
- Assign **credit to software development** as a scientific discipline, including training activities,
- Establish **policies for the hiring and retention** of researchers specialising in computing,
- Address the **gap in formal software training** that is not always given by universities as part of a physicist's education.

# Analysis of Types of Software Skills Needed

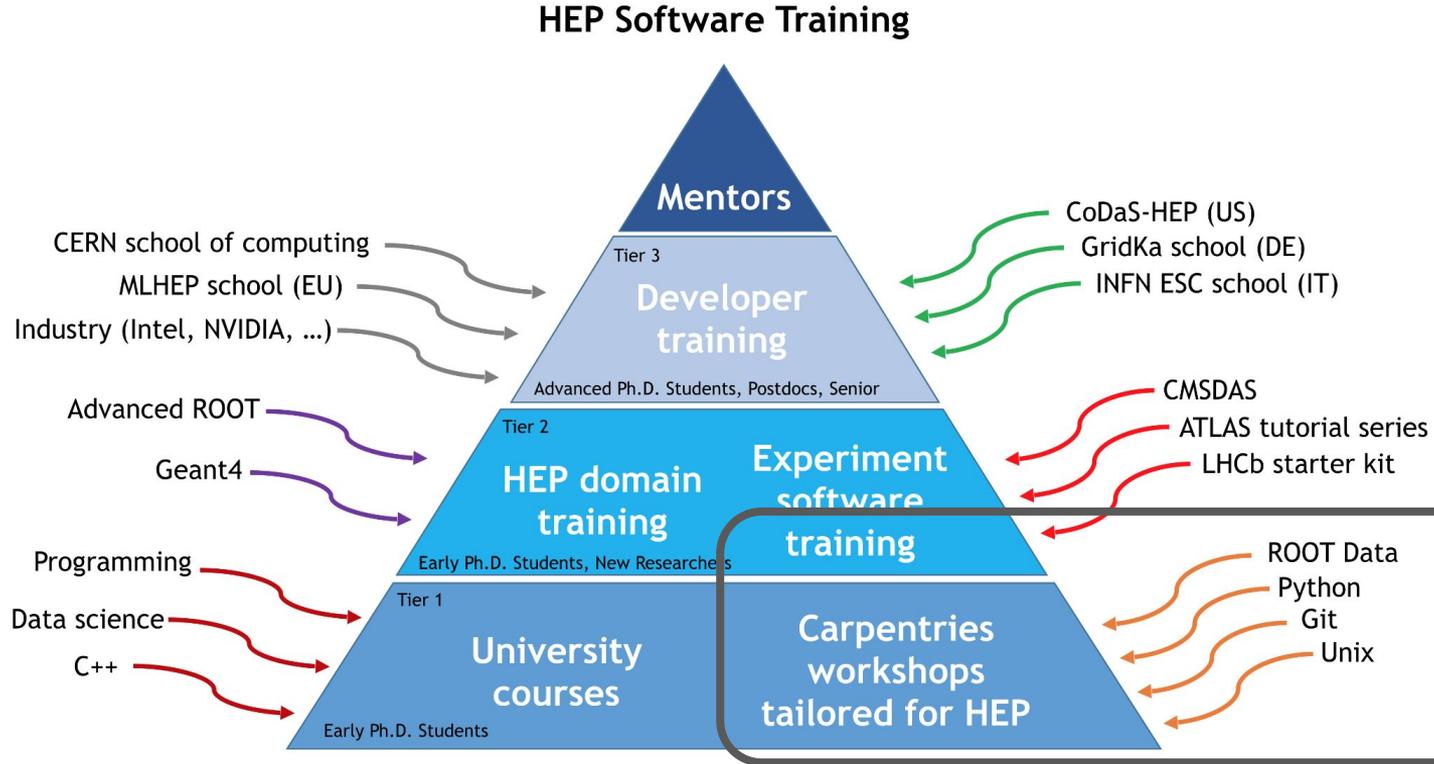
Software Training in HEP (2021), [DOI: 10.1007/s41781-021-00069-9](https://doi.org/10.1007/s41781-021-00069-9)

1. **Fundamental and generic software engineering** (e.g., unix, version control, C++, and continuous integration),
2. **Domain-specific HEP packages and practices** (e.g., the ROOT data format and analysis framework),
3. **More advanced knowledge involving specialized techniques**, including programming for HPC and HTC, machine learning and data science tools, and techniques to maintain software projects at all scales.



This is forming the basis for the activities  
of the **HSF Training Working Group!**

# Software Training Pyramid



# HSF Training Partners

## Partner organizations

Working with experiment training groups, HEP initiatives supported by NSF (such as **IRIS-HEP**, **FIRST-HEP**), by CERN (**SIDIS**), and organisations like **Software Carpentry**



**SIDIS**  
Software Institute for  
Data-Intensive Sciences



**THE  
CARPENTRIES**

## Joint goals

- **Standardizing** of software training curriculum
- **Facilitating coordination** between experiments (e.g. in coordination with the HSF Data Analysis working group, e.g. July 27 “Data analysis training in HEP experiments”, <https://indico.cern.ch/event/1175097/>)
- **Providing guidelines** for organization of HEP software training events, **whether in-person or virtual**, see e.g. <https://hepsoftwarefoundation.org/training/howto-event.html>

# Until 2020: In-Person Training Workshops

HSF Training software tutorials through 2020:

- **In-person participation only** (or self-study)
- Approximately 35 participants per workshop

Impact on ecological and social sustainability:

- **Typical carbon footprint**  $\sim 0.5$  t CO<sub>2</sub>e / person:
  - Intra-continental travel: 0.4 t CO<sub>2</sub>e per person
  - Hotel stays:  $\sim 25$  kg CO<sub>2</sub>e per person per night
  - Compare with estimated average EU (US) annual carbon footprint of 7 (16) t CO<sub>2</sub>e per person
  - **A workshop increases one's footprint by 5% to 10%**
- **Travel limits the accessibility** to research groups with access to sufficient grant funding



# 2020: HSF Training Pivot to Virtual Workshops

Shift to virtual training events over the past 3 years:

- COVID-motivated, but this new training modality is here to stay
- Since 2018: 17 online software trainings, **1300+ participants trained**
- **Broader participation** from HEP and related fields, beyond CERN focus

Advantages of virtual workshops:

- Large worldwide audience (few 10s to 100+), basic training offered ~quarterly
- Recording and archiving of entire workshops
- Logistics easier: no PC rooms

But also disadvantages:

- Lower engagement, distractions
- Meaningful interactions harder



# Training Curriculum: The Basics

## Basics

<h3>The UNIX Shell</h3> <p>A guide through the basics of the file systems and the shell.</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>	<h3>Version controlling with git</h3> <p>Track code changes, undo mistakes, collaborate. This module is a must.</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>	<h3>Programming with python</h3> <p>Get started with an incredibly popular programming language.</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>	<h3>SSH</h3> <p>Introduction to the <b>Secure Shell (SSH)</b></p> <p>⚠️ Status: Early development</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>	<h3>Machine learning</h3> <p>Get behind the buzzword and teach machines to work for you intelligently!</p> <p>📖 Start learning now!</p> <p>📺 Watch the videos!</p> <p>🔧 Contribute!</p>	<h3>Matplotlib for HEP</h3> <p>Make science prettier with beautiful plots!</p> <p>⚡ Status: Beta testing</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>
--	---	--	---	---	--

## Software Development and Deployment

<h3>Version controlling with git</h3> <p>Track code changes, undo mistakes, collaborate. This module is a must.</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>	<h3>CI/CD (gitlab)</h3> <p>Continuous integration and deployment with <b>gitlab</b>.</p> <p>📖 Start learning now!</p> <p>📺 Watch the videos!</p> <p>🔧 Contribute!</p>	<h3>CI/CD (github)</h3> <p>Continuous integration and deployment with <b>github actions</b>.</p> <p>📖 Start learning now!</p> <p>📺 Watch the videos!</p> <p>🔧 Contribute!</p>	<h3>Docker</h3> <p>Introduction to the <b>docker</b> container image system.</p> <p>📖 Start learning now!</p> <p>📺 Watch the videos!</p> <p>🔧 Contribute!</p>	<h3>Unit testing</h3> <p>Unit testing in python.</p> <p>⚡ Status: Beta testing</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>	<h2>HSF Software Training Center:</h2> <p><a href="https://hepsoftwarefoundation.org/training/curriculum.html">hepsoftwarefoundation.org/training/curriculum.html</a></p>
---	---	---	---	--	---

## C++ corner

<h3>HEP C++ Course</h3> <p>A full introduction to C++ based on a series of slides and exercises.</p> <p>📖 Start learning now!</p> <p>📺 Watch the videos!</p> <p>🔧 Contribute!</p>	<h3>Basic Modern C++</h3> <p>A brand new C++ course that is currently in development.</p> <p>⚠️ Status: Early development</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>	<h3>Build systems: <b>cmake</b></h3> <p>Building code is hard. <b>CMake</b> makes it easier.</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>
---	---	--

## HEP specific tools

<h3><b>uproot</b></h3> <p>Reading and writing ROOT files without having to install ROOT.</p> <p>⚡ Status: Beta testing</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>	<h3>SciKit HEP</h3> <p>A collection of packages for particle physics analyses in Python.</p> <p>⚡ Status: Beta testing</p> <p>📖 Start learning now!</p> <p>🔧 Contribute!</p>
--	--

# Typical HSF/Carpentries Training Layout: Docker

## ✦ The HSF Training Curriculum



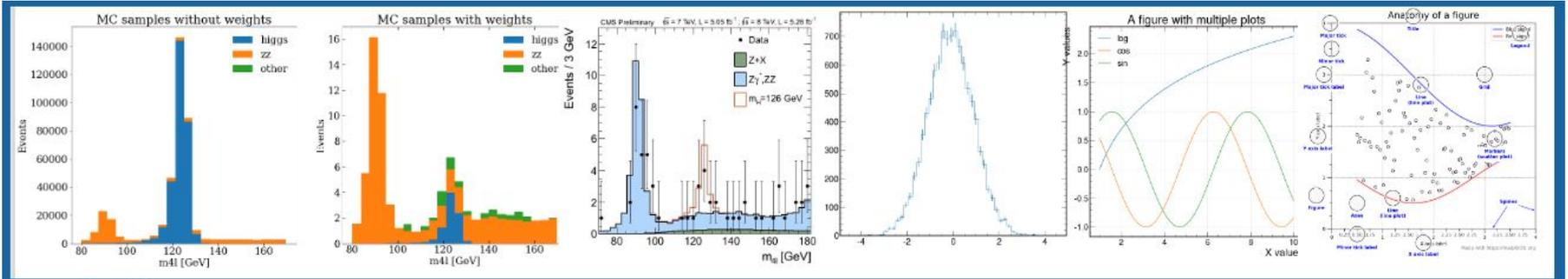
This training module is part of the Training Curriculum, a series of training modules that serves HEP newcomers the software skills needed as they enter the field, and in parallel, instill best practices for writing software.

## Schedule

	Setup	Download files required for the lesson
00:00	1. Introduction	What are containers?
00:10	2. Pulling Images	How are images downloaded? How are images distinguished?
00:25	3. Running Containers	How are containers run? How do you monitor containers? How are containers exited? How are containers restarted?
00:45	4. File I/O with Containers	How do containers interact with my local file system?
01:05	5. Writing Dockerfiles and Building Images	How are Dockerfiles written? How are Docker images built?
01:35	6. <a href="#">Removal of Containers and Images</a>	How do you cleanup old containers? How do you delete images?
01:45	7. Coffee break	Coffee or tea?
02:00	8. Gitlab CI for Automated Environment Preservation	How can gitlab CI and docker work together to automatically preserve my analysis environment? What do I need to add to my gitlab repo(s) to enable this automated environment preservation?
02:45	9. Running our Containerized Analysis	How do I run my full analysis chain inside docker containers?
03:30	10. Optional: Running Containers on LXPLUS Using Singularity	How can I run a container on LXPLUS?
03:45	11. Bonus: Using CMD and ENTRYPOINT in Dockerfiles	How are default commands set in Dockerfiles?
03:45	12. Challenge Examples	How to do a few more things?
04:05	Finish	

The actual schedule may vary slightly depending on the topics and exercises chosen by the instructor.

# Regular Schedule of Training Workshops



## Matplotlib Training

21–22 Apr 2022  
Virtual  
Europe/Zurich timezone



## Software Carpentry (Virtual)

28–30 Mar 2022  
US/Eastern timezone



## Software Carpentry (Virtual)

13–15 Dec 2021  
Europe/Zurich timezone



# Next Workshop: September 28-30 ([indico](#))

September 28:

- Unix shell
- Git

September 29:

- Python

September 30:

- Uproot and Scikit-HEP
- ROOT and Jupyter
- RDataFrame in Python

The screenshot shows a detailed workshop schedule for Wednesday, 28 September. The schedule is organized into time slots with corresponding activities and speakers.

Time	Activity	Speakers
09:30 → 10:00	Help with setup 30 min before start of the workshop	
10:00 → 10:20	Introduction: Introduction & group picture	Kilian Lieret, Michel Hernandez Villanueva (DESY), Sudhir Malik (University of Puerto Rico (US)), Wouter Deconinck
10:20 → 11:30	Unix	Devasena Inupakutika, Zongru (Doris) Shao (CASUS/HZDR)
11:30 → 12:00	Coffee Break	
12:00 → 13:00	Unix: Automating Tasks with the Unix Shell	Devasena Inupakutika, Zongru (Doris) Shao (CASUS/HZDR)
13:00 → 14:00	Lunch	
14:00 → 15:00	Git	Devasena Inupakutika, Zongru (Doris) Shao (CASUS/HZDR)
15:00 → 15:30	Coffee Break	
15:30 → 17:00	Git	Devasena Inupakutika, Zongru (Doris) Shao (CASUS/HZDR)

# HSF Training Workshop: Challenges

- As a Carpentries member, we can offer several workshops with instructors from the global Carpentries community: not always familiar with HEP (but that's generally OK for the basics of python, unix shell, git).
- We have facilitated HSF community members in becoming certified Carpentries instructors so we can be less dependent on Carpentries.
- We need to **find additional community instructors** because of limited resources of developers of ROOT, scikit-hep,...
- We need to **develop a pipeline of educators**, from students to mentors to primary instructors.

# And Join Our Efforts

- As a **facilitator** (organization of a virtual training workshop, e.g. targeted to a specific community)
- As an **instructor**: no Carpentries training required for the HEP-specific components, and you will gain teaching experience
- As a **mentor** during an upcoming workshop: best way to get your feet wet :-)

**Weekly meetings:** Mondays at 4pm CEST

<https://indico.cern.ch/category/10294/>

Open to all!



# Further Information on HSF Training Workshops

- **Training events:** <https://indico.cern.ch/category/11386/>
- **Material:** All the training modules developed so far resides: <https://hepsoftwarefoundation.org/training/curriculum.html>
- **Community:** Our training community is listed here: <https://hepsoftwarefoundation.org/training/community.html>
- **Procedure:** how to request and organize a training: <https://hepsoftwarefoundation.org/training/howto-event.html>

**Funding:** Provided by the IRIS-HEP/FIRST-HEP





# IRIS-HEP Training Grand Challenge 2 (Nov 1, 2021)

## Scalability

- All students and postdocs should be able to receive training in both the introductory material and the more advanced material.
- Scale with the number of incoming students each year.

## Sustainability

- Community processes for continuous renewal of both instructors and training materials.

## Training Scope

- Curriculum with introductory, intermediate, and advanced material
- Broadly meet the needs of the community.
- Evolve over time as needed.

## Diversity and Inclusion

- Participation should represent our community.
- Aim to represent society at large through earlier engagement in training pipeline.

## HSF IRIS-HEP Training Challenge 2

1 November 2021  
Europe/Zurich timezone

Enter your search term

Overview
Timetable
Contribution List
Registration
Participant List
Videoconference
Contact
✉ kilian.lieret@posteo.de
✉ Sudhir.Malik@cern.ch
✉ michel.hernandez.villan...

## HSF IRIS-HEP Training Challenge 2

### The big goal!

The HSF Training Working Group has brought together the HEP community and has focused efforts not only on running a number of training events, but also the development of a first draft of a common curriculum (*Comput Softw Big Sci* 5, 22 (2021)).

The effort encompasses both funded projects such as IRIS-HEP, FIRST-HEP (recently joined by SWIFT-HEP in the UK), but also a large number of enthusiastic volunteers. In this workshop we propose to define, plan and build on and expand the effort in the coming 3 years by defining a clear target in the form of a community "Training Challenge".



### What are the next steps?

We are now working to define, with the larger community, a series of specific goals for the period 2021-2023 in four categories and to work with the community to achieve them.

**Scalability** - We aim for sufficient scalability in the training activities such that all students and postdocs can receive training in both the introductory material and the more advanced material. In the steady state we expect a required scale approximately equal to the number of incoming students each year.

**Sustainability** - We aim to develop community processes by which both the instructors involved in training activities, and the training materials themselves, are continually renewed and meet the other two goals.

**Training Scope** - We aim for a curriculum (introductory, intermediate, advanced) that broadly meets the needs of the community and evolves over time as needed.

**Diversity and Inclusion** - The participation in the training should be representative of our community and (as we engage earlier in the pipeline) should work to represent the society at large