Demystifying High Performance Computing for Life Scientists: Curriculum for Intro to HPC

Todd Fallesen[1], Camille Charoy[1], Ruaridh Gollifer[2], Camilla D.K. Harris[2], Stefania Marcotti[1], Kimberly Meechan[2], John Roche[3]

1- Crick Advanced Light Microscopy, The Francis Crick Institute
2- Advanced Research Computing Centre, University College London
3- ITO, High Performance Computing Team, The Francis Crick Institute

Funding Provided by The Crick/University Partner Networking Fund

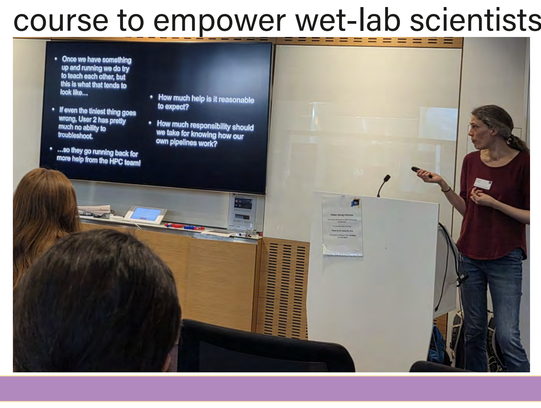
Introduction

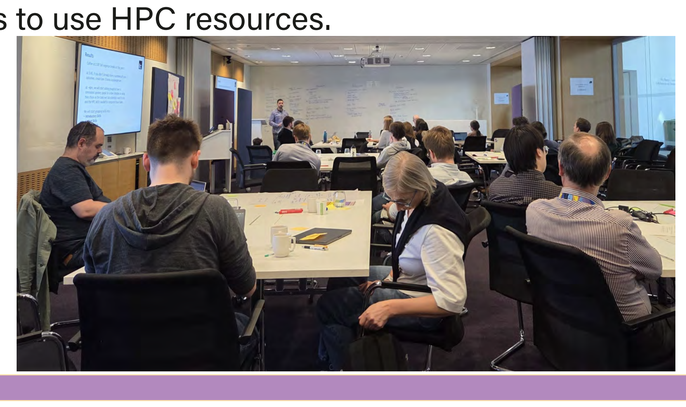
High Performance Computing (HPC) has become an important tool for wet-lab scientists, particularly those working in BioImage analysis. Microscopy data analysis was traditionally performed on personal computers, but with the rise of complex datasets, where terabyte sized mutli-dimensional image sets are increasingly common, analysis must move to high performance computing. HPC training is commonly designed for scientists with previous computational experience, comfortable interacting with computers at a low-level way, rather than those who have limited scientific computing experience, such as many wet-lab scientists.

We held a one-day workshop at the Francis Crick Institute where we brought together wet-lab scientists, bioimage analysts, bioinformaticians and HPC engineers. The wet-lab scientists presented their motivations for using HPC in their research, where they stressed that HPC was seen as more of a tool to finish their biological analysis, rather than something they wanted to learn in depth. Together the group discussed what skills in computing and HPC wet-lab scientists would need to do their own analysis.

Through this workshop, we realised that any curriculum to teach HPC would have to include some basic computing as well, as many learners wouldn't have prior exposure to low-level computing.

We created an 8-hour curriculum structured to give both foundational computing and introductory HPC skills to wet-lab scientists. This curriculum was trialled at The Francis Crick Institute to positive feedback. We present this curriculum here as an adaptable framework for other institutions to design a course to empower wet-lab scientists to use HPC resources.





Student Motivation

To understand their motivation to attend, we asked wet lab scientists why they would take a course in HPC.

- Manage large datasets unmanageable on personal computers
- Desire to use machine learning analysis
- Visualization of large 3D image sets
- Build and maintain analysis pipelines for their data
- Scientists don't want to ask their collaborators to do their work for them

Adult learners want to learn skills to satisfy a need. Making the course releavant to those needs will be more beneficial than a generic course.

Learning Outcomes

- Independently access HPC resources
- Confidently request appropriate resources
- Manipulate large image data sets efficiently
- Implement a simple image analysis pipeline on HPC
- Request assistance from local points of contact
- Access additional support material and resources

Course Tracks

The course material is augmented by image analysis specific examples. We have created three sets of examples using common image analysis workflows: CellProfiler, Python and Fiji. These examples are taught along with the core course material. In all cases, we give examples of using shell scripting to perform image file management, and to run image analysis pipelines reproducibly.

Curriculum

**Lesson with Image Analysis Software Example*

Introductory Material (30 minutes)
- Institutional resources:
  - Local Points of Contact
  - Available infrastructure and costs
- Introduce learners to LLM's
  - Warn against reliance and pitfalls

Introduction to the Shell (90 minutes)
- Accessing the command line
- Introduction to Shell
  - Highlight GUI and CLI differences
  - Why we use CLI in HPC
- File and directory navigation and management*
- Running a shell script*

Introduction to HPC (5 hours)
- Why use a cluster?!
- Connecting to a remote system (SSH)
- Different types of Nodes
- Exploring remote file systems*
- Transferring files*
- Scheduler Fundamentals
- Accessing software via Modules*

Being a good HPCitizen (1 hour)
- Using resources effectively
- Using shared resources responsibly
- Backing up data and scripts appropriately
- How to get help

"Asynchronous" material that students can work through on their own is an important part of our curriculum. Lessons highlighted in orange can be structured as aynchronous content. This would allow for a shorter in-class component and give students useful reference materials while working alone. A thorough wiki and jargon buster for the HPC installation is also valuable.