PASC23: Research Software Engineers (RSEs) in HPC
Monday 26th June 2023

Developing Research Software Skills Pathways to Support the Future of High Performance and Exascale Computing

Jeremy Cohen

EPSRC RSE Fellow, Department of Computing

Director of Research Software Engineering Strategy

Imperial College London

Photo by Ali Shah Lakhani on Unsplash

Overview

- **Background**
  - Research is changing!
  - The ”skills gap”
- **Learning Pathways**
  - Why do we need them?
  - Challenges in developing learning pathways
- **UNIVERSE-HPC**
  - Brief overview of the UNIVERSE-HPC project
- **Future work and next steps**

Photo by Ali Shah Lakhani on Unsplash

Research is changing!

- **We are seeing an accelerating shift towards “digital research”**
  - Technical elements now part of research across almost all domains
  - Work is increasingly computationally intensive – bigger data, larger simulations, novel architectures, etc.
  - Greater complexity, more specialist skills, things change quickly
- **Who provides the necessary technical skills?**
  - Traditionally, often (self taught?) researchers
  - Now, increasingly Research Software Engineers and other RTPs
  - Big changes in the software, data and infrastructure fields
  - Different drivers and aims when working in a research context
- **Why is this important?**
  - Greater focus on sustainability and management of software / data
  - REPRODUCIBILITY!

Photo by Ali Shah Lakhani on Unsplash

The “skills gap”

- **There is now extensive, easily accessible beginner-level training material**
    - Carpentries, CodeRefinery, university modules, online tutorials, etc.

- **Intermediate and advanced material much harder to find**

- **How do we bridge the “skills gap” between beginner knowledge and expert professionals?**

Advanced/expert-level
The skills gap
Beginner-level

Photo by Ali Shah Lakhani on Unsplash

The “skills gap”: Basic-level Skills

- **Many different training and skills providers supporting the research community – some open source examples:**
- **The Carpentries (https://carpentries.org/)**
  - Software Carpentry; Data Carpentry; Library Carpentry
  - Unix Shell; Introductory Python / R, Version Control (Git)
- **CodeRefinery (https://coderefinery.org/) – Basic to intermediate**
  - Version Control (Git), Jupyter notebooks, testing, reproducible research, modular code development
- **HPC Carpentry* (https://www.hpc-carpentry.org/) – Basic to intermediate**
  - Python for HPC, Chapel, Terminal, parallelization, Snakemake

*HPC Carpentry is not part of the core Carpentries suite but is a related community project.

Photo by Ali Shah Lakhani on Unsplash

The “skills gap”: Advanced Skills

- **Limited scope for general training**
  - Courses run by groups including ARCHER2, PRACE, Pawsey, …
  - Courses can be focused around specific infrastructure.
- **Much skill development the result of on-the-job training and experience**
  - Takes a long time to build the necessary experience and skills.
- **Infrastructure, hardware, software all changing rapidly**
  - At the advanced level, in particular, need ongoing development of new skills.
- **Proper pathways and structures around training are key**

Photo by Ali Shah Lakhani on Unsplash

Learning and skills pathways (I)

- **Why do we need them?**
  - Research Software Engineering is a cross-cutting discipline.
  - Practitioners come from a range of research or industry backgrounds.
  - Need to provide routes that highlight how to gain necessary competencies from a range of starting points.

- **Support more effective and structured skill development**
  - Self-taught developers frequently learn only the skills they need , when they need them.
  - Easy to miss key related skills and best practices that support, e.g. quality, efficiency, …
  - Possible to progress a long way on limited foundational knowledge – this can lead to long-term future challenges.

Photo by Ali Shah Lakhani on Unsplash

Learning and skills pathways (II)

- **Huge increase in the need for computational skills in research**
  - Software, data and computation now a core part of research lifecycle in almost all domains.
  - Demand is growing but there is a skills shortage.
  - Time to develop specialist skills too great for the speed of growth of their use in research.

- **Clear and structured pathways should**
  - Stimulate growth of training material development in required areas.
  - Reduce the time for RSEs/researchers to develop key skills.
  - Provide a clearer idea of where to go next for practitioners wanting to expand their learning.

Photo by Ali Shah Lakhani on Unsplash

Challenges in developing learning pathways

- **Domain-based or skills-based structure?**
  - e.g. “I currently know basic Python development and how to use Git/GitHub, what should I be looking to know next?”
  - e.g. “I am a computational chemist with three years of experience including developing software, what skills should I be looking to learn to make me a more effective researcher?”

- **Skills-based structures are easier to develop but may not map directly onto researchers expertise – likely to be better for RSEs**

- **Many different domains, different views on what’s important**

Photo by Ali Shah Lakhani on Unsplash

Developing learning pathways: An example

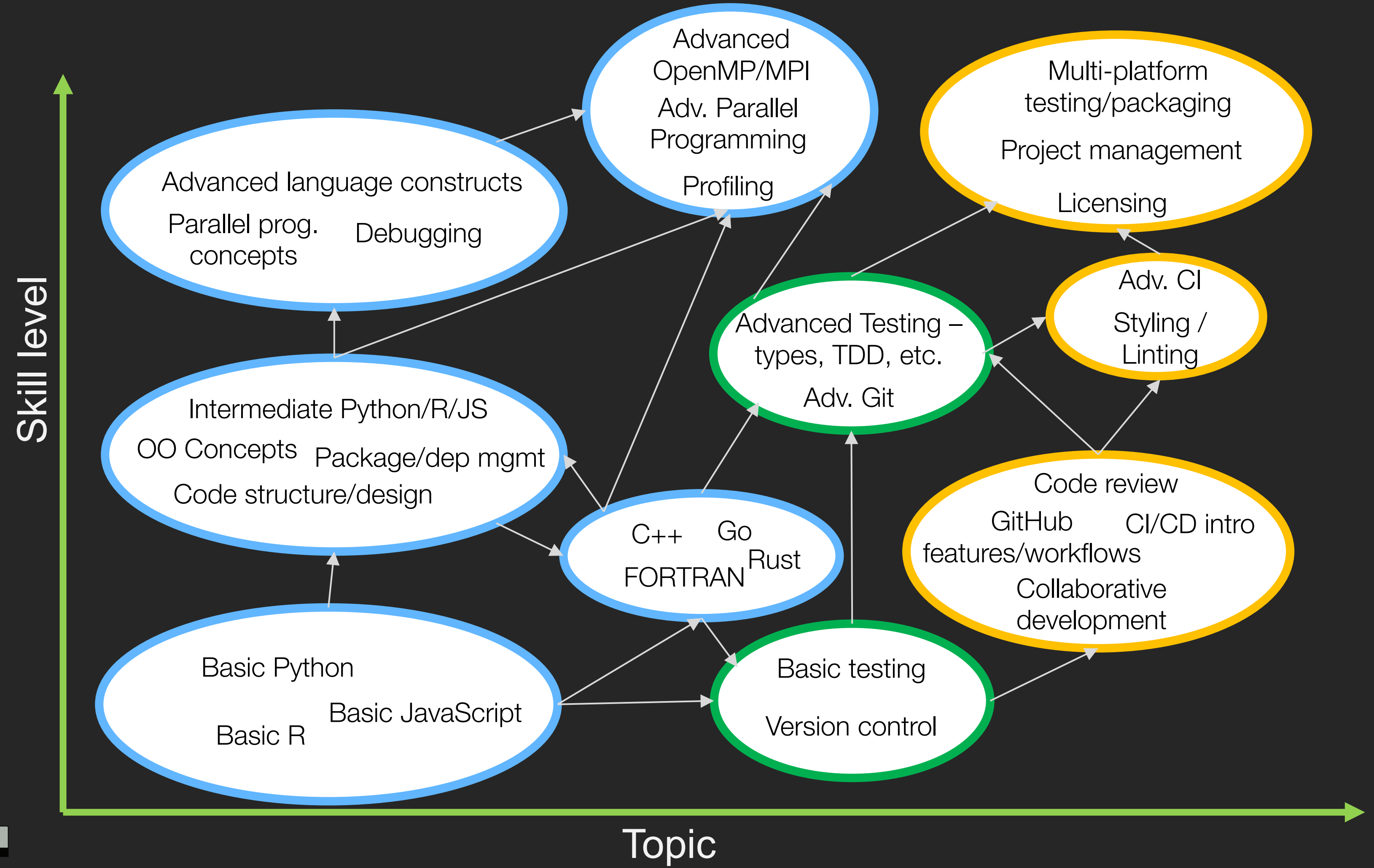

Photo by Ali Shah Lakhani on Unsplash

UNIVERSE-HPC:

Understanding and Nurturing an Integrated Vision for Education in RSE and HPC

https://www.universe-hpc.ac.uk

With thanks to Neil Chue Hong and Weronika Filinger, UNIVERSE-HPC Project for UNIVERSE-HPC slide content

Photo by Mikael Kristenson on Unsplash

Competencies and Skills

Identify the competencies and skills required by RSEs as they progress through their careers. Understand where these competencies are already being taught.

Learning Pathways

Define different curriculum, learning pathways, and delivery mechanisms for providing training in RSE competencies, including as taught programmes, online/self-paced learning and professional development.

Course Development + Delivery

Development of missing modules. Packaging of existing and new modules into different formats. Pilot delivery of courses.

Community Support + Contributions

Facilitate professional networking and peer support for RSEs. Develop a community of maintainers for open materials.

Equity, Diversity and Inclusion

Slide content: Neil Chue Hong

Future work and next steps

- **Understand more about what the community wants**
  - What skills do people want/need to learn?
  - What do *you* think is missing?

- **Build on great examples**
  - Have you seen particular examples of structured training content that you think are great?
  - What do we need to see more of?
  - Maintain awareness of available content

- **Test emerging pathways within the community**
  - Test workshops / approaches / structures
  - Get feedback!

Photo by Ali Shah Lakhani on Unsplash

Thank you

Questions?