Experiences from the Fortran Modernisation Workshop

Wadud Miah (Numerical Algorithms Group)

Filippo Spiga and Kyle Fernandes (Cambridge University)

Fatima Chami (Durham University)

http://www.nag.co.uk/content/fortran-modernization-workshop

Workshop Overview

- Computational scientists are the biggest Fortran users;
- This workshop focuses on teaching modern Fortran, tools and libraries that are commonly used within computational science;
- The workshop presents the concepts and theory followed by a practical hands on exercises;
- This workshop is only for computational scientists. It does not cover generic programming for other disciplines - there are better programming languages for the other domains;
- It provides a more holistic view of code development for computational science and the complexities of verification and validation.

Fortran Usage on Archer[1] HPC Service (1)

- Programming language usage is:

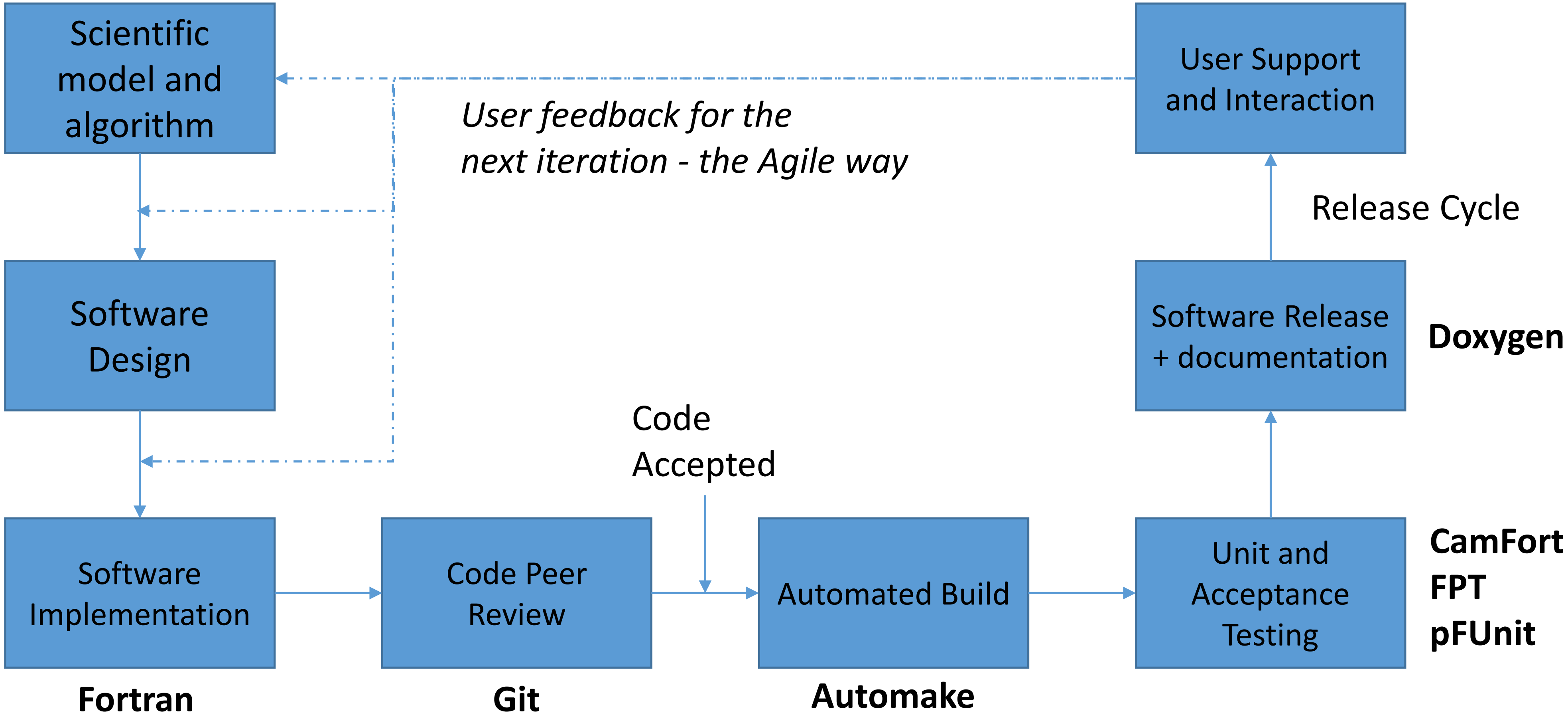
Programming Language	%Time	%Jobs
Fortran	**73.58**	**80.03**
C++	7.2	4.74
C	7.2	3.14
Python	0.7	0.99
Others	13.8	11.75

- Fortran usage statistics is ≈ 70% for the Top 500 supercomputers [2];
- *Fortran is the dominant programming language of computational science and engineering*.

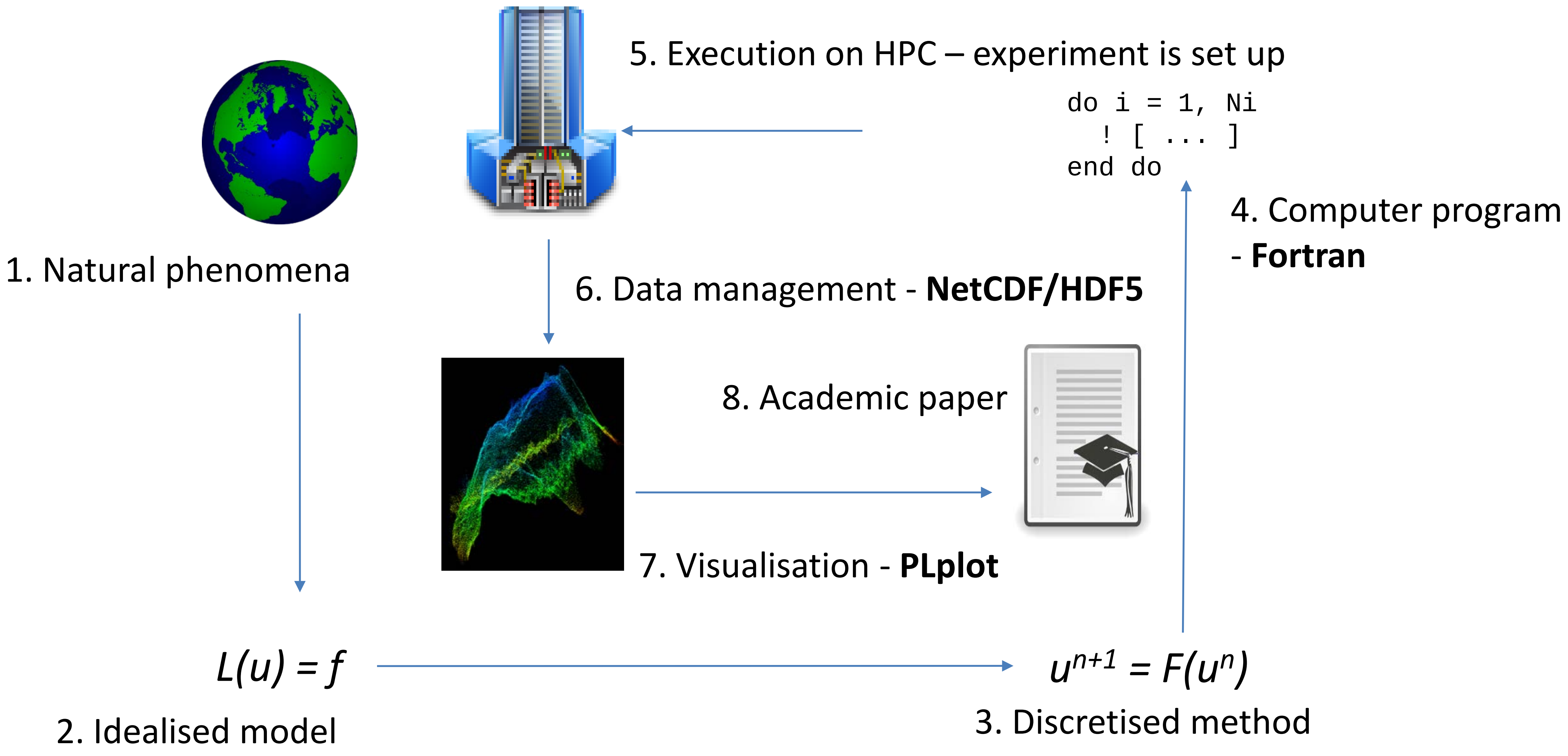
[1] The UK National Supercomputing Service, ARCHER. www.archer.ac.uk

[2] www.top500.org

Software Development Workflow


Scientific model and algorithm
User feedback for the next iteration - the Agile way
User Support and Interaction
Release Cycle
Software Design
Software Release + documentation
Doxygen
Code Accepted
Software Implementation
Code Peer Review
Automated Build
Unit and Acceptance Testing
CamFort
FPT
pFUnit
Fortran
Git
Automake

Computational Science Workflow



1. Natural phenomena

$L(u) = f$

2. Idealised model

$u^{n+1} = F(u^n)$

3. Discretised method

4. Computer program - **Fortran**

```
do i = 1, Ni
  ! [ ... ]
end do
```

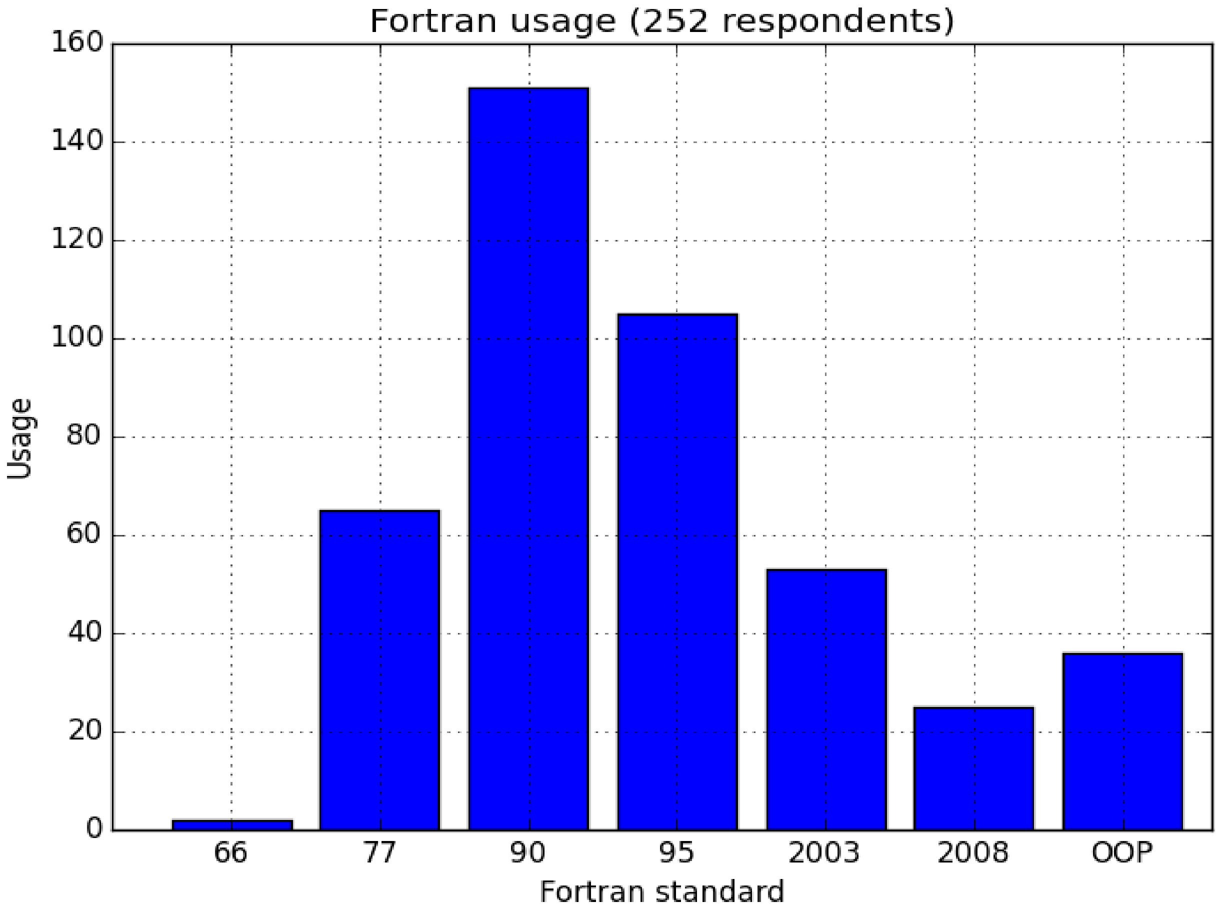
5. Execution on HPC – experiment is set up

6. Data management - **NetCDF/HDF5**

7. Visualisation - **PLplot**

8. Academic paper

Current Practice of Fortran Developers



Version Control

- The need for version control is obvious and allows software teams to better manage their code;
- Code features are developed in a separate branch and then merged with the master branch;
- Code releases are tagged so developers have a snapshot of a release;
- Distributed version control systems such as Git and Mercurial are becoming increasingly popular;
- A version control system as a valuable tool for Fortran programmers!

Automake

- Automake allows a consistent way to build any Fortran codes;
- A valuable tool for Fortran programmers particularly for medium to large projects;
- Fortran modules are placed in their individual files and Automake only builds modules that have changed and not all the modules;
- Another indispensable tool for Fortran programmers!

Fortran Code Verification

- The survey identified code verification as a weakness in Fortran developers;
- Unit testing frameworks are very rarely used so pFUnit [1] is taught at the workshop which can test serial, OpenMP and MPI codes;
- Dimensional analysis and stencil verification using CamFort [2] is also covered;
- FPT [3] for mixed precision bugs such as overflow and truncation;
- Forcheck [4] will be covered in the next revision of the workshop.

[1] http://pfunit.sourceforge.net/
[2] https://github.com/camfort/camfort
[3] http://www.simconglobal.com
[4] http://www.forcheck.nl/

Fortran Code Documentation With Doxygen

- Special comments are placed in the Fortran code which Doxygen [1] parses and runs in an automated manner;
- Equations in LaTeX are also parsed to create a HTML page;
- Doxygen documentation is mainly used to document the code so the team understand what each Fortran module does;
- It is also used to create user documentation on using APIs, e.g. libraries;
- We plan to replace Doxygen with Fortran Documenter [2].

[1] www.doxygen.org [2] https://github.com/cmacmackin/ford

Scientific Data Management with NetCDF

- Data management is an important topic and it is addressed in the workshop by promoting the usage of NetCDF file format;
- Data management is aided by the use of global attributes or metadata that are describe the dataset;
- Metadata can include: what model is being solved, experimental parameters, numerical solvers used, compiler used to build code and anything relevant to the experiment;
- The metadata should allow others to recreate the dataset and is used by data management system such as Eudat [1];
- Fortran 90 API exists for NetCDF.

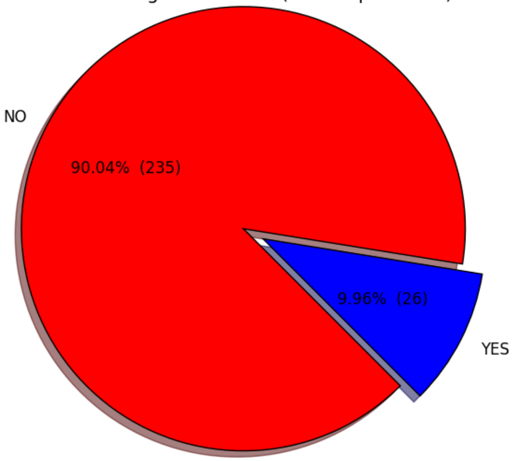
[1] www.eudat.eu

In-situ Visualisation With PLplot

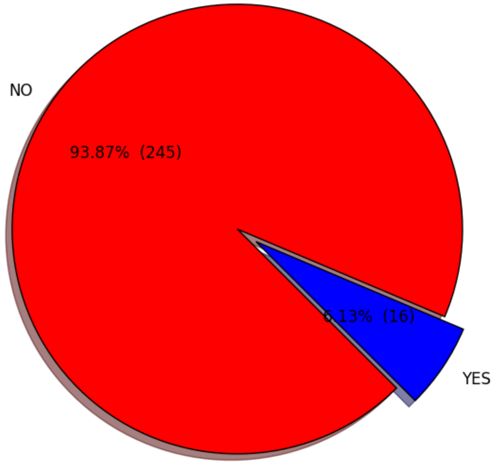
- In-situ visualisation allows the scientist to visualise the solution as the simulation is running;
- This allows the scientist to test their models and parameters, and if the solution diverges or produces non-physical results, the simulation can be terminated;
- This saves the scientists a lot of time and saves a lot of CPU cycles particularly early on in the simulation;
- In addition, visualisation is done quicker as the data is not saved on disk which also results in saving disk space;
- PLplot has Fortran 90 bindings.

Current Practice of Fortran Developers

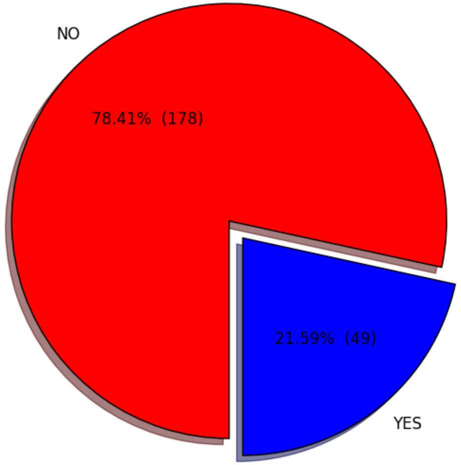
Unit testing frameworks (261 respondents)



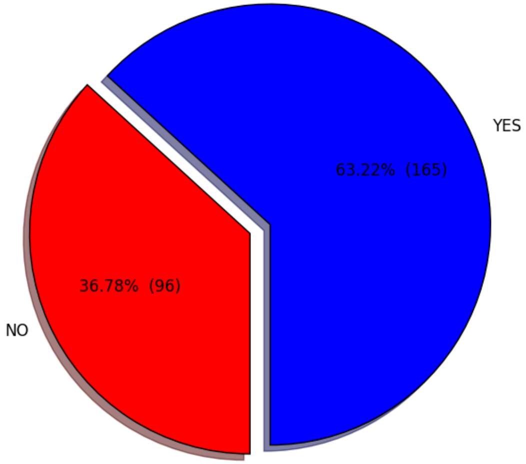
In-situ visualisation (261 respondents)



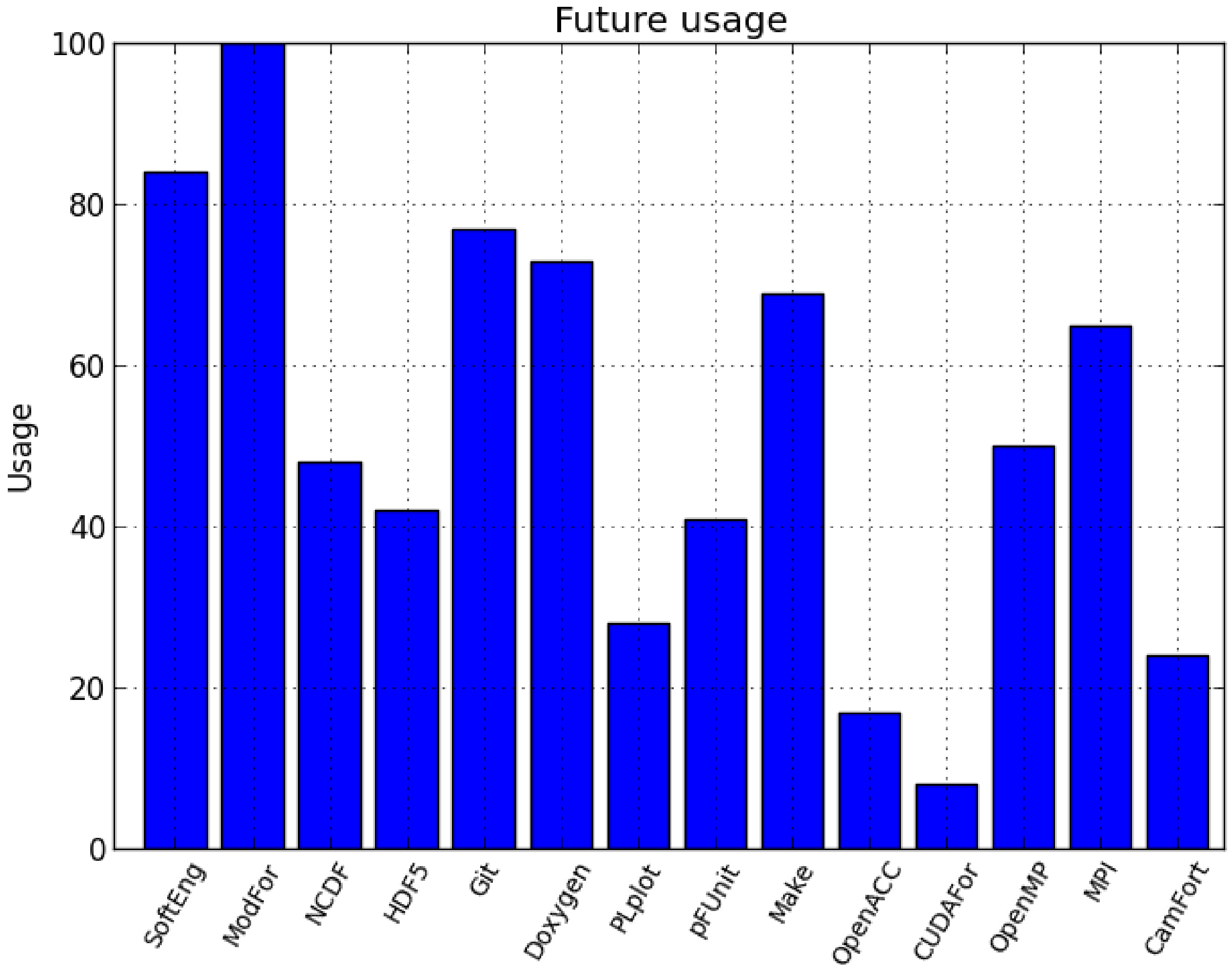
Software engineering (227 respondents)



Version control (261 respondents)



What Attendees Will Use Afterwards



Fortran Tools and Practice

- Fortran code verification has been recognised as a weakness which is being addressed by the workshop;
- More needs to be done to promote modern Fortran standards to increase portability of codes;
- Computational scientists need to be supported for the entire computational science workflow and not just for the code development workflow;
- **Optimised and efficient code should be written in a portable manner using the Fortran language standard to produce performance portable code** which is the emphasis of the workshop;
- The next workshop will be held at STFC Daresbury on 27-28/OCT. Visit the workshop web page for more details [1] and please spread the word;
- I can be reached at **wadud.miah@nag.co.uk**

[1] http://www.nag.co.uk/content/fortran-modernization-workshop