ExCALIBUR Project

Research Software Engineer Knowledge Integration Landscape Review

Authorship: ExCALIBUR Design & Development Working Groups
Version: 1.3.2 FINAL
Date: 16th June 2021

DOI: 10.5281/zenodo.4986062

Table of Contents

Foreword

The key role of software development and curation has long been recognised by the UK computational science community and the importance of the Research Software Engineer to UK science has grown markedly over the past decade. Recognition of the role now spans all of UKRI's Research Councils. The early work of the Software Sustainability Institute led to a meeting in 2012 where a small group discussed the lack of careers for software developers in academia. That meeting led to a nationwide campaign, a vibrant international community and a name for a new research role: the Research Software Engineer or RSE. In 2019 the Society of Research Software Engineers was launched as a charitable incorporation that replaced the UK Research Software Engineers Association which was established in 2013.

Today RSEs operate across the UKRI research base and within the National Laboratories and Public Sector Research Establishments. They take many roles within research projects and have a wide variety of skills. This report focusses on RSEs who work in the domain of High Performance Computing (HPC). We are at a critical moment in the development of large-scale computing – on the cusp of the Exascale era. The work of the ExCALIBUR programme is helping UK science prepare for this next generation of supercomputers. This report sets out the challenges faced in growing the number of RSEs in the UK with a specific focus on HPC. The importance is clear – without investment in software, realising the full value of investment in supercomputing hardware will be impossible. Investment in software means investment in people.

This report provides a comprehensive review of the skills required by RSEs in HPC and their future education and training needs. We wholeheartedly support the recommendations the report makes and look forward to a new era of scientific software supporting research across all of UKRI as we approach the Exascale era.

Professor Sir Richard Catlow FRS FLSW FRSC FInstP
Professor of Computational and Materials Chemistry
Department of Chemistry
University College London

Mr Neil Chue Hong FBCS
Chair, ExCALIBUR Programme Steering Committee
Director, Software Sustainability Institute

1 Executive Summary

- This report has been written by members of the ExCALIBUR Project community.
- It presents a Knowledge Integration Landscape Review focussed on the role of the Research Software Engineer as we prepare for the arrival of Exascale Supercomputers.
- The Exascale Computing ALgorithms & Infrastructures Benefiting UK Research (ExCALIBUR) project is a 5-year, £46m programme of activities funded through the UK Government's Strategic Priorities Fund.
- UKRI's Engineering and Physical Sciences Research Council and the Met Office are leading this programme in partnership with the UK Atomic Energy Authority, the Natural Environment Research Council, the Medical Research Council and the Science and Technologies Facilities Council.
- The report documents the concept of the Research Software Engineer (RSE) and their role in many of today's computational science projects.
- There are many different types of RSE – they support many different scales of digital research. This report focusses on those who work in the domain of High Performance Computing (HPC) with a specific focus on preparing for Exascale supercomputing.
- The report assesses the HPC skills required for the Exascale before summarising Exascale Software Programmes in the EU, USA and Japan.
- It assesses current HPC Training in the UK and Europe and documents the major training and skills requirements for RSEs working in HPC. It then identifies where major training and skills gaps exist today before focussing on the issue of long-term career development for RSEs.
- The report makes a number of recommendations:

1) UKRI should continue to invest in the development of Research Software Engineering in the UK.
2) As we grow the overall number of RSE staff in Universities, National Laboratories and other research organisations we should also grow the number of such staff with specific High Performance Computing skills.
3) An ExCALIBUR Training Programme for Research Software Engineers who want to focus on High Performance Computing should be established.
4) A long-term training and education strategy should be developed to both train the next generation of RSEs with HPC skills and also fill the training and skills gaps identified in this report.
5) A variety of different training models should be adopted – including postgraduate study, workshops, hackathons and bootcamps. Wherever possible training should be made available as online training as well as in face-to-face training opportunities.
6) Clear career paths for Research Software Engineers, and funding opportunities for software development allowing them to apply and develop skills, are crucially important to ensure that, once trained, the knowledge they have gained stays in the research sector and grows over time. The contribution of software engineering needs to be recognised in university recruitment and promotion procedures.
7) UKRI should ensure that it supports the message that Research Software Engineers are a highly valued resource at Universities, National Laboratories and other research organisations by providing clear guidance for inclusion of RSEs on grants.
8) Greater collaboration and transfer of skills by RSEs in both directions between the academic and industrial research sectors should be encouraged, particularly from industry to academia.
9) The UKRI Exascale Supercomputer Project's software programme should ensure that it encompasses a variety of different types of software activity and ensure they are contributing to developing the RSE community with HPC skills in the UK.

2 The Research Software Engineer concept

The Society of RSEs defines a Research Software Engineer as "...people in a variety of roles who understand and care about both good software and good research." Research Software Engineers may occupy dedicated roles in which research software engineering is their main task, or they may hold research or academic positions in which research software engineering is part of their portfolio of responsibilities. Many Research Software Engineers will not have the term in their job title. Many Research Software Engineers start off as researchers who spend time developing software to progress their research. Because they enjoy this work and have invested in developing specialist skills, they continue to focus on software and its use in research. Research Software Engineers will generally have or gain some research experience – the role is not that of a software engineer who supports research. Others start off from a more conventional software-development background and are drawn to research by the challenge of using their software development knowledge to further research. Research Software Engineers have a key role in enabling research.

The attraction of being a Research Software Engineer[1] is similar to that of being a researcher: the role attracts people who want to know how the world works. The Research Software Engineer works with researchers to gain an understanding of the problems they face and then develops, maintains and extends software to provide the answers. But their remit extends well beyond the project, particularly when one of the research outputs is software designed to be run by many users and requires support beyond the project lifecycle.

The 'traditional' research software engineering role arose from the realisation that:

a) Too many researchers spend too much of their time developing their own code to work within their local environment or attempting to modify code so that it will work on a new environment;
b) Some researchers become excited by the software development process itself to support collaborative research; and
c) While most researchers are now competent software developers, not all researchers are good software engineers, which can lead to issues in portability and long-term code maintenance.

This final point is an important one. In many cases, scientific researchers may not be the best software engineers and while the code they have developed may be suitable for their own purposes, a lack of transparency makes results virtually unreproducible and their research subject to potentially unjustified criticism. At the other end of the scale is the sheer complexity of mission critical codes nowadays – an example being the Met Office's operational code which has at least 100 active developers with the trunk being updated on a 24/7 basis by partners around the world. This example is also important because it makes the point that not all RSEs work in academic settings.

Furthermore, those researchers that have focussed on software engineering have often found it difficult to publish their results. Leading to a low publication rate. Given a high scientific publication rate is often a pre-cursor to obtaining fellowship funding or lectureship, such researchers have often found it difficult to establish a long-term stable career path through no fault of their own.

It is important to note that Research Software Engineers enable research using software across almost all of the UKRI research base. Furthermore, the software they develop and often publish as Open Source, runs at a variety of scales – from lab-based instruments to a researchers' laptops to the largest of the UK's supercomputers.

[1] https://www.software.ac.uk/blog/2013-08-23-ten-reasons-be-research-software-engineer

While there is no 'typical' research software engineer, their role is clear. They are integral members of any research team which aims at generating any digital output, whether that is through development, modelling or experimental analysis. As with any key member of a research project, the RSE should ideally be involved from the proposal stage, through design and to project close, particularly where projects have a significant software development aspect. This is one of the strengths of the EPSRC eCSE programmes – many RSEs have lead bids to these programmes – often the first proposal they have written. Engaging RSEs in proposal development allows for such research proposals and outcomes to achieve the world class science to which we all strive. Their experience in the co-design process ensures that the outcomes are valid, but also achieved in the shortest possible time frame.

Framed within the context of modern research practices, the role takes on added significance with the importance of reproducible science and automated tools for verification. Indeed, the codes developed and used within the research process are as significant as the final result. The RSE should not be thought of just a software engineer who you can call upon to improve your code prior to publication; it is much more effective to engage the RSE early rather than trying to redesign any code or workflows to ensure they are reproducible. Cockbill, May and Mitchell[2] have demonstrated the cost savings of this co-design process. This engagement helps provide a more rapid 'time to science' as well as better quality science. However, with the current structure of many organisations, it is appreciated that RSEs typically only become available once funding is secure. Maintaining high quality RSEs is imperative to ensure investment in software is maximised for the long term to allow them to support such a co-design process.

The importance of RSEs has been demonstrated by both the growing number in the UK and its increasing international adoption with active associations in many countries around Europe and in the USA and South Pacific region.

A subset of Research Software Engineers work in the area of High Performance Computing (HPC), which is often referred to as Supercomputing. This document specifically focusses on the specific skills of such team members and considers how to develop more such people to support the growth required by the HPC community as we enter the Exascale era. What attracts such people to work in the area of HPC is a:

- Desire to contribute to world-leading science.
- Desire to make best use of their combination of scientific and technical skills.
- Desire to make a difference to major projects.
- Desire to be at the cutting edge of both science and technology, e.g. working with some of the largest computers in the world.

The need for Research Software Engineers focussed on HPC is wide ranging; from cosmology to quantum chromodynamics, HPC is used at all scales of scientific discovery. Indeed, with the field of HPC undergoing an explosion in technologies and becoming a cornerstone of research in many diverse fields, the need for RSEs with HPC experience has never been greater. As we move towards the Exascale there will be new programming paradigms, novel and heterogeneous architectures. While initial Exascale systems are likely to be a combination of CPU and GPU nodes, research into future architectures will explore a rich and diverse landscape of hardware. Taking advantage of these will be role of the HPC RSE.

Running poorly optimised code on future HPC systems will be very costly; even if we look at the current generation of pre-Exascale machines SUMMIT (200PFLOPS) has a power requirement of 10.1MW, while Fugaku (537PFLOPS) consumes nearly 29.9MW of power, and that is not even considering the capital costs

[2] Stuart A. Cockbill, Andrew May, Val Mitchell, "The Assessment of Meaningful Outcomes from Co-design: A Case Study from the Energy Sector", She Ji: The Journal of Design, Economics, and Innovation, Volume 5, Issue 3, 2019, Pages 188-208, ISSN 2405-8726, https://doi.org/10.1016/j.sheji.2019.07.004

to procure and deploy the hardware or the resource cost in terms of administration and maintenance. Ensuring codes are designed to run efficiently on future Exascale systems – indeed all HPC systems – will represent a significant long term cost saving, meaning investment in RSEs will provide long term cost savings and allow more useful science to be performed.

Exascale systems are being developed in a number of countries and regions worldwide. The demand for these systems from research communities will probably outstrip the available resource. Employing, training and offering a good career path to RSEs specialising in HPC will help mitigate this issue, allow them to bring their specialist knowledge, as well as knowledge of current software development best practices, to ensure co-designed software is scalable, robust, verifiable, portable and maintainable for the long term, offering a significant return on investment. There is some evidence that RSEs specialising in HPC are under-represented overall today and that there is a need for intervention to address this.

3 HPC skills required for the Exascale

This section discusses the computational science and HPC skills required to deliver leading-edge science on the next generation of HPC systems at the Exascale. While there are many core RSE skillsets that span a wide number of research domains there are also RSE skillsets specific to HPC. In order to grow the number of RSEs able to work in HPC and on Exascale projects we need to understand what those skills are. By understanding these skills we can also outline, as we do later in this document, the training and education required to develop the next generation of RSEs working in the HPC domain.

It is also important to note that the HPC skills required by RSEs in this domain have changed over time. There is much greater cross-over nowadays between HPC and Data Science skills – particularly due to the growth of the use of GPUs in HPC systems which can be used not only for accelerating modelling and simulation codes but also artificial intelligence applications. As Exascale hardware will support both the largest modelling and simulation and AI challenges, the skills required by RSEs working in this domain are markedly broadening.

3.1 Traditional HPC skills

Typical skills an RSE may require at the Exascale may include:

- Need to identify "best practice" software engineering for the Exascale across the likely technology platforms (predominately CPUs, or CPUs + GPUs, perhaps with other novel accelerators)
- Exascale debugging & profiling
- Common numerical algorithms and their Exascale challenges
- Developing novel algorithms (e.g. for large eigenvalue methods)
- Advanced parallel paradigms (e.g. PGAS, Parallel STL)
- Novel libraries/frameworks (e.g. SYCL, kokkos)
- Mixed language HPC programming, and heterogeneous computing (e.g. OpenACC, OpenMP5).
- Job management and execution including data management and I/O for the Exascale

Computational science and HPC has traditionally focussed on the data production side (e.g. the results of simulation calculations), while data science has usually been a separate discipline (with the exception of interpretation of simulation results or computational steering and in-situ visualisation). Though data production (prediction) and analysis historically have been relatively separate, these two fields are converging. There are two main drivers behind this: on the one hand, the rapid evolution of machine learning as a data analysis technique has not only pushed the underlying discipline, it also has become the main driver behind hardware evolution. Classic HPC thus has to master these trends. Furthermore, researchers are not exclusively interested in the forward problem (prediction) but want to run sensitivity analyses or solve inverse problems. This requires the coupling of data analysis with data production.

RSEs come from a wide variety of backgrounds and although many may focus on a particular application domain many move between domains throughout their career. At the same time, the majority do not come through an applied mathematics or computer science[3] degree. In the UK, for many years there has been a strong distinction between the Computer Science and Computational Science communities. As we approach the Exascale there is a need to rethink the underlying mathematics of computational science. A key area for training and development should be increasing the number of RSEs with a mathematics or computer science background who can also understand the science behind particular applications or collaborate effectively with those who do. However, as discussed later in this report, there is a strong

[3] 27% of UK RSEs have a computer science undergraduate degree (data from Software Sustainability Institute).

argument that publications are not the correct metric against which the success of a career in RSE should be evaluated.

There are many different types of RSEs from a wide range of backgrounds. The following points are specific to RSEs working in the HPC domain:

1. Fundamental principles of numerical analysis. While not all RSEs have the skills to make a contribution on the modelling side or create new numerical schemes, they have an understanding of the fundamental numerical properties such as how well-posed a challenge is or how numerically stable. This knowledge is essential to understand and interpret numerical properties of simulation outcomes – a skill many domain experts lack.
2. Understand principles and theory behind frequently used numerical building blocks and their interaction (e.g 'The Outer Loop'). The most prominent representative among these blocks are linear equation system solvers. Today, such building blocks are often used as black box components. While using these components is reasonable to create fast code quickly, using them really efficiently requires a deeper understanding of their strengths and weaknesses, usage, and design decisions behind the code. This distinguishes an engineer from someone who simply plugs building blocks together – a pattern that is found often today in the machine learning community with TensorFlow etc.
3. Compiled languages and low-level APIs. There's an omnipresent trend for high-level languages such as Python and Julia to dominate the teaching landscape. As a result, many software projects lack a pool of developers who have a sound understanding of the two most widely used languages - C/C++ and Fortran. This is a gap RSEs can fill and we urgently need people who can do so, as we otherwise run risk that our established software ecosystem written in C and Fortran cannot be used and be improved anymore. The skill set here also comprises deep knowledge of MPI and a many-core programming approach such as OpenMP or C++ threads.
4. RSEs should be able to identify and apply the appropriate techniques and collaboration tools for a given project (e.g. Agile or Waterfall). Version control and continuous integration are a must today. Further to that, a Research Software Engineer has to be able introduce different techniques into teams successfully (e.g. test-first strategies, refactoring, pair programming etc). RSEs should understand the importance of good Quality Assurance and code management.
5. Hardware and machine architectures. Exascale software will require software that exploits all levels of efficiency and all machine capabilities. To be able to write such software, developers need a sound understanding of how a machine is constructed: They have to know the basics behind network topologies and routing, NUMA effects, cache levels, memory pinning, and so forth.
6. Systematic testing and debugging. Exascale software will be highly parallel, and searching for errors in massively parallel software is tedious and cumbersome. It is important that developers are emancipate from the printf-debugging approach. Proper software engineers should be trained in systematic debugging and correctness checks supported by tools.
7. Systematic performance analysis. To make software fast, engineers need the skills to systematically break down runtime cost into steps, map them onto compute units, and explain the behaviour taking into account the algorithmic paradigms plus the hardware paradigms. A complex ecosystem for performance analysis does exist, and it will be up to RSEs to allow research code developer groups to benefit from these tools. Strong optimisation skills coupled to an understanding of the abstraction to aid portability will be required.
8. Communication skills are also important. A successful RSE has to be able to engage in effective two-way communication with scientist colleagues; they need to be able to elicit requirements from scientists and translate these into development plans; they also need to be able to convince

scientists of the benefits of doing things "properly". This is particularly important at the Exascale as many applications will need significant and disruptive changes.

9. Since large HPC systems are not typically available during development, there is often a need to focus on mini-apps to demonstrate scalability, and carry the work of the mini app into the production code, which is a significant additional skill needed within this environment.

3.2 Artificial Intelligence and Machine Learning skills

In parallel to the HPC initiatives, advances in Artificial Intelligence (AI) and Machine Learning (ML) have also enabled the scientific community to advance the frontiers of knowledge. Exascale systems will support both traditional modelling and simulation applications and also large-scale AI and ML applications.

On the AI front, one key area of activity will be the analysis of complex and large-scale experimental datasets. With the scientific community generating huge amounts of data, often ranging into petabytes, from observatories to large-scale experimental facilities, the skillsets for enabling AI for Science[4] are new. These skillsets also begin to deviate from what is demanded by industries, requiring novel AI software and capabilities. In many cases, techniques used in some of the scientific datasets need the scale of HPC systems, which can have unique architectural features demanding software attention and investment, such as large-scale I/O subsystems and heterogeneous compute elements. The skillsets at Exascale need to ensure that the RSE community can develop relevant, next-generation ready AI software stacks.

Although it is difficult to provide an exhaustive list of skills needed for AI at Exascale, here is a list of key skills that the AI/HPC community would need to ensure we are able to embrace AI at the Exascale:

1. **Ability to understand and develop surrogate models**
   Surrogate models provide an avenue for developing AI-based, rapid computational models which can, not only be used as an approximate approach, but also to speed up some of the computationally intensive HPC stages of a workflow. However, developing such models need a detailed knowledge of the relevant discipline and the ability to understand the HPC codes.
2. **Containerisation**
   Ability to package AI/HPC solutions that are truly portable, secure and scalable is an essential aspect of developing and deploying AI/ML solutions, particularly when considering scientists as end-users. To this end, ability to develop or engineer custom tools and code generators for containerisation is an essential skill. This is a bigger issue than simply AI but has come to prominence thanks to NVIDIA's NGC catalogue[5].
3. **Scalable AI algorithms**
   From initial pre-processing to learning to inference, AI/ML plays a pivotal role. However, it is essential to ensure that relevant algorithms remain scalable to the volumes of data being generated or observed.
4. **Algorithms to quantify uncertainty in surrogate models**
   Although AI-based surrogate models are increasingly becoming attractive, the downside is that their uncertainty to variations in inputs or initial conditions or even to the model hyper-parameters can serve as important criteria for qualifying surrogate models for their suitability or applicability.
5. **HPC/AI hybrid application development**
   Although AI and HPC have offered advances, it will be inevitable to develop hybrid applications, where HPC and AI components exploit each other for collective advances in science and engineering.

[4] More information on AI for Science is given in Annex A

[5] https://ngc.nvidia.com/

6. **Large-scale complex generative models**
   AI/ML-based generative model are attractive way to generate synthetic datasets that either approximate HPC simulations or can feed into complex HPC simulations. However, developing complex generative modes will be a key requirement in future Exascale systems. There is a clear opportunity here for RSEs to liaise between domain scientists and data scientists.
7. **Debugging and profiling AI models**
   Although AI/ML tools are attractive, debugging them, particularly with complex data and model settings will be an essential skill in Exascale systems.
8. **Extending, customising and enhancing open-source ML frameworks**
   Although open-source tools, such as, TensorFlow and PyTorch, are versatile and can often be used as they are, in some of the cases, for example when developing new ML algorithms, it becomes as a necessity to modify/extend/customise these frameworks. Such endeavours demand a great deal of understanding of these frameworks, and good software engineering skills.
9. **Data lifecycle management**
   The notion of Exascale is not limited to compute, particularly in the setting of large-scale experimental facilities and science observatories. One of the skills that is essential for developing robust ML algorithms or benchmarks or ML techniques is ensuring that the underlying data is maintained, managed well and securely handled. This is particularly important with the growing use of health data where the RSE may also need to consider ethical and information governance issues.

4 Summary of International Exascale Software Programmes

This section presents a short summary of the Exascale software programmes in the USA, Japan and Europe to give an idea of the scale of the investment in people and skills internationally. Many of these programmes date back over the past decade.

4.1 EU: The EuroHPC initiative

The EuroHPC Joint Undertaking (https://eurohpc-ju.europa.eu/) is a joint initiative between the EU, European countries and private partners to develop a World Class Supercomputing Ecosystem in Europe. It is a legal and funding entity, created in 2018 and located in Luxembourg. It allows the EU and EuroHPC participating countries to coordinate their efforts and pool their resources with the objective of deploying in Europe world-class Exascale supercomputers and developing innovative supercomputing technologies and applications. EuroHPC seeks to provide computing solutions, improving cooperation in advanced scientific research, boosting industrial competitiveness, and ensuring European technological and digital autonomy.

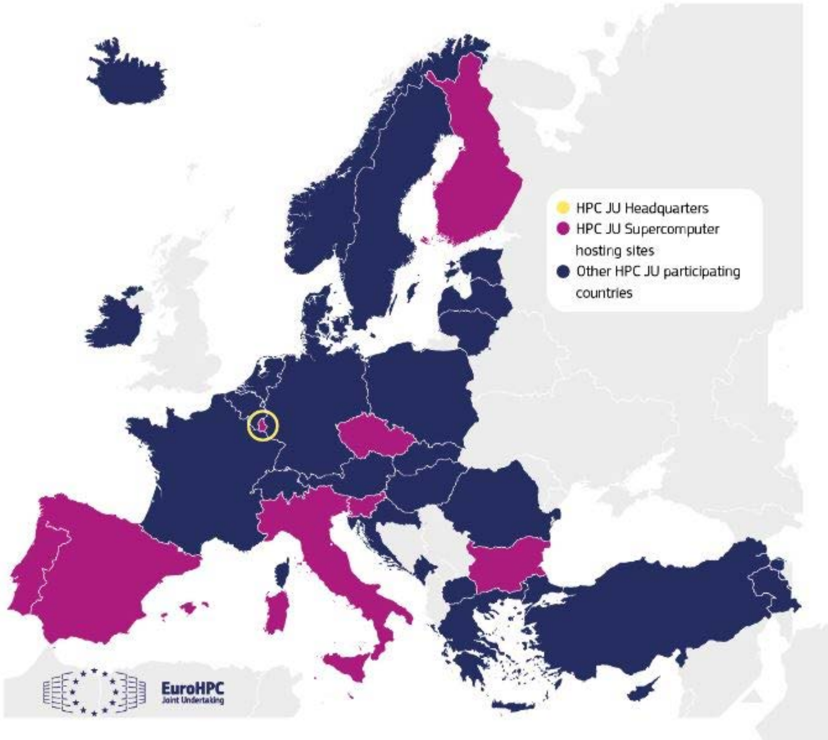
Currently, it is supporting the following activities:

- **Developing a world-class supercomputing infrastructure**: procuring and deploying by 2021 in the EU three pre-Exascale supercomputers and five petascale supercomputers. These new machines will be located across the European Union and will be available to Europe's private and public users, scientific and industrial users everywhere in Europe. The three pre-Exascale supercomputers will be located at Barcelona Supercomputing Centre (Spain), CSC – IT Center for Science (Finland), and CINECA (Italy). The five petascale supercomputers will be located in Sofiatech Park (Bulgaria), IT4Innovations National Supercomputing Center (Czech Republic), Luxprovide (Luxembourg), Minho Advanced Computing Centre (Portugal) and IZUM (Slovenia). The pre-Exascale supercomputers at CSC and CINECA have recently been announced and will provide in excess of 0.75 exaflop/s peak performance from early 2022.
- **Supporting research and innovation activities**: developing and maintaining an innovative European supercomputing ecosystem, stimulating a technology supply industry (from low-power processors to software and middleware, and their integration into supercomputing systems), and making supercomputing resources in many application areas available to many public and private users, including small and medium-sized enterprises. Through its research and innovation agenda, the EuroHPC JU is also strengthening the European knowledge base in HPC technologies and bridging the digital skills gap, notably through the creation of a network of national HPC Competence Centres. The Competence Centres will act locally to ease access to European HPC opportunities in different industrial sectors, delivering tailored solutions for a wide variety of users.

The EuroHPC Joint Undertaking is composed of public and private members:

- Public members: the European Union (represented by the Commission). Member States and Associated Countries that have chosen to become members of the Joint Undertaking: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Montenegro, the Netherlands, North Macedonia, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, and Turkey. The UK has declined to join EuroHPC.

- Private members: representatives from the two participating private partners, the European Technology Platform for High Performance Computing (ETP4HPC) and the Big Data Value (BDVA) associations.



The EuroHPC Joint Undertaking is jointly funded by its public members with a current budget of around €1.1 billion for the period 2019-2020. Most of the funding comes from the current EU long-term budget, the Multiannual Financial Framework (MFF) with a contribution of €536 million. This sum is expected to be matched by a similar amount from the participating countries. Private members will also provide additional contributions to the value of over €420 million, through participation in the Joint Undertaking's activities. The Joint Undertaking provides financial support in the form of procurement or research and innovation grants to participants following open and competitive calls.

HPC is one of the key digital topics where the EU's investment should significantly increase in the next MFF (2021-2027). For this next financial period, the Digital European Programme (DEP), Horizon Europe (H-E) and Connecting Europe Facility-2 (CEF-2) are the main EU funding programmes that will be used to finance the EuroHPC JU. On 18 September 2020, the European Commission proposed a new Council regulation allowing the EuroHPC JU to continue the development of HPC in Europe for the next decade – with a total funding envelope of €8bn. The new regulation aims at replacing the Council Regulation (EU) 2018/1488 establishing the EuroHPC JU. It sets out an ambitious mission to provide Europe with a world-leading hyper-connected supercomputing and quantum computing infrastructure, which will be easily and securely accessible from anywhere in Europe. The new regulation will also enable support to research and innovation activities for new supercomputing technologies, systems applications and products as well as the development of necessary skills to use the infrastructure and form the basis for a world-class HPC ecosystem in Europe.

The European Commission started funding Exascale research and technological development (RTD) projects in 2012 through Framework 7. This activity grew throughout the Horizon 2020 programme and consisted of series of different initiatives with over 150 projects related to HPC being funded during Horizon 2020 including:

- **PRACE** (ongoing) is the European shared HPC research infrastructure funded by Member States and Associated Countries. It has been supported by the European Commission through a series of implementation projects (PRACE1-IP to PRACE6-IP). It has its own Council which the UK is represented on. The partners are split into Hosting Members who provide access to Tier 0 systems and General Partners who fund software development teams at the Hosting Members. General Partners also contribute access to their systems through a DECI programme of resource sharing. The PRACE RI focusses on supporting scientific research projects based on a peer review process assessing the scientific value of the proposals (industry is able to use PRACE machine for research on the same conditions). As EuroHPC grows in importance, PRACE will become one of the organisations that supports its activities. The UK is currently the largest General Partner in PRACE having declined to become a Hosting Member in 2010.
- **RTD projects** (ongoing) have focused on the development and exploration of the new hardware and software technologies required to develop the first generation of Exascale supercomputers and the applications that will run on them. The first three projects were DEEP, MONT BLANC and CRESTA. The UK has been involved in a number of these projects and has led several of them with EPCC, Arm Ltd, the University of Manchester and STFC Hartree Centre being the main UK organisations involved.
- **Centres of Excellence** (CoE) (ongoing) in Computing Applications, projects whose task is to develop, maintain and enhance HPC applications in various domains such as material sciences, weather and climate, biomedicine, industrial applications as well as tackle other related issues such as performance and training.
- **EuroHPC Competence Centres** have been established across Europe since 2020 through the EuroCC project. The UK's involvement is led by EPCC in collaboration with the Hartree Centre. The network of centres across Europe is similar but not entirely the same as those involved in the PRACE network.
- **EXDCI / EXDCI-2** projects (2015-2018). These projects aimed to support the road-mapping, strategy-making and performance-monitoring activities of the EU ecosystem for high performance computing.
- **FORTISSIMO** (ongoing) in a series of projects, originally funded by the Factories of the Future programme and led by EPCC, over 90 companies have been introduced to HPC and HPDA for the first time. Fortissimo has now become part of EuroHPC through the FF4EuroHPC project with the leadership being transferred to HLRS Stuttgart.ETP4HPC (ongoing). the European HPC Technology platform, is an industry-led association and represents the European HPC technology value chain. ETP4HPC's objective is to build a globally competitive HPC technology provision industry in European. Its main deliverable is the European HPC Strategic Research Agenda (SRA), which defines the priorities for research and development in HPC technology (and applications). The European funding organisations (such as the European Commission or the EuroHPC Joint Undertaking) use the SRA to define research project contents.

The UK and European Commission have announced that the UK will be an Associate Country within the Horizon Europe programme. However, the UK will not associate to the Digital Europe or CEF-2 programmes. This means that although the UK can join some EU Exascale projects funded by Horizon Europe, we will not be able to join projects that are focused on hosting or operating Exascale systems. Furthermore, Digital Europe also has significant amounts of funding for advanced skills and expert development which the UK will be unable to directly access. Access to the Exascale systems funded by EuroHPC JU is also unlikely. The UK may need to leave the PRACE Research Infrastructure.

4.2 US: The Compelling Case for Exascale Computing

The DOE-led Exascale Computing Initiative (ECI) (https://www.exascaleproject.org/), a partnership between two DOE organizations, the Office of Science (SC) and the National Nuclear Security Administration (NNSA), was formed in 2016 to accelerate research, development, acquisition, and deployment projects to deliver Exascale computing capability to the DOE laboratories by the early to mid-2020s. Budget of about $200,000,000 every year.

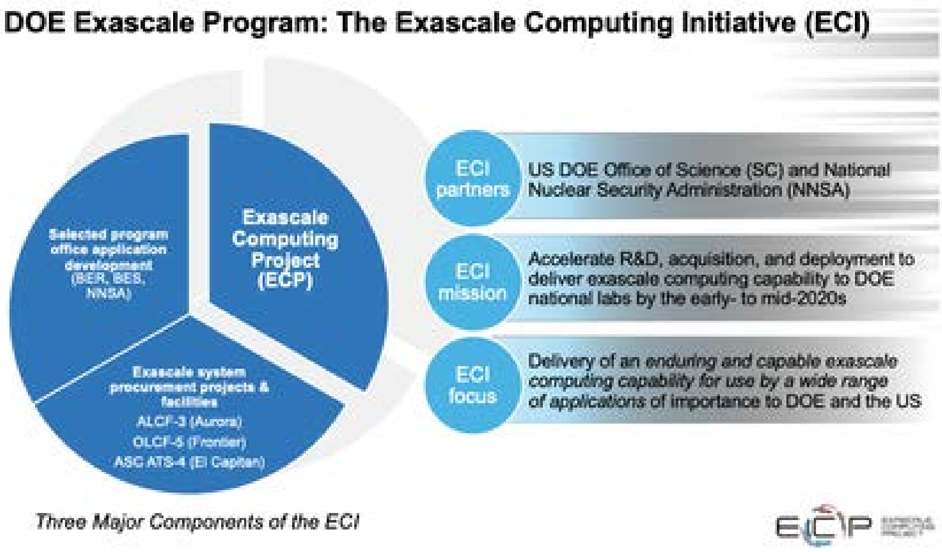
There are three major components of ECI:

- **Selected program office application development**: DOE Office of Science Biological and Environmental Research program and Basic Energy Sciences program and DOE National Nuclear Security Administration
- **Exascale system procurement projects and facilities**: ALCF-3 (Aurora), OLCF-5 (Frontier), ASC ATS-4 (El Capitan)
- **Exascale Computing Project**

The ECI consists of three main activities: (1) SC and NNSA computer facility site preparation investments, (2) computer vendor nonrecurring engineering activities needed for the delivery of Exascale systems within this time frame, and (3) the Exascale Computing Project (ECP), which was launched in 2016 and brings together research, development, and deployment activities as part of a capable Exascale computing ecosystem to ensure an enduring Exascale computing capability for the nation.

ECP, a 7-year project, is focused on delivering specific applications, software products, and outcomes on DOE computing facilities. Integration across these elements with specific hardware technologies for the manifestation of Exascale systems is fundamental to the success of ECP.

The outcome of ECP is the accelerated delivery of a capable Exascale computing ecosystem to provide breakthrough solutions that address our most critical challenges in scientific discovery, energy assurance, economic competitiveness, and national security. Thus, the aim is not simply a matter of ensuring more powerful computing systems.



ECP is designed to create more valuable and rapid insights from a wide variety of applications ("capable"), which requires a much higher level of inherent effectiveness in all methods, software tools, and ECP-enabled computing technologies to be acquired by the DOE laboratories ("ecosystem").

ECP's leadership team has staff from six of the largest DOE national laboratories, but overall, the project has participation from 15 of the 17 DOE laboratories. Today ECP is composed of approximately 1,000 researchers, scientists, participating US HPC systems companies, and project management experts in support of the project's key research focus areas: Application Development, Software Technology, and Hardware and Integration. ECP will also play a key role in helping to drive new training programs throughout the US HPC ecosystem to prepare application developers, researchers, and scientists to take full advantage of future-generation Exascale environments.

Exascale-capable applications are a foundational element of ECP and will be the delivery vehicle for solutions and insights to key national challenges and emerging technical areas such as machine learning and artificial intelligence. Problems heretofore intractable will be accessible with ECP applications.

Summary of activities:

- Twenty-four Application Development teams[6] have been actively engaged in targeted development and capability enablement for 3 years.
- The applications have been well-defined Exascale challenge problem targets with associated "science work rate" goals.
- Initial performance experiences on pre-Exascale systems (Summit, Sierra) are exceeding expectations. Software technologies play an essential enabling role as the underlying technology to application integration and effectiveness on computing systems. An expanded and vertically integrated software stack is being developed to include advanced mathematical libraries and frameworks, extreme-scale programming environments, tools, and I/O and visualization libraries.

Summary of activities:

- The Software Technology[7] research focus area has 33 projects.
- Seventy software technology products are being actively developed for next-generation architectures.
- Regular assessment of software stack products ensures line-of-sight to applications and HPC facilities.
- Plans for from-source and broad containerized delivery of products via Software Development Kits (SDKs) and the Extreme-scale Scientific Software Stack[8] are being executed.

ECP is focused on integration of applications, software, and hardware innovations to ensure a capable Exascale computing ecosystem. Working closely with the DOE HPC facilities, the project supports US HPC vendor research and development of innovative architectures for competitive Exascale system designs.

Summary of activities:

- The Hardware and Integration research focus area has 15 projects.
- The return on the PathForward vendor hardware R&D element has been evident in recent Exascale request-for-proposals responses.

[6] More information can be found at http://exascaleproject.org/research/#application
[7] More information can be found here http://exascaleproject.org/research/#software
[8] https://e4s-project.github.io/

- Plans for deployment and continuous integration of SDKs into DOE HPC facilities are being executed.

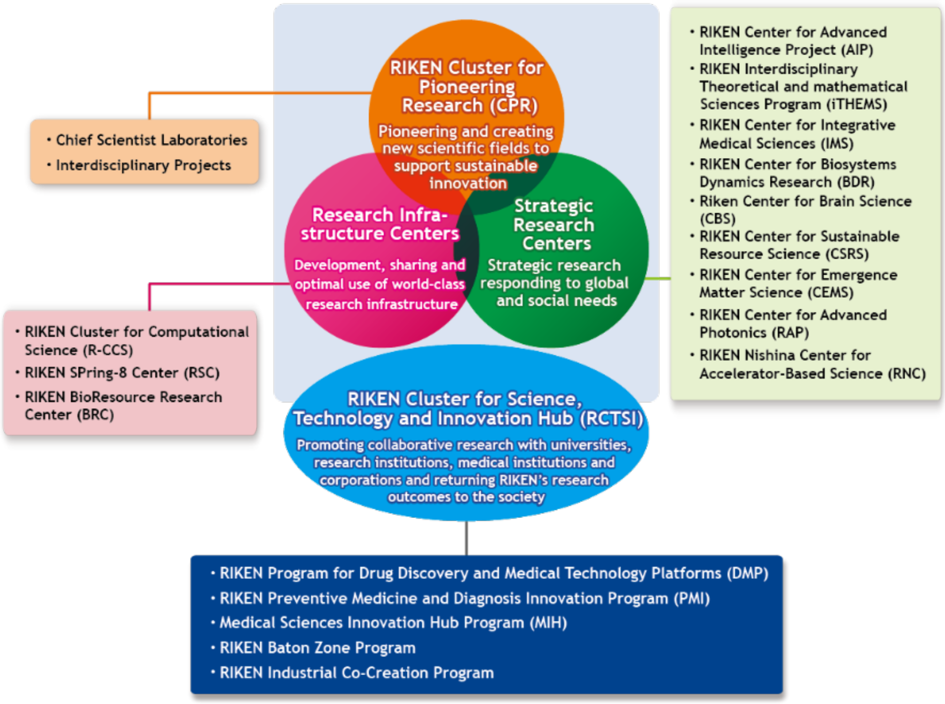
Prioritized performance engineering of applications targeting the first three Exascale systems is under way.

Related and previous initiatives in the US:

- **Advanced Scientific Computing Research** (ongoing): The Advanced Scientific Computing Research (ASCR) programme's mission is to advance applied mathematics and computer science; deliver the most sophisticated computational scientific applications in partnership with disciplinary science; advance computing and networking capabilities; and develop future generations of computing hardware and software tools for science and engineering in partnership with the research community, including U.S. industry. ASCR supports state-of-the-art capabilities that enable scientific discovery through computation. The Computer Science and Applied Mathematics activities in ASCR provide the foundation for increasing the capability of the national HPC ecosystem by focusing on long-term research to develop software, algorithms, methods, tools and workflows that anticipate future hardware challenges and opportunities as well as science application needs. ASCR's partnerships and coordination with the other Office of Science (SC) programs and with industry are essential to these efforts. At the same time, ASCR partners with disciplinary sciences to deliver some of the most advanced scientific computing applications in areas of strategic importance to SC and the Department of Energy (DOE). ASCR also supports world-class, open access HPC facilities and high performance networks for scientific research. Budget of about $900,000,000 every year.
- **Accelerated Strategic Computing Initiative** (ASCI; 1996-2006): launched by the US Department of Energy in 1996, ASCI was focused on rewriting and modernising simulation software to certify the US nuclear weapons stockpile without experimental testing. Over $6bn were spent to improve the predictive capability at the US National Labs at Los Alamos (LANL), Lawrence Livermore (LLNL) and Sandia (SNL). This improvement was to be delivered by:
    - i. Substantially improved physics, resolution and materials data
    - ii. New massively parallel architectures (for the time; 10-100 TFLOPs)
    - iii. Developing applications and analysis tools, massively parallel platforms, operating and networking systems
    - iv. Developing and extending physics models

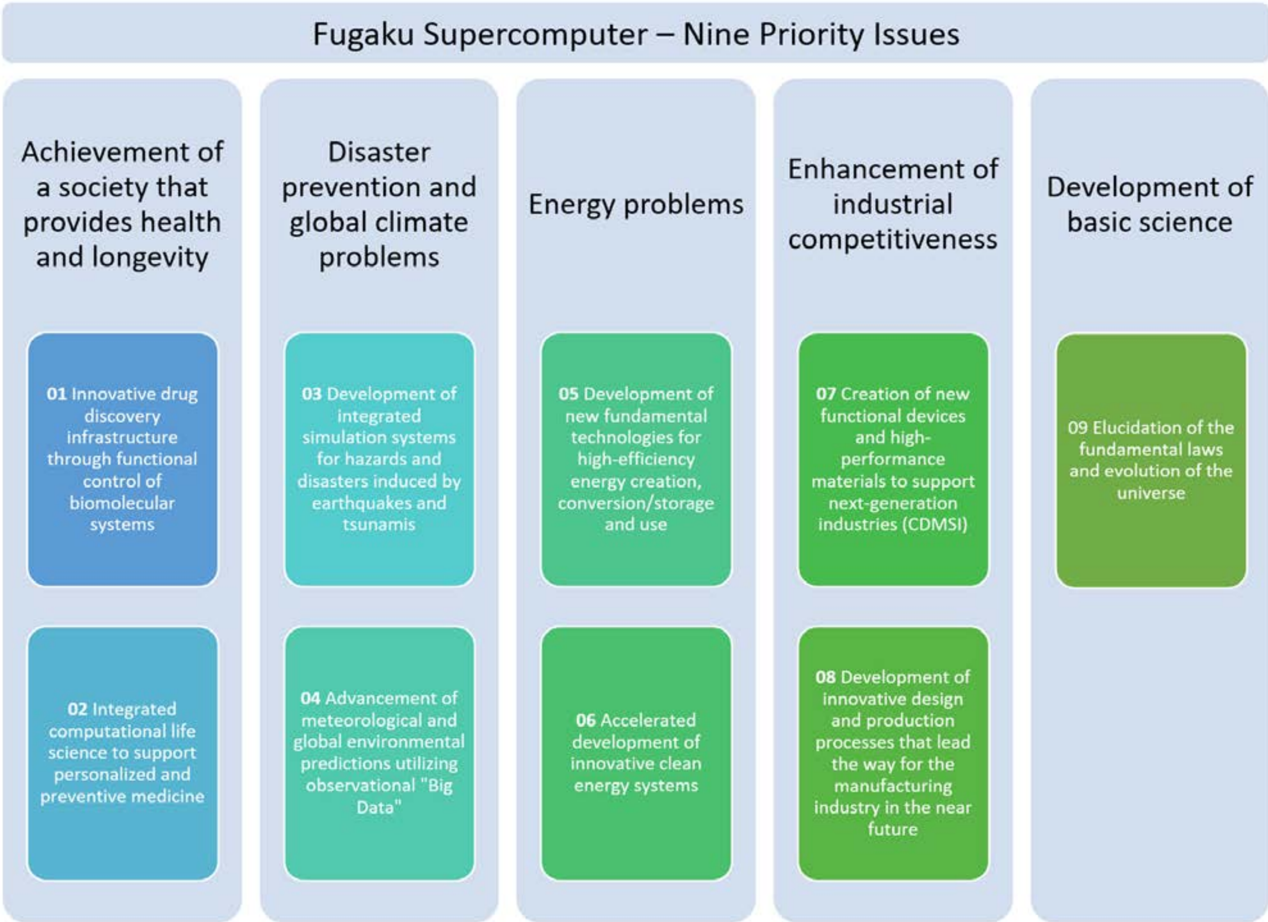
4.3 Japan: Fugaku Supercomputer Development Project

RIKEN (https://www.riken.jp/en/about/) is Japan's largest comprehensive research institution renowned for high-quality research in a diverse range of scientific disciplines. The Japanese Exascale programme is led by the RIKEN Center for Computational Science (https://www.r-ccs.riken.jp/en/). RIKEN was founded in 1917 as a private research foundation in Tokyo and has grown rapidly in size and scope. Today RIKEN encompasses a network of world-class research centers across Japan, with main campuses in Wako, Tsukuba, Yokohama, Kobe and Harima offering state-of-the-art facilities that rank among the best in the world. R-CCS and the Fugaku supercomputer is based in Kobe. This high-quality, high-performance research environment, combined with a uniquely bottom-up approach to scientific innovation, has enabled RIKEN to foster an environment in which researchers are able to thrive. RIKEN is also an international institute, with more than 600 non-Japanese research personnel from around the world.



The supercomputer Fugaku development plan initiated by Japan's Ministry of Education, Culture, Sports, Science and Technology in 2014, has set the goal to develop: (1) the next generation flagship supercomputer of Japan (the successor to the K computer); and (2) a wide range of applications that will address social and scientific issues of high priority. The six-year budget for the system and related technology development is about $1bn. Fugaku, the fastest supercomputer in 2020, which is currently installed at the RIKEN Center for Computational Science (R-CCS) in Kobe, Japan, is being developed under a national plan to design Japan's next generation flagship supercomputer and to carry out a wide range of applications that will address high-priority social and scientific issues. It will be put to use in applications aimed at achieving Japan's Society 5.0 plan, by running applications in areas such as drug discovery; personalized and preventive medicine; simulations of natural disasters; weather and climate forecasting; energy creation, storage, and use; development of clean energy; new material development; new design and production processes; and—as a purely scientific endeavor—elucidation of the fundamental laws and evolution of the universe. In addition, Fugaku is currently being used on an experimental basis for research on COVID-19, including on diagnostics, therapeutics, and simulations of the spread of the virus. The new supercomputer is scheduled to begin full operation in fiscal 2021 (which starts in April 2021) although it has been fully operational since mid-2020 (being completed around 6 months ahead of schedule).

The Japanese Exascale programme was designed over a 10-year period in a process that focused not just on the hardware development required but also on a set of nine "priority issues" that the use of the supercomputer will focus on. These nine areas of research were agreed on through collaborative process that spanned the Japanese academic and industrial research communities.


Fugaku Supercomputer – Nine Priority Issues
Achievement of a society that provides health and longevity
Disaster prevention and global climate problems
Energy problems
Enhancement of industrial competitiveness
Development of basic science
01 Innovative drug discovery infrastructure through functional control of biomolecular systems
03 Development of integrated simulation systems for hazards and disasters induced by earthquakes and tsunamis
05 Development of new fundamental technologies for high-efficiency energy creation, conversion/storage and use
07 Creation of new functional devices and high-performance materials to support next-generation industries (CDMSI)
09 Elucidation of the fundamental laws and evolution of the universe
02 Integrated computational life science to support personalized and preventive medicine
04 Advancement of meteorological and global environmental predictions utilizing observational "Big Data"
06 Accelerated development of innovative clean energy systems
08 Development of innovative design and production processes that lead the way for the manufacturing industry in the near future

5 Current HPC training provision

Significant HPC training provision has existed in the UK (and Europe through PRACE) for a number of years. This section summarises current provision at undergraduate, postgraduate and PDRA (and onwards) levels. Following this summary of current provision, a set of training gaps and other issues – such as a lack of UK MSc and PhD students – are identified.

5.1 Current UK HPC Training

Current training available in the UK in alphabetical order is listed below. Please note that this may not be a comprehensive list.

Bristol

- Advanced Computing Research Centre (ACRC) training https://www.bristol.ac.uk/acrc/acrc-training/. The ACRC has helped establish Bristol University as a world-class centre for research and teaching in advanced computing systems. It delivers hundreds of millions of core hours to researchers across multiple supercomputers and manage Petabytes of research data.

Cambridge

- MPhil in Scientific Computing https://www.postgraduate.study.cam.ac.uk/courses/directory/pcphmpscm/study. The core lectures are on topics of high-performance scientific computing, numerical analysis and advanced numerical methods and techniques
- High Performance Computing: An Introduction https://training.cam.ac.uk/course/ucs-hpc-intro. The course aims to give an introductory overview of HPC in general, and of the facilities of the High Performance Computing Service (HPCS) in particular.
- High Performance Computing Summer School https://www.gianna.phy.cam.ac.uk/news/high-performance-computing-summer-school. The University of Cambridge under the auspices of the Gianna Angelopoulos Programme for Science Technology and Innovation is organising a two-week summer school in High Performance Computing. Applications are invited for scholarships to attend the summer school which will be hosted by the Centre for Scientific Computing, University of Cambridge, 6th-17th September 2021. (Location to be confirmed; either held remotely or in Cambridge, UK). The overall aim of this two-week course is to provide course attendees with a strong background in elements of HPC techniques suitable for general science and technology projects.

Cardiff

- Big Data and High-Performance Computing Summer School https://www.cardiff.ac.uk/study/international/summer-schools/summer-school-programmes/big-data-and-high-performance-computing. This summer school was cancelled due to the Covid-19 pandemic. In this Big Data and High-Performance Computing summer school, specialists intended to teach attendees how to apply methods drawn from data science and analytics to help businesses and government organisations make better decisions. In addition, academic experts intended to improve attendees' knowledge of computer programming and provide practical applications to show how Big Data and HPC can solve real-world challenges including gravitational wave research and astronomical satellites.

Cranfield

- MSc in Computational and Software Techniques in Engineering
  https://www.cranfield.ac.uk/courses/taught/computational-and-software-techniques-in-engineering. Engineering software development is one of the key areas in the European information technology sector. It is a fast-moving subject of crucial importance to industry and forms the basis for a wide and ever-growing variety of applications. This course with its blend of skills-based and subject-specific material, has the fundamental objective of equipping attendees with the generic hands-on skills and up-to-date knowledge adaptable to the wide variety of applications that this field addresses.

Durham

- MSc in Scientific Computing and Data Analysis
  https://miscada.phyip3.dur.ac.uk . Advances in many fields from science, engineering and economy will be driven by those skilled in handling large parallel machines and data sets with extreme volume or velocity. The Durham MSc in Scientific Computing and Data Analysis (MISCADA) trains students in these areas, equips them with essential professional, entrepreneurial and collaborative skills, and gives them the opportunity to apply all acquired knowledge to challenging, state-of-the-art problems from a computational specialisation area. The programme targets an audience with excellent technical skills (in particular mathematics and programming) and makes the students understand **how modern scientific computing and data analysis tools work**. Students in this programme have to specialise in either Computational Astrophysics, Computational Particle Physics, Mathematical Finances, or Earth and Environmental Sciences.
- As part of its ExCALIBUR activities, Durham organises a series of performance analysis workshops: https://tinyurl.com/performanceanalysis2021

Edinburgh

- MSc in High Performance Computing and MSc in High Performance Computing with Data Science
  http://www.epcc.ed.ac.uk/msc . The MSc in High Performance Computing provides students with a thorough grounding in HPC technologies and their practical application. Its main target is students who have a keen interest in programming, computer science and would like to learn about HPC and parallel programming. The MSc in High Performance Computing with Data Science provides students with a thorough grounding in HPC technologies together with a practical understanding of the key ideas and techniques of data science and the HPC and other software tools that underpin them.
- Computational Science & Engineering training
  http://www.archer2.ac.uk/training. As the CSE provider for the ARCHER and ARCHER2 services, EPCC also provides a large number of training courses both online and around the UK each year. The training ranges from basic skills to teach users how to use ARCHER and write basic parallel programs, to very in-depth training focussing on specific applications. Training material and detailed documentation is available freely available online. A lot of courses are run in collaboration with other activities such as the PRACE training programme.
- HPC Carpentry
  EPCC was a key developer of the application of the Software Carpentry training methodology to HPC. It regularly runs HPC Carpentry courses for the national services.
- PRACE Training
  https://training.prace-ri.eu/ . As described in more detail below (see PRACE), EPCC has been one of

14 PRACE Training Centres for a number of years. PRACE organises a programme of 90+ training courses across Europe every year.

- Supercomputing MOOC
https://www.epcc.ed.ac.uk/education-training/online-learning/supercomputing-mooc . EPCC also offers a Supercomputing MOOC and hosts various online training resources for the National HPC services it hosts and has recently launched an online MSc focussed on Data Science. An Online self-service course is also now available.
- PhD Programme
https://www.epcc.ed.ac.uk/education-training/phds-high-performance-computing . EPCC operates a PhD programme largely focused on HPC and has been a key part of the HPC Europa Transnational Access research visit programme, which funds collaborative research visits of up to three months' duration for computational scientists working in any discipline which can benefit from access to some of the most powerful computing facilities across Europe.

Imperial College London

- MSc in Applied Computational Science and Engineering
https://www.imperial.ac.uk/study/pg/earth-science/computational-science/. This course covers the key topics of numerical methods, computational science, and how to solve large scale problems by applying novel science and engineering approaches. It is suitable for graduates of disciplines including mathematics and physical sciences, geophysics and engineering, and computer science. This immersive, hands-on MSc course enable students to develop their skills and techniques for a range of science and engineering applications utilising HPC resources. Students learn alongside world-class researchers in the Department of Earth Science and Engineering.
- Yearly HPC Summer School
https://www.imperial.ac.uk/computational-methods/news-and-events/. The Research Computing Service and the Computational Methods Hub organises an annual HPC summer school. This series of events bring together the scientific community, external lecturers and the Research Computing team for one week of tutorials, lectures and exchange of ideas. In addition, they are planning to hold Research Software Engineering sessions.

Liverpool

- MSc in Big Data and High Performance Computing
https://www.liverpool.ac.uk/study/postgraduate-taught/taught/big-data-msc/overview/. The MSc Big Data and High Performance Computing programme provides students with an in-depth understanding of big data analysis and processing using high performance computing technology. This MSc programme enables students to gain a sought-after qualification in areas of computing in great demand worldwide. The programme was developed between the STFC Hartree Centre and the University, and has had recent industrial input.

Software Sustainability Institute (SSI)

- Carpentry Programmes
https://www.software.ac.uk/programmes-and-events/carpentry-programmes. The SSI provides a wide range of "Carpentry Programmes" focussed on Software, Data and Libraries.
- Research Software Engineer support
https://www.software.ac.uk/research-software-engineers. The SSI developed the concept of a Research Software Engineer and has a number of training programmes and support activities focused on supporting RSEs during their career.

- Research Software Camps
  https://www.software.ac.uk/research-software-camps. The SSI runs free online Research Software Camps twice a year over the course of two weeks. Each Camp focusses on introducing and exploring a topic around research software, thus starting discussions among various research communities – including the HPC community.

Various other programmes

- HPC Autumn Academy
  https://www.csc.cam.ac.uk/academic/cpd/hpcacademy. The HPC Autumn Academy grew out of a consortium of UK Universities and Research Institutes that was formed to ensure that appropriate high-quality training in advanced aspects of High Performance Computing is available to researchers in the UK. The consortium was established in 2010 in response to a call from the Engineering and Physical Sciences Research Council, and consists of 12 Universities and one Research Council Laboratory. A key feature of the Consortium's approach is that it seeks to support supervisors in the training of their PhD students, while not limiting its training to PhD students. The annual HPC Autumn Academy is a short (2-week) residential course and is now led by Cambridge.
- The HEP Software foundation
  https://hepsoftwarefoundation.org/workinggroups/training.html. The foundation offers specific training in particle physics software, with the ultimate goal to develop a complete software curriculum for those involved in software development in particle physics. The organisation is centred around CERN with contributions from various other funding agencies. The main skills that are developed include python, C++ as well as training in parallel programming (CUDA, SYCL, etc). It links together schools offered by other providers. The ultimate training provision is provided by mentors.
- AI for Science training
  The current training provisions for AI for Science (at Exascale) is limited. There are training programmes inside STFC Campuses (Rutherford Appleton Laboratory and Darebury sites), and DiRAC (in collaboration with SciML in RAL), and from various US labs. These are in the form of:
    - Lectures, including Hands-on practical sessions
      Although there are a number of learning resources for AI, they are rarely aimed at Science or Engineering.
    - Generic AI/ML training materials. Although they offer an understanding of the basic principles, they largely ignore the practical settings of dealing with noisy, large-scale scientific datasets.
    - Seminars
      These cover some of the state-of-the-art aspects of AI/ML. Several such events take place in various data intensive CDTs, national labs, and universities.
    - Special targeted workshops
      These events are very useful for developing very specific skillsets among the community. For example, one-day workshops focussed on developing surrogate models or generative models can offer a very good insight into the techniques. A number of events are organised by various research groups, but rarely aimed at the whole community.

5.2 Current European HPC Training

Current training available in Europe in alphabetical order is listed below. Please note that this is not a comprehensive list but does represent the better known activities.

Cineca (Italy)

- Courses and schools are offered each year by Cineca on supercomputing and related subjects; see https://eventi.cineca.it/en/hpc/catalogue. Cineca is a non-profit Consortium, made up of 70 Italian universities, 5 Italian Research Institutions and the Italian Ministry of Education. Today it is the largest Italian computing centre, one of the most important worldwide. With more seven hundred employees, it operates in the technological transfer sector through high performance scientific computing, the management and development of networks and web based services, and the development of complex information systems for treating large amounts of data.

Dublin (Ireland)

- MSc in High Performance Computing; see https://www.tchpc.tcd.ie/node/943. The aim of the course is to train students in practical applications of high-performance technical computing in industry, finance and research. Course content includes computer architecture, software optimisation, parallel programming, classical simulation and stochastic modelling. Application areas include simulation of physical, chemical and biological systems, financial risk management, telecommunications performance modelling, optimisation and data mining. The course has a number of optional elements, allowing specialization in application areas.

UVSQ + UParis-Saclay + Telecom SudParis + INSTN (France)

- Master's degree in High Performance Computing, see http://www.chps.uvsq.fr/, which is The *Calcul Haute Performance Simulation* (CHPS) master's degree is a professional and a research programme which aims to train high-level scientific executives able to master two major technological developments: the use of high-performance computers and digital simulation. It offers two courses: "High Performance Computing and simulation" and "Modeling and simulation with High Performance Computing".

Ecole des Mines de Paris - PSL and Université Côte d'Azur (France)

- MSc in high performance computing and data sciences; see http://web.univ-cotedazur.fr//education/training?AIHPC18&lang=en#.X_XF8VP7RhE. This MSc program focuses on advanced education and the practice of HPC and data-driven applications. It provides students with an in-depth understanding of parallel programming, modeling, simulation and analysis of intensive data. Graduates will be prepared to pursue an exciting career in the fast-growing fields of HPC and Data Sciences.

HLRS (Germany)

- Courses and schools are offered each year by HLRS on supercomputing and related subjects; see https://www.hlrs.de/training/. The HLRS was established in 1996 as Germany's first national HPC center. It is a research and service institution affiliated to the University of Stuttgart offering services to academic users and industry. Over the last 20 years HLRS has built up world-leading expertise in supporting and training end-users, from a variety of application fields with a focus on engineering. HLRS has built expertise in fields like parallel programming, numerical methods for HPC, visualization, Grid and Cloud concepts, as well as Big Data, Machine Learning and Artificial Intelligence.

IDRIS (France)

- Courses and schools are offered each year by IDRIS on supercomputing and related subjects; see https://cours.idris.fr/php-plan/affiche_planning.php?total. The Institute for Development and

Resources in Intensive Scientific Computing (IDRIS), founded in 1993, is a centre of excellence in intensive numerical calculations which serves the research branches of extreme computing. This concerns the application aspects (large-scale simulations) as much as the research inherent to high performance computation (calculation infrastructures, resolution methods and associated algorithms, processing of large data volumes, etc.). IDRIS is the major centre of very high performance intensive numerical computation for the French National Centre for Scientific Research (CNRS). Together with the two other national centres, CINES (the computing centre for the French Ministry of Higher Education and Research) and the Very Large Computing Center (TGCC) of the French Alternative Energies and Atomic Energy Commission (CEA), and coordinated by GENCI (Grand Équipement National de Calcul Intensif), IDRIS participates in the installation of national computer resources for the use of government-funded research which requires extreme computing means.

PRACE (EU)

PRACE is engaged in a wide range of pan-European training activities to develop skills and competences related to HPC and computational science. It has several approaches to conduct this training:

- Face-to-face Training Events; see https://training.prace-ri.eu/. They are mainly focused on general HPC skills and competences aimed at intermediate to advanced users of HPC. Some introductory courses are still included in its training programme. PRACE organises annually a programme of 90+ courses via its 14 PRACE Training Centres (in Austria, Belgium, Czech Rep., Finland, France, Germany, Greece, Ireland, Italy, the Netherlands, UK, Slovenia, Spain, Sweden). It is complemented by Seasonal Schools and special On-demand events run in collaboration with other projects and European Centers of Excellence (CoEs).
- Summer School; see http://www.ihpcss.org/. Every year, PRACE collaborates with partners from Canada (the SciNet Consortium), Japan (RIKEN R-CCS) and the U.S. (XSEDE) to organise the International HPC Summer School for the world's brightest talents in HPC along with world-leading computational scientists and HPC instructors. Initiated by NSF's TeraGrid and EU's DEISA project, the first school was organized in Italy in 2010, and was then continued by NSF's XSEDE and EU's PRACE projects with locations alternating between North America and Europe. Initially starting with a total of 60 graduate students and post-docs from Europe and the US, this number was increased to 70 students in 2013 when Japan joined in through RIKEN and then further increased to 80 students in 2014 when Canada additionally joined initially through Compute Canada/Calcule Canada, with a transition to the SciNet HPC Consortium Canada in 2018. Selection of students takes place after an open call typically published in December/January. Typically, a high diversity of science areas and more than 20 nationalities from four continents are represented. The objective of the Summer School is to familiarize the best students of the respective continents or countries in computational sciences with a strong bond to supercomputing with all major state-of-the-art aspects related to HPC for a broad range of scientific disciplines, catalyze the formation of networks, provide mentoring through faculty members and supercomputing experts from renowned HPC centers, and to facilitate international exchange and open further carrier options.
- The Summer of HPC programme; see https://summerofhpc.prace-ri.eu/. Since 2013, the Summer of HPC programme is aimed at stimulating interest in HPC among students by providing an initial 1-week training course followed by working in a European HPC centre abroad for approx. 2 months in the summer (July – August).

- Online/Remote Learning
  PRACE has developed online training resources that are accessible globally. These resources include:
  - PRACE Tutorials – short tutorials on a few HPC subjects
  - PRACE MOOCs – are regularly organised on the FutureLearn platform.
  - Training material repository – this contains files (e.g. presentations, notes, hand-outs) that are typically aggregated from past PRACE training events.

5.3 Training and skills requirements– present and future

The preceding sections capture a significant number of training programmes. However, there is strong evidence that many RSEs have little or no HPC skills and yet there is significant and growing demand for them. To grow the HPC RSE community it is therefore important to identify where training and skills gaps lie both today, and in the future, as we enter the Exascale era. The following skills, either learned through project work or through training, are commonly held by successful RSEs working in the HPC domain:

Core skills

1. Introduction to parallel computing
2. General HPC programming paradigms
3. Supercomputer architectures – present and future
4. Underpinning mathematics of numerical modelling

Programming skills

1. Key HPC languages – C/C++ and Fortran
2. Domain Specific Languages and code generation frameworks
3. Message passing – specifically general MPI training
4. Data handling using MPI-IO
5. Understanding how MPI works – not just how it is used
6. Heterogeneous architecture programming
7. Use and programming of accelerators
8. How compilers work – from high level code to optimised low level code
9. Complex workflow implementation and management – both application and data
10. Building fault tolerant code
11. Numerical reproducibility

Debugging, performance profiling and optimisation

1. Profiling and performance analysis
2. Performance modelling
3. Debugging skills
4. Code tuning using performance analysis and models
5. Algorithmic optimisation

Software career skills

1. Being a successful computational scientist
2. Scientific software development processes
3. Modern source code management
4. Maintaining large collaborative code bases
5. Developing re-usable code – focus on optimised libraries

6. Paper writing skills

AI and Machine Learning

1. Machine Learning for science and engineering
2. Embedding AI and Machine Learning surrogate models in HPC codes
3. Optimising and debugging Machine Learning models on large GPU clusters
4. Expert usage of containers for AI and Machine Learning
5. Frameworks for distributed learning
6. AI and Machine Learning benchmarking

Application and Data Workflows

1. Data workflows and their optimisation
2. Job management systems
3. Use and optimisation of memory and storage hierarchies
4. Data movement
5. Dataset representations, use and optimisation
6. Distributed data management – across multiple sites
7. Large-scale data visualisation
8. In-situ data visualisations
9. Data processing of large data-sets
10. Data analytics
11. Containers for HPC

6 Addressing the training and skills gaps

Section 5.3 identified the key skills that an RSE working in the HPC domain as we approach the Exascale era requires. Although training in many of these areas is already provided through the activities listed in Sections 5.1 and 5.2, there are a number of areas where gaps exist today.

The authors of this report have identified the following key training and skills gaps where investment in provision would help to accelerate the growth of the pool of RSEs in the UK properly prepared for the challenges that the first Exascale supercomputers will bring.

6.1 Parallel paradigms

While there is considerable existing material to help new HPC software developers use specific parallel paradigms (for example, how to program using the Message Passing Interface), there are clear training gaps in terms of explanation of how particular parallel paradigms work. A lack of this understanding can lead to the wrong approach being taken or the right approach being applied poorly. For example, there is very little understanding of the pros and cons of short- versus long-message algorithms.

Suggested solutions to tackle this issue include:

- Providing easy access to collections of expert advice.
- Capturing and publicising implementation specific features of libraries that support particular parallel paradigms. For example, implementation specific details of different MPI libraries on particular system types.
- Providing better information on environment variables and how they relate to specific systems and libraries.
- Better training courses – focussed on teaching by example – and linked to benchmarking and optimisation strategies and including a focus on specific system types.

6.2 Data and application workflows and their optimisation

Until recently, the worlds of HPC and Data Science have been reasonably separate. However, as the use of AI and Machine Learning has grown, whilst at the same time traditional modelling and simulation applications have produced ever increasing amounts of data, these different worlds are becoming increasingly interconnected. It is becoming common for a successful RSE to need to understand how to manage large datasets often produced in complex application and data workflows. As the scale of computing available on Exascale systems becomes available, the convergence of these skill sets – which in the past have more often been provided by Data Managers in large experiments for example – will accelerate.

Suggested solutions to tackle this issue include:

- Providing more Data Science training on HPC courses and vice versa.
- Bringing the computational science use of AI and the computer science use of AI together.
- Developing training in data and information management including training on data compression and encryption for securing handling of sensitive data.
- Provision of training in the management and visualisation of very large datasets building on the experience of communities that have coped with these challenges for many years (e.g. the numerical weather simulation and the experimental particle physics communities).

6.3 HPC and AI benchmarking of systems, codes and models

The intelligent use of benchmarking and performance modelling is of critical importance as the scale of supercomputers and the amount parallelism they contain, continues to grow exponentially. Only through such use can we understand which codes can be optimised for the Exascale through incremental changes or which need disruptive change (e.g. a new algorithm or data representation).

Suggested solutions to tackle this issue include:

- Improved training in benchmarking – including comparisons of different benchmarks (HPL versus HPCG or IO500 versus individual IO benchmarks).
- Differentiating clearly between the use of benchmarking for procurement and the use of benchmarking for code optimisation – what approaches work best where and why.
- Teaching how to compare different software frameworks. In the AI domain comparing different machine learning models and frameworks (e.g. TensorFlow versus PyTorch). In the HPC domain comparing different parallelisation approaches (e.g. OpenACC versus OpenMPI).
- Learning how to look at both sides of a problem. Given a machine, how does a kernel or code perform on it. Given a set of kernels or codes, what kind of machine is best suited to them.

6.4 Heterogeneous architecture programming

While the highest scientific throughput will probably be achieved on cores-only systems, for practical reasons (space and power) most Exascale systems will include accelerators. For the first systems in Europe and the USA these will be some form of GPU. There is therefore a need to greatly increase the number of scientific software developers and RSEs who proficient at programming heterogeneous computing architectures and particularly GPUs. The UK has shied away from the adoption of such systems to date, and this is now a critical gap in the nation's computational science skill set.

Suggested solutions to tackle this issue include:

- Increased training opportunities on GPU training courses, including hackathons.
- A review of the core computational science codes used in the UK today to understand which have no GPU version and which already have a strong GPU version (developed here or overseas).
- The transfer of knowledge from one application domain to another will be important and this should be a clear target.
- Existing experts should be encouraged to transfer their knowledge to new users.
- A better understanding of abstraction versus performance realisation should be used to set a clear software strategy for the use of GPU programming frameworks in the UK. As memory models simplify, is this still a big issue? Do we over value code portability?

6.5 Containerisation

As indicated in Section 6.2 the worlds of HPC and Data Science are converging. Today there are many challenges with running Data Science frameworks on HPC systems and HPC applications on Cloud infrastructure. The use of containers to package up software applications and their various libraries is becoming increasingly common. Knowledge and skills in this area are still quite limited. There is therefore a clear need to enhance them.

Suggested solutions to tackle this issue include:

- New training courses need to be developed in this area not just for RSEs but also for DevOps[9] type staff in HPC centres.
- People who do understand containerisation need to be exposed to MPI and Parallel Paradigms from the world of HPC.
- Training on Data Science for HPC users should be developed.
- There should be engagement with vendors to help guide them in their decisions as they seek to support the ongoing convergence of these areas.

[9] The term DevOps is commonly used to refer to members of the IT community who combine software development (Dev) with IT operations skills (Ops).

7 Long-term Career Development

As we move towards a new generation of new Exascale supercomputers, we need to look carefully into the roles of Research Software Engineers (RSEs) and how these roles can map onto career paths within (or in the neighbourhood of) academia.

RSEs typically work in Universities or National Labs. Within universities, their role is often seen as secondary to academic success: RSEs provide essential support and assist academic staff to write internationally leading papers and secure grant funding for their research groups. Because of the metrics typically used, RSEs struggle to be recognised in their own right. Within national laboratories, the duration of projects is longer and the continuity of project funding is more assured. In this environment, RSEs have greater opportunity to be directly linked to the project's success.

For academic staff (some of who are also RSEs), the metrics of success (and hence promotion) are typically based around publication of papers and results (or patents). The greater the impact of this work the greater the success. For software development, this approach comes unravelled. The emphasis on open-software makes patents for software or algorithmic developments extremely unlikely. Furthermore, although an RSE's work may facilitate new or better science, they may not contribute to the scientific application of their software, and hence do not automatically gain co-authorship on the resulting papers. The Journal of Open Research Software (https://openresearchsoftware.metajnl.com/) was created to address this issue by providing a place where papers describing research software applications can be published. Furthermore, applying DOIs to software is becoming more common and this helps to increase citations for the RSEs who have been key to its development. However, since RSEs are not primarily focussed on the writing of publications, there is a strong argument that publications should not be a significant metric of career success as an RSE.

Another metric of academic success is the level of grant funding secured. In the current funding system, many RSEs are employed in roles that are prevented from submitting independent funding applications. When funding proposals are assessed, applications are typically scored on their direct scientific impact rather than on the value of the science that will be facilitated down-stream. NERC has attempted to combat this by introducing the concept of a "technology-led grant" which explicitly focusses on research with this sort of indirect impact. Other funders could consider this model. There is lots of evidence that submissions by RSEs, or in fact any researcher, to the conventional grants system which focus primarily on software development are unlikely to succeed. One of the strengths of the eCSE programme for ARCHER and now ARCHER2 is that it is designed specifically to fund such projects and often RSEs will have their first grant success through it.

Some organisations, such as the UK Met Office and EPCC, have created career-based progression structures which recognise the skill-sets and growth of software focussed staff. In the UK Met Office, the equivalent of RSEs progress through their careers based on their:

- Technical Insight and Independence
- Underpinning Technical Skills, Experience and Knowledge
- Scientific Software Assurance
- Technical Strategic Thinking
- Technical Leadership, Coaching and Mentoring
- Communication and Representation

In many institutions, however, the lack of high-visibility metrics and funding opportunities appropriate to RSE positions make it hard to convert a fixed term position (typically derived from grant funding obtained by academic staff) into long-term or permanent positions. The lack of a long-term career path is detrimental both to the RSEs themselves and to the projects on which they work. The breadth of the RSE role is another factor that makes it difficult to define criteria that fit everybody. It is, however, important to propose some criteria and make sure that they cover most instances.

Providing a career path for RSEs that does not rely on the conventional academic metrics is crucial for several reasons.

- Fair treatment of workers who provide important contributions for research based on high performance computing.
- A clear career path is necessary in order to attract and retain good people. Otherwise, we create a perennial cycle of training new people who leave at the first opportunity to a job with more security, which could be to a postdoc or a job in industry. This factor grows in importance with the individual RSE.
- It is a necessary condition to create a community of RSEs strongly engaged with the frontier of HPC.

Addressing the career-path issue is vital. Computing hardware is becoming more specialised and greater in-depth knowledge is required to use it efficiently. RSEs will be key to implementing developments that are necessary to enable new science. RSEs provide the essential knowledge and skills to enable new science on future computing hardware. It is clear by looking at the forthcoming generation of new supercomputers that the near future will be dominated by heterogeneous architectures with a variety of accelerators. Efficient usage of these new machines will require a constant update and re-design of the software used by the UK researchers and industries. These tasks require a highly-skilled workforce, with some skill sets being common across disciplines and some other skill sets being very domain specific.

We will summarise here the potential roles of RSEs, trying to highlight how these roles maps into jobs within and outside academia.

7.1 Maintain & develop software

We can foresee that RSEs will be involved in the following activities:

- Porting existing algorithms to new architectures.
- Implementing and/or designing new algorithms. Designing new algorithms is a proper research field by itself and would require RSEs with strong domain-specific knowledge.
- Working in close contact with industry to understand and exploit the latest and forthcoming hardware solutions.
- Innovating: technical innovation in HPC that allows application developers to master the very high levels of heterogeneous parallelism that we will see in Exascale systems.
- Renewing a focus on software code quality assurance.
- Developing new methods for managing complex data and application workflows on Exascale systems.

The last of these activities has the potential to generate impact statements for future REF evaluations, and it is essential that credit is given directly to the RSE. In many cases, however, the beneficiaries of this work are the academic staff who are subsequently able to work more efficiently and at greater scale. It is essential to develop approaches for rewarding RSEs for their contribution.

7.2 Benchmarking

RSEs will play a leading role in benchmarking and hence in optimising the exploitation of new resources. These activities, which are crucial to increase the scientific return per £ invested, will include for instance:

- benchmarking of new code;
- benchmarking of new architectures, identifying bottlenecks.

These are skills that are probably common across all disciplines that make use of HPC. They are nonetheless sophisticated skills that are usually developed through many years of practice. Having a proper career path for RSEs in academia would guarantee that there are *senior* RSEs that can train and mentor the new generations. We come back to this point later.

7.3 Active participation in research

RSEs with domain-specific skills can lead the implementation of software for new research projects and be directly involved in scientific research. This should not be a requirement for the job, but rather a possibility. This will be one of the main drivers to attract skilled PDRAs into these positions, which can otherwise be seen as *second choice.* RSEs that author papers can be submitted for REF and, therefore, are more valuable to Schools, which can then be convinced to co-fund some of these positions. A key principle we should be aiming for is that RSEs be recognised as researchers but researchers who are focussed on different parts of the problem to the scientists. RSEs should not be understood to simply be implementing things that others have already developed (albeit poorly) – they have a much wider role to play than this and deserve to receive credit for it.

7.4 Training

Training and advanced education will play a central role to ensure that we develop a community of users that can effectively exploit the full power of Exascale computing. These users should span both academia and industry, with an active interface between the two.

For example, the ExCALIBUR ExALAT Design & Development Working Group has identified several directions that offer training opportunities:

- Courses that target new developments – e.g. programming GPUs.
- Training of students and PDRAs.
- Training of the next generation of RSEs. This will become increasingly important as the complexity of the tasks increases with Exascale hardware.
- Contributions to Degree Programmes at home institutions, especially at MSc level.
- Establish a number of projects with industrial partners, which would be the foundation of the interface between academia and industry. RSEs will be in charge of supervision of students who do placements with industry.

Activities like MSc courses generate revenue and, therefore, are more likely to be supported by Universities.

7.5 The RSE Career Path

Looking at the above tasks we can see several facets that should allow us to design a career path for RSEs. The National Laboratories and Universities need to recognise the increasingly high level of skills needed for an RSE in the era of Exascale hardware. RSEs can work on a broad portfolio of projects across multiple disciplines but can also be highly trained in a specific domain. Individual job profiles should emerge from the needs of the scientific community, and not be set in stone with a top-down approach. Besides the traditional role in developing and maintaining code, RSEs can take an active role in the teaching programme of universities, contribute to a research activity, train their peers, and develop impact cases. RSEs can operate close to the boundary with industry, so that employment in the private sector would also be a natural outcome for some of them, similarly to what happens to PDRAs when they switch to jobs in industry. These RSEs would become effective messengers transferring knowledge between the two communities. In all cases it is possible to envisage positions that are co-funded by some combination of UKRI, Universities and Industry.

We need, however, to distinguish two different aspects of the career path.

1. **The duration of contracts:** Skilled RSEs need to be offered long-term job security. Being employed on a series of successive 2-year (and often less) contracts leads to demotivated staff and to rapid job turn over and skill loss. A partial solution to this could be e.g. the creation of a university wide pool of RSEs, as seen at 29 universities across the UK, that can be deployed to different projects as funding streams ramp up and dissipate. This requires the host establishment to take on the risk of smoothing over peaks and troughs in funding. This approach will support a core of RSEs at a university but is unlikely to support the many thousands of RSE who are embedded in research projects.
2. **Career progression:** Current promotion criteria do not favour the progression of RSEs. A greater appreciation of the RSE role is needed. One way this would be greatly helped would be by making many more RSE fellowships available to help Universities establish a pool of RSEs. Another important avenue is allowing RSEs to take ownership of the codes or techniques that they developed. This is facilitated nowadays by the possibility to publish software repositories with an associated DOI. Other KPIs could include the participation to training and managing of a pool of RSEs, and the participation to (and design of) specific courses that could be included in degree programmes. At the same time, this needs to be integrated into the goal of achieving longer-term job stability. We should aim to have a career progression that parallels that of academic staff: starting with fixed-term contracts, moving to permanent positions and potentially to promotion.

Finally, we note that the RSE movement does not currently have its own accreditation system, i.e. CSci, CEng etc, rather RSEs need to align with either IoP, IMA, BCS etc. The Society of Research Software Engineering is currently reviewing plans to implement their own accreditation system.

8 Conclusions and Recommendations

This report has focussed on the role of the Research Software Engineer (RSE) in the domain of High Performance Computing (HPC) as we prepare for the arrival of Exascale supercomputers. The report is intended to inform decision making by the ExCALIBUR project, and more broadly UKRI, with regard to investment in future RSE activities in the HPC domain.

The ExCALIBUR (Exascale Computing ALgorithms & Infrastructures Benefiting UK Research) project is a 5-year, £46m programme of activities funded through the UK Government's Strategic Priorities Fund. UKRI's Engineering and Physical Sciences Research Council and the Met Office are leading this programme in partnership with the UK Atomic Energy Authority, the Natural Environment Research Council, the Medical Research Council and the Science and Technologies Facilities Council.

This report has presented a broad overview of the RSE role, with a particular focus on the RSE skills that are required for HPC. It has reviewed similar International Exascale Software Programmes for comparison purposes and also looked at current HPC training provision in the UK and Europe. In so doing it has identified a set of training and skills requirements and as a result a number of training and skills gaps in current provision. Finally, it has looked at issues surrounding the long-term career development of RSEs.

This landscape review leads us to make the following key recommendations:

1. The Research Software Engineer role is a concept that has significantly grown in importance over the past decade. Research by the Software Sustainability Institute has shown that 92% of academics make use of some type of research software, with 69% regarding it as fundamental to their work. UKRI should continue to invest in the development of Research Software Engineering in the UK.
2. The skills required by Research Software Engineers are very broad, spanning almost all areas of the research base and all scales of computing and data management. This report has identified specific skills in High Performance Computing that are required by Research Software Engineers. As we grow the overall number of such staff in Universities, National Laboratories and other research organisations we should also grow the number of such staff with specific High Performance Computing skills.
3. An ExCALIBUR Training Programme for Research Software Engineers who want to focus on High Performance Computing should be established.
4. A long-term training and education strategy should be developed with two distinct aims. Firstly, building on existing training, the strategy should outline how to grow the next generation of Research Software Engineers with High Performance Computing skills. Secondly, building on the training and skills gaps identified in this report, training should be developed for existing Research Software Engineers working in this domain to ensure they have the necessary skills as we approach the Exascale era.
5. Training can take many forms. A variety of different training models should be adopted – including postgraduate study, workshops, hackathons and bootcamps. Wherever possible training should be made available as online training as well as in face-to-face training opportunities.
6. Clear career paths for Research Software Engineers, and funding opportunities for software development allowing them to apply and develop skills, are crucially important to ensure that, once trained, the knowledge they have gained stays in the research sector and grows over time. Too often such staff move from one research group post to the next with no continuity of employment and as a result they leave the sector entirely. The contribution of software engineering needs to be recognised in university recruitment and promotion procedures.

7. UKRI should ensure that it supports the message that Research Software Engineers are a highly valued resource at Universities, National Laboratories and other research organisations by providing clear guidance for inclusion of RSEs on grants. All of these organisations have a strong role to play in supporting long-term career paths for those who choose this role.
8. Many companies employ Research Software Engineers but often do not recognise the similarity of the role with such staff in the University sector. Greater collaboration and transfer of skills in both directions between the academic and industrial research sectors should be encouraged, particularly from industry to academia.
9. As UKRI develops the Exascale Supercomputer Project's software programme it should ensure that it encompasses a variety of different types of activity including an enlarged eCSE programme, sustained support for our largest computational projects and consortia, and also support for the development of new computational codes and challenges from across UKRI's research base and the industrial research sector including SMEs.

Annex A: AI for Science

Exascale systems will support large-scale AI challenges. With regard to what has become known as AI for Science, activities across various national labs and universities, can broadly be defined under the following four key areas:

1. AI-enabled applications
2. AI algorithms and foundational research
3. AI software infrastructure
4. New hardware technologies for AI, including edge initiatives.

Research, and developmental activities taking place across various universities and national labs are organised into these key areas, covering one or more of the following partially overlapping strands:

- Chemistry, materials and nanoscience
- Earth and environmental sciences
- Biology and life sciences
- High energy physics
- Nuclear physics
- Fusion
- Engineering and manufacturing
- Smart Energy
- AI for Computer Science
- AI Foundations and open problems
- Software environments and software research
- Data lifecycle and infrastructure
- AI for imaging
- Hardware architectures
- AI at the edge
- Facilities integration and AI ecosystem

In doing so, the efforts across national and international labs can be considered to cover a number of activities covering the following scopes:

- Providing a framework for organizing the research software engineering community
- Creating a thorough assessment of needs, issues, and strategies for the AI for Science community to follow and adhere to
- Initiating development of a coordinated roadmap for AI software and AI software stack
- Encouraging and facilitating collaboration in education and training
- Engaging and coordinating the vendor community in cross-cutting efforts

Through a number of research programmes (funded and to be funded by various funding bodies), the AI for science community is striving to achieve the following, overarching, set of outcomes:

1. Accelerate the pace at which scientific discoveries are made. For example, this can include design, discovery, and evaluation of new materials, or discovery of exoplanets, or developing digital twins.
2. Advance the development of new hardware and software systems, instruments and simulation data streams;
3. Identify new science and relevant theories from large-scale data sources;
4. Improve the nature of conducting or planning scientific experiments by providing AI-enabled inference capabilities as part of the experimental loop;
5. Enable the design, evaluation, autonomous operation, and optimization of aspects that underpin the science and engineering behind large-scale experimental facilities, like synchrotrons;
6. Advance the development of autonomous, laboratories and scientific workflows;
7. Dramatically increase the capabilities of Exascale and future supercomputers by

8. Capitalizing on AI surrogates;
9. Automate the creation of “FAIR”-aware data sources; and
10. Develop algorithms, tools, methods and techniques for enabling the above.

Annex B: Authors of this Report

This report was written as a collaborative effort by members of the ExCALIBUR community:

Prof Mark Parsons (Editor)	The University of Edinburgh
Dr Alastair Basden	Durham University
Prof Richard Bower	Durham University
Mr Neil Chue Hong	The Software Sustainability Institute
Prof Davide Constanzo	The University of Sheffield
Mr Shaun de Witt	UK Atomic Energy Authority
Prof Luigi Del Debbio	The University of Edinburgh
Dr Glen Greed	UK Met Office
Dr David Ham	Imperial College London
Dr Phil Hasnip	The University of York
Dr Adrian Hines	UK Met Office
Prof Simon Hettrick	The Software Sustainability Institute
Dr Sylvain Laizet	Imperial College London
Dr Jeyan Thiyagalingam	Science & Technology Facilities Council
Prof Tobias Weinzierl	Durham University
Prof Garth Wells	University of Cambridge
Prof Scott Woodley	University College London
Dr Nigel Wood	UK Met Office

The authors were supported and guided by Dr Luke Davis, Dr Elizabeth Bent and Mrs Sarah King from UKRI's Engineering and Physical Sciences Research Council.