UNIVERSITY OF BIRMINGHAM | BEAR BIRMINGHAM ENVIRONMENT FOR ACADEMIC RESEARCH | software

Research Software Group

BEAR | Advanced Research Computing

2021 Report

https://www.birmingham.ac.uk/bear-software

bear-software@contacts.bham.ac.uk

Contents

Introduction

Research Software Group

The Research Software Group (RSG) is part of Advanced Research Computing at the University of Birmingham. The RSG was formed in November 2017, starting with two Research Software Engineers (RSEs) - Simon Branford and Andrew Edmondson. We designed the free Advice, Coding, and Coaching model, and began working with researchers with the aim of improving the research software written and used by the researchers. BEAR Software's mission is summed up in the words of the Software Sustainability Institute: “better software, better research”[1].

Over the next few years the group grew steadily and by the end of 2021 the RSG had ten team members, with an eleventh joining in early 2022.

The RSG work is split into three parts:

1. We support users of BEAR's high performance computing services BlueBEAR, BEAR Cloud and CaStLeS. A significant part of this work is installing, curating and supporting over a thousand applications and libraries.
2. We provide free advice, coaching and coding work, collaborating with researchers from PhD students to senior academics across the whole University.
3. We join research projects or groups for longer-term funded collaborations. This can be via funding dedicated RSEs managed by ARC, or by projects requesting time from our growing Pool of RSEs for funded work.

[1] https://www.software.ac.uk/resources/publications/better-software-better-research

The Research Software Group exists to:

- Enable the University of Birmingham's research community to get the best from their research software
- Provide specialist software engineering advice and support to researchers and research software engineers (RSEs)
- Help to enhance the University's reputation for high quality research
- Help researchers get the most from BEAR services, maximising the return on the University's investment in BEAR

2020 saw our first major funded coding engagement, where a member of the RSG worked for 6 months (at 50%) on a research project. 2021 saw a significant increase in the amount of research-funded work we were able to take on, a large part due to the new Pool RSEs we were able to recruit. At the end of 2021 we have two Pool RSEs in post, and another joining in early 2022. In the 2021-22 financial year we will have carried out over a quarter of a million pounds worth of funded work.

Advanced Research Computing

BEAR
BIRMINGHAM ENVIRONMENT
FOR ACADEMIC RESEARCH

The Birmingham Environment for Academic Research (BEAR) is a collection of complementary IT resources managed by Advanced Research Computing (ARC). These services are designed to support research and our standard services are all free at the point of use.

The Team has grown rapidly in recent years following the 2015 Vice Chancellor's Review and in direct response to the demand and specific drivers from our academic community:

- the recognition of the benefits advanced computational techniques can bring to a whole raft of disciplines
- the tsunami of data just waiting to be analysed
- the skills gap and the need to develop capability to enable the exploitation of compute power
- the need for experts to support researchers
- the demands and expectations of funding bodies
- the demand for “more” from established users of HPC, including the rapidly growing need for GPU resources to support research into and using Artificial Intelligence
- the need to provide fit for purpose technology for competitive advantage

The twenty members of ARC are organised in three Groups including the Research Software Group with its focus on fostering good practice, embedding skills and supporting the use of software. Alongside sit our Research Engagement and Data Group and the Architecture Infrastructure and Systems Group, the former making sure our services are known across campus and listening to the needs of researchers with the latter designing and building award-winning compute and storage platforms.

With over two thousand projects and over seven thousand researchers currently using one or more BEAR service, adoption continues to grow and keeps the team very busy with diverse challenges, reflecting the breadth of the University's research.

The report that follows provides an introduction to the work of the Research Software Group and gives brief case studies to illustrate the benefits partnership with the RSG, and ARC, can deliver.

Carol Sandys, Head of Advanced Research Computing and Research Engagement

Research Software Engineers

The value of good Research Software Engineering

The term 'Research Software Engineer' (RSE) was coined in March 2012 at the Collaborations Workshop. For a history of RSEs please see the State of the Nation report from 2017: https://doi.org/10.5281/zenodo.495360

Two recent, major UK government reports highlight the importance and value that RSEs can bring:

> It is difficult to overemphasise the interdependent nature of software, skills, and hardware. Without the necessary skills, good software cannot be written or maintained. Without the necessary software, hardware cannot be made use of effectively. The UK is recognised for its capabilities in computer science and software development, alongside strengths in the research domains that make use of large-scale computing. [...]
> Producing software for large-scale computing requires the use of specialist programming languages and tools, as well as constant updating and optimising to ensure software can run on the latest systems. With the increased scale and diversity of hardware, commonly used software will require significant updates to utilise future generations of supercomputers. Ensuring software is sustainable and will continue to be useable, available, and efficient in the future is critical to ensuring benefit from large-scale computing. This requires the development and embedding of best practice in software development across the computing ecosystem.[2]

---

[2] Government Office for Science, 'Large-scale computing: the case for greater UK coordination', 2021, https://www.gov.uk/government/publications/large-scale-computing-the-case-for-greater-uk-coordination, p43]

> Software lies at the heart of many research and innovation activities. It is needed to control the instrumentation to record data. It is used for the complex modelling required to understand cell function, fusion reactions and the climate. It is the enabling technology behind major advances such as decoding the human genome and the discovery of the Higgs boson and it lies at the heart of strategically important technologies such as AI. Software can be a few lines of code written by a single researcher or a major framework developed over decades by dedicated teams of researchers and software engineers. In academia, 92% of UK researchers use research software and 69% report that it is fundamental to their research with software development a research activity in its own right. The near-ubiquity of software means that it is not possible to disentangle the quality of the software from the quality of the research. Unreliable and untested software leads to unreliable results that cannot be trusted.[3]

The most concise form of the argument in favour of good software engineering is the famous quote from the SSI:

> 'Better software, better research'[4]

[3] UKRI, 'The UK's research and innovation infrastructure: opportunities to grow our capability', 2019, https://www.ukri.org/files/infrastructure/the-uks-research-and-innovation-infrastructure-opportunities-to-grow-our-capacity-final-low-res/, p125

[4] Carole Goble, 2014, https://www.software.ac.uk/resources/publications/better-software-better-research

The Team

Dr Andrew (Ed) Edmondson - Group Leader

Known as 'Ed', he started his career as a software engineer and team leader at QinetiQ, after completing an MMath at the University of Oxford. He left QinetiQ to complete a BA in Theology at Birmingham Christian College after which he worked part-time as a senior developer at AppliansSys working on embedded Linux and Python firmware for network appliances.

He completed a part-time PhD in New Testament Textual Criticism in the Institute for Textual Scholarship and Electronic Editing (ITSEE) at the University of Birmingham supervised by Professor David Parker. The title of his PhD thesis is 'An analysis of the coherence-based genealogical method using phylogenetics'. He is currently an Honorary Fellow of ITSEE.

In 2016 Ed joined Advanced Research Computing at the University of Birmingham and founded the Research Software Group. He is an active member of the Society of Research Software Engineering, and was the Programme Chair of the 2019 UK RSE Conference.

Dr Simon Branford - Deputy Leader and Senior RSE

Simon's experience is in computational research and he have provided high performance computing and mathematical expertise on several research projects. Prior to moving to Birmingham, he was a postdoc at the University of Reading in two different areas: evolutionary biology and meteorology.

His postdoc research built on the skills he developed during his degrees. He started at the University of Oxford, where he obtained an MMath. After this, he moved to the University of Reading where he did an MSc in Network Centred Computing with a specialisation in high performance computing; and then a PhD, where he researched the use of hybrid Monte Carlo algorithms for linear algebra problems.

Simon is:

- an EasyBuild maintainer;
- a certified Carpentries Instructor and teaches on several of the workshops provided through BEAR Training;
- a member of the Tier-2 Technical Working Group.

Cerys Lewis - RSE (Web)

Cerys has over 18 years of full stack web development experience, using a variety of languages. Prior to becoming a Web RSE for University of Birmingham, she worked for the University of Warwick on their internal student administration application software, 'Tabula'. Previously to that she had a background in Retail Software developing web applications and websites. She is a graduate from Royal Holloway, University of London with a BA(Hons) in English Literature.

Mike Allaway - Senior RSE, College of Arts and Law

Mike has worked in IT within the HE sector since 2012. He started in a user support role at Cardiff Metropolitan University and moved to Birmingham in 2015 to transition into a software development role at the Birmingham Clinical Trials Unit. He joined the RSG in 2018 as an RSE and became a Senior RSE in 2019.

Mike holds a BSc with First-class honours in Business Management & Information Systems from Swansea University, as well as an MSc with Distinction in Computing from Cardiff University. In addition to his role in the RSG at Birmingham, Mike manages the website and database for the LINGUINDIC research project at the University of Oxford.

Dr James Carpenter - Research Applications Specialist

James studied and worked in the Department of Music at the University of Birmingham for 18 years before moving to the Research Software Group in autumn 2018. He completed a PhD in electroacoustic composition in 2013 and has variously worked as a teacher of composition, computer music and studio-related disciplines in the music departments at the University of Birmingham and University of Bristol. Immediately prior to joining the RSG he was Studio Technician in the Birmingham Dept of Music where part of his role involved the administration of the Mac systems, as well as the development of bespoke solutions for the technical management of large-scale and idiosyncratic multichannel loudspeaker concert systems.

Dr Simon Hartley - Senior RSE (HPC)

Simon has a degree in theoretical physics and a PhD in Physics and AI. He moved to his new role as a High Performance Computing RSE in 2020.

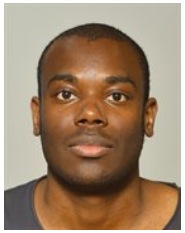
His previous position was working as the Research Software Engineer for the Centre for Computational Biology at University of Birmingham.

In his research career he has written software in C/C++, C#, MATLAB, R, Python, HTML5, BASH, JavaScript, MySQL for physics (dielectric materials), medical applications (MRI analysis and Genetic Epidemiology), interactive systems (UX and HCI Research), real-time sensor systems for light rail/trams (FPGA and mems sensors), Computer Vision Systems for Monitoring Civil Engineering (London Underground), point cloud processing/visualization for automatic model 3D generation (Radiotherapy) and has built autonomous robots which have been deployed in nuclear power stations.

Dr Mohammad Afraz Ahmed - Senior RSE

Afraz holds an MSci in theoretical physics and a PhD in computational materials science, both from the University of Birmingham. After his PhD, he worked for several years as a research fellow in the PRISM2 group in the School of Metallurgy & Materials, developing research software for multiscale materials modelling. He joined the Research Software Group in March 2020.

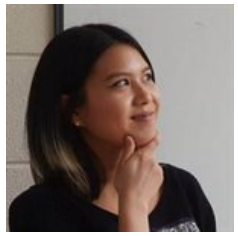
Dr Gavin Yearwood - Research Applications Specialist



Dr Gavin Yearwood first attended the University of Birmingham as an undergraduate, he remained there completing his PhD and then a further research position, all within the school of Metallurgy and Material Science.

His focus during his PhD and further research was in the production, application and optimisation of a physics based model. He built a thermomechanical predictive model and inserted it in a commercial finite element software. It was from the use of this commercial software that Gavin became familiar with Advanced Research Computing (ARC), having to run lots of large scale simulations as well as trying out various open source software to determine their effectiveness in future research projects.

Dr Jenny Wong - Senior RSE



Jenny holds a BSc and MMath in Mathematics and a PhD in Fluid Dynamics from the University of Leeds. After her PhD, she spent 3 years in France at l'Institut de Physique du Globe de Paris and l'Institut des Sciences de la Terre for her postdoctoral research investigating the dynamics of the Earth's core, using mathematical and numerical models. Jenny joined the Research Software Group in December 2021.

Previous members (left during 2021)

The following team members left in 2021. We include them here to recognise the great work they did this year in the RSG.

Dr Ryan Pepper - Senior RSE

Ryan completed an MSci in Theoretical Physics at the University of Birmingham in 2015 and then undertook a PhD in the CDT for Next Generation Computational Modelling at the University of Southampton, where he studied nanomagnetic systems and developed research software for the study of these. After his PhD he worked at Siemens on a commercial CFD package for the electronics industry.

BEAR Software

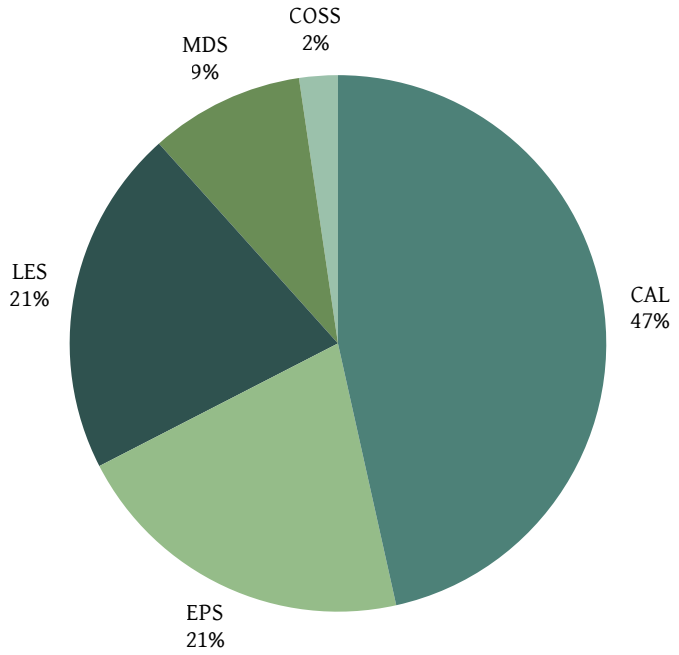
Advice, Coaching, Coding and Mentoring

The RSG offers advice sessions, and short coaching or coding engagements to all researchers in the University, free of charge, under the banner of 'BEAR Software'. We also offer free mentoring for RSEs across campus. The chart shows how our sessions were distributed between the five colleges in 2021.

The colleges are described at https://www.birmingham.ac.uk/university/colleges, but in brief are:

- Arts and Law (CAL)
- Engineering and Physical Sciences (EPS)
- Life and Environmental Sciences (LES)
- Medical and Dental Sciences (MDS)
- Social Sciences (COSS)

The large figure for CAL engagements reflects the fact that CAL fund a dedicated RSE, and implies that there is probably considerable demand in other colleges that we are not yet able to meet.



BEAR Software: Advice

If a researcher either has need for new research software or wishes to improve their existing software, then BEAR Software RSEs can be engaged to talk through the needs of the researcher and offer advice on how best to achieve those needs. To request Advice please fill in the form at https://intranet.birmingham.ac.uk/bear/sd/bear-software-advice .

BEAR Software RSEs are able to offer advice on the whole life-cycle of research software. Example topics include:

- Specifying the requirements for a new piece of software or for additions to an existing piece of software.
- Designing and architecting the software.
- Writing the software and suggestions on which programming languages and tool-kits are best suited for the software being written.
- Documenting code effectively so that when a researcher returns to it at a later date it is easy to understand, and so that it could be made available for others to edit if desired (e.g. Open Source).
- Writing user documentation so that installing, and using (including expected input and output of the software) the software is an easier process.
- Good practices for enabling Open Research.
- Producing or using a testing framework or infrastructure for the software, so that a researcher can easily spot if changes or additions to part of the software, or any third party software it relies on, breaks the existing functionality of the software.
- Porting the software to a new platform or system and how to approach optimisation of the software.
- Upgrading the software, and any third party software that may be in use, and how to ensure that the upgrades have not broken the functionality of the software.
- How to manage the release of the software, so that it is available, in an effective and useful way, for use by others in the research community and beyond.
- Integrating a researcher's software with third party software and libraries.
- Using version control to allow a researcher to manage, and track, changes to their software, data, papers, talks, etc.
- Help with specifying the research software requirements of grant applications and specific assistance with the Technical Appendix.
- Provide recommendations about software licenses and licensing issues.

- Advise on available training courses - those available locally, online, and further afield. We can also discuss tailoring existing training courses to meet the specific needs of a research group.
- General discussion about research software or research computing.

By following the BEAR Software Advice a researcher should be able to improve the reliability and maintainability of the research software they write/use and this will help the researcher with the reproducibility and robustness of their research.

BEAR Software: Coaching

If a researcher, or research group, has the need for specific research software expertise, then an RSE can be requested, free of charge, for multiple half-day coaching sessions (max. 10). During these sessions the RSE would be embedded in the research group (physically or virtually) to work with you to complete specific agreed objectives. The RSG can also provide mentoring to a RSE, to enable the RSE to develop technical and non-technical skills. To request Coaching please fill in the form at https://intranet.birmingham.ac.uk/bear/sd/bear-software-coaching-coding .

This work could follow on from an initial advice meeting where the further need would be identified and clearly defined objectives would be specified and agrees. Example tasks might include:

- Migrating a project to use version control (e.g. gitlab or github).
- Reviewing code.
- Designing and implementing a testing framework.
- Writing user documentation, licensing info etc.
- Testing, debugging, and fixing issues.
- Implementing a new feature.
- Porting the software to a new operating system.
- Designing and implementing a release process.

The BEAR RSE will work with you to complete these tasks, for example doing pair programming. Our intention is that after the sessions you will be able to continue the work in your own team and have learnt technical, research software skills. Of course, later follow-on engagements are available, demand permitting.

Example of a typical coaching engagement

The BEAR Software RSE would meet with the researcher for a number of half-day Coaching sessions, usually one session per week. During each of these sessions there would be a task, or tasks, to complete during the session. Between sessions the researcher would work further on the project and this would identify further tasks for the follow on engagements.

BEAR Software: Coding

If a researcher, or research group, has the need for specific research software expertise, then an RSE can be requested, free of charge, for up to ten days of coding time (normally spread over several weeks). To request Coding please fill in the form at https://intranet.birmingham.ac.uk/bear/sd/bear-software-coaching-coding .

This work could follow on from an initial advice meeting where the further need would be identified and clearly defined objectives would be specified and agrees. Example tasks might include:

- Implementing a new feature.
- Developing a research website.
- Re-implementing an algorithm in a different language.
- Adding parallelisation or CUDA (GPU) support to an application.

Of course, later follow-on engagements are available, demand permitting.

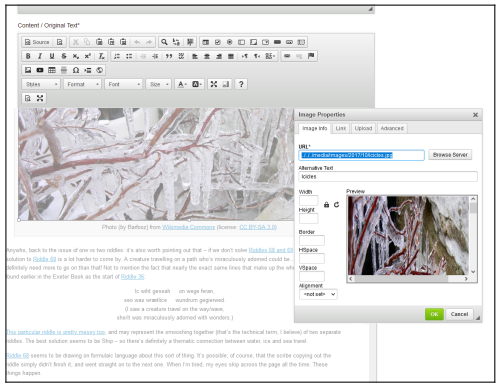
Coding Case Study – Riddle Ages Phase 2

Researcher(s)

Megan Cavell

Technologies

Web, Django, Python, HTML, CSS, JavaScript



Project Description

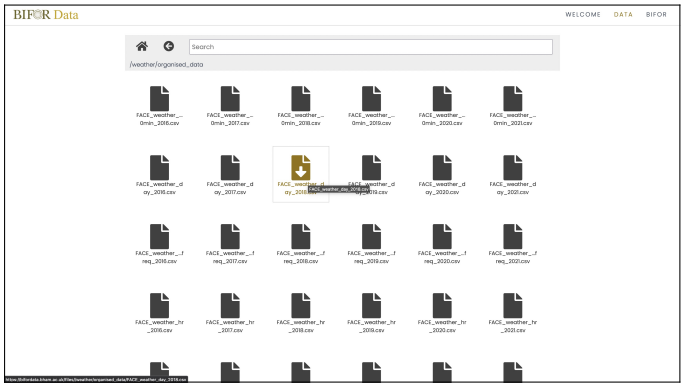
The Riddle Ages websites aims to provide Modern English translations of riddles from the entire early medieval riddle tradition — including the 7th- and 8th-century Latin riddles that inspired the Old English riddles as well as providing commentaries on interesting points about each riddle.

Building on the successful transition of the site content from the WordPress blogging site in phase one and subsequent launch of The Riddle Ages website, phase two added WYSIWYG form integration for admin functionality, which allows the researcher and her team to control all text elements, edit the underlying source, create full posts with tags and linking to other relevant posts and to upload and embed YouTube video, images and documents on demand in all parts of the site as well as adding site areas.

I'm very grateful to the whole team for phase 2 of the website, and especially for addition of the WYSIWYG. This has saved a great deal of time previously spent coding, which means my team can now focus on populating the website with more content. - Dr Megan Cavell

---

Coding Case Study - BIFoR Website



Researcher(s)

BIFoR team, College of Life and Environmental Sciences

Technologies

Web, Django, Python, HTML, CSS, JavaScript

Project Description

The Birmingham Institute of Forest Research (BIFoR) aims to provide fundamental science, social science and cultural research of direct relevance to forested landscapes anywhere in the world. They make the evidence-based case for forests as part of one-planet living.

The RSG worked with Rob MacKenzie, Giulio Curioni, and the BIFoR team to develop a website that allows open access to view and download the team's research data stored on the University's Research Data Store (RDS). By building this solution, the team have found it easier to collaborate with external partners who otherwise would have limited access to the team's data, given that the RDS requires a University IT account to access.

The website offers an intuitive interface for end users to browse the directories and files on the RDS, as well as for the BIFoR team to easily synchronise the data with the website.

'The Research Software Group totally "got it". They understood our remit and delivered an excellent product to schedule and without multiple iterations, exactly what one wants from Professional Services!' - Rob MacKenzie

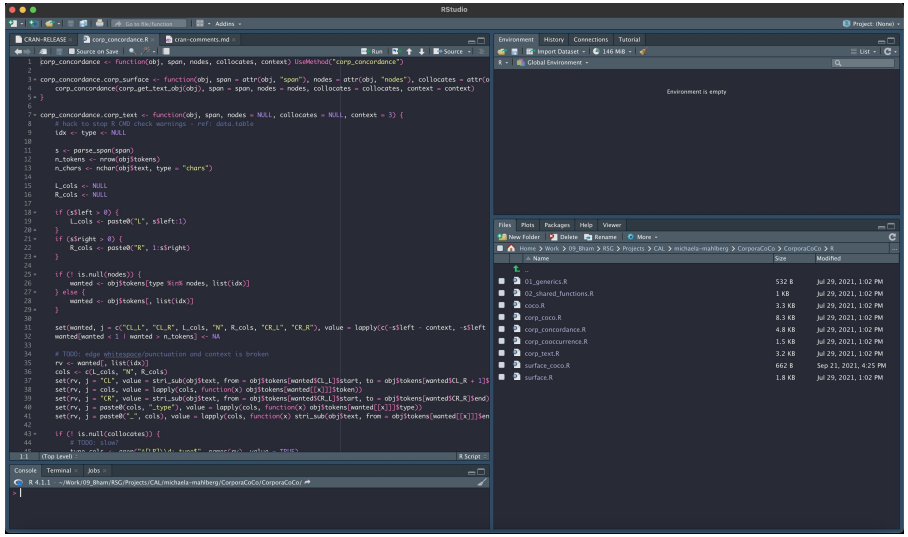
Coaching Case Study - R work with the TDA

Researcher(s)

Michaela Mahlberg, College of Arts and Law

Technologies

R



Project Description

The Times Digital Archive is a full-text facsimile of more than 200 years of The Times, one of the most highly regarded resources for eighteenth-, nineteenth-, and twentieth-century news coverage, with every page of every issue from 1785 to 2014. This historical newspaper archive allows researchers an unparalleled opportunity to search and view the best-known and most cited newspaper in the world online in its original published context.

This coding engagement set out to achieve two objectives:

1. Finish converting the full TDA text into a suitable format for Michaela's research
2. Test and develop a new release of the CorporaCoCo R library and publish to CRAN

We worked closely with the research team through this process, successfully processing the raw TDA data. This included identifying and separating different types of content (e.g. articles with 0, 1, or multiple paragraphs of text) and then combining the resulting data in a way that was required for the analysis.

We then updated the existing CorporaCoCo R library, adding software quality enhancements (e.g. automated testing and linting) and fixed various issues to make it an overall more robust and useful tool for research. We're currently working with the research team to publish this update to CRAN so that it can be openly shared and used by others in the community for their own research.

'Working with Ed and Mike on this project showed what true collaboration between academics and RSEs should be like. Digital humanities projects will only scale if they are successfully embedded in university supported infrastructures. Developing this way of working further, making it part of the day-to-day of research, will significantly enhance the university's capabilities for innovation, especially in arts and humanities subjects. Ed and Mike are simply brilliant to work with, which is a real bonus.' - Michaela Mahlberg

---

Coaching Case Study - Turing Research Data Study Group, September 2021

Researcher(s)

Alan Turing Institute; Professor Iain Styles

Technologies

Baskerville Portal: *JupyterLab*, *Python*, *R*

Project Description

Advanced Research Computing provided technical and user support on the Baskerville Tier 2 compute resources for a two-week *Data Study Group*, organised by the *Alan Turing Institute*. The groups are described on the ATI website as:

> Intensive "collaborative hackathons" hosted remotely, which bring together organisations from industry, government, and the third sector, with talented multi-disciplinary researchers from academia.

The RSG developed a bespoke *JupyterLab*-based interactive application via *Baskerville Portal*, providing GUI access to Python, R and Julia for a variety of GPU-focussed workflows.

'The Data Study Groups are a great way to bring together a diversity of viewpoints and skills to help organisations to identify new approaches to difficult problems. They're also a tremendous opportunity for early career researchers to challenge themselves on problems that are a bit different to what they usually work on, to gain experience with real-world challenges. The University is delighted to be have worked with the Alan Turing Institute to successfully deliver these activities, and RSG's support has been invaluable in enabling this to happen, giving the research teams access to state of the art infrastructure that significantly accelerate their progress.' - Professor Iain Styles, the University's Turing lead

---

Coding Case Study - Deploying AIDA on the European Space Agency's SWE Portal

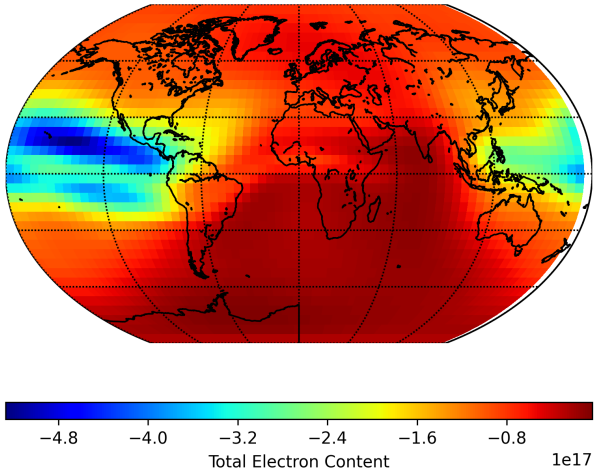
Researcher(s)

Dr. Sean Elvidge and Dr. David Themens, College of Engineering & Physical Sciences

Technologies

Web, Django, Python, HTML, CSS, JavaScript, Celery, SLURM

Project Description



Difference between background model and post data assimilation analysis of the total electron content in the ionosphere

The ionosphere is a complex and rapidly changing environment with complex statistical properties. In the last 15 years, a wide range of ionospheric data assimilation models have been developed. They employ a number of different background models and assimilation techniques. In all cases they have been implemented with the aim of making a prediction of the current state of the ionosphere and then updating this prediction with data.

Sean's group have developed AIDA, which is based on a state-of-the-art data assimilation technique known as the local ensemble transform Kalman filter (LETKF). This coding project is for the deployment of AIDA on the European Space Agency's (ESA) SWE portal, by hosting a Django site including an API for the ESA to use, and also a web UI for Birmingham researchers and others from further afield. The Django site will also host/serve additional datasets from the Research Data Store.

To date, a Django website has been developed and deployed which allows users to submit Slurm jobs to run the AIDA code on two dedicated nodes on the BlueBEAR HPC cluster. An API that reflects the current functionality of the website was developed using the Django REST Framework. Celery has been used to submit jobs and to monitor job statuses, and the infrastructure for scheduling periodic runs of AIDA has been developed and is ready for when planned changes to the statistical model have been incorporated.

In terms of running the AIDA model, three modes of operation are planned. First is for users to trigger a run of the model via the website (or ESA portal API), selecting which remote data sources to include. Second is for users to do the same thing but using data they have uploaded themselves, rather than downloading the remote data sources. Third is regular, near-realtime runs of the model based on incoming satellite data.

Coding Case Study - ExCALIBUR Hardware and Enabling Software Programme

Researcher(s)

Prof. Iain Styles, College of Engineering & Physical Sciences and Simon Thompson, Advanced Research Computing

Technologies

Fortran, C, C++, Easybuild.

Project Description

The ExCALIBUR (Exascale Computing ALgorithms & Infrastructures Benefiting UK Research) Hardware and Enabling Software research programme provides grant funding for testbed projects that give UK researchers early access to novel hardware and software technologies that may play a part in future exascale systems and services[5]. As part of this programme, Prof. Iain Styles and Simon Thompson have been awarded funding to set up a testbed for evaluating a novel accelerator technology from NextSilicon, in collaboration with University of Birmingham HPC systems partner Lenovo.

During this coding project, a range of HPC apps and mini-apps have been evaluated using NextSilicon's software simulator. This has involved porting of the applications to the new platform, verifying correctness of results, analysing performance and comparing to CPU and GPU performance (both in terms of execution time and energy usage), as well as investigating the impact of code changes on performance. In addition, Easybuild toolchains have been made for the NextSilicon platform, in order to automate the building of applications.

[5] https://excalibur.ac.uk/themes/hardware-and-enabling-software

Delivery of the hardware is expected in 2022. Once installed, the technology will be evaluated and subsequently made available to the UK HPC community as part of the Baskerville EPSRC Tier 2 service.

'This technology could substantially improve the performance of a wide range of scientific software without requiring any changes to the code. This contrasts with other accelerator technologies which often require significant rewriting of software. We are very much looking forward to receiving the new hardware so we can fully evaluate its potential following very promising early results in simulation.' - Prof. Iain Styles (project lead).

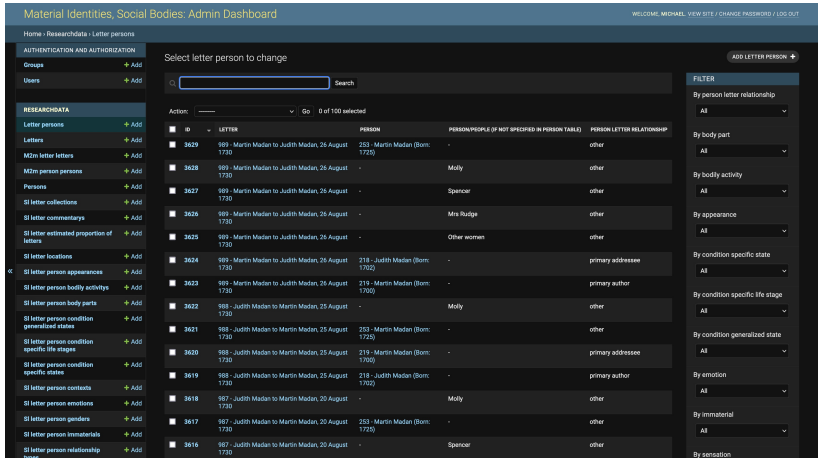
Coaching Case Study - Working with Social Bodies Data

Researcher(s)

Karen Harvey, College of Arts and Law

Technologies

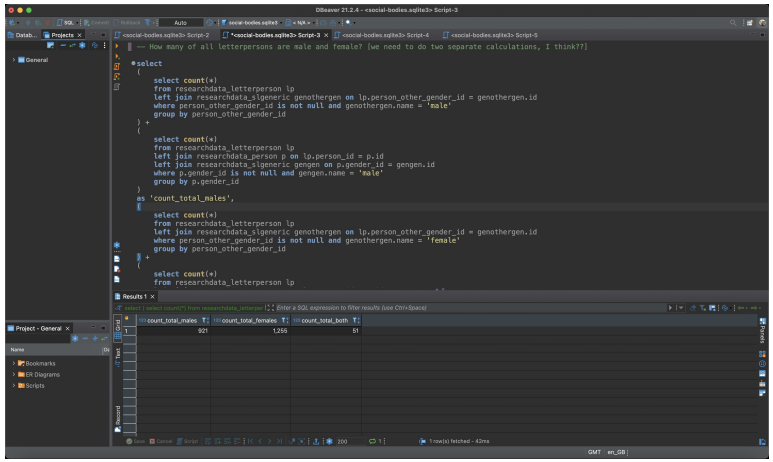
SQL, SQLite, CSV, Excel



Project Description

"Material Identities, Social Bodies: Embodiment in British Letters c.1680-1820" is a Leverhulme Trust funded project based in the School of History and Cultures. The project uses eighteenth-century letters to explore the relationships between the physical body, self and social identity, and experiences of "embodiment". The project team are transcribing and analysing thousands of letters between correspondents of family and kin, friendship, faith and business.

The RSG had previously developed the project's website: https://socialbodies.bham.ac.uk/. The website includes an admin dashboard interface to allow the research team to easily manage the project's data. However, the team required a greater level of control for more in-depth data extraction from the SQL database for further analysis. Following discussions between the team and the RSG, it was agreed that the best way for them to achieve this would be to learn SQL (Structured Query Language).



Over a 2-month period Mike coached the project team, consisting of Karen Harvey (PI), Sarah Fox (Postdoctoral Researcher), and Emily Vine (Postdoctoral Researcher). We started with basic data extraction using simple SQL SELECT queries, as the team had no prior knowledge of SQL. As newer topics were introduced over multiple coaching sessions and the team's SQL skills quickly developed, we were able to develop more complex queries using more advanced SQL concepts, such as multiple joins between tables, aggregate functions, and subqueries.

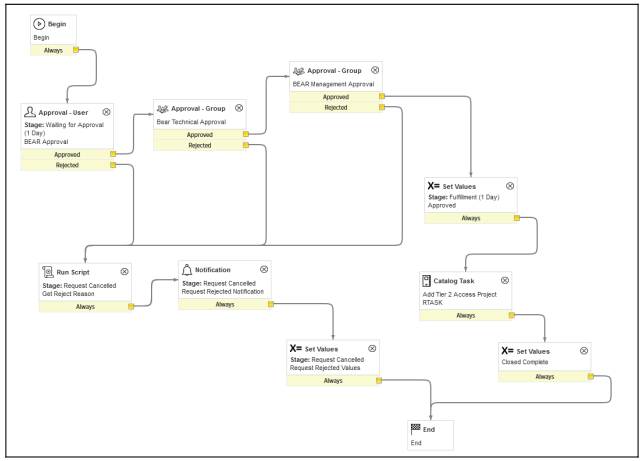
The research team are now confident SQL programmers in a relatively short space of time and have a much greater understanding and level of control of their data. This will inevitably lead to more in-depth data analyses and thus more meaningful research outputs.

'The support we received for Material Identities, Social Bodies has been outstanding: responsive, thoughtful and exactly tailored to the project's objectives. The coaching from Mike was excellent. This provided an important foundation in SQL skills before leading us on to tackle the more difficult and specific challenges of our project. With these new skills in SQL, we have been enabled to conduct more advanced research using our data. This has had a direct and significant impact on the research questions that we are able to address and a transformative effect on the project as whole.' - Karen Harvey

---

ServiceNow development

The University of Birmingham's IT Services uses ServiceNow as an incident and request platform, amongst other things. Requests from researchers for use of BEAR resources all go through ServiceNow workflows and tickets, and are then fed into an approvals system and implemented by the team. This allows provision of access to BEAR services, including the Research Data Store, BlueBEAR HPC, GPU and AI services, GitLab, Virtual Machines, and the EPSRC Tier 2 systems Baskerville and Sulis.



Cerys has acquired the necessary skills in ServiceNow development to be able to make a range of changes to the ServiceNow workflows for BEAR. This is really valuable as it allows us to be much more responsive and keep our ServiceNow workflows in sync with our actual processes as they change over time and allows for a better user experience for those using the service through a continuously improving user interface and scripting.

> "Taking control of the flow of these essential processes and both clarifying and simplifying interactions for researchers will deliver significant benefits. Thanks Cerys for embracing the challenge and delivering a robust solution that does the job." - Carol Sandys

RSE Support of HPC Midlands+

As part of the University's membership of the HPC (High Performance Computing) Midlands+ Consortium, the University funds 50% of an RSE. Aston University does the same and purchase this time from ARC. The RSE(HPC) works collaboratively with research groups from across both universities for the definition, documentation, development and satisfactory completion of research software projects. A particular priority has been to engage with researchers at all development stages and disseminate best practices in the development of sustainable research software. For 2021 the dedicated RSE for Aston and Birmingham, was split across Dr. Simon Hartley and Dr. Simon Branford. For the first part of the year this was supporting those using Athena and, once Athena was decommissioned, this moved to supporting the Sulis.

Athena - 2017 to April 2021

Athena was a 512-node HPC Midlands+ cluster hosted in Loughborough, decommissioned in April 2021. In 2017 ***Athena*** was ranked 417th in the top 500 HPC systems achieving a Linpack Performance Rmax of **499.067 TFlop/s**. Support for Athena Included the tasks of allocation of resources, using the SAFE Tier 2 System, and supporting researchers in using the HPC system.

Final Athena Project Usage

In total we supported over 60 users from Birmingham and Aston doing calculations for projects ranging from fluid dynamics to investigations in blackhole astronomy. These users completed over 100,000 Jobs on the system using over 31,000,000 cpu/h.

Usage Summary

University	Raw Usage / CPUhs	Jobs (Total)
Aston	3,389,722.88	13,861
Birmingham	27,985,225.77	90,812
Totals	31,374,948.65	104,673

SULIS - From Summer 2021

Sulis is a new state of the art £3 million supercomputer which provides researchers with a higher capability system to run high throughput calculations. It's the first machine of its kind in the UK to offer a focus on enhanced ensemble computing workflows, addressing unmet needs in the research and innovation sector.

Sulis is currently in the first round of six-month projects. Initial impressions are favourable, but there are new challenges in this role when supporting a new system which has a total of 25,728 AMD EPYC compute cores and 90 NVIDIA A100 GPUs.

The detailed SULIS system specifications are:

Compute nodes

Dell PowerEdge R6525 compute nodes each with 2 x AMD EPYC 7742 (Rome) 2.25 GHz 64-core processors; 128 cores per node; 167 nodes; 21,376 compute cores; 512 GB DDR4-3200 RAM per node

GPU nodes

Dell PowerEdge R7525 **nodes each with 3 x NVIDIA A100 40 GB RAM passively-cooled GPUs**; 2 x AMD EPYC 7742 (Rome) 2.25 GHz 64-core processors per node; 30 nodes; 90 GPUs; 3,840 CPU cores; 512 GB DDR4-3200 RAM per node

High-memory nodes

Dell PowerEdge R6525 nodes each with 2 x AMD EPYC 7742 (Rome) 2.25 GHz 64-core processors; 128 cores per node; 4 nodes; 512 cores; **1 TB DDR4-3200 RAM per node**

RSEs Dedicated to Academic Areas

RSE work in the College of Arts and Law

The College of Arts and Law (CAL) funds a dedicated Senior RSE, Mike Allaway, in the Research Software Group.

Services Offered

In keeping with the RSG's service offerings, the main services Mike provides CAL are advice sessions, coaching engagements, and coding engagements, as detailed below. In addition to which, Mike also contributes significantly to the technical aspects of grant funding applications, resolves general enquiries relating to research software, and contributes to technical aspects of the College's strategy, such as how to utilise software and data to make the College's research more open.

As we found in 2020, the demand for advice sessions (face to face or on Zoom) was considerably lower during the COVID-enforced times of working from home. Coaching and coding demand does not seem to be similarly affected.

Detailed case studies of some of the work done with CAL researchers can be found later in this report.

Advice

- There have been 3 advice sessions for CAL in 2021 in 3 of the College's 5 schools, namely:
  - School of History and Cultures
  - Languages, Cultures, Art History and Music

  - English, Drama and Creative Studies

Coaching

- There have been 4 active coaching engagements for CAL in 2021 that have contributed to research across 3 of the College's 5 schools, namely:
  - Philosophy, Theology, and Religion
  - School of History and Cultures
  - Languages, Cultures, Art History and Music
- These coaching engagements have covered topics such as fundamentals of programming in Python, relational database design and development with SQL, and data analysis in Python.
- The researchers who have received this coaching range from PhD students to Professors.

Coding

- There have been 10 coding engagements worked on for CAL in 2021, 8 of which were completed by the dedicated CAL Senior RSE and 2 by other members of the RSG. These engagements have contributed to research across 4 of the College's 5 schools, namely:
  - Birmingham Law School
  - Philosophy, Theology, and Religion
  - English, Drama, and Creative Studies
  - Languages, Cultures, Art History and Music
- The outputs of these coding engagements have mostly been web applications that offer a broad range of functionality, such as collecting research data, presenting research data, promoting the work of research projects, and helping users to engage with research projects.

RSE work in the Centre for Computational Biology (CCB)

For much of 2021 the CCB funded 50% of a dedicated RSE, split across Dr Simon Hartley and Dr Ryan Pepper in the Research Software Group.

Ryan and Simon helped support the CCB in particular in the lead up to the start of the new teaching year. The CCB's research-led-teaching MSc programmes use the same BEAR services as for the CCB's BEAR-supported research work. Ryan and Simon provided specialised applications on BlueBEAR for the CCB's increasing number of taught students, an increase due to the expansion of the successful MSc Bioinformatics course with new Dubai and Distance Learning versions, and the Health Data Science MSc. They also provided a large amount of support to the staff and students of these programmes in the first few weeks of teaching, with Ryan presenting to the Edgbaston MSc students about getting set up on BEAR Portal and Simon running a lab session in the medical school library.

RSE support

Like the wider RSG, Simon and Ryan delivered advice, coaching and coding support to the CCB and wider research community within MDS and LES. The advice is usually personalised on topics such as: Creating pipelines on CaStLeS; Job dependencies; Array jobs; Using Python on CaStLeS; or Parallelisation of R.

Coaching sessions were usually bespoke and tailored to research groups, catering for up to five people, however most engagements were one on one. They covered topics such as: Python; Django; HTML5 and JavaScript development.

Training

Simon and Ryan both delivered Software Carpentry and NVidia CUDA training. Due to the pandemic, these have continued online. During the year, they taught the following courses, either to groups of up to 20 people or personalised for individuals or small groups:

- Introduction to Linux
- Introduction to BlueBEAR/CaStLeS
- Software Carpentries on Bash, Git, Python, R, and MATLAB
- Fundamentals of CUDA for C++

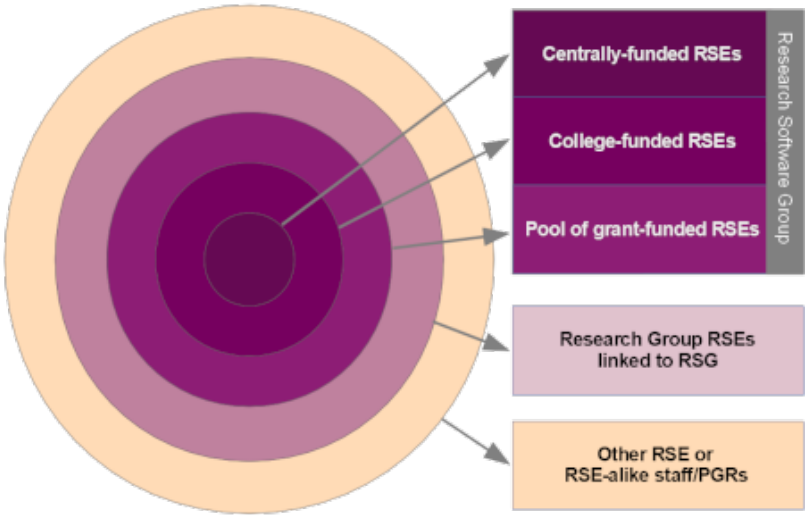
'Over the course of summer 2021 I was working against tight deadlines to develop the first module in the new distance learning Masters course in Bioinformatics. As part of this, we needed to develop an approach for automated grading of students' R code. We could not have implemented this successfully without Ryan's help. He installed the software on the Blue Bear systems and comprehensively tested it for us with the examples code we provided. One of the challenges with this was that I found it really difficult to follow the online documentation for the software. Then, once we got the software to work, the developers released a new version that required a completely different code specification for the grader! Thankfully, Ryan spotted that this was why things weren't working and patiently walked me through how to interpret the documentation and change the notebooks I had developed to match the new requirements. What I have really appreciated is his patience with me in learning how to use the new software - I was never made to feel like I was asking too much. Secondly, rather than going back and forth with messages within the IT ticket, Ryan proactively suggested we have a zoom call to go over it and resolve the issues much more efficiently. This definitely saved a lot of time and by working through it together on the call we were able to solve the problems more collaboratively. Thanks so much for all of the support!' - Dr Lindsey Compton, Lecturer in the CCB

RSG 2.0

RSEs on Funded Projects

The Research Software Group offers both free (centrally-funded) engagements and project-funded engagements. Funded engagements are usually longer than free ones, or involve more RSEs.

The idea is to allow a research group/project to pay for one or more RSEs with particular skills for a period of time. This could be a short project (e.g. two RSEs running a two-week workshop) or a long project (e.g. 50% FTE for 2 years). To achieve this we have a central pool of RSEs that can work on such projects without them, and their skills and experience, leaving the university when the project finishes.



Many RSEs will continue to operate in colleges, departments and research groups independently of the RSG. However, there are now three ways for RSEs to become part of the RSG to a greater or lesser extent. The diagram is described further below.

The diagram is comprised of five groups of RSEs:

- First the centrally funded RSEs who provide support for BEAR's main services and offer the centrally-funded BEAR Software Advice/Coding/Coaching services.
- Secondly are the college-funded RSEs who are line-managed in the RSG but work exclusively for the relevant college or academic unit.

- Thirdly the pool of grant-funded RSEs, who can be available for including in grant applications, and will grow over time as more RSEs are recruited to it. Please discuss your grant application with us before including a pool RSE in the proposal. See below for more information on including a BEAR RSE in grant applications.
- Fourthly, some research groups will continue to recruit and manage their own RSEs, but we are now offering the option of a formal 'dotted line' to the RSG to assist with managing the RSEs and allow them to be associated with peers with similar skills.
- Finally some RSEs will continue to work independently of the RSG, meeting up informally at events or in the Research Computing Community slack workspace.

Please discuss any grant applications with us before including a pool RSE in the proposal. See https://intranet.birmingham.ac.uk/bear-software/funded for more information on including a BEAR RSE in grant applications.

Baskerville

Baskerville Introduction

Baskerville, named after John Baskerville (28 January 1707 – 8 January 1775) a Birmingham industrialist, is a EPSRC funded tier 2 HPC system. The Baskerville consortium is University of Birmingham, Diamond Light Source, The Rosalind Franklin Institute, and The Alan Turing Institute; with the system having NVIDIA, Lenovo, and OCF as technology partners.

As is the case with all of ARC's compute and storage services, the deployment and delivery of Baskerville is a team effort, with all three groups in ARC playing their part. The Architecture, Infrastructure and Systems group led the design, delivery and deployment of Baskerville, with the RSG designing the identity management systems, managing the projects and users on the system, and installing and curating the software used by researchers to carry out their research.

The technological specifications are:

1. 46 SD650-N V2 liquid cooled compute trays

   –2x Intel® Xeon® Platinum 8360Y CPUs, each with 36 cores at 2.4GHz (with boost to 3.5GHz)
   –512GB RAM (16x 32GB DDR4)
   –1TB NVMe M.2 device (used for OS and available as /scratch-local)
   –1x 25GbE NVIDIA® Mellanox® (on-planar ConnectX-4 port)
   –1x HDR (200Gbps) NVIDIA Mellanox Infiniband port (ConnectX-6 PCIe gen4 adapter)
   –NVIDIA HGX-100 GPU planar: 4x NVIDIA A100 40GB GPGPUs - connected together with NVLink

2. ~5400 TB of storage (~5 PB of spinning disk storage and 0.5 PB of flash storage)

3. Lenovo's 'direct to node warm water' cooling.

There was a tight window between Baskerville being funded and ARC having the first pilot users on the system in June 2021. The hard work of many team members allowed this short turn around to be possible and this enabled us to host a DiRAC Hackathon and have the first round of EPSRC Access to HPC users on the system in early July.

Baskerville Testimonials

The following testimonials are from a wide range of users from different institutions describing their experience of using Baskerville:

Prof Matthew Foulkes applied to the EPSRC Access to HPC call for Baskerville resources for his FermiNet group. Wan Tong Lou, Gino Cassella and Halvard Sutterud all found accessing Baskerville very easy, following the instructions they to started using Baskerville on the day they were granted access. They used Baskerville to carry out the first quantum Monte Carlo calculation of the ground state of a continuum electronic system subject to periodic boundary conditions using their Fermionic Neural Network, or FermiNet. They believe that the work they have accomplished during the pilot phase will result in 3 separate publications. They found the Baskerville support team very helpful in dealing with a few tricky technical issues; both promptly and professionally.

'Access to Baskerville really let us supercharge our code development process. The powerful GPUs allowed us to carry out highly accurate quantum Monte Carlo calculations of large many-particle systems, which famously require both large memory and high precision. The large number of GPUs and good interconnect also allowed us to implement and test a multi-node MPI version of our code, allowing us to study much larger systems than would have been possible on a single node. To our knowledge, the work we have carried out using Baskerville would not have been possible using any other facility available in the UK.' - Halvard Sutterud

The Advanced Simulation and Modelling of Virtual Systems (ASiMoV) group is a project led by Prof Stephen Jarvis from the University of Birmingham, with collaboration from Dr Istvan Reguly and Dr Gihan Mudalige from the University of Warwick. They were some of the earliest adopters of Baskerville and found from the start that Baskerville was stable and easy to set up

their applications. They found the documentation clearly structured and it was easy to find what they needed. In the early days they found a few software or configuration issues, the team would often respond in minutes, and would go out of their way to help resolve them.

'The EPSRC Prosperity Partnership on Advanced Simulation and Modelling of Virtual Systems (ASiMoV) has been granted an allocation on Baskerville to assess the feasibility of running a production-grade industrial computational fluid dynamics simulation on a GPU accelerated system. Thanks to a DSL-based development approach, a number of optimisations and performance benchmarking on Baskerville, we estimate that a complex coupled unsteady simulation of a production-grade aero-engine compressor consisting of 10 blade rows and almost 5 billion nodes will complete a full revolution in 8 hours; a simulation that would currently take a similar sized classical CPU cluster system several days.' - EPSRC Prosperity Partnership on Advanced Simulation and Modelling of Virtual Systems (ASiMoV).

Dr Andrey Brukhno and Dr Michael Seaton (SCD/STFC) were granted access to Baskerville via Diamond light Source. They are currently in the process of accumulating data from atomistic and coarse-grain (DPD) simulations to write a paper focused on comparison between the outcomes from the simulations at the two resolutions / scales.

The Rosalind Franklin Institute have a number of research projects using Baskerville, including: The Pokéminos project, run in collaboration with CCP-EM, designed to simulate a simplified view of data collected by cryo electron tomography, and other data acquisition methods, so that classifier models like those derived from Variational Auto Encoders can be trained efficiently to automatically label real-world data during collection. DisTRaC (Distributed Transient Ram Ceph), a project designed to allow for temporary distributed parallel files systems and object stores to be spun up on RAM disks within the pool of compute nodes allocated to a cluster job. The Parakeet project provides a digital-twin to real world cryo electron microscopes, and serves to produce high-fidelity simulations of cryo electron tomography scans as they would be captured by real hardware. Within RFI's Structural Biology theme we are able to offload to Baskerville the execution of cryo electron microscopy reconstruction and refinement pipelines such as RELION and cryoSPARC. The RFI were delighted to contribute challenge data that was

used by the Turing Data Study Group, supported by ARC in collaboration with the Alan Turing Institute, where participants were tasked with processing and analysing data using the Baskerville cluster.

'The Baskerville documentation has been very useful, and serves as a robust starting point for our users there upon their account creation. This has largely been sufficient to get new users logged in for the first time on their own, lowering our support burden and empowering our science users.' - Dr Joss Whittle

---

PhD student Jordan Dorrell from the University of Birmingham found getting onto and using Baskerville very easy, especially with prior experience on BlueBEAR. The way the information was presented on Baskerville had a greater clarity and depth compared to what he had previously seen on BlueBEAR. Help from the support team they found very helpful particularly Simon Branford who helped identify CUDA and OpenMP problems that were killing the parallelisation efficiency of their script, which they doubt they would have solved by themselves. The facilities have enabled them to scale up their training sets for more accurate neural networks that would not have been possible on a machine with fewer/slower GPUs.

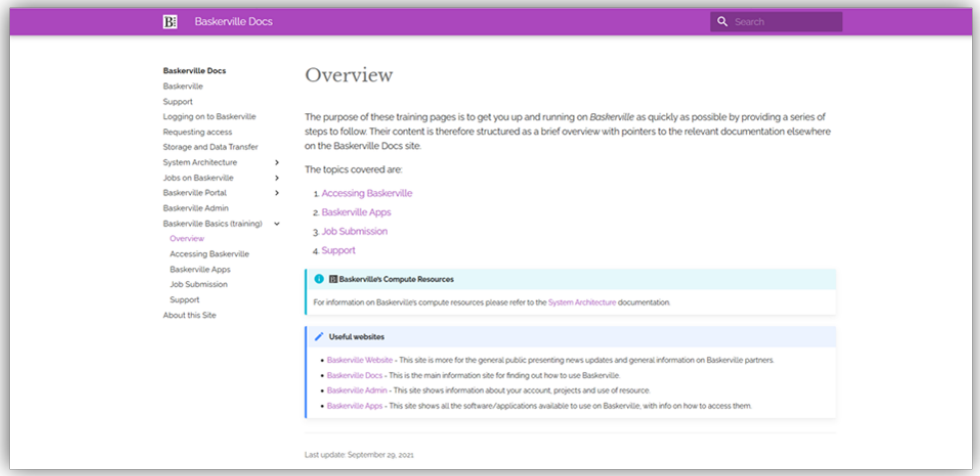
Baskerville Basics

Baskerville basics is an educational section on the docs.baskerville website (https://docs.baskerville.ac.uk):

1. It provides tutorials and educational information that is useful to someone who has prior HPC experience.
2. The information is GPU focused since Baskerville is a GPU-accelerated HPC system.
3. Users will have produced a submission script that they can refer to when running their own examples.
4. It outlines the methods researchers should use to get help and advice.
5. Combined with the general documentation on the rest of https://docs.baskerville.ac.uk, this should answer all the questions researchers might have when getting started with Baskerville.

Key features present in this tutorial include:

- How to access Baskerville
- Baskerville apps: the website, loading modules and the difference between test and live environments
- SLURM: sbatch, squeue and scancel
- Building a submission script, obtaining example CUDA code and operating it with a range of GPUs to see the effect
- Methods for file transfer



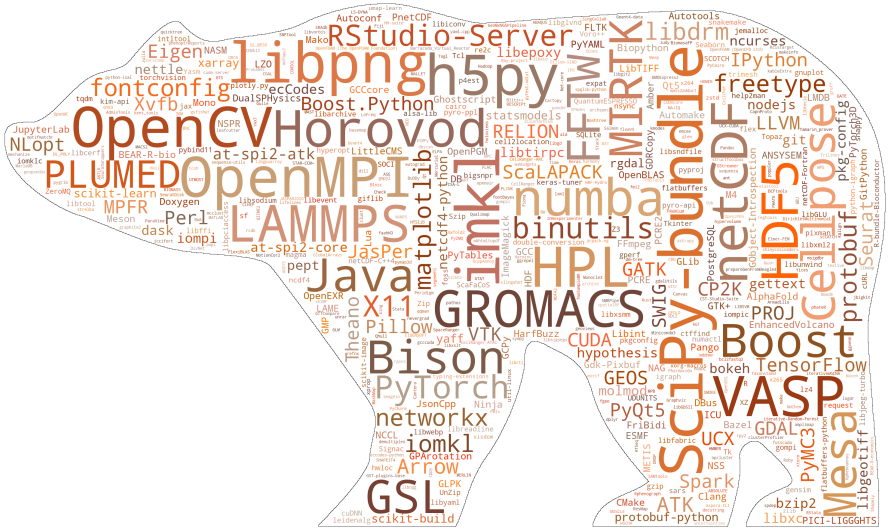
The tutorial is structured in such a way to maximise self-teaching, posing questions and pointing users to where they can find more information. In the pilot phase of Baskerville we sought researcher's opinions about Baskerville basics as well as the docs website. Users have found it easy to navigate as well as informative.

BEAR Support and Applications

BEAR Support

In 2021 the Research Software Group handled over 2000 Service Now tickets, covering a broad range of areas including BEAR Project admin requests, application installations (see separate section for further information), licensing queries, HPC-specific queries and other general support tickets.

This year again saw improvements to BEAR Admin (see separate section on automation), which is used in the processing of common tickets in the RSG (and Advanced Research Computing more broadly) and have helped to minimise human-error in many tasks.



Application Installation

We built **577** applications (counted by name only), or **1004** counting installed modules, therefore including versions and multiple toolchains. In total we now have **1188** applications (counted by name) across the BEAR systems.

Application installation requests (recorded via Service Now) during 2021: **208**

Intel's Ice Lake Nodes

2021 saw the addition of Intel's new *Ice Lake* CPUs to BlueBEAR. Due to the way we install applications, using EasyBuild, much of this progressed without issue, with a few applications requiring discussion with developers to support the new processors. In early 2022, we will be introducing new GPU Ice Lake nodes, which contain NVIDIA A30s and A100s.

BlueBEAR and BEAR/CaStLeS Cloud Operating System Upgrade

This was mentioned in last year's report but the process of performing the OS-upgrades was delayed due to significant changes to the CentOS support model. The end result of these changes meant that we ultimately upgraded to RHEL 8 (Redhat Enterprise Linux version 8) instead in Spring 2021. Much of the the preparatory work concerning reinstalling the applications was undertaken during the final few months of 2020, with some time in spring 2021 being spent fixing the remaining issues.

Singenbatch command

There were several commercial applications that were simply unable to run under RHEL 8 so as a stop-gap we provided a wrapper-script-based containerisation solution. The command, named singenbatch, behaves similarly to Slurm's sbatch command but wraps the contents of the submission script (apart from the #SBATCH headers) so that it runs within a CentOS 7 Singularity container. We are no longer widely supporting this solution as most application developers have now provided RHEL 8 compatible versions of their binaries (where we're unable to compile from source) – however, there are certain application versions that require ongoing support which are still unable to run under RHEL 8 and in these specific cases we are providing ongoing containerisation support.

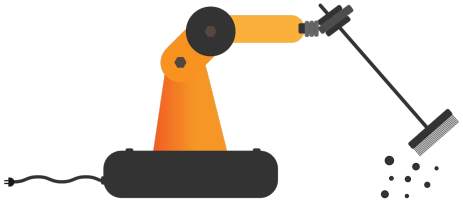
EasyBuild and Simon Branford

As of 2021, the Research Software Group's Deputy Leader Simon Branford is now an EasyBuild maintainer, whereby he regularly contributes to the project by facilitating with the processing of the community's large number of GitHub Pull Requests, alongside other strategic decision making. This is of huge benefit to the RSG as it ensures that we're fully abreast of any forthcoming updates, known issues etc.

Internal BEAR Systems

To support various BEAR services we have several systems and websites that rely heavily on automation to reduce the scope for human error and to save significant time for the team. In 2021 we continued our gradual improvement of these systems to allow them to more closely reflect the current ARC workflows.

Automated processes for BEAR and Baskerville



We maintain a database of users, projects, groups, etc. for the thousands of BEAR projects. When deploying Baskerville in 2021 we designed and deployed an equivalent system, learning lessons from BEAR. We are now in the process of retrofitting some of the new ideas from Baskerville back in to BEAR. All new user/project creations and modifications on BEAR and Baskerville are handled automatically, with any changes being pushed out to Slurm, LDAP, storage systems etc. with no human intervention. We began moving from a wholly-manual process to our now near-fully-automated process in 2017, and have virtually eliminated the errors and mistakes that manual processes are naturally prone to.

In 2021 we have tried to push further, and now run various processes each month that automate the task of identifying resources that are no longer used and can be removed (for example projects where the storage has been empty for a long time), or changed (for example downgrading virtual machines that are underutilised). The main way we do this is to interact with the University's IT Service Desk using it's ServiceNow API - and automatically create tickets to communicate such changes with users of our services. The process of automatically creating tens of tickets each month saves a considerable amount of person-time, and simultaneously improves the quality of the tickets we create.

We also now automatically email the project owner of every project twice a year, making sure the information about the project is still correct and encouraging them to re-register the project if required using a self-service system. All parts of this process are handled by ARC's systems, with no ARC person-time required unless the project owner wants to ask questions. In 2022 we are intending to introduce more self-service features, eliminating more manual tasks for ARC staff. This kind of automation allows the researcher to have a faster result with a reduced risk of errors - when compared to a process involving a human.

Continuous Integration

We use git to store all our code and scripts and much of our documentation. Wherever possible we then add continuous integration, which automatically runs various tasks whenever changes are pushed to either GitLab or GitHub. These tasks include:

- Static code analysis - 'linting'. This is absolutely invaluable to us - enforcing various coding standards (in Python, R, bash, Markdown, and others), while simultaneously frequently identifying real bugs in the code. For example where a variable is used before being defined, perhaps in a rarely used code path.
- Unit testing - We have a lot of unit tests, and measure the code coverage of them. While we don't try for 100% coverage, due to the often diminishing returns of striving for the final few percent, we still try to achieve a high level of coverage where possible. The continuous integration pipelines run the unit tests automatically whenever code is pushed to the repository.
- PDF generation - Some of our work involved writing documents. Often (for example for this annual report) we write the documents in Markdown, and automatically generate PDFs of them using continuous integration.
- Website generation - Some websites we have written are generated automatically using continuous integration - for example for hosting on **GitHub Pages**.

Training

Carpentries

The Carpentries is a fiscally sponsored project of Community Initiatives, a registered 501(c)3 non-profit organisation based in California, USA. The Carpentries teaches foundational coding, and data science skills to researchers. Over the course of this year, we have delivered training workshops covering Python, R, and MATLAB programming languages, and version control with Git. Due to the pandemic, this training has been delivered over Zoom rather than in the classroom, and we have invested time over the last year reviewing our delivery of the curriculum to adapt to the resulting challenges.

In addition to the Carpentries instructors within ARC, we now have a network of eleven Carpentries instructors across campus. This allows us to run many more courses than we could with just our internal instructors. In 2021, we have run 17 Carpentries courses with approximately 200 attendees overall.

Introduction to BlueBEAR

Previously titled *BEAR Necessities*, the *Introduction to BlueBEAR* course is presented by a member of the Research Software Group and runs roughly once a month during the University's term time. It's a half-day course that introduces participants to the concepts of supercomputers and batch workflows and then provides specific info on how to run jobs on BlueBEAR. It also covers interactive jobs and GUI jobs using *BEAR Portal*, as well as additional info about BEAR's other services.

RSE Provided Nvidia Deep Learning Training

As part of the University's membership of the NVIDIA Deep Learning Institute (DLI) Ambassador Program the RSG has certified instructors for several NVIDIA courses:

- Fundamentals of Accelerated Computing with CUDA C/C++
- Fundamentals of Deep Learning
- Accelerating CUDA C++ Applications with Multiple GPUs
- Accelerating Data Engineering Pipelines

In 2022 we will be expanding the number of instructors we have to three and will increase our number of courses offered.

The details of the GPU systems the university researchers now have access to are:

BlueBEAR

bbgpu GPU nodes

Intel Broadwell **nodes each with 1 x NVIDIA GP100GL Tesla P100 16 GB RAM**; 2 x Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz 10-core processors per node; 128 GB DDR4-2133 RAM per node

castles GPU nodes

Intel Broadwell **node each with 2 x NVIDIA GP100GL Tesla P100 16 GB RAM**; 2 x Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz 10-core processors per node; 128 GB DDR4-2133 RAM per node

castlespowergpu and *bbpowergpu* GPU nodes

IBM® Power System AC922 **nodes each with 4 x NVIDIA V100 16 GB RAM**; 2 x IBM® Power9® 8335-GTX CPUs 3.8GHz 18-core processors per node; 1024 GB DDR4-2666 RAM per node

Forthcoming in early 2022

Intel IceLake based nodes, with 2 x NVIDIA A100 or 2 x NVIDIA A30

Baskerville

Lenovo ThinkSystem SD650-N **nodes each with 4 x NVIDIA A100 40 GB RAM**; 2 x Intel® Xeon® Platinum 8360Y CPUs 2.4GHz 36-core processors per node; 512 GB DDR4-3200 RAM per node

SULIS

Dell PowerEdge R7525 **nodes each with 3 x NVIDIA A100 40 GB RAM**; 2 x AMD® EPYC® 7742 (Rome) 2.25 GHz 64-core processors per node; 512 GB DDR4-3200 RAM per node

Community, conferences and meetings

RSE involvement in community and technical groups, conferences, and meetings during 2021

- Afraz and Ed attended the monthly ExCALIBUR H&ES Technical Working Group
- Ed attended the HPC Midlands Plus management board meetings for Sulis
- Ed attended the Baskerville consortium management board meetings
- Ed is part of the Society of RSE Mentoring Programme
- Simon B. is part of the Tier 2 Technical Working Group
- Simon B. hosted the Sixth EasyBuild Users Meeting and several EasyBuild Tech Talks
- At CIUK, Ed presented **Installing and curating software for heterogeneous compute environments**
- Ryan contributed to The Carpentries syllabus discussion meetings
- At SeptembRSE, Ryan was on the panel **Industrial Software Development Practices: Do they work in Academia? Should RSEs learn and use them?**
- During SeptembRSE there was discussion on regional RSE groups and we agreed to try to set up 'RSE Midlands', a SIG of the Society aiming to hold an in-person one-day event in early 2022.
- At the Sixth EasyBuild Users Meeting, Simon B. presented **EasyBuild at the University of Birmingham**
- Ed, with Carol and Simon T., presented all about BEAR and Ed presented on the difficulties recruiting and retaining staff to RUGRIT
- Ed, James, and Gavin attended CIUK, representing Baskerville in the CIUK Research Zone.
- Several members of the team attended the Baskerville launch event
- Ed attended various SeptembRSE sessions, including a national RSE leaders meeting.
- Cerys attended DjangoCon

Contact Us

BEAR Software

For information about BEAR Software, see https://www.birmingham.ac.uk/bear-software

For any help with anything related to research software at the University of Birmingham please email bear-software@contacts.bham.ac.uk

BEAR / Advanced Research Computing

For information about BEAR, see https://www.birmingham.ac.uk/bear

For general help or information about any BEAR service (BlueBEAR HPC, Research Data Store, ...) contact the BEAR team by email at bearinfo@contacts.bham.ac.uk

Follow the Advanced Research Computing's twitter feed @uob_bear at https://twitter.com/uob_bear

Join our Research Computing Community slack workspace at https://bham-rse.slack.com/signup

Requests, Faults, Complaints

The IT Service Desk is your route to find answers. Find all Advanced Research Computing items here: https://intranet.birmingham.ac.uk/bear/sd/bear

For all other IT Services items/help or to log faults and any complaints visit: https://www.itservicedesk.bham.ac.uk/