UK researchers and large-scale GPU HPC systems

Andy Turner, EPCC, University of Edinburgh

a.turner@epcc.ed.ac.uk

epcc

Overview

- UK HPC landscape
  - National Compute resources
  - UK Next National Supercomputing Service
  - AI Research Resource
- Results of GPU preparedness survey
- Environmental sustainability for HPC in the UK

Acknowledgements: UKRI Digital Research Infrastructure team provided information on UK roadmap.

UK HPC Landscape

UK planned ecosystem

UK-NNSS	Large GPU-based modelling and simulation system Focussed on strategic national goals
NCR	Selection of HPC architectures Combine to support a wide range of HPC use cases
AIRR	Range of GPU-based systems Designed to support AI requirements

UK National Services

Single access portal

Federated access support

EuroHPC Joint Undertaking

University-level HPC Resources

Other HPC Services

Next UK National Supercomputer (UK-NNSS)

Target mission-driven research and support strategic national goals, with a focus on grand challenges and the ability to conduct computational research at very large-scale.

- Specifically targeting modelling and simulation rather than AI workloads
- Based at EPCC, University of Edinburgh
- Investment of up to £750m for:
  - Hardware: large-scale GPU system
  - Host site and operating costs
  - Service provision
  - Software programme
- Service currently planned to go live in Q4 2027:
  - Construction started on data centre upgrade in 2026
  - Hardware procurement being finalised
  - Service will operate for a minimum of 5 years

Work begins on UK's new £750m supercomputer - https://www.bbc.co.uk/news/articles/c39yr3p8m4xo

National Compute Resources (NCR)

NCRs will complement flagship investments, including UK-NNSS and AIRR, as a portfolio of diverse technologies available to the UK's research and innovation communities. Partially available now.

- CPU-based systems
  - **Cirrus NCR (EPCC, University of Edinburgh, HPE):** 640 compute nodes consisting of 2 x AMD EPYC 9845; 184,320 cores; 200Gbps interconnect https://www.cirrus.ac.uk/
  - **Charger NCR (UCL/DataVita, HPE):** 148 compute nodes consisting of Intel Xeon CPUs; 296 processors; 37888 cores; 400Gb/s interconnect; 8 Blackwell RTX Pro 6000 visualisation GPU
  - **Isambard 3 NCR (University of Bristol, HPE):** 380 Nodes consisting of NVIDIA Grace-Grace CPUs; 54,720 cores; 200Gb/s interconnect https://docs.isambard.ac.uk/
- GPU-based systems
  - **Baskerville NCR (University of Birmingham, Lenovo):** 22 nodes with 8x NVIDIA B200 GPU (176 GPU); 400Gb/s interconnect https://www.baskerville.ac.uk/
  - **Zenith NCR (University of Cambridge, Dell):** 28 nodes with 8x AMD MI355X GPUs (224 total); 400 Gb/s interconnect (shared resource with AIRR)
  - **Mary Coombs NCR (Daresbury, Lenovo):** 157 nodes with 4x NVIDIA H100 GPUs (628 GPU), Infiniband NDR200 interconnect https://www.hartree.stfc.ac.uk/technologies/mary-coombs/

AI Research Resource (AIRR)

The AI Research Resource (AIRR) is a suite of advanced computers that provides world-leading, AI-specialised computing capacity to public researchers, academia, small and medium size enterprises. Available now.

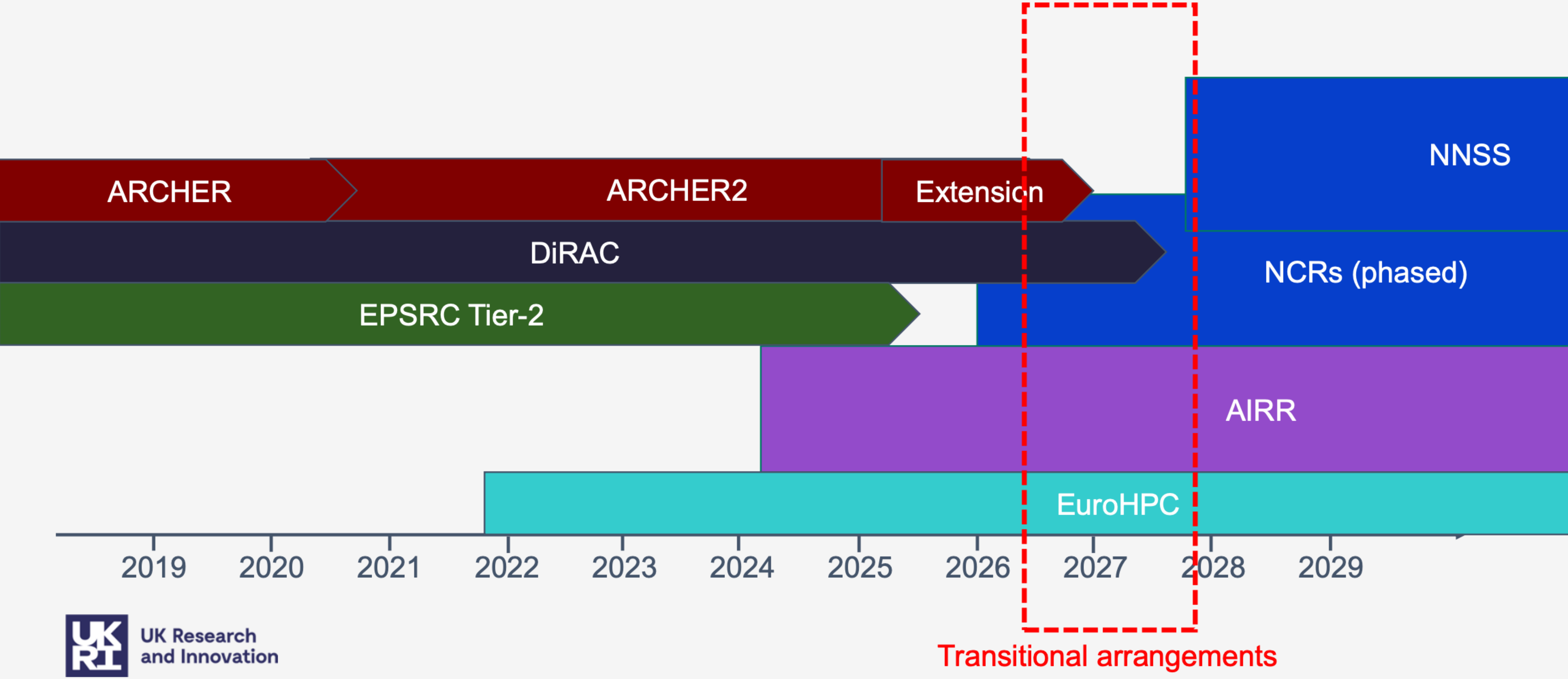
Systems:

- Isambard AI (University of Bristol, HPE):
  - 21 exaflops of AI compute, 5,448 NVIDIA GH200 superchips in total, 200 Gb/s interconnect
- DAWN (University of Cambridge, Dell)
  - 256 Dell PowerEdge XE9640 server nodes with 2x Intel Xeon Platinum CPUs and 3x Intel Data Centre GPU Max 1550
- Zenith (coming soon) (University of Cambridge, Dell):
  - 28 nodes with 8x AMD MI355X GPUs (224 total); 400 Gb/s interconnect (shared resource with NCR)

https://www.gov.uk/government/publications/ai-research-resource

UK R&I compute provision: a time of transition

OFFICIAL



UK Research and Innovation

HPC Software GPU preparedness

HPC science area GPU readiness

Research application area	Software examples	GPU Readiness 2023	GPU Readiness 2026
Electronic structure of materials	VASP, CP2K, Quantum Espresso	LIME	LIME
Modelling turbulence and combustion	OpenFOAM, Xcompact3d, Nektar++	ORANGE	AMBER
Biomolecular simulation	GROMACS, NAMD	GREEN	GREEN
Classical materials modelling	LAMMPS	GREEN	GREEN
Modelling fusion technology and processes	EPOCH, GENE, Smilei, BOUT++	AMBER	LIME
Complex inorganic materials	ORCA, ChemShell	AMBER	AMBER
Modelling complex multiscale systems	FEniCS, HemeLB	GREEN	GREEN
Lattice Quantum Chromodynamics	Grid	GREEN	GREEN
Cosmology/astrophysics	SWIFT, PROMPI	RED	AMBER
Solar system/planet formation	LARE2D/LARE3D	RED	ORANGE

RED – No GPU implementation available for any part of the HPC software requirements.
ORANGE – Some early development GPU implementation available for part of HPC software requirements - major development still needed to make it production ready.
AMBER – Partial GPU implementation of HPC software requirements available - development still needed to make it production ready.
LIME -- GPU implementation of HPC software requirements available - minor development still needed to make it production ready.
GREEN – GPU implementation of HPC software requirements already available for production research with a reasonable coverage of software functionality.

Other research area readiness

- UK-NNSS and NCR available across all research areas – not just aimed at science domains
- Readiness of other domains to use such systems has not been formally assessed in the same way
  - More uncertainty on what needs to be done and where
  - Less links between the communities and HPC RSEs
- Some early success in agent-based modelling software development (https://flamegpu.com/)
- Work ongoing via the Living Benchmarks project to engage with wider research domains

GPU software development

ExCALIBUR 10

UK research programme that ran from 2018-2025 and to deliver the next generation of high-performance simulation software for the highest-priority fields in UK research

https://excalibur.ac.uk/

archer2

GPU eCSE Programme *Funding to develop simulation and modelling codes to support use of GPU-based architectures in a sustainable way. (No more calls at the moment, projects still running.)*

UKRI Living Benchmarks

- *Support capital investments in digital research infrastructure (DRI) to ensure they meet community needs and provide best value for money.*
- *Develop a sustainable community of benchmarking experts*
- *Develop an open source, sustainable set of living benchmarks*

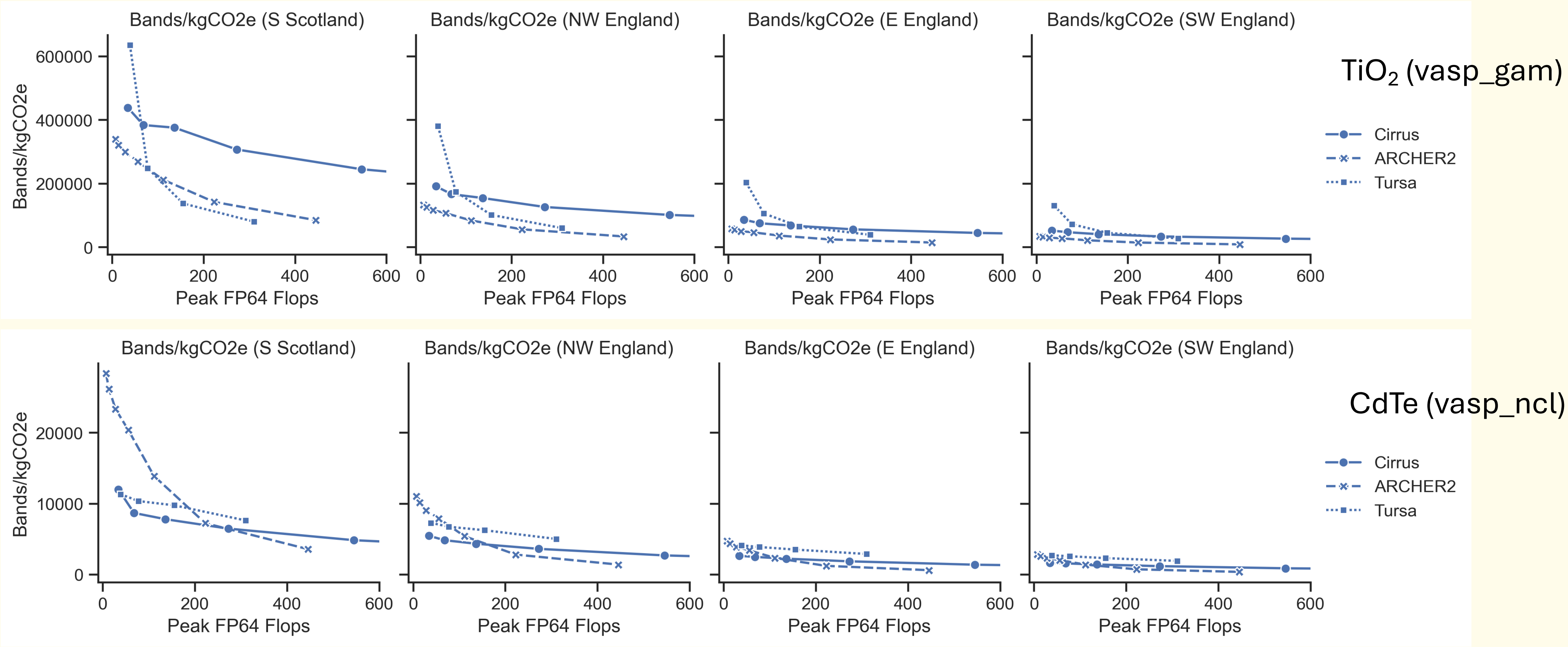
HPC environmental sustainability

Lifetime emissions estimates

	ARCHER2	Cirrus	Tursa
Approx. power draw (kW)	2,600	295	215
Carbon intensity (gCO_2e/kWh)	19.5	19.5	19.5
Lifetime	6 years	6 years	6 years
Operational emissions ($kgCO_2e$)	3,224,398	365,845	266,633
Embodied emissions ($kgCO_2e$)	4,400,000	405,000	420,000
Lifetime emissions ($kgCO_2e$)	7,624,398	770,845	686,633
Operational : Embodied	42% : 58%	47% : 53%	39% : 61%

- For all systems, the embodied emissions are a little larger than the operational emissions

Emissions efficiency by location



Carbon Intensity

0.0195 kgCO$_2$e/kWh | 0.0540 kgCO$_2$e/kWh | 0.1285 kgCO$_2$e/kWh | 0.2180 kgCO$_2$e/kWh

Markers correspond to nodes (1, 2, 4, 8, ..., 64) or GPUs (4, 8, 16, 32)

<table>
<tr><th></th><th>Embodied emissions dominate</th><th>Operational emissions dominate</th></tr>
<tr><td rowspan="1">Vendors</td><td colspan="2">• Responsible marketing to encourage reduction of consumption
• Responsible manufacturing
• Provide affordable support for long service lifetimes</td></tr>
<tr><td rowspan="2">Service Operators</td><td>• Run services for as long as possible
• Optimise service for performance</td><td>• Optimise system for energy efficiency
• Enable carbon aware scheduling on services</td></tr>
<tr><td colspan="2">• Purchase minimum possible amount of hardware
• Make emissions assessment a key part of system procurement
• Site HPC systems in locations with low carbon intensity
• Policies and application processes to support/encourage emissions-efficiency</td></tr>
<tr><td rowspan="2">RSEs</td><td>• Optimise software for maximum performance
• Algorithms to exploit new hardware innovations</td><td>• Optimise software for energy efficiency</td></tr>
<tr><td colspan="2">• Training for users on how to maximise emissions efficiency
• Embed emissions assessment into benchmarking and performance analysis</td></tr>
<tr><td rowspan="2">Users</td><td>• Work to maximise performance</td><td>• Work to maximise energy efficiency</td></tr>
<tr><td colspan="2">• Use the minimum amount of resource possible for the proposed work
• Make datasets FAIR to avoid replication and maximise re-use
• Produce a responsible computing plan for use of HPC systems</td></tr>
</table>

Responsible Computing Plan

Plan to minimise resource use

- What is the minimum resource needed?
- Do I need to test at smaller scale first?
- Would a simpler method be sufficient?
- Am I planning duplicate runs that are not really needed?

Plan data storage

- Do we really need to keep all intermediate files?
- Does each collaborator need their own copy of the dataset?
- Can rarely used data be compressed or moved to archival storage?
- Have we planned time to review and tidy data during the project?

Choose resources carefully

- Can I use shared or existing resources instead of buying new hardware?
- Is this workload better suited to HPC, cloud, or local compute?
- Do I have access to locations or services with lower-carbon electricity?
- If I buy hardware, do I have a credible plan for reuse?

Run software carefully

- Have I tested this on a smaller job first?
- Am I requesting more compute or memory than I really need?
- Could this run at a lower-carbon time or in a lower-carbon location?
- Am I recording enough information to avoid rerunning this work unnecessarily?

Aim for largest impact

- Is this code run often enough for optimisation to matter?
- Is this used by other people as well as me?
- Have I identified the actual bottleneck?
- Would specialist support help me make a bigger improvement more quickly?

Finish projects well

- Which files genuinely need to be retained?
- What can be deleted now?
- Could publishing this data or code reduce duplicated work elsewhere?
- Have I left enough documentation for others to reuse what has been done?

UK environmentally-sustainable DRI

- Training:
  - Green Software Use on HPC (Carpentries Incubator, HPC Carpentry)
  - SparkHub training on sustainable DRI
  - Imperial College sustainable DRI course
- Projects/resources:
  - NetDRIVE – UK Net Zero Digital Research Infrastructure project
  - Green DiSC – Digital sustainability certification
  - Digital Humanities Climate Coalition Toolkit – Advice on improving environmental sustainability for researchers (not just humanities)
- Communities:
  - Environmentally Sustainable Computational Science (ESCS) Forum
  - GreenRSE SIG within UK Society of Research Software Engineering

Future look

- UK-NNSS funding includes a large-scale software development programme
  - Not yet clear on timescales or form this will take
- UK Government funding small number of National Supercomputing Centres
  - Provide core funding to support long-term software development and retention of core skills within the community
  - EPCC announced as the first UK National Supercomputing Centre in 2025
- UKRI developing plans for Community Centres of Excellence (CCEs)
  - Support development of software with links to domain-specific expertise and links to specific research communities
  - Will interface directly with National Supercomputing Centres and national facilities (UK-NNSS, NCRs and AIRR)

GPU software development

UK research programme that ran from 2018-2025 and to deliver the next generation of high-performance simulation software for the highest-priority fields in UK research

https://excalibur.ac.uk/

1. High priority use cases
    - Delivered software development for pre-identified objectives of strategic importance for UK research and development
2. Emerging requirements for high performance algorithms
    - Enabled software with exascale-ready algorithms in communities that have not been traditional users of supercomputing
3. Hardware and enabling software
    - Provided access to novel computing hardware and software for research software developers to evaluate new approaches and algorithms
4. Cross-cutting research
    - The pan-programme work to integrate knowledge between different research and application domains that enabled faster progress and development efficient
5. RSE knowledge integration
    - Developed skills and knowledge within the UK RSE community to support next generation supercomputing systems

ARCHER2 GPU eCSE Programme

Funding to develop simulation and modelling codes to support use of GPU-based architectures in a sustainable way.

- Transforming existing software that targets CPU architectures into software that can exploit GPU-based architectures
- Implementation of algorithmic improvements within existing GPU software
- Improving the portability of GPU software
- Improving the performance of software running on GPU-based architectures
- Improving the scalability of software to enable effective use of an increased number of GPUs
- Improving GPU-based software to enhance sustainability and maintainability
- Improvements to GPU-based software to allow new research to be carried out
- Adding new functionalities to existing GPU-based software or enhancing existing GPU software for use in new research areas/workflows

https://www.archer2.ac.uk/ecse/reports/

archer2

UKRI Living Benchmarks

- *Support capital investments in digital research infrastructure (DRI) to ensure they meet community needs and provide best value for money.*
- *Develop a sustainable community of benchmarking experts*
- *Develop an open source, sustainable set of living benchmarks*

Developed open-source GPU-focussed benchmarks to support UK-NNSS procurement:

- Applications: Grid (QCD), CP2K (materials), DOLFINx (finite element), LAMMPS (materials
- Synthetics: covering memory floating point, IO, interconnect
- https://github.com/orgs/UKNNSS-Benchmarks/repositories

Funded work to develop GPU benchmarks focussed on UK use cases:

- Flame GPU extension to support AMD GPU: https://flamegpu.com/
- NEMO GPU benchmark for UK use cases: https://github.com/ukri-bench/benchmark-nemo
- Quantum Espresso GPU benchmark: https://github.com/ukri-bench/benchmark-quantumespresso

Working to integrate frameworks such as Spack, ReFrame/Ramble to support automated running and performance evaluation.

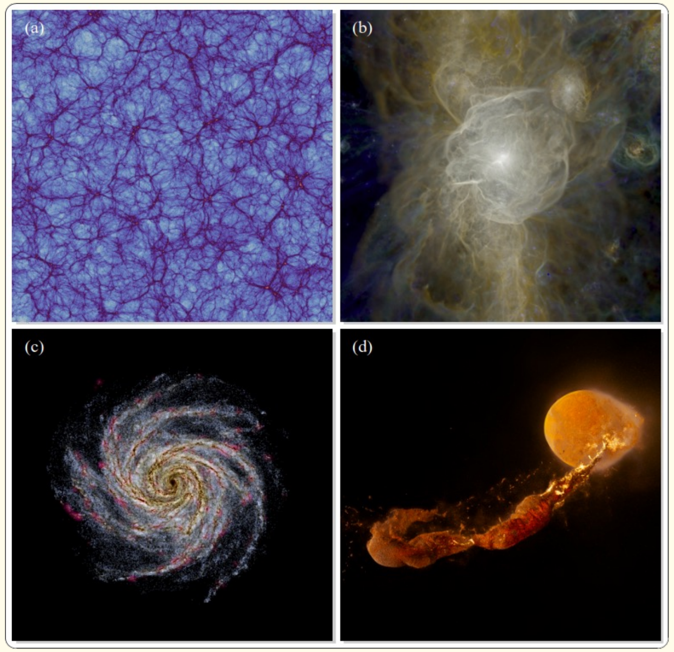
GPU eCSE examples

GPU-SWIFT

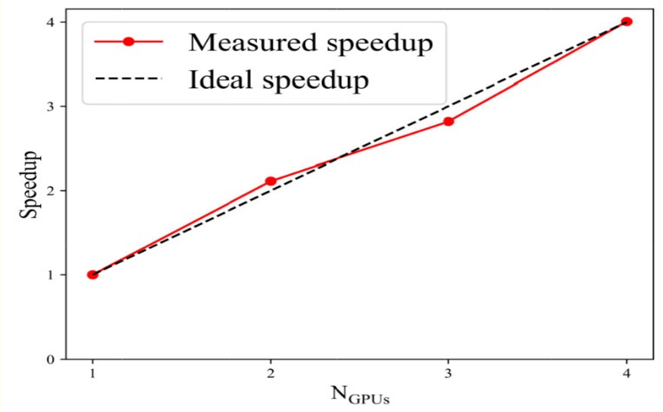
- Massively parallel hybrid HPC software package for cosmological & astrophysical simulations
- Starting point: Proof of concept GPU-acceleration of hydrodynamics solver

What GPU-eCSE programme enabled:

- Improvements in algorithms and optimisation (Nasar et al. 2026), leading to 2-4x speedup and ~30% improvement in energy efficiency.
- Current WIP leading to additional ~2x speedup
- Complete refactor of GPU-offloading mechanism, further improving performance and preparing it for future maintainability and portability
- Investigation of optimal memory storage strategies for particle hydrodynamics, leading to further >25% speedup (Ivkovic et al., in prep)



Schaller et al. 2024, swiftsim.com



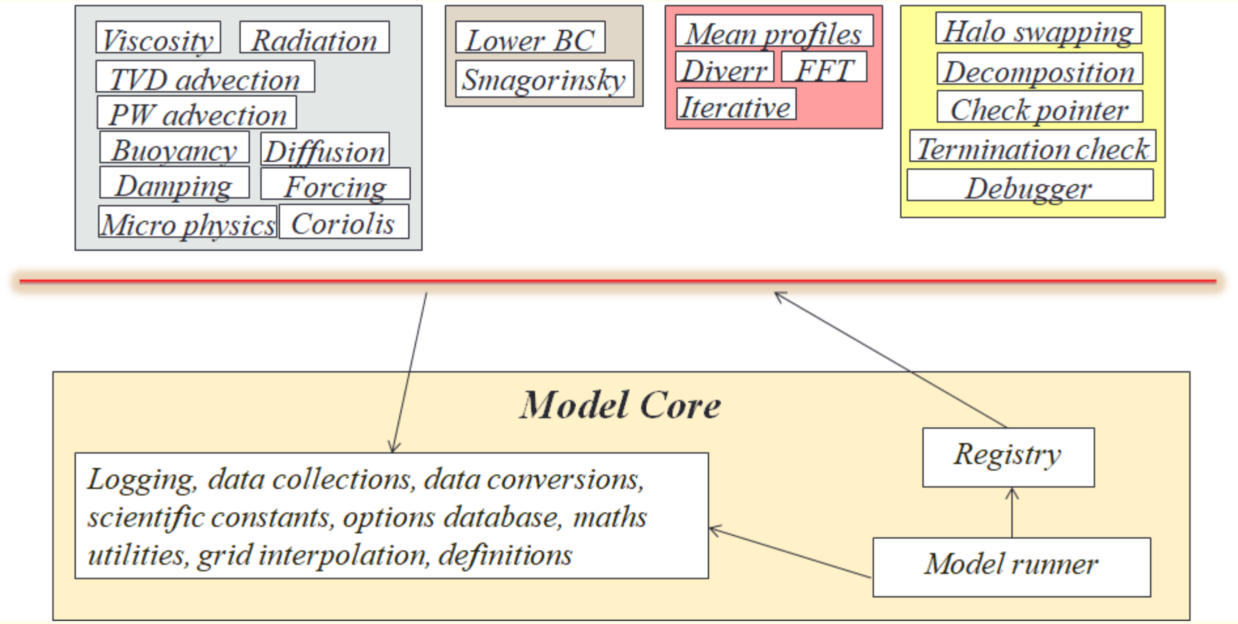
Strong scaling using 5123 particles on up to 4 Grace Hoppers. (Nasar et al. 2026)

MONC: GAINING GREATER INSIGHTS INTO CLOUDS AND TURBULENT PROCESSES VIA GPUS

Nick Brown, EPCC at University of Edinburgh
Christopher Day, EPCC at University of Edinburgh
Steven Boeing & Mark Richardson, University of Leeds
Todd Jones, University of Reading

Met Office NERC Cloud model (MONC)

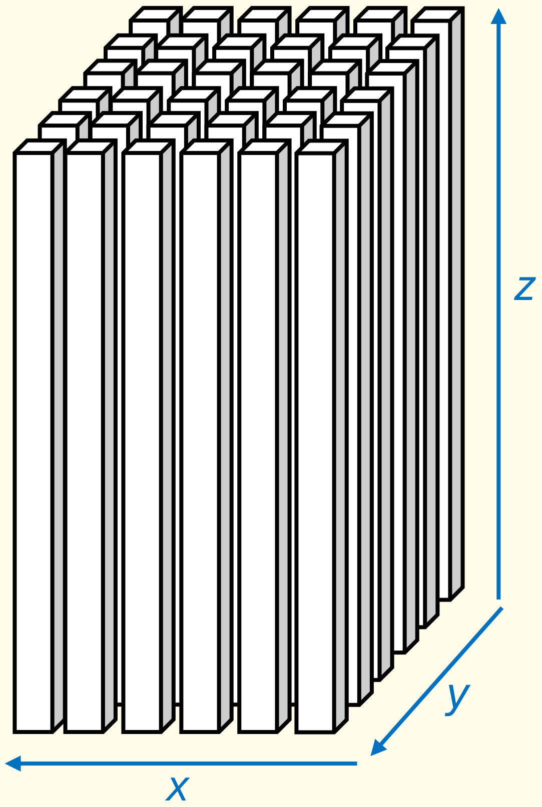
- A redevelopment of a model from the early 1990s called the LEM
- Been operational for around 11 years
- Used to model atmosphere at high resolution
  - Turbulence parameterisation, boundary layer flows, convection, fog, driver for sub models, developing parameterisation schemes, response of ice clouds to aircraft emissions



- Architected as independent components which plug in and out, with a central core that provides marshalling and control, as well we utility functionality
  - A specific run is based around a set of components being enabled
  - Community has grown up around this

Adopting OpenMP

- MONC is used on a range of HPC machines, so portability is very important
  - In this work we are focussing on using OpenMP *target* to drive the GPU offload




```
!$OMP target enter data map(alloc: su, sv, sw, sth, sq)
……
!$OMP target enter data map(to: u, v, w, th, q)
……
!$OMP target teams distribute parallel do collapse(2) nowait
do x = local_domain_start_index(X_INDEX), local_domain_end_index(X_INDEX)
    do y = local_domain_start_index(Y_INDEX), local_domain_end_index(Y_INDEX)
          call advect_scalar_field(y, x, dtm, u,&
            v, w, th, sth, global_grid,&
            local_grid, parallel, halo_column, field_stepping)
    end do
end do
!$OMP end target teams distribute parallel do
……
#pragma omp taskwait

!$OMP target exit data map(from: su, sv, sw, sth, sq)
```

- Beneath this there are 11 Fortran subroutines
  - Copy static data on once, avoidance of global/shared variables, pulling out MPI communications, expanding dimensions of flux arrays so each GPU thread is writing to its own memory

AMD Instinct MI210 GPUs are.... a bit slow

	ARCHER2 CPU (s)	ARCHER2 GPU no USM (s)	ARCHER2 GPU with USM (s)
First transfer	-	0.233	-
First compute	-	0.053	0.355
First total	-	**0.286**	**0.355**
Subsequent transfer	-	0.00002	-
Subsequent compute	0.006	0.042	0.103
Subsequent total	**0.006**	**0.042**	**0.103**

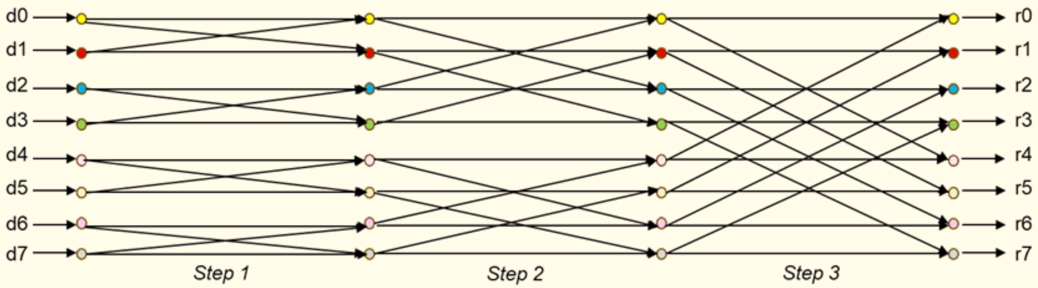
- Full status cloud testcase over three MPI processes, one process per GPU (so three GPUs in total). Time is for a TVD advection over *th* for a timestep.
- The AMD toolchain on ARCHER2 is also a pain as it doesn't handle asynchronous data transfers for derived types, so these need to be manually unpacked/repacked which adds to boilerplate code

Nvidia Grace Hopper (GH200) is much faster

	GH200 CPU (s)	GH200 GPU separate memory (s)	GH200 GPU unified memory (s)
First transfer	-	0.336	-
First compute	-	0.006	0.017
First total	-	**0.286**	**0.017**
Subsequent transfer	-	0.0008	-
Subsequent compute	0.0020	0.0005	0.0004
Subsequent total	**0.0020**	**0.0013**	**0.0004**

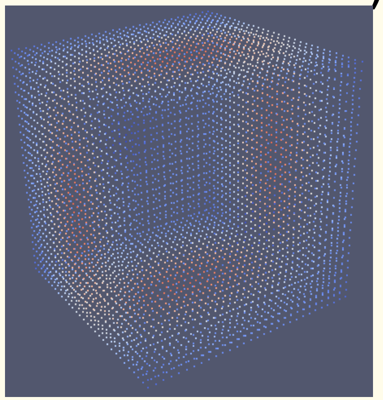
- Full status cloud testcase over three MPI processes, three process per GPU (so one GPU in total). Time is for a TVD advection over *th* for a timestep.
- Using nvfortran 16.1, toolchain is newer and more powerful (we don't have the same issues as with the AMD toolchain)

Next steps....

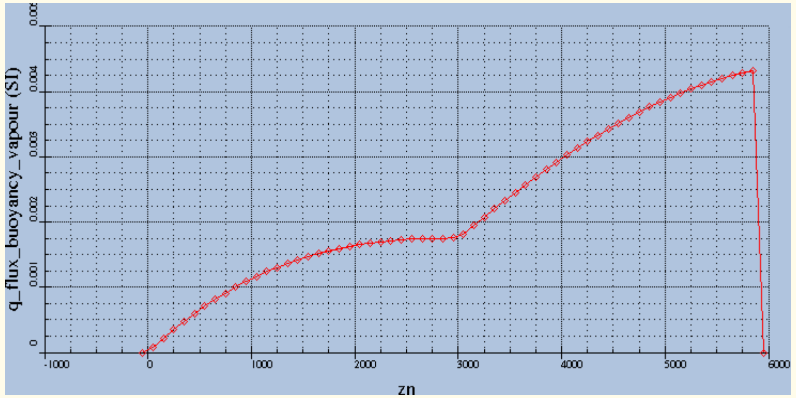


- Have started looking at FFTs
  - FFT GPU libraries, so in theory can work with off the shelf components, but this forces us to place more emphasis on the communication aspect (TVD advection has some halo-swapping but we currently drive this via MPI from the host)

- We need to make IO GPU aware
  - The other piece of the puzzle is transforming raw computational data into higher level diagnostics
  - TVD advection calculates diagnostic values which are often useful for reporting



Prognostics



Diagnostics

- Are currently ignoring these, we can copy back to the host and handle as usual, but think that we might be able to optimise this and are looking at integrating ADIOS2 here are part of that