

Beating data bottlenecks in weather and climate science

Bryan N. Lawrence^{*†‡}, Julian M. Kunkel[‡], Jonathan Churchill[§], Neil Massey[¶], Philip Kershaw[¶], Matt Pritchard[¶]
^{*}National Centre for Atmospheric Science, [†]Department of Meteorology, [‡]Department of Computer Science
 University of Reading, Reading, U.K.
[§]Scientific Computing Department, [¶]Centre for Environmental Data Analysis, RALSpace
 STFC Rutherford Appleton Laboratory, Didcot, U.K.
 Corresponding Author: *bryan.lawrence@ncas.ac.uk*

Abstract—The data volumes produced by simulation and observation are large, and growing rapidly. In the case of simulation, plans for future modelling programmes require complicated orchestration of data, and anticipate large user communities. “Download and work at home” is no longer practical for many use-cases. In the case of simulation, these issues are exacerbated by users who want simulation data at grid point resolution instead of at the resolution resolved by the mathematics, and/or who design numerical experiments without knowledge of the storage costs.

There is no simple solution to these problems: user education, smarter compression, and better use of tiered storage and smarter workflows are all necessary – but far from sufficient. In this paper, we introduce two approaches to addressing (some) of these data bottlenecks: dedicated data analysis platforms, and smarter storage software. We provide a brief introduction to the JASMIN data storage and analysis facility, and some of the storage tools and approaches being developed by the ESIWACE project. In doing so, we describe some of our observations of real world data handling problems at scale, from the generic performance of file systems to the difficulty of optimising both volume stored and performance of workflows. We use these examples to motivate the two-pronged approach of smarter hardware and smarter software – but recognise that data bottlenecks may yet limit the aspirations of our science.

Index Terms—HPC, exascale, big data, extreme data, POSIX, object store, NetCDF

I. INTRODUCTION

Weather and climate science exploit vast amounts of observational data and generate vast amounts of simulation data. Data volumes and velocity are increasing rapidly. This growth in data is driven by computing capacity – both within instruments and in supercomputing. Major weather centres will approach an exabyte of data in the near future, years before they have access to exascale computing, and so we believe the *first* exascale challenge for the scientific community is a data challenge, and the computing challenge [1] will follow! In this paper, we concentrate on the bottlenecks introduced into the relevant workflows by the volume and velocity of that data, and describe some existing and proposed solutions.

II. CONTEXT

The growth in data volumes arises from the inexorable exploitation of computing in instruments and simulation. In particular, both the weather and climate communities seek to

develop ever higher resolution models of the earth system, and run them in ensembles (e.g. see Figure 4 in [2]) Such extra resolution leads directly to larger volume output, and handling that in a timely manner brings velocity issues – can the input/output, storage, and workflow systems deal with the data in a timely manner?

An immediate question when faced with such issue is “Do we really need all that data?”. The answer is almost certainly not, insofar as some of the data being written out has little meaningful information content — being beyond the meaningful resolution [3] — yet it is written because people think it *might* be useful in the future. Similarly, perhaps not all ensemble members need to be fully written out, and both temporal resolution, and opportunities for online analysis before writing data out should all be considered. However, in many cases where the eventual analysis is not yet determined, the full resolution data may be needed since re-running models may be too expensive, or even impossible. In any event, reductions in output from “one-off” decisions will only postpone bottlenecks being introduced by the drive to higher resolution, they will need to be addressed.

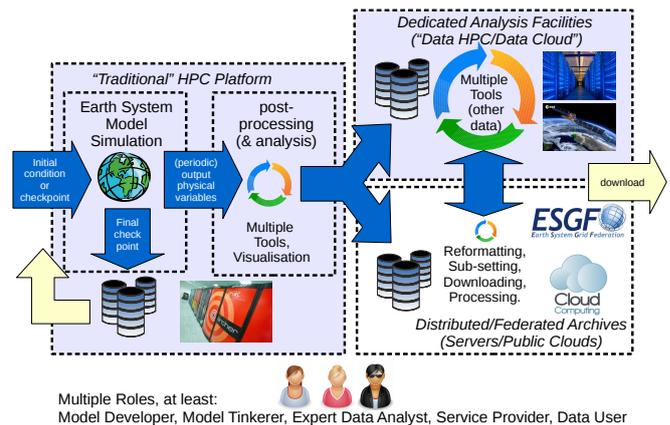


Fig. 1. Heterogeneity in the workflow platforms and requirements: from traditional HPC platforms, to dedicated analysis facilities, and data management and distribution systems, all with different requirements, serving users with a multitude or roles.

The workflow environment involved is complicated. Traditional HPC platforms have been augmented by dedicated

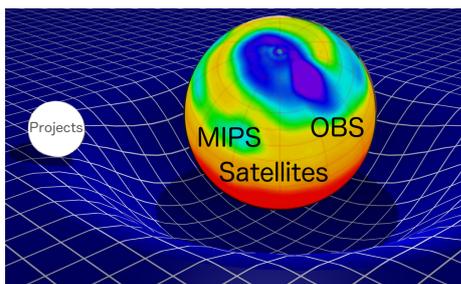


Fig. 2. Data Gravity: the JASMIN concept of a data commons is providing a large managed archive of data from ground based observations, as well as simulations from major modelling campaigns (MIPs). That provides an incentive for users to bring and share their own project data.

data analysis facilities, and complicated systems for distribution such as the Earth System Grid Federation [1] in use (figure 1). The storage capacity, types, and performance requirements are very different, and each has a different class of storage bottleneck to consider. On HPC platform the main issue is often performance — reaching sustained I/O performance from and to disk. On analysis platform there is I/O performance and storage to consider, and in data distribution systems, network [5] and software issues dominate to the point where most groups rely on dedicated local archives rather than personal downloading (e.g. see [6], in particular their figure 2).

III. CUSTOMISED HARDWARE

In the UK academic community, large weather and climate simulations are primarily carried out on one of two national supercomputers: ARCHER (in Edinburgh) or NEXCS (a portion of the Met Office supercomputer, in Exeter). Neither have large storage and/or analysis systems, and data output is migrated to JASMIN, a data analysis supercomputer for environmental supercomputing (near Didcot, in Oxfordshire). Dedicated high bandwidth network links are available to supplement backbone networks for data transfer.

JASMIN has been designed for environmental data analysis. As of September 2018 it has over 40 PB of storage, and over 10,000 cores distributed between a batch cluster and a community cloud. JASMIN implements a data commons (fig. 2) utilising the managed archive from the Centre for Environmental Data Analysis (CEDA, <https://ceda.ac.uk>) to underpin the services provided by JASMIN (fig. 3).

JASMIN is configured with a high performance storage environment [7], which is heavily used — data read rates exceed 1 PB/day for multi-day periods (fig. 4). However, despite the heavy use, there is considerable performance “left-on-the-floor”, as not all user codes can make effective use of the input/output performance available.

As of early 2018, the storage was divided into five classes: home, user scratch, group work space (GWS), and archive; with most of the space allocated to the archive (5 PB) and GWS (>12PB). Users are generally assigned access to one

LOTUS ----- Optimised High Performance Data Analysis Environment	Community Cloud ----- Customisable (with high performance route to archive)	CEDA Data Services ----- Remote access to archive & catalogues. Download etc
 CEDA Archives		
JASMIN – Data Intensive Computer Storage, Compute and Network Fabric Batch Compute, Private Cloud, Disk, Tape		

Fig. 3. JASMIN provides a range of services which exploit the CEDA archives and the customised hardware, the most important of which are the LOTUS batch cluster, the Community Cloud, and the CEDA data services, which together provide “Platform-as-a-Service”, “Infrastructure-as-a-Service” and “Software-as-a-Service”.

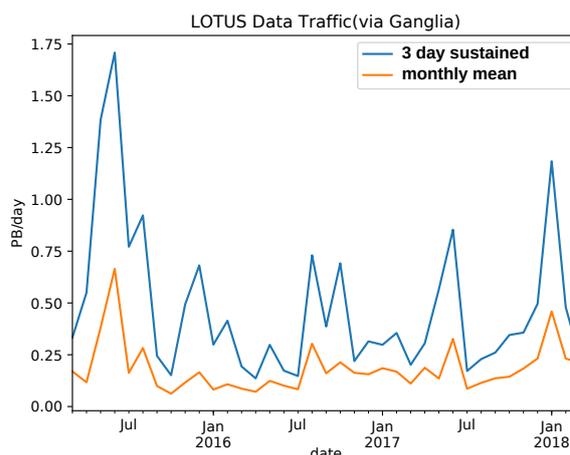


Fig. 4. Network traffic from storage into LOTUS showing data movement over several years. The blue line is the three day sustained average, the orange line, the monthly mean.

or more GWS, and the GWS allocations are constrained within consortium allocations controlled by an external board.

During most of the previous years, growth on disk was almost linear (fig 5, top panel¹). However that linear growth did not represent user demand, which was heavily constrained: the middle panel of fig 5 shows the archive on disk, and how it has been constrained by the allocation cap, despite the higher underlying growth in most of the archive — one example of which is the Sentinel data, shown in the bottom panel of figure 5. The group workspaces were also constrained: Figure 6 shows that much of the user growth was within GWS

¹Note that the early 2018 increase was due to data replication associated with an upcoming upgrade.

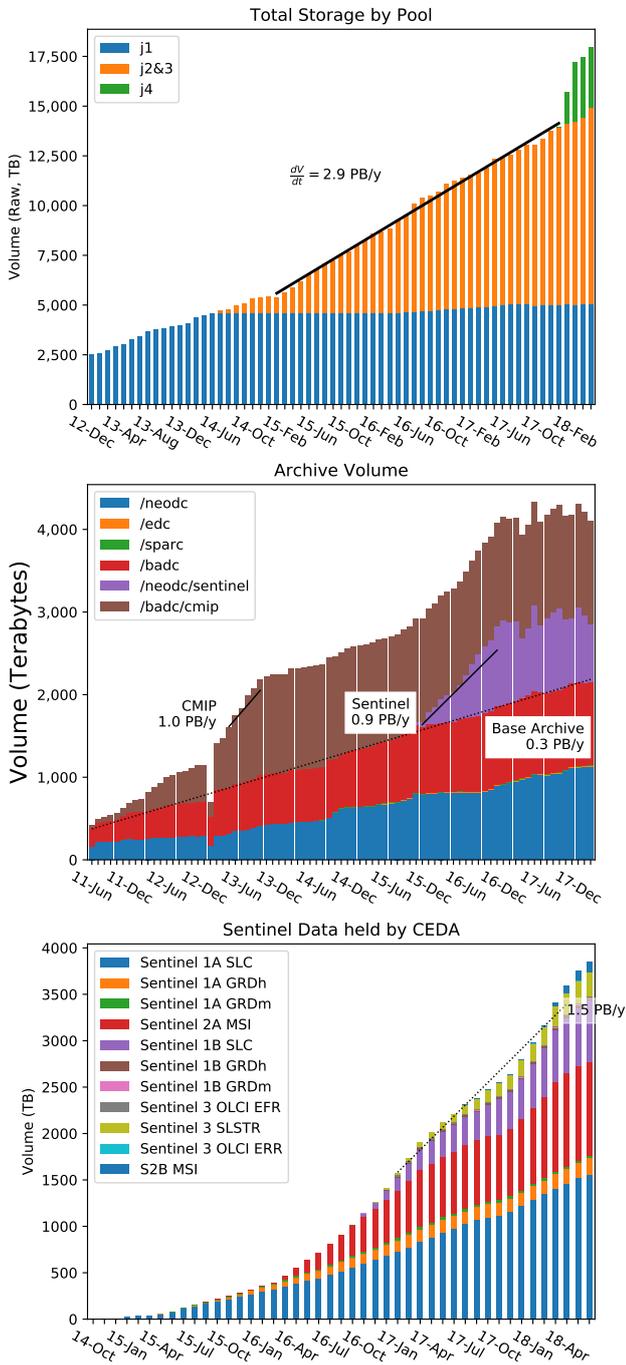


Fig. 5. Three aspects of storage volume growth: the total disk usage by storage pool (see text); total archive volume on disk, and total of the Sentinel data held in the archive (on tape and disk). Key points to note are that the overall linear growth (as opposed to exponential growth) is because of constraints on the archive and group work spaces sizes on disk (see text).

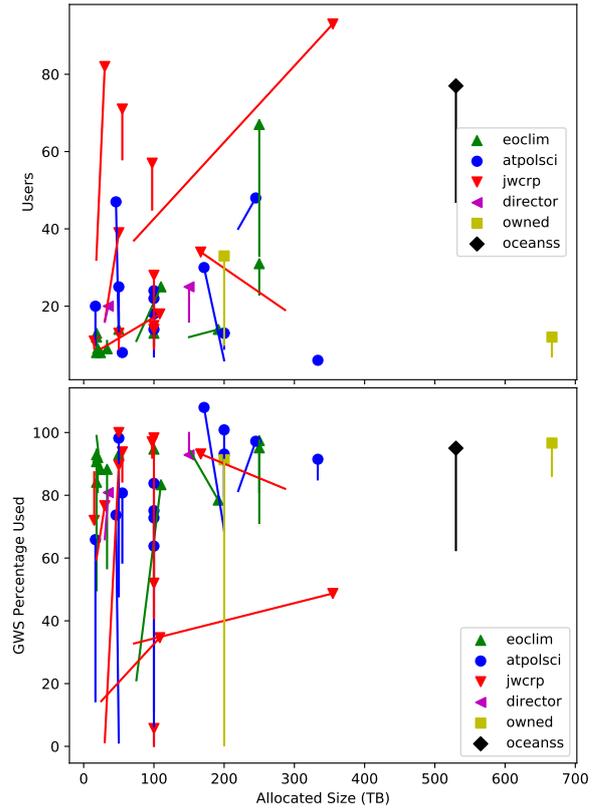


Fig. 6. Usage and fill factor on selected Group Work Spaces over 18 months to October 2017. Lines show change during this period, from beginning to end (denoted with the icons, which indicate the consortium). The top panel shows how the number of users has grown with most workspaces, while the bottom shows how the group work spaces have filled up to their allocations. In both cases, some GWS have also changed in size over that period.

and that users were constraining themselves to fit within their GWS allocations.

The split between archive and GWS, and the constraints which are applied to both, provide key mechanisms used by JASMIN to turn what would have otherwise been exponential data growth into relatively manageable linear growth (although even linear growth will not be affordable on disk if it exceeds the Kryder rate [8]).

IV. CUSTOMISED SOFTWARE

Some of the solutions to volume and velocity need to be addressed in both hardware and software.

Where volume and velocity combine, performance becomes an issue. As already noted, not all existing workflows make good use of parallel file systems, and may be more suited for other storage media. However, even where workflows are well suited for parallel file systems, the file systems themselves bring limitations which arise from the failure of POSIX at scale to handle high volume concurrent metadata look-ups and very large numbers of processes attempting to access a handful of files. Solutions generally involve application level tuning to local systems, resulting in poor performance portability.

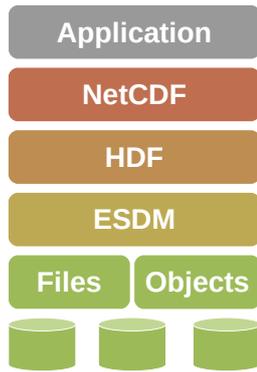


Fig. 7. The Earth System Data Middleware, ESDM, lies between the HDF5 library and storage volumes.

Migrating to object stores is one possible solution [9], but only as part of a plan which addresses higher performance at write time using traditional interfaces. However, object stores are subject to a declining Kryder rate too, so tape is an integral part of planning at most sites, including JASMIN, providing lower performance (and cheaper) storage where the “coldest” data can still be accessed quickly.

Object stores and tape bring another set of issues in that there is little available portable software in the weather and climate community which can easily exploit such storage in workflows. Data placement and appropriate metadata are key, but hierarchical namespaces are limiting, users generally do not have control over data placement, and system controls are often blind to expected usage and workflow requirements. While sophisticated solutions for tape usage exist at major sites (e.g. the ECMWF MARS system [10]), they do not yet incorporate object stores, or if they can (or will soon), they do not deliver portable solutions.

The Centre of Excellence in Simulation of Weather and Climate in Europe (ESiWACE, <https://esiwace.eu>) is addressing these issues in a focused attempt to develop portable software. Currently there are two strands of activity:

- 1) The *Earth System Data Middleware* (ESDM), which provides a library which sits between traditional HDF and NetCDF interfaces and storage to deliver performance portability; and
- 2) The *Semantic Storage Tools* which are aimed at providing suitable portable interfaces to both tape and object storage and providing users the ability to manage their own placement on tiered storage, without losing visibility of their metadata.

A. Earth System Data Middleware

The ESDM targets performance portability by providing software that can be linked into existing applications, but take advantage of knowledge of the local storage environment. Design goals include: (1) Ease of use and deployment; (2) Exploiting knowledge of data structures and scientific metadata to provide efficiency, (3) Supporting multiple read-patterns

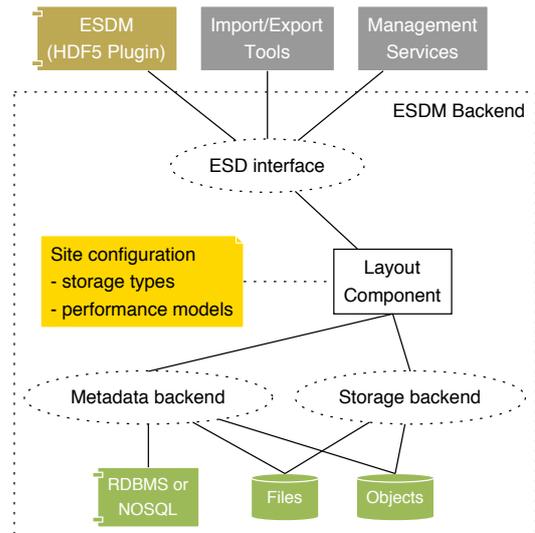


Fig. 8. Core architecture of the ESDM. The HDF plugin and external tools interface to the layout component which is configured with information about available storage types and their expected performance. Metadata and storage backends can use whatever is available.

efficiently, and; (4) Reducing the penalties of shared file access (i.e. deliver “lock-free” writes in parallel applications).

Ease of use is delivered by providing a library which can be linked into existing applications using HDF or NetCDF (figure 7 along with configuration which involved site-specific optimised data layout schema, figure 8). Administration and user tools will provide import export and monitoring.

Performance is delivered by exploiting knowledge of the scientific structures to deliver the necessary lock-free writes by handling data as atomic fragments.

The current status of the ESDM software is that prototypes have been built on a number of systems, and it has been demonstrated to perform significantly better on Lustre file systems than the native HDF5 writing to Lustre. Details of that performance, and results on other systems will appear elsewhere. User management tools are not yet available.

Future plans include exploiting internal ESD backend daemons to rearrange data for multiple different access patterns requested by “usage hints” delivered at write-time, or via the user-tools interface. These daemons will also be able to rearrange data on the fly for export to remote sites (for example, via Globus).

B. Semantic Storage Tools

The semantic storage tools target direct use of object stores by user software, as well as user-controlled data management in a tiered storage environment.

Currently it is not easy to exploit object stores directly in normal user workflows, most software is predicated on systems and libraries which expect to be working with POSIX filesystems. S3NetCDF (fig 9) addresses this for Python users by providing a drop-in-replacement for NetCDF-python. NetCDF datasets are fragmented using the Climate Forecast

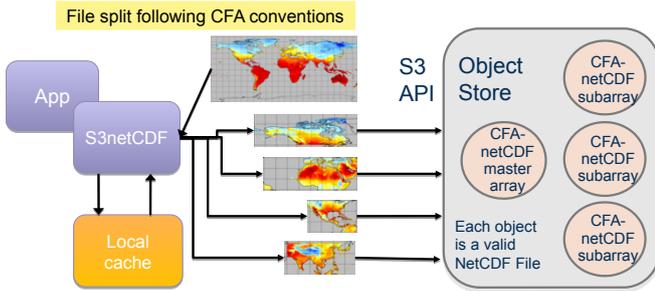


Fig. 9. S3NetCDF involves a splitter which utilises the CF conventions to split a multi-field NetCDF file into a master file and smaller fragments, which are all stored separately (on tape, or on disk in an object store or file system).

Aggregation conventions [11] resulting in a set of individual NetCDF files which can be stored as objects or files, along with a master array describing how they are aggregated. Users can keep the master array on normal disk, and S3NetCDF simply opens that file, and reads/writes the fragments into/from memory from/into storage.

S3NetCDF exists as a functional prototype, but it accesses fragments in serial, and performance is relatively poor. It is currently being rewritten to exploit the available parallelism to deliver what is hoped to be even better performance than is available using normal POSIX disk access.

Even without direct access to object stores from user codes, object stores can be treated like tape, and used for stashing “colder” data for later use. However, where users are managing this process, the major problem is maintaining information about what is on such storage. Lists of filenames are inadequate, and local bespoke solutions do not allow users to manage their data across multiple sites. These issues are being addressed by the development of cache facing software that (1) manages data migrations, and (2) allow users to manage their own metadata about what is where.

This software, currently known as CacheFace, depends on three key internal components: a data migration utility, a cache management utility, and a metadata system. Development on each is underway, with the data migration tool reaching a sufficient level of maturity so as to be deployed on JASMIN (as the JASMIN Data Migration App) in the final quarter of calendar year 2018. The other two only have rudimentary prototypes, but have the same fundamental requirements as the ESDM and S3NetCDF, so development is expected to be relatively swift. Exploiting these underlying similarities will be one of the goals of the ESiWACE2 project beginning in 2019, with long-term maintenance of the tools being picked up by institutional partners.

V. RELATED WORK

A number of sites are developing hybrid HPC/cloud solutions, and some are at a similar scale in terms of compute, e.g [12]. However, we believe JASMIN is unique in terms of the co-location with a managed archive crossing a wide range of environmental data, although the Polish Innovation Testbed

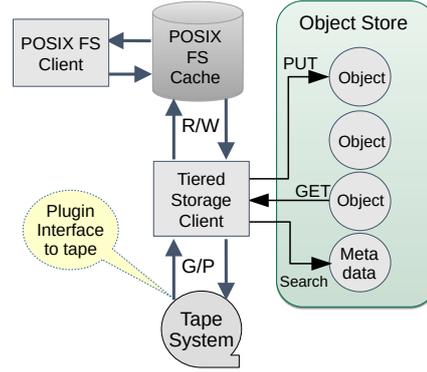


Fig. 10. CacheFace will provide a POSIX front end which manages and migrates data between storage tiers while exposing NetCDF and other metadata to the user regardless of where the data is stored, whether on tape, or disk.

hosts a range of earth observation data [13], and a number of sites are providing computational facilities alongside ESGF climate data.

The ESDM is built on a middleware heritage that some may argue began with ADIOS [14], and has many characteristics in common with sophisticated solutions for buffering data flow in tiered disk storage (e.g. [15], [16] and managing scientific workflows [17]. The ESDM differs from these more generic solutions, by attempting to make use of our domain specific knowledge about the contents of NetCDF data.

There are broadly two current approaches to exploiting object stores: attempts to use middleware to unify I/O stacks (as we are doing with the ESDM) or providing “POSIX-like” or “POSIX-light” file system interfaces that drop some of the full POSIX requirements in order to exploit object stores efficiently. Differing examples from the research and commercial sphere include MarFS [18] and QuoByte². Our approach is somewhat different, by again using our domain specific knowledge (specifically, the Climate Forecast conventions [19]) and the ability to split metadata and data to effectively exploit object stores. The same domain specific knowledge provides the key point of difference for our CacheFace development from other data migration and caching systems (of which there are too many to reference here).

VI. SUMMARY

Modern weather and climate workflows demand customised data analysis environments with specialised hardware and user configurable software environments (virtualisation, containerisation etc). These requirements are met in the UK by the JASMIN facility which co-locates a “community cloud” with a managed archive, a large batch cluster, and a sophisticated tiered storage system. Over the last few years, most user workflows have been able to be accommodated on JASMIN disk, with tape used only for backup and long-term archive, however, projections of future demand suggest that “disk-only”

²<https://www.quobyte.com>

workflows will need to be supplanted by workflows which include more tiers of storage, including tape.

Within those workflows more parallelism will be necessary to avoid unreasonable wall clock times, but such parallelism will not eventuate without both new algorithms and approaches by users and the widespread availability of more efficient and smarter storage middleware and data management software. The European ESiWACE project is addressing these middleware and data management software requirements by developing two families of products: high performance middleware to lie beneath commonly used software libraries like HDF and NetCDF4 (the “Earth System Data Middleware, ESDM”), and user deployable portable tools to manage data in a tiered storage environment.

These two approaches to beating data bottlenecks, smarter hardware and software, will not be enough on their own. The reality of storage economics coupled with feasible data production volumes and velocity mean that despite technology innovations, the most important approach to these data bottlenecks will be avoiding the problem in the first place by writing less data! This means that experimental design and analysis workflows will need radical rethinking — a process that will inevitably involve the entire scientific community, not just the technical experts.

ACKNOWLEDGEMENTS

JASMIN is supported by the UK Natural Environment Research Council and Science and Technology Facilities Council. The ESiWACE project is supported by the European Union’s Horizon 2020 research and innovation programme under grant agreement No 675191. The authors are grateful to the entire JASMIN and CEDA teams and the ESiWACE team involved with WP4: Exploitability (of storage).

REFERENCES

[1] B. N. Lawrence, M. Rezny, R. Budich, P. Bauer, J. Behrens, M. Carter, W. Deconinck, R. Ford, C. Maynard, S. Mullerworth, C. Osuna, A. Porter, K. Serradell, S. Valcke, N. Wedi, and S. Wilson, “Crossing the chasm: How to develop weather and climate models for next generation computers?” *Geoscientific Model Development*, vol. 11, no. 5, pp. 1799–1821, May 2018.

[2] J. Mitchell, R. Budich, S. Joussame, B. Lawrence, and J. Marotzke, “Infrastructure strategy for the European Earth system modelling community 2012–2022,” 2012.

[3] S. Abdalla, Lars Isaksen, Peter A.E.M. Janssen, and Nils Wedi, “Effective spectral resolution of ECMWF atmospheric forecast models.” *ECMWF Newsletter*, vol. 137, pp. 19–23, 2013.

[4] D. N. Williams, B. N. Lawrence, M. Lautenschlager, D. Middleton, and V. Balaji, “The Earth System Grid Federation: Delivering globally accessible petascale data for CMIP5,” in *Proceedings of the 32nd Asia-Pacific Advanced Network Meeting*, New Delhi, Dec. 2011, pp. 121–130.

[5] E. Dart, M. F. Wehner, and Prabhat, “An Assessment of Data Transfer Performance for Large-Scale Climate Data Analysis and Recommendations for the Data Infrastructure for CMIP6,” *ArXiv e-prints*, vol. abs/1709.09575, 2017, primaryClass: cs.DC.

[6] V. Balaji, K. E. Taylor, M. Juckes, B. N. Lawrence, P. J. Durack, M. Lautenschlager, C. Blanton, L. Cinquini, S. Denvil, M. Elington, F. Guglielmo, E. Guilyardi, D. Hassell, S. Kharin, S. Kindermann, S. Nikonov, A. Radhakrishnan, M. Stockhause, T. Weigel, and D. Williams, “Requirements for a global data infrastructure in support of CMIP6,” *Geoscientific Model Development*, vol. 11, pp. 3659–3680, Sep. 2018.

[7] B. Lawrence, V. Bennett, J. Churchill, M. Juckes, P. Kershaw, S. Pascoe, S. Pepler, M. Pritchard, and A. Stephens, “Storing and manipulating environmental big data with JASMIN,” in *2013 IEEE International Conference on Big Data*, Oct. 2013, pp. 68–75.

[8] P. Gupta, A. Wildani, E. L. Miller, D. Rosenthal, I. F. Adams, C. Strong, and A. Hospodor, “An economic perspective of disk vs. flash media in archival storage,” in *2014 IEEE 22nd International Symposium on Modelling, Analysis & Simulation of Computer and Telecommunication Systems*. IEEE, 2014, pp. 249–254.

[9] D. Goodell, S. J. Kim, R. Latham, M. Kandemir, and R. Ross, “An Evolutionary Path to Object Storage Access,” in *2012 SC Companion: High Performance Computing, Networking Storage and Analysis*, Nov. 2012, pp. 36–41.

[10] M. Grawinkel, L. Nagel, M. Masker, F. Padua, and A. Brinkmann, “Analysis of the ECMWF Storage Landscape,” in *Proceedings of the 13th USENIX Conference on File and Storage Technologies*. Santa Clara: USENIX Association, 2015, pp. 15–27.

[11] D. Hassell, “The CFA-netCDF conventions,” <http://www.met.reading.ac.uk/~david/cfa/0.4/cfa.html>.

[12] Y. Li, X. Zhang, A. Srinath, R. B. Getman, and L. B. Ngo, “Combining HPC and Big Data Infrastructures in Large-Scale Post-Processing of Simulation Data: A Case Study,” in *Proceedings of the Practice and Experience on Advanced Research Computing - PEARC '18*. Pittsburgh, PA, USA: ACM Press, 2018, pp. 1–7.

[13] A. Romeo, S. Pinto, S. Loekken, and A. Marin, “Cloud Based Earth Observation Data Exploitation Platforms,” New Orleans, Dec. 2017.

[14] J. Lofstead, F. Zheng, S. Klasky, and K. Schwan, “Adaptable, metadata rich IO methods for portable high performance IO,” in *IEEE International Symposium on Parallel & Distributed Processing, 2009. IPDPS 2009*. IEEE, 23–29 May 2009, pp. 1–10.

[15] A. Kougkas, H. Devarajan, and X.-H. Sun, “Hermes: A Heterogeneous-aware Multi-tiered Distributed I/O Buffering System,” in *Proceedings of the 27th International Symposium on High-Performance Parallel and Distributed Computing*, ser. HPDC '18. New York, NY, USA: ACM, 2018, pp. 219–230.

[16] B. Dong, S. Byna, K. Wu, Prabhat, H. Johansen, J. N. Johnson, and N. Keen, “Data Elevator: Low-Contention Data Movement in Hierarchical Storage System,” in *2016 IEEE 23rd International Conference on High Performance Computing (HiPC)*. Hyderabad, India: IEEE, Dec. 2016, pp. 152–161.

[17] J. Wang, D. Huang, H. Wu, J. Yin, X. Zhang, X. Chen, and R. Wang, “SideIO: A Side I/O system framework for hybrid scientific workflow,” *Journal of Parallel and Distributed Computing*, vol. 108, pp. 45–58, Oct. 2017.

[18] J. Inman, W. Vining, G. Ransom, and G. Grider, “MarFS, a Near-POSIX Interface to Cloud Objects,” *login.*, vol. 42, no. 1, p. 6, 2017.

[19] D. Hassell, J. Gregory, J. Blower, B. N. Lawrence, and K. E. Taylor, “A data model of the Climate and Forecast metadata conventions (CF-1.6) with a software implementation (cf-python v2.1),” *Geoscientific Model Development*, vol. 10, no. 12, pp. 4619–4646, Dec. 2017.