Storing and manipulating environmental big data with JASMIN

B.N. Lawrence*†§, V.L. Bennett†¶, J. Churchill‡, M. Juckes†§, P. Kershaw†¶, S. Pascoe †§, S. Pepler †§¶, M. Pritchard†§¶ and A. Stephens†§.
*Department of Meteorology, University of Reading, Reading, UK.
§National Centre for Atmospheric Science, UK. ¶National Centre for Earth Observation, UK.
†Centre for Environmental Data Archival, ‡Scientific Computing Department,
STFC Rutherford Appleton Laboratory, Didcot, UK.

***Abstract*—JASMIN is a super-data-cluster designed to provide a high-performance high-volume data analysis environment for the UK environmental science community. Thus far JASMIN has been used primarily by the atmospheric science and earth observation communities, both to support their direct scientific workflow, and the curation of data products in the STFC Centre for Environmental Data Archival (CEDA). Initial JASMIN configuration and first experiences are reported here. Useful improvements in scientific workflow are presented. It is clear from the explosive growth in stored data and use that there was a pent up demand for a suitable big-data analysis environment. This demand is not yet satisfied, in part because JASMIN does not yet have enough compute, the storage is fully allocated, and not all software needs are met. Plans to address these constraints are introduced.**

Keywords: Curation, Climate, Earth Observation, Big Data, Cloud Computing

I. Introduction

The JASMIN super-data-cluster is the central node of a geographically spread environmental e-infrastructure. It deploys petascale fast disk connected via low latency networks to a range of computing services initially designed to serve a range of clients, but primarily those with "big data handling needs" from UK atmospheric and earth observation science. JASMIN is managed and delivered by the UK Science and Technology Facilities Council (STFC) Centre for Environmental Data Archival (CEDA). The JASMIN core systems were installed at the STFC Rutherford Appleton Laboratory (RAL) in early 2012. The wider JASMIN e-infrastructure includes additional nodes in Leeds, Bristol, Reading as well as the system at RAL.

In this paper, we introduce the configuration and first year of experience with the central JASMIN infrastructure. [1] discuss the generic scientific requirements and infrastructural context in some detail. Here we present a short summary of those aspects before a detailed discussion of the current central architecture, some of our configuration choices and consequential experience, some examples of usage from the primary communities, and plans for the near future.

A. Scientific and Infrastructural Context

Data handling has become an intimidating part of the scientific workflow for many in the earth sciences community, particularly those doing climate science, whether with earth observation data or simulated data or both. This workflow extends from acquisition of third-party data as inputs and/or for validation, through manipulation of what can be highly heterogeneous and/or large volume datasets, to the storage of intermediate and final products. Many such final products (from UK science funded by the Natural Environment Research Council, NERC) find their way into a NERC designated data centre - three of which (atmospheric science, earth observation, and solar terrestrial physics) are hosted at CEDA.

Management of the core JASMIN infrastructure contributes to the CEDA remit to both curate data and facilitate science. It will be seen that JASMIN provides support both for the scientific workflow of the research community (via group workspaces and computing capability) and the curation activities of CEDA (via the archive and associated computing). An important distinction is between storage which is part of the scientific workflow (delivered by group workspaces) and storage for the curated archive (the archive), which has architectural implications.

Some exemplar projects which show the scale of data handling problems in the JASMIN communities are introduced in this section.

1) CMIP5 and CEDA: The recent fifth coupled model intercomparison project (CMIP5, see [2]), has so far produced over 2 petabytes (PB) of *requested* data from over 100 different numerical experiments run by 29 different modelling centres using 61 different climate models. These data are stored in a globally distributed archive currently using 23 geographically distinct data nodes. However, any given user generally wants some data from all locations, and for a sufficiently large community, they want a large portion of that data available to them alongside their computational resource. Hence, while the source data is globally distributed, there are many public and private replicants of that data. We might assume in fact, that there are tens of PB of CMIP5 *requested* data "out there". (The italicised reference to requested indicates that the source modelling data centres would have produced much more than that requested for the public intercomparison - with many centres producing PB of which only a fraction made its way into the requested archive.)

Within the global federation of data nodes (the Earth System Grid Federation, ESGF, [3]), three have agreed to manage

replicants of as much of the requested data as possible, and one of those is the National Centre for Atmospheric Science's British Atmospheric Data Centre (BADC, a core component of CEDA). Accordingly, CEDA is currently managing 167 TB within ESGF, with an additional 650TB available to JASMIN users via the CEDA archive (not yet in ESGF). Those volumes will be increasing, as eventually more than 50% of the CMIP5 requested archive will be replicated into JASMIN. This will provide the UK atmospheric science community with an environment which maximises the ease of handling CMIP5 data while minimising the resources consumed in doing so (people, stored copies, power), and ensure that the CMIP5 data is curated for the long term (the CMIP5 curation task is being done in an international partnership under the auspices of the Intergovernmental Panel for Climate Change, IPCC).

CMIP5 is only one of many curated datasets within CEDA, although it is now the largest one (at over 0.85 PB). The entire CEDA archive currently consists of 1.8 PB of data, much of which was migrated from network attached storage into JASMIN over the last year.

2) UPSCALE and the JWCRP: High resolution climate modelling stresses supercomputing from both the computational and data handling perspectives. Problems with the latter include handling the data as it is produced and providing a suitable analysis environment for some years thereafter. Traditionally the analysis environment has been alongside the high performance computing resource, but modern climate modelling involves significant amounts of intercomparison between differing simulations and data. Some data movement is inevitable, the question then becomes how to minimise it? One way is by providing a centralised data analysis facility, and moving as much data as possible there - resulting in N data transfers for N simulations, rather than the $N \times N$ transfers if each user copies data to their local institution.

As an example, one high resolution climate modelling experiment, UPSCALE [4], a project run under the auspices of the UK Joint Weather and Climate Research Programme (JWCRP, joint between NERC and the UK Met Office) was carried out in 2012. UPSCALE involved the use of 144 million core hours on the German supercomputer HERMIT, and it produced 330TB in 2012. The data were brought back to JASMIN using gridFTP at a rate which varied between 1 and 10 TB per day. At JASMIN, a second copy was kept until the data had also been copied to the Met Office tape archive (yet another wide area network replication, but at a much lower bandwidth than the link between HERMIT and JASMIN). At its peak, the UPSCALE archive online at JASMIN approached 600TB - and it is now around 380TB (including post-processed products). These data are expected to provide a hugely valuable resource for the study of current and future climate, complementing previous simulations at coarser resolutions, hence some UPSCALE data will eventually become part of the ESGF and the CEDA archive. Meanwhile, much of the post-processing involves comparisons with CMIP5 and earth observation data.

3) Mission reprocessing, CEMS and NCEO: The Earth Observation community are a major client of the JASMIN infrastructure, particularly the facility for Climate and Environmental Monitoring from Space (CEMS). CEMS itself consists of two components: the academic CEMS infrastructure, running on JASMIN, and the commercial CEMS infrastructure part of the new UK Satellite Applications Catapult centre (http://sa.catapult.org.uk/cems/climate-and-environmental-monitoring-from-space/). The academic component is delivered by the National Centre for Earth Observation in the CEDA infrastructure, where again, the role is both curation and facilitation.

One example of a CEMS project is the CEMS-Globalbedo project [5] which exploits the Moderate Resolution Imaging Spectroradiometer (MODIS, http://modis.gsfc.nasa.gov/). The aims of this project are to provide an interface to an existing 1km global resolution bidirectional reflectance distribution function (BRDF) product for a 14 year dataset at 8 day resolution (a 53 TB dataset); and support a new 500m resolution BRDF and albedo product for 13 years (45 TB). These data are key inputs to a range of weather, climate, and earth observation algorithms. The project has been provided a 100 TB group workspace, and computing via hosted processing. At the time of writing, the project is generating thousands of batch compute jobs per week.

This is one example of "whole mission reprocessing", a job that hitherto was done rarely. Two other such examples from CEMS running on JASMIN include two different products calculated from ATSR (the Along Track Scanning Radiometer, http://atsrsensors.org/) brightness temperature data held in the CEDA archive: land surface temperature [6] and clouds [7].

II. The JASMIN super data cluster

The entire JASMIN system is a distributed computing environment [1], here we are concentrating on the core system. The technical architecture was chosen both to deliver ease of management and to provide a very flexible high performance storage and analysis environment. Ease of management was a major consideration, because limited staff resources were (and are) available, and flexibility, because we expect the community to migrate their workloads (in all directions) between batch computing, hosted processing, and a full cloud environment. In making these distinctions, we are distinguishing between a cloud providing infrastructure as a service (with limited access to high performance disk), a hosted processing environment (allowing users to manage their own pre-configured virtual machines with high performance storage access), and a traditional batch computing environment (again with access to high performance disk).

A. Architecture

The JASMIN architecture is depicted in figure 1. The system essentially consists of six major components:

1) The low latency core network (based on Gnodal switches);
2) The Panasas storage sub-system;
3) The batch compute system ("Lotus HPC");
4) The data compute systems providing both bare metal compute and the hypervisors for virtual machines;
5) A High Memory System and
6) Two image stores to support the private disks of the virtual machines.

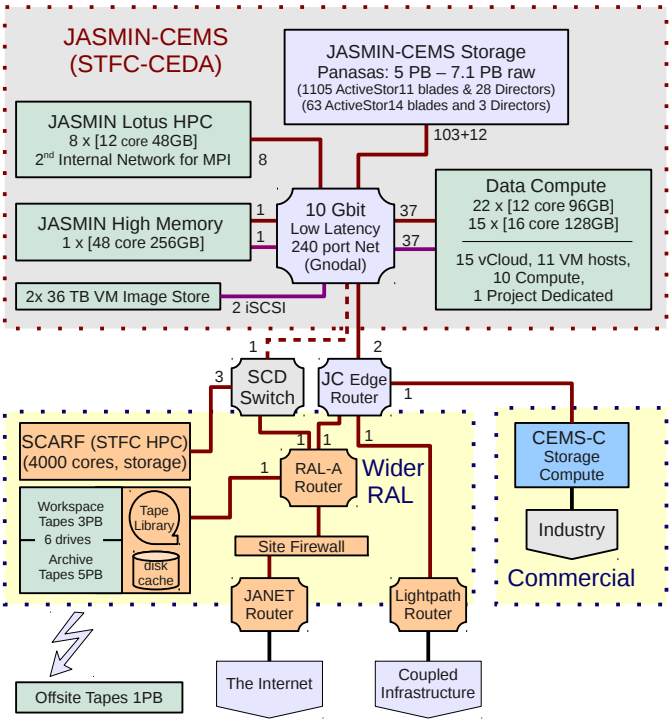


Fig. 1. Key JASMIN components: there are two major sections to the core infrastructure: the JASMIN core facilities, and the network links to the commercial component of CEMS, the tape archive and remote resources. The core facilities include the fast storage (connected to the low latency network with 115 10-Gbit/s connections), the data compute services, the "Lotus" HPC (batch compute service), the high memory service, and two image stores. The wider RAL facilities include the commercial CEMS component and the tape archive (the SCARF HPC facility is currently connected by a dedicated line, soon to be removed). Traffic from offsite passes through one of three network links:the commercial link to CEMS, the main academic route to JANET, and a lightpath router (to specific remote HPC and data archives).

Currently the system provides 600 cores and 5 PB of usable disk. It is connected to the commercial component of CEMS (CEMS-C) at 10 Gb/s, to the SCARF HPC system at 10 Gb/s and to the rest of RAL at 20 Gb/s. Although not physically located within the JASMIN system the RAL Scientific Computing Department's (SCD) tape library and disk cache subsystem provides the key capability both to backup the archive, and for group workspace users to exploit "elastic tape" for disk overflow (these terms are explained below). SCD also provide access for STFC users of JASMIN to their general purpose HPC system, SCARF, which has 4000 cores and a local 210 TB (usable) Panasas storage system.

The JASMIN architecture is designed to support a very MAD analysis environment [8]: Magnetic, in that it is attractive and easy to load data into the environment; Agile, in that it supports a wide range of analysis modes, and; Deep, in that it is possible to deploy very sophisticated algorithms. However, while the MAD paradigm was invented in the context of extending traditional database analysis techniques to support map-reduce, we have here attempted to take the concept and deploy it to support traditional academic file-based analysis workflows. In doing so, we have rejected the use of HADOOP [9] since the combination of high volume and diverse workload means that a constrained data-layout can lead to very unbalanced systems, and poor performance [10] — and solutions such as increased data replication, and data migration are not tenable at petascale. Instead, we are using high performance parallel storage to deliver the I/O performance benefit of HADOOP alongside a range of computing virtualisation options to support a variety of parallel algorithms, including map-reduce. We think this is more practical and has more longevity than more technical approaches such as embedding scientific formats into the HADOOP stack (e.g. [11]).

B. *JASMIN services and virtualisation*

The three major constituencies of JASMIN — curation, atmospheric and earth observation science — have a range of requirements for data analysis and services [1].

The curation requirement is primarily to provide reliable storage and compute capable of supporting data management and data services, and this requires a clean distinction between the curated archive and associated services, and the wider user computing environment. At petascale it is difficult for a service provider to optimise the computing environment for all customers, so JASMIN is designed to push as many possible decisions on the computing environment up the stack. The bare metal computing is managed by SCD under a Service Level Agreement to CEDA, who both manage their own (virtualised) computing, and provide hosted computing to customers. Three computing environments are made available to users: a hosted processing environment (utilising virtual machines managed and deployed by SCD - but in many cases, configured by CEDA), a cloud environment, and a batch computing environment. The latter has three sub-classes: high memory, standard computing, and parallel (systems in the Lotus HPC cluster have a second network card used to provide MPI on a separate network from the I/O traffic). CEDA itself also uses both the hosted processing environment (for most data services), and the batch computing environment (for large data management tasks such as routine checksums of the petascale archive).

General user communities (such as NCAS and NCEO) are allocated group workspaces with guaranteed access to storage up to a specific volume, while major projects (such as UPSCALE) get allocated group workspaces in their own right. Such communities also get allocated a virtual machine for hosted processing, for which they take most of the management responsibility. In some cases multiple machines for one community make organisational sense: for example, there are two UPSCALE VMs: one for the academic partners, and one for the Met Office. Each is managed independently, with very different software environments. Communities with small computational demands, or without the ability to manage their own computing, share access to generic science processing machines. Whatever their hosted processing environment, all get access to batch processing if desired. At the time of writing 32 group workspaces have been allocated with 50 virtual machines in their support.

To support the efficient deployment of machines in this hosted environment CEDA has developed a repository of specially built RPM packages (see http://proj.badc.rl.ac.uk/cedaservices/wiki/JASMIN/ScientificAnalysisVM/Packages).

The virtualisation environment has given a degree of flexibility both for users and administrators of the system.

However, as a managed environment it has required close co-ordination between the users and the SCD and CEDA support teams. In some cases, users have desired a higher degree of autonomy and control over their hosted environments and this is in part driven by practical experience using public clouds. VMware vCloud was deployed in the initial phase to provide such a service. Compute, network and storage resources can be partitioned between different individual users or whole organisations in virtual data centres. vCloud provides a both a web portal and RESTful web service interface enabling administrators of such virtual data centres to configure the available resources. However, in practice, concerns about the granting of root privileges and issues with integration with the Panasas storage have thus far prevented vCloud from being fully exploited.

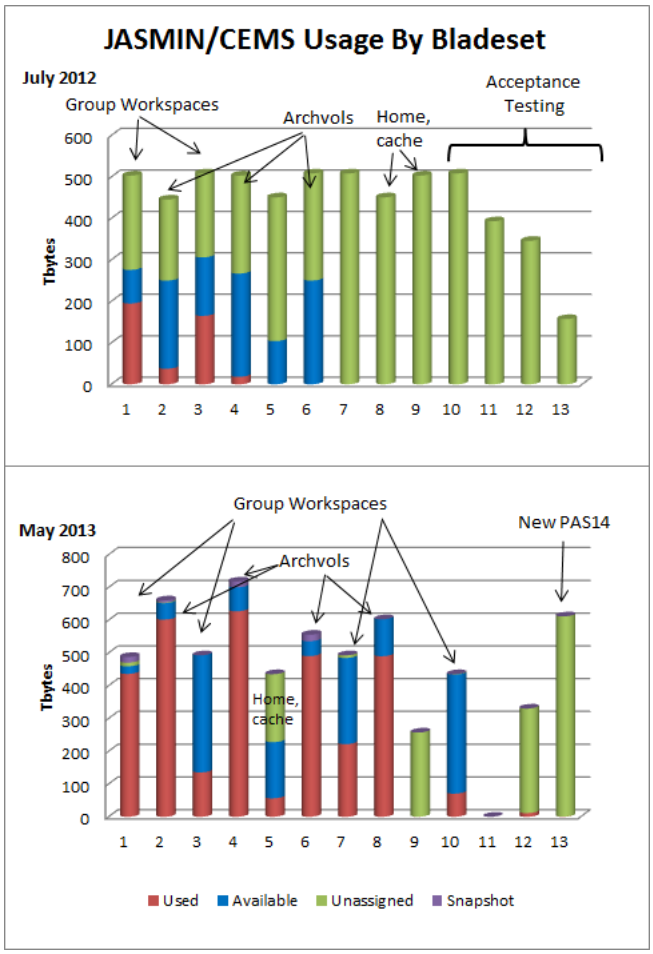


Fig. 2. Panasas storage layout in JASMIN/CEMS: as initially configured in July 2012 and ten months later in May 2013. Points to note include the changing size of file systems and the filling of the storage. Note that we use the term "available" to denote allocated storage, from a planning point of view, this storage is already used since it cannot be shared between projects.

C. Storage Configuration

A key design and operational decision in the JASMIN system was how to make use of the massively parallel storage resource — this in a context where risk of failure, compute capacity and network bandwidth (a function of availability network interfaces on both the client and server side) are all important.

The Panasas storage is deployed in multiple bladesets as discussed in [1]. From a data management and security perspectives, there was a requirement to separate storage used for long term archive storage from those used for active science (the group workspaces). We did this by using differing bladesets. Bladesets can be expanded by adding additional shelves, but not decreased in size (without first draining all data from them). An initial bladeset layout of 400 TB usable (500 TB raw) was chosen to provide a balance between fault tolerance, data management (integrity and governance) and performance, knowing that the management functions of Panasas made it relatively easy to subsequently adjust these sizes upward.

Fault tolerance issues include minimising the consequences of a range of possible failure modes: disk failures, blade failures, bladeset failures, and whole system failures. More (smaller) bladesets mitigate against the consequences of bladeset failures, but fewer (larger) bladesets are easier to manage in terms of data layout and have higher performance (since each blade contributes one network interface for two disks), but conversely, are harder and riskier to restore in the event of a failure. (Larger bladesets have a higher possibility of a second blade failure during reconstruction following a first blade failure, which would mostly result in data loss. This is a consequence of only having RAID5 available. RAID6 is promised for a Panasas firmware release in 2014 which will allow significantly larger bladesets.)

When considering disk usage, we consider that storage is "used" once it has been allocated, whether or not it is filled, since at that point a community has expectations that they own that storage, and so it needs to be available for them. For these large projects which have multi-year storage residence times, we do not believe it possible for us to share disk and make space on demand by flushing "old" data, a point we pick up below in the discussion on backup. In that sense, once allocated we consider our communities to be sharing our storage management, but not the storage — we leave the sharing to negotiations between the users within the communities we serve.

The initial layout of the storage (as of July 2012) is shown figure 2a. At that time (shortly after commissioning), very little data had been migrated from the legacy CEDA environment, and little data had arrived from outside. Nonetheless, we can see that a significant portion of the storage had been allocated, and the logical distinction between group workspace storage (bladeset 1 and 3) and archival storage (2,4,6) was already made. Home and cache were on bladeset 5, with the rest free or in acceptance testing. Ten months later, the original storage was effectively full, and new PAS14 storage units had been purchased. At this time we can see that several bladesets had been grown in size (effectively by cannibalising bladeset 9 and 10 before the latter began to be used). This flexibility, which was not available in a NAS and NFS environment, has been key to minimising both system management time and internal data migration time within the storage environment.

The new PAS14 storage is expected to have higher performance (compared to the original PAS11) for small files (filesystem metadata is stored on solid state disk, so metadata intensive tasks such as small file manipulation are expected to perform much better). In addition, PAS14 provides an even

higher bandwidth-to-storage ratio (20Gbps per 80TBytes raw vs 10Gbps per 60TB raw).

As well as the layout, there are various configuration parameters associated with the Panasas storage. Typical HPC parallel file systems have a rather straightforward connection to massively parallel and (often) homogeneous compute nodes, and possibly to a handful of post-processors (and/or data transfer systems). The JASMIN storage already has heterogeneous compute nodes (with more heterogeneity expected in future phases), and more importantly, both physical and virtual clients. One rather unexpected result from the initial parameter testing involved in configuring the storage for 10 Gb/s networks (as opposed to the default 1 Gb/s network) was the observation that the network drivers on the virtual clients initially performed better than those on the physical clients. As a consequence, detailed storage setup parameters were only set after a comprehensive parameter sweep to find the optimal configuration. (At the time of writing, the distinction between network driver performances is suspected to be associated with particular combinations of operating system and network interface card firmware.)

D. Backup at Petascale

Nearly all supercomputer environments provide a storage environment that is split up into a backed up "home" environment and a scratch "work" environment. The general assumption is that in the case of loss of data in the work/cache environment, the data can be re-generated. By contrast, data archives go to great lengths to ensure that all data is backed up, preferably with multiple replicants, some off-site.

JASMIN supports both environments; home directories and the archive are fully backed up, and for data where CEDA is the archive of last resort, offsite data copies are kept. Group workspaces are not backed up, but a new "elastic tape service" is about to be deployed. Formal backup for group workspaces in a petascale data analysis environment would be time-consuming, expensive, and inefficient: system managers have no easy way of identifying important data from temporary intermediate products, and if the latter themselves approach petascale, a lot of time and money could be spent storing data never to be used. Nonetheless, it is not tenable in the JASMIN environment to assume data can be regenerated - for example, the UPSCALE experiment could not be repeated because of the scale of the resources required. However, users do know what data is important and needs backing up. In the case of UPSCALE, the data is backed up in the Met Office tape archive, and some will migrate to the formal CEDA archive, but these options are not available to all users, and so JASMIN will provide (in summer 2013) nearline storage which is under deliberate manual control — as opposed to being subject to automated policies such as might occur in a hierarchical storage management system (we are not convinced that sensible policies can be established for the group workspaces). Instead, group workspace managers will be able to control what is online and what is nearline, via exploiting their own own tape volumes — a so-called "elastic tape service" that is easily expandable.

III. Early Experience with JASMIN

While selected science users had access to JASMIN directly after installation, the major task within the first year has been to migrate the CEDA archive (and small amounts of user data) into the JASMIN archive and group workspaces. That data migration has taken the best part of a year, although the bulk of the work was completed in six months. Hosted processing has been available since mid-2012, and Lotus since late 2012.

A. Data Migration Experience

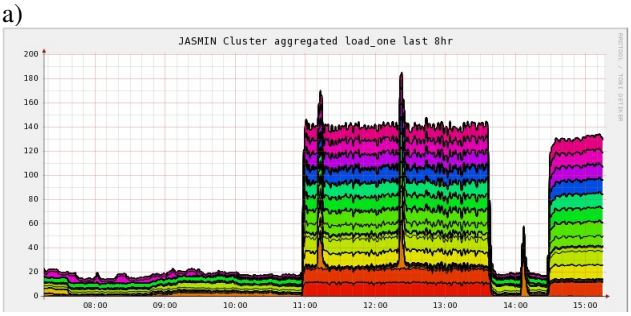
The migration of 1.2 Pb of existing archive data from legacy NAS storage to the new Panasas storage was non-trivial. The legacy NAS storage held many datasets, some fixed in size, some still growing daily, on an organically-grown plethora of 10-50TB filesystems, with filesystem boundaries not necessarily representing any meaningful division of data.

The challenge was to ensure that the data was all migrated safely into sensible filesystems, checked, and the legacy systems retired — this all in an active archive, growing and exploited daily, without the users noticing. For some datasets and filesystems, target filesystem size was not easy to predict a priori, since the final size on Panasas depended on the proportion of small (<64KB) files. The bulk of the copy operation was performed by two "worker nodes" using the Panasas pan_pcopy utility to perform parallelised copy from the legacy NAS. Performance (and number of copy nodes and threads used) during this step were limited by legacy network and filesystem issues. Bandwidths varied between 2 to 300 MB/s, leading to initial transfer times for some dataset/filesystem combinations of days to weeks. Once this operation completed for each partition, two further rsync operations "mopped up" any remaining data which for whatever reason had failed to copy in the initial operation (e.g. data ingested to the dataset during the process), before the new copy of the data was officially marked as the primary copy. Meanwhile, a tape backup of the primary copy was used to perform a bitwise comparison of the dataset to detect (and fix) any instances of data corruption (before or after transfer). Only once all these steps had been completed for each dataset and all partitions on a given legacy NAS server, would that server be disconnected from the network. To date, a total of 30 legacy NAS servers have been disconnected, with final data wiping and power down to follow. This was all done transparently to users. Future data migrations will be vastly simpler.

B. Lotus Usage

Lotus was only made incrementally available to users through the fourth quarter of 2012, and documentation is still incomplete, so usage has been limited. Nonetheless, usage has grown, with tens of thousands of data analysis jobs completed per week by May 2013. Typical Lotus jobs consume the entire small cluster, and would stress traditional (non-HPC) storage systems. Figure 3 shows Lotus performance during a satellite reprocessing job. It can be seen that the I/O load from just one host is sustaining around 3 Gb/s read, so all 8 nodes would have been demanding 24 Gb/s from the one bladeset! This job appears to have been compute bound on JASMIN, where it would have been I/O bound in a traditional

a)



b)

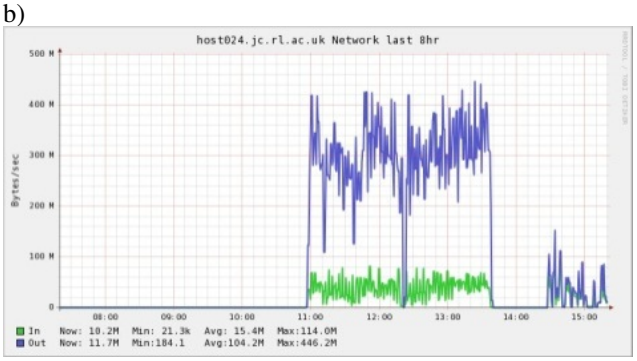


Fig. 3. Lotus usage in 2013. Panel a) show the cluster load for one specfic task during early testing, with panel b) showing the I/O from one node in the cluster (the load was homogeneous across nodes).

data analysis environment. Clearly if more compute capacity had been available, the job could have completed faster — however, even with the computing limit, the suite of jobs completed approximately one hundred times faster than the previous analysis environment. In this case it has meant that whole mission reanalysis tasks that previously took months can be completed in days, completely changing the nature of what can be done with new analysis algorithm testing and evaluation.

One of the reasons why we provided an "MPI capable" resource as well as support for more coarse grained parallelisation via virtual clusters and batch jobs on multiple hosts, was to support algorithms requiring large memory. These are often necessary to take advantage of non-linear calculations on fine resolution simulation domains. Some can best be handled by large memory servers (hence the inclusion of a large memory server in JASMIN), particularly when the calculations are complex and unlikely to be repeated. However, thus far, we have not yet explored in detail how users are taking advantage of these facilities.

C. The impact of highly parallel disk

In the original JASMIN architecture design we assumed that scientific users of the service would use virtual machines (VMs) with archive and group workspace access as replacements for servers or desktop computers in their own institutions. This has been the pattern, but as we have observed typical users, we have seen them exploiting workflows based on relatively low expectations for input/output.

Traditional NFS file servers are only able to serve a handful of processing threads, so many scripted work flows have grown up that serially feed these threads with new work. A common pattern is one machine or thread per temporal unit (often a calendar month) of a dataset, leading to utilisation of o(10) concurrent processing tasks. Access to a massively parallel file system allows orders of magnitude scale up in the number of parallel processing threads. To migrate users from traditionally scripted processing to use massively parallel I/O capability we needed a method of migrating users from a traditional server/VM environment to a higher throughput environment. We achieved this by the use of extensions to a "virtual head node" concept first developed for the STFC SCARF cluster.

Traditional high performance/throughput clusters have shared machines called "head" nodes. These allow users to login and submit jobs into the cluster but dont run jobs themselves. They provide users with editors, compilers and analysis tools. For resiliency the head node for the STFC SCARF cluster had been virtualised several years ago, so in JASMIN we were able to take advantage of that to provide communities (or even users) their own virtual head node machine running a dedicated/customised full GUI desktop. Unlike a real head node, users can install software and run code on it, both directly and via a batch queue. This allows users to test code and batch submission scripting without affecting the main physical cluster (or being limited by waiting times in the wider cluster queue). The VM batch queue is managed remotely by the physical cluster scheduler but initially only provides the internal VM resources. When testing is complete, and the software environment and batch scripts have reached sufficient maturity, the VM batch queue can be easily extended to have access to the full computational cluster, providing significant increases in resources. Any additional system software requirements will have already been tested in the same environment as the rest of the cluster (since the VM was cloned from the traditional head node), so can be quickly rolled out.

Using this approach we have found that the sub-projects maintain ownership of their workflow in their VM, still have significant flexibility, but have that coupled with the ability to scale out to the wider cluster. This means that some workflows have already moved from o(10) to o(10^4) threads.

However, at the time of writing Lotus and JASMIN have relatively small compute capacity, so users cannot gain the full benefit of massive parallelisation. We describe below a method for exploiting massively parallel computation in the current environment (using SCARF), and then subsequently, our plans for enhancing the JASMIN computational environment.

D. Exploiting SCARF

For some projects, the limited JASMIN compute can be supplemented by access to SCARF. One such project involves retrieving cloud properties from ATSR radiance measurements [7]. The retrieval algorithm involves the repeated application of an algorithm to level 1 satellite data collected in thousands of files. As such, it is an "embarrassingly parallel" task with both intensive I/O and compute.

The ideal environment for handling this would involve tasks queued onto thousands of cores connected to high performance disk - which is precisely the situation within the SCARF cluster, but not yet within JASMIN. The SCARF environment includes a heterogeneous collection of compute connected by

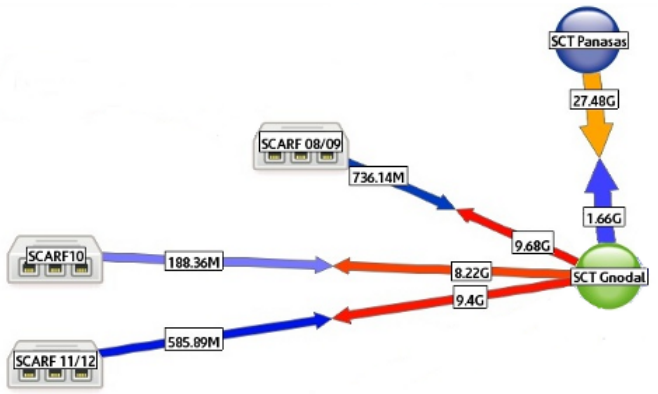


Fig. 4. Instantaneous network traffic between the SCARF computational nodes and storage during a CEMS parallel data analysis job. Colours indicate how close the traffic is to saturating bandwidth (red colours are closest). The SCARF compute nodes are behind the three grey switches.

switches to a top switch, itself connected to a local SCARF Panasas disk store, and to the JASMIN storage. Hence, the cloud retrieval workflow currently involves copying data from JASMIN storage to SCARF storage, and then the analysis involves reading and writing to the local storage within each task. Products are eventually written back to JASMIN.

A snapshot of the network performance associated with these tasks appears in figure 4. At this time, the cloud retrievals were running on about 1000 cores on approximately 100 nodes (a mix of 10 cores/node and 12 cores/node jobs). Each node has a 1 Gb/s connection to the storage. Although the rest of SCARF was busy, the I/O load was dominated by these jobs, which could have been asking up to 100 Gb/s - which is why we can see the three SCARF storage links are saturated at near 10 Gb/s. We can also see that the SCARF Panasas link (50 Gb/s) was not saturated.

Time to solution for these sort of jobs will be massively improved if enough compute is available within JASMIN: there would be no need for the starting and finishing copying steps, and more importantly, some tasks would return to being compute bound. If enough compute was available, a similar job would then be better configured to use 50 nodes, each connected at 10 Gb/s and utilising one JASMIN bladeset. In such cases we would expect the bladeset I/O capability to be balanced with the I/O demand from the compute nodes.

IV. JASMIN Futures

It is clear from the first year usage that further JASMIN expansion in both storage and adjacent compute is required by the pent-up demand from the community. However, that demand is effectively infinite: simulations can be run at higher resolution (with more output), and multiple versions of earth observation products can be compared and contrasted. Execution times can be driven downwards, but how much of that drive downwards could result from more hardware, and how much from better software tools? What is the right balance of computing to storage, and between high performance storage, and less performant (including tape) storage?

We already know that inefficient but easy to use workflow can become prevalent in the community provided resources are available. Sometimes this is the right thing to happen, since inefficient execution may be balanced by easy construction of the workflow. Similarly, experience elsewhere suggests that more than two-thirds of data stored in a simulation environment can remain unread. Such data may be unread due to a variety of factors, ranging from a non-existent user community (maybe the wrong data was produced, it is not of sufficient quality, or inadequate metadata means potential users are not even aware of the data), through to an active user community who do not have the physical, software, or human resources to exploit the data.

In terms of the curated archive, we have more than a decade of experience that suggests we can manage some of the information requirements, but that many users have been previously hindered by not having suitable resources to manipulate the data. Our one year of JASMIN experience suggests that many of these communities can be significantly aided by an expansion of physical resources, and our experience with SCARF and JASMIN shows that significantly more compute will help those communities. We have also identified new environmental science communities who will benefit from a shared data storage and analysis platform (including those providing hydrological services). To that end, we plan a significant hardware expansion over the next two years, with o(3000) cores, o(5) PB of disk, and o(10) PB of tape to be added in two phases. With the new investment, we believe we will have the right balance of storage and compute for the target communities.

Regardless of the hardware investment, we recognise that some applications will still be compute bound. In particular, we expect that both genomics and climate service applications may sporadically need more resources than we can deliver. To that end, we are also investing in the development of a "cloud broker" which will manage the allocation of resources between our private cloud and public commercial clouds. This federation layer would be an abstraction over our the application programming interfaces to our VMware service and selected commercial clouds. However, while projects such as Helix Nebula (http://helix-nebula.eu) have demonstrated utility for science applications to use public clouds, other projects report less success using commercial clouds (e.g. see the survey by [12]). For much of our work load, we expect the cost (in time and money) of moving high volume data in and out of commercial clouds to be prohibitive, hence we plan the provision of significant internal cloud resources.

Some of our community have already built workflows suitable for clouds, but many have not. It is not yet clear whether the uptake of our batch compute reflects that balance, or that we simply do not have enough cloud resource yet, and so batch computing is more efficient from a user perspective. Currently we also have an issue with our internal cloud in that we are unable to export our high volume storage with full parallel access into our cloud since we cannot yet adequately constrain access. While we are working on solutions our cloud can only make use of relatively poorly performant NFS access to the high volume storage. That too will affect the balance of use. Most of us believe that managed clusters like LOTUS or SCARF allow users to get the best performance and to get on with science rather than worrying about how to get a virtual cluster to work for them, but they are not universal views,

particularly given that some of our workload will be portal-based user-services, for whom different metrics of service and performance will be necessary. There is a tension here between the constraints of working with a shared massively parallel file system and massively scalable compute. We believe the scale of our upgrades will both allow us to serve the user communities, and to explore both sides of this debate.

Whatever the workflow users settle on, whether batch or cloud orientated, most will have to make significant improvements to their workflow to exploit massive parallelisation. We have presented an example of how such workflows can be developed within a constrained virtual environment before exploiting massive batch computing, but thus far these sort of interventions are very resource intensive. We do not have the resources to intervene on such a scale for most users, and we do not yet have suitable documentation in place to help users develop their own solutions. To that end, we are also procuring improvements in documentation and training materials.

V. Summary

We have presented the JASMIN architecture, first year of usage, and near term plans. The physical architecture consists of 600 cores and 5 PB of fast disk connected by low latency networking. The compute environment supports a range of virtualisation options, from batch to cloud computing. A diverse and growing user community is exploiting JASMIN, examples include high resolution climate modelling and whole satellite mission analysis for cloud and land surface retrievals. The use of Panasas for storage has been very successful, with flexibility, reliability, and low management overheads being key to that success. However, the existing JASMIN environment is underpowered in compute, the storage is filling, and difficulties exporting the high performance disk into the local VMware cloud computing environment remain.

JASMIN users are becoming accustomed to a new analysis environment, and early adopters are getting significant improvements in their workflow, completely changing the nature of the science they can undertake. However, thus far, the JASMIN support team has not yet been able to invest in comprehensive user documentation or training, so not all the community has seen the benefits of these investments. To fully exploit JASMIN, changes in software and workflow will be necessary for most users, and these will take time to engender.

Recognising the limitations with the existing physical infrastructure, and with new user communities anticipated, hardware, software and documentation will all be upgraded over the next two years.

References

[1] B. N. Lawrence, V. Bennett, J. Churchill, M. Juckes, P. Kershaw, P. Oliver, M. Pritchard, and A. Stephens, "The JASMIN super-data-cluster," *ArXiv e-prints*, Apr. 2012.

[2] K. E. Taylor, R. J. Stouffer, and G. A. Meehl, "An overview of CMIP5 and the experiment design," *Bulletin of the American Meteorological Society*, vol. 93, pp. 485—498, Oct. 2011. [Online]. Available: http://journals.ametsoc.org/doi/abs/10.1175/BAMS-D-11-00094.1

[3] 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. [Online]. Available: http://usymposia.upm.my/index.php/APAN_Proceedings/32nd_APAN/paper/view/155

[4] M. S. Mizielinski, M. J. Roberts, P. L. Vidale, R. Schiemann, M. E. Demory, J. Strachan, T. Edwards, A. Stephens, M. Pritchard, P. Chiu, A. Iwi, J. Churchill, C. D. C. Novales, J. Kettleborough, W. Roseblade, P. Selwood, M. Foster, M. Glover, and A. Malcolm, "High resolution climate modelling; the UPSCALE project, a large simulation campaign," *Geoscientific Model Development*, vol. In preparation, 2013.

[5] J.-P. Muller, P. Lewis, J. Fischer, P. North, and U. Framer, "The ESA GlobAlbedo project for mapping the earths land surface albedo for 15 years from european sensors," in *Geophysical Research Abstracts*, vol. 13, Vienna, 2011, p. 10969. [Online]. Available: http://www.globalbedo.org/docs/Muller-GlobAlbedo-abstractV4.pdf

[6] D. Ghent and J. Remedios, "Developing first time-series of land surface temperature from AATSR with uncertainty estimates," in *Geophysical Research Abstracts*, vol. 15, 2013, p. 5016. [Online]. Available: http://adsabs.harvard.edu/abs/2013EGUGA..15.5016G

[7] C. A. Poulsen, R. Siddans, G. E. Thomas, A. M. Sayer, R. G. Grainger, E. Campmany, S. M. Dean, C. Arnold, and P. D. Watts, "Cloud retrievals from satellite data using optimal estimation: evaluation and application to ATSR," *Atmospheric Measurement Techniques*, vol. 5, no. 8, pp. 1889—1910, Aug. 2012. [Online]. Available: http://www.atmos-meas-tech.net/5/1889/2012/

[8] J. Cohen, B. Dolan, M. Dunlap, J. M. Hellerstein, and C. Welton, "MAD skills: new analysis practices for big data," *Proceedings of the VLDB Endowment*, vol. 2, no. 2, pp. 1481—1492, 2009. [Online]. Available: http://dl.acm.org/citation.cfm?id=1687576

[9] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The hadoop distributed file system," in *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on*, 2010, pp. 1—10. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5496972

[10] H. Herodotou, H. Lim, G. Luo, N. Borisov, L. Dong, F. B. Cetin, and S. Babu, "Starfish: A self-tuning system for big data analytics," in *Proc. of the Fifth CIDR Conf*, 2011. [Online]. Available: http://x86.cs.duke.edu/~gang/documents/CIDR11_Paper36.pdf

[11] J. Buck, N. Watkins, J. Lefevre, K. Ioannidou, C. Maltzahn, N. Polyzotis, and S. Brandt, "Scihadoop: Array-based query processing in hadoop," Technical Report UCSC-SOE-11-04, UCSC, Tech. Rep., 2011.

[12] G. Sakellari and G. Loukas, "A survey of mathematical models, simulation approaches and testbeds used for research in cloud computing," *Simulation Modelling Practice and Theory*. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1569190X13000658