Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Large scale computational science on federated international grids: The role of switched optical networks

P.V. Coveney [a,*], G. Giupponi [a], S. Jha [b], S. Manos [a], J. MacLaren [b], S.M. Pickles [c], R.S. Saksena [a], T. Soddemann [d], J.L. Suter [a], M. Thyveetil [a], S.J. Zasada [a]

[a] Centre for Computational Science, Department of Chemistry, University College London, 20 Gordon Street, London, WC1H 0AJ, UK
[b] Center for Computation & Technology, Johnston Hall, Louisiana State University, Baton Rouge, LA 70803, USA
[c] Manchester Computing, Kilburn Building, The University of Manchester, Oxford Road, Manchester, M13 9PL, UK
[d] Rechenzentrum Garching der MPG, Max-Planck-Institut für Plasmaphysik, Boltzmann Str. 2, 85748 Garching b. München, Germany

ARTICLE INFO

Article history:
Received 7 July 2007
Received in revised form
16 September 2008
Accepted 22 September 2008
Available online 11 October 2008

Keywords:
Grid computing
Switched optical networks
Distributed applications
Computational chemistry
Network communications

ABSTRACT

The provision of high performance compute and data resources on a grid has often been the primary concern of grid resource providers, with the network links used to connect them only a secondary matter. Certain large scale distributed scientific simulations, especially ones which involve cross-site runs or interactive visualisation and steering capabilities, often require high quality of service, high bandwidth, low latency network interconnects between resources. In this paper, we describe three applications which require access to such network infrastructure, together with the middleware and policies needed to make them possible.

1. Introduction

We define grid computing [1,2] as distributed computing conducted transparently by disparate organisations across multiple administrative domains. The focus of grid computing has been predominantly on the provision of compute and data resources, with the bandwidth and quality of service (QoS) of the networks connecting them seen as a lesser priority, due to the fact that many users still use the resources provided on the grid as single 'island machines', running their computational code at one site and retrieving the data when it is finished. As applications emerge which seek to exploit the power of 'the grid', by making use of the resources provided in concert, for example by connecting a computational resource to a high performance visualisation machine, the performance characteristics of the networks connecting resources becomes increasingly important.

In a switched-circuit network the user has sole use of a dedicated network connection, thus eliminating contention with other traffic and providing excellent, predictable, quality of service. Switched-circuit networks can be implemented in various ways. For example, there has been much recent work on allocating users or groups sole use of individual wavelengths in multi-wavelength optical fibres. Such configurations provide contention-free links with deterministic jitter characteristics. Lambda networking involves using different wavelengths (lambdas) of light in fibres for separate connections. Lambda networks provide high-levels of quality of service by giving applications and user communities dedicated lambdas on a shared fibre infrastructure. The implementation requires Dense Wavelength Division Multiplexing (DWDM) to accommodate many wavelengths on a fibre, optical switches, and other optical networking equipment. In the UK, dedicated connections are available via the UKLight network [3] which uses manually-configured SDH circuits, which provide dedicated connections through time-division multiplexing rather than wavelength multiplexing.

Grid computing applications have so far mostly made use of best-effort, shared TCP/IP networks, that is the network has simply been the glue that holds the middleware-enabled computational resources together. In contrast, by using switched networks, the networks themselves are schedulable, "first class" grid resources. These types of high QoS switched networks form the basis of the next generation of network-centric applications. Light-paths provide several features that are not possible using regular, production, best-effort networks, but which are needed for high performance grid applications. These include higher bandwidth

[*] Corresponding author.
E-mail address: P.V.Coveney@ucl.ac.uk (P.V. Coveney).

0167-739X/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.future.2008.09.013

connections (*e.g.* [4]), user-defined networks [5], implementation of novel protocols [6] which all provide essentially contention-free, high quality-of-service links.

The purpose of this paper is to document the set of steps that it was necessary for us to take in order to use dedicated light-paths in several real large-scale scientific simulation scenarios. We describe our use of the UKLight network as a key component of the underpinning infrastructure to conduct large scale distributed scientific simulations that require simultaneous, interactive access to compute and/or visualisation resources. First, we describe our motivations for using such an infrastructure, then we go on to describe the middleware tools and policies required to make such a system usable. We discuss the benefits of using such a network based on the performance testing we carried out to evaluate the value added by the infrastructure. Finally, we describe the scientific studies that we have conducted, which have benefited from the high bandwidth and QoS provided by the UKLight network.

1.1. Motivating applications

As the power of computational resources grows, the applications that seek to exploit them become more ambitious; applications that can successfully orchestrate grid resources to build a system more useful than the sum of its parts can produce a step jump in the scope of research possible, such as the work of Pickles et al. [7]. Applications that require access to high QoS networks typically use the network to support interactive visualisation, coupled models, computational steering [8], and/or distributed applications where one model is run over a number of sites simultaneously, for example using MPICH-G2/MPI-g [9]. High performance network interconnects are essential to all three of these use cases for efficient transfer of the volumes of data created and in, the former two cases, to maintain response times required for interactive access. In addition to the applications discussed above, high quality of service networks also have more mundane but equally important uses, for example to quickly ship large amounts of data created at single grid sites back to a user's local resources for further analysis. Multi-site distributed simulations using MPI-g and light-path networks build on a breadth of earlier work on running simulations across multiple super computing resources, from the CASA [27] testbed linking HPC sites in the early 1990s, through the I-WAY [28] demonstrations and beyond.

Scientists who require interactive access to machines, for example for steering and visualisation, as well as cross-site applications, also need to be able to schedule time on individual resources – both compute and networking – as well as tools to allow them to easily co-reserve those resources, so that all the required resources are available when they are needed. This, in turn, leads to a demand on resource providers to implement policies and tools that allow such reservations to be made as and when required, so that such methodologies can be incorporated into a user's normal research activities, rather than just providing such facilities on an *ad hoc* basis.

Moreover, the resources provided by a single grid may not be sufficiently powerful or appropriate to run large scale distributed models, and resources provided by multiple grids have to be federated in order for a particular investigation to be conducted. This conflates the problems of interactive resource use; each grid has its own policies and systems for making advanced reservations, if it has any at all. Compounding this, the high performance network provision between grids may also be limited or non-existent. Nevertheless, such obstacles must be overcome to make efficient use of available federated resources.

2. Middleware requirements

As the number of resources provided by a grid grows, end users face increasing difficulty in keeping track of the resources available to them, the policies and tools for making advanced reservations on these resources, and the diverse grid middleware and the access mechanisms for job submission. In this section, we present three tools designed to allow the user to interact with (federated) grid resources and to make cross site advanced reservations of CPU time in a transparent manner.

2.1. The application hosting environment

The Application Hosting Environment[1] (AHE) [10] is a lightweight mechanism for representing scientific applications as stateful WSRF web services [11], and allowing users to interact with those applications using simple client tools. The AHE enables the launching of hosted applications on a variety of different computational resources, from national and inter-national grids of super computers, through institutional and departmental clusters, to single processor desktop machines. It does so transparently, meaning that the end user is presented with a single interface and access mechanism to launch applications on all of these resources. The AHE also provides mechanisms for file transfer and job management. File transfer allows users to move input and output data between their desktop and target computational resource. Job management allows users to monitor and terminate applications while they are running.

The AHE takes a service oriented approach and is currently a client of the GridSAM job submission web service [13] developed at Imperial College London. AHE launches an instance of a hosted application by submitting a Job Submission Description Language (JSDL) [12] document which describes the application being run to GridSAM. GridSAM, features connectors on to several different distributed resource management systems (DRMs) including Globus 2 [19] and UNICORE [20]. In the case of the GridSAM Globus connector, GridSAM translates the JSDL job description into Globus Resource Specification Language, which it then submits to the Globus Gatekeeper running on the target resource. In the case of UNICORE, GridSAM translates the JSDL into a UNICORE Abstract Job Object. GridSAM is responsible for staging files in to and out of the grid resource that it is providing an interface to, and for managing the execution of the job on that resource. The AHE interacts with GridSAM to monitor the status of the application it has launched.

The AHE web services are developed in Perl using the WSRF:Lite toolkit [14]. The services run inside a container (usually Tomcat using a modified version of the standard Tomcat CGI servlet in order to run the WSRF::Lite services). The AHE also makes use of a file-staging service, typically a WebDav server. Users upload files to this service, from where they are further staged to the grid resource by GridSAM. The service also holds data being staged back to the user by GridSAM. This is a result of two design constraints on the client: (a) that the user should not have to open any incoming firewall ports in order to use the client and (b) that the user does not need to maintain installations of file transfer software on their client machines (for example GridFTP); the AHE encapsulates details of the underlying grid middleware in use. For a fuller discussion of the architecture and how the different technologies used interact see [15].

Since its initial release in March 2006, we have worked to extend the reach of the AHE server centrally deployed at University College London, in addition to supporting users with their own

[1] http://www.realitygrid.org/AHE.

deployments of the server. Currently the UCL AHE deployment is configured to submit jobs to federated resources from the US TeraGrid [43], UK National Grid Service (NGS) [16] (which both run the Globus middleware), and EU DEISA grid [17] (which runs the UNICORE middleware). The AHE provides transparent user access to all three of these grids, giving a single interface to submit jobs to both UNICORE and Globus. This is made possible due to agreements between the different resource providers to recognize certificates issued by each others' Certificate Authorities (CAs). For a full discussion of the use of AHE to access federated international grid resources see [18].

2.2. GridSAM plug-in for UNICORE

While interoperability for grid users is a natural requirement, it is a complex problem for middleware deployers, due to the different architectural approaches taken by the main grid middleware providers. From version 2 onwards Globus [19] is organised as an extensible toolkit from which components can be selected and used as required. Conversely, UNICORE tries to deploy a vertical solution whose components are all tailored to fit together, and a detailed knowledge of the Network Job Supervisor (NJS) [20] is required to interface to the UNICORE server.

The Open Grid Forum [21] tries to address the problems of interoperability by recommending standards to the grid community. The most prominent recommendation is the Open Grid Services Architecture (OGSA) [22] which combines standardisation efforts in the realm of compute job handling as well as in the realms of information access and data. Although it is planned that future releases of grid middleware projects, such as the Globus Toolkit and UNICORE, will be OGSA compliant, there is pressure from the end-user community to provide inter-operability now.

So-called Science Gateways [23] are one way of achieving interoperability. Here, users interact with web applications which present the functionality of underlying applications via a web portal and allow users to formulate tasks and submit jobs using their web-browser. With the emergence of the JSR-168 specification, the so-called portlet specification [24], it becomes much simpler to deploy the same web application (or a part of it — the portlet) in different contexts for different underlying grid infrastructures. Hence, the user employs the same web interface regardless of the underlying infrastructure. In order to connect such a web application to UNICORE, within the DEISA project [17] the Job Management Enterprise Application (JMEA) [25,26] was developed by Rechenzentrum Garching (RZG). JMEA is a Java enterprise application which allows clients such as web applications or web services to submit jobs to and manage them with UNICORE [29].

GridSAM can be seen as a framework for creating a job submission and monitoring service. It is already shipped with support for various underlying resource management systems and provides a set of web service methods for the submission and monitoring of jobs. These are implemented in Java, employing the Apache Axis 1.x web service stack [30] and the Hivemind microkernel [31]. Job submissions have to be formulated in the Job Submission Description Language (JSDL), an Open Grid Forum standard. This ensures interoperability, at least on the level of user job submission. As a framework, GridSAM offers extension points which admit a customisation for a given infrastructure or environment, by providing a plug-in mechanism that allows new Distributed Resource Manager (DRM) connectors to be developed. GridSAM currently ships with support for CondorG [32] and Globus [19], but lacks support for UNICORE. To address this shortcoming, we have used JMEA as the basis for a GridSAM UNICORE DRM connector.

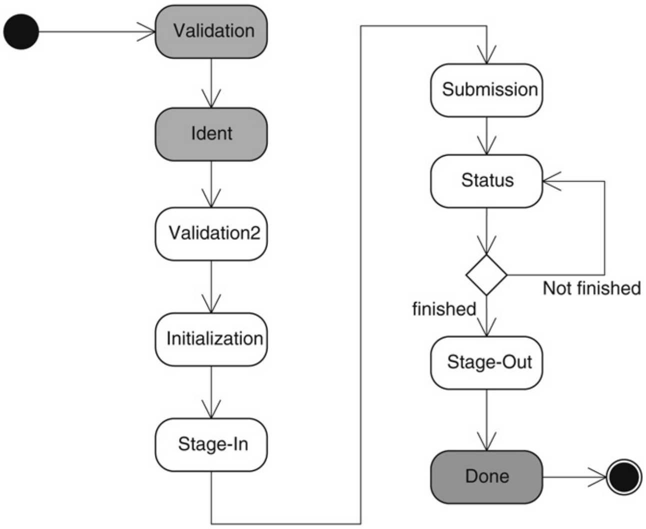


Fig. 1. State diagram of the GridSAM resource management connector. White states indicated places where GridSAM has been extended to support UNICORE.

GridSAM plug-ins have to extend an abstract class in order to define the sequence of stages which need to be executed once a job arrives at GridSAM. Fig. 1 sketches the different stages a job traverses from submission to its end as states in a UML-like state diagram. In the first validation stage, the general validity of the submitted JSDL document is checked. Next, in the identification stage, a job request is created within GridSAM and the job gets an identifier. It then encounters the first UNICORE specific stage (Validation2) where a check on the JSDL document is performed to ensure that it contains all fields necessary to submit a job to a UNICORE based infrastructure. Then, the UNICORE job is initialised by creating a temporary job staging area at the submission site. All the files to be staged, as defined in the JSDL document, are then transferred to this staging area using GridFTP [33]. In the submission stage, the remainder of the JSDL document is translated into UNICORE's Abstract Job Object (AJO) format. The resulting AJO is then submitted via JMEA to UNICORE. The job within GridSAM next enters the status stage where its status is periodically monitored. Since JMEA takes care of the monitoring, GridSAM's monitoring information is directly retrieved from JMEA. Once the job has finished, results are staged out to the locations specified initially and the "done" stage is entered.

For interaction with UNICORE, the so-called Explicit Trust Delegation model [34] is employed, which allows an agent (JMEA in this case) to submit jobs on behalf of a user, who is trusted to be authenticated by the agent. In order to identify a user, GridSAM had to be extended to make the information about the authenticated user available at the submission stage for every supported security model. Since UNICORE requires a user certificate to be sent along with the AJO, a certificate pool has to be employed by the GridSAM connector to JMEA, for use when the GridSAM container is using point to point security (HTTPS).

2.3. HARC: The highly-available resource co-allocator

HARC, the Highly-Available Resource Co-allocator, is an open-sourced system that allows users to reserve multiple distributed resources in a single step [36–38]. These resources can be of different types, including multiprocessor machines and visualisation engines, dedicated network connections, storage, the use of a scientific instrument, and so on. Currently, HARC can be used to book high performance computing resources, and light-paths across certain networks based on GMPLS (Generalised Multiprotocol Label

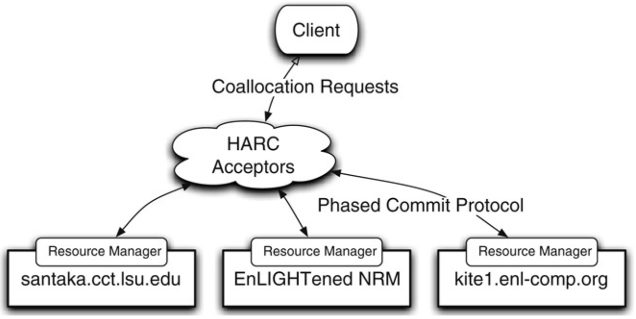


Fig. 2. The HARC architecture, showing the relationship between the client, the acceptors, and the resource managers (RMs).

Switching; see for example [72]) with simple topologies. The HARC architecture is shown in Fig. 2. While not directly related to our work on HARC, we acknowledge previous work on bookable VPNs using ATM networking, such as those described in [35].

HARC uses a phased commit protocol to allow multiple resources to be booked in an all-or-nothing fashion (i.e. atomically). Paxos Commit [39] is used, rather than the classic 2-Phase Commit (2PC), to avoid creating a single point of failure in the system. Paxos Commit replaces 2PC's single transaction manager (TM) with a number of processes, or *Acceptors*, which perform the same function as the TM. The Paxos Consensus algorithm guarantees consistency, so clients can talk to any Acceptor to find the results of their requests. The overall system functions normally provided a majority of Acceptors remain in a working state. This gives a deployed system of five Acceptors a far longer Mean Time to Failure than that of any single Acceptor.

HARC is designed to be extensible, and so new types of Resource Manager can be developed without requiring changes to the Acceptor code. This differentiates HARC from other co-allocation solutions. The assumption is that the underlying resource has a scheduler capable of reserving the resource (or part thereof) for a specific user; the RM amounts to a small piece of code that interacts with this scheduler on the user's behalf.

The HARC Compute Resource Manager currently supports the schedulers PBSPro, LoadLeveler, Torque/Maui and Torque/Moab, but can be adapted to work with any scheduler that supports user-settable advance reservations. HARC has been demonstrated successfully, for example at iGrid2005 [40], at GLIF 2006 and SC 06,[2] and, more recently, was used to schedule some of the optical network connections being used to broadcast Thomas Sterling's HPC Class, broadcast live in High-Definition Video from Louisiana State University.[3]

There are two deployments of HARC in use today: the EnLIGHTened testbed in the United States [41] and on NorthWest Grid [42], a regional Grid in England. A trial deployment is underway on the TeraGrid, and HARC is being evaluated for deployment on the UK National Grid Service. A Network Resource Manager that interfaces to the ESLEA Circuit Reservation Software [44] is also being considered. This would allow HARC to be used to co-allocate parts of the UKLight network.

3. Performance testing of UKLight

Currently the UKLight network provides a fast connection between the Centre for Computational Science at University College London and various supercomputing resources such as HPCx [45] and the TeraGrid. We carried out a series of tests on the performance of the link between UCL and HPCx, as well as UCL and the TeraGrid. Preliminary results showed that the link was not as fast as expected. Two methods could have been used in order to improve the bandwidth of the link. The first is to use GridFTP in order to invoke multiple streams over one link. The problem with this method is that it is difficult to implement on a network such as UKLight. This led us to use common software such as SSH, along with network tuning, to achieve maximum bandwidth over a single stream. Specifically, we needed to tune the TCP window size in order to achieve maximum bandwidth. This section describes the methodology we used to test the system and the results we obtained once network parameters were tuned.

3.1. Methodology

Two dedicated network testing Linux systems with gigabit ethernet cards were set up, giving us the freedom to experiment on non-production machines. The first connection studied was between a Linux box connected to the same UKLight switch as UCL's SGI Prism visualisation engine and a machine connected to the UKLight switch of HPCx. The second connection was between UCL and the TeraGrid's IA-64 Linux cluster at NCSA. NCSA's network parameters were already tuned, so a separate machine was not needed for testing there.

Iperf [46] is a network tool which was used to test UDP and TCP bandwidth between networked computers. Iperf UDP tests utilised the maximum bandwidth of the connection before packet loss was seen. This test was carried out on both the production network and UKLight. In all UDP tests the grid resource acted as the client while the UCL Linux machine was the server. The results of the UDP tests can be used to calculate the bandwidth delay product (BDP) of the link, which is defined as the product of the bandwidth and the round trip time. The round trip time is the time elapsed for a message to travel to a remote site and back again. The BDP helps determine the maximum window size for TCP communication.

In order to get the best performance from a network, the TCP window size defined on the kernel and application side needs to be tuned. The Linux kernel parameters which we needed to adjust were as follows:

/proc/sys/net/core/wmem_max
/proc/sys/net/core/rmem_max
/proc/sys/net/ipv4/tcp_rmem
/proc/sys/net/ipv4/tcp_wmem

In addition, we used an application called High Performance Enabled SSH/SCP (HPN-SSH) [47]; this is a patch for recent OpenSSH releases which allows adjustment of the TCP window size within the application. With these changes in place, we then tested the performance of SCP on all connections, including the production network and UKLight with untuned and tuned network parameters.

3.2. Results

The UDP tests showed that the connection between UCL and other grid resources could be as high as 40 Mbytes/s, as summarised in Table 1 (this is 7% greater than, but is within 10% of, the provisioned link speed, a discrepancy due to the specifics of that network link). The TCP tests showed that a maximum of 34.4 Mbytes/s could be achieved, as shown in Table 2. Using these parameters, the Linux kernel parameters were tuned and HPN-SSH was used to compare the data transfer rates for the production network, as well as the untuned and tuned UKLight network. As summarised in Table 3, a massive improvement can be seen using tuned network parameters. Using untuned parameters we

[2] See http://www.gridtoday.com/grid/884756.html.

[3] See http://www.cct.lsu.edu/news/news/201.

Table 1
Results for UDP tests of UKLight connection between UCL and grid resources. The grid resource acted as the Iperf client while the UCL Linux machine was the server.

Grid resource	Maximum bandwidth (Mbytes/s)	Round trip time (ms)	Bandwidth delay product
NCSA	40	92	3.2 MB
HPCx Linux	32	8	300 kB

Table 2
Results for TCP tests of UKLight connection between UCL and grid resources. The grid resource acted as the Iperf client while the UCL Linux machine was the server.

Grid resource	Maximum bandwidth (Mbytes/s)	Maximum window size (MB)
NCSA	34.3	4
HPCx Linux	34.4	3

Table 3
Comparison of networks using the maximum bandwidth obtained from SCP. A substantial improvement is seen using UKLight over the academic Super Janet network. No parameter tuning was performed on the Super Janet network.

Grid resource	Maximum bandwidth (Mbytes/s)		
	Janet network	UKLight (untuned)	UKLight (tuned)
NCSA	0.6	0.7	16
HPCx Linux	4.5	8	28

obtained 4.5 Mbytes/s over the standard production network and 8 Mbytes/s over the UKLight dedicated connection with HPCx. Using tuned parameters this increased to 28 Mbytes/s over the Uklight connection. This pattern was repeated for the NCSA connection, giving 0.6 Mbytes/s untuned and 16 Mbytes/s tuned. The notional line-rate of the link in use was 37.5 Mbytes/s.

4. Exploiting the infrastructure

To illustrate the utility of high performance, low latency, high bandwidth network connections when conducting large scale distributed scientific investigations, we describe three studies of applications which require cross site resource allocation, transparent access mechanisms and high QoS network interconnects between machines.

4.1. *DNA translocation using NAMD*

The transport of bio-molecules such as DNA, RNA and poly-peptides across protein membrane channels is of primary significance in a variety of areas. Although there has been a flurry of recent activity, both theoretical and experimental [48,49], aimed at understanding this crucial process, many aspects remain unclear.

Of the possible computational approaches, classical molecular dynamics (MD) simulations of bio-molecular systems have the ability to provide insight into specific aspects of a biological system at a level of detail not possible with other simulation techniques. Indeed, MD simulations can be used to study details of a phenomenon that are often not accessible experimentally [50] and would certainly not be available from simple theoretical approaches. However, the ability to provide such detailed information comes at a price: MD simulations are extremely computationally intensive — prohibitively so in many cases. As was discussed by Jha et al. [51], advances in both the algorithmic and the computational approaches are imperative to overcome such barriers.

SPICE, the Simulated Pore Interactive Computing Environment project [51], implements a method, henceforth referred to as SMD-JE, to compute the free energy profile (FEP) of nucleic acid translocation along the vertical axis of an alpha-hemolysin protein pore. This method reduces the computational requirement for the problem of interest by a factor of at least 50–100, at the expense of introducing two new variable parameters, with a corresponding uncertainty in the choice of their values. The computational advantages are maintained by performing a set of "preprocessing simulations" which, along with a series of interactive simulations, help inform an appropriate choice of the parameters. To benefit from the advantages of the SMD-JE approach and to facilitate its implementation at all levels – interactive simulations of large systems, the pre-processing simulations and finally the production simulation set – we use the infrastructure of a federated trans-Atlantic grid [52].

Interactive simulations involve using the visualiser as a steerer, for example to apply a force to a subset of atoms, shown in Fig. 3(b), and require bi-directional communication — there is a steady-state flow from the simulation to the visualiser as well as from the visualiser to the simulation. As a consequence of requiring geographically distributed resources, high-end interactive simulations are dependent on the performance of the network between the scientist (at the visualiser) and the remote simulation. Unreliable communication leads not only to a possible loss of interactivity but, equally seriously, a significant slowdown of the simulation as it waits for data from the visualiser.

On switching traffic flow from the production network to a high QoS network, we found an improvement in the performance (measured as wall clock per each MD time step) of around 50%. The simulation performance over the production network varied, that is it was sensitive to prevailing network conditions. Interactive MD simulations thus require high quality of service (as defined by low latency, jitter and packet loss) networks to ensure reliable bi-directional communication. Such large-scale interactive computations thus require both computational and visualisation resources to be co-allocated with networks of sufficient QoS [52].

4.1.1. *Experiment*

In order to quantify the impact of network performance characteristics on the efficiency of an interactive MD simulation a series of measurements were made under controlled conditions. The full details of our methodology are described in [53] and are presented here in outline.

The two compute resources on which the molecular dynamics simulation (using NAMD [54]) and visualiser were run (VMD [55]), were connected via a dedicated circuit on the UKLight network, provisioned at 300 Mbps.

To control the characteristics of the network, a third system was introduced between the visualiser and the UKLight link. This system acted as an IP bridge and employed the NISTnet [56] package to modify the traffic flow. The wall-time per simulation timestep (t_s) was measured for interactive simulations over a range of network characteristics controlled by NISTnet. The parameters varied were (1) Packet transmission delay (α), to simulate different latency network paths, and (2) Packet Loss (β), to simulate packet loss due to network congestion.

4.1.2. *Quality of service: Latency*

The effective latency of the UKLight link was varied to emulate different paths, service times and congestion — all characteristic of best effort networks between two given end points.

Not surprisingly, the time taken for each timestep increases linearly with greater latency. Our results are plotted in Fig. 4. It is interesting to note that, although the average wall-time taken per simulation time-step is of the order of hundreds of milli-seconds, introducing latencies of a few milli-seconds has a significant effect. This is attributed to the fact that a significant fraction of the simulation time is spent waiting for I/O operations to complete. Thus we conclude that reducing any avoidable latency is a good performance enhancing strategy.

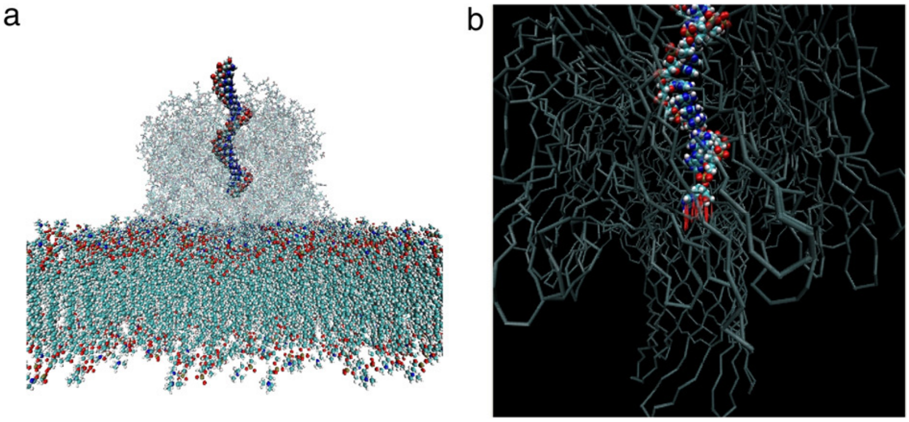


Fig. 3. (a) shows a snapshot of a single-stranded DNA polymer beginning its translocation through the α-hemolysin protein pore which is embedded in a lipid membrane bilayer. Water molecules are not shown. (b) shows an interactive steered molecular dynamics simulation in progress. The red arrow-headed lines represent the forces that are applied to the end residues of the DNA to guide and speed up its translocation through the pore. Information about the forces are sent from the visualiser to the simulation; this is used by the simulation to compute the updated configuration, which is then sent to the visualiser. Reproduced from [51]. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

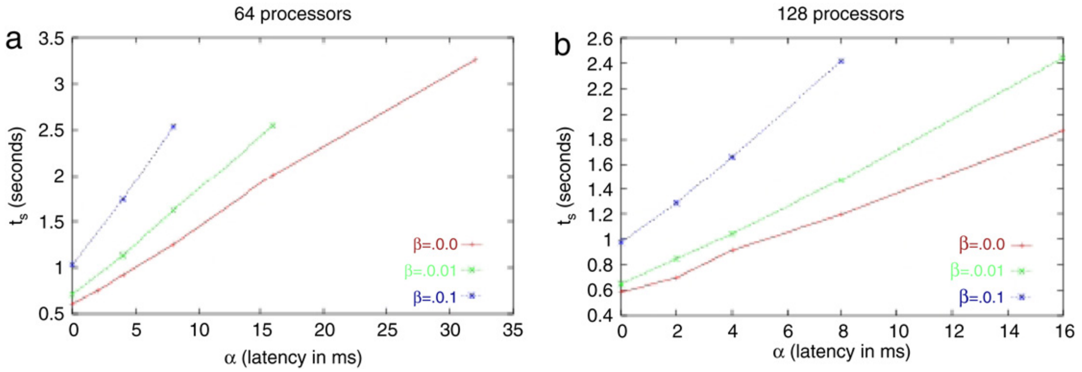


Fig. 4. Plots showing the effect of latency on the performance (t_s) with (a) 64 processors and (b) 128 processors. An increase in latency leads to a linear increase in the wall-time taken per simulation timestep, independent of the number of processors used. The performance degradation remains linear for different values of β.

4.1.3. *Packet loss effects*

Packet loss is interpreted by the TCP protocol as an indication of congestion and causes the window size to be immediately reduced to a minimum (4096 bytes in this case) and then renegotiated up. The effect of increasing β is to increase the frequency of window size reduction (reducing the average size over time) and consequently reducing effective throughput. We observe that, in general, default settings for TCP window size parameters are unsuited to networks with high bandwidth-delay products (*cf.* Section 3.2).

4.2. *Amphiphilic fluid dynamics using LB3D*

In this section, we describe our lattice-Boltzmann, using the LB3D code, simulations that were supported by the high bandwidth, low latency, high QoS UKLight network. We first describe the scientific motivation for our work and then describe the main features of the LB3D code that determine our network requirements. Finally, we discuss a new meta-computing approach called geographically distributed domain decomposition (GD^3) for which high bandwidth, low latency, reliable network connections are critical, and the technical challenges in deploying these simulations on transatlantic grids connected via UKLight.

4.2.1. *Scientific motivation*

We perform large-scale lattice-Boltzmann simulations of complex amphiphilic fluids. Amphiphiles are molecules which have a lipophilic (possessing an affinity for oil species) tail group and a hydrophilic (possessing an affinity for water) head group, so that when immersed in a mixture of oil and water, the molecules tend to align themselves at the interface with their head- and tail- groups oriented towards the water and oil domains, respectively. Amphiphilic systems, composed of at least one amphiphilic species, are of immense industrial importance, mainly due to the tendency of amphiphilic molecules to align at interfaces in a solution of immiscible species thereby self-assembling into a variety of cubic and non-cubic morphologies. Our lattice-Boltzmann method, implemented in the Fortran 90 code, LB3D, correctly simulates such self-assembly in ternary amphiphilic systems composed of oil, water and amphiphiles. In Fig. 5, we see the isosurface rendering of the colour order parameter, $\phi(\mathbf{r})$, for a self-assembled (i) non-cubic, periodic lamellar phase, (ii) bicontinuous, cubic gyroid phase and (iii) bicontinuous, random sponge phase. LB3D has been used to study the self-assembly dynamics of the gyroid mesophase and has yielded predictions of a power-law scaling behaviour of the average squared mean curvature indicative of diffusive scaling on the same length-scale as the gyroid domains within the system [57]. We are also investigating the rheological behaviour of self-assembled amphiphilic mesophases by imposing a stationary

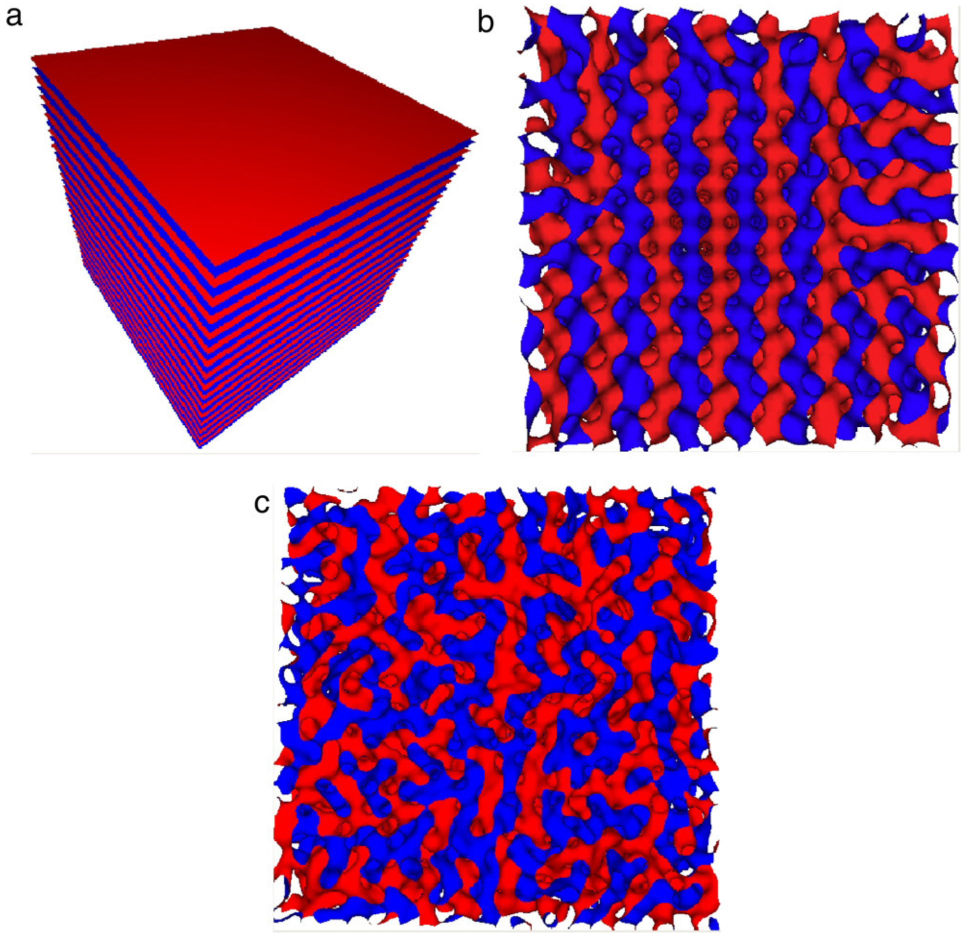


Fig. 5. Lattice-Boltzmann simulations, performed using LB3D, of the self-assembly of a ternary amphiphilic mixture from an initially random phase into (a) non-cubic lamella phase, (b) bicontinuous cubic gyroid phase and (c) bicontinuous sponge phase.

Couette flow, see Fig. 6. Complex visco-elastic behaviour for the gyroid phase is observed in our rheological simulations — in particular shear-thinning has been reported under an imposed steady Couette flow while a transition from liquid-like to solid-like behaviour is observed under oscillatory Couette flow [58]. Such self-assembled mesophases are finding applications in nanomaterials synthesis in designing materials with specific functional properties, as well as various applications in biotechnology. The main objective of our current work is to study the rheological properties of very large, defect-containing gyroidal systems (of up to 1024^3 lattice sites) using the lattice-Boltzmann method.

4.2.2. Main features of the LB3D code

LB3D has been under development for over 7 years and is widely deployed on the UK NGS including HPCx and on the US TeraGrid. Given sufficient computational resources, LB3D can be used to study grand challenge problems in mesophase fluid dynamics. The simulation algorithm in LB3D consists of a three-dimensional lattice-Boltzmann solver for multi-component fluid dynamics [59] on a regular computational lattice. LB3D has been parallelised using the domain decomposition scheme [60] using MPI (Message Passing Interface) and scales linearly up to at least 1024 processors. It correctly simulates the self-assembly dynamics [57] and rheology [58] of cubic and lamellar mesophases in ternary amphiphilic systems. LB3D is a memory intensive application code requiring approximately 1 kB of memory per lattice-site to store state data. The compute-intensive part of the algorithm consists of the collision and propagation steps. Because of the non-local interaction forces between different species in the amphiphilic mixture, two communication steps per cycle are required to exchange state data for lattice-sites on the sub-domain boundaries between neighbouring processors. LB3D checkpoints system state and visualisation data-sets at regular intervals. For models of size 1024^3 LB3D requires 1.07 TB of total memory to run; writing checkpoint files requires disk space of the order of Terabytes. Each visualisation step requires emission of a 4.3 GB visualisation dataset which is transferred over UKLight and rendered by a high performance visualisation resource. Network performance and availability of lambda networks is hence essential for us to overcome challenges in data transfer and storage, visualization and computation [58,61].

4.2.3. Geographically distributed domain decomposition

The large amount of memory required to run these simulations is often not available to us on a single supercomputer. We have pursued a new network-intensive meta-computing approach called geographically distributed domain decomposition or GD^3 [62] that can overcome this memory bottleneck. In this approach, a single MPI simulation is split across processors on geographically distributed supercomputers, specifically, HPCx (and previously CSAR) in the UK and TeraGrid in the US. Network characteristics, such as bandwidth, latency and reliability during the simulation run, are all critical in the GD^3 approach. The grid middleware that we use to launch cross-site GD^3 simulations is called MPICH-G2 [9] and its newer pre-release version called MPI-g.

Our initial aim is to split the 1024^3 lattice-sites LB3D simulation across resources on the US TeraGrid and HPC resources in the

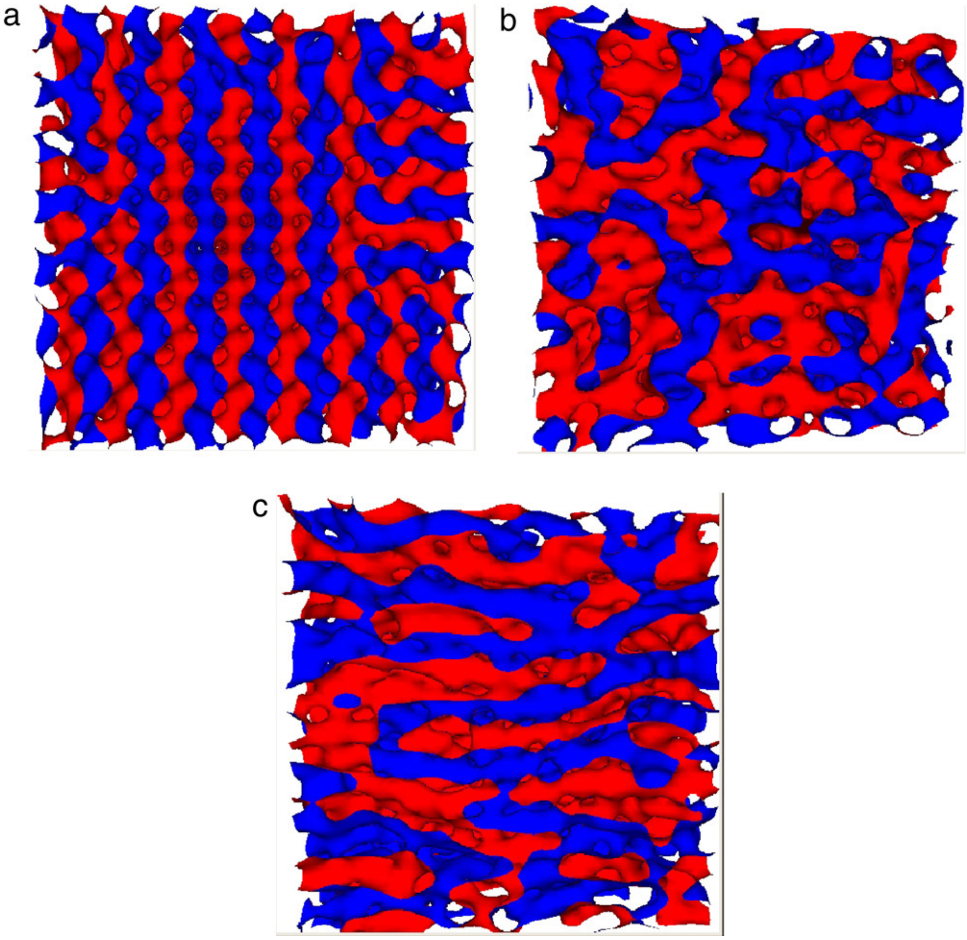


Fig. 6. Lattice-Boltzmann simulations, performed using LB3D, of the rheology of a gyroidal ternary amphiphilic mixture at imposed shear strain rates (a) $U = 0$, (b) $U = 0.1$ and (c) $U = 0.2$ in lattice units. We observe non-Newtonian rheological behaviour, in which the stress response varies non-linearly with applied strain rate, for the gyroid system.

UK connected via UKLight and the StarLight network [63] to the TeraGrid optical backbone. From a scientist's perspective there are many technical challenges that need to be overcome in order to efficiently run cross-site simulations. The simulation code needs to achieve maximum overlap between computation and communication by taking advantage of MPI's asynchronous communication calls. Unlike the previous MPICH-G2 version, MPI-g implements asynchronous communications and is well-suited to take advantage of latency hiding optimisations in the code. Moreover, the UDT (UDP-based Data Transfer) communication protocol has been proposed for future versions of the grid middleware instead of the TCP protocol currently used (UDT is based on the UDP protocol but with added reliability). This is estimated to improve cross-site performance by a factor of two [62]. In order to be included in an MPICH-G2/MPI-g cross-site framework, the participating machines need to be suitably configured to have externally addressable nodes and compatible firewall policies. This poses a problem for relatively less grid-enabled machines such as the Cray XT3 machine (Bigben) at the Pittsburgh Supercomputing Center, HPCx and the forthcoming flagship UK supercomputer, HECToR [64].

Within the Vortonics project at Supercomputing Conference 2005 [65], a trans-atlantic cross-site run, over UKLight on TeraGrid machines and the now decommissioned Newton machine at CSAR on the UK NGS [16], was performed by Boghosian et al. using another lattice-Boltzmann simulation code which has a similar communication pattern to LB3D [62]. In situations where the memory requirements of the simulation are too large to fit onto a single supercomputer, their results provide support for the viability of GD^3 as compared to alternatives such as swapping portions of the simulation to disk or, worse, waiting for a larger machine to become available.

4.3. Large scale atomistic clay simulations

Large scale atomistic simulations, which we define as containing more than 100,000 atoms, provide a bridge between atomistic and mesoscale simulations [66–68]. Operating over at least tens of thousands of atoms, emergent mesoscopic properties are observed in full atomistic detail. This is especially important for systems that are highly anisotropic, possessing behaviour in different spatial directions that operate over vastly different length scales. An example is a clay platelet, where the lateral dimensions are commonly found in the micron range yet the thickness of the sheet is as small as nanometers [66]. The resulting high surface area is very useful; polymers containing dispersed aluminosilicate clay sheets such as montmorillonite and layered double hydroxides (LDH) are found to have increased thermal and mechanical properties. These are examples of nano-composites, where at least one of the components possesses a dimension in the nanometre range [67].

It is often difficult to predict how nanocomposites will behave from theories of conventional composite behaviour due to the disparity of dimensions; hence the need for large-scale molecular simulation to sample all possible length scales. To model aluminosilicate clay sheets we perform many simulations at various system sizes; the largest to date approaches that of a realistic clay platelet and contains almost ten million atoms. Such large scale simulations are unprecedented in aqueous mineral simulation; using computational grid resources allows the turnaround on these many simulations to be on a feasible timescale for scientific research. The grid resources we need to use are, however, distributed within many geographically separate grids: the US TeraGrid, the UK National Grid Service and the EU DEISA grid. We use the Application Hosting Environment (AHE) as the middleware system to federate these grids. AHE allows easy and transparent access to grid resources by providing a uniform interface to run applications on multiple resources with different underlying middlewares. AHE can be used to simultaneously access, for example, applications running on both the US TeraGrid and EU DEISA grid, even though US TeraGrid uses Globus as the underlying middleware as opposed to UNICORE for DEISA (*cf.* Section 2.

4.3.1. *Findings*

Previous large-scale simulations of intercalated clay-polymer nanocomposites revealed long wavelength, low amplitude thermal undulations [69]. The undulations are not observed in smaller clay models; indeed it often assumed that clay sheets are rigid bodies. Small simulation sizes implicitly inhibit long wavelength clay sheet flexibility due to the periodic boundaries used in condensed matter molecular dynamics, which effectively pin the clay sheet at the edges of the simulation cell. Through the resources available, we have been able to simulate clay platelets of sufficient size that we can extract materials properties such as the bending modulus using the techniques similar to those used for analysing the thermal fluctuations of biological membranes [70]. A first step towards understanding the material properties of a composite is to appreciate the materials properties of each component; for montmorillonite (see Fig. 7) and LDH (see Fig. 8) this value is uncertain [68]. The simulations exhibit collective thermal motion of clay sheet atoms over lengths greater than 150 Å. From the thermal bending fluctuations we calculate material properties. We estimate the bending modulus for montmorillonite to be 1.6×10^{-17} J, corresponding to an in-plane Young's modulus of 230 GPa and 1.0×10^{-19} J and 138 GPa for the bending modulus and in-plane Young's modulus respectively for LDH. These values can then be used in composite-theory models to help predict and understand the mechanical enhancement of clay-polymer nanocomposites.

4.3.2. *Exploiting the infrastructure*

To effectively exploit this federation of grids described above, we must be able to move files between the grid resources quickly and easily. For example, checkpoint files are moved to restart simulations on different grids and data files are shipped back to local machines for subsequent analysis. Although no interactive use was required in this study, high performance and high bandwidth network interconnects between computational resources were required as the volume and size of files produced was of the order of 100 GB. Future work aims to couple these fine grained atomistic simulations with mesoscale coarse-grained simulation methods such as the elastic membrane model [71] and coarse-grained molecular dynamics. Information passed between the two components of the coupled model can be handled by AHE, reducing the complexity of managing these applications for the user. Such coupled models will require co-allocation of resources and efficient transfer of data between the coupled codes.

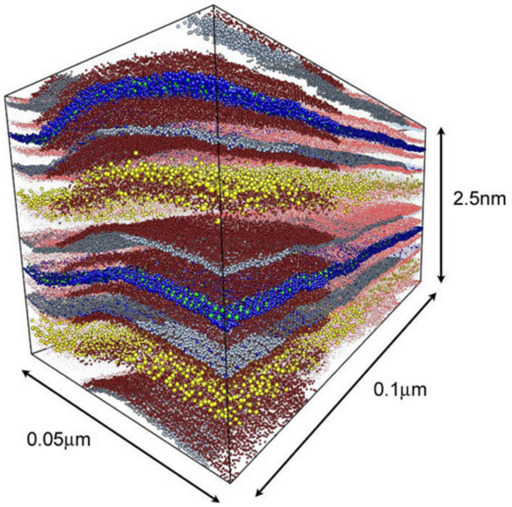


Fig. 7. Snapshot visualisation of a sodium montmorillonite system, containing 1,055,000 atoms at 1 ns of simulation. Spatial directions and dimensions are indicated in the figure. The *z* direction has been expanded by 15 times. The thermal undulations, through which we can determine the material properties of the clay, can be clearly seen. Atoms are coloured as follows: Mg = green, Al = blue, O = red, Si = grey, Na = yellow, H = white. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

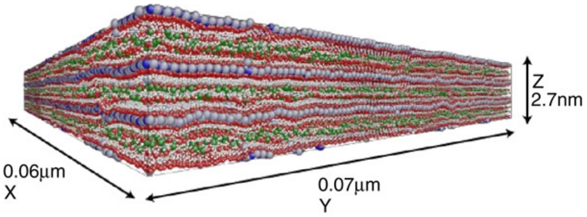


Fig. 8. Snapshot visualisation of Mg, Al-layered double hydroxide containing 1,026,432 atoms at 1.5 ns of simulation. The spatial directions and dimensions are indicted in the figure. The *z* direction has been expanded by 3 times to assist viewing of the thermal undulations. Atoms are coloured as follows: Mg = grey, Al = blue, O = red, Cl = green, H = white. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

5. Conclusions

Not only can grids use light-paths to more closely integrate distributed environments, they *must* use light-paths if the full power of grid computing is going to be realised for the effective execution of many high-end computing projects. SPICE provides an example of a large-scale problem that depends on using algorithms amenable to distributed computing techniques and then implementing them on grids. In order to effectively utilise these algorithms, interactive simulations on large computers are required. In order to enable meaningful interactive exploration, the responses must be computed in reasonable times. Thus as larger systems are studied [73] not only will larger computers be required, but the need for efficient and reliable communication will also grow.

The wide range of scientific work reported here that exploits the UKLight infrastructure and international federated grid resources allows us to draw a number of conclusions. It is clear from our performance testing results that UKLight makes a significant impact when transporting large files from grid resources. In order to achieve the maximum bandwidth of this link over a single stream, network parameters must be tuned. The results also highlight the fact that UKLight provides an efficient way to transport data for more complicated purposes such as real-time visualisation and computational steering [51]. Visualisation of molecular simulations is very important but for system sizes greater than 1 million atoms, most visualisation is currently carried out only after the simulation has completed.

Ideally, we would like to be able to carry out real-time visualisation in order to see the evolution of a system while it is running. In addition to this, computational steering provides a way to interact with and monitor these simulations. Real-time visualisation and computational steering are activities which cannot be carried out on batch systems, still used by most high performance computing facilities. This means that advanced reservation and co-scheduling of the resources are needed, so that the user can schedule an interactive session that may involve multiple resources.

From a usability point of view, cross-site and interactively steered simulations depend critically on the availability of automated mechanisms for the advanced reservation and co-scheduling of compute resources and networks. Currently, there is a lack of grid middleware tools that allow scientists to routinely schedule and launch cross-site simulations. Concerted efforts in this area, for example combining the AHE and HARC in conjunction with policies and facilities provided by grid resource managers, are required to allow scientists to exploit advances in various areas of computing in a coherent fashion. The principal purpose is to allow us to do science that was not possible before.

6. Future work

The computational scientist would benefit from the closer integration of the Application Hosting Environment, the HARC co-allocator and the RealityGrid steering system [8]. In our future work, we plan to integrate the AHE with the ReG steering system, so that steered applications and their associated steering web service can be launched as trivially as more conventional applications in the AHE. We plan to extend the AHE client interface so that it can be used to make cross-site reservations via HARC and then launch applications in those reservations. AHE will also be extended to support the launching of cross site MPI-g based applications, providing the end user with a single client interface in which they can make reservations for compute and network resources and also easily launch the applications that will exploit such coallocations. We also plan to interface HARC with the ESLEA Control Plane Reservation software [74], to allow seamless co-reservations of compute and network resources.

This middleware stack will be utilised by the GENIUS[4] (Grid Enabled Neurosurgical Imaging Using Simulation) project, on which we are now embarking, which is concerned with performing brain blood flow simulations in support of clinical neurosurgery using a bespoke lattice-Boltzmann simulation code, HemeLB, distributed across a combination of UK and US grid resources, in combination with medical imaging data.

Acknowledgements

This work has been supported by EPSRC grant number GR/T04465/01 (ESLEA), by EPSRC Grant EP/D500028 (SPICE) and the EPSRC funded RealityGrid project (GR/R67699 and EP/C536452/1); these grants have also supported our work with computational time on the UK's CSAR and HPCx facilities. Our work has also been supported by several NFS/TeraGrid grants including NRAC MCA04N014, PACS ASC030006P, MRAC DMR070013N and DMR070014N and DAC ASC070019T and DMR070003P. We thank the DEISA Consortium (co-funded by the EU, FP6 projects 508830/031513), for support within the DEISA Extreme Computing Initiative.

We would like to thank Prof. Nick Karonis and Dr. Brian Toonen for their assistance and support for MPICH-G2/MPI-g.

The implementation of HARC was supported in part by the National Science Foundation "EnLIGHTened Computing" project [41], NSF Award #0509465.

[4] http://wiki.realitygrid.org/wiki/GENIUS.

References

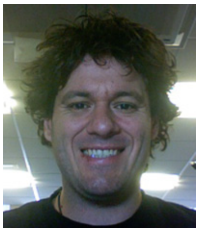
[1] P.V. Coveney (Ed.), Scientific grid computing, Philosophical Transactions of the Royal Society A 363 (2005).
[2] I. Foster, C. Kesselman, S. Tuecke, The anatomy of the grid: Enabling scalable virtual organizations, International Journal of Supercomputer Applications 15 (2001) 3–23.
[3] Uklight network. http://www.uklight.ac.uk.
[4] A. Hirano, L. Renambot, B. Jeong, J. Leigh, A. Verlo, V. Vishwanath, R. Singh, J. Aguilera, A. Johnson, T.A. DeFanti, The first functional demonstration of optical virtual concatenation as a technique for achieving terabit networking, Future Generation Computer Systems 22 (2006) 876–883.
[5] J. Mambretti, R. Gold, F. Yeh, J. Chen, AMROEBA: Computational astrophysics modeling enabled by dynamic lambda switching, Future Generation Computer Systems 22 (2006) 949–954.
[6] R.L. Grossman, Y. Gu, D. Hanley, M. Sabala, J. Mambretti, A. Szalay, A. Thakar, K. Kumazoe, O. Yuji, M. Lee, Data mining middleware for wide-area high-performance networks, Future Generation Computer Systems 22 (2006) 940–948.
[7] S.M. Pickles, J.M. Brooke, F.C. Costen, E. Gabriel, M. Müller, M. Resch, S.M. Ord, Metacomputing across intercontinental networks, Future Generation Computer Systems 17 (8) (2001) 911–918.
[8] S. Pickles, R. Haines, R. Pinning, A. Porter, A practical toolkit for computational steering, Philosophical Transactions of the Royal Society A 363 (1833) (2005) 1843–1853.
[9] N. Karonis, B. Toonen, I. Foster, MPICH-G2: A grid-enabled implementation of the message passing interface, Journal of Parallel and Distributed Computing 63 (5) (2003) 551–563.
[10] P.V. Coveney, R.S. Saksena, S.J. Zasada, M. McKeown, S. Pickles, The application hosting environment: Lightweight middleware for grid-based computational science, Computer Physics Communications 176 (6) (2007) 406–418.
[11] S. Graham, A. Karmarkar, J. Mischkinsky, I. Robinson, I. Sedukin, Web Services Resource Framework, Tech. Rep., OASIS Technical Report. 2006. http://docs.oasis-open.org/wsrf/wsrf-ws_resource-1.2-spec-os.pdf.
[12] Job Submission Description Language Specification, OGF. http://forge.gridforum.org/projects/jsdl-wg/document/draft-ggf-jsdl-spec/en/21.
[13] http://gridsam.sourceforge.net.
[14] WSRF Lite. http://www.sve.man.ac.uk/research/AtoZ/ILCT.
[15] S.J. Zasada, R. Saksena, P.V. Coveney, M. Mc Keown, S. Pickles, Facilitating user access to the grid: A lightweight application hosting environment for grid enabled computational science, in: Second IEEE International Conference on e-Science and Grid Computing, 2006, pp. 50–58.
[16] UK National Grid Service. http://www.ngs.ac.uk/.
[17] DEISA Grid. http://www.deisa.org/.
[18] S.J. Zasada, B.G. Cheney, R.S. Saksena, J.L. Suter, P.V. Coveney, J.W. Essex, Production level scientific simulation management on international federated grids, in: Proceedings of the TeraGrid 07 Conference, 2007.
[19] I. Foster, Globus toolkit version 4: Software for service-oriented systems, in: Network and Parallel Computing, 2005, pp. 2–13.
[20] The UNICORE Project. http://www.unicore.org.
[21] The Open Grid Forum. http://www.ogf.org.
[22] I. Foster, H. Kishimoto, A. Savva, D. Berry, A. Djaoui, A. Grimshaw, B. Horn, F. Maciel, F. Siebenlist, R. Subramaniam, et al. The open grid services architecture, Global Grid Forum Draft, draft-ggf-ogsa-ogsa-011, September 23.
[23] Teragrid Science Gateways Program. http://www.teragrid.org/programs/sci_gateways/.
[24] Java Specification Request 168, JSR-168 (Portlet Specification). http://jcp.org/en/jsr/detail?id=168.
[25] Job Management Enterprise Application, JMEA. http://sourceforge.net/projects/jmea,.
[26] T. Soddemann, Job management enterprise application, JMEA, in: Proceedings of the Euro-Par 2006 Workshop, 2007, pp. 253–262.
[27] C.R. Mechoso, J.D. Farrara, J.A. Spahr, Running a climate model in a heterogeneous, distributed computer environment, in: Proceedings of the Third IEEE International Symposium on High Performance Distributed Computing, 1994, pp. 79–84.
[28] T. DeFanti, I. Foster, M. Papka, R. Stevens, T. Kuhfuss, Overview of the I-WAY: Wide area visual supercomputing, International Journal of Supercomputer Applications 10 (1996) 123–130.
[29] T. Soddemann, Science gateways to DEISA: User requirements, technologies, and the material sciences and plasma physics gateway: Research articles, Concurrency and Computation: Practice & Experience 19 (6) (2007) 839–850.
[30] Apache Axis. http://ws.apache.org/axis.
[31] Apache Hivemind. http://hivemind.apache.org/.
[32] CondorG. http://www.cs.wisc.edu/condor/condorg/.

[33] GridFTP. http://www.globus.org/toolkit/docs/4.0/data/gridftp/.
[34] D. Snelling, S. Van Den Berghe, V. Qian Li, Explicit Trust Delegation: Security for dynamic Grids, Fujitsu Scientific and Technical Journal 40 (2) (2004) 282–294.
[35] C. Bouras, C. Chantzi, V. Kapoulas, A. Panagopoulos, I. Sampraku, A. Sevasti, Performance issues of bandwidth management in ATM networks, International Journal of Communication Systems 16 (2003) 151–169.
[36] HARC The Highly-Available Resource Co-allocator. http://www.cct.lsu.edu/~maclaren/HARC.
[37] J. MacLaren, M.M. Keown, S. Pickles, Co-allocation, fault tolerance and grid computing, in: Proceedings of the UK e-Science All Hands Meeting 2006, pp. 155–162.
[38] J. MacLaren, Co-allocation of compute and network resources using HARC, in: Proceedings of Lighting the Blue Touchpaper for UK e-Science: Closing Conference of the ESLEA Project, PoS(ESLEA)016, 2007. http://pos.sissa.it/archive/conferences/041/016/ESLEA_016.pdf.
[39] J. Gray, L. Lamport, Consensus on transaction commit, ACM TODS 31 (1) (2006) 130–160.
[40] A. Hutanu, G. Allen, S.D. Beck, P. Holub, H. Kaiser, A. Kulshrestha, M. Liška, J. MacLaren, L. Matyska, R. Paruchuri, S. Prohaska, E. Seidel, B. Ullmer, S. Venkataraman, Distributed and collaborative visualization of large data sets using high-speed networks, Future Generation Computer Systems, The International Journal of Grid Computing: Theory, Methods and Applications 22 (8) (2006) 1004–1010.
[41] EnLIGHTened Computing Highly-dynamic Applications Driving Adaptive Grid Resources. http://www.enlightenedcomputing.org.
[42] NW-Grid: the NorthWest Grid. http://www.nw-grid.ac.uk/.
[43] TeraGrid. http://www.teragrid.org/.
[44] A.C. Davenhall, P.E.L. Clarke, N. Pezzi, L. Liang, The ESLEA circuit reservation software, in: Proceedings of Lighting the Blue Touchpaper for UK e-Science: closing conference of ESLEA Project', PoS(ESLEA)015, 2007. http://pos.sissa.it/archive/conferences/041/015/ESLEA_015.pdf.
[45] http://www.hpcx.ac.uk.
[46] Iperf. http://dast.nlanr.net/Projects/Iperf.
[47] High performance enabled SSH/SCP. http://www.psc.edu/networking/projects/hpn-ssh.
[48] D.K. Lubensky, D.R. Nelson, Physical Review E 31917 (65) (1999);
R. Metzler, J. Klafter, Biophysical Journal 2776 (85) (2003);
S. Howorka, H. Bayley, Biophysical Journal 3202 (83) (2002).
[49] A. Meller, et al., Physics Review Letters 3435 (86) (2003);
A.F. Sauer-Budge, et al., Physics Review Letters 90 (23) (2003) 238101.
[50] M. Karplus, J.A. McCammon, Molecular dynamics simulations of biomolecules, Nature Structural Biology 9 (9) (2002) 646–652.
[51] S. Jha, P.V. Coveney, M.J. Harvey, R. Pinning, SPICE: Simulated Pore Interactive Computing Environment, in: Proceedings of the 2005 ACM/IEEE conference on Supercomputing.
[52] B. Boghosian, P.V. Coveney, S. Dong, L. Finn, S. Jha, G. Karniadakis, N. Karonis, Nektar, SPICE and Vortonics — Using federated grids for large scale scientific applications, in: Proceedings of Challenges of Large Applications in Distributed Environments (CLADE) 2006, Paris, vol. IEEE Catalog Number: 06EX13197, 2006, pp. 32–42, ISBN 1-4244-0420-7.
[53] M.J. Harvey, S. Jha, M.A. Thyveetil, P.V. Coveney, Using lambda networks to enhance performance of interactive large simulations, in: 2nd IEEE International Conference on e-Science and Grid Computing, Amsterdam, 4–6 December 2006.
[54] J.C. Philips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R.D. Skeel, L. Kale, K. Schilten, Scalable molecular dynamics with NAMD, Journal of Computational Chemistry 26 (2005) 1781–1802.
[55] W. Humphrey, A. Dalke, K. Schulten, VMD—Visual Molecular Dynamics, Journal of Molecular Graphics 14 (1996) 33–38.
[56] M. Carson, D. Santay, NISTNet—A linux-based network emulation tool, Computer Communication Review 6 (2003) 111–126.
[57] J. Chin, P.V. Coveney, Chirality and domain growth in the gyroid mesophase, Proceedings of the Royal Society of London Series A 462 (2006) 3575–3600.
[58] G. Giupponi, J. Harting, P.V. Coveney, Emergence of rheological properties in lattice Boltzmann simulations of gyroid mesophases, Europhysics Letters 73 (2006) 533–539.
[59] J. Harting, J. Chin, M. Venturoli, P.V. Coveney, Large-scale lattice Boltzmann simulations of complex fluids: Advances through the advent of computational grids, Philosophical Transactions of the Royal Society A. 1833 (2005) 1895–1915.
[60] W. Gropp, E. Lusk, A. Skjellum, Using MPI, MIT Press, 1994, pp. 59–97.
[61] M. Venturoli, M.J. Harvey, G. Giupponi, P.V. Coveney, R.L. Pinning, A.R. Porter, S.M. Pickles, Robust grid-based environment for large scale lattice-Boltzmann simulations, in: Proceedings of the UK e-Science All Hands Meeting, 2005. http://www.allhands.org.uk/2005/proceedings/papers/465.pdf.
[62] B. Boghosian, L.I. Finn, P.V. Coveney, Moving the data to the computation: Multi-site distributed parallel computation. http://www.realitygrid.org/publications/GD3.pdf.
[63] Starlight network. http://www.startap.net/starlight/.
[64] HECToR. http://www.hector.ac.uk.
[65] Vortonics. http://hilbert.math.tufts.edu/~bruceb/VORTONICS/index.html.
[66] P. Boulet, P.V. Coveney, S. Stackhouse, Simulation of hydrated Li^+, Na^+ and K^+ montmorillonite/polymer nanocomposites using large-scale molecular dynamics, Chemical Physics Letters 389 (4–6) (2004) 261–267.
[67] H.C. Greenwell, W. Jones, P.V. Coveney, S. Stackhouse, On the application of computer simulation techniques to anionic and cationic clays: A materials chemistry perspective, Journal of Materials Chemistry 16 (8) (2006) 706–723.
[68] J. Suter, P. Coveney, H. Greenwell, M.-A. Thyveetil, Large-scale molecular dynamics study of montmorillonite clay: Emergence of undulatory fluctuations and determination of material properties, Journal of Physical Chemistry C 111 (23) (2007) 8248–8259.
[69] H. Greenwell, A. Bowden, B. Chen, P. Boulet, J. Evans, P. Coveney, A. Whiting, Intercalation and in situ polymerization of poly (alkylene oxide) derivatives within M-montmorillonite (M = Li, Na, K), Journal of Materials Chemistry 16 (2006) 1082–1094.
[70] E. Lindahl, O. Edholm, Mesoscopic undulations and thickness fluctuations in lipid bilayers from molecular dynamics simulations, Biophysical Journal 79 (1) (2000) 426–433.
[71] R. Chang, G. Ayton, G. Voth, Multiscale coupling of mesoscopic-and atomistic-level lipid bilayer simulations, The Journal of Chemical Physics 122 (2005) 244716.
[72] A. Farrel, I. Bryskin, GMPLS: Architecture and Applications, Morgan Kaufmann, Amsterdam, 2006.
[73] K.Y. Sanbonmatsu, S. Joseph, C.-S. Tung, Simulating movement of tRNA into the ribosome during decoding, PNAS 102 (44) (2005) 15854–15859.
[74] A.C. Davenhall, P.E.L. Clarke, N. Pezzi, L. Liang, The ESLEA circuit reservation software, in: Proceedings of Lighting the Blue Touchpaper for UK e-Science: Closing Conference of ESLEA Project, PoS(ESLEA)015, 2007.

P.V. Coveney holds a Chair in Physical Chemistry and is Director of the Centre for Computational Science (CCS) at UCL. He holds an Honorary Professorship in Computer Science, also at UCL. Coveney leads the large EPSRC RealityGrid Project, funded from 2001–2005 as a Pilot Project, and from 2005–2009 under a Platform Grant. Coveney has pioneered the application of scientific grid computing including the use of computational steering to harness distributed grid infrastructure in order to solve challenging scientific problems in the physical, life and biomedical sciences. Coveney is currently chair of the UK Collaborative Computational Projects Steering Panel.

G. Giupponi is currently a post-doc researcher at the Computational Biochemistry and Biophysics Lab (CBBL), University Pompeu Fabra in Barcelona. He was awarded a Ph.D. in Physics by the University of Leeds in 2004. He then joined the Centre of Computational Science (CCS), University College London until May 2007. His research interests are macromolecular modelling and dynamics, fluid dynamics, systems biology and high performance computing using multiprocessor machines and Cell hardware.

S. Jha is an Assistant Research Professor in the CS Department at Louisiana State University and a Senior Research Scientist at the CCT at LSU. He also leads the Simple API for Grid Applications (SAGA) efforts at the Open Grid Forum and amongst other things is working on developing e-Science applications using SAGA.



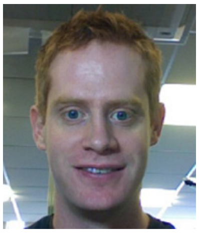
S. Manos is a postdoctoral research fellow at the Centre for Computational Science, University College London. He recently completed his Ph.D. at the University of Sydney, Australia, which explored the design of novel microstructured optical fibres using genetic algorithms and genetic representations capable of evolving designs with variable complexity. His current research activities focus on the use of data mining, visualisation and genetic algorithms in the design of new ceramic materials, and blood flow simulation and visualisation as a clinical tool for neurosurgeons. His general research interests include automated computational design, and the use of grid computing from an applied, real world perspective.

J. MacLaren is a Researcher in Middleware for Distributed Computing at the Center for Computation & Technology at Louisiana State University. He has been working in Grid and Distributed Computing since 2001, on a variety of projects, but usually on scheduling-related issues. While at CCT, he has developed the HARC Co-allocation software. Previously, he worked at the University of Manchester, from where he also received his M.Phil. and Ph.D.; he also has a B.Sc. in Maths/Computer Science from the University of York.

S.M. Pickles worked in commercial computing with ICL (Australia) before turning his hand to computational science. In 1998, he gained his Ph.D. in theoretical physics from the University of Edinburgh, then moved to the University of Manchester at the start of the CSAR supercomputing service. He has been engaged in Grid computing since 1999. His activities include building production Grids and applications to exploit them, and the development and standardisation of middleware. He is currently Technical Director of the UK National Grid Service, and Area Director for the Compute Area of the Open Grid Forum.

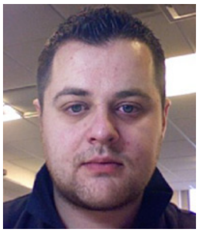
R.S. Saksena is an EPSRC post-doctoral Research Fellow at the Centre for Computational Science, University College London. She received her M.Phil. and Ph.D. from the University of Manchester and has a B.Sc. in Computational Science/Chemistry from the National University of Singapore. She is currently studying self-assembly phenomena and rheology of complex fluids using the lattice-Boltzmann method and she has previously been a developer on the AHE project. Her research interests include soft matter physics, molecular dynamics, Monte Carlo and lattice-Boltzmann simulation methods, parallel and distributed computing.

T. Soddemann is working as a Software Architect for the Computing Center of the Max Planck Society. He is responsible for the integration of high performance applications in materials science into Grid environments especially within the DEISA compute infrastructure. While working for the Max Planck Institute for Polymer Research, he received his Ph.D. from the Johannes Gutenberg University, Mainz, Germany, and was a post-doctoral research fellow at the Johns Hopkins University, Baltimore, MD.



J.L. Suter received his M.Chem. from the University of Oxford and Ph.D. from University of Cambridge under the supervision of Professor Michiel Sprik in the Department of Chemistry. Research undertaken at Cambridge included using first principle methods to study clay minerals properties in collaboration with Schlumberger Cambridge Research. Currently based at the Centre for Computational Science, University College London, his work addresses multi-scale materials modelling and analysis of clay-polymer nanocomposites using high-perfomance and Grid computing methods.

M. Thyveetil is a final year Ph.D. student, under the supervision of Professor Peter V. Coveney at the Centre for Computational Science, University College London. Her Ph.D. research focuses on the study of layered double hydroxides using large-scale molecular dynamics techniques and Grid computing to obtain materials properties of the system, as well as to model its interaction with large biomolecules.



S.J. Zasada has a first degree in Computer Science from the University of Nottingham and a Masters degree in Advanced Software Engineering from the University of Manchester. He was responsible for implementing the WS-Security specification in Perl for use by the WSRF::Lite toolkit, and is currently lead developer on the AHE project. In addition, he is working on a Ph.D. in Computer Science, investigating the design and development of lightweight grid middleware solutions.