Mini-Grids: Effective test-beds for GRID application

John Brooke, Martyn Foster, Stephen Pickles, Keith Taylor, Terry Hewitt[1]

MRCCS, University of Manchester, Oxford Road, Manchester M13 9PL,
`j.m.brooke@man.ac.uk`,
`http://www.csar.cfs.ac.uk/staff/brooke`

Abstract. We describe a computing environment that we call a "mini-GRID". This represents a hetereogeneous group of resources for computation, data storage, archival and visualization which can be connected via private or public networks to other resources (called "guest systems") on a temporary basis as required. The mini-GRID displays the heterogeneity and some of the complexity of a full computational GRID, but in a more limited environment and can be considered to be under the control of a few organisations (or even a single organisation) making non-technical organisational issues less problematic. As such, the mini-GRID provides a flexible and controllable, but realistic test-bed for trialling GRID applications, particularly with regard to issues such as accounting and resource brokering. However, its heterogeneity, the size and complexity of the architectures involved, and its integral connection with local, national and super-national networks, prevent it from being considered as a cluster of workstations.

1 What is a mini-GRID?

We describe in this paper an environment that we call a "mini-GRID". This is a collection of computational and data processing resources that has a heterogeneous structure (multi-architecture, multiple levels of data storage) but is organisationally simple, e.g. under the control of a single organisation. We consider these mini-GRIDs to be of research interest in two ways. Firstly, a likely structure for the development of the world-wide GRID is the gradual connection and integration of such local GRIDs, each serving both its own hinterland and also acting as a node for wide-area applications. Secondly, because they are of sufficient complexity to test out key middle-ware components of the GRID, e.g. coupled applications, resource brokers, accounting and billing systems, without needing to address questions of local or national autonomy. We follow the line of thought developed in Chapter 1 of [1] that the development of the GRID involves the simultaneous development of local and remote links, the development of the railway infrastructure around Chicago being an example.

The particular mini-GRID that we describe is based on services provided at the University of Manchester for both local and UK-wide use. The UK-wide service is called CSAR (Computing Services for Academic Research) and is provided by a consortium CfS (Computation for Science) which is a collaboration between CSC, the University of Manchester and SGI. From its very conception this service was envisaged as being multi-component and not necessarily restricted to a particular vendors hardware. It is a dynamic configuration, that can be added to by the provision of guest machines based either at Manchester or at other CfS sites (e.g. CSC Supercomputer Centre at Farnborough). However, because it is intended to be viewed as a single resource in terms of funding, an internal economy has needed to be developed that allows for the conversion and trading of resources between the various components of the service. We describe this economy further in Section 3. This economy has been operating since late 1998 and provides valuable lessons for other GRID projects since some sort of GRID economy will have to emerge to permit the sharing and inter-convertibility of resources. We show the current CSAR configuration in Figure 1, for more information about CSAR and CfS see `http://www.csar.cfs.ac.uk`

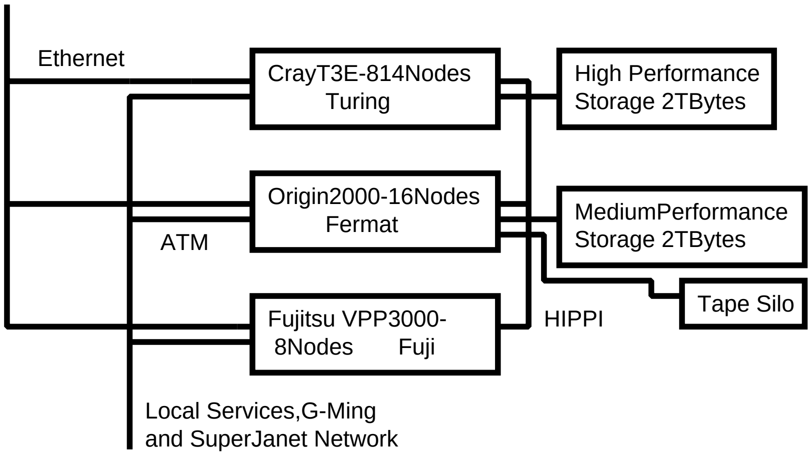

Fig. 1. The current CSAR configuration showing the multi-architectural structure. Guest systems can be added to this as required. There are three levels of data storage, high-performance attached to the T3E, medium peformance attached to the O2000 and tape-archiving via the tape silo.

Our mini-GRID also includes services provided over a local high-speed network covering the North-West of England. This originally started as a 155-Mbit/s ATM network covering the Greater Manchester area (G-MING in Figure 1). This has been extended to other sites in the North-West of England, primarily educational institutions. These networks allow local colleges, hospitals, local government offices and local businesses to access centralised facilities creating a local GRID. Since this GRID connects through the University of Manchester which is becoming a node on continental and trans-continental GRIDs, we have a route by which local access can grow to world-wide access. We briefly describe in Section 5 a project which has used the local GRID in both of these ways and we show how this could be integrated into the economy developed under the CSAR service, to allow the GRID node to recover costs from local users and give an incentive to expand its service for local access.

The plan of this research note is to present the results of ongoing experiments designed to explore the exploitation of the mini-GRID structure described above. In Section 2 we describe work on running coupled simulations across the differing components of the CSAR configuration. Section 3 describes the development of the internal mini-GRID resource economy. Section 4 describes experiments designed to test the usefulness of the Globus toolkit in this context. Section 5 describes the links over metropolitan and regional networks and how this is being extended to provide global links. Finally we draw some preliminary conclusions in Section 6 and describe some new projects which will extend the functionality of the mini-GRID.

2 Running coupled applications in a mini-GRID

We describe work to utilise the core mini-GRID structure of the CSAR service (Figure 1) for running coupled applications. Since we wish to run between machines, current vendor-supplied versions of MPI cannot be used and we investigated both PVM and special MPI libraries for running across machines.

2.1 PVM - Its Continuing Role As A Grid Builder

PVM, Parallel Virtual Machine [2] first made its public appearance as Version 2, back in March 1991. Since then it has expanded and prospered, becoming available on an ever widening range of machine architectures; until, now, we have reached Version 3.4.

Being familiar, the virtues of PVM and its usefulness for GRID work is sometimes ignored. PVM lacks some of the 'sophistication' of MPI, with the latter's Derived Datatypes, and support for Cartesian mesh-based communications, for example. However, PVM's big strength remains: the fact that is was designed from the outset for connecting together collections of physically-separated heterogeneous machines, something which MPI is only beginning to address with initiatives like PACX-MPI [3] and STAMPI [4]. PVM's range of facilities may be limited to the essentials, but these provide all that is required to send data back and forth, between different computer systems, including well-defined procedures for installing the software, starting it up, and building and operating a desired configuration.

Prompted by the OCCAM [5] group's need to couple a shared-memory atmosphere model running on an SGI Origin2000 with an ocean model running on a Cray T3E-1200E, (Fermat and Turing respectively in Figure 1), we developed a demonstrator which proved the PVM concept for this application [6]. It was subsequently successfully developed to harness the two models for real production runs. Some lessons were learnt along the way.

Firstly, in order to utilise the inter-machine functionality of PVM, it was sometimes necessary to undo the carefully hand-crafted 'optimizations' which manufacturers have included (for performance reasons) in their machine specific implementations of PVM. For example, in the T3E version, only the first task, by default, communicates via the daemon, for the others `send` and `receive` are implemented on top of SHMEM. Source and target tasks are identified by non-portable processor numbers, rather than the standard task identifiers which would otherwise be allocated by the daemon. This non-standard behaviour prevents any task other than the first from seeing the outside world, completely opposing one of PVM's philosophical foundations. But, fortunately, if required, one can restore full visibility and task identifiers by setting an environment variable, `PVM_PE_LIST`, to `all`.

Secondly, bits may be missing from a particular implementation. For example, it was necessary to install the public-domain version of PVM over SGI's for the Origin2000, before we received a sensible error message which indicated the true cause of a problem we were having. It turned out that the encoding `PVMDATAINPLACE` was not implemented for `PVMFINITSEND` on SGI machines. This was absent from the documentation. Subsequent to our notifying the vendor, a documentation bug report was issued, but the omission remains.

Thirdly, PVM, like any other grid-creating system, relies heavily on the integrity of the underlying network. Our attempts to extend the demonstrator to include CSAR's 8-processor Fujitsu VPP300 (Fuji in Figure 1) were unsuccessful, for reasons still unknown. Briefly, communications between 'Fermat' and 'Fuji' were unreliable, in a non-repeatable way. They may have been affected by the level of other traffic on the ATM line connecting the two machines, and thus various internal timeouts came into play, causing 'hang-ups'.

2.2 Implications for GRID computing

PVM already has a rudimentary way of measuring and comparing the performance of participating machines for the purpose of load-balancing. (This is made evident when one examines the list of component architectures from the 'console'.) This is a topic of considerable interest in running distributed applications over the GRID. It may be that the existing and very much alive PVM could be enhanced to encompass a more suitably refined description. The work described here is available at [6]. For a description of using PACX-MPI across a global metacomputer see [7]. Our experiments here highlight the need for reliable message passing libraries that can run across machines of different architectures and which adhere to internationally agreed standards.

3 Accounting for resource usage on a mini-GRID

The University of Manchester has developed a web-based user registration system, which has been used for some years now in the administration of both local and national computing services. Features that this system provide include project management, resource allocation, and user self-registration. In 1998, the advent of the CSAR service, a privately financed initiative, saw extensive enhancements to the registration system, especially in the areas of resource management and accounting. The new facilities were developed in response to:

1. demand from the UK academic HPC community for the ability to use allocated resources in a more flexible manner, and
2. the need to account to the funding bodies.

We claim that the problems of resource management and accounting will be of increasing importance to HPC service providers, and that satisfactory solutions to these problems in the context of grid computing do not yet exist. We therefore describe our solutions in the simpler context of a mini-GRID, where the issues of cross-institutional co-operation and site autonomy do not arise. It is our thesis that many aspects of our solutions must be reflected in solutions to the general, pan-institutional problem.

When a research council approves a peer-reviewed application for computing resources on the CSAR service, a new project is set up in the registration system.

The right of a project to utilise computing resources of various types (eg. CPU time, disk and tape storage on specific machines) is represented by quantities of *specific resource tokens* of corresponding types. Tokens are valid for the lifetime of the project.

In addition to specific resource tokens, we also introduce generic service tokens. Generic service tokens have a notional cash equivalent, indicative of the expected cost of the project to the research councils.[1] In agreement with the research councils, exchange rates between generic and specific resource tokens are fixed from time to time, to reflect depreciation of the underlying asset. Table 1 shows the exchange rates currently in force.

Resource Token	Value in Generic Tokens
1 T3E PE Hour	0.024
1 Gbyte-Year of T3E Disk	6.752
1 Origin 2000 CPU Hour	0.025
1 Gbyte-Year of O2000 Disk	4.292
1 Gbyte-Year of HSM/Tape	0.596
1 Fujitsu VPP CPU Hour	0.345
1 Gbyte-Year of Fujitsu Disk	4.292
1 Day training	3.000
1 Day support	11.364

Table 1. Resource exchange rates in the CSAR service effective as of June 2000.

The computing resources required by a project are listed on a schedule accompanying the original grant application. The schedule also lists the amount

---

[1] The research councils are billed periodically according to actual usage.

of each resource expected to be consumed in each six-month period of the life of the project. The CSAR service provides a web-based *resource calculator* `http://www.csar.cfs.ac.uk/admin/forms/calculator.shtml` to assist the principle investigator in preparing this schedule. New projects are primed with sufficient generic service tokens to meet the expected requirements. Before a project can begin to use the service, the generic tokens must be traded for specific resource tokens. The initial trade is often performed by CSAR staff, but may be performed by the principle investigator if desired. A project may subsequently trade unused resource tokens of one type for resource tokens of another type, subject to availability of the desired resource. The flexibility that this provides is one of the factors that differentiates CSAR from its predecessors in the UK.

We impose a minimum period between successive trades by any one project in order to avoid abuses such as exploitation of arbitrage opportunities arising out of possible rounding errors, or attempts to corner the market in a limited resource.

The number of specific resource tokens of any type in the trading pool at any one time is limited so as to reflect the (projected) capacity that the service can provide.

A project's cumulative usage of each resource is updated daily in batch mode. CPU time and tape storage are charged on the basis of actual consumption. Permanent disk storage is charged according to integrated disk quota, sampled daily; note however that the principle investigator is empowered to alter disk quotas freely between a lower limit of actual usage and an upper limit set by CSAR. Principal investigators are notified automatically when the project's usage of a resource first exceeds 90%, 95% and 100% of the total available; the third notification is accompanied by a withdrawal of access privileges.

It is in the interest of any HPC provider to optimize the capacity of the service to meet projected demand. Under-utilized resources are obviously undesirable. On the other hand, having insufficient resources makes it necessary to turn work away; but if the increased demand can be anticipated, it is possible to finance the purchase of additional hardware. We facilitate this by introducing a *capacity plan* for each new project, capturing the projected usage from the schedule on the original grant application, and encouraging principle investigators to review these capacity plans as their projects develop.

We believe that the regulated micro-economy embodied in our registration, trading pool, capacity planning and accountancy systems has features that most providers of pay-as-you-go supercomputing services should find desirable. We encourage grid developers to take cognizance of the considerations that have informed our approach, as these are likely to be shared by the supercomputing centres that will become the major grid nodes of the future. Although we are not advocating the introduction of resource tokens on a global scale, we think that the resource brokers of a truly ubiquitous computational grid must embrace some kind of currency to mediate negotiations, and that grid nodes must not only be able to advertise availability of resources but also to contract to a pricing policy when accepting an offered job.

4 Globus on the mini-GRID

We describe work in progress to evaluate how Globus [8] can be adapted to serve our mini-GRID environment. We deployed Globus (1.1.2) on a variety of workstation and server machines including the Cray T3E, Origin 2000, Solaris and Linux workstations. We found no major obstacle with the installation of the software on these platforms though some scripts needed tuning to individual system requirements. All batch queue systems used were NQE/NQS based. Some questions were raised regarding the overhead of running Globus on HPC machines, these are still being studied.

Our aim was to develop tools which allowed a uniform job submission interface to the different machines on the mini-GRID (see Figure 1). The requirement to make this user-friendly meant that the Globus command-line interface had to be adapted. We were able to develop software which was attractive and simpler to use than the default tools supplied by vendors with the batch system. With further effort these tools could be expanded to provide a much more flexible approach to high performance computing.

An important aspect of the Globus environment is the seamless transition between scales of computing resource. Globus comes into its own when operational practice migrates the user from the workstation through super-computing centre to specialized resource, without changing working practice significantly. However without developing a uniform job submission interface, the divide between HPC and desktop processing will remain in the near future. The job submission models supported by Globus are insufficiently sophisticated to cope with typical methods employed by the CSAR user community. There exists other software designed for job submission which overcomes some of these limitations, the UNICORE project being a key example [9]. A rapprochement of the Globus and Unicore approaches would seem to be very much in the interest of users of resources such as the mini-GRID described here.

Outside the administrative domain, Globus is found to provide the necessary infrastructure to enhance utilization of computing resources around the UK. In particular the local grid environment can be extended to encompass other major computing facilities in the UK. This activity is mainly centered upon coupling the T3E machines at Edinburgh (EPCC) and Manchester (CSAR), in order to balance the job load between the two machines. Initially the mechanism employed will be to prioritize queues on each machine for different classes of job and publishing estimates of execution time for various job specification via the MDS. In the longer term this may be replaced or complemented by a knowledge-based third party broker.

5 A Metropolitan area GRID

The computing service at the University of Manchester has an important role in the provision of networking and high-performance services over metropolitan and regional networks. These utilise ATM at 155 Mbits/s, allowing the creation of Permanent Virtual Circuits (PVCs) giving a guarenteed networking Quality of Service (QoS). We describe here results from two projects that show in different ways how issues arising from provision of services delivered locally via the mini_GRID has implications for the development of global GRID services.

In the RCNET [10] project, a collaboration between a local engineering consultancy (REL) and the University of Manchester allowed REL to move and coordinate its engineering work on a global scale. This involved connecting the company network to GMING and thus providing a route to European and global networks. This is important to REL because their mode of operation means that they typically establish working offices in areas where their services arise and these may migrate around the globe according to developments in, for example, the oil industry. These satellite offices may be very "light" in terms of equipment and so access to remote processing power is essential. Also the company's technical experts may be geographically distributed and video-conferencing and collaborative working are highly desirable.

5.1 Use of GMING

There were two ways in which access to the metropolititan-area network GMING changed the working practice of REL. Firstly, access to large servers at the University of Manchester allowed REL to run jobs which were too big for their workstations. These were run on two central machines, a Fujitsu VPX240 vector processor and an SGI Origin2000. A cost-benefit analysis was carried out to determine the savings to REL of using the local GRID in this way, rather than purchasing extra expensive workstations. This analysis is available at the RCNET WWW site [10].

Secondly the RCNET workstations were networked via an ATM switch and a link was made to a similar cluster of SG Indy workstations at UoM. The ATM technology allowed the two ATM switches linking these clusters to be connected and the two clusters were then both part of the same emulated LAN. This allowed parallel jobs to be run across the clusters and allowed the possibility of extra work from REL to be run on the UoM cluster, thus freeing REL workstations for intensive pre and post processing. LSF was used to manage the load on the UoM cluster, and account of the performance of the cluster under varying loads is given in [11].

A different use of GMING is also shown by the NOVICE project in which a large visualization server available over GMING allows hospitals throughout the Greater Manchester Conurbation to visualize medical scans from patients records. The visualization would be impossible on local equipment, since it is insufficiently powered to perform manipulation of very large datasets and to run specialist visualization software. An important feature of NOVICE is that images from the central server are sent to the remote stations using VRML. There are very delicate issues of security since private medical records are being delivered via the local GRID. These issues are all very relevant to the wider GRID, more details can be found via the NOVICE WWW site at [12].

5.2 Extension of the local GRID to intercontinental networks

In the RCNET project videoconferencing experiments were carried out between the Norwegian National Point of Presence in Oslo and the University of Manchester over the pan-European JAMES ATM network. REL(Manchester) offices are close to the University and thus a means for collaborative working was established. The main aim of the experiments was to test Quality of Service networking issues by comparing the running of videoconferencing and collaborative working over two routes, the normal Internet traffic and a dedicated ATM PVC (Permanent Virtual Circuit) which established a guaranteed bandwidth of 2 Mbits/s. The qualitative results were that videoconferencing and collaborative working were much better over the ATM PVC even though the bandwidth of the Internet route was potentially greater (34 Mbits/s). Quantitative experiments were performed to ascertain the reason for this difference.

File Size	ATM Mean	ATM SD	Internet Mean	Internet SD
1 Mbyte	198.0 kbytes/s	2.7 kbytes/s	314.6 kbytes/s	145.9 kbytes/s
5Mbyte	199.5 kbytes/s	5.5 kbytes/s	241.3 kbytes/s	69.2 kbytes/s
10Mbyte	203.3 kbytes/s	3.2 kbytes/s	251.9 kbytes/s	27.2 kbytes/s
50Mbyte	191.3 kbytes/s	0.9 kbytes/s	260.7 kbytes/s	26.1 kbytes/s

Table 2. Comparison of the mean time and standard deviation (SD) of files of various sizes. The files were transferred between Manchester and Norway between identical machines but via two routes, an ATM PVC of 2 Mbits/s and an internet shared traffic route at 34 Mbits/s.

A full account of the experiments is available at [10] but the most telling were the results of sending files of different sizes via ftp. We present these results in Table 2. It will be seen that the mean rate of transfer for both methods is comparable but the standard deviation is much lower for the ATM as would be expected. The size of file transmitted makes a considerable difference to the standard deviation over the internet route. Our conjecture is that this is because the file transmission time for sending a large file is much greater than the time scale on which the shared-traffic internet route bandwith varies. We draw the analogy with turbulent fluid flow; on length and time scales greater than those of the turbulence the flow can appear to be smooth. Our qualitative and quantitative results indicate that for applications such as videoconferencing and distributed collaborative working, the effects of this “network turbulence” over shared-traffic routes needs to be taken into account.

Another important factor is latency, which is very low in an ATM PVC. To test the effect of this factor we performed tests on the startup of the Netscape browser, both locally and remotely (Figure 2). This latter involves the sending of many messages of different sizes between the two sites, it thus illustrates in a simple way some of the performance issues involved in distributed collaborative working.

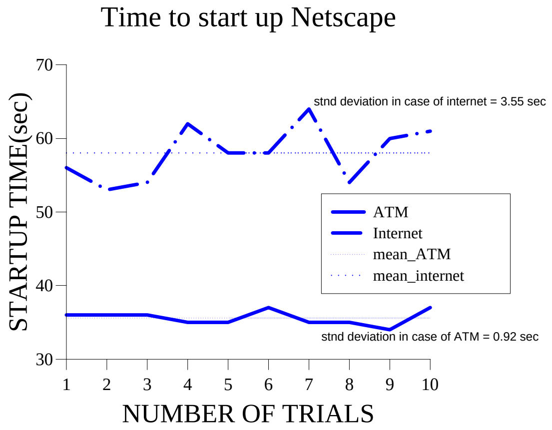

Fig. 2. Comparison of Netscape startup for ATM and Internet. The two routes are as described in Table 2

5.3 The modular structure of the GRID

The RCNET project was conceived before the concept of the GRID was drawn to the international community. However the whole networking and processing of RCNET is based on a modular structure, going from local to global and indicates clearly the considerations that we discuss in Sections 1 and 6. What needs to be supplied for this to be a working model, is a reliable layer of middleware that can direct jobs on the local GRIDs to centres of computational power, e.g. a resource broker. Also, there needs to be some means of “costing” this work and addressing QoS issues (for a commercial firm turnaround time may be crucial to meet their contracts). The issue of costing is taken up in Section 3. The question of a resource broker is a subject for future work.

6 Conclusion and future plans

As William of Occam pointed out in the 13th century, it is bad scientific practice to introduce a term unless it helps to reduce and clarify the scientific description of phenomena. An objection to the term “mini-GRID” as used here would be that there is scaling at all levels on the GRID, hence the term “mini-GRID” is meaningless. Our argument in this paper is that there is a natural structural scale for a mini-GRID, related to the wider GRID as a modular structure. Firstly the mini-GRID should relate to a particular, defined, networking complex. We suggest that a good candidate is a metropolitan area network or a node on an intercontinental GRID. Thus the mini-GRID will have an internal structure. We think that many academic supercomputing centres are likely to develop in this way. They can stabilise their funding by using their specialist expertise to attract income from local industry. In a less quantifiable fashion, by providing services to schools and civic organisations they can help to make the spending of public money more politically acceptable. An example of this sort of consideration, is the siting of large scale computing resources in regions where it is desirable to stimulate economic and social development. The Federal Swiss supercomputing centre CSCS is an example [13].

Other reasons why we predict that nodes on the computational GRID are likely to have a multi-component structure of the scale we describe here, involve the increased probability that new architectures are likely to be funded at sites that have proven expertise and a critical mass of expertise. Running a GRID node requires expertise in networking, hardware support, maintenance of infrastructure, operating systems, programming, numerical methods and particular scientific disciplines. We return to the analogy of the growth of Chicago as described in Chapter 1 of [1]. All industrialised societies develop cities on major infrastructure nodes and the development of a city involves a feedback loop. As population and infrastructure is attracted to a transportation node, more local infrastructure is needed to serve this, which attracts more population and infrastructure, in a positive feedback loop.

We believe that there is strong evidence that nodes on the GRID will need to be of a certain size and complexity to be self-sustaining. There will certainly be GRID structure below this size but it will tend to associate with a major GRID node. The GRID node can regulate an organizational regime that enables middleware, such as resource brokers, to access its resources on both sub-node and super-node scales. Between such nodes, there is a layer of political and organisational complexity and difficulty over and above the technical challenges involved.

The work we describe here on the scale of our mini-GRID is helping to clarify some of these issues, both technical and organisational. We seek to open a dialogue with other centres who recognize this mini-GRID concept as being useful. We are encouraged that the work described here is to be continued via a major European Union project, EUROGRID, which aims to connect major European sites each of which could be regarded as having the mini-GRID structure as described here. The setting up of a GRID structure across the European Union is of particular interest in the context of a global GRID, since the European Union is a federation of politically independent states with national funding councils for scientific research. There is also a strong emphasis in the EU Fifth Framework [14] on improving the quality of life for EU citizens. Thus the local aspects of the mini-GRID as described in Section 5 become very relevant.

Acknowledgements

The authors are grateful to all those who contributed to the design and development of the registration system and trading pool, especially Phil Stringer, Geoff Lane, Victoria Pennington and John Rawlins. Thanks to Fumie Costen for producing the figures.

References

1. I.Foster and C. Kesserlman, editors. The GRID: Blueprint for a new computing infrastructure. Morgan Kaufmann Publishers, Inc., San Francisco, California, 1998
2. For PVM details see URL http://www.epm.ornl.gov/pvm/pvm_home.html
3. Edgar Gabriel, Michael Resch, Thomas Beisel, Rainer Keller, 'Distributed Computing in a heterogenous computing environment' in Vassil Alexandrov, Jack Dongarra (Eds.) 'Recent Advances in Parallel Virtual Machine and Message Passing Interface', Lecture Notes in Computer Science, Springer, 1998, pp 180-188.
4. H. Koide, et al, MPI based communication library for a heterogeneous parallel computer cluster, Stampi, Japan Atomic Energy Research Institute, http://ssp.koma.jaeri.go.jp/en/stampi.html
5. The OCCAM group maintain highly-tuned models of global ocean circulation: see URL http://www.soc.soton.ac.uk/JRD/OCCAM/welcome.html
6. Running PVM Interactively Across Turing and Fermat, K.Taylor, CSAR Technical Report, 1999. Available at URL http://www.csar.cfs.ac.uk/staff/taylor
7. S. Pickles, F. Costen, J. Brooke, E. Gabriel, M. Müller, M. Resch, S. Ord, The problems and the solutions of the metacomputing experiment in SC99, in Marian Bubak, Hamideh Afsarmanesh, Roy Williams, Bob Hertzberger (Eds.) HPCN (Europe) 2000 Proceedings, Lecture Notes in Computer Science, Springer, 2000, pp 22-31
8. Details of the Globus project can be found at URL http://www.globus.org
9. Details of the Unicore project are at URL http://www.fz-juelich.de/unicore
10. J.M. Brooke and P. Jacob, editors RCNET: Exploiting HPCN in an Engineering Consultancy Environment Techical Reports and Papers.
    See URL: http://www.man.ac.uk/MVC/research/rcnet/
11. F. Costen, J.M. Brooke and M.A. Pettipher Investigation to make best use of LSF with high efficency In Proceedings of IEEE International Conference on Cluster Computing, Melbourne 1999, 211-221
12. M. Cooper, editor Network-Oriented Visualization in the Clinical Environment Espirit EU Project.
    see URL http://www.man.ac.uk/MVC/projects/NOVICE/
13. The C^3 Communications and Computing Camp is described at URL://www.cscs.ch (follow links for Education).
14. Details of the EU IST Call for Proposals Information Society Technolgies Programme at URL http://www.cordis.lu/ist/