Argonne National Laboratory Tech Report ANL/MCS-TM-278
Open Middleware Infrastructure Institute Technical Report OMII-CO-002
UKeS-2004-08 available from http://www.nesc.ac.uk/technical_papers/UKeS-2004-08.pdf

State of Grid Users:

25 Conversations with UK eScience Groups

Jennifer M. Schopf
Argonne National Laboratory

Steven J. Newhouse
Open Middleware Infrastructure Institute

October 25, 2004

1 Introduction

During July and August 2004 we visited various applied science and middleware groups in the U.K. in order to gather basic information on the services and functionality these projects are using. Our motivation was to help guide the development of future activities and priorities within the U.K.'s Open Middleware Infrastructure Institute [OMII] and the Globus Alliance [Globus] and to inform the wider Grid community of the status of some current services. We held meetings with application developers with some Grid (generally Globus Toolkit 2 or Globus Toolkit 3) or Web services experience, those with software that might be of broader use or interest, and those who have expressed dissatisfaction with current tools, in order to understand their issues in more detail. The twenty-five groups, listed in Appendix A, included representative applications from biology, chemistry, physics, climatology, and other scientific fields, as well as a smaller set of basic tool builders. In addition, informal discussions took place at several workshops during this time to get a broader scope in certain areas [secWS, SerWS, NFNN].

Meetings with groups varied from half-hour to half-day time slots and covered a wide variety of topics, concentrating on current use and needs. We considered performing structured interviews based on a standard questionnaire, but differences in backgrounds and knowledge of the interviewees made this approach ineffective, so the interviews were not formally structured as such. Instead, we asked what functionality the groups had tried in the past, what their applications needed today from the current Grid infrastructures, and what functionality the group was considering for near-future plans. Most meetings ended by our inquiring what functionality the group thought was most important and still lacking in today's tools or services.

Over the course of the interviews several basic ideas began to repeat themselves; and while we cannot claim to have interviewed all possible groups, the topics covered by the second half of our discussions only reinforced the initial data, thereby implying that we had established the significant issues for this particular community.

In this article we detail the results of our conversations with users. Our intention is to share this data with a broader audience. While the results are not unexpected, we note that the ranking of needs by the users was quite different from what many tool developers have assumed. We detail our findings in the areas of the continued need for training, security, service functionality as seen by the users, details on tools, and build/infrastructure comments. In Section 7 we highlight the most commonly stated concerns that emerged from these interviews.

2 Training and Education

For many of the groups we spoke to, the vision of the Grid—the use of distributed resources across different organizations—has not yet been fully accepted, let alone implemented. Almost all the applied science groups felt that the basic concepts of the Grid, and the maturity of software being developed to support these activities, had been oversold, and they were therefore now much more cautious in adopting new software infrastructures. Even in this Grid-friendly arena, we encountered many groups that didn't understand why they should consider Grid tools over their usual SSH and scp.

All the participants—users, developers, and system administrators—expressed a strong need for basic common practices within the Grid. Although Web services, firewalls, build instructions, and security were top areas of concern, there was an expressed need for better documentation of "common practices" across all these areas.

Many users felt that the Grid material was pitched at the wrong level. For example, for Web services, many of the middleware or application developers we spoke with were looking for a hands-on three-hour approach to understanding the basics—not a high level vision and not low-level tuning. The users we spoke with were more interested in API instruction—how to use the common tools and to understand the tradeoffs among these tools. Example applications that demonstrate the use of these APIs were seen as an essential element of any software infrastructure and much preferred to vague statements of capability.

Firewalls continue to be a headache for users and developers, and a cure-all for system administrators. There was general agreement that all stakeholders in network security (the firewall administrator, the local system administrators, and the users) need better instruction on the interactions between firewalls and the commonly deployed services and software. Documents such as the Globus Toolkit firewall requirements document [Welch03] and others [Hillier02] provide a solid base on which to build a common practice document describing good ways for system administrators and users to interact over firewalls. While Web services are seen as a way to "drill through" firewalls using commonly opened ports, the introduction of protocol-sensitive firewalls will eliminate this option and force greater communication among all the stakeholders involved in the network.

In general, the documentation provided by the middleware offerings was perceived to be lacking in detail and accuracy. Particular frustration was expressed about builds and packaging. This frustration highlights an important point. Users are likely to persevere in trying to use an infrastructure that they have been able to install. If they are not able to install the software and verify that it is working at some level, they are likely to give up and move on. However, details on specifically what was missing in the documentation were difficult or impossible to discover. The real solution may be to aid in better coordination between documentation writing and user communities.

3 Security

If security isn't easy to use, users will find a way to not use it[Surridge02]. Indeed, many groups had no security infrastructure associated with the tools or services in use. Common reasons included the configuration of services on local systems or behind firewalls, the lack of an agreed-upon security infrastructure among all sites, the concern about overhead or the effect security would have on performance, and the lack of control over remote service security. Most groups in this class agreed that this situation was not ideal but was acceptable for the time being, and none of them had near-term plans to alter the situation. Some of these groups that had adopted certificate-based authentication systems were using short or non-existent passphrases or were sharing certificates between individuals, effectively rendering these systems insecure.

For those groups implementing a security infrastructure, the greatest concern was with data integrity rather than with authorization or authentication on a system. Their primary worry was that data would be corrupted or lost, not that someone would use illicit cycles on their machines. Of course, in projects dealing with medical records, considerable restrictions do exist on who may view patient records, including medical images and written notes.

Another area of concern was the different approaches to delegation in Web services and in the Globus toolkit. Delegation within the GSI architecture involves the transfer of an IETF-compliant proxy certificate from a client to the server. The service on the server may then use this certificate to "impersonate" the user in order to access other services. An example is GridFTP, which can use this ability to access files owned by that user on another system.

Delegation is needed in a Grid services environment for third-party transfers, many portal interactions, and even some workflow scheduling approaches. This requires a level of trust by which the host providing the files is willing to provide access to the requesting service. In most industrial scenarios, however, delegation is not even contemplated because there is never a sufficient level of trust with a third party.

Web services are able to use specifications such as WS-Trust and WS-Policy to express a delegated authority but do not pass on a proxy certificate in the process. The service receiving the delegated authority is therefore unable to impersonate the remote user in the same manner as GSI-based services. Intermediary services, such as workflow engines, access the required services through their own identity, as opposed to that of the entity that originally invoked the service. Hence, there is no agreed-upon industry standard, although solutions proposed by the Globus Alliance are in the standards process at the IETF [TWE+04].

In addition to these general concerns, users identified several tools needed in the security area:

- Tool to verify the network connectivity between clients and services to ensure that any firewall configuration changes (or other network alterations) would not inhibit established user activity
- Tool to verify that a service was secure, which in most cases meant that the messages being passed were encrypted or that no one could erase needed data from a system
- Security audit tools to verify that patches and known exploit-prevention software were still functioning properly after upgrades or changes to the system

4 Functionality

Almost every group we spoke with was using Grid environments to support their applied science activities, so the functionality they wanted was for their day-to-day work, not farther-out speculative needs. These primarily fell into two categories: job submission and tracking, and file transfers. A few projects were using tool “add-ons”, such as visualization tools, data format translators, or policy management tools, but these were always strongly tied to the project domain and narrowly scoped. When asked about other possible functionality or services that could be used, we were told these were not on the six-month horizon most groups were currently considering (see Section 4.3 for more detail).

4.1 Job Submission and Tracking

Job submission was the most common first service in use by the projects we spoke with. For most projects, this was a simple, dependable, “run my application” interface that was in the users “comfort zone” and behaved as expected. Different users defined their comfort zones differently; indeed, most middleware developers believe the phrase “behave as expected” to be nondeterministic. However, every group we spoke with was performing job submission, and many had adapted a standard tool, such as the Globus Toolkit job submission [GRAM] or Condor [Condor], for project-specific use.

Job submission was generally being done on well-known resources or services. Users were not trying to figure out what machine or service to work with. They had a small set that were used most of the time. Occasionally a user would have policy questions of the resource (“How many free Matlab licenses are there I can use?” “How many jobs can I submit to the queue today)”, but the larger discovery questions were not an issue for these users.

The functionality associated with job submission that was most commonly felt to be missing (or in current development by a project) was a way to track jobs once they had been submitted. Most groups reported problems in which a job had been submitted (or a service request had been made) and something had not performed correctly, but they were unable to determine where, why, or how to fix that problem. Groups admitted that they frequently used Grid tools for job submission and file transfers but resorted to SSH to debug what was happening on the system. Every group had experienced the phenomenon of a job run completing as expected one day but failing on the next for unknown reasons.

Several tools were identified as needed:

- Tools to aid in failure identification
- Better logging services to debug failure causes, and debugging paths through the system and for these logs to be centralized (from the user's perspective) to provide a single point to start the investigation
- Job-tracking services, in general

4.2 File Transfer

Most users were transferring files using Grid tools such as GridFTP and were happy with the service level they experienced. Some groups needed reliable file transfers, either because they had many small files to transfer and it was easy for one in a thousand to have problems and be left behind, or because they had such large files that the file transfer time was greater than the mean time to failure for some system component, often a flaky network connection. Nevertheless, since most reliable file transfer services require delegation, several groups were not sure how to move forward on this front.

A few groups we spoke with were beginning to examine higher-level file transfer services, such as provenance services, access to databases, or replication, but these groups were still primarily prototyping these efforts.

The tool most commonly requested in this space was one to help diagnose the problems, including that of slow performance, seen on systems when performing large file transfers. Performance on WANs still has a very high variability because many components are involved and contention can vary widely over time. Invaluable would be a tool to help users understand where a problem is being caused so they can better understand who to contact.

4.3 Other Services

What surprised us most about the tools and services in use by the groups we spoke with was not what they were using but what they *weren't*. Following is a list of services not currently considered to be essential by a significant majority of this set of users:

- Notification, except for job progress tracking

- Registries or resource discovery
- Reservations, brokering, coscheduling, other advanced scheduling techniques
- Job migration or application checkpointing
- Accounting and pricing
- Data migration
- Instruments

The reasons for these exclusions are many. Most of the projects we spoke with were hands-on, application-oriented approaches as opposed to research-oriented projects. The software most of them were using was expected to be of production-release quality; they were not interested in prototypes or proof-of-concept software that was not resilient to failures. And in general, the groups were having enough of a challenge getting the basic functionality up and stable, so higher-level services were not considered an immediate priority within the next year.

5 Tools

In our discussions, users expressed a wide variety of opinions—often contradictory—about what they would like tools to look like. Most of what was stated came as no surprise. What was surprising, however, was the importance given to some issues.

One of the chief complaints was that many Grid tools offered horizontal functionality and not end-to-end solutions for a given problem. Users requested one-solution tools that would work easily for the 80% rule, and could be used for the rest. Of course, what 80% was for one group was not necessarily what it was for another; we had requests both to have everything needed to use the tool bundled together in one place, and yet to avoid reinstalling software already present on the system.

Users also suggested having smaller, simple tools that could be composed together similar to the UNIX piping mechanism. For instance, users wanted to be able to build a workflow by picking and choosing from basic functions, while using small scripts ("shims") to fit between these functions when necessary to translate between mismatched APIs or data formats..

APIs were another topic that users strongly debated. Users agreed that tools should have simple, compact APIs at the user level, and they strongly felt that this API could be different from the one used by a developer if necessary. They also wanted the user API to be in the language or environment of the user's community. For example, if the tool was for biologists, a Perl interface was recommended; if the tool was for the physics community, Python would be more appropriate.

6 Builds, Upgrades, and System Stability

One of the main problems that users focused on was the lack of reproducible and verifiable builds and the lack of general stability in both builds and tool use in much of today's Grid software. A concern was the large size and complexity of current software distributions that make debugging build failures and any incremental development work on the code base very difficult. With an increasing number of commercial and research organizations offering software components, simplification of build, packaging, and dependency infrastructure was seen as a priority in order to promote interchange.

When building middleware, users and system administrators wanted a hands-off process that would work the same way every time the software was built. In addition, users expressed the need for verification tools—ways to be sure that the build was successful and that the full desired functionality was installed properly. They wanted the prerequisites for all modules to be verified at the start of the build process, in order to ensure that the build would run to completion. They wanted errors or further configuration actions to be listed at the end of the build output and not reported intermittently during the build process.

Similar needs were expressed for upgrades. Users wanted the documentation for an upgrade to include detailed descriptions of the changes to the system and possible incompatibilities, and to be as straightforward as possible, with verification tests similar to those desired for initial builds. Users recognized that many services were being used "off label," that is, services written for one function were being used in a setting not envisioned by the designer, and hence upgrades were inevitable. What was desired, however, was a more explicit enumeration of the *effects* of the upgrade. Being able to upgrade components within a distribution rather than reinstalling the whole distribution was seen as highly desirable. Likewise, users wanted the installation of software requiring privileged access to be minimized and separated from the main build process in order to allow different administrative roles to perform different actions, for instance, system configuration, from database configuration and the build user.

In general, one of the strongest requests was for better system stability and understanding of system stability. Far too often an application that runs perfectly well one day will fail the next, frequently for no easily discernible reason. Because many Grid systems involve tens if not hundreds of components, the mean time to failure has decreased significantly, and better monitoring of the background system is needed in order to detect and debug these issues before they affect users. Many systems are running benchmarks of verification suites, but these tests frequently do not "look like" user applications. For example, just because ping is working between two systems does not mean that large file transfers will also function.

Users expressed a need for tools to help debug why failures happened, and who to talk to in order to fix them. For example, almost every group we spoke with had had difficulties transferring large files at one time or another, and not known what was going wrong. Simple "common practices" documents or tools to help users walk through the path of their file transfer would go a long way to addressing these issues.

Having established that the system works, users also wished to see how well it works. Those groups not experiencing outright failures still wanted additional information in order to better understand the performance characteristics of their applications.

7 Overarching Concerns

During our meetings we identified eight open areas of concern that were repeated by many groups. In no particular order, these areas are as follows:

- **Training and education, especially for security concerns.** Security is seen as extremely challenging, and system administrators, developers, and users all want more information about common practices and current approaches. (Sections 2 and 3)
- **Delegation for Web services.** Many current Grid tools need to be able to perform delegation, and the lack of an industry standard or even a well-understood set of tools for Web services is of great concern. (Section 3)
- **Job tracking.** Having conquered the initial challenge of job submission using Grid tools, users are now concerned with understanding where a job is in its lifetime, where it is failing, why, and what to do next. (Section 4.1)
- **Dependable builds.** Software that builds nondeterministically, is hard to install, or doesn't include verification test suite is seen as unacceptable by today's users. (Section 6)
- **Composability for functionality.** The desire to have tools perform individual functions has been supplemented by the need to be able to compose these functions together in order to achieve a chain of services to solve application specific problems. (Section 5)
- **Wrappers for usability.** Most users want a layer between them and the tool in order to bring the functionality into their own comfort zone. These wrappers do not add functionality per se but significantly increase the usability and usefulness of a service. (Section 5)
- **Verification and instability analysis.** With the overall time to failure for Grid components decreasing as their number increases, there is a strong need for better verification and instability analysis to discover and resolve problems before a user happens upon them. (Section 6)
- **User-oriented diagnostic tools.** Most diagnostic tools solve problems other than those seen at the user-level. Tools that look like normal user applications and can help an average user diagnose failures are a strong current need. (Section 6).

8 Summary

Over the course of several weeks in July and August 2004 we spoke with 25 UK eScience project groups about their use of Grid functionality and services. What resulted is a picture of current application and user needs of these services, and some suggestions for ways to move forward. This data is now influencing the directions of both the Globus Alliance and the OMII.

The strongest result that came from these discussions was the simple need for on-going conversations between tool developers and users. Grid tool developers must continue to talk and interact with application scientists; without such interaction, the tools are for nothing.

9 Acknowledgements

This work has been supported by the UK e-Science Core programme (through the Open Middleware Infrastructure Institute) and by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract W-31-109-ENG-38. We would like to thank all the people we visited for their hospitality and the uninhibited technical discussions.

Appendix A: Project Listing

R. Baldock, MRC Human Genetics Unit and NeSC, Edinburgh Mouse Atlas Project
http://genex.hgu.mrc.ac.uk/

The MRC-funded Mouse Atlas provides an international database of mouse embryo data in close collaboration with other mouse resources around the world. The atlas provides a 3D+time spatio-temporal framework for mapping in situ gene-expression data. The data is a combination of 2D and 3D expression patterns and images, which are spatially mapped to allow spatial as well as textual query. Some data requires significant computation for reconstruction, mapping, analysis and visualisation. This project is in the process of moving from a CORBA-based infrastructure to a Web services infrastructure. A primary consideration in the selection of the technology is stability.

M. Baker, Portsmouth, OGSA Testbed
http://dsg.port.ac.uk/projects/ogsa-testbed

The consortium of Manchester, Westminster, Reading, Daresbury Laboratory, Portsmouth, and Southampton is one of two EPSRC-funded projects to deploy and evaluate the Globus Toolkit v3. While many problems were found with this initial deployment, this group aided in resolving them in software and documentation and in development of additional tools for use in UK eScience projects.

R.Baxter, EPCC, eDIKT
http://www.edikt.org/

The ELDAS component of the SHEFCE-funded eDIKT project uses standard tools, including Eclipse, and infrastructures to provide access to data resources. A partial implementation of the Global Grid Forum's DIAS (Data Integration and Access Standard) Working Group specification has been developed using EJB's hosted within a JBoss container.

N. Chue Hong, EPCC, OGSA-DAI
http://www.ogsadai.org.uk/

The OGSA-DAI project, and its follow-on OGSA-DAIT, provides a reference implementation of the GGF DIAS specification, which defines uniform access to federated data sources (including files) that may be stored in more than one heterogeneous database. Issues facing the project include the ability to delegate actions across different infrastructures and to execute efficient non-file-based transfers between services where the dataflow at the client and the server may be adjusted in response to varying network conditions. OGSA-DAI is funded by the DTI.

D. Chadwick, Salford, PERMIS
http://www.permis.org/

The discussion with Chadwick covered several projects exploring authorization technologies based around PERMIS and SAML, and projects deploying these solutions in order to evaluate their effectiveness. It was found that these technologies (while relatively mature) needed supporting management tools (e.g., a graphical interface to define and manipulate the XML Security Policies) and an attribute authority infrastructure to contain and manage user roles and capabilities, such as such as SIGNET or the Community Authorization Server (CAS). Many of these projects are based around X.509 certificates, but work is also being done to use Shibboleth as a mechanism to undertake inter-organisation authorisation assertions. These projects are funded by JISC and EPSRC.

D. Colling, IC, GridPP2
http://www.gridpp.ac.uk/

GridPP2 is the current focus of Grid activity for high energy physicists in the UK and involves substantial middleware development as well as deployment on a wide variety of testbeds. This software must be able to accept thousands of jobs at a central broker that must also handle file staging from remote replicas. PPARC funds this work.

T. Cooper-Chadwick, Southampton, gYacht/gShip
http://www.soton.ac.uk/~gyacht/

The Southampton e-Science Centre projects G-Yacht and G-Ship focus on delivery of usable performance predicting (such as speed or seakeeping) design tools for yachts and ships through computational modelling on the Grid. Interaction with the Grid resources is through a portal and HiCOG, which provides access to GT2 and Condor-enabled compute resources.

S. Cox, Southampton, GeoDise
http://www.geodise.org/

GeoDise is an EPSRC-funded pilot project developing an infrastructure to support engineering optimization through the evaluation of parameterized designs on the Grid. Its main feature is the integration of a Grid-capable Matlab. This functionality covers three primary areas, called as toolboxes: compute (creating a proxy, launching a job, etc.), data (enabling files to be archived, queried, and retrieved through file based metadata), and the conversion of Matlab data structures into XML and vice versa. Generic Web services can also be imported into the environment and invoked from within Matlab.

M. Daw, Manchester, Access Grid & MUST
http://www.agsc.ja.net
http://www.sve.man.ac.uk/Research/AtoZ/MUST/

The JISC-funded Access Grid Support Centre (AGSC) will provide support for UK AG deployments and central services such as an IG Recorder and IG Pix (from inSORS), a virtual venue server and an H323 bridge. The MUST (Multicast Streaming Technology) project is exploring reliable multicast protocols to support Grid applications.

W. Emmerich, UCL, eMinerals and OGSI Testbed
http://eminerals.org/

As part of the e-Minerals project, Condor pools at UCL, Cambridge, Bath, and Reading have been linked by using Condor-G and the Globus Toolkit v2. Federation of Condor pools using Web services is now being explored through a Core programme-funded project in collaboration with the Condor team. Resources are orchestrated through a server-side Business Process Execution Language (BPEL) engine that invokes remote services.

M. Ghanen, Imperial, DiscoveryNet
http://www.discovery-on-the.net/

The DiscoveryNet project, funded by EPSRC, is motivated by knowledge discovery within environmental modeling, geohazard modeling, and gene expression array chips. It integrates different distributed data sources (databases, Web sites, etc.) into a single workflow using a graphical paradigm. Once defined, this workflow can then be "published" and made accessible through a portal.

M. Giles, Oxford, gViz
http://www.visualization.leeds.ac.uk/gViz/

The gViz project enables the visualization of a computational fluid dynamics simulation statically and spatially partitioned by using Metis on a clustered resource from a remote location. MPI is used to communicate between clusters. This work also uses gSOAP, a package that exposes C code through a WSDL interface, to provide a message passing layer, similar to PVM, between nodes. DTI and EPSRC fund this work.

S. Lloyd, Oxford, eDiamond
http://www.ediamond.ox.ac.uk/

Funded by DTI, eDiamond uses GT3 and OGSA-DAI to expose medical records for viewing and analysis between different medical centres. The primary challenge within the project is dealing with the security issues relating to the viewing of medical data. The deployment of this infrastructure on remote resources has been simplified by the use of scripts to ensure repeatable hands-off installation and configuration.

C. Goble & N. Sharmen, Manchester, ^{my}Grid and Integrative Biology Project
http://www.mygrid.org.uk/
http://www.integrativebiology.ox.ac.uk/

The ^{my}Grid project supports a variety of biology experiments on the Grid, with a focus on semantic properties. A strong element of the project is the integration of different processes (e.g., invocation of a BLAST query on a remote server or looking up sequence information from a database) into an analysis workflow by the user. The provenance of this workflow is recorded, allowing any derived results to be fully described and recreated at a later date. ^{my}Grid makes use of and has also contributed significantly to the development of the Taverna workbench, the Freefluo workflow enactment engine and the Scufl workflow language. These projects are supported by EPSRC and BBSRC.

J. MacLaren and J. Brooke, Manchester, Brokering activities at Manchester Computing
http://uombroker.sourceforge.net/docs/server/overview-summary.html

Brokering is an important area in Grids for which there are many experimental solutions but few production services. One such broker, developed by the University of Manchester initially under funding from EUROGRID and GRIP, works with the Unicore infrastructure and is now being further developed within European projects such as UniGridS and DEISA (Distributed European Infrastructure for Supercomputing Applications).

A. Martin, Oxford, ClimatePrediction.NET
http://climateprediction.net/

ClimatePredicition.NET, funded by NERC, uses a structure similar to SETI@Home to simulate future possible climates. A desktop client retrieves an initial data set from a central server to initiate the simulation. On completion the final climate model is uploaded to one of several servers. These servers are accessible by the scientists attempting to derive knowledge from the simulations by linking the initial conditions and final solution.

M. McKeown, Manchester, OGSI:Lite and WSRF:Lite
http://www.sve.man.ac.uk/Research/AtoZ/ILCT

The PERL-based containers OGSI:Lite and WSRF:Lite have been used to develop persistent services and to expose applications as services. C-based clients interact with the WSRF:Lite container using gSOAP before accessing a C-based service wrapped in PERL. The container can be deployed on platforms that do not support Java and can be run through CGI under Apache.

Andy McNab, Manchester, GridPP2
http://www.gridpp.ac.uk/

During GridPP1 considerable work was done to develop several elements of access control infrastructure. One activity, GridSite, required the development of GACL (Grid Access Control Language) and supporting libraries. This language has been reused in several projects with interfaces from Python, Perl, C, and C++.

S. Pickles, Manchester, TeraGyroid and GRENADE
http://www.realitygrid.org/TeraGyroid.html
http://mrccs.man.ac.uk/research/grenade/

The GRENADE project has used a plugin mechanism within the KDE desktop to provide access to remote compute resources on the Grid for GT 2.4. The scientist is able to use the familiar "drag and drop" paradigm to launch jobs on remote resources and integrate remote files spaces into the local disk space.

A. Porter, Manchester, RealityGrid, and M. Rider, Manchester, eViz
http://www.eviz.org/

The RealityGrid and eViz projects, both funded by EPSRC, are concerned with the running and steering of physical simulations, and providing on-line visualization of application data. Pre-Web service components of the Globus Toolkit are used to support third party data transfers into and out of the computational resource. Job execution, monitoring, and debugging are performed by using GRAM (although the use of other systems such as Unicore have been explored) and SSH to access standard output and error logs.

A. Rector, Manchester, CLEF
http://www.clinical-escience.org/

The MRC pilot project CLEF is investigating how clinical care can be linked to post-genomic databases to add gene based reasoning into the treatment process. A key challenge is to ensure that the clinical records have to be cleansed of patient information before being exposed to the other services. The resulting infrastructure will build on elements of MyGrid (e.g., workflow, portal and Web services).

R. Sinnott, Glasgow, BRIDGES
http://www.brc.dcs.gla.ac.uk/projects/bridges/public/people.htm

BRIDGES (Bio Medical Informatics Grid Enabled Services) is a two-year DTI-funded project being used to explore authorization to medical data. This is one of several security-related projects including DyVOSE, which is exploring the dynamic delegating of trust when issuing certificates. More recent work has provided a GT3 interface to running BLAST on ScotGrid resources.

L. Smith, EPCC, QCDGrid
http://www.epcc.ed.ac.uk/computing/research_activities/grid/qcdgrid/

The PPARC-funded QCDGrid provides a distributed data store with some interactive analysis capability at four sites for the U.K. QCD community. GT2.4 software including replica management tools was extended to enforce project-specific policies. Work is now beginning on defining common interfaces to enable interaction with other QCD-based Grids in the United Stated.

T. Sloan, EPCC, INWA
http://www.epcc.ed.ac.uk/inwa/

The Innovation Node: Western Australia (INWA) project was funded by the ESRC Pilot Projects in e-Social Science programme to inform businesses and regional policy regarding Grid computing. This project involves the integration of private commercial data with publicly available datasets using a Grid infrastructure between the U.K. and Australia consisting of Globus Toolkit 2 and 3, Sun Grid Engine, Transfer-queue Over Globus (TOG), OGSA-DAI, and the Grid data service browser from the FirstDIG project.

L. Yang, B. Yang, NeSC, AI Workflow
http://dream.dai.ed.ac.uk/e-Science/

AI Workflow, which is in the early stages of research, aims to use proof planning technologies to define and map workflows to resources. The longer-term goal is to use this work to define and exploit different qualities of service to different resources.

References

[Condor] "The Condor Project," http://www.cs.wisc.edu/condor/

[Globus] "The Globus Alliance," http://www.globus.org

[GRAM] “Grid Resource Allocation and Management (GRAM),” http://www.globus.org/gram

[Hillier02] “The Use of Firewalls in the U.K. e-Science Grid: ETF Level 2 and beyond”, Jon Hillier, 2002, http://e-science.ox.ac.uk/events/firewall-workshop/FirewallIdeas.pdf.

[NNFN] “Networks for Non-Networkers Workshop,” University College London, UK, July 13–14, 2004, http://grid.ucl.ac.uk/NFNN.html

[OMII] “Open Middleware Infrastructure Institute,” http://www.omii.ac.uk

[SecWS] “UK Workshop on Grid Security Practice,” Oxford, July 8–9, 2004, http://e-science.ox.ac.uk/events/security-workshop/

[SerWS] “Service Grids: Current Activity & Middleware Requirements,” NeSC Edinburgh, UK, July 22–23, 2004, http://www.nesc.ac.uk/action/esi/contribution.cfm?Title=415.

[Surridge02], “Rough Guide to Grid Security”, Mike Surridge, 2002, http://www.nesc.ac.uk/esi/events/121/RoughGuidetoGridSecurityV1_1.pdf.

[TWE+04] S. Tuecke, V. Welch, D. Engert, L. Pearlman, M. Thompson, “Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile,” IETF RFC 3820, July 2004, http://www.faqs.org/rfcs/rfc3820.html.

[Welch03] V. Welch, “Globus Toolkit Firewall Requirements, version 5,” July 2003. http://www.globus.org/security/firewalls/Globus%20Firewall%20Requirements-5.pdf