GLAS-PPE/2004-05

UNIVERSITY
of
GLASGOW

Department of Physics & Astronomy
Experimental Particle Physics Group
Kelvin Building, University of Glasgow
Glasgow, G12 8QQ, Scotland
Telephone: ++44 (0)141 339 8855 Fax: +44 (0)141 330 5881

A Grid for Particle Physics – from testbed to production

The GridPP Collaboration
http://www.gridpp.ac.uk/collaboration_members

This paper was compiled on behalf of the GridPP collaboration by: D. Britton, P. Clarke, **J. Coles**, D. Colling, A. Doyle, S.M. Fisher, A.C. Irving, J. Jensen, A. McNab and D. Newbold

Abstract

The GridPP project, in close association with the European DataGrid (EDG) and the LHC Computing Grid (LCG) projects, reached a key milestone this year with the successful deployment of a production Grid testbed. This paper describes the value-added middleware developed to make the testbed function for users across the globe, provides some examples of the use applications have made of it and shares many of the lessons learned. The final section briefly reviews the GridPP approach for moving from a testbed-oriented project to a production service able to meet the needs of the Large Hadron Collider (LHC) and other particle physics programmes.

A Grid for Particle Physics – from testbed to production

Presented by the GridPP Collaboration
http://www.gridpp.ac.uk/collaboration_members

This paper was compiled on behalf of the GridPP collaboration by: D. Britton , P. Clarke, **J. Coles**, D. Colling, A. Doyle, S.M. Fisher, A.C. Irving, J. Jensen, A. McNab and D. Newbold

Abstract

The GridPP project, in close association with the European DataGrid (EDG) and the LHC Computing Grid (LCG) projects, reached a key milestone this year with the successful deployment of a production Grid testbed. This paper describes the value-added middleware developed to make the testbed function for users across the globe, provides some examples of the use applications have made of it and shares many of the lessons learned. The final section briefly reviews the GridPP approach for moving from a testbed-oriented project to a production service able to meet the needs of the Large Hadron Collider (LHC) and other particle physics programmes.

1 Introduction

The GridPP[1] project, in close association with the flagship EDG[2] and LCG[3] projects, has achieved remarkable successes in the last three years, leading to a pervasive Grid testbed existing throughout Europe and including the UK as a major component. It is perhaps not always appreciated that this is a highly functional production Grid prototype, to which real user production jobs can be submitted in a properly manageable way across national boundaries. Put simply, a user can submit a job to a broker in Italy, and find it running in Glasgow or Budapest a few minutes later. There are few such Grids in operation in the world today. This success was perhaps best underlined at the February 2004 EU Review of EDG where all participants were commended for the level of achievement attained by the development project - which resulted in approximately 1 million lines of middleware code being developed. EDG members have continued their participation in this work by migrating to the LCG and Enabling Grids for E-Science in Europe[4] (EGEE) projects which focus on deployment. A snapshot of the European LCG sites is shown in Figure 1; an estimate of available LCG resources is shown in Table 1.

2 The infrastructure

Deploying low level tools, such as Globus[5], does not in itself produce a computing Grid. To enable a diverse set of users scattered throughout the world to access such a Grid requires several "value-added" components. GridPP and EDG developed good working prototypes for all of these, and in this section they are reviewed.

Figure 1: A snapshot of the European LCG sites in June 2004. There are also sites (not shown) in Taiwan, Japan and America. For a live map see the LCG operations website[6].

2.1 Workload Management[7]

The first step in using the Grid is to describe a job in a way that the Grid Middleware can understand. This is done using the high level Job Description Language (JDL) developed by the EDG project based on the Condor ClassAd language[8]. Using this JDL a user is able to specify the characteristics of the job itself (the executable name, the parameters required, the number of instances to consider, standard input/output/error files etc.), its required resources (CPU, storage etc.), and even how the execution sites should be ranked (e.g. by fastest CPU, greatest number of free CPUs etc.). Importantly the data required by the job is also specified in the JDL script. After the job has been described, the JDL script must be communicated to the Grid Middleware. This is done either by using an appropriate Application

Programming Interface (API) or via an easy to use command line and graphical User Interface (UI).

Country	CPUs	Storage (GB)	Country	CPUs	Storage (GB)
Austria	5	600	Japan	100	1900
Canada	220	1800	Netherlands	300	4400
CERN CH	850	1005000	Pakistan	0	70
Czech Republic	80	90	Poland	10	100
France	2	75	Portugal	5	40
Germany	225	1800	Russia	10	20
Hewlett Packard	25	30	Spain	340	2000000
Hungry	85	20	Taiwan	300	3200
Israel	35	200	UK	430	1005000
Italy	1000	8000			
			Total	**4022**	**4032345**

Table 1: An estimate[9] of the number of CPUs and Gigabytes of storage available by country to LCG users in June 2004.

The UI software is "light-weight" and generally installed on the user's desktop machine. This software is able to communicate with a service node - the heart of the workload management system. Whereas each desktop may have the UI installed, there are typically only a limited number (one or two) of service nodes per user community. Each service node runs a number of critical processes which effectively combine to produce a super-scheduler:

- *Network Sever* - a generic network daemon that is responsible for accepting incoming requests from the UI.
- *Workload Manager* - given a valid request this process decides how to satisfy the request and takes the appropriate action.
- *Match Maker* (or *Resource Broker*) - supports the Workload Manager. Given JDL specifications the resource broker interacts with the Grid information services to find the resource best matched to its requirements.
- *Job Adapter* - performs final job formulation and submits the job to the Job Controller.
- *Job Controller* - is responsible for the actual job management; it is essentially a wrapper around CondorG[10] for the job submission.
- *Log Monitor* - monitors the progress of jobs through CondorG by parsing the CondorG log files.
- *Logging and Bookkeeping* (L&B)[11,12] - gathers information from different components of the workload management system to provide a state view of each job.

In addition to the fundamental elements described above, more advanced functionality was developed. This includes mechanisms for dealing with dependent jobs (where for instance a job will only start if previous jobs were successful), job checkpointing (to allow automatic recovery of interrupted computations), interactive steering of jobs and job accounting.

In developing the Work Management System (WMS) described above, limitations in the scalability of the Metacomputing Directory Services (MDS) had to be overcome. On occasion an incorrectly configured or very slow MDS could hang and break the Grid. Adding a Berkely database to provide a cache of the latest good data when the MDS had failed helped to resolve this problem. Eventually a Relational Grid Monitoring Architecture (R-GMA) implementation (see next section) was used instead of MDS. It was also found that a single rogue site (for example one that dumps all jobs received or publishes wrong information – such as a zero estimated completion time) could attract jobs but not process them. Later releases of the WMS were more modular and reliable; use of new daemons to check and restart services provided a big increase in reliability.

The WMS described was implemented and deployed around the EDG testbed with service nodes at 5 sites around Europe. During the final 3 months of the EDG Project some 60,000 jobs were run through it using the service nodes at NIKHEF and Imperial College London. The Workload Management Sytem has been used by other Grid projects including DataTAG and GRID.IT. It is also the basis of the current standard release (LCG2) installation of the LCG Workload Management System.

The WMS architecture and components are being re-engineered to use web services. Only when the performance of the current system is surpassed by the re-engineered system will it be replaced. Even then, many of the re-engineered components will be based on their pre-web service equivalents.

2.2 Information and Monitoring

To operate a Grid, static and dynamic information about its various components (Grid resources, applications and networks) must be shared and monitored. Within EDG this was achieved through a relational implementation of the Grid Monitoring Architecture (GMA) of the Global Grid Forum [13] (GGF); this implementation is known as R-GMA[14].

R-GMA makes use of a relational model where the interfaces allow users to publish information via an SQL *Insert* statement and to issue queries using an SQL *Select* statement. The system creates the impression of one large Relational Database Management System (RDBMS) per Virtual Organisation (VO). All information is time stamped, with support for continuous or one-off queries [15]. Within EDG it was used to provide information to the resource broker on details of the available resources. It was also used for network monitoring, service status and was being tested by some experimental groups for monitoring the status of their jobs.

By the end of the EDG project the code had stabilised well and could be relied upon by the resource broker. A great deal was learned about producing a scalable robust system. Early versions of the code suffered from the creation of too many threads, the use of too many sockets and the use of too much memory. Correcting these problems required redesign work. Development benefited greatly from a very large number of unit tests. However a distributed system can only be partially tested this way. It was necessary to have a quite sizeable dedicated testbed on which to run stress tests in order to overcome identified timing issues. At the time of writing, the code is in the final stages of being approved by LCG[3] to be added to its standard release package. In the first instance it will be used primarily for the LHC experiments to monitor their production jobs. This monitoring requires that a job (or a job wrapper) publishes information to R-GMA. This is collected together and republished by a Secondary Producer. The user can then issue queries to determine the status of all their jobs wherever they may be running.

The R-GMA code is currently being re-engineered as a component of EGEE. The changes being made include: migration to web services, replication of the Registry and Schema (to avoid a single point of failure), support of multiple Virtual Organisations and a much more elegant API, allowing further types of Producer to be introduced easily. The work is described in more detail in another paper at this conference.[16]

2.3 Security and Virtual Organisation Management

For a Grid to function, a scalable and comprehensive identity and credential management system is required to allow resource providers to verify identity and capability of people belonging to large distributed VOs. Without this a scalable Grid, serving many and varied users, is not possible.

Most of the middleware used by GridPP and LCG relies on public key cryptography in the form of X.509 [17] user and server certificates for authentication. This is the basis of the authentication and authorization systems of the Globus Tookit[5], the EDG middleware [18] (developed by GridPP and its EDG partners), and the industry-standard secure website technology, HTTPS [19]. Currently, the middleware involves three classes of component[18]: Certification Authorities which furnish users and services with certificates; Authorization Servers which define Grid-wide group memberships and roles; and Local Policy decision points (through which local operations are permissible given a set of credentials).

X.509 user and server certificates were initially issued by the UK High Energy Physics Certification Authority at the Rutherford Appleton Laboratory. This Certification Authority was operated as part of GridPP to enable members to participate in EDG and other projects; it was able to operate relatively quickly due to the small size and history of previous collaboration within the High Energy Physics computing community. This played a significant role in the rapid start up of the initial UK-wide GridPP Grid. As the project matured, use was made of a newly formed UK e-Science Certification Authority also located at RAL.

GridPP members led the EDG efforts to agree standard practices to be followed by the many national Certification Authorities used for Grid authentication. This effort has not only enabled GridPP to accept certificates from outside the UK with the confidence that similar verification procedures have been used (for example, requiring photographic ID), but also for UK eScience certificates to be trusted by many Grid projects across the world.

Authorization servers[18] were developed, and are now available, in both Pull mode (in which sites periodically pull a list of valid members from a central service) and Push mode (in which users obtain a short-lived attribute certificate which they present to sites to prove group membership). Pull mode servers have been deployed using the VO-LDAP, GridSite LDAP (Lightweight Directory Access Protocol) and VOMS-httpd systems; and Push mode using Virtual Organisation Management Service (VOMS) attribute certificates[18].

Under EDG local policy decisions were generally made using the Globus "grid-mapfile" mechanism (a simple list of acceptable users' certificate names), by the EDG Local Centre Authorization Service[18] (LCAS), or by the Grid Access Control Language[18] (GACL) policy language and the associated parser. However, some local policy decisions were made implicitly, according to the Unix filesystem permissions associated with the Unix users to which each Grid user was mapped. This remains the least mature portion of the security system, since many components still use the coarse-grained Globus "grid-mapfile" mechanism and Unix filesystem permissions. This has several limitations, most notably the inability of a single user to use multiple group or role memberships. The fine-grained XML policies used by GACL, which support authorization both by user certificates and by VOMS attributes, will resolve these limitations when fully used in the deployed Grid.

Further middleware development is planned as part of the EGEE project. This will involve full use of the existing fine grained GACL policy support in the HTTP(S) GridSite server, the workload management logging service, and the LCAS site authorization service. New components are being developed to provide a Grid-based security environment for Web Services, both in Java and using the GridSite extensions to Apache (the latter for services written in C/C++ and scripting languages).

2.4 Storage Element

The EDG Storage Element[20,21] (SE) was designed to provide a uniform interface to mass storage systems and disk and to integrate with EDG Replica Management Services[22]. It was developed with secure interfaces and a flexible architecture with pluggable features such as access control. In EDG, it was deployed as an interface to the CASTOR[23] storage systems at CERN and UAB Barcelona, to the Rutherford Appleton Laboratory (RAL) ATLAS storage facility, and to IN2P3's HPSS system. It was also deployed as an interface to disk storage at various sites throughout Europe.

While most EDG middleware could be deployed on "standard" machines with "standard" operating systems, the mass storage systems often had specific requirements. For example the Rutherford Laboratory mass storage system required AIX, whereas other sites needed Irix. CASTOR provided data access libraries that were installed in the same location as those of HPSS, but they were subtly different. The impact of this was that the effort required supporting installations and configuration of the interface was relatively large; often the whole team was involved in support issues, which constrained further development. The effort required to support some biomedical mass storage systems (DICOM servers) was significantly underestimated, at least in part due to stringent security and confidentiality requirements, so this element of work was not completed within the lifetime of EDG. However, it is being investigated again within the EGEE project[24].

The SE was deployed as a tool to transfer data between CERN and RAL for the ATLAS experiment. Higher-level tools performed the actual transfers between SEs and registered the files in replica catalogues. This was very successful as the system was stress-tested leading to several bugs being identified and fixed. Overall it was found that interoperability is highly desirable and as a result work is now underway to standardise and implement a Storage Resource Manager (SRM)[25] interface. In fact, the SE's original interface was based on SRM, but simplified due to project deployment time constraints. This work area benefited greatly from software reuse, for instance GACL[26] was used to implement access control.

2.5 Data management

Data management work comprises two key areas, Grid Metadata Management and Grid Query Optimisation.

Grid Metadata is organised information about Grid users, data to be processed and Grid resources. It is required in order that automated interactions between data representing these elements are possible. Within EDG, a technology demonstrator service called Spitfire was developed to meet the challenge of managing metadata. There are a variety of papers and presentations discussing the progression of Spitfire and its web based technologies. These together with the code and documentation for the

Spitfire service can be obtained from the EDG-WP2 site [27].

GridPP members have been involved in the DAIS [28] working group of the Global Grid Forum to ensure that Particle Physics use-cases are being adequately dealt with in this forum. Metadata management is essential to the experiments, but was not hitherto properly or coherently addressed. In recognition of this a UK metadata management group has been formed[29]. This group will work on the overall coordination and implementation of metadata technologies across the HEP experiments. The architecture and implementation of the data management services under demanding Grid conditions is the topic of another paper[22].

Grid Query Optimisation is required because analysis jobs running in a Grid environment will access a large number of data files and many will not be local to the system on which the job is running. The purpose of optimising access is two-fold: to speed up the individual analysis job for the end user and to increase the global throughput of the Grid. Experimental optimisation algorithms are being developed, and in order to test and evaluate these, a data-centric Grid simulation program, OptorSim, has been built. The simulation code and results so far available are on the OptorSim site[30].

Some of the more basic optimisation work has been applied in the Replica Optimisation Service (ROS) of EDG's replica management system. This service implements functionality (getBestFile) that allows the Grid to decide which replica to fetch for a job based upon live network and storage monitoring statistics. Future work in this area is looking at use of the SE for automatic mirroring of data.

3 Application Usage Examples

The prototype EDG production Grid was used for developing and testing several real life particle physics applications. We give two examples here. It was also used for several biomedical applications and a summary from this work is also given in this section.

3.1 CMS Experiment

CMS[31] is one of the general-purpose detectors for the CERN LHC accelerator, due to start full operation in 2007. LHC is a major driver for the deployment of Grid technologies in high-energy physics, since it will produce highly complex data at a rate unprecedented in either science or commerce. The CMS experiment is currently in the final stages of defining its data acquisition and analysis strategy, and has already exercised a prototype worldwide computing system at the 5% scale. By 2007, CMS will produce several petabytes of data per year, all of which must be made available to physicists worldwide.

The structure of computing for CMS will reflect that for LCG more generally, with a hierarchy of computing centres. A central Tier-0 centre will be located at CERN, and will encompass the online computing system (trigger farm) and prompt reconstruction facilities. Raw and reconstructed data will be distributed to Tier-1 centres for further processing and bulk analysis, as well as second-pass reconstruction where necessary. CMS plans to deploy up to five Tier-1 centres, one of which will be managed through GridPP and sited at Rutherford Laboratory. The GridPP Tier-2 centres will also be available to CMS physicists.

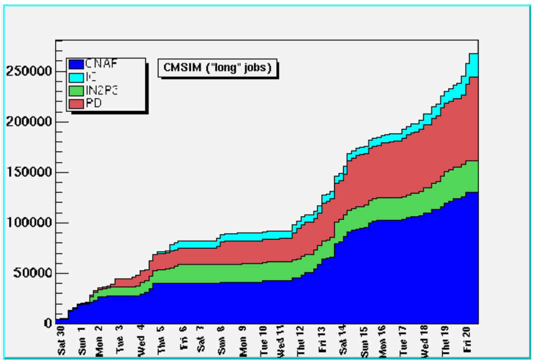


Figure 2: Number of events produced as a function of time (in days) in the CMS pre-challenge production. The different colours represent different Tier-1 and Tier-2 centres across Europe.

In order to smoothly ramp up the worldwide computing capacity available to the collaboration, and to reach a full understanding of the associated challenges, CMS has carried out a series of 'data challenges' (DC) of increasing scale and complexity. These challenges have several purposes, not least of which is to provide very large simulated data sets to the collaboration in order to refine its reconstruction and analysis approach. The latest challenge[32], DC04, took place in March-April 2004, and consisted of operating the entire computing system end-to-end at an event rate of 25Hz and simulated luminosity of 0.2×10^{34} $cm^{-2}s^{-1}$. This corresponds to 5% of the computing workload of the fully operational experiment by 2008. Over 75 million events were fully generated, simulated using GEANT, reconstructed, reduced to summary data and analysed during the data challenge, resulting in over 0.15PB of useful data stored worldwide.

Grid workload management tools were tested during the pre-challenge production of simulated events (see Figure 2), and continue to be used in the post-challenge distributed analysis exercise. Data management components from LCG and other projects were also used during DC04. In particular, the RLS component was used as both a replica catalogue and metadata catalogue, in conjunction with a GridPP-developed data distribution management system. Data movement and mass storage interfaces were facilitated through a variety of tools, including LCG replica management tools, native SRM and SRB. The LHC Pool Of persistent Objects (POOL) component was used for all event data storage after the GEANT simulation phase, and the accumulated POOL catalogue built up during the challenge is now in use for analysis.

DC04 clearly revealed both strengths and weaknesses of the proposed CMS computing model, and of the Grid computing approach. Specification and development of ad-hoc tools to meet these inevitable scaling issues has already provided further useful insights. In particular, severe scaling issues with several critical middleware components were experienced and possible approaches to overcome these problems identified. Areas of functionality not fully addressed by the available middleware or CMS application portfolio were identified, and tools developed to cover these areas are now being integrated back into the relevant projects. CMS is now looking ahead to the pre-challenge production for DC05, in which it is hoped to exercise the computing system at a yet greater level, and gain further insight into the limitations currently imposed by software, organisation and the computing fabric itself. The results of DC04 will be fully documented in the CMS Computing Technical Design Report, due for publication in early 2005.

3.2 QCDGRID

The UK's national lattice QCD collaboration, UKQCD, currently stores and requires rapid access to around five terabytes of data, a figure that is expected to grow dramatically as the collaboration's purpose built supercomputing system, QCDOC[33], comes into production service towards the end of 2004. This data is stored on QCDgrid[34], a data Grid currently composed of six storage elements at four separate UK sites: Edinburgh, Liverpool, Swansea and RAL.

The QCDgrid software builds on the Globus toolkit[5] (v2.4). The VDT installation[35] has been the most convenient. The toolkit is used for basic Grid operations such as data transfer, security and remote job execution. It also uses the Globus replica location manager to maintain a directory of the whole Grid. The EDG software is used for VO management and security. Custom written QCDgrid software is built on Globus to implement various QCDgrid client tools and the control thread. It is open source and is available at the NeSC source forge site[36]. The system has a central control thread running on one of the storage elements which constantly scans the Grid, ensuring that all files are stored in at least N suitable locations where N is at least two. This provides security and convenience of access and works very well.

For ease of data management, metadata is in the form of XML documents and is stored in an XML Database - UKQCD uses eXist, which is an open source database that can be searched using the XPath query language. The collaboration has developed what is now an internationally accepted schema (QCDML) for validation and manipulation of the metadata and has deployed associated browsers.

The job submission component of QCDgrid is also built on the Globus toolkit. It allows data generation and analysis jobs to be submitted to Grid machines. It has been successfully tested on a system containing mixed architectures (SUN Solaris and Linux Enterprise/Fedora). It is integrated with the existing data Grid and has the expected job monitoring and output retrieval features. It is planned to use this system to process jobs using QCDOC and a variety of machines and clusters in UK sites. It can inter-operate with other Grid systems which use Globus but, not necessarily, the QCDgrid software. In future developments we will cooperate with the ILDG[37] 'Grid-of-Grids' partnership that UKQCD initiated under the GridPP project [38]. The protocols will utilise web services.

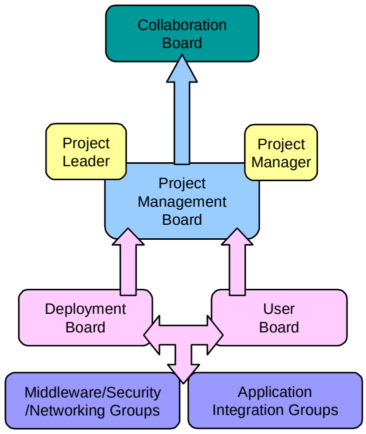
3.3 Biomedical applications[39]

The biomedical work package of EDG focused on the deployment of biomedical applications on the Grid testbed. Goals of this work were threefold: to demonstrate the relevance of Grids for life science; to test the EDG middleware and feedback requirements to the middleware developers; and to raise awareness of the impact of Grids in the life science community.

Several applications were developed and deployed. The applications covered PROTEUS - for linking gene expression to gene promoter analysis (currently being converted into a web-based service); 3D structuring of proteins, which was not deployed on the early testbed as MPI was not available; Grid Protein Sequence Analysis (GPS@) - a Grid-aware web server for protein sequence analysis[40] which has to deal with very large data sets; Phylojava, - a Grid-based application to speed up the calculation of phylogetic trees [41] ; and Parasitology – an environment for sharing large sets of data (with associated knowledge) between laboratories. Of these 5 applications, 4 were deployed successfully and showed promising results (jobs being run across many European sites) when compared to existing methods, though it was clear that additional services would need to be developed to make them more attractive to researchers.

Other biomedical areas were investigated for their practicality on the Grid middleware. Medical data management was one area that could not be implemented due to the lack of fine-grained access control on the Storage Elements of the application testbed at the time (see Storage section). This prevented the use of sensitive medical data on the Grid. Mammography analysis tests revealed that the middleware overhead is prohibitive for applications with a large number of short jobs where a compromise needs to be made between application splitting (parallelisation) and the number of jobs to be handled. However, Monte-Carlo simulations for PET/Single Photon Emission Computed Tomography (SPECT) using the GEANT Application for Tomographic Emission (GATE) ware very successful in the Grid environment.

Several important lessons were learned from the biomedical applications work. Firstly, it is essential to have dialogue between application areas – as this allows identification of common and application specific requirements. Secondly, application independent developers who can act as an interface between application developers and middleware groups are extremely useful. Thirdly, researchers are soon discouraged if they have to spend too much time testing their applications in an unstable computing environment.



4 The future – toward a production Grid

The PPARC funded GridPP-2 project will build on the successes of GridPP. The principal goal is to move from an era where middleware development and testbed construction has been the focus, to an era where building a production quality Grid running production quality services to meet the LHC challenge is the priority. The GridPP-2 project started on 1 September 2004.

Figure 3: GridPP-2 management structure

4.1 Management Structure

The overall management structure of GridPP-2 closely resembles that of GridPP and is shown diagrammatically in Figure 3. Central to this structure is the Project Management Board which monitors and coordinates all aspects of the project to ensure that all project objectives are being met. The User Board has the primary responsibility for seeing that the GridPP project remains science driven and that the project develops to meet the needs of the UK Particle Physics community. The User Board can be seen as generating the requirements and providing feedback for the Deployment Board which is charged with the roll-out of an operational Grid service to a level of stability and ease of use consistent with a production facility.

4.2 Technical Evolution of Middleware

The very limitations which meant that the prototype Grid could be successfully deployed on a relatively short timescale, now become the principle issues for the next few years. These are (i) removing the dependence on the limited Linux deployment platform arising due to the complex and nature of the middleware prototypes and particle physics analysis and simulation jobs (ii) the need for true robust production quality code (iii) the need for scalable and businesslike operational procedures.

In order to ensure a robust and stable infrastructure, this middleware work will be undertaken in close collaboration with EGEE (which has stated its intention to ensure other platforms can be supported.), and the Open Middleware Infrastructure Institute[42] (OMII).

5 Conclusion

During the last three years, the GridPP project has been at the forefront of developing pioneering Grid middleware, and the deployment of a usable Grid testbed across Europe, principally with the European Data Grid project. This work provided valuable lessons in terms of reliability, scalability and expectation management. The next few years will see Grids for science becoming the norm for on-demand computing. The lessons learned in GridPP-1 will be used in GridPP-2 to concentrate upon deployment of robust and stable code, such that application areas can rely upon use of the infrastructure for major data challenges in the run up to the start of the LHC programme in 2007. We will work particularly closely with the LCG and EGEE projects during this period.

References

[1] http://www.gridpp.ac.uk/
[2] http://eu-datagrid.web.cern.ch/eu-datagrid/
[3] http://lcg.web.cern.ch/LCG/
[4] http://public.eu-egee.org/
[5] The Globus Toolkit http://www.globus.org/
[6] http://goc.grid-support.ac.uk/gppmonWorld/gppmon_maps/lcg2.html
[7] A full description of the DataGrid Workload management system can be found in "The DataGrid Workload Management System: Challenges and Results" *Journal of Grid Computing*. Submitted
[8] Raman, R., M. Livny and M. Solomon. "Matchmaking: Distributed Resource Management for High Throughput Computing". In: *Proceedings of the seventh IEEE International Symposium on High Performance Distributed Computing (HPDC7*), Chicago,IL, 1998.
[9] The estimate relied on the information being published to the LCG information service on 25th June 2004 and should only be used as a guide.
[10] Frey, J., T, Tannebaum, I. Foster, M. Livny, and S. Tuecke. "CondorG: A ComputationManagement Agent for Multi-Institutional Grids". In: *Proceedings of the tenth IEEE International Symposium on High Performance Distributed Computing (HPDC10)*, San Francisco, California, 2001.
[11] L. Matyska, A. Krenek, M. Ruda, M. Vocu, Z. Salvet, J. Sitera, J. Pospisil and D. Kouril. 'The Grid Job Monitoring Service.' In: *Terena Networking Conference*, 2002.
[12] Ludek Matyska et al., "Distributed Tracking, Storage, and Re-use of Job State Information on the Grid". *CHEP 2004*, Interlaken, Switzerland. Accepted.
[13] http://www.ggf.org/
[14] http://www.r-gma.org
[15] A. Cooke, A.Gray, L. Ma, W. Nutt, J. Magowan, P. Taylor, R. Byrom, L. Field, S. Hicks, J. Leake, M. Soni, A. Wilson, R. Cordenonsi, L. Cornwall, A. Djaoui, S.M. Fisher, N. Podhorszki, B. Coghlan, S. Kenny, D. O'Callaghan. "R-GMA: An Information Integration System for Grid Monitoring". In: *Proceedings of the Tenth International Conference on Cooperative Information Systems*, 2003
[16] Rob Byrom, Brian Coghlan, Andy Cooke, Roney Cordenonsi, Linda Cornwall, Martin Craig, Abdeslem Djaoui, Steve Fisher, Alasdair Gray, Steve Hicks, Stuart Kenny, Jason Leake, Oliver Lyttleton, James Magowan, Robin Middleton, Werner Nutt, David O'Callaghan, Norbert Podhorszki, Paul Taylor, John Walk and Antony Wilson. "Production Services For Information and Monitoring in the Grid". *Proceedings of AHM 2004*.
[17] "Internet X.509 Public Key Infrastructure Certificate and CRL Profile". *IETF RFC 2459*
[18]"EU DataGrid: Final Security Report" (D7.7) http://edms.cern.ch/document/414762
[19] "HTTP Over TLS". *IETF RFC 2818*
[20] http://www.nesc.ac.uk/events/ahm2003/AHMCD/pdf/085.pdf
[21] https://edms.cern.ch/document/422510/3
[22] D. Cameron *et. al.* "Replica Management Services in the European DataGrid Project". *Proceedings of AHM 2004*.
[23] http://castor.web.cern.ch/castor/
[24] https://edms.cern.ch/document/474424
[25] https://forge.gridforum.org/projects/gsm-wg/
[26] http://www.gridpp.ac.uk/gacl/
[27] http://edg-wp2.web.cern.ch/edg-wp2/spitfire/index.html
[28] http://www.cs.man.ac.uk/grid-db/
[29] http://www.gridpp.ac.uk/datamanagement/metadata
[30] http://edg-wp2.web.cern.ch/edg-wp2/optimization/optorsim.html
[31] http://cmsinfo.cern.ch/
[32] http://www.uscms.org/s&c/dc04/
[33] QCDOC, a purpose built massively parallel computer for lattice QCD calculations.See: http://www.ph.ed.ac.uk/ukqcd/community/qcdoc/
[34] QCDgrid, A grid Resource for QCD. See: http://www.epcc.ed.ac.uk/computing/research_activities/grid/QCDgrid/
[35] Virtual Data Toolkit, see http://www.cs.wisc.edu/vdt/
[36]The QCDgrid open source software project. See http://forge.nesc.ac.uk/projects/QCDgrid/
[37] International Lattice Data Grid, see http://www.lqcd.org/ildg
[38] A.C. Irving, R.D. Kenway, C.M. Maynard and T. Yoshie, *Nucl Phys B (Proc Suppl)* 129&130 (2004) 159-163.

[39] Based on the final report of the Grid-aware Biomedical Applications group (EDG WP10): https://edms.cern.ch/document/420546/2.0/

[40] http://gpsa.ibcp.fr/

[41] http://pbil.univ-lyon1.fr/phylojava

[42] http://www.omii.ac.uk/