

Towards Dynamic Data Grid Framework for eResearch

A.B.M. Russel and Asad I. Khan

Monash University

{abm.russel@its, asad.khan@infotech}monash.edu.au

Abstract

The scale at which scientific data is produced will undergo a massive change in the near future. Many sophisticated scientific discovery laboratories or the installation of sensor networks would produce a large amount of data. Research in protein crystallography for instance can produce hundreds of Terabytes of data from a single crystallography beamline. These data have to be saved for future use and made available for collaborative use by researchers. There is a need to develop a framework which can deal with storing such data volumes. This framework should also handle disparate data sources tightly integrated with the users' applications and large data streams arising from instruments and sensors. This paper presents an initial study into the framework for servicing large dynamic data sets over a national grid for eResearch.

Keywords: Data Grid, eResearch, GridFTP, SRB, CIMA.

1 Introduction

Hey and Trefethen [19] identified streaming instruments and sensor as major contributors towards what they label as the data deluge. Lately this view has also been accepted by the UK Joint Information Systems Committee (JISC) [22]. Many sophisticated scientific discovery laboratories such as the synchrotron or long term environmental research (LTER) [26] sensor networks e.g. a coral reef monitoring network would produce a large amount of very valuable data. Such data are dynamic in nature and are collected from scientific instruments such as sensors, astronomical telescopes, spectrometers, and particle accelerators. Research in structural biology, particularly relating to protein crystallography alone can produce about 0.5 Terabyte/day from a single synchrotron beamline with a Petabyte of storage being required for data curation over a few years period. In this scenario a single and local file system will not have the capability to handle such large data volumes. The evolution of Distributed computing solves the limitation of single data storage by sharing data over network from distributed data storage. However the problem of how to deal with diverse data

sources and how to integrate distributed data storages must be addressed first. These issues can be best resolved through the use of Grid based technologies. The Grid [11, 12] is a hardware and software infrastructure that provides access to data, computational, and communication resources distributed over wide area environments. The Grid technology can provide us enormous computing power from a collection of heterogeneous distributed computing resources (Compute Grid) and massive data storage capabilities by integrating distributed data storage resources (Data Grid) [23]. Nevertheless, the problem of how to deal with very large data (Petabyte size or more) and how to make such large datasets available over the large network in Grid still remains as an open issue. Thus there is a need to develop a dynamically scalable (data grid) framework which can deal with storing and transporting such large raw or processed data volumes as well as making these raw or processed data available online for collaborative research.

2 Background

The consequence of not planning and provisioning for these large data streams would entail data losses of unprecedented proportions. To lessen the data loss, we identify three different data storage requirements for supporting eResearch:

a. Static data storage:

This is the data type which has been refined as part of the scholarly publication process and contains the data sets and the applications which made use of these data sets for the publication.

b. Dynamic raw data storage:

This type of data could be generated from instruments and sensors. This can also be obtained from any computer simulations. The data are often needed to be refined using different data curation mechanisms. It also requires vast amount of storage along with a considerable amount of computing power to process such large volumes of data for permanent storage.

c. Fused data storage:

This type of data combines inputs from the several data sources in order to make it available to the researchers. Data fusion is often performed in several stages. The process requires large amounts of storage and a considerable amount of computing power.

For all these cases a dynamic data framework is required which would provide scalability and flexibility while

implementing highly reliable and cost effective solutions for storing large volumes of data. The prime aim of such a framework should be able to handle transient demands for storing and transporting large data volumes over a national Grid.

3 Towards Data Grid Infrastructure

To meet the present needs of eResearch an open standard based solution would be more appropriate. In this context, some large scale deployed global Data Grids projects are European DataGrid [10], PPDG [31] and BIRN [7]. In the following subsection we first identify the key components for building a Grid infrastructure for dealing dynamic data streams from various eLab apparatuses such the synchrotron beamlines and the LTER networks [26].

3.1 Computing Components

This section discusses different computational toolkits which can be used to establish a dynamic data Grid framework for processing large volume of data.

3.1.1 The Globus Toolkit

Globus Toolkit (GTK) [15] has become the de facto standard for building Grid infrastructures and applications. One of the key features of GTK is the security infrastructure using Globus Security Infrastructure (GSI) [14]. It uses PKI based X509 certificates to identify participating software services and people and establishes trust relationship among the participants using Gridmap-file. Another key feature of GTK is GridFTP. GridFTP provides data movement facilities via a GSI enabled FTP service. GridFTP also provides replica location services via the Globus Replica Catalog. The computing resources for the running job can be monitored using the Monitoring and Discovery Service (MDS). MDS is also GSI-enabled and offers an indexing service to access information across multiple information servers. The Globus Resource Allocation and Management (GRAM) protocol provides resource management facilities. The GRAM server, known as gatekeeper, offers simple interfaces to manage jobs running on underlying batch systems like Portable Batch System (PBS) [30], Load Sharing Facility [25], and Sun Grid Engine (SGE) [36]. Unlike many Grid middleware available at present notably Legion [17], Condor [9], Unicore [4], and Gridbus [16], GTK provide four major pillars including GSI, GridFTP, MDS & GRAM for deploying large scale Grid infrastructure

3.1.2 The Sun N1 Grid Engine

The Sun N1 Grid Engine is a complete Distributed Resource Manager (DRM) solution from Sun Microsystems. This can provide policy-based workload management and dynamic provisioning of application workloads by creating a grid of network-connected servers, workstations, and desktop. Here, a user submits a job and declares a requirements profile for the job. When a job queue is ready for a new job, the resource manager determines suitable jobs for that queue and then

dispatches the job with the highest priority or longest waiting time; it starts new jobs on the most suitable or least loaded queue. The computing tasks are also distributed across the grid in accordance with resource requirements for the job, user requests for the job, and administrative policies. The Sun N1 Grid Engine version 6 is used to create grids at enterprise level with the following features [36]:

- Flexible, extensible resource management capabilities
- Supports hierarchical policies
- Detailed statistics gathering for analysis and utility billing
- Sophisticated scheduler with advanced planning abilities
- Industry-standard APIs and Globus Toolkit integration

3.2 Security

This section discusses different security mechanisms that could be adopted in the proposed framework for reliability.

3.2.1 PKI

Public Key Infrastructure (PKI) is used in development of Grid infrastructure. PKI reduces the online dependency on the source of authenticity i.e. the Certificate Authority (CA). The identity of a user is ascertained by proving that they hold the private key of a pair of keys. The public part is contained within an X.509 [20] certificate which is verified by the other party. Only the private key is able to decrypt messages encrypted with the public key and vice versa. But PKI does not provide delegation and single sign-on facility.

3.2.2 Kerberos

Kerberos [24] is a network authentication protocol. It is designed to provide strong authentication for client/server applications by using secret-key cryptography. Kerberos uses symmetric encryption and a trusted third party Key Distribution Center (KDC) to authenticate users to a suite of network services. After usual login to the own machine, a user can send a request and principal using *kinit* to the KDC for a Ticket-granting Ticket (TGT) from the Authentication Server (AS). The KDC checks for the principal in its database. If the principal is found, the KDC creates a TGT, which is encrypted using the user's key and returned to that user. The TGT is set to expire after a certain period of time and stored in the client machine's credentials cache. Whenever the user needs access to a network service, the client software uses the TGT to request a new ticket for that specific service from the Ticket-granting Server (TGS). The service ticket is then used to authenticate the user to that service transparently. Kerberos requires a global naming convention and considerable server interaction.

3.2.3 GSI

The Grid Security Infrastructure (GSI) [40] overcomes the problem of PKI and Kerberos. GSI uses X.509-based Proxy Certificates [38] to allow a process to use the certificate rather than the user principals. The process then can read the private key of the GSI proxy credential. Keys corresponding to X.509 certificates are encrypted and only unlocked with a passphrase. Keys for GSI proxy certificates usually have short lifetime as these are stored unencrypted with only file permissions protection. The client side can use a GSI proxy credential to communicate over SSL [13] with a server. Besides, a credential storage server MyProxy [28] can be used to store GSI credentials and delegating GSI credentials to authorised processes on the Internet. Similar to MyProxy, the Community Authorization Service (CAS) of Globus Toolkit [15] provides resource provider an ultimate control on their distributed resources by specifying course-grained access control policies. But currently there is no support for CAS-based authorization for web services. This can be overcome by building an authorization module that extracts and parse the Security Assertion Markup Language (SAML) [34] assertion stored in a X.509 certificate generated by CAS. The extracted signature on the assertion needs to be verified to trust the CAS certificate. Shibboleth [35] is open source software that uses SAML as a transport mechanism for controlling access to shared resources and web available services in Grid. When a user requests a resource from a resource provider, Shibboleth sends attributes about the user to the resource provider before making the user log in to the resource provider's site. This allows the resource provider's site to decide whether or not to grant access to the user.

3.3 Storage Components

This section discusses different storage components that can be used to establish a dynamic data Grid framework for storing large volume of data.

3.3.1 SAN

Storage Area Network (SAN) can provide a great storage solution in a dynamic data grid framework by forming a network-based storage. SAN can physically decouple the directly attached storage from other components of the computer. Thus physical disks can be located as a separate resource anywhere on the network. SAN allows multiple paths for data access within this distributed disk resource. The data bandwidth also scales-up in-step with increases in the number of storage systems on the network thus creating highly scalable data storages.

3.3.2 Network Area Storages

Network File System (NFS) [27] is a network protocol for taking local file system requests and redirecting them to another server. NFS does not include an actual file system and relies on local file systems at the server to store data. On the other hand Andrew File System (AFS) [29] expands the client/server file system concept

to a fully distributed file systems by enhancing security and scalability. AFS uses Kerberos [24] authentication to ensure that both the service requester and the service provider have to prove their identities before a service is granted. Like AFS, Distributed file systems provide a uniform global name space, hence the files appear under the same name on all clients and all portions of the name space can be accessed using standard file system methods. However, existing distributed file systems require servers to access file data on behalf of clients to ship that data to the client. A high performance alternative to distributed file systems is a Parallel file system. Parallel file systems provide the same file system name space to a cluster of computers. These file systems rely on the high-speed communication facilities but do not adapt well to low speed networks. To overcome the bottleneck of a centralised file server, within parallel files system, a Serverless file systems could be used. Performance for serverless file systems require all computers in the file system to share a secure network which is challenging in wide area distributed environment like Grid. The Hierarchical File Systems can resolve the scalability issue by using layered approach to make many computers appear to share the same storage subsystem. Thus, a GSI enabled hierarchical file systems is required for storing large volume of data in the dynamic data Grid.

3.4 Storage Resource Broker

3.4.1 The SRB

To provide a uniform access to distributed storage across a range of storage devices, the Storage Resource Broker (SRB) [33, 34] can be used. The SRB supports file replication as either offline or on-the-fly. It supports metadata associated with a distributed file system, such as location, size and creation date information. In contrast with traditional network file systems, SRB is attractive for Grid applications as it deals with large volumes of data, which can transcend individual storage devices. It also deals with metadata specific to the content.

3.5 Data Transfer

This section discusses data transfer mechanism to be used in the proposed data Grid framework.

3.5.1 GridFTP

The GridFTP [2] provides several features for Grid environments, such as security, parallel & partial file transfers and third party transfers. Hence it is ideally suited for large data transfers using a secure protocol in Grid. GridFTP is also integrated with SRB for increasing data transfer speed by parallelising data inputs/outputs (I/Os) from the storage and can increase service reliability and efficiency by spreading these (parallel) I/Os among different types of devices e.g. disks and nearline storages being managed by the storage resource middleware.

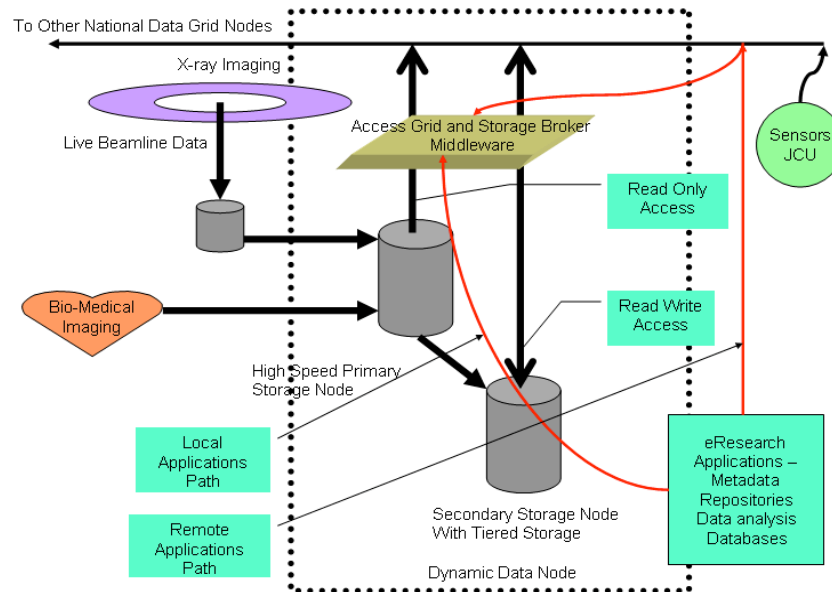


Figure 2: An overview of the dynamic data grid framework in context with digital data sources, applications, and network.

4.1.2 Data Curation

The proposed framework will provide digital data curation i.e. longevity of stored information by providing a very scalable system and economical means to store digital data produced from sensors and instruments in a highly reliable manner. A dynamic data storage Grid automatically entails data longevity owing to its ability to preserve digital outputs from research indefinitely within this framework through its core features – highly distributed architecture, very large storage capacities, offsite data mirroring, backups, and archival. Changes made to the stored data can be persistently tracked by the system owing to the high level of integration which exists in the framework.

4.1.3 Data Fusion

Gathering information from heterogenous sources is challenging as differences in data format may require advanced pre-processing and post-processing techniques to enable accurate and reliable fusion of data from different sources. Data fusion [39] is a process of combining data originating from different sources to maximize useful information. This framework will also be able to integrate different data fusion methodologies at different levels. Examples of different levels [37] of data fusion include:

- Raw data level fusion robustly merges the raw data from the source.
- Feature level fusion extracts the features from each source and registers the detected features.
- High-level data fusion combines a priori knowledge with or without pre-processing of data. These data could be based on local observations and/or the information communicated from neighbouring nodes.

4.1.4 Data Management

Data management is essential for data intensive applications. The framework will allow authorised user to manipulate stored data e.g. listing, upload and download, copying, deleting, updating and renaming.

4.1.5 Interoperability and Scalability

The high scalability and on-demand resource requirements of dynamic data could be met through the combination of GridFTP and a set of suitable storage middleware. The Globus based access (i.e. the GridFTP) ensures that an integration path with evolving standards such as the web services would remain open in future. At present the Web Service Resource Framework (WSRF) [3] is emerging as the preferred way to extend Web services to support common patterns e.g. lifetime management, state management and notification that arise in grid computing.

4.1.6 Data Caching

Many data intensive applications are not able to talk to each other owing to being developed or generated in separate silos. This requires us to write specific middleware in most cases. Data are however essentially distributed, hence if we lose one source we can recreate the lost information by getting portions of data from other sources to carry on. Instances where SAN mirroring is not feasible, the distributed nature of the data grid would be of value, particularly with similar grid nodes caching frequently accessed data within their secondary storage repositories. The SRB architecture presents a distinct advantage in making disparate applications and data repositories to talk to one another in a standard manner. Thus, SRB's multi-faceted architecture would be used to strengthen overall service reliability in addition to its storage middleware functions for the secondary data storage.

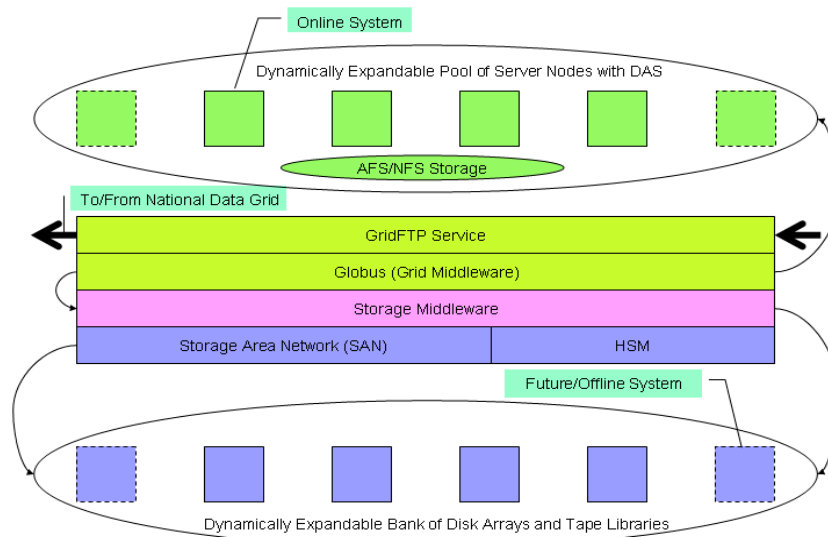


Figure 3: A layered architecture of dynamic data Grid framework.

4.2 Framework Architecture

Figure 3 shows the layered architecture for the dynamic data Grid framework. The key requirements of this framework is to develop standard user interfaces and provide “light” interfaces where necessary to applications requiring particular data format translations. The technology should allow client organisations to access data services within the agreed parameters e.g. of uptime, cost, and response time, for their operations. The main characteristics of the dynamic data approach include:

- The framework is able to horizontally scale-up, adding more resources on the network as the demand increases.
- The framework is able to move its operations anywhere on the network, transparently to the user, for optimal performance.

This framework will also store raw data produced from local sources such as synchrotron [5] as well as distant sources such as LTER sensors [26]. The data storage will support the requirements of scientific research communities, metadata repositories, and virtual laboratories. The data requirements of metadata tools/repositories and semantic grid will also be met through this framework. The framework will store raw data but also include metadata collected at the data source e.g. the information included within the file system. Other forms of data e.g. required for generating more sophisticated metadata, semantic web ontologies, and curation will also be made available.

This framework will also store very high data volumes, e.g. in Petabytes by contemporary standards, within the system very economically and to move large proportions of it over the network in a timely manner. Such challenges are handled by the SAN. The SAN has mirrored storage, as well as redundant network configurations. In the event of one of the network switches, or one of the disks failing, there is no loss of service as the SAN can automatically fail-over to a

redundant switch or disk, without any intervention from the operator. SANs only work over a short distance due to their high speed transfer capabilities. Hence SAN must be located close to the major data sources. The principle of data locality suggests that vast amounts of data must be kept physically close to the data sources and the applications that make use of it – send the results and not the data – paradigm. However while pursuing the data locality consideration the ability to move data where it is most needed at a given time needs to be considered. This is best addressed by applying a dynamic data approach.

4.3 Framework Implementation

Two major storage nodes are considered (see Figure 2) while implementing the dynamic data Grid framework. The primary node stores data streams in real-time and the secondary node provides full access to the storage of historical data obtained from various sensors for data analysis. The following key features are considered while implementing the dynamic data grid framework:

- Capable of ultimately storing and managing Petabytes of distributed data within the parameters set by the data producers and data consumer organisations/entities.
- Capable of integrating scientific instruments and sensors into a grid computing environment with Web Services interfaces using CIMA interfaces.
- Capable of handling a large number of concurrent data access/management requests.
- Supports upwards integration with higher level data manipulation, linking, and metadata services requirements for data curation, virtual laboratories and telepresence applications and metadata and semantic grid applications.
- Supports downwards integrations with various types of storage systems and allows a very high level of horizontal scalability within the network. Heterogenous data stores, e.g. SAN, Network File System/Andrew File System (NFS/AFS) nodes, hierarchical storage management (HSM), and

redundant array of independent disk (RAID) can all be transparently accessed and managed.

- f. Supports highly scalable network architecture for storing and moving large volumes of data to and from the system very efficiently.
- g. Provides a production class service.

5 Applications

The data streams arising from the synchrotron can be made available in real-time through the primary storage node. The secondary storage node would provide full access to the storage of historical data obtained from various sensors such as the synchrotron. Apart from synchrotron, examples of use would include a coral reef data and information management system of the National Oceanographic Data Center (NODC) that integrates biological data from reef ecosystem monitoring and research programs with interrelated chemical, biological and physical data of surrounding marine areas. Other projects which can make use of the framework would include Nanostructural Analysis Network Organisation (NANO), Gemini Observatory and Square Kilometre Array project, Australian Proteome Analysis Facility, Arafura–Timor Research Facility, and OPAL.

6 Experimental Results

A Grid environment has been established at Monash University Information Technology Services consisting of 32 grid nodes along with two head nodes, two authentication nodes and two storage nodes. Each node is AMD Opteron based 64 bits architecture with dual processors. Each node is having Scientific Linux OS (2.6 kernel) with processing speed of 2.4GHz and 4GB RAM. On each node Globus Toolkit 4 has been deployed as grid middleware. For grid resource management Sun N1 Grid Engine has been installed. All the 32 nodes and head nodes are also running GridFTP sever. We start our experiment with identifying available bandwidth among the head node and compute nodes. We initially took Iperf [21] measurements followed by GridFTP measurements for varied streams to identify the significance of parallel streams with fixed TCP window size. We examined from no parallel (e.g. 1) to 256 parallel streams and found no significance difference (95%-100% similarities) among them. The difference is only with the overheads of authentication, protocol itself & performance reporting back to the requester.

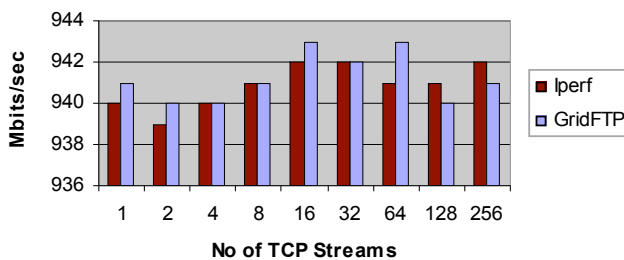


Figure 4: Performance comparison between Iperf [21] & GridFTP [2] with varied parallel streams.

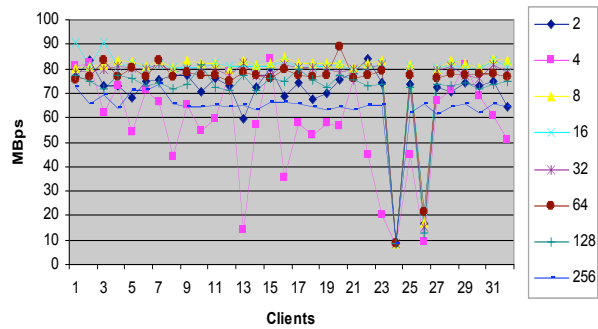


Figure 5: GridFTP performance comparison with 32 client nodes with varied parallel streams.

Next we extend our data transfer experiment for large dataset size of 1GB. For our experiment we transfer 256GB in total. Initially we calculate bottleneck bandwidth using Iperf and RTT using ping. Using these two values we measure TCP window size. We perform our experiments with varied parallel streams (with the interval of 2^x , where values of x ranges from 1 to 8) for a single client node. We observe that with the increase of parallel streams the GridFTP bandwidth also increases. We perform further experiments for 2 to 32 different client nodes and found that for bulk data transfers, between 8 to 64 parallel streams the performance increases and after 64 parallel streams it does not provide significant improvement as shown in Figure 5.

7 Conclusion

Dynamic data grids are envisaged as one of the essential building blocks for the 21st century eResearch infrastructure, which will ultimately allow advanced scientific research and collaboration to be conducted using as simple an apparatus as a handheld mobile computer. This paper presents an initial study into the framework for servicing large dynamic data sets over a national grid for eResearch. Our study into the framework would hopefully provide a road map for developing such framework within higher education and research organisation in Australia.

8 Acknowledgements

This work was partly sponsored through 100% equipment grant from Alphawest and Sunmicrosystems under their Sun Fire 2004 Grid Computing Program.

9 References

- [1] Australia's Academic and Research Network. Available online: <http://www.aarnet.edu.au/engineering/aarnet3/>
- [2] Allcock, W., Bester, J., Bresnahan, J., Chervenak, A., Liming, L., Meder, S., Tuecke, S., "GridFTP protocol specification", Technical report, Global Grid Forum GridFTP Working Group, September 2002.
- [3] Atkinson, M., "Grid & Web services: Web Service Resource Framework (WSRF), Presentation at Core e-Science Programme Town Meeting to Discuss Grid & Web Services, Friday, 23rd April 2004, Jolly St Ermins Hotel, London.

- [4] Almond, J. and Snelling, D., "UNICORE: uniform access to supercomputing as an element of electronic commerce", *Future Generation Computer Systems*, 15, pp. 539-548, NH-Elsevier, 1999.
- [5] Australian Synchrotron.
Available online: <http://www.synchrotron.vic.gov.au/>
- [6] Bramley, R., Chiu, K., Huffman, J.C., Huffman, K. and McMullen, D.F., "Instruments and Sensors as Network Services: Making Instruments First Class Members of the Grid, Indiana University CS Department Technical Report 588, 2003.
- [7] The Biomedical Informatics Research Network. Available online: <http://www.nbimr.net>
- [8] Buyya, R., "The Virtual Laboratory Project: Molecular Modeling for Drug Design on Grid", *IEEE Distributed Systems Online*, Vol. 2, No. 5, 2001.
- [9] Condor.
Available online: <http://www.cs.wisc.edu/condor/>
- [10] European DataGrid,
Available online: <http://eu-datagrid.web.cern.ch/eu-datagrid/>
- [11] Foster, I., Kesselman, C., *The Grid: Blueprint for a New Computing Infrastructure*, First Edition, Morgan Kaufmann Publisher, 1999.
- [12] Foster, I., Kesselman, C., *The Grid – Blueprint for a New Computing Infrastructure*, Second Edition, Morgan Kaufman Publisher, 2004.
- [13] Freier, A.O. Karlton, P. Kocher, P.C. "The SSL Protocol Version 3.0 Transport Layer Security Working Group INTERNET-DRAFT", 18 November 1996.
Available online: <http://wp.netscape.com/eng/ssl3/>
- [14] Foster, I., Kesselman, C., Tsudik, G., and Tuecke, S., "A Security Architecture for Computational Grid", 5th ACM conference on Computer and Communications Security, New York, 1998, pp. 83-91.
- [15] Globus Project (The Globus Alliance). Available online: <http://www.globus.org>
- [16] GRID computing and BUSiness technologies (GRIDBUS). Available online: <http://www.gridbus.org/>
- [17] Grimshaw, A., Wulf, W. et al., "The Legion Vision of a Worldwide Virtual Computer". *Communications of the ACM*, vol. 40(1), January 1997.
- [18] Grid and Next Generation Network.
Available online: <http://www.granget.net/>
- [19] Hey, T., Trefethen, A.E., "The UK e-Science Core Programme and the Grid," *Future Generation Computer Systems*, vol. 18, pp. 1017-1031, 2002.
- [20] Housley, R. Ford, W. Polk, W. Solo, D. , "RFC 2459 Internet X.509 Public Key Infrastructure Certificate and CRL Profile", January 1999.
- [21] Iperf. Available online:
<http://dast.nlanr.net/Project/Iperf>
- [22] The Joint Information Systems Committee, 2004. "The Data Deluge: Preparing for the explosion in data", Available online: http://www.jisc.ac.uk/index.cfm?name=pub_datadeluge
- [23] Johanstona, W.E., "Computational and Data Grids in Large-Scale Science & Engineering", *Future Generation Computer Systems*, Vol. 18, No. 8, October, 2002.
- [24] Kerberos: The Network Authentication Protocol.
Available online: <http://web.mit.edu/kerberos/www/>
- [25] Load Sharing Facility, Platform Computing Inc. Available online: <http://www.platform.com/products/LSF/>
- [26] The Long Term Ecological Research (LTER) Network.
Available online: <http://www.lternet.edu/>
- [27] Networked File Systems (NFS), RFC 1094 (Version 2), RFC 1814 (Version 3), RFC 3010 (Version 4)
- [28] Novotny, J. Tuecke, S. and Welch, V. "An Online Credential Repository for the Grid: MyProxy". *Proceedings of the Tenth International Symposium on High Performance Distributed Computing (HPDC-10)*, IEEE Press, August 2001.
- [29] OpenAFS UserGuide. Available online:
<http://www.openafs.org/pages/doc/UserGuide/auusg010.htm>
- [30] Portable Batch System, Altair Grid Technologies
Online: <http://www.openpbs.org/>
- [31] Particle Physics data Grid. Available online:
<http://www.ppdg.net>
- [32] Rajasekar, A.K. and Moore, R.W. "Data and Metadata Collections for Scientific Applications", *European High Performance Computing conference*, Amsterdam, Holland, June 26, 2001.
- [33] Rajasekar, A., Wan, M. Moore, R. "MySRB & SRB - Components of a Data Grid", *The 11th International Symposium on High Performance Distributed Computing (HPDC-11)* Edinburgh, Scotland, July 2002.
- [34] Security Assertion Markup Language Technical Overview.
Available online: <http://xml.coverpages.org/SAML-TechOverviewV20-Draft7874.pdf>
- [35] Shibboleth. Available online:
<http://shibboleth.internet2.edu/shib-tech-intro.html>
- [36] Sun N1 Grid Engine 6, Sun Microsystems Inc.
Online: <http://www.sun.com/software/gridware/index.xml>
- [37] Sun, H, et al, 1994, "Study on an algorithm of multisensor data fusion", *Proc. IEEE Proceedings of the National Aerospace and Electronics Conference*, Vol.1, pp.239-245.
- [38] Tuecke, S. Welch, V. Engert, D. Pearlman, L. Thompson, M. "RFC3820 Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile", Jun 2004.
- [39] Wald, L., "Some terms of reference in data fusion". *IEEE Transactions on Geosciences and Remote Sensing*, 37, 3, 1190-1193, 1999.
- [40] Welch, V. Foster, I. Kesselman, C. Mulmo, O. Pearlman, L. Tuecke, S. Gawor, J. Meder, S. Siebenlist, F. "X.509 Proxy Certificates for Dynamic Delegation". *3rd Annual PKI R&D Workshop*, 2004.