WebGRelC: Towards Ubiquitous Grid Data Management Services

Giovanni Aloisio, Massimo Cafaro, Sandro Fiore, Maria Mirto
Center for Advanced Computational Technologies, University of Lecce, Italy
Center for Euro Mediterranean Climate Changes, Italy
{giovanni.aloisio, massimo.cafaro, sandro.fiore, maria.mirto}@unile.it

Abstract
Nowadays, data grid management systems are becoming increasingly important in the context of the recently adopted service oriented science paradigm. The Grid Relational Catalog (GRelC) project is working towards an integrated, comprehensive data grid management solution. This paper describes WebGRelC, which is a dedicated grid portal allowing data handling, publishing, discovery, sharing and organization, and its underlying data grid services.

1. INTRODUCTION

As pointed out by Ian Foster, we have in the last few years moved towards a new, service oriented science [1] in which software is envisioned as services, and services as platforms. Increasingly, services besides computation rich are also data rich, producing a huge amount of data distributed across multiple data servers. There is a growing need for a grid infrastructure allowing scientific communities sharing data securely, efficiently and transparently. Datasets once created need to be visualized, published, downloaded, annotated etc. Discovery mechanisms, such as searchable metadata directories must be provided to find relevant data collections. Integration and federation services need to cope with independently managed legacy datasets, to infer new knowledge from existing distributed data. Although fundamental building blocks such as distributed file systems and semantic storage already exist, data grid management systems are still in the pioneering phase. The main challenge is to design and implement reliable storage, search and transfer capabilities of numerous and/or large files over geographically dispersed heterogeneous platforms.
The Grid Relational Catalog (GRelC) project [2], a data grid research project developed at the Center for Advanced Computational Technologies (CACT) at the University of Lecce, is working towards an integrated, comprehensive data grid management solution. It provides, besides traditional command line and graphical interfaces, a dedicated grid portal allowing data handling, publishing, discovery, sharing and organization. Grid portals are web gateways to grid resources, tools and data. They hide the underlying grid technologies and provide advanced problem solving capabilities to solve modern, large scale scientific and engineering problems. This paper describes WebGRelC, which is the GRelC grid portal, and its underlying data grid services.
The outline of the paper is as follows. In Section 2, we present the GRelC data grid services underlying the portal, whereas in Section 3 we describe the portal architecture. In Section 4, we discuss the current implementation, technologies and related issues, whereas in Section 5 we present a use case related to the use of the portal for a bioinformatics experiment, whilst Section 6 recalls related work. We draw our conclusions in Section 7.

2. GRELC DATA GRID SERVICES

The main goal of the GRelC project is to provide a set of data grid services to access, manage and integrate data (i.e. databases and files) in grid environments. GRelC data grid services already implemented are: data access (DAS), data storage (DSS) and data gather (DGS) service.
The *Data Access Service* (DAS) has been designed to provide a uniform, standard interface to relational and not-relational (i.e. textual) data sources. It is an intermediate layer, which lies between grid applications and Database Management Systems.
The *Data Storage Service* (DSS) provides a comfortable, lightweight solution to disk storage management. It manages efficiently and transparently huge collections of data spread in grid environments, promoting flexible, secure and coordinated storage resource sharing and publication across virtual organizations. DSS besides data handling and remote processing operations, also provide publication and information discovery capabilities, as needed due to the large number of stored objects. The DSS represents a high performance implementation of the grid workspace concept, which is a virtualized

and grid enabled storage space that a community of users can use to share and manage their files/folders taking into account fine-grained data access policies. Grid workspaces represent grid storage spaces accessible by authorized users sharing common interests. Within a DSS, data is fully organized into workspaces and for each one of them, the DSS admin must define a set of authorized users, groups and VOs, the workspace administrators and the physical mounting point.
Finally, the *Data Gather Service* (DGS) offers data federation capabilities providing a second level of virtualization (data integration). This service, which lies on top of DASs or DSSs, allows the user looking at a set of distributed data sources as a single logical entity, thus implementing a data grid federated approach.
The proposed WebGRelC architecture provides ubiquitous web access to widespread grid enabled storage resources and metadata, so it is currently concerned with both *Data Storage and Data Access Services*. Clients are also available both as command line and graphical interface to manage collections of files, but the Grid Portal interface better addresses and increases both transparency and pervasiveness.

3. WEBGRELC

The WebGRelC architecture represents an integrated solution to easily, transparently and securely manage data (collections of files) stored within grid storage repositories (GRelC DSSs) and metadata stored within datasets (accessed by means of GRelC DAS). In the following subsections, we will describe the WebGRelC portal, discussing main goal and challenges, grid architecture, subcomponents, security issues and metadata management.

3.1 Goal and Challenges

The main goal of WebGRelC is to supply users with Grid data management services through transparent, user friendly, secure and pervasive access to web pages. Moreover, through the WebGRelC Grid Portal scientists can explore data stored within storage resources, discover the data they need and retrieve the relevant data collections through web interfaces (which provides location independence).
More in detail, the main challenges are connected with:
a) *security*: it represents a fundamental and crucial requirement in a data grid environment. We need to address security at different levels providing mutual authentication among grid services, users and machines, data encryption and delegation support. Moreover, security concerning HTTP Internet connections needs to be properly addressed. Within our system, we basically chose to adopt the Globus Grid Security Infrastructure [3];
b) *user-friendliness*: the portal must provide user-friendly web pages to simplify the interaction between the users and the data grid environment. Our choice leverages current web technologies, including XHTML, CSS etc. but we also plan to switch to recent developments, including the use of portlets, Java Server Faces etc.;
c) *pervasiveness*: the proposed solution leverages the pervasiveness of Web technology to provide users with ubiquitous grid data management facilities. It is worth noting here that client requirements consist just of a standard web browser;
d) *transparency*: the proposed grid portal must conceal a lot of details about grid storage components, data transfer protocol details, heterogeneous storage resources, technological issues, command line parameters/options, and so on. This requirement is multifaceted and in a data grid environment it concerns, among the others, access, location, namespace, concurrency and failure transparency. Several design and technical choices highlighted within this work address all of these transparency issues.

3.2 Portal Architecture

The grid portal architecture (Fig. 1) follows a standard three-tier model. The first tier is a *client browser* that can securely communicate to a web server over an HTTPS connection (no other specific requirements are imposed). On the second tier, the Web Server implements the WebGRelC portal (WGP) which consists of several components (see Section 3.2.1) leveraging (i) GRB [4], and (ii) GRelC DSS, DAS and SDAI client libraries. WebGRelC interacts with a MyProxy server [5] for secure user's credentials (proxy) storage and retrieval and with the Portal Metadata Catalogue (PMC) to manage user's profiles. Finally, *GRelC DSSs* are deployed on the third tier, the data grid infrastructure, providing a lightweight and grid enabled solution for disk storage management.

3.2.1 WebGRelC Portal

WebGRelC represents the core of the proposed architecture, implementing a grid portal able to

retrieve data and present them (via HTTP protocol), within HTML pages.
As can be seen in Fig. 2 the WebGRelC portal consists of the following:
- *Profile Manager*: it handles the user's profile managing metadata stored within the Portal Metadata Catalog (PMC relational database). It allows (i) inserting, updating and deleting personal information as well as (ii) managing a list of available grid enabled storage resources, workspaces, etc. Currently, the PMC runs on different DBMSs by means of the GRelC Standard Database Access Interface (SDAI, see Section 4.3);
- *Credential Manager*: it allows configuring the credentials to be used for a given set of resources, retrieving them from a MyProxy server. After this initial configuration step, the WebGRelC grid portal transparently retrieves the credentials needed to access specific data sources;
- *Remote Administration*: it provides basic functionalities to access and manage metadata information stored within GRelC DSS Metadata Catalog. Through the portal, the user can remotely manage administration information about (i) users, groups and VOs, (ii) internal workspaces configurations, (iii) data access control policies, etc. Moreover, the proposed grid portal provides admin sections for logging (to display information related to all of the operations carried out at the DSS side), and check-coherence (to report system coherence problems between data (content) and metadata (context));

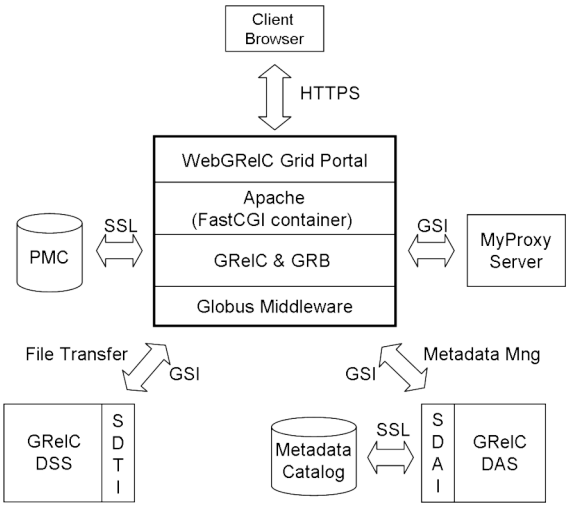

Figure 1. WebGRelC architecture

- *File Manager*: it provides support for (i) basic files transfer operations such as download and upload of files, (ii) workspace browsing, (iii) displaying of file and folder metadata as well as (iv) data access, creation and deletion capabilities (Posix-like oriented). File Manager also supports parallel and partial file transfer as well as file copy between two GRelC DSSs (both push and pull mode are currently available). Along with synchronous functionalities, the proposed grid portal supports the GRelC Reliable File Transfer (G-RFT) mechanism, an asynchronous service primarily intended to be used to reliably copy files from one DSS to another one. Users can submit G-RFT requests simply by filling out a web form containing information about (i) DSS source and destinations, (ii) myproxy access parameters, (iii) file transfer options related to data transfer protocol (HTTPG, GridFTP) and data transfer mode (push or pull), (iv) request options connected with priority level of the G-RFT request, the maximum number of retries to be used in case of failed data transfer, related delay and backoff (linear, exponential). Moreover, within the WGP, users can display status and options about submitted G-RFT requests, abort a data transfer, as well as resubmit again G-RFT requests;
- *Metadata Manager*: it provides basic functionalities to (i) annotate files, that is to publish and manage metadata at the GRelC DAS side, (ii) display metadata information about files, folders, workspaces and schema, (iii) manage metadata schema modifying the list of elements associated to the stored objects and (iv) query metadata information in order to retrieve a list of objects satisfying conjunctive search conditions (basic digital libraries capabilities).

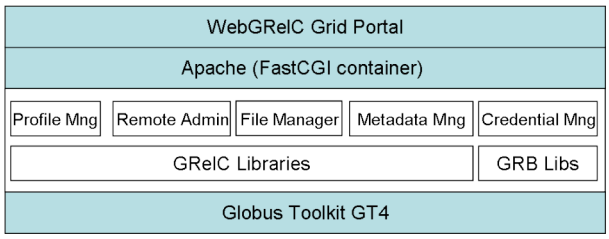

Figure 2. Detailed view of WebGRelC architecture

3.3 Security Issues

To login into the WGP the user must supply the correct username and password. The portal security model includes the use of HTTPS protocol for secure communication with the client browser and secure cookies to establish and maintain user sessions. Moreover, we decided to adopt the Globus Toolkit Grid Security Infrastructure (GSI), (a solution widely accepted

and used in several grid projects) in order to perform security tasks within the data grid environment:
- mutual authentication between WGP and MyProxy servers DAS or DSS;
- communication protection (by means of data cryptography) for the data exchanged between WGP and MyProxy server, DAS or DSS;
- delegation mechanisms to perform data management tasks on the grid.
Finally SSL support is provided when the WGP Profile Manager interacts with the PMC.

3.4 Metadata

Metadata management is crucial within such a data grid system. To aid scientists in discovering interesting files within collections of thousands or millions of files the proposed architecture besides basic data management facilities must provide metadata publication and semantic search capabilities.
Currently, metadata management concerns two different types of metadata: internal (or low level) and application oriented (or high level). In the former case, metadata is related to the physical stored object (for example creation date, file owner, size, etc.) and is system defined, so users cannot modify neither metadata attributes (schema) nor their values (instances). This kind of metadata is managed by the GRelC DSS.
In the latter case, metadata is application specific and related schema can vary depending on the particular context (in this case, through the portal, authorized users can annotate both files and workspaces adding, deleting and updating application level metadata and related metadata schema). Application oriented metadata are stored within the GRelC DAS. Finally, to provide a basic semantic search capability, within the WGP a simple search form has been provided.

4. WEBGRELC IMPLEMENTATION

In the following subsections, we will describe the WebGRelC implementation, discussing involved grid portal technologies, grid middleware, GRelC and GRB libraries.

4.1 Grid Portal Technologies

The WebGRelC grid portal has been developed using a Model View Controller design pattern. To efficiently address performance and modularity we adopted the Fast CGI technology leveraging the Apache Web Server.
Taking into account the main subcomponents of the WGP, within the portal we provided XHTML based web pages to:
- support credential delegation and single sign-on to the Grid;
- configure grid storage resources and services as well as manage grid portal user profile;
- manage DSS users, groups and VOs authorizations;
- upload and download files as well as transfer data from a storage resource to another one;
- perform activities related to digital libraries (metadata based search engine) accessing the GRelC DAS;
- manage metadata and metadata schema related to the objects stored within the storage resources;
- submit and manage G-RFT requests.

4.2 Middleware

The current version of WebGRelC is based on the *Globus Toolkit 4.0.3* (latest stable release as of November 2006) as grid middleware; basically we exploited the Globus GSI libraries.
Web service components such as GRelC DSS and DAS strongly exploit the *gSOAP Web Services development Toolkit* [6]. It offers an XML to C/C++ language binding to ease the development of SOAP/XML Web services in C and C/C++. gSOAP provides a transparent SOAP API using proven compiler technology. These technologies leverage strong typing to map XML schemas to C/C++ definitions. Strong typing provides a greater assurance on content validation of both WSDL schemas and SOAP/XML messages. As a result, SOAP/XML interoperability is achieved with a simple API relieving the user from the burden of WSDL and SOAP details, thus enabling her to concentrate on the application-essential logic. The compiler enables the integration of (legacy) C/C++ software in SOAP applications that share computational resources and information with other Web services, possibly across different platforms, language environments, and disparate organizations located behind firewalls. Finally, to guarantee a secure data communication channel between WGP and DSS or DAS, we utilized the GSI support, available as a gSOAP plug-in [7].
We did not use the Globus Toolkit 4.0.3 C WS Core, which is a C implementation of WSRF (Web Services Resource Framework), because it lacks a usable authorization framework. Even though it is possible to develop WRSF grid services in C (we are already migrating our software to the Globus Toolkit implementation of

WSRF), it is not possible to deploy production level services. Indeed, grid services deployed in the C WS Core container can only use the default SELF authorization scheme (a client will be allowed to use a grid service if the client's identity is the same as the service's identity), which is useless for a production service. Unfortunately, the globus-wsc-container program does not have options to handle different authorization schemes. A possible solution could be the development of a customized service that uses the globus_service_engine API functions to run an embedded container, setting the GLOBUS_SOAP_MESSAGE_AUTHZ_METHOD_KEY attribute on the engine to GLOBUS_SOAP_MESSAGE_AUTHZ_NONE to omit the authorization step and then using the client's distinguished name to perform authorization. However, with the Globus Toolkit 4.0.3 it is not possible for a C grid service to retrieve the distinguished name of a client contacting it, so this is not a viable option and we have to wait for the next stable release of the Globus Toolkit (4.2) that should provide major enhancements to the C WS Core, including a usable authorization framework.

4.3 GRelC Libraries

The GRelC libraries mainly address data management activities. Basically the WebGRelC grid portal exploits the following libraries: GRelC SDAI, SDTI, DAS and DSS libraries.
The *GRelC SDAI* library (used within the WGP Profile Manager component) provides a transparent and uniform access to the PMC relational database. It exploits a plug-in based architecture leveraging dynamic libraries. Currently SDAI wrappers are available for PostgreSQL, MySQL, SQLite, Unix ODBC and Oracle DBMSs. An SQLite PMC implementation has a twofold benefit: it provides extreme performance (due to the embedded database management) and it increases service robustness and reliability because it does not depend on an external DBMS server.
The *GRelC SDTI* provides a data management library to transfer files between WGP and DSS (get/put) or couples of DSSs (copy). Basically, it is a C library (leveraging a plug-in based approach), which virtualizes the data, transfer operations providing high level interfaces connected with get/put/copy (parallel and partial). Two basic modules related to GridFTP and HTTPG (HTTP over GSI) are currently available. Further drivers covering additional protocols (such as FTP, SFTP or SCP) are actively being developed and will be easily added to the system due to the modular design and implementation of this library. This library is used within the WGP File Transfer component.
The *GRelC DAS* client library provides many functionalities related to the DAS component. Among the others, it allows (i) managing metadata, (ii) submitting semantic queries, (iii) browsing metadata, etc. This library is used within the Metadata Manager.
Finally, the *GRelC DSS* client library provides many functionalities related to the interactions with the DSS components. Among the others, it allows (i) managing file transfer and workspaces, (ii) submitting, monitoring and deleting G-RFT requests, (iii) managing users, groups and VOs membership, etc. This library is extensively used within the following WGP components: File Manager and Remote Administration.

4.4 GRB Libraries

GRB software is developed within the Grid Resource Broker project at the CACT of the University of Lecce. Currently, GRB Team supplies users with several production libraries mainly connected with (i) job submission, (ii) resource discovery, (iii) credential management and (iv) grid file transfer.
For WebGRelC development, we exploited the *grb_gridftp* [8] and *grb_myproxy* libraries to respectively transfer data among grid nodes and manage/retrieve user's credentials.

5. USE CASE: WEBGRELC FOR BIOINFORMATICS DATA

The WebGRelC Grid Portal is part of the SPACI [9] middleware (with regard to data management operations) and it is also actively used within the SEPAC [10] production grid. Several GRelC DSSs are now deployed in Europe in the cities of Lecce, Naples, Cosenza, Milan and Zurich managing different storage resources and several tens of thousands of files primarily related to biology experiments.
Through the proposed grid portal, bioinformatics can manage and share their workspace areas, annotate files connected with their experiments. Application level metadata can be published and stored within the system, just filling out web forms provided by the grid portal. Search and retrieval operations on metadata allow users finding desired objects within the system

displaying query results on several formats (HTML tables, plain text or XML).

More in detail, we defined several workspace areas to store experimental results, structure protein, etc.

Bioinformatics workspace contains files produced by the experiments and files that contain the protein structure, retrieved by the UniProt KnowledgeBase (UniProtKB) data bank [11]. The experiment carries out multiple sequence alignment (MSA) among each of the human proteins available in the UniProtKB database (about 70.845 sequences, retrieved by the *uniprot_sprot_human.dat* and *uniprot_trembl_human.dat* flat files) and those stored in the UniProt NREF data bank.

Homologue sequences are hence matched to identify functional domains. Indeed, multiple alignment is important for studying regions that during the evolution are conserved and that characterize, with a good probability, biological functionality of the sequence.

PSI-BLAST (Position Specific Iterative - Blast) [12], available in the NCBI toolkit, has been used for MSA. After running one experiment we produced about 70 thousands alignments files (in XML format) storing them within bioinformatics DSS workspace. The XML Schema Definition of the PSI-Blast output is shown in Fig. 3.

Each resulting XML file contains all of the sequences producing significant alignments for a specified number of iterations (in such an experiment we have considered 2 iterations). Metadata set description of the experiment includes among the others, protein identifier, e-value, score, accession number, etc. Figure 4 illustrates an example query related to the bioinformatics domain.

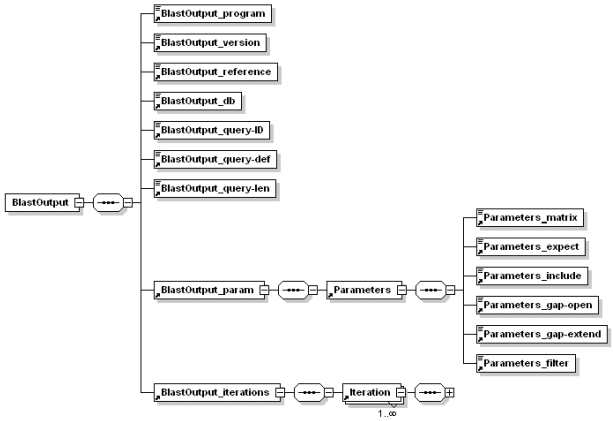

Figure 3. XSD of the PSI-BLAST output

The user can choose a data workspace, and submit a query choosing the output format. A number of files are returned matching the search criteria. The user can then select a file of interest in order to display all of the relevant file metadata. She can also copy, annotate or download these files, as needed.

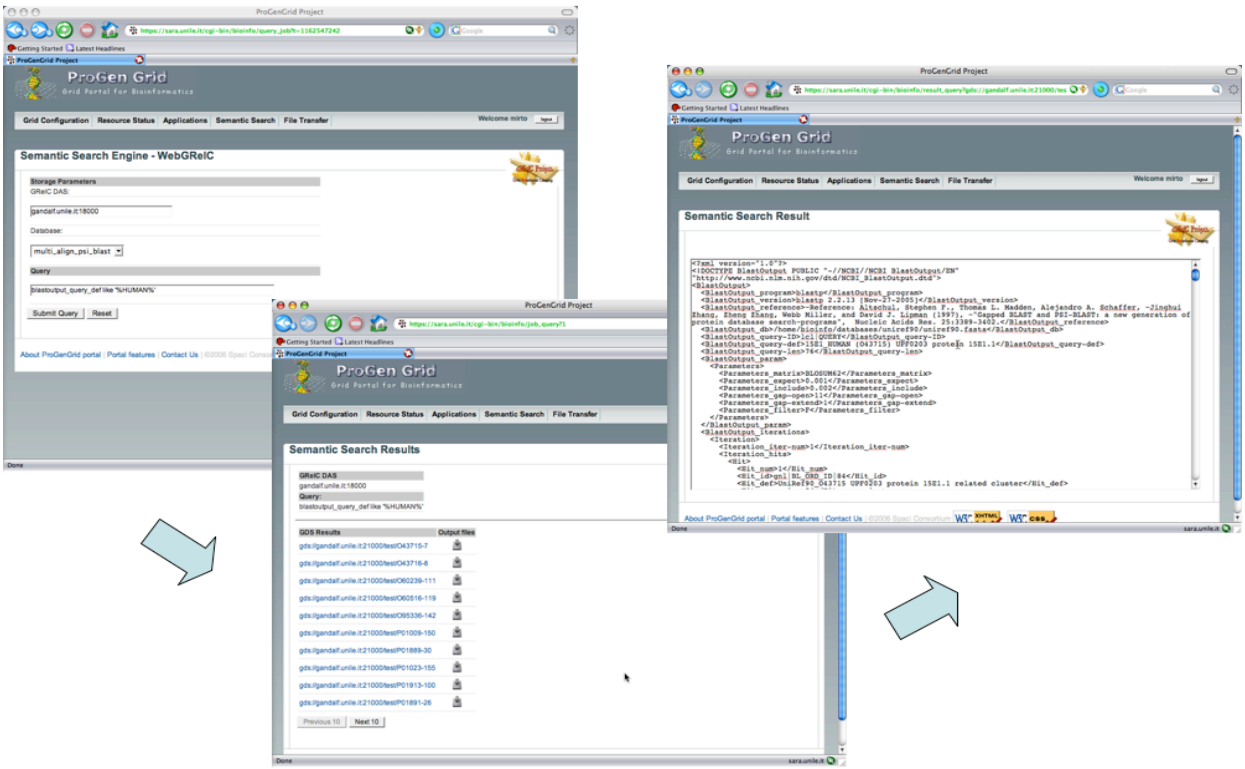

Figure 4. Semantic Search Engine: example query and results

6. RELATED WORK

MySRB [13] is a web-based resource sharing system that allows users to share their scientific data collections with their colleagues. It provides a system where users can organize their files according to logical cataloguing schemes independent of the physical location of the files and associate metadata with these files. MySRB uses the Storage Resource Broker (SRB) [14] and the Metadata Catalog (MCAT) developed at SDSC as its underlying infrastructure. MySRB and WGP share the same goals.

The aim of the DataPortal project [15] is to develop the means for a scientist to explore data resources, discover the data they need and retrieve the relevant datasets through one interface independently of the data location. Separate instances of the DataPortal are currently being installed as part of the grid environments of the eMinerals and eMaterials projects. The new web services architecture of the portal has allowed an easy integration with other services such as the CCLRC HPCPortal. In the current system, these are integrated using standard Web protocols, such as web services, http and SOAP, since support for emerging grid technologies, such as OGSA and grid services is not available.

7. CONCLUSIONS AND FUTURE WORK

The paper presented an overview of the WebGRelC Grid Portal. We presented the portal architecture, and discussed its implementation. WebGRelC bridges the gap between scientists and their data located in grid environments, providing an effective, production-level, data grid management system. The portal is currently in production in the European SPACI and SEPAC grids. The recently released WSRF based Globus Toolkit will be supported in the near future, as soon as the tools provided will be stable and mature enough for production usage.

Future work related to data grid management concerns a complex semantic search engine (based on a P2P federated approach leveraging GRelC DGSs) developed and tested within the SEPAC Production Grid.

References

[1] I. Foster, "Service-Oriented Science, Science", 308, pp 814-817

[2] G. Aloisio, M. Cafaro, S. Fiore, M. Mirto, "The Grid Relational Catalog Project", Advances in Parallel Computing, "Grid Computing: The New Frontiers of High Performance Computing", L. Grandinetti (Ed), pp.129-155, Elsevier, 2005.

[3] Foster, I., Kesselmann, C., Tsudik G., Tuecke, S., "A security Architecture for Computational Grids", Proceedings of 5th ACM Conference on Computer and Communications Security Conference, pp. 83-92, 1998.

[4] G. Aloisio, M. Cafaro, G. Carteni, I. Epicoco, S. Fiore, D. Lezzi, M. Mirto , S. Mocavero, "The Grid Resource Broker Portal", to appear in Concurrency and Computation: Practice and Experience, Special Issue on Grid Computing Environments

[5] Basney J, Humphrey M, Welch V, "The MyProxy Online Credential Repository", Software: Practice and Experience: 2005 (9):801--816

[6] Van Engelen, R.A., Gallivan, K.A, "The gSOAP Toolkit for Web Services and Peer-To-Peer Computing Networks", Proceedings of IEEE CCGrid Conference, May 2002, Berlin, pp- 128--135

[7] Aloisio G, Cafaro M, Epicoco I, Lezzi D, "The GSI plug-in for gSOAP: Enhanced Security, Performance, and Reliability", Proceedings of Information Technology Coding and Computing (ITCC 2005), IEEE Press, Volume I, pp. 304-309

[8] G. Aloisio, M. Cafaro, I. Epicoco, "Early experiences with the GrifFTP protocol using the GRB-GSIFTP library", Future Generation Computer Systems, Volume 18, Number 8 (2002), pp. 1053-1059, Special Issue on Grid Computing: Towards a New Computing Infrastructure, North-Holland

[9] The Italian Southern Partnership for Advanced Computational Infrastructures. http://www.spaci.it

[10] The Southern European Partnership for Advanced Computing. http://www.sepac-grid.org

[11] A. Bairoch, R. Apweiler, C.H. Wu, W.C. Barker, B. Boeckmann, S. Ferro, and et al, "The Universal Protein Resource (UniProt)", Nucleic Acids Res., (33):154–159, 2005. http://www.uniprot.org.

[12] S.F. Altschul, T.L. Madden, A.A. Schffer, J.Zhang, Z. Zhang, W. Miller, and D.J. Lipman, "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs", Nucleic Acids Res., 17(25):33893402, 1997 http://www.ncbi.nlm.nih.gov/cgi-bin/BLAST/nph-psi.

[13] Michael Wan and Reagan Moore, Arcot Rajasekar, "MySRB & SRB Components of a Data Grid", The 11th International Symposium on High Performance Distributed Computing (HPDC-11) Edinburgh, Scotland, July 24-26, 2002

[14] Chaitanya Baru, Reagan Moore, Arcot Rajasekar, Michael Wan, “The SDSC Storage Resource Broker”, Proc. CASCON'98 Conference, Nov.30-Dec.3, 1998, Toronto, Canada

[15] DataPortal http://www.e-science.clrc.ac.uk/web/projects/dataportal