# Workflow level interoperation of grid data resources.

**Tamas Kiss**[1]
**Peter Kacsuk**[2]
**Gabor Terstyanszky**[1]
**Stephen Winter**[1]

[1] School of Informatics, University of Westminster
[2] MTA-SZTAKI

# Workflow Level Interoperation of Grid Data Resources

Tamas Kiss, Peter Kacsuk, Gabor Terstyanszky and Stephen Winter

*Abstract*—**The lack of widely accepted standards and the use of different middleware solutions divide today's Grid resources into non-interoperable production Grid islands. On the other hand, more and more experiments require such a large number of resources that the interoperation of existing production Grids becomes inevitable. This paper, based on the current results of grid interoperation studies, defines generic requirements towards the workflow level interoperation of grid solutions. It concentrates on intra-workflow interoperation of grid data resources, as one of the key areas of generic interoperation, and describes through an example how existing tools can be extended to achieve the required level of interoperation.**

*Index Terms*—**Grid workflow, interoperation, grid data resources, SRB, OGSA-DAI**

## I. Introduction

Grid computing has now reached the phase when significant production level resources are available for scientists to run computationally intensive experiments and to access large distributed data collections. These production grids enable the construction and execution of experiments in a previously unimagined scale triggering even more resource intensive scenarios. Unfortunately, current grid resources are provided as non-interoperable "production grid islands" that cause difficulties when the experiment outgrows the capabilities of one specific production grid. Different production Grids are based on different grid middleware, use different tools and policies for authentication and authorization, describe and submit jobs differently, and provide data services based on a variety of protocols and access mechanisms. This makes the utilization of resources from several production grids rather cumbersome.

No wonder that the Grid community devotes quite a lot of effort towards grid interoperability. The most notable among these efforts is the Grid Interoperation Now (GIN) Community Group [1] of the Open Grid Forum (OGF). The GIN, on one hand, is devoted to short term *interoperation* of production Grids by defining what needs to be done using existing technologies. On the other hand, the GIN also supports long term *interoperability* defined as the native ability of Grids and Grid middleware to interact directly via common open standards [2]. Although interoperation is not the perfect solution, it can provide immediate access to heterogeneous grid resources. The definition, recognition and implementation of standards provide a better solution but can also take much longer. These two terms, interoperation and interoperability, will be used based on the above definitions throughout this paper.

The GIN focuses on four key areas: information services, job submission, data movement, and authorization and identity management. The aim of their work is to achieve the interoperation of the most widely recognized middleware level tools in the above mentioned 4 areas, and also to contribute towards long term standardization.

Although, this interoperation and interoperability enables the utilization of a diverse set of resources, there is still a long way to go until grid end-users can seamlessly access these resources in a transparent way from a high-level environment. The typical higher level tools in grid computing include workflow engines that enable the construction and execution of complex grid applications, and grid portals that provide access to the grid via a simple Web browser interface. Interoperation at this high level is necessary in order to make the results of the GIN available for large user communities. Also, as these high level tools are capable to hide the middleware level incompatibilities from the user, in some cases they even provide an easier and more convenient solution for interoperation than the middleware level tools.

This paper focuses on data movement as one specific aspect of interoperation. It analyzes and provides solutions how the interoperation of different grid data resources can be solved at the level of workflows. Section 2 gives an overview of Grid interoperation/interoperability work and defines the different aspects of workflow level interoperation. Section 3 describes the generic workflow level requirements towards intra-grid data interoperation, while section 4 introduces how these requirements are being implemented in the P-GRADE [3] grid portal and workflow engine. Finally, section 5 gives conclusions and highlights future work.

## II. Workflow Level Interoperation

As it was described in the Introduction, the major force behind grid interoperation and interoperability research is the GIN Community Group. The GIN is carrying out several demonstrations and investigates current interoperation of middleware solutions in its four identified areas [2] [11]. In

Manuscript received November 12, 2007. This research work is supported by the FP6 Network of Excellence CoreGRID funded by the European Commission (Contract IST-2002-004265).

T. Kiss is with the University of Westminster, 115 New Cavendish Street London, UK, W1W 6UW, phone: 44-2079115000 ext 3542; e-mail: kisst@ wmin.ac.uk ).

P. Kacsuk is with MTA-SZTAKI, 1111 Kende u. 3, Budapest, Hungary, e-mail: kacsuk@sztaki.hu

G. Terstyanszky and S.Winter are with the University of Westminster, 115 New Cavendish Street London, UK, W1W 6UW

978-0-7695-3156-4/08 $25.00 © 2008 IEEE
DOI 10.1109/CCGRID.2008.53

conjunction with the GIN activities several research projects are investigating the middleware level interoperation of current production Grids. The OMII-UK (Open Middleware and Infrastructure Institute) [4] team at Southampton has investigated the interoperation of job submission mechanisms [5] based on the Job Submission Description Language (JSDL) [6], the Open Grid Services Architecture (OGSA) Basic Execution Service (BES) [7] and the High Performance Computing Profile (HPC-Profile) [8] standards. Other tools, like GridSAM [28] or the NGS Application Repository [29] also apply JSDL as the standard job description language and submit the described jobs to a variety of grid middleware. Within the framework of the European CoreGrid project a meta-broker concept has been introduced to solve Grid interoperation and interoperability at job submission level [42]. This solution is also based on JSDL but can also interoperate with Grids using various non-standard job submission languages [43]. The two largest UK-based production Grids, the UK National Grid Service (NGS) [9] and the GridPP (UK Computing for Partical Physics) [10] have recently analyzed different aspects of middleware level interoperation proposing short and long term solutions [12]. European projects, like the GRIP [13] or more recently the OMII Europe [14] has been launched to solve long term interoperability between the most widely used Grid middleware, including Globus [15], gLite [16] and Unicore [17].

Although middleware level interoperation is the foundation, end-users typically require high level tools to interact with resources. Our aim is that building on the GIN activities and also by extending them we define different aspects of workflow level grid interoperation and provide interoperable solutions for end-users. Grid workflow systems are widely utilized in e-Science applications in order to compose and orchestrate the execution of several jobs and services and to automate data transfer between them. Examples include Taverna [18], Triana[19], Kepler[20], P-GRADE [3], or the OMII-BPEL [31] workflow engines. Despite the widespread utilization of grid workflow engines, relatively small effort has been put into workflow level interoperation so far. Grid workflow solutions are usually coupled with one particular grid technology and come with a custom user interface that does not integrate well into Web-based grid portals. The different workflow solutions use different workflow languages for representation and cannot share components or data with each other. The Workflows Hosted in Portals project (WHIP) [21] targets some areas of workflow interoperation. It defines a set of software plug-in utilities that supports the interaction of workflows and Web portals enabling the easier integration of new workflow engines into grid portal frameworks. However, no research project is dedicated at the moment to the comprehensive study of workflow level grid interoperation. We summarize different aspects of this problem area below.

Grid interoperation at the level of workflows can be achieved inside one workflow or within several different workflow systems. Therefore, we can talk about intra- or inter workflow interoperation, respectively. In both cases, we have to examine two different aspects of interoperation. One aspect is the execution of workflow components on computing resources spanning several grids. The other is the movement of data between heterogeneous data resources and workflow components. Table 1 summarizes these four areas of workflow level interoperation. Let us introduce these four cases in bit more detail.

Intra-workflow interoperation of workflow component execution means that jobs or services of the same workflow can be mapped to different production grids based on different grid middleware solutions. In [22] we showed how the P-GRADE grid portal, as an example for a workflow system integrated into a Web portal interface, supports this level of interoperation of large production Grids, including the GT2 based UK NGS [9], TeraGrid [13] and Open Science Grid [24], the GT4-based WestFocus Grid [25], and the LCG/g-Lite based EGEE Grid [26]. We also showed that workflow components can either be jobs that are submitted directly by the user or selected from a central application repository, but can also be legacy applications invoked as services [27].

Intra-workflow interoperation of grid data resources allows data to be input from or output to different file storage systems or database solutions, located in several different grids. Our current paper focuses on this particular topic and will suggest solutions in section 3.

TABLE I
WORKFLOW LEVEL INTEROPERATION

| Workflow level interoperation | WF component execution | Data resources |
|---|---|---|
| Intra-workflow | Jobs or services of one particular workflow, Heterogeneous computing resources from different grids | Jobs or services of one particular workflow, Heterogeneous data sources from different grids |
| Inter-workflow | Jobs or services of multiple workflow systems, Heterogeneous computing resources from different grids | Jobs or services of multiple workflow systems, Heterogeneous data sources from different grids, Data conversion between WF systems |

Inter-workflow interoperation allows embedding a workflow created in workflow concept “A” into a workflow based on workflow concept “B”. For example, a Triana workflow, already developed by someone and fulfilling some required functionality could be called from inside a Taverna workflow. In case of inter-workflow interoperation the participating workflows can individually all support the intra-workflow interoperation concept (e.g. the individual workflows can be mapped to several grids), and the different workflows can also use a different set of grids they can be mapped to (e.g. workflow “A” can be mapped to Grids “X” and “Y”, while workflow “B” can be mapped to grids “Y” and “Z”).

A solution framework for workflow component execution in case of inter-workflow interoperation was suggested in [30]. The suggested solution wraps workflows into Web service interfaces creating Workflow Grid Services (WFGS), and publishes them in workflow registries and repositories where other workflow engines can find them and request their execution. The advantage of the proposed solution is that it does not require the standardization of workflow representations and only some relatively minor extensions of existing workflow solutions are needed.

Inter-workflow interoperation of grid data resources includes one additional challenge compared to the intra-workflow case. The input/output data representation of different workflow systems are typically different requiring necessary data conversions for interoperation. Here a standard input/output data format that is handled by all interoperating workflow systems, or mediation between the different data representations are necessary. Solution for this scenario will be investigated and proposed in a forthcoming paper.

We have to note that although we were concentrating on workflow component execution (job submission) and data transfer only, these two areas also cover some aspects of the other two GIN interoperation areas: authorization/identity management and information services. For example, the capability to execute workflow components in several grids requires authorized access to resources in these grids. Also, the broker based execution and mapping of workflow components needs access to information services in multiple grids. Solutions presented in [22] and also in section 4 of this paper reflect on these aspects.

## III. Intra-workflow Interoperation of Grid Data Resources

When defining the requirements towards intra-workflow interoperation of grid data resources we use the work of the GIN in the area of data movement as a starting point. The GIN focuses on the interoperation of the two most widely used file storage systems, SRM (Storage Resource Manager) [32] and SRB (Storage Resource Broker) [33].

SRM is a protocol for Grid access to mass storage systems (e.g. tapes, disks, or disk arrays). SRM does not do any data transfer. The protocol is used to ask a storage system to make a file ready for transfer, or to create space in a disk cache to which a file can be uploaded. The file is then transferred via the means of a file transfer protocol, typically GridFTP [34]. SRM and GridFTP form the basis of data handling in the EGEE Grid, for example.

SRB on the other hand is a more all-encompassing single solution for data management, including file movement, file replica and metadata management. SRB is widely utilized in Globus based grids like the US TeraGrid or the UK NGS.

Although the GIN successfully demonstrated the interoperation within different SRM implementations and also between different SRB deployments [2], there is basically no interoperability between these two (SRB and SRM) solutions at the moment. This divides grid file storage systems into two non-interoperable islands.

Although SRM and SRB are non-interoperable at middleware level, Grid workflow solutions can successfully mediate between these different file storage systems at a higher level. Direct transfer between these storage systems may not be possible, however, a workflow component can still get its input data from various resources, freely combining SRB and SRM solutions. The component then applies its analysis or transformation on the data and feeds it back to either SRM or SRB storage. Section 4 will present an example for this solution, where some input files of a P-GRADE workflow were coming from EGEE storage elements based on SRM, while others were stored in SRB resources or in GridFTP catalogues. These data resources can not only be heterogeneous in nature but can also be situated in different grids that may require different grid user certificates for authentication.

Grid related research and development activities mainly concentrated on systems until recently where data was stored in flat files. However, many scientific and industrial applications rely on database management systems to provide a more structured access to mass amount of data. Although database integration to grid systems is addressed on several forums like the DAIS (Data Access and Integration Services) [35] working group of the OGF or the OGSA-DAI project [36], interoperation of file storage solutions and databases is not included in the GIN activities. While direct data transfer would be rather difficult between these logically different solutions, it is possible to combine file and database resources at workflow level. The second main area of our investigation is how to feed data from relational or XML databases to computational workflows and how to combine these with more traditional file storage systems.

Figure 1 shows the generic requirements towards intra-workflow interoperation of grid data resources. The input files of the workflow (shown in the middle of the picture) can come from file systems, like SRM or SRB, or from database management systems and the result can also be fed into any of these solutions. Both the file systems and the databases can be located in different production grids, and the jobs of the workflow can also be mapped to different grids. These grids can be based on different grid middleware and may require different user certificates for authentication.

The forthcoming section will show how some of these generic requirements are already supported by the P-GRADE portal and workflow engine, and how we plan to achieve the generic interoperation presented here, in the future.

## IV. Interoperating Data Resources in P-GRADE

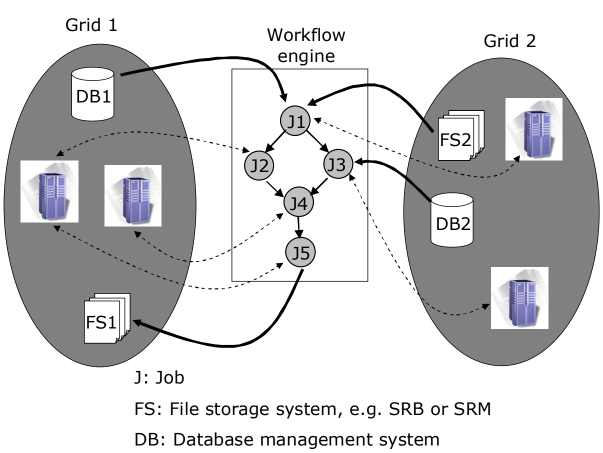


Fig. 1. Generic requirements towards intra-workflow interoperation of grid data resources

## WORKFLOWS

This section presents how the generic principles defined in section 3 can be implemented for a real grid workflow system. This section uses the P-GRADE portal and workflow engine as an example. However, the solutions presented here are valid and relevant for other grid workflow systems too.

P-GRADE is a workflow-oriented grid portal and application hosting environment that supports the execution of workflows represented as DAGs (Directed Acyclic Graph) in multiple grids. The generic structure of a P-GRADE workflow is explained on figure 2. Detailed description of the portal and its workflow engine can be found in several publications such as [3], or on the P-GRADE portal Website [37]. This paper concentrates only on data handling issues and shows how generic intra-workflow interoperation of grid data resources is achieved by P-GRADE.

Inputs and outputs of P-GRADE workflow components are represented as files. These files can either be stored locally on the portal server, or could be remote files stored on grid storage locations. The initial design of P-GRADE supported only GridFTP file collections as remote files (as the portal was initially designed for Globus 2 based grids). When P-GRADE was connected to the LCG/g-Lite based EGEE grid the

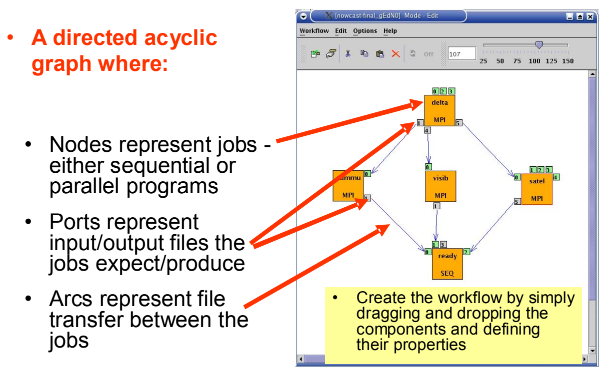


Fig. 2. A P-GRADE portal workflow

handling of GridFTP file catalogues were extended towards EGEE storage elements. Although these storage elements are using SRM for data access, the files are still transferred via GridFTP making this integration relatively painless. The latest production release of the portal (2.5) supports these two types of remote storage: GridFTP file catalogues and SRM based EGEE logical file systems and storage elements. These remote file handling mechanisms can be freely combined within a P-GRADE workflow allowing to take input files from both systems and also to feed the output back to either of them. Both the storage and the computing resources can be in multiple and different grids allowing interoperation and multiple grid support. Figure 3 shows local and remote file handling mechanisms in P-GRADE 2.5.

Besides the workflow level integration described above, an SRM browser portlet has also been developed for the P-GRADE portal by Middle East Technical University of Ankara and used as a production service for TR-Grid (Turkish National Grid) [44].

Although these features of P-GRADE allow combining GridFTP and SRM based file catalogues, two extensions of these data handling mechanisms are required in order to achieve the generic interoperation scenario presented in the previous section. The remaining part of this section describes how P-GRADE has been extended with SRB support and how database access through OGSA-DAI can be integrated into the current architecture.

### *A. Accessing SRB resources from P-GRADE*

SRB functionalities can be offered in the P-GRADE portal in two different ways. The first solution is to extend the portal with an SRB browser portlet. This solution connects the portal to SRB file catalogues and allows users to browse and perform file operations on these resources. Adding an SRB browser

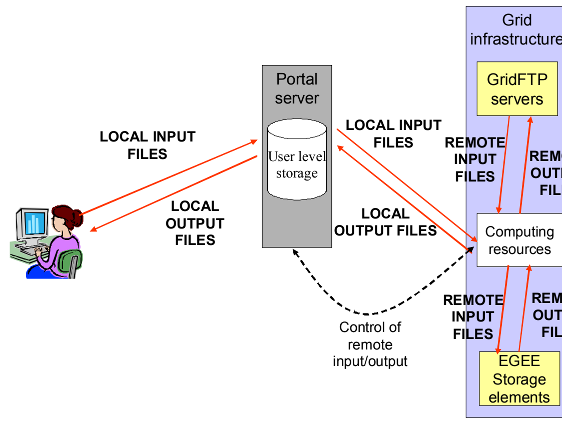


Fig. 3. Local and remote files in PGRADE workflows

portlet to P-GRADE definitely enhances its capabilities and it is a rather useful tool for many end-users. They can now perform a wide range of operations on SRB resources using the portal's graphical user interface. However, it takes us only a little bit closer to workflow level interoperation as the data still has to be manually copied from and to the SRB resource and fed into the workflow. Figure 4 illustrates the SRB browser portlet implemented for P-GRADE. Although there are some other similar portlets available (e.g. in the BIRN portal [39]), the reason for implementing our own solution was justified by its flexible architecture. The portlet was designed in a plug-in structure which allows easily extending its functionality towards other file storage systems like GridFTP or EGEE file catalogues. The portlet, besides the usual file and directory operations, also supports metadata creation and handling.

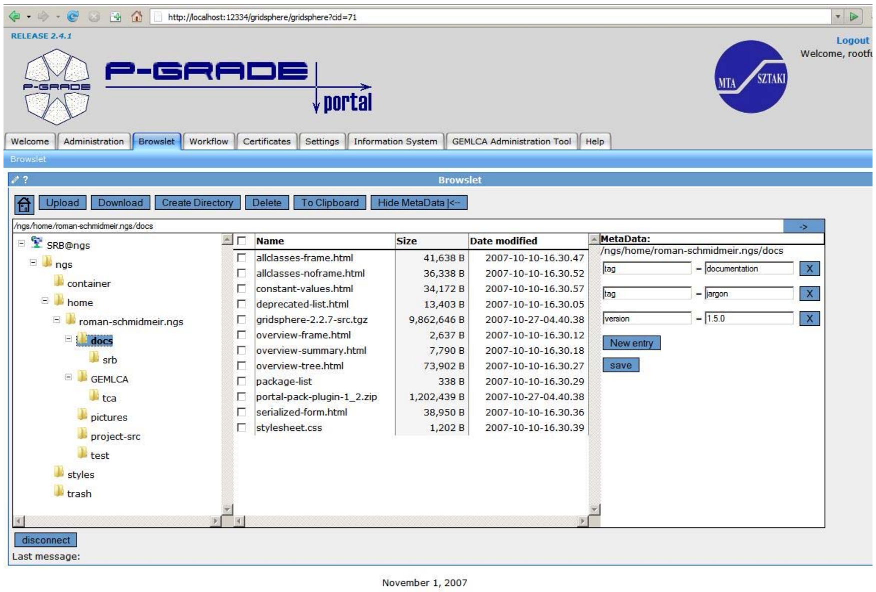

Fig. 4. SRB browser portlet in P-GRADE portal

In order to achieve real workflow level interoperation of SRB and the already supported data sources (GirdFTP catalogues and EGEE storage elements), workflow level integration of P-GRADE and SRB was necessary [38]. While the SRB browser portlet was implemented independently from the portal and simply plugged in to the common GridSphere [40] platform, the workflow level integration raised many more challenges.

As figure 2 shows input and output files in P-GRADE are represented as ports. The natural way to integrate SRB with P-GRADE was to extend it with a new port type called SRB. This integration raises three challenges.

The first challenge is setting up and configuring the SRB client environment. When dealing with SRB environment descriptions, the aim was to enable access to multiple SRB servers at the same time. These SRB servers can be located in different Grids and may require different certificates to access. P-GRADE allows storing multiple certificate proxies on the portal server at the same time, mapping them to different Grids. Our aim was to extend this multi-Grid capability to SRB resources. In order to set up the client environmental variables, an SRB Settings portlet has been developed and added to P-GRADE allowing users creating, uploading and modifying SRB environmental files (*MdasEnv)*. Users can load MdasEnv files from their own file-system, can view and modify existing SRB environmental files and most importantly they can create new ones on the fly. This portlet has been designed and implemented in such way that enables the portal to handle multiple MdasEnv files for a particular user. These environmental files are mapped to Grids and this way linked to potentially different Grid user certificates. The solution extends the multi-Grid capabilities of the portal to SRB resources and allows users to connect to multiple SRB servers concurrently, independently of their Grid membership.

The second challenge was enabling the creation of SRB input and output ports in the P-GRADE workflow editor. This required the introduction of a new port type called "SRB" (besides the currently existing "Local" and "Remote" types), and also embedding a small SRB browser into the editor to select input files and output file locations. In the workflow editor the user first selects the environmental file to be used. As this file is mapped to a particular Grid, the portal knows which certificate to use when accessing the selected collection. Following this, the SRB file path is either given directly or selected by using the built in SRB file browser.

The final challenge is to actually access the SRB data collection to retrieve the selected files before job execution, and also to copy SRB output files back to the SRB resource. When moving these files the portal machine can easily become a bottleneck if all SRB file transfers occur and terminate on this machine. In order to overcome this shortcoming the implemented solution utilises direct file transfer between the SRB resource and the executor site whenever it is possible (see figure 5). The portal checks whether the executor site has an SRB client installed by looking for the SRB client executable using GSI SSH

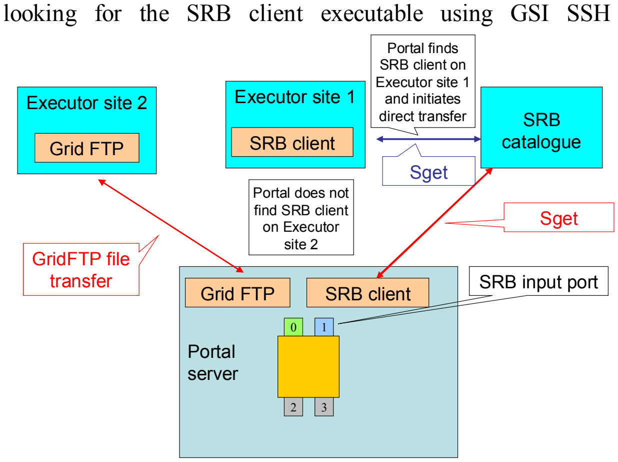


Fig. 5. Direct and indirect transfer of SRB data in P-GRADE

(Executor site 1 on figure 5). If it finds a client on the site then the portal utilises this client to directly transfer input/output files. If the executor site does not support SRB (Executor site 2) or the direct transfer failed, then the file is first transferred to the portal server utilising the portal's SRB client, and then to the executor site by the means of GridFTP (in this case the maximum size of the transferred file has to be limited to protect the portal server).

This workflow level integration now allows the seamless interoperation of SRB catalogues, GridFTP file systems and EGEE storage elements (based on SRM) at the level of P-GRADE workflows. Figure 6 illustrates a workflow simulating urban car traffic [41]. As it is shown on the figure, jobs of the workflow are running in different grids (US OSG, UK NGS, EGEE) and utilise data resources based on different technologies (SRB, SRM, GridFTP, local) from these different grids .

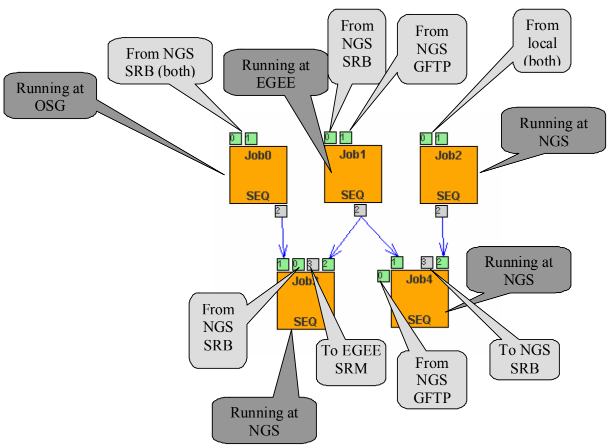


Fig.6 Workflow demonstrating the interoperation of SRB, SRM and GridFTP data sources in P-GRADE

*B. Integrating P-GRADE with Data Resources Exposed by OGSA-DAI*

In order to achieve the generic intra-workflow interoperation of grid data resources as shown on figure 1, data coming from relational or other databases has to be also accommodated into workflows. Providing support individually for every database solution available on the market is non-feasible. Although the DAIS Working Group of the OGF is working on the specification of a generic interface [35] to expose database operations, there is also a clear need for higher level services abstracting database access and making application development easier. Such generic tool exposing database operations as services is OGSA-DAI [36]. OGSA-DAI was selected as the middleware level solution to be integrated with P-GRADE due to its flexible architecture and widespread utilization by the grid community.

As P-GRADE workflows use ports to represent input/output files, the integration of primarily file-based storage systems like SRB and SRM were relatively straightforward. Unfortunately data coming from relational or XML databases does not fit so nicely into this concept.

The primary aim when accessing a database in the grid is to move computation to the data rather than the other way round. This is justified by the huge size of databases and also by the fact that the service representation requires the conversion of data to SOAP-XML format. When integrating OGSA-DAI to P-GRADE we have to keep this very important feature of OGSA-DAI intact and move the data only when it is absolutely necessary. But when is it absolutely necessary? Watson [45] lists an important exception from the generic rule. Moving the data is inevitable when some complex analysis has to be carried out on a data set delivered as a result of a database operation (e.g. a query), and the computing resources available on the database server are not adequate to run this analysis. It is quite common in an e-Science scenario that the result of a query produces a set of data that forms the input of a parameter sweep experiment. P-GRADE supports the creation of parameter study workflows, as explained in [46]. Therefore, an important requirement towards the integration is that data coming as a result of OGSA-DAI queries should be able to serve as input to these parameter sweeps. As the parameter study workflows may require hundreds or ever thousands of resources to run, the only feasible solution is to move the data to computational resources where the analysis is carried out.

Based on the above, the major principles of OGSA-DAI P-GRADE integration are the following:

1. Allow a set of database operations to be delivered to OGSA-DAI resources as P-GRADE workflow components (move operations to data).
2. Allow delivering the results of OGSA-DAI operations, when it is required, to form the input of parameter study workflows in P-GRADE.

To provide a first proof of concept implementation of the generic integration principles, a set of OGSA-DAI portlets has been developed and integrated to the P-GRADE portal. Besides the common database operations (e.g. select, update etc.) the portlets also support the delivery of an OGSA-DAI query to a set of files that can serve as inputs for P-GRADE parameter study workflows.

The four OGSA-DAI portlets implemented offer the following functionality:

– *ServiceManager* portlet: allows connecting to an OGSA-DAI data service and listing resources and resource properties of already connected services.
– *DataBrowser* portlet: to browse the content of selected data resources.
– *QueryManager* portlet: to run queries on selected database and either display results on screen or deliver them to a set of files via GridFTP.
– *ManipulationManager* portlet: to update database content either via an update query or by delivering content from a set of files.

Figure 7 shows the *QueryManager* portlet and illustrates its major functionalities.

A typical use case where the results of database operations form the input of a parameter study workflow is as follows:

1. The user sets up the target OGSA-DAI connections using the *ServiceManager* portlet.

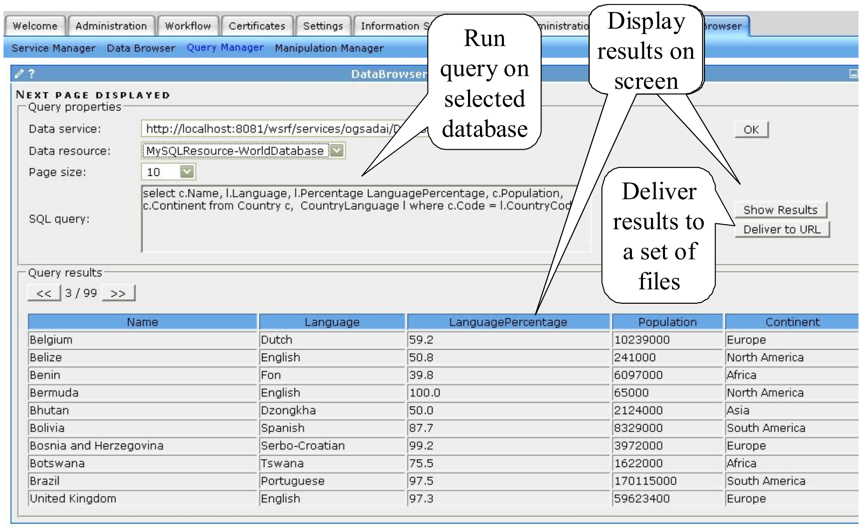


Fig. 7. OGSA-DAI *QueryManager* portlet in P-GRADE portal

2. OGSA-DAI service and database resource are selected in the *QueryManager* portlet.
3. Using the *QueryManager* portlet a set of queries are defined and run on the database.
4. Results of the queries are delivered to a set of files. The *QueryManager* portlet allows to compress the data and to slice it up to a given number of files.
5. The user defines the parameter study workflow pointing to the files delivered from OGSA-DAI as input parameter sets. P-GRADE automatically generates the defined number of workflow instances, runs them on the associated computing resources and delivers the results back to a set of files.
6. The *ManipulationManager* portlet merges the result data and delivers it back to the OGSA-DAI database.

The current implementation conforms to the generic principles (listed before) as it moves the operations to the database in the form of an OGSA-DAI perform document, and it is also capable to move and store the data remotely when it is needed for parameter study experiments.

Although the current solution provides instant advantages for end-users by combining the power of P-GRADE parameter sweep workflows with data stored in OGSA-DAI data resources, it is also a rather cumbersome solution and needs further improvement. The offered functionalities are not seamlessly integrated into the P-GRADE workflow system at the moment. The query has to be performed manually from an independent portlet before feeding its result to the workflow. Also, the OGSA-DAI data operations should be carried out as integral part of a P-GRADE workflow. In order to fulfill these requirements a new job type has to be defined in P-GRADE, called OGSA-DAI job. An OGSA-DAI job takes an OGSA-DAI perform document as input and produces an OGSA-DAI response document as an output. As additional parameters, the identity of the target OGSA-DAI data services and resources has to be specified. These OGSA-DAI jobs can then be embedded into P-GRADE workflows and combined with all standard features, for example multi-grid execution or SRB/SRM data handling support. Work is currently undertaken to fully implement this solution.

## V. Conclusions and Future Work

This paper defined the generic requirements towards workflow level grid interoperation at both intra- and inter-workflow levels. The interoperation has to be provided for both the execution of workflow components and also for the data sources and data movements involved. We concentrated on intra-workflow interoperation of grid data resources and showed how the P-GRADE portal and workflow engine are being extended to fulfill the generic requirements.

Future work involves fully implementing the presented solutions and incorporating them to production level P-GRADE installations. Work on inter-workflow interoperation is also under way by defining requirements and potential solutions for the interoperation of the most widely used workflow engines like Taverna, Triana, Kepler or P-GRADE.

## References


[1] Open Grid Forum (OGF) Grid Interoperation Now Community Group (GIN-CG): https://forge.gridforum.org/sf/projects/gin

[2] Editor: Morris Riedel: GIN – Data Management and Movement (GIN – DATA), Experiences from interoperation scenarios in production Grids, DRAFT version 1.00, 30/04/2007, http://forge.ogf.org/sf/docman/do/downloadDocument/projects.gin/docman.root.current_drafts.gin_data_experience_doc/doc14499;jsessionid=A21053D63FF203CBD617A8FA3BD0F984

[3] P. Kacsuk, G. Sipos: Multi-Grid, Multi-User Workflows in the P-GRADE Grid Portal, Journal of Grid Computing Vol. 3. No. 3-4., 2005, Springer, 1570-7873, pp 221-238, DOI: 10.1007/s10723-005-9012-6

[4] OMII-UK (Open Middleware and Infrastructure Institute) Website: http://www.omii.ac.uk/

[5] R. Boardman, S. Crouch, H. Mills, S. Newhouse, J. Papay: Towards Grid Interoperability, Proceedings of the UK e-Science All Hands Meeting 2007, Nottingham, UK, 10th-13th September 2007, pp 127-134, ISBN 978-0-9553988-3-4

[6] Ali Anjomshoaa et. al.: Job Submission Description Language (JSDL) Specification, GFD-R.056, 7 November 2005, www.gridforum.org/documents/GFD.56.pdf

[7] I. Foster et. al.: OGSA Basic Execution Service version 1.0, 17 March, 2007, http://www.ogf.org/Public_Comment_Docs/Documents/Apr-2007/ogsa-bes-v33.pdf
[8] Blair Dillaway et. al.: HPC Basic Profile Version 1.0, 2 October, 2007, http://www.ogf.org/Public_Comment_Docs/Documents/Feb-2007/HPC_Basic_Profile_v1.0.pdf
[9] The UK National Grid Service Website, http://www.ngs.ac.uk/
[10] GridPP, UK Computing for Particle Physics Website, http://www.gridpp.ac.uk/
[11] Editor: Morris Riedel: GIN – Information Services and Schema (GIN – INFO) Experiences from interoperation scenarios in production Grids, DRAFT version 1.00, 02/10/2007, http://omii-europe.org/OMII-Europe/docs/2007-02-10_OGF_GIN-Info_v_1_00.pdf
[12] J. Jensen, G. A Stewart, M. Viljoen, D. Wallom, S. Young: Practical Grid Interoperability: GridPP and the National Grid Service, Proceedings of the UK e-Science All Hands Meeting 2007, Nottingham, UK, 10th-13th September 2007, pp 119-126, ISBN 978-0-9553988-3-4
[13] The GRIP project Web site, http://www.gloriad.org/gloriad/projects/project000032.html
[14] OMII Europe Project Website, http://omii-europe.org/OMII-Europe/igiiw2007.html
[15] Globus Team, Globus Toolkit 4.0 Release Manuals, http://www.globus.org/toolkit/docs/4.0/
[16] The gLite Website, http://glite.web.cern.ch/glite/
[17] The Unicore Website in wikipedia, http://en.wikipedia.org/wiki/UNICORE
[18] T. Oinn, et. al.: Taverna: A tool for the composition and enactment of bioinformatics workflows. *Bioinformatics Journal*, 20(17):3045–3054, June 2004
[19] D. Churches et.al: Programming Scientific and Distributed Workflow with Triana Services. *Grid Workflow 2004, Concurrency and Computation: Practice and Experience*, Vol 18, Issue 10, August 2006, pp 1021-1037, ISSN 1532-0626
[20] M. Jones, E. A. Lee, and Y. Zhao: Scientific workflow management and the Kepler system. *Concurrency and Computation: Practice and Experience,* Vol 18, Issue 10, August 2006, pp 1039-1065, ISSN 1532-0626
[21] A. Harrison, I. Kelley, K. Mueller, M. Shields, I. Taylor: Workflows Hosted In Portals, Proceedings of the UK e-Science All Hands Meeting 2007, Nottingham, UK, 10th-13th September 2007, pp 32-39, ISBN 978-0-9553988-3-4
[22] P. Kacsuk, T. Kiss, G. Sipos, Solving the Grid Interoperability Problem by P-GRADE Portal at Workflow Level, Conf. Proc. of the Grid-Enabling Legacy Applications and Supporting End Users Workshop, within the framework of the 15th IEEE International Symposium on High Performance Distributed Computing , HPDC'15, Paris, France, pp 3-7, June 19-23, 2006
[23] The TeraGrid Website, http://www.teragrid.org
[24] The Open Science Grid Website, http://www.opensciencegrid.org/
[25] The WestFocus Grid Alliance Website, http://www.westfocus.org.uk/ICT/p54g12_About_the_project.aspx
[26] The EGEE Web page, http://public.eu-egee.org/
[27] T. Delaittre, T. Kiss, A. Goyeneche, G. Terstyanszky, S.Winter, P. Kacsuk: GEMLCA: Running Legacy Code Applications as Grid Services, Journal of Grid Computing Vol. 3. No. 1-2., 2005, Springer, 1570-7873, pp 75-90
[28] W. Lee, A. S. McGough, and J. Darlington: Performance Evaluation of the GridSAM Job Submission and Monitoring System, Conf. Proc. of the UK e-Science All Hands Meeting, Nottingham, UK, 2005, ISBN 1-904425-534, pp 915-922
[29] D. Meredith, M. Maniopoulou, A. Richards, M. Mineter: A JSDL Application Repository and Artefact Sharing Portal for Heterogeneous Grids and the NGS, Proceedings of the UK e-Science All Hands Meeting 2007, Nottingham, UK, 10th-13th September 2007, pp 110-118, ISBN 978-0-9553988-3-4
[30] P. Kacsuk, T. Kiss: Towards a scientific workflow-oriented computational World Wide Grid, CoreGrid Technical Report, TR-0115, (To be published)
[31] B. Wassermann, W. Emmerich, B. Butchart. Sedna: A BPELbased environment for visual scientific workflow modelling. In Ian Taylor, Ewa Deelman, Dennis Gannon, and Matthew Shields, editors, *Workflows for e-Science*, pp 428–449. Springer, New York, Secaucus, NJ, USA, 2007
[32] A. Sim, A. Soshani editors, Storage Resource Manager Interface Specification version 2.2, 09.05.2007, http://www.ogf.org/Public_Comment_Docs/Documents/2007-10/OGF-GSM-SRM-v2.2.pdf
[33] Arcot Rajasekar et. al. Storage Resource Broker - Managing Distributed Data in a Grid, Computer Society of India Journal, Special Issue on SAN, Vol. 33, No. 4, pp. 42-54, Oct 2003.
[34] W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, L. Liming, S. Tuecke: GridFTP: Protocol Extension to FTP for the Grid, March 2001, http://www-fp.mcs.anl.gov/dsl/GridFTP-Protocol-RFC-Draft.pdf
[35] Open Grid Forum Data Access an Integration Services Working Group (DAIS), http://forge.ogf.org/sf/projects/dais-wg
[36] Mario Antonioletti et. al.: The design and implementation of Grid database services in OGSA-DAI, Concurrency and Computation: Practice and Experience, Volume 17, Issue 2-4, Pages 357 – 376, Special Issue: Grids and Web Services for e-Science, 2005 John Wiley & Sons, Ltd.
[37] P-GRADE portal Website, http://www.lpds.sztaki.hu/pgportal/
[38] T. Kiss, A. Tudose, P. Kacsuk, G. Terstyanszky: SRB Data Resources in Computational Grid Workflows, Conf. Proc. of the UK e-Science All Hands Meeting, 10 - 13 September 2007, Nottingham, UK, ISBN 978-0-9553988-3-4, pp 643-650.
[39] J. Novotny, R. Manansala, T. Nguyen: BIRN Portal Overview, Portals & Portlets 2006, 17-18 July 2006, Edinburgh, UK http://www.nesc.ac.uk/action/esi/download.cfm?index=3246
[40] J. Novotny, M. Russell, O. Wehrens: GridSphere: an advanced portal framework, Euromicro Conference, 2004. Proceedings. 30th Volume , Issue , 31 Aug.-3 Sept. 2004 pp: 412 - 419 DOI 10.1109/EURMIC.2004.1333397
[41] T. Delaitre, A.Goyeneche, T.Kiss, G.Z. Terstyanszky, N. Weingarten, P. Maselino, A. Gourgoulis, S.C. Winter: Traffic Simulation in P-Grade as a Grid Service, Conf. Proc. of the DAPSYS 2004 Conference, pp 129-136, ISBN 0-387-23094-7, September 19-22, 2004, Budapest, Hungary
[42] A. Kertész, I. Rodero and F. Guim: Data Model for Describing Grid Resource Broker Capabilities, CoreGRID Workshop on Grid Middleware in conjunction with ISC'07 conference, Dresden, Germany, June 25-26, 2007.
[43] A. Kertész, G. Sipos, P. Kacsuk: Multi-Grid Brokering with the P-GRADE Portal, In Proceedings of the Austrian Grid Symposium (AGS'06), OCG Verlag, Austria, 2006.
[44] TR-Grid portal Website: http://portal.grid.org.tr:8080/gridsphere/gridsphere
[45] Paul Watson: Databases in Grid Applications: Locality and Distribution, In Proceedings of the Database: Enterprise, Skills and Innovation. 22nd British National Conference on Databases, BNCOD 22, Sunderland, UK, July 5-7, 2005. Lecture Notes in Computer Science Volume 3567 pp. 1-16, Springer-Verlag 2005, ISBN 3-540-26973-8
[46] P. Kacsuk, Z. Farkas, G. Hermann: Workflow-level parameter study support for production Grids, ICCSA'2007, (proc. in Springer LNCS), Kuala Lumpur, 2007, ISSN: 0302-9743 (Print) 1611-3349 (Online) ISBN 3-00-011592-7, DOI: 10.1007/978-3-540-74484-9_74