Solving the Grid Interoperability Problem by P-GRADE Portal at Workflow Level

Peter Kacsuk\textsuperscript{1}, Tamas Kiss\textsuperscript{2}, Gergely Sipos\textsuperscript{1},

\textsuperscript{1}MTA SZTAKI, 1111 Kende utca 13
Budapest, Hungary
e-mail: \{kacsuk, sipos\}@sztaki.hu

\textsuperscript{2}Centre for Parallel Computing, University of Westminster
115 New Cavendish Street, London, W1W 6UW
e-mail: kisst@wmin.ac.uk

Abstract—Grid interoperability has recently become a major issue at Grid forums. Most of the current ideas try to solve the problem at the middleware level where unfortunately too many components (information system, broker, etc.) should be made interoperable. As an alternative concept the P-GRADE portal is the first Grid portal that tries to solve the problem at the level of workflows. It means that the components of a workflow can be executed simultaneously in several Grids. In this way the user can exploit more parallelism than inside one Grid. More than that the workflow level completely hides the low level Grid details for the end-user who does not have to learn the low level Grid commands of different Grids. In this way porting workflow application between different Grids can be done with minimal user efforts. The paper describes those features and techniques that are provided and used by the P-GRADE portal to solve the Grid interoperability problem.

I. INTRODUCTION

There have been several attempts to make existing production Grids and Grid technologies interoperable. A well known example is the work carried out in the framework of the GRIP European project to make Globus and Unicore interoperable [1]. Recently a new EU project called as the OMII Europe has been launched to solve interoperability between GT4 [2], gLite [3] and Unicore [4] at several levels including job submission, security and portal levels. However, the portal level interoperability only means that Gridsphere [5] is going to be ported for all the mentioned Grid middlewares. Another example of trying to solve interoperability at the job submission level is the new Condor version [6] that is able to submit jobs to different GT versions, Unicore and NorduGrid [7].

In the current paper we show that interoperability can be solved in a much higher level, namely at the workflow level that could be part of a Grid portal. Indeed, P-GRADE Grid portal [8] is the first Grid portal that tries to solve the interoperability problem at the workflow level with great success. It means that the components of a workflow can be executed simultaneously in several Grids. In this way the user can exploit more parallelism than inside one Grid. More than that the workflow level completely hides the low level Grid details for the end-user who has not to learn the low level Grid commands of different Grids. In this way porting workflow application between different Grids can be done with minimal user efforts. It also eliminates application porting work when a production Grid moves to a new Grid middleware. This happened, for example, in EGEE where the LCG middleware is changed to gLite. Users who developed their application under the P-GRADE portal at the workflow level can use their applications in the gLite based EGEE Grid as well without any modification of the application. Similar changes of technologies are foreseen for many production Grids including UK NGS [10], US OSG [14] and US TeraGrid [15]. These Grids currently use GT2 technology but in the near future they will move most probably to GT4. Since P-GRADE portal already supports both GT2 and GT4, users of these Grids will be lifted from tedious application porting activity if they use the workflow environment of P-GRADE portal.

Since second generation Grid middleware like GT2 supports job submission activities meanwhile third generation Grid middleware like GT4 provides service-oriented usage of the Grid, P-GRADE portal supports both job submission and service invocation inside the workflows. For the purpose of service invocation P-GRADE portal was integrated with the GEMLCA legacy code architecture [9] that was developed by the University of Westminster to turn legacy codes into Grid services with minimal user efforts without touching either the source or binary code of legacy applications.

In the paper we describe those features and techniques that are provided and used by the P-GRADE portal to solve the Grid interoperability problem. Section 2 will explain those features and techniques in P-GRADE portal that are provided for job submission. These include techniques for handling multiple Grid certificates, for connecting and tailoring the portal for several Grids, for
II. SUPPORTING FEATURES AND TECHNIQUES IN P-GRADE PORTAL FOR JOB SUBMISSION

The job submission interoperability is supported by several functions in P-GRADE portal:

- Grid setting management
- Certificate management
- Workflow definition and management
- Information system access

A Grid is typically identified by its information system and hence when the P-GRADE portal is installed, the portal administrator can connect the portal to several Grids defining their information system. The definition of an information system includes the type (MDS or LCG), the host URL where the information system service runs, its port identifier and BaseDn identifier. The next task of the portal administrator is to define the resources for each Grid that is connected to the portal. A resource is identified by its URL and local job manager.

Once the Grids are connected by identifying their information system the users can define the resources they would like to use inside the Grids. If the user wants to use the standard resources defined by the portal administrator, a simple click on the Load Default button in the “Settings” portlet window will load the user’s portal session with the URL and job manager of the resources as defined by the portal administrator. However, the user can tailor the resources to his own needs. He can remove unstable resources he does not like or can add other resources he has got private access to.

After defining the Grid environments the user should upload his Grid certificates belonging to the connected Grids into the MyProxy server that works in strong interaction with the portal. Of course, if several Grids accept the same certificate it is enough to upload their common certificate. Before the user would like to run a workflow in one or more Grids the corresponding certificates should be used to generate (download) certificate proxies for the required Grids. For uploading certificates and downloading certificate proxies the portal provides a special “Certificates” portlet window.

When the Grid certificate proxies are available they can be used to submit jobs into the corresponding Grids. The next issue here is how to define where a job component of a workflow should be executed. This can be done by the “Job Property” window in the Workflow Editor of the portal.

Once the workflow is defined and its components are allocated to different Grids it is the task of the portal workflow manager to control the execution of the workflow according to the user-specified Grid allocations. When the portal components are executed on several Grids the portal workflow manager takes care of submitting the jobs with the certificate proxy belonging to the allocated Grid. It is also the task of the workflow manager to take care of moving the input and output files among the workflow components even if they are processed in various Grids. In such case the portal server plays a relay station between the different Grids.

III. EXTENDING THE P-GRADE PORTAL TOWARDS SERVICE-ORIENTED GRIDS BY GEMLCA

GEMLCA (Grid Execution Management for Legacy Code Applications) represents a general architecture for deploying legacy applications as Grid services without re-engineering the code or even requiring access to the source files. GEMLCA adds an additional layer on top of a service-oriented Grid middleware, like Globus Toolkit version 4 (GT4). The GEMLCA layer is responsible to hide the legacy nature of the application. It communicates with the client through SOAP-XML messages, gets input parameter values, submits the legacy executable to a local job manager like Condor or PBS (Portable Batch System), and returns the results to the client in SOAP-XML format.

The deployment of a new legacy code service with GEMLCA means to expose the functionalities of this legacy application by describing the program’s execution environment and input/output parameters in an XML-based Legacy Code Interface Description (LCID) file. This file is used by the GEMLCA layer when communicating with the client and the local job manager.

GEMLCA provides the capability to convert legacy codes into Grid services. However, an end-user without specialist computing skills still requires a user-friendly Web interface (portal) to access the GEMLCA functionalities: deploy, execute and retrieve results from legacy applications. In order to achieve this aim, GEMLCA was integrated with the P-GRADE Grid portal. Following this integration, end-users can include legacy code services in their workflows, running on different GEMLCA Grid resources. The workflow manager of the portal contacts the selected GEMLCA resource and passes the actual parameter values of the legacy code to it. It is then the task of the GEMLCA resource to execute the legacy code with these actual parameter values and to
deliver the results back to the portal. The portal was also extended with a GEMLCA administration portlet that automates the creation of the LCID file and only requires the user to fill in a Web form with environmental and input/output parameter attributes.

As it was described in section II, the P-GRADE portal enables access to resources in Globus 2 based Grids. However, the integration of GEMLCA into the portal extended the GT2-based P-GRADE portal towards service oriented Grids. Besides traditional GT2 job submission, users can also utilise GT4 GEMLCA resources within the same workflow, bridging GT2 and GT4 based Grids. These GT2/GT4 components can communicate by sending input/output files to each other and their different nature at middleware level is transparent from the user’s point of view. However, as on one hand the GT2 job is submitted with its executable to a remote site, the GT4 GEMLCA job is accessed as a service through SOAP-XML service invocation.

Service-oriented Grid middleware could be the future, however, current production Grid systems, like the UK National Grid Service (NGS) or the Open Science Grid (OSG) [14], are still based on more robust and mature second generation Grid technologies, such as GT2. Users of these production Grids may wish to form virtual organisations and create code repositories that can be shared by the different members of these VOs. Code published in these repositories can then be easily accessed by authorised users and included in workflow applications using custom input parameter values. In order to fulfil this objective a modified version of GEMLCA, the GEMLCA Repository (GEMLCA-R) was created. GEMLCA-R enables the owner of the legacy code to publish the application in a central repository. This publication process is identical to the deployment of GEMLCA legacy codes and requires the creation of the XML-based LCID file. This process, as it was described earlier, is also supported from the GEMLCA administration portlet. After publication, other authorised users can browse this repository when creating a workflow, and include any of the published codes in their workflow applications. The repository is implemented in GT4 but submits the legacy code as a job to GT2 based production Grids.

Figure 1 illustrates how GEMLCA and GEMLCA-R extend the functionality of the portal and provide workflow level interoperability between GT2 and GT4 based Grids. As it is shown on the figure, jobs within a workflow can be traditional GT2 jobs (1) where the executable is coming form the user. This concept is extended by GEMLCA and GEMLCA-R. In case of GEMLCA-R the execution is still happening through job submission, however, in this case the executable is stored in and selected from the GEMLCA Repository (2). The third option is to include a legacy code service in the workflow (3). In this case the legacy code is deployed on the GT4 Grid site as a GEMLCA legacy code and accessed through service invocation.

![Figure 1. GEMLCA and GEMLCA-R extending the capabilities of the P-GRADE portal](image)

IV. CONNECTING GRID GENERATIONS AND TECHNOLOGIES WITH THE P-GRADE PORTAL AND GEMLCA

Sections II and III described those features of the P-GRADE portal and GEMLCA that enable the submission of workflows into multiple Grids based on different underlying technologies and middleware. Here we present those experiments and demonstrations that were specifically designed to illustrate how the previously described features can be combined in order to make different Grid solutions interoperable at the workflow level. System administrators of existing Grids can immediately utilise these solutions in order to extend the capabilities of their infrastructure without compromising its current reliable operation.

A. Connecting Second Generation Grids

Most of the current production Grid systems are based on second generation Grid technology. The basis of the underlying middleware is in most cases the Globus Toolkit, however because of substantial variations and modifications, these Grids are not naturally interoperable with each other. As the P-GRADE portal supports access to multiple Grids at the same time, and as it also supports both LCG and Globus-based Grids, the portal can be utilised to connect these Grid systems and map the execution of different workflow components to different resources in different Grids. As the portal is also integrated with GEMLCA-R, users can not only submit their own executable to Grid resources but can also browse the GEMLCA repository and select executables published by other users.

The experiment presented on figure 2 illustrates how different jobs of a workflow can be mapped to resources within the LCG based EGEE production Grid and the
GT2 based UK NGS. If a user has access to both of these Grids, using the same or different certificates, jobs of a workflow can be mapped in different ways. The user can submit his executable to any of the Grids (1 and 2). In case of LCG-based Grids the LCG broker can also be used besides direct mapping. Executables can also be selected from the GEMLCA repository and submitted (3 and 4), currently only by direct mapping, to any Grid sites. Any P-GRADE GEMLCA portal installation is capable to support the above functionalities. Portal administrators only have to define both Grids with their default resources, and users have to assign the appropriate certificate to each Grid before submitting workflows. As there are currently many users in Europe with access rights to both of these Grids, they can easily utilise the combined power of the resources.

Figure 2. Connecting GT2 and LCG-based Grids

B. Extending Second Generation Grids with Third Generation Resources

Most of the production Grids are based on second generation middleware at the moment. However, most of them are considering the transition to service oriented middleware, like GT4 or g-Lite. The P-GRADE GEMLCA portal is capable to seamlessly assist this transition. GT4 GEMLCA resources can be added to GT2 or LCG based Grids without any modification of the underlying infrastructure. In this case the GT4 resource is becoming an integral part of the original Grid. Users can map workflow components to these resources using the proxy certificate accepted by that particular Grid, without being aware of the differences of the underlying layers. Figure 3 illustrates, that a GT2 GEMLCA resource, set up at University of Westminster, is added at workflow level to the GT2-based UK NGS. Users can still utilise GT2-based job submission either directly (1) or from GEMLCA-R (2). However, they can also invoke GEMLCA legacy code services (3) through SOAP-XML service invocation but in a user transparent way. This GT4 GEMLCA resource has already been working at production level and available for every NGS user since February 2006.

Figure 3. Extending GT2 Grids with GT4 resources

C. Connecting Second and Third Generation Grids

As The P-GRADE portal is capable to connect to multiple Grids, and as through GEMLCA it also supports service-oriented Grid systems, it became possible to connect separate GT2 and GT4 based Grids from the same portal at workflow level. In order to demonstrate this concept the UK NGS was connected to a GT4 testbed, called the Westfocus Grid, comprising clusters at Westminster and Brunel Universities. The GT2 and GT4 Grids were defined as separate Grids in separate administrative domains that could potentially accept different user certificates. As it is illustrated on figure 4 jobs within a workflow can be submitted to GT2 resources within the NGS (1 and 2), or could be services deployed in the Westfocus Grid (3 and 4).

Figure 4. Connecting GT2 and GT4 Grids

D. Traffic Simulation Workflow Bridging Grid Generations and Middleware

In order to demonstrate the described capabilities of the P-GRADE GEMLCA portal, a workflow analysing urban car traffic was created [16] (see figure 5). The
workflows consists of three different applications. Manhattan (job0) is a road network generator that generates input for traffic simulators, like MadCity. MadCity (jobs 1 and 2) is a discrete-time traffic simulator that simulates traffic on a road network and indicates how the traffic density is changing on different roads and junctions. Finally, a comparator component (job 3) inputs the results of several simulations and creates a graph showing how the traffic density in time depends on several input parameters. In the presented example, jobs of the workflow were mapped to 3 different Grids based on three different underlying middleware. The Manhattan generator (job0) was running in Poznan on the LCG-based EGEE Grid submitted directly there as a job. The first simulator (job1) was a GEMLCA GT4 legacy code service that has been deployed at Westminster University within the Westfocus Grid. The second simulator (job2) was a legacy code submitted from the GEMLCA repository to the GT2 NGS site at Manchester. Finally, the comparator component was a GT2 job submitted directly to Oxford within the NGS. Besides the different underlying architectures the Grids were also using different proxy certificates. The execution graph in figure 5 illustrates that the workflow completed successfully demonstrating the workflow level interoperability of this very diverse infrastructure.

Figure 5. Traffic Simulation Workflow Running in 3 Different Grids

V. CONCLUSION AND FURTHER WORK

We have shown in the paper that P-GRcade portal extended with GEMLCA technology can serve as a bridge between different production Grids that are based on different Grid middleware technologies. The obvious advantages of such a bridge are the following:

1. end users can access any of these Grids from their workflow
2. virtualization, resource sharing and collaboration can be realized through the boundaries of different production Grids
3. porting the applications between production Grids does not require any porting efforts.

The P-GRcade portal serves SEEGRID (South-East European Grid) [13], the Central European VO of EGEE (VOCE), HunGrid (the Hungarian VO of EGEE), EGRID (the economics Grid VO of EGEE). The P-GRcade GEMLCA portal is already used as a service for the UK NGS (GT2) and for the Westfocus Grid (GT4). We plan in the near future to extend the EGEE portals with the GEMLCA service and to connect P-GRcade portal to NorduGrid, too.

REFERENCES

[12] The WestFocus Gridalliance web page, 