Integration of the gUSE/WS-PGRADE and InSilicoLab portals with DIRAC

A. Puig Navarro¹, D. Viana Casals¹, R. Graciani Díaz¹, A. Casajús Ramo¹, Á. Balaskó², P. Kacksuk², J. Kocot³, D. Harezlak²

¹ ICC, University of Barcelona
² MTA SZTAKI Laboratory of Parallel and Distributed Systems
³ Institute of Computer Science AGH, ACC Cyfronet AGH

E-mail: apuig@ecm.ub.edu

Abstract. The gUSE (Grid User Support Environment) framework allows to create, store and distribute application workflows. This workflow architecture includes a wide variety of payload execution operations, such as loops, conditional execution of jobs and combination of output. These complex multi-job workflows can easily be created and modified by application developers through the WS-PGRADE portal. The portal also allows end users to download and use existing workflows, as well as executing them. The DIRAC framework for distributed computing, a complete Grid solution for a community of users needing access to distributed computing resources, has been integrated into the gUSE/WS-PGRADE system. This integration allows the execution of gUSE workflows in a distributed computing environment, thus greatly expanding the capability of the portal to several Grids and Cloud Computing facilities. The main features and possibilities of the gUSE/WS-PGRADE-DIRAC system, as well as the benefits for users, will be outlined and discussed.

Introduction

Many different users and communities require the access to different kinds of computing resources to fulfill their scientific goals, including local clusters, grids and clouds. Frameworks like gUSE/WS-PGRADE [1] and InSilicoLab [2] provide the means of creating, publishing and maintaining web portals which simplify the configuration and submission of complicated job workflows and applications. By using them, users are protected from much of the inherent complexity of job configuration for submitting to the wide range of resources at their disposal.

The DIRAC middleware [3] has been integrated within these portals, giving the users a two-fold benefit: on one side, the integration provides transparent access to the wide range of computing resources needed by the different communities, while on the other it allows these users to keep the simplified—and maybe customized—interfaces of the web portals, which hide much of the complexity of the job submission and configuration process.

This paper is organized as follows: the gUSE/WS-PGRADE and InSilicoLab projects are described in Section 1 and their integration with DIRAC is detailed in Sections 2 and 3. Prospects for future developments are outlined in Section 4, and the conclusions for this work are drawn in Section 5.
1. The gUSE/WS-PGRADE and InSilicoLab web portals

Two portals have been considered as good candidates for integration with the DIRAC middleware: gUSE/WS-PGRADE and InSilicoLab.

On one hand, the grid User Support Environment (gUSE) is a grid virtualization environment which allows easy access of user communities to different Grid resources, providing a very flexible workflow management system for complex jobs. The portal interface for gUSE is called WS-PGRADE and it is shown in Fig. 1. It includes a graphical workflow manager that allows to edit (see Fig. 2), configure and publish workflows. In addition, gUSE allows the creation of application-specific portals by adding portlets to the WS-PGRADE portal and hiding the underlying workflow framework within the application-specific portlet. Examples of this capability are the CancerGrid [4] and ProSim [5] web portals.

On the other hand, InSilicoLab is an application portal designed to ease the use of computational software on Grids; all complexity is hidden from the user, as shown in Fig. 3, and therefore the applications can be run on Grids without technical knowledge of how they operate. It allows for a high customization of the Grid management environment and it allows to tailor the user interface to the needs of each user community. Born to support several chemistry applications, InSilicoLab currently supports Physics collaborations such as CTA [6].

2. Integration of DIRAC and gUSE/WS-PGRADE

As proof-of-concept, DIRAC has been integrated into the Java-based gUSE/WS-PGRADE web portal. However, DIRAC lacks a Java-based API and, at the time of starting this work, interaction with it was only possible through its Python API or through its command line scripts. Therefore, this proof-of-concept integration has been achieved with a wrapper (in the form of a new Java-based WS-PGRADE submitter) around the DIRAC command line scripts. In order to have a functional WS-PGRADE submitter component, the following methods have been provided:

- actionJobSubmit, used for the submission of the jobs configured via the web interface,
Figure 2. Workflow editor of the gUSE/WS-PGRADE portal.

Figure 3. InSilicoLab application portal for the CTA Collaboration.
Table 1. Mapping of the DIRAC job status to the gUSE/WS-PGRADE portal job status.

| DIRAC status | gUSE status |
|--------------|-------------|
| Received     | Submitted   |
| Waiting      | Waiting     |
| Staging      | Waiting     |
| Scheduled    | Scheduled   |
| Running      | Running     |
| Completed    | Finished    |
| Done         | Finished    |
| Failed       | Error       |
| Stalled      | Error       |
| Killed       | Error       |

wraps the `dirac-wms-job-submit` command. In this method, a JDL file is built from the portal job configuration, and the result is submitted using a DIRAC command line call.

- **actionJobAbort**, which is used to cancel a submitted job, wraps the `dirac-wms-job-delete` command and it is used to remove jobs from the system.

- **actionJobOutput** allows to obtain the output sandbox of the job by making use of the `dirac-wms-job-get-output` command.

- **actionJobStatus** monitors the job status within DIRAC with the `dirac-wms-job-status` command. The different DIRAC statuses are mapped to the portal statuses, as shown in Table 1, allowing the portals to keep their own job status systems.

Once the Java submitter class has been integrated with the DIRAC command line scripts, DIRAC has been added to the list of submitters of a test version of the gUSE/WS-PGRADE portal and test jobs have been sent.

3. Integration with InSilicoLab

Cyfronet, one of the largest Polish supercomputing and networking centres [7], has also carried out a number of tasks which led to a preliminary integration of a DIRAC job submission client in the InSilicoLab web portal. For the reasons presented in the previous section, currently the integration is based on a series of DIRAC command line wrappers which provide basic job submission functionality and user proxy generation.

The wrappers use a common DIRAC client installation and any multiuser requirements, which are typical for web portals, are handled by appropriately setting their execution environment and arguments, e.g. user proxy location or job output directory. The implementation uses Apache Commons Exec library for managing the wrapping processes and a set of regular expressions for parsing the output, and results so far have been good for the entire job life-cycle.

4. Prospects

To expand the integration possibilities, the proof-of-concept implementation, based on wrapping command line scripts, needs to be substituted by a more robust solution that allows a more general and flexible interaction.

To this end, DIRAC provides a new language agnostic API (reference) that can be used by any third party software. This new API has been designed following the RESTful principles, so any language with libraries to issue standard HTTP queries may use it. It allows clients to use the OAuth authorization standard for delegating the user credentials to a third party solution,
such as website or an application; these delegated credentials can be used by the third party software to query to DIRAC on behalf of the users. In addition, GSI proxies can still be used to authenticate.

This new API will further expand the possibilities communities have to integrate DIRAC into their distributed computing models. Following this idea, a new implementation, based on this new API, of the integration solution discussed in the previous sections is currently under development for the gUSE/WS-PGRADE and InSilicoLab portals; it is expected that it will provide a better interoperability between the services, as well as more robustness and better error handling capabilities.

5. Conclusions
A proof-of-concept of the integration between DIRAC and Java-based portals has been developed in the case of the gUSE/WS-PGRADE and InSilicoLab portals by wrapping the DIRAC command line. This integration allows communities using this portal to access their custom workflow management tools coupled with the DIRAC resource management capabilities, and opens the way for other portals to achieve a similar integration with DIRAC.

However, the current implementation lacks flexibility and ease of use due to the lack of a native Java API for DIRAC. The DIRAC RESTful interface will provide a tighter integration with respect to this proof-of-concept and will allow to easily expand this integrated solution to other portals —and, in consequence, to more communities. DIRAC will become the first interware solution to provide such interface for interoperability with any third party software for job submission, integrating them with any possible distributed computing resource that the user has access to.

Acknowledgments
The presented work has been financed by Comisión Interministerial de Ciencia y Tecnología (CICYT) (project FPA2010-21885-C02-01 and CPAN CSD2007-00042 from Programa Consolider-Ingenio 2010), and by Generalitat de Catalunya (AGAUR 2009SGR01268).

References
[1] grid User Support Environment http://www.guse.hu/
[2] InSilicoLab http://insilicolab.grid.cyfronet.pl/
[3] Tsaregorodtsev A et al. 2003 Computing in High-Energy Physics and Nuclear Physics 2003
[4] CancerGrid http://cancergrid.eu.w3h.hu/
[5] ProSim Project https://sites.google.com/a/staff.westminster.ac.uk/engage/Home/
[6] Actis M et al. (CTA Consortium) 2011 Exper.Astron. 32 193–316 (Preprint 1008.3703)
[7] Academic Computer Centre CYFRONET AGH https://www.cyfronet.pl/