

Portal de acceso a recursos HPC en entornos Grid

Accessing to HPC Resource and Job Monitoring Information Using the HPC-Europa Portal

◆ F. Guim, I. Rodero, J. Corbalán y J. Labarta

Resumen

En un entorno Grid formado por diversos centros computacionales con diversas tecnologías e infraestructuras involucradas, resulta muy útil disponer de sistemas para proporcionar un único punto de acceso a los recursos de forma sencilla y transparente. Dentro del proyecto HPC-Europa se ha desarrollado un sistema de acceso a diversos centros europeos con tradición en computación de altas prestaciones basado en un portal web. En este texto se presentan las principales características de este sistema y se describe una demostración efectuada de su funcionamiento realizada durante la tercera reunión de la Red Temática en Grid Middleware.

Palabras clave: portales web, grid, computación de altas prestaciones.

Summary

We present the infrastructure that we have designed and implemented in the HPC-Europa European project that allows uniform access to job monitoring information from different virtual organizations. The presented system abstracts user to the complexities of the underlying systems of each middleware.

The architecture and technologies used in the portal are described. We show all the features that user can use in the portal to personalize his/her monitoring environment: how and which information has to be presented.

Keywords: Grid computing, job monitoring, Grid portals, HPC-Europa

1. Introducción

Los sistemas Grid están formados por un conjunto de recursos heterogéneos que proporcionan distintas funcionalidades, como por ejemplo capacidades de computación o almacenamiento de datos. El acceso a estos recursos se realiza a través de un *middleware*, que es el componente del Grid que facilita la infraestructura necesaria para poder interactuar con dichos recursos. Normalmente para realizar la gestión de los recursos se acostumbra a utilizar *broker*, ya que es una tarea que puede resultar demasiado compleja para los usuarios finales. Estos componentes reciben peticiones de los usuarios o aplicaciones que quieren acceder a recursos que tienen bajo su dominio, y en caso de ser necesario, interactúan con el *middleware* para llevar a cabo las acciones solicitadas. Actualmente hay disponibles diversas iniciativas de brokers (por ejemplo: eNANOS [11], GridWay [6], Nimrod-G [1], etc.) y de *middleware* (por ejemplo: Globus [4], Unicore [12], Condor [9], etc.). La manera de interactuar con cada uno de los *brokers* suele ser distinta, esto implica que si un usuario o aplicación quiere acceder a centros que están gestionados por distintos tipos de *brokers*, se tiene que conocer los distintos mecanismos de acceso y sus correspondientes API.

Dada la situación descrita, para facilitar el acceso a diversos sitios controlados por diferentes *brokers* se hace impracticable exigir que el usuario aprenda los distintos métodos de acceso. El SPA (*Single Point of Access* - Punto Único de Acceso) presentado persigue solucionar este problema. La idea fundamental es proporcionar una interfaz única para todos los *brokers* y *middleware* que están disponibles por debajo, facilitando al usuario la interacción con todos los recursos de altas prestaciones que hay disponibles en cada uno de los centros computacionales.

El SPA está enmarcado en el proyecto europeo HPC-Europa [7], concretamente dentro del JRA2. En dicho proyecto participan seis centros europeos, cada uno de ellos dispone de su propio *broker* y

◆
Dentro del proyecto HPC-Europa se ha desarrollado un sistema de acceso a diversos centros europeos con tradición en computación de altas prestaciones basado en un portal web

◆
La idea es proporcionar una interfaz única para todos los *brokers* y *middleware* disponibles por debajo, facilitando al usuario la interacción con todos los recursos disponibles



cierto *middleware*. Como se puede ver en la figura 1, facilita el acceso a cinco tipos distintos de brokers que trabajan con cuatro tipos distintos de *middleware*.

El acceso al SPA se realiza a través de un portal web desarrollado sobre la infraestructura de GridSphere [5]. Actualmente el portal proporciona las siguientes funcionalidades:

- Definición y ejecución de trabajos en los distintos centros disponibles.
- Monitorización de los trabajos enviados.
- Gestión de *accounting* de los usuarios del SPA.
- Gestión de los ficheros que el usuario tiene distribuidos en los distintos centros.

◆
Para definir las características y requerimientos de los trabajos se utiliza el lenguaje JSDL



◆
El "Job Submission Portlet" es el encargado de facilitar al usuario la interfaz web para definir las características de los trabajos y enviarlos al centro deseado

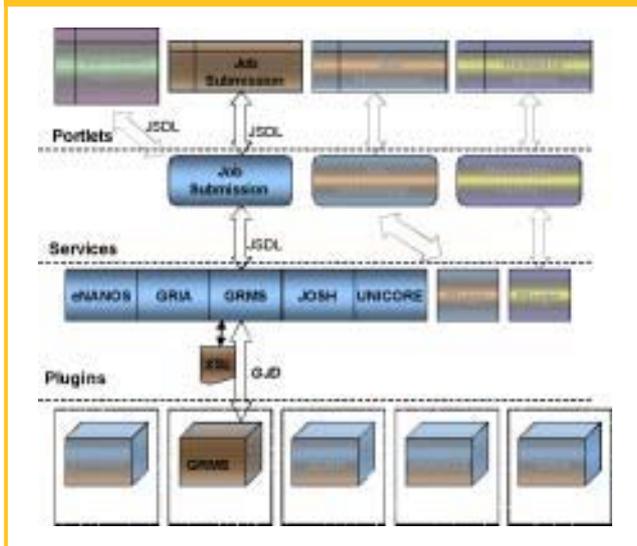
Para definir las características y requerimientos de los trabajos se utiliza el lenguaje JSDL [2] propuesto como estándar por un grupo de investigación del GGF [3]. Este lenguaje está basado en XML y trata de englobar todos los atributos necesarios para definir un trabajo y requerimientos que acostumbran a utilizar los diferentes brokers. Como es posible que ciertos sistemas no soporten todos los atributos que se contemplan en el esquema del JSDL, el portlet de "Job Submission" da soporte a cualquier subconjunto de atributos que sea capaz de gestionar el broker del sistema subyacente.

En la figura 2 se muestra la estructura interna del SPA que está dividida en cuatro capas. La primera está compuesta por los *portlet*s que son los encargados de facilitar a los usuarios la interacción con el SPA. Los *portlet*s facilitan las interfaces web para cada una de las funcionalidades presentadas anteriormente, y también el procesamiento de las acciones que pueden realizar los usuarios. Por ejemplo, el "Job Submission Portlet" es el encargado de facilitar al usuario la interfaz web para definir las características de los trabajos y enviarlos al centro deseado.

Cuando el usuario realiza una acción determinada en el portal mediante un *portlet*, se procesa la información que el usuario ha facilitado y, en caso de necesitar acceder a las capas inferiores, se invoca al servicio correspondiente que llevará a cabo la operación. Por ejemplo, si el usuario ha rellenado el formulario para enviar un trabajo y presiona el botón de envío, el portlet de "Job Submission" recogerá toda la información e invocará al servicio de "Job Submission Service" para que procese dicha acción.

Para realizar la integración de los diferentes *brokers* en el SPA, se disponen de unos "plugins" que implementan las funcionalidades básicas que requieren los servicios del portal. Es necesario

FIGURA 2: ESTRUCTURA DEL SPA



un *plugin* para cada tipo *broker* y es éste el encargado de invocar a las operaciones del *broker* correspondiente. Como cada uno dispone de su propio lenguaje de descripción de trabajos y sus propios mecanismos de acceso, el *plugin* será el encargado de transformar el documento JSDL, que describe un trabajo en el portal, al lenguaje propio de cada entorno de ejecución. Por ejemplo, en el caso de GRMS se realiza una transformación del documento JSDL a un documento GJD (GRMS Job Description) mediante una hoja de estilo XSL, o en el caso de eNANOS se utiliza un "gateway" intermedio para realizar ciertas transformaciones y acceder a

las funcionalidades del eNANOS Resource Broker.

2. Descripción de la demostración realizada

Durante la demostración realizada se ha mostrado cómo el usuario interactúa con el SPA. Para poder empezar a trabajar en el portal un usuario debe autenticarse: primero tiene que autenticarse en el portal Web de GridSphere a través de la página principal del mismo, y posteriormente tiene que autenticarse como usuario de Grid a través de la ventana de configuración del portlet de "Job Submission". La gestión de la seguridad dentro del entorno del SPA (y en consecuencia en los diversos centros) se realiza mediante certificados X.509. En la ventana de configuración el usuario debe especificar la dirección del repositorio de credenciales (un servidor de MyProxy [10]), y el nombre de usuario y contraseña correspondiente a su credencial.

Una vez que el usuario ya se ha validado, no tiene que autenticarse más para acceder a cualquiera de los centros, el propio SPA gestionará la seguridad de las conexiones con cada uno de los centros, así como la transferencia de ficheros, etc. Este sistema facilita la tarea al usuario, ya que con una sola validación es capaz de acceder a todos los recursos que hay debajo de la infraestructura del SPA, independientemente del centro u organización virtual a la cual pertenezca.

Durante la demostración se ha visto cómo el usuario puede enviar trabajos a los distintos centros de forma uniforme y homogénea a través del portlet de "Job Submission". Hemos podido definir el trabajo con los formularios del portlet y lanzarlo a un centro, por ejemplo al BSC. También hemos podido definir los requerimientos del trabajo, tanto *software* como *hardware* y los ficheros que han de ser transferidos antes y después de su ejecución. Se ha mostrado cómo se pueden importar las definiciones de trabajos a través de ficheros XML que se encuentran en el mismo ordenador que se utiliza para acceder al portal y que incluyen un documento JSDL.

Para empezar a trabajar, un usuario debe autenticarse: primero en el portal Web de GridSphere y posteriormente como usuario de Grid a través de la ventana de configuración del portlet de "Job Submission"

El usuario puede enviar trabajos a los distintos centros de forma uniforme y homogénea a través del portlet de "Job Submission"



El usuario puede personalizar las vistas del *portlet* de monitorización

Este modelo de único punto de acceso (SPA) para entornos Grid puede ser útil dentro del contexto de IRISGrid

A continuación hemos procedido a enviar el mismo trabajo en repetidas ocasiones y se ha mostrado cómo el usuario puede monitorizar la evolución de estas ejecuciones a través del *portlet* de "Job Monitoring". Para cada trabajo se visualiza tanto un conjunto de campos fijos, por ejemplo el nombre del trabajo enviado, como otros campos que dependen del estado en que se encuentre, por ejemplo el tiempo que lleva ejecutándose o el tiempo esperado de finalización si está ejecutándose. También hemos visto cómo se pueden llevar a cabo acciones sobre los trabajos que están ejecutándose, durante la demostración se han cancelado un subconjunto de los trabajos enviados.

En la última parte de la demostración se ha mostrado cómo el usuario puede personalizar las vistas del *portlet* de monitorización. El usuario puede cargar perfiles predefinidos, desde el básico que sólo visualiza la información elemental (es decir, los trabajos que están ejecutándose, finalizados o cancelados) hasta el perfil más avanzado que muestra absolutamente toda la información disponible en el sistema. Por otro lado, el usuario también puede realizar una configuración más personalizada, el *portlet* permite definir el tipo de trabajos que quiere visualizar y especificar qué campos quiere mostrar de todos los disponibles para cada uno de los estados. Finalmente, se ha visto cómo el usuario puede especificar el tiempo de refresco de la página principal de monitorización y así seguir la ejecución de los trabajos en la escala temporal deseada.

3. Conclusiones

En esta demostración se ha presentado un entorno de trabajo para plataformas basadas en Grid que trata de proporcionar un acceso homogéneo y transparente a recursos computacionales de altas prestaciones mediante un portal web. Se ha visto cómo uno de los principales objetivos de esta infraestructura es facilitar sustancialmente las tareas al usuario, proporcionando una interfaz sencilla y fácilmente accesible.

En la demostración se han presentado las funcionalidades básicas que proporciona el portal para ejecutar y gestionar trabajos; se han ejecutado trabajos reales mediante el portal, se han monitorizado y se ha actuado sobre ellos.

El sistema presentado es altamente extensible, para añadir un nuevo centro o entorno de ejecución, solamente basta con añadir un nuevo *plugin* que proporcione acceso al *broker* correspondiente y hacer algunos pequeños cambios de configuración. También es posible ampliar o incorporar nuevas funcionalidades al portal, ya que está implementado mediante *portlets* que pueden ser gestionados individualmente.

Este modelo de único punto de acceso (SPA) para entornos Grid puede ser útil dentro del contexto de IRISGrid [8]. La implementación de un SPA a nivel estatal podría facilitar acceso de forma sencilla, a recursos de altas prestaciones, a todos los grupos de investigación que lo requieran.

Francesc Guim, Ivan Rodero
(fguim@ac.upc.edu) (irodero@ac.upc.edu)
Julita Corbalán, Jesús Labarta
(juli@ac.upc.edu), (jesus@ac.upc.edu)
Barcelona Supercomputing Center (BSC-CNS)
UPC

Referencias

- [1] D. Abramson, R. Buyya, J. Giddy. *A Computational Economy for Grid Computing and its Implementation in the Nimrod-G Resource Broker*. Future Generation Computer Systems. 18(8), 2002
- [2] A. Anjomshoaa, F. Brisard, M. Drescher, D. Fellows, A. Ly, S. McGough, D. Pulsipher, A. Sawa (Ed). *Job Submission Description Language (JSDL) Specification, Version 1.0*. 15 June 2005.
- [3] JSDL Working Group Web Site. <https://forge.gridforum.org/projects/jsdl-wg/>
- [4] Globus Project Web Site. www.globus.org
- [5] GridSphere Web Site. www.gridsphere.org
- [6] GridWay Web Site. <http://asds.dacya.ucm.es/GridWay/>
- [7] HPC-Europa Web Site. www.hpc-europa.org
- [8] IRISGrid Web Site. www.irisgrid.es/
- [9] M. Litzkow, M. Livny, M. Mutka. *Condor – A Hunter of Idle Workstations*. Proc. 8th International Conference of Distributed Computing Systems (ICDCS 1988). IEEE CS Press. San Jose, CA, USA. January 1998.
- [10] MyProxy Credencial Management Service Web Site. <http://grid.ncsa.uiuc.edu/myproxy/>
- [11] I. Rodero, J. Corbalán, R.M. Badia, J. Labarta. *eNANOS Grid Resource Broker*. P.M.A. Sloot et al. (Eds.): EGC 2005, LNCS 3470, Amsterdam, June 2005, pp. 111-121
- [12] Unicore Web Site. www.unicore.org