

# Capítulo 4

## Implementación y Evaluación

### 4.1. Descripción de la infraestructura

La figura muestra de forma esquemática los elementos que integran el ambiente de grid bajo el cual se instaló y configuró el portal OGCE versión 2.5:

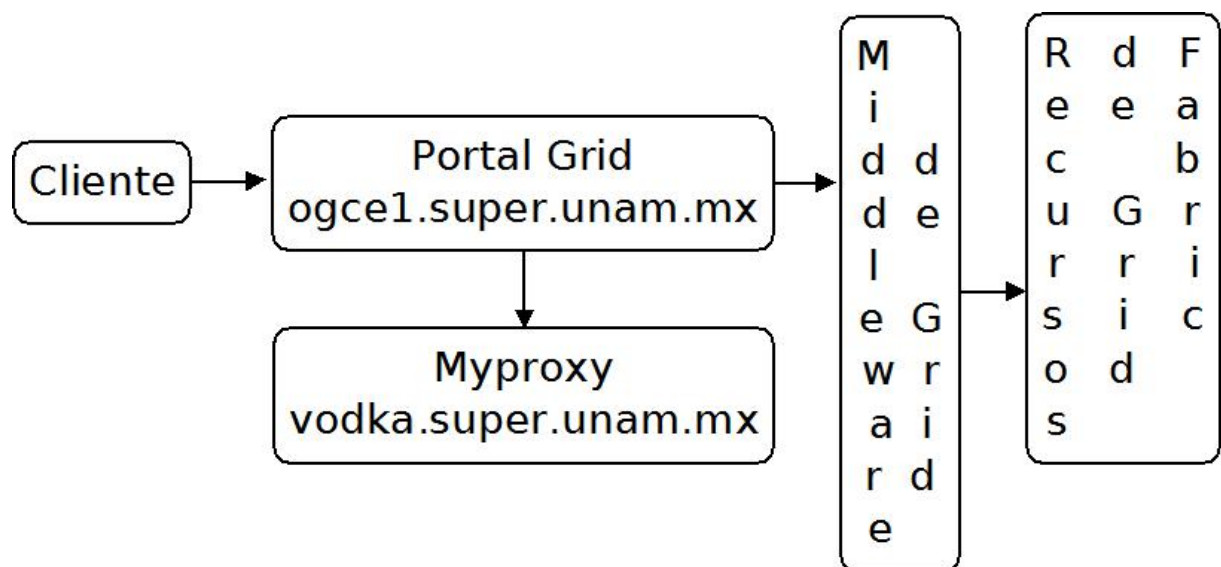


Figura 4.1: Elementos con los que interactúa un portal grid

El portal grid OGCE versión 2.5 se encuentra instalado en una máquina virtual del Departamento de Supercómputo ogce1.super.unam.mx, el sistema operativo de la

máquina virtual es Fedora 8, cuenta con un procesador Intel Xeon a 1.8 GHz y tiene de memoria RAM 512 MB.

La parte del middleware de grid correspondiente al servidor de manejo de certificados Myproxy fue instalado en la máquina `vodka.super.unam.mx`, cuenta con sistema operativo Ubuntu 8, tiene un procesador Intel Pentium 4 a 2.26 GHz y tiene de memoria RAM 756 MB.

Los restantes partes del middleware de Grid se instalaron en un nodo de login de la Supercomputadora y la versión del middleware es GT5. Este nodo fue dado de alta en el DNS con el nombre de `balam-ln2.super.unam.mx` con el fin de poder acceder a este nodo mediante los servicios del middleware. Los cuales son:

- `gridftp`
- `gram5`

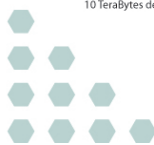
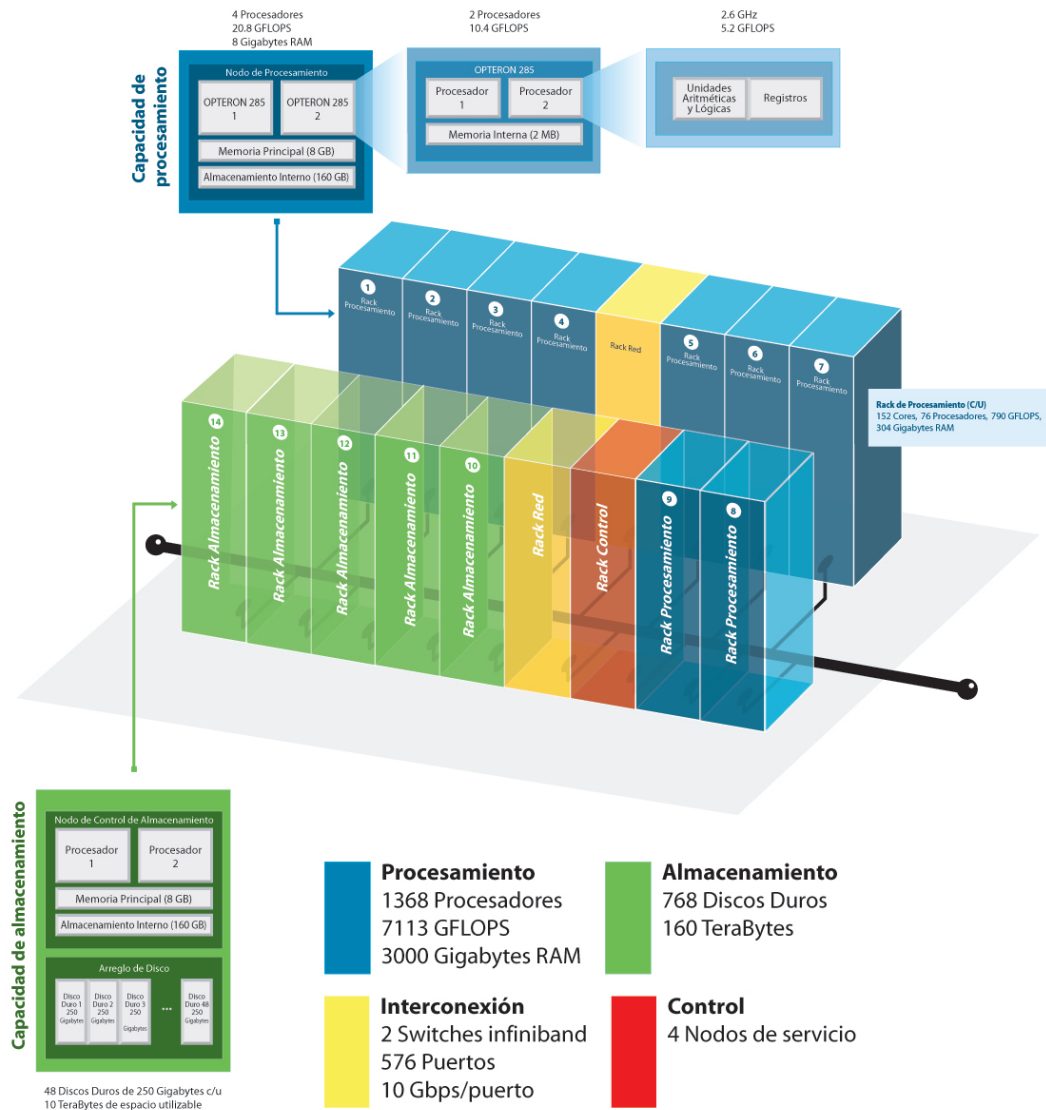
Los recursos de *fabric*, son los nodos de cálculo de la Supercomputadora Kanbalam. Los nodos de cálculo son 337 nodos de cálculo regulares, cada nodo con dos procesadores AMD Opteron Dual Core con 8 GB de RAM y 5 nodos de cálculo especiales con dos procesadores AMD Opteron Dual Core por nodo y 64 GB de RAM. El manejo de estos recursos se lleva a cabo mediante el software de administración de recursos LSF.



Figura 4.2: Arquitectura de un cluster LSF



# Procesamiento y memoria de KanBalam



## 4.2. Configuración

### 4.2.1. Configuración del Servidor de MyProxy

A continuación se muestran los pasos para la instalación del servidor de Myproxy en la máquina `vodka.super.unam.mx`:

1. Como root, crear un usuario llamado globus  
% adduser globus
2. Crear un directorio, en el cual se instalará GT5.  
% mkdir /usr/local/globus-5.0/
3. Cambiar de dueño y de grupo al anterior directorio.  
% chown globus:globus /usr/local/globus-5.0/
4. Como globus, agregar la variable de ambiente GLOBUS\_LOCATION  
% export GLOBUS\_LOCATION=/usr/local/globus-5.0/
5. Configurar y compilar myproxy:  
% ./configure --prefix=\$GLOBUS\_LOCATION  
% make gsi-myproxy
6. Agregar el contenido de \$GLOBUS\_LOCATION/share/myproxy/etc.services en /etc/services
7. Copiar el archivo \$GLOBUS\_LOCATION/share/myproxy/etc.xinetd.myproxy a /etc/xinet.d/myproxy y modificar las variables de ambiente  
server = /usr/local/globus-5.0/sbin/myproxy-server env = GLOBUS\_LOCATION=/usr/local/globus-5.0 LD\_LIBRARY\_PATH=/usr/local/globus-5.0/lib
8. Como root, reiniciar el demonio de xinetd  
% kill -HUP pid\_de\_xinetd

## 4.3. Configuración del Portal Grid

### 4.3.1. Instalación de Java

Para el portal Grid, es necesario contar con una versión de Java superior o igual a la 1.5.

En nuestra implementación se instaló Java 1.6.6 [22], debido a que no se encontró ningún paquete que configurará la última versión del Ambiente de Ejecución de de Java (*Java Runtime Environment*, JRE) de Sun y mantuviera al mismo tiempo los JREs que proporciona el Sistema Operativo.

1. Instalación del *Java Development Kit*, JDK.
  - a) Entrar a la página de Sun:  
`http://java.sun.com/` o de `http://www.oracle.com/technetwork/java/index.html`
  - b) Dar click en Java SE en la sección de Descargas de Software, *Software Downloads*
  - c) Dar click en *Previous Releases*
  - d) Dar click en *Archived Releases*
  - e) De la sección *Java 2 Platform Standard Edition (J2SE)*, elegir de las opciones de *JDK/JRE - 6* la actualización 6 (*6 Update 6*) y dar click en *Go*.
  - f) Dar click en *Download JDK*
  - g) Seleccionar en Plataforma Linux, aceptar la licencia y dar click en continuar.
  - h) Dar click en `jdk-6u6-linux-i586-rpm.bin` y guardarlo en Disco Duro.
  - i) Agregar permisos de ejecución al archivo descargado.  
`chmod +x jdk-6u6-linux-i586-rpm.bin`
  - j) Como root, ejecutar el archivo  
`./jdk-6u6-linux-i586-rpm.bin`
2. El paso anterior instalará el JDK y el JRE pero sólo configurará adecuadamente al JDK; para configurar adecuadamente el JRE, es necesario descargar el paquete `jpackage`[23]:
  - a) Entrar a la siguiente dirección:  
`ftp://jpackage.hmdc.harvard.edu/JPackage/1.7/generic/RPMS.non-free/`

- b) Descargar el paquete compatible con la versión de Java instalada. En nuestro caso, se descargó la versión 1.6.6:  
`java-1.6.0-sun-compat-1.6.0.06-1jpp.i586.rpm`
- c) Instalar el paquete jpackage[24]  
`rpm -Uvh java-1.6.0-sun-compat-1.6.0.06-1jpp.i586.rpm`

### 3. Configuración del JRE

Esta configuración funciona en distribuciones de Linux basadas en RPMs (Mandrake, Red Hat, SuSE, etc).

- a) Configurar el JRE con el que deseamos trabajar, con el siguiente comando:  
`alternatives --config java`

```
[root@ogce3 ~]# alternatives --config java
```

```
There are 3 programs which provide 'java'.
```

Selection	Command
++ 1	/usr/lib/jvm/jre-1.7.0-icedtea/bin/java
2	/usr/lib/jvm/jre-1.5.0-gcj/bin/java
3	/usr/lib/jvm/jre-1.6.0-sun/bin/java

- b) Seleccionar el número que corresponda a la versión 1.6. En nuestro caso, tecleamos 3.

### 4.3.2. Descarga del Portal OGCE

1. Entrar a la página de OGCE  
`http://www.collab-ogce.org/ogce/index.php/Main_Page`
2. Dar click en *OGCE Portal* en la sección de componentes descargables.
3. Ir a la sección *Portal Toolkit 2.5 Download* y descargar el archivo compactado y comprimido:  
`ogce-portal-only-release-2.5.tar.gz`

### 4.3.3. Instalación del Portal OGCE

1. Descomprimir y descompactar el archivo que contiene el portal en el directorio \$HOME de una cuenta dedicada al servicio del portal exclusivamente, en nuestro caso la cuenta dedicada se llama irving.

```
% tar -zxvf ogce-portal-only-2.5-release.tar.gz
```

2. Hacer una liga al directorio anteriormente descomprimido.

```
% ln -s ogce-portal-only-2.5-release ogce-portal-only
```

3. Cambiarse al directorio ogce-portal-only y descompactar el archivo maven-2.0.7-bin.tar.gz.

```
% tar -zxvf maven-2.0.7-bin.tar.gz
```

4. Agregar las siguientes variables de ambiente. Para hacer que esta configuración sea permanente en una cuenta de usuario dedicada, agregar las siguientes líneas al archivo \$HOME/.bashrc

a) export JAVA\_HOME=/usr/lib/jvm/jre-1.6.0-sun

b) PATH=\$HOME/ogce-portal-only/maven-2.0.7/bin:\$PATH

La primer línea actualiza la variable de ambiente de Java.

La siguiente línea agrega al PATH los programas que utiliza el administrador de proyectos Apache Maven.

5. Editar las propiedades (*properties*) del archivo pom.xml

a) Cambiar <portal.server.ip/> por la dirección IP en que se va a encontrar el portal grid.

```
<portal.server.ip>132.248.202.135</portal.server.ip>
```

6. Para instalar el portal, ejecutar el siguiente comando:

```
mvn clean install
```

7. Finalizado el proceso de compilación del portal, iniciar el servidor Apache Tomcat mediante el comando:

```
./startup-tomcat.sh
```

8. Accesar a la página web del portal:  
`http://132.248.202.135:8080/gridsphere`
9. Y crear la cuenta de administrador del portal introduciendo la información solicitada por el portal como nombre de usuario, correo electrónico, contraseña, etc.

#### 4.3.4. Configuración del Firewall

En caso de que el portal esté detrás de un firewall, realizar el siguiente procedimiento:

1. Como root, editar el archivo  
`/etc/sysconfig/iptables`
2. Agregar la siguiente línea, antes de cualquier otra que contenga un *REJECT*  
`-A RH-Firewall-1-INPUT -m state --state NEW -p tcp -m tcp --dport 8080 -j ACCEPT`
3. Actualizar la nueva regla del Firewall  
`/etc/init.d/iptables restart`

#### 4.3.5. Configuración del Portlet de Administración de Archivos

1. Estando en el directorio donde se instaló el portal "ogce-portal-only", parar el servidor Tomcat:  
`./shutdown-tomcat.sh`
2. Editar el archivo "pom.xml"
  - a) Agregar los hosts "balam-ln2.supercomputo.unam.mx,vodka.super.unam.mx" entre las etiquetas  
`<gridftp.host.names>` y `</gridftp.host.names>`
3. Reconstruir el portlet de Administración de Archivos, *File Manager*, ejecutando el siguiente comando:  
`mvn clean install -f portlets/file-manager/pom.xml`



4. Reiniciar el servidor Apache Tomcat:

```
./startup-tomcat.sh
```

### 4.3.6. Configuración del Portlet de Envío de Jobs

La configuración del Portlet de Envío de Jobs, `gp-job-submission` es similar a los pasos 1 y 4 de la sección 4.3.5

1. Editar el archivo "pom.xml"
  - a) Agregar el nuevo host "balam-ln2.supercomputo.unam.mx,vodka.super.unam.mx" entre las etiquetas `<gram.host.names>` y `</gram.host.names>`
2. Reconstruir el portlet, ejecutando el siguiente comando:

```
mvn clean install -f portlets/gp-job-submission/pom.xml
```

Debido a que el portlet de Envío de Jobs no presentaba ciertas características del lenguaje RSL del middleware y que son particularmente útiles en el uso de la Supercomputadora, como los parámetros de Queue y JobType fue necesario modificar parte del código fuente de este portlet:

A continuación se presentan las modificaciones realizadas:

En el archivo `submit.jsp` del directorio `$HOME/ogce-portal-only/portlets/gp-job-submission/src/main/webapp/jsp` se agregó lo siguiente:

```
<%-- JOBTYPe --%>
<label for="jobtype">JobType</label>
<input id="jobtype" name="jobtype" type="input"
      class="portlet-form-input-field gp-form-input-field"/>
<br/>

<%-- QUEUE --%>
<label for="queue">Queue</label>
<input id="queue" name="queue" type="input"
      class="portlet-form-input-field gp-form-input-field"/>
<br/>
```

---

## CAPÍTULO 4. IMPLEMENTACIÓN Y EVALUACIÓN

---

En la clase GRAMJobSubmissionPortlet.java del directorio \$HOME/ogce-portal-only/portlets/gp-job-submission/src/main/java/edu/tacc/gridport/portlets/interactive se añadió lo siguiente:

```
jbean.setJobtype(request.getPreferences().getValue(handle + ".jobtype", null));
jbean.setJobtype(request.getPreferences().getValue(handle + ".queue", null));

req.getPreferences().setValue(handle + ".jobtype", currentJobBean.getJobtype());
req.getPreferences().setValue(handle + ".queue", currentJobBean.getQueue());
```

En la clase JobBean.java del directorio \$HOME/ogce-portal-only/portlets/gp-job-submission/src/main/java/edu/tacc/gridport/portlets/interactive/beans se agregó:

```
private String jobtype;
private String queue;
/**
 *      * Get the jobtype associated with the job
 *          *
 *              * @return Returns the jobtype.
 *                  */
public String getJobtype() {
    return jobtype;
}
/**
 *      * Set jobtype to be associated with the job
 *          *
 *              * @param jobtype The jobtype to set.
 *                  */
public void setJobtype(String jobtype) {
    this.jobtype = jobtype;
}
/**
 * *      * Get the queue associated with the job
 * *          *
 * *              * @return Returns the queue.
 * *                  */
public String getQueue() {
    return queue;
}
```

```

    }
    /**
 * *      * Set queue to be associated with the job
 * *      *
 * *      * @param queue The queue to set.
 * *      */
    public void setQueue(String queue) {
        this.queue = queue;
    }

```

En la clase GRAMJobManager.java del directorio \$HOME/ogce-portal-only/portlets/gp-job-submission/src/main/java/edu/tacc/gridport/portlets/interactive/helpers se agregó lo siguiente:

```

        String jobtype = req.getParameter("jobtype");
        newJob.setJobtype(jobtype);
        String queue = req.getParameter("queue");
        newJob.setQueue(queue);
        if (job.getJobtype() != null && !(job.getJobtype()).equals("")) {
            rslSb.append("(jobType=\"" + job.getJobtype() + "\")");
        }
        if (job.getQueue() != null && !(job.getQueue()).equals("")) {
            rslSb.append("(queue=\"" + job.getQueue() + "\")");
        }
        System.out.println("El rsl es: " + rsl);

```

Adicionalmente este portlet generaba un RSL incorrecto porque una vez dados los archivos de error y salida mediante el portal grid, el RSL agregaba otras líneas referentes a los mismos archivos. Este comportamiento no permitía almacenar nada en los archivos de error y de salida. Por lo anterior fue necesario comentar algunas líneas de la clase GRAMJobPreWS.java ubicada en el mismo directorio de la clase (GRAMJobManager.java) anterior:

```

//newRSL.append("(stdout=x-gass-cache://$(GLOBUS_GRAM_JOB_CONTACT)
stdout anExtraTag)");
//newRSL.append("(stderr=x-gass-cache://$(GLOBUS_GRAM_JOB_CONTACT)
stderr anExtraTag)");

```

Finalmente, se recompiló este portlet como se mostró en 4.3.6

### 4.3.7. Configuración de la CA

Para configurar la CA a utilizar, se deben solicitar a la CA: su certificado y sus políticas de firmado (*CA Signing policy*).

Para la configuración se deben seguir los siguientes pasos:

1. Copiar la información al directorio `$HOME/.globus/certificates` de la cuenta dedicada a la implementación del portal.

```
% cp vodka.ca.tar.gz $HOME/.globus/certificates
```

2. Descomprimir la información

```
% cd $HOME/.globus/certificates
```

```
% tar -zxvf vodka.ca.tar.gz
```

### 4.3.8. Creación de cuentas

1. Con cuenta de administrador del portal, entrar al portlet de administración (*Administration*).
2. Ingresar a la sección de usuarios (*Users*).
3. Y dar click en Crear nuevo usuario (*Create new user*)
4. Llenar los campos solicitados: nombre de usuario, nombre, correo electrónico, organización, tipo de cuenta, contraseña en el portal.

The screenshot displays the OGCE (Open Grid Computing Environments) web interface. At the top left is the OGCE logo. On the top right, there is a 'Logout' link and a user greeting: 'Welcome, Irving Alvarez'. Below the logo is a navigation bar with tabs for 'Welcome', 'Administration', 'jobsubmit-portlet', 'proxymanagement-portlet', and 'gp-job-submission'. Underneath this is a secondary navigation bar with links for 'Portlets', 'Users', 'Groups', 'Roles', 'Layouts', and 'Messaging'. The main content area is titled 'User Account Manager' and contains a form for 'Edit User Information'. The form includes a warning: 'LEAVE PASSWORD FIELD BLANK TO KEEP EXISTING PASSWORD IF EDITING AN EXISTING USER'. The fields are: 'User Name', 'Full Name', 'Email Address', 'Organization', 'Role In GridSphere' (a dropdown menu currently set to 'USER'), 'Disable account?' (a checkbox), 'Password', and 'Confirm password'. At the bottom of the form are 'Save User' and 'Cancel' buttons.

Figura 4.3: Administración de cuentas

### 4.3.9. Configuración de cuentas de usuario

Una vez conectado al portal grid, aparece el portlet de bienvenida, en este portlet el usuario del portal podrá configurar su información personal, cambiar su contraseña y seleccionar los portlets a utilizar. Para seleccionar los portlets, ir a la sección de configurar membresía de grupo (*Configure group membership*) y marcar (dar un click) los siguientes portlets:

- file-manager
- gp-job-submission
- proxymanager-portlet

Y finalmente dar de alta los cambios, dar click en el botón de *Save*

The screenshot shows the 'Profile Manager' web interface. At the top, there are tabs for 'Welcome', 'Settings', and 'Layout'. The main content area is titled 'Edit Settings for' and includes the following fields:

- Last Login Time: Monday, December 6, 2010 5:34:17 PM PST
- User Name: icac
- E-Mail: irving@super.unam.mx
- Locale: English (with a dropdown menu showing options like America/Knox\_In, America/La\_Paz, America/Lima, America/Los\_Angeles, America/Louisville, America/Macelo, and America/Managua)
- Full Name: Irving Carlos Alvarez
- Organization: UNAM

Below these fields is a 'Save' button. To the right, there is a 'Configure group membership' section with a table:

Groups:	Group Description:	Role in Group
<input type="checkbox"/> condor-g	Condor-G Job Submission Portlet	USER
<input type="checkbox"/> condor-job-submission	Condor Job Submission Portlet	USER
<input checked="" type="checkbox"/> file-manager	Comprehensive File Management Portlet	USER
<input type="checkbox"/> gltac	Generic Service Toolkit	USER
<input checked="" type="checkbox"/> gp-job-submission	Batch Job Submission Portlet	USER
<input type="checkbox"/> gpir-portlet	GPIR TGUP Browser Portlet	USER
<input type="checkbox"/> ifema-portlet	IFEMA Portlet	USER

Below the table is another 'Save' button. To the left of the table is an 'Update password' section with fields for 'Enter original password:', 'Password:', and 'Confirm password:', followed by a 'Save' button.

Figura 4.4: Configuración de cuenta de usuario

### 4.3.10. Configuración del Portlet de Manejo de Certificados Proxy

1. Entrar al portlet de Servidor de Manejo de Certificados Proxy, *ProxyManager-Portlet*
2. Dar click en el botón de *Edit* que se encuentra en la parte superior izquierda del portlet.
3. Cambiar el valor del Hostname por "vodka.super.unam.mx"
4. Poner el nombre de usuario en el servidor de Manejo de Certificados y dar el tiempo de vida del proxy o aceptar el default que es de 2 horas.
5. Por último actualizar estos parámetros en el portal, dar click en *Customize*.

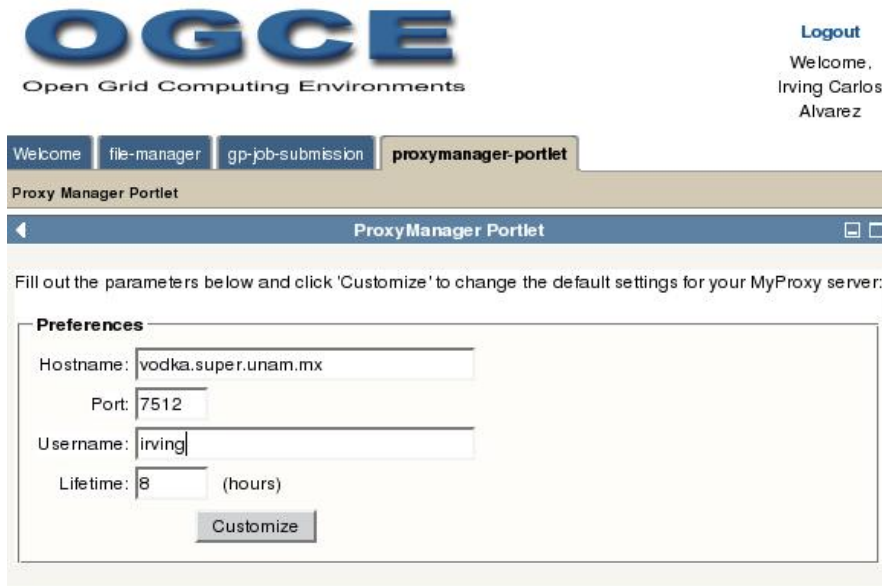


Figura 4.5: Configuración del Servidor de certificados

## 4.4. Pruebas de funcionamiento

### 4.4.1. Ingreso al Portal

Para ingresar al Portal Grid OGCE ingrese su nombre de usuario y contraseña.

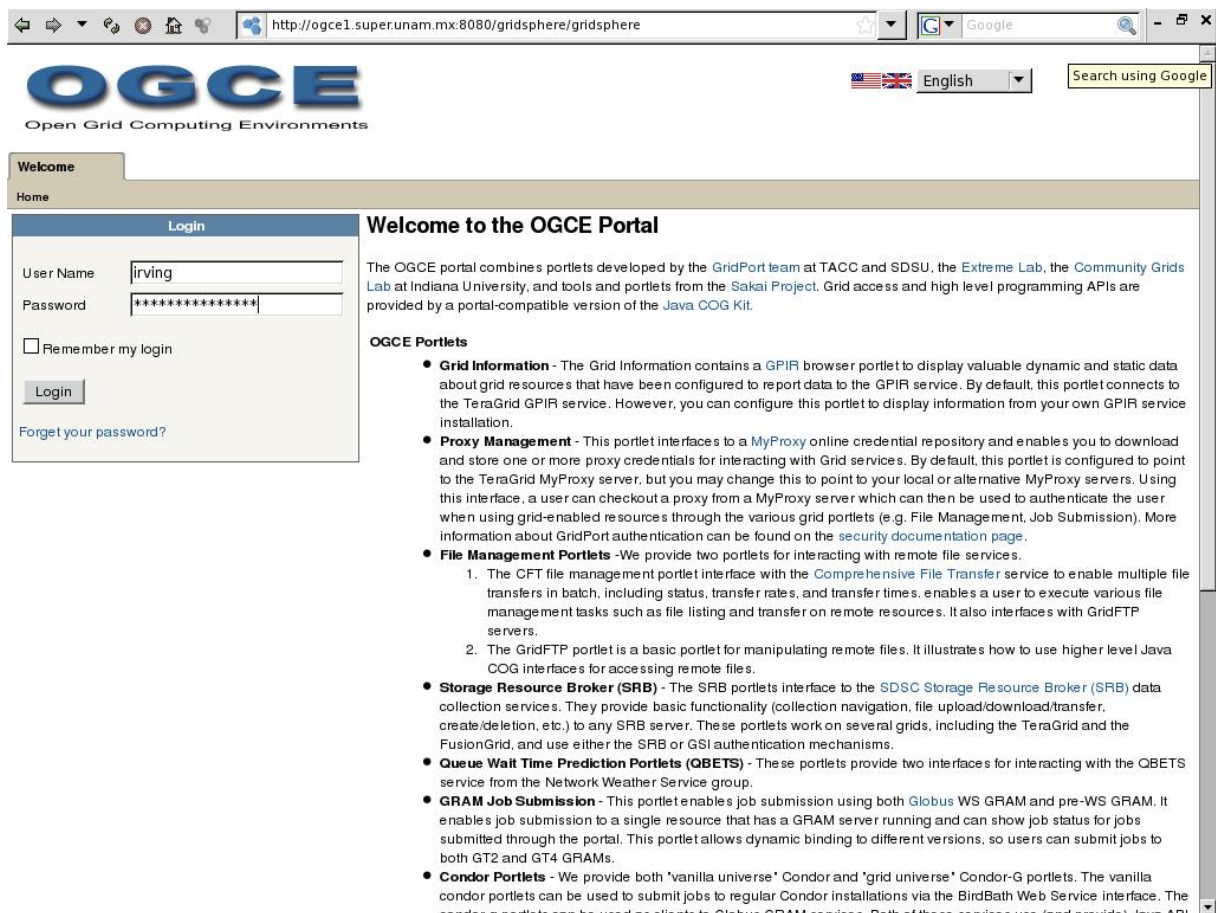


Figura 4.6: Pantalla de inicio del portal grid



### 4.4.2. Uso del Portlet Proxy–Manager

Seleccionar el portlet ProxyManager y dar click en el botón *Get new Proxy*

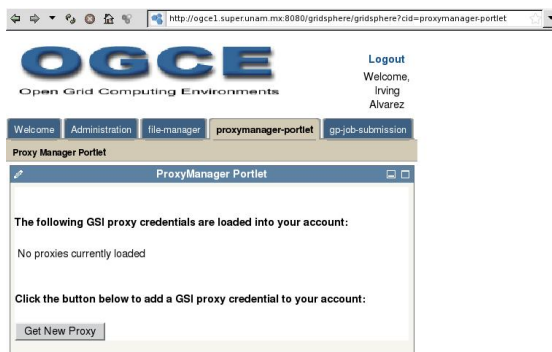


Figura 4.7: Portlet del servidor de certificados

Proporcionar la contraseña del certificado proxy y dar un click en el botón de *Get Proxy From MyProxy*.

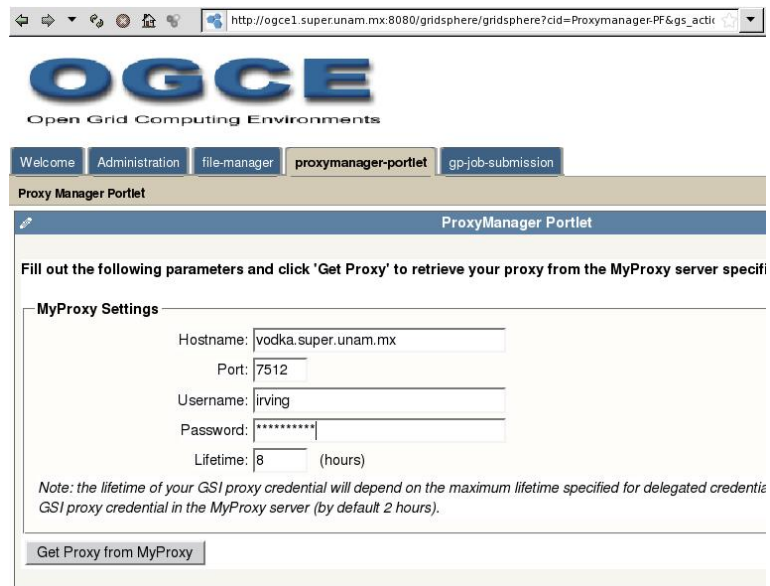


Figura 4.8: Solicitud de un certificado proxy

## CAPÍTULO 4. IMPLEMENTACIÓN Y EVALUACIÓN

Lo anterior, mostrará que se obtuvo un certificado proxy del usuario del portal grid.

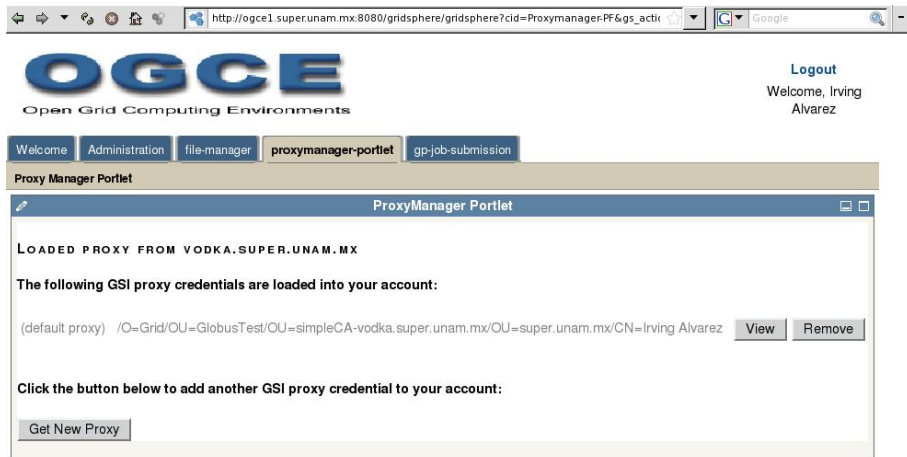


Figura 4.9: Obtención del certificado proxy vía el portal

Para ver el contenido del certificado proxy, dar click en *view*

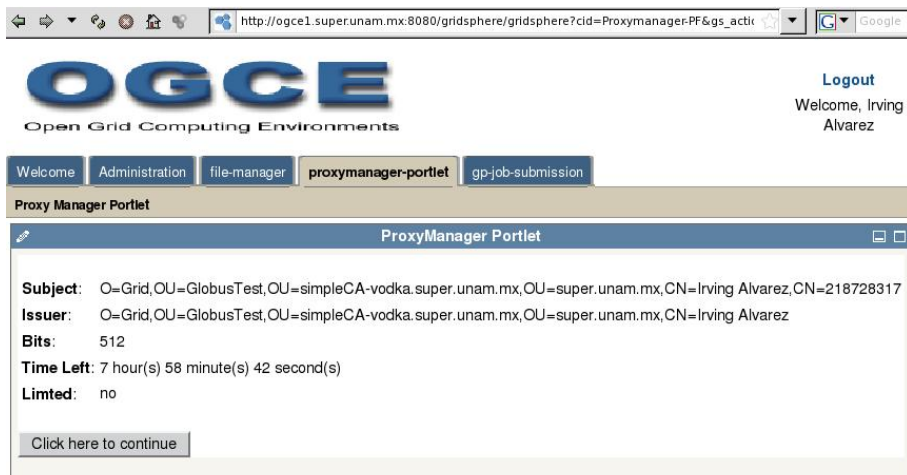


Figura 4.10: Información del certificado proxy

### 4.4.3. Uso del Portlet File–Manager

Seleccionar el Portlet File–Manager, este portlet tiene dos secciones una de listado de archivos de origen y otra de destino.

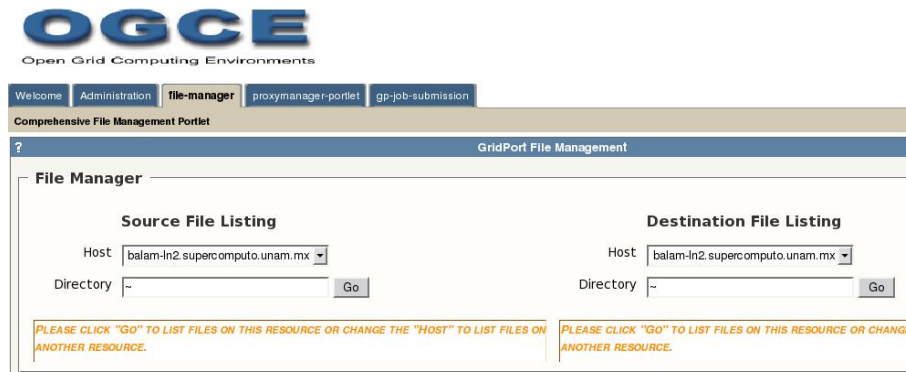


Figura 4.11: Portlet de manejo de archivos

Seleccionar en el listado de archivos de origen el host "balam-ln2.supercomputo.unam.mx" y despues dar click en *Go* para desplegar los archivos.

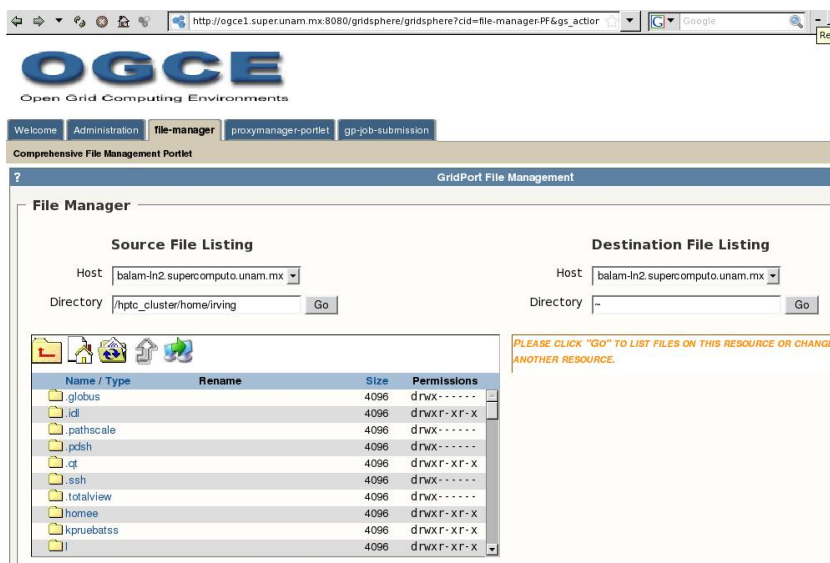


Figura 4.12: Listado de archivos de una máquina origen

Las acciones que se pueden realizar en ambas secciones son las siguientes:

- Ir un directorio atrás
- Ir al directorio \$HOME del usuario
- Actualizar el listado de archivos
- Subir archivos a un host origen o destino desde la máquina local del usuario.

Adicionalmente el host origen, puede transferir archivos hacia el host destino.

Dar click en el botón de Subir archivo (*Upload file*), seleccionar un archivo desde la máquina local y subir el archivo, dar click en Upload.

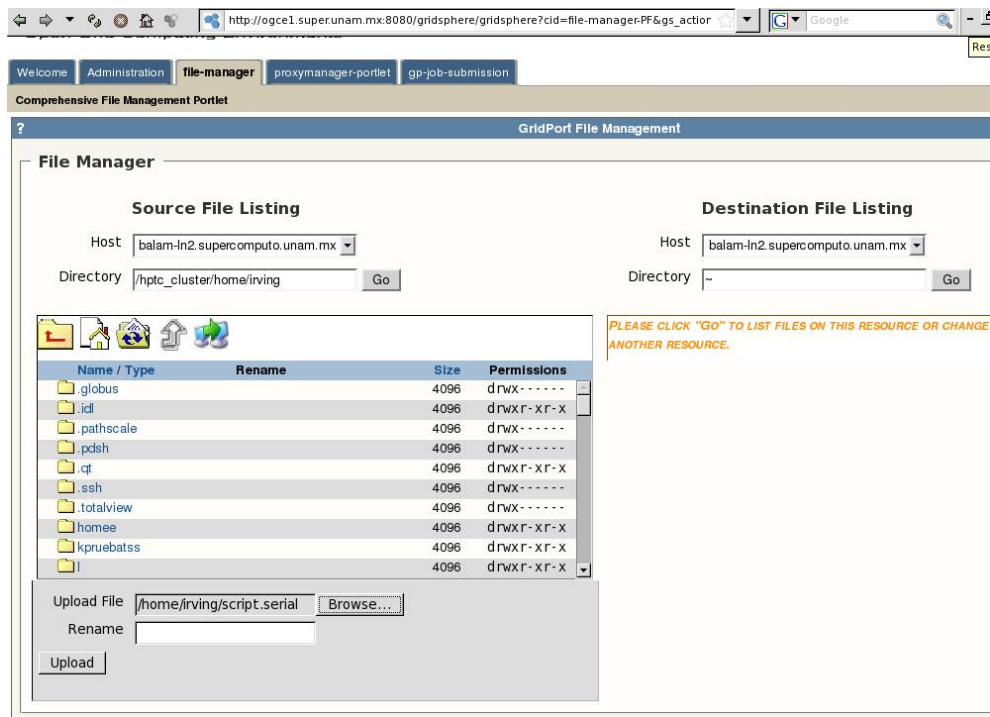


Figura 4.13: Transferencia de archivo desde una máquina local

## CAPÍTULO 4. IMPLEMENTACIÓN Y EVALUACIÓN

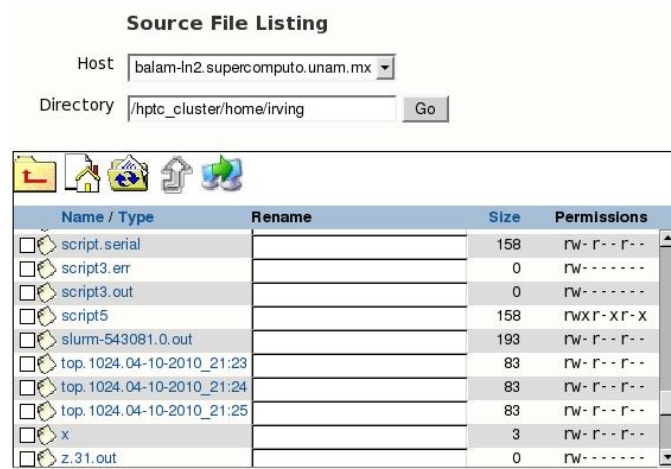


Figura 4.14: Archivo en la máquina origen

Seleccionar en el listado de archivos destino el host "vodka.super.unam.mx" y después dar click en *Go* para desplegar los archivos.

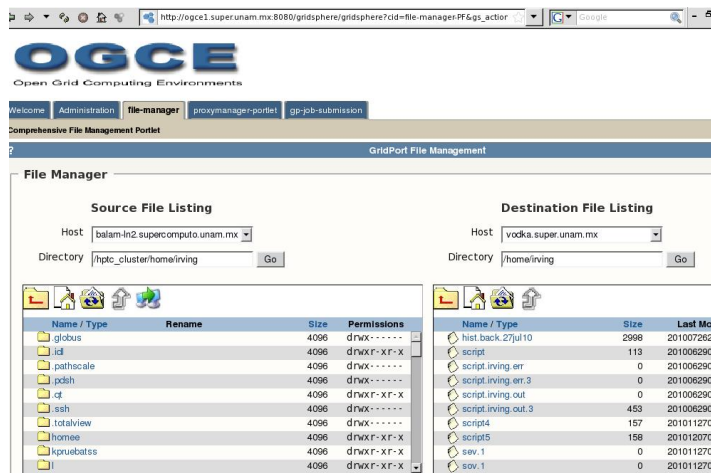


Figura 4.15: Listado de archivos en máquina destino

## CAPÍTULO 4. IMPLEMENTACIÓN Y EVALUACIÓN

Seleccionar el archivo "script.serial" en el listado de archivos origen y dar click en el botón Transferir archivos (*transfer files*).

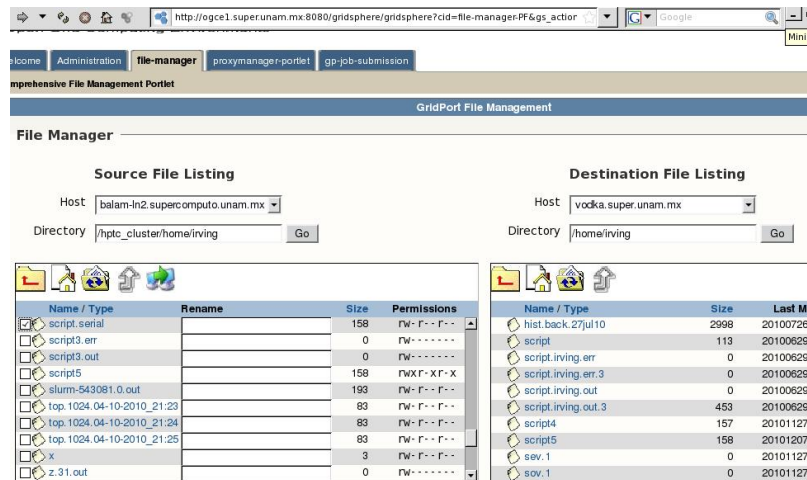


Figura 4.16: Transferencia de archivo desde máquina origen a máquina destino

A continuación se observa el archivo transferido en el host destino enviado de un host origen.

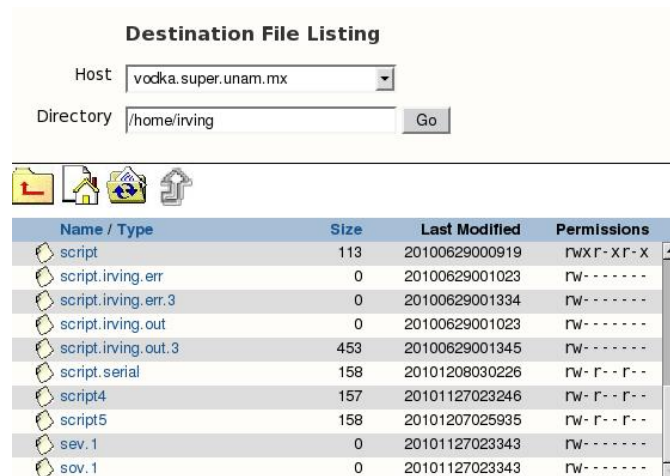


Figura 4.17: Archivo en máquina destino

#### 4.4.4. Uso de Portlet Gp-Job-Submission

Para poder hacer uso de este portlet, dar click en el portlet gp-job-submission. Para mostrar el funcionamiento de este portlet y su interacción con el equipo de supercómputo, ejecutaremos el siguiente programa:

```
#!/bin/sh
/bin/pwd
/usr/bin/whoami
/bin/hostname
/bin/date
if [ $# -eq 2 ]
then
    echo ‘/bin/sleep $1’
    /bin/sleep $1
    echo ‘wake up’
    /bin/date
    for ((i=1; i<=$2; i++))
    do
        echo $i
    done
else
    echo ‘Para ejectar este script introduzca dos numeros’
fi
```

Para ejecutar el programa, en el portlet de gp-job-submission, seleccionar el host `balam-ln2.supercomputo.unam.mx`, completar los parámetros necesarios y solicitar la ejecución del programa dando click en el botón *Submit*, como se muestra a continuación:

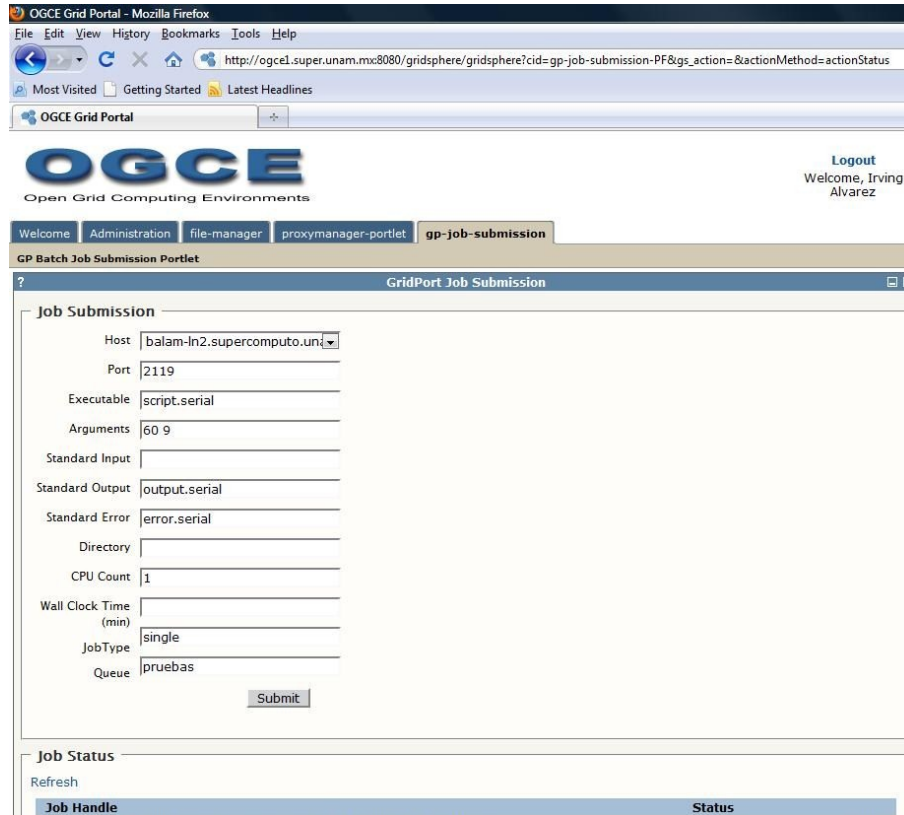


Figura 4.18: Portlet de Envío de Jobs

Una vez solicitada la ejecución del programa, el portal genera un RSL (*Resource Standard Language*), el cual contiene los parámetros de ejecución del programa en lenguaje del middleware de grid para su ejecución. A continuación se muestra el RSL generado por el portal:

```
El rsl es: &(executable=script.serial)(arguments=60 9)(count=1)
(stdout="output.serial")(stderr="error.serial")(jobType="single")
(queue="pruebas")
```

Este RSL es utilizado por el servidor GRAM para solicitar recursos de cálculo a la supercomputadora, esta petición se realiza al Sistema de Calendarización de jobs y Administración de Recursos LSF.



## CAPÍTULO 4. IMPLEMENTACIÓN Y EVALUACIÓN

El sistema LSF cuando dispone de los recursos de cómputo, despacha los recursos para la ejecución del programa. Esta asignación de recursos y las propiedades del job se puede verificar vía consola, como se aprecian a continuación:

```
[irving@n340 ~]$ bjobs -l

Job <113451>, User <irving>, Project <default>, Status <RUN>, Queue <pruebas>,
Command <#!/bin/sh;## LSF batch job script built by Glob
us Job Manager;##BSUB -q pruebas;#BSUB -i /dev/null;#BSUB
-o /hptc_cluster/home/irving/output.serial;#BSUB -e /hptc_
cluster/home/irving/error.serial;#BSUB -n 1;X509_USER_PROX
Y=''/hptc>
Thu Jan 13 08:01:28: Submitted from host <n339>, CWD <${HOME}>, Input File </dev/
null>, Output File </hptc_cluster/home/irving/output.seria
l>, Error File </hptc_cluster/home/irving/error.serial>, R
equested Resources <type=any>;
Thu Jan 13 08:01:32: Started on <lsfhost.localdomain>, Execution Home </hptc_cl
uster/home/irving>, Execution CWD </hptc_cluster/home/irvi
ng>;
Thu Jan 13 08:01:32: slurm_id=548513;ncpus=1;slurm_alloc=n314;

SCHEDULING PARAMETERS:
           r15s   r1m  r15m   ut       pg    io   ls    it    tmp    swp    mem
loadSched   -     -     -     -       -     -   -     -     -     -     -
loadStop    -     -     -     -       -     -   -     -     -     -     -

[irving@n340 ~]$
```

Para monitorear el estado del job mediante el portal grid, este portlet muestra en su parte inferior los jobs enviados a ejecución, su estado y las acciones que se pueden realizar a los mismos. Para ver el estado actual del job, dar click en en *refresh*. A continuación se muestran los estados por los que pasa el job:

<a href="https://balam-ln2.supercomputo.unam.mx:44393/16145717058644622496/1863408318102375595/">https://balam-ln2.supercomputo.unam.mx:44393/16145717058644622496/1863408318102375595/</a>	DONE	Delete
<a href="https://balam-ln2.supercomputo.unam.mx:44393/16145718159368047261/1863408318102399662/">https://balam-ln2.supercomputo.unam.mx:44393/16145718159368047261/1863408318102399662/</a>	PENDING	Cancel Delete

Figura 4.19: Estado pendiente del job

## CAPÍTULO 4. IMPLEMENTACIÓN Y EVALUACIÓN

---

<a href="https://balam-ln2.supercomputo.unam.mx:44393/16145717058644622496/1863408318102375595/">https://balam-ln2.supercomputo.unam.mx:44393/16145717058644622496/1863408318102375595/</a>	DONE	Delete
<a href="https://balam-ln2.supercomputo.unam.mx:44393/16145718159368047261/1863408318102399662/">https://balam-ln2.supercomputo.unam.mx:44393/16145718159368047261/1863408318102399662/</a>	ACTIVE	Cancel Delete

Figura 4.20: Estado activo del job

<a href="https://balam-ln2.supercomputo.unam.mx:44393/16145717058644622496/1863408318102375595/">https://balam-ln2.supercomputo.unam.mx:44393/16145717058644622496/1863408318102375595/</a>	DONE	Delete
<a href="https://balam-ln2.supercomputo.unam.mx:44393/16145718159368047261/1863408318102399662/">https://balam-ln2.supercomputo.unam.mx:44393/16145718159368047261/1863408318102399662/</a>	DONE	Delete

Figura 4.21: Estado finalizado del job

La salida del job es la siguiente:

```
Sender: LSF System <lsfadmin@lsfhost.localdomain>
Subject: Job 113451: <#! /bin/sh;## LSF batch job script built by Globus
Job Manager;#;
```

```
#BSUB -q pruebas;#BSUB -i /dev/null;
#BSUB -o /hptc_cluster/home/irving/output.serial;
#BSUB -e /hptc_cluster/home/irving/error.serial;
#BSUB -n 1;
```

```
X509_USER_PROXY='' /hptc> Done
```

```
Job <#! /bin/sh;## LSF batch job script built by Globus Job Manager;#;
#BSUB -q pruebas;#BSUB -i /dev/null;#BSUB -o /hptc_cluster/home/irving/
output.serial;#BSUB -e /hptc_cluster/home/irving/error.serial;#BSUB -n 1;
X509_USER_PROXY='' /hptc> was submitted from host <n339> by user <irving>.
Job was executed on host(s) <lsfhost.localdomain>, in queue <pruebas>,
as user <irving>.</hptc_cluster/home/irving> was used as the home directory.
</hptc_cluster/home/irving> was used as the working directory.
Started at Thu Jan 13 08:01:32 2011
Results reported at Thu Jan 13 08:02:47 2011
```

Your job looked like:

```
-----
# LSBATCH: User input
#! /bin/sh
```

```
#
# LSF batch job script built by Globus Job Manager
#

#BSUB -q pruebas
#BSUB -i /dev/null
#BSUB -o /hptc_cluster/home/irving/output.serial
#BSUB -e /hptc_cluster/home/irving/error.serial
#BSUB -n 1

X509_USER_PROXY='' /hptc_cluster/home/irving/.globus/job/balam-ln2.supercomputo.
unam.mx/16145718159368047261.1863408318102399662/x509_user_proxy''; export X50
9_USER_PROXY
GLOBUS_LOCATION='' /global/opt/appl/globus-5.0''; export GLOBUS_LOCATION
GLOBUS_GRAM_JOB_CONTACT='' https://balam-ln2.supercomputo.unam.mx:44393/16145718
159368047261/1863408318102399662/''; export GLOBUS_GRAM_JOB_CONTACT
HOME='' /hptc_cluster/home/irving''; export HOME
LOGNAME='' irving''; export LOGNAME
GLOBUS_GASS_CACHE_DEFAULT='' /hptc_cluster/home/irving/.globus/.gass_cache'';
export GLOBUS_GASS_CACHE_DEFAULT
export PATH=/opt/appl/scripts-grid:$PATH

#Change to directory requested by user
cd /hptc_cluster/home/irving
/hptc_cluster/home/irving/script.serial ''60'' ''9''
```

-----

Successfully completed.

Resource usage summary:

CPU time	:	0.12 sec.
Max Memory	:	5 MB
Max Swap	:	31 MB

The output (if any) follows:

```
/hptc_cluster/home/irving
irving
n314
Thu Jan 13 08:01:32 CST 2011
/bin/sleep 60
wake up
Thu Jan 13 08:02:32 CST 2011
1
2
3
4
5
6
7
8
9
```

PS:

Read file </hptc\_cluster/home/irving/error.serial> for stderr output of this job.

~~~~~

ESTADISTICAS DEL JOB

|                           |                |
|---------------------------|----------------|
| Duracion:                 | 1.25 minutos   |
| Recursos consumidos:      | 0.02 horas-cpu |
| CPU Utilizado:            | 0.00 minutos   |
| Eficiencia de uso de CPU: | 0.13 %         |

## 4.5. Pruebas de aplicaciones

A continuación se mostrará como utilizar varias aplicaciones mediante el uso del portal grid. Para hacer uso de estas aplicaciones fue necesario modificar un archivo de configuración de Globus Toolkit y crear scripts que carguen el ambiente que utilizan las aplicaciones, esto fue debido a que el middleware no proporciona un ambiente para cada aplicación o usuario específico.

### 4.5.1. Gaussian

Para el uso de la aplicación de gaussian, se hizo el programa o script "script.gaussian", el fin de este script es cargar el ambiente que necesita la aplicación Gaussian para su correcto funcionamiento. El contenido del script es el siguiente:

```
#!/bin/bash

export TMPU="/global/$LOGNAME"
export GAUSS_SCRDIR=$TMPU

export GAUSS_EXEDIR="/opt/appl/g09/bsd:/opt/appl/g09/private:/opt/appl/g09"
export GAUSS_ARCHDIR="/opt/appl/g09/arch"
export GMAIN="/opt/appl/g09/bsd:/opt/appl/g09/private:/opt/appl/g09"
export G03BASIS="/opt/appl/g09/basis"
export F_ERROPT1="271,271,2,1,2,2,2,2"
export PATH="/opt/appl/g09/bsd:/opt/appl/g09/private:/opt/appl/g09:$PATH"

g09 < "$@"
```

Y se modificó el archivo \$GLOBUS\_LOCATION/lib/perl/Globus/GRAM/JobManager/lsf.pm:

```
if (-f '/opt/appl/scripts-grid/' . $description->executable) {
    $description->add('executable',
    '/opt/appl/scripts-grid/' . $description->executable);
}
```

Esta modificación también fue necesaria para Nwchem (ver sección 4.5.2).

Para solicitar la ejecución de la aplicación se colocan los siguientes parámetros en el portal grid:

The screenshot shows a web browser window with the URL `http://ogce1.super.unam.mx:8080/gridsphere/gridsphere?cid=gp-job-submission`. The page title is "Job Submission". The form contains the following fields:

- Host: `balam-ln2.supercomputo.unam.mx`
- Port: `2119`
- Executable: `script.gaussian`
- Arguments: `pruebaOGCE/g09/test919.com`
- Standard Input: (empty)
- Standard Output: `pruebaOGCE/g09/t919.o`
- Standard Error: `pruebaOGCE/g09/t919.e`
- Directory: (empty)
- CPU Count: `4`
- Wall Clock Time (min): (empty)
- JobType: `single`
- Queue: `pruebas`

A "Submit" button is located at the bottom of the form.

Figura 4.22: Parámetros para la ejecución de un job de Gaussian

- En el campo *Executable* se proporciona el nombre del programa que carga el ambiente, en nuestro caso `script.gaussian`.
- En el campo *Arguments* se proporciona el archivo de entrada de gaussian.
- El campo *Standard output* y *Standar error* corresponden a los archivos de salida y de error estándar respectivamente incluyendo el directorio donde se colocarán.

- En el campo *CPU count* sólo se pueden proporcionar hasta 4 procesadores debido al tipo de licencia que se cuenta de esta aplicación.
- En el campo *Job type* se proporciona single.
- En el campo *Queue* se proporciona la cola en que se ejecutara la aplicación.

Y finalmente se da click en *Submit* para enviar el job.

### 4.5.2. Nwchem

El programa que carga el ambiente de nwchem y ejecuta a nwchem es el programa script.nwchem, el contenido de este programa es el siguiente:

```
#!/bin/bash

export PATH=/opt/hptc/bin:/opt/appl/bin:$PATH

/opt/hpmpi/bin/mpirun -srun nwchem "$@"
```

El archivo de entrada "input2.in" de esta aplicación es el siguiente:

```
title "formaldehyde ECP deck"
start ecpchho
geometry units au
  C      0.000000  0.000000 -1.025176
  O      0.000000  0.000000  1.280289
  H      0.000000  1.767475 -2.045628
  H      0.000000 -1.767475 -2.045628
end
basis
  C  SP
    0.1675097360D+02 -0.7812840500D-01  0.3088908800D-01
    0.2888377460D+01 -0.3741108860D+00  0.2645728130D+00
    0.6904575040D+00  0.1229059640D+01  0.8225024920D+00
  C  SP
    0.1813976910D+00  0.1000000000D+01  0.1000000000D+01
  C  D
```

CAPÍTULO 4. IMPLEMENTACIÓN Y EVALUACIÓN

---

```

0.8000000000D+00 0.1000000000D+01
C F
0.1000000000D+01 0.1000000000D+01
O SP
0.1842936330D+02 -0.1218775590D+00 0.5975796600D-01
0.4047420810D+01 -0.1962142380D+00 0.3267825930D+00
0.1093836980D+01 0.1156987900D+01 0.7484058930D+00
O SP
0.2906290230D+00 0.1000000000D+01 0.1000000000D+01
O D
0.8000000000D+00 0.1000000000D+01
O F
0.1100000000D+01 0.1000000000D+01
H S
0.1873113696D+02 0.3349460434D-01
0.2825394365D+01 0.2347269535D+00
0.6401216923D+00 0.8137573262D+00
H S
0.1612777588D+00 0.1000000000D+01
end
ecp
C nelec 2
C ul
      1      80.0000000      -1.60000000
      1      30.0000000      -0.40000000
      2       0.5498205      -0.03990210
C s
      0       0.7374760       0.63810832
      0     135.2354832     11.00916230
      2       8.5605569     20.13797020
C p
      2      10.6863587      -3.24684280
      2      23.4979897       0.78505765
O nelec 2
O ul
      1      80.0000000      -1.60000000
      1      30.0000000      -0.40000000
      2       1.0953760      -0.06623814
O s

```



```
0          0.9212952      0.39552179
0          28.6481971     2.51654843
2          9.3033500      17.04478500
0 p
2          52.3427019     27.97790770
2          30.7220233     -16.49630500
end
scf
  vectors input hcore
  maxiter 20
end
task scf
```

Para ejecutarlo desde el portal grid, se introdujeron los siguientes parámetros:

- En el campo *Executable* se proporciona el programa de script.nwchem.
- En el campo *Arguments* se proporciona el archivo de entrada de nwchem.
- El campo *CPU count* puede ser variable y depende de la cola seleccionada.

Y finalmente se da click en *Submit* para enviar el job.

The image shows a web browser window with the address bar containing the URL: `http://ogce1.super.unam.mx:8080/gridsphere/gridsphere?cid=gp-job-submission-PF&gs_`. The main content area is titled "Job Submission" and contains a form with the following fields and values:

|                       |                                |
|-----------------------|--------------------------------|
| Host                  | balam-ln2.supercomputo.unam.mx |
| Port                  | 2119                           |
| Executable            | script.nwchem                  |
| Arguments             | pruebaOGCE/nwchem/input2.in    |
| Standard Input        |                                |
| Standard Output       | pruebaOGCE/nwchem/pin2.o       |
| Standard Error        | pruebaOGCE/nwchem/pin2.e       |
| Directory             |                                |
| CPU Count             | 4                              |
| Wall Clock Time (min) |                                |
| JobType               | single                         |
| Queue                 | pruebas                        |

At the bottom of the form is a "Submit" button.

Figura 4.23: Parámetros para la ejecución de un job de Nwchem

### 4.5.3. MPI

Para ejecutar MPI en Kanbalam, fue necesario hacer modificaciones en el archivo `$GLOBUS_LOCATION/lib/perl/Globus/GRAM/JobManager/lsf.pm`, las modificaciones fueron:

```
print JOB "export PATH=/opt/appl/scripts-grid:/opt/hpmpi/bin:
/opt/hptc/bin:\$PATH\n";
print JOB "export LD_LIBRARY_PATH=/opt/hpmpi/lib/linux_amd64:
\$LD_LIBRARY_PATH\n";
print JOB "$mpirun -srun", ' ' ;
```

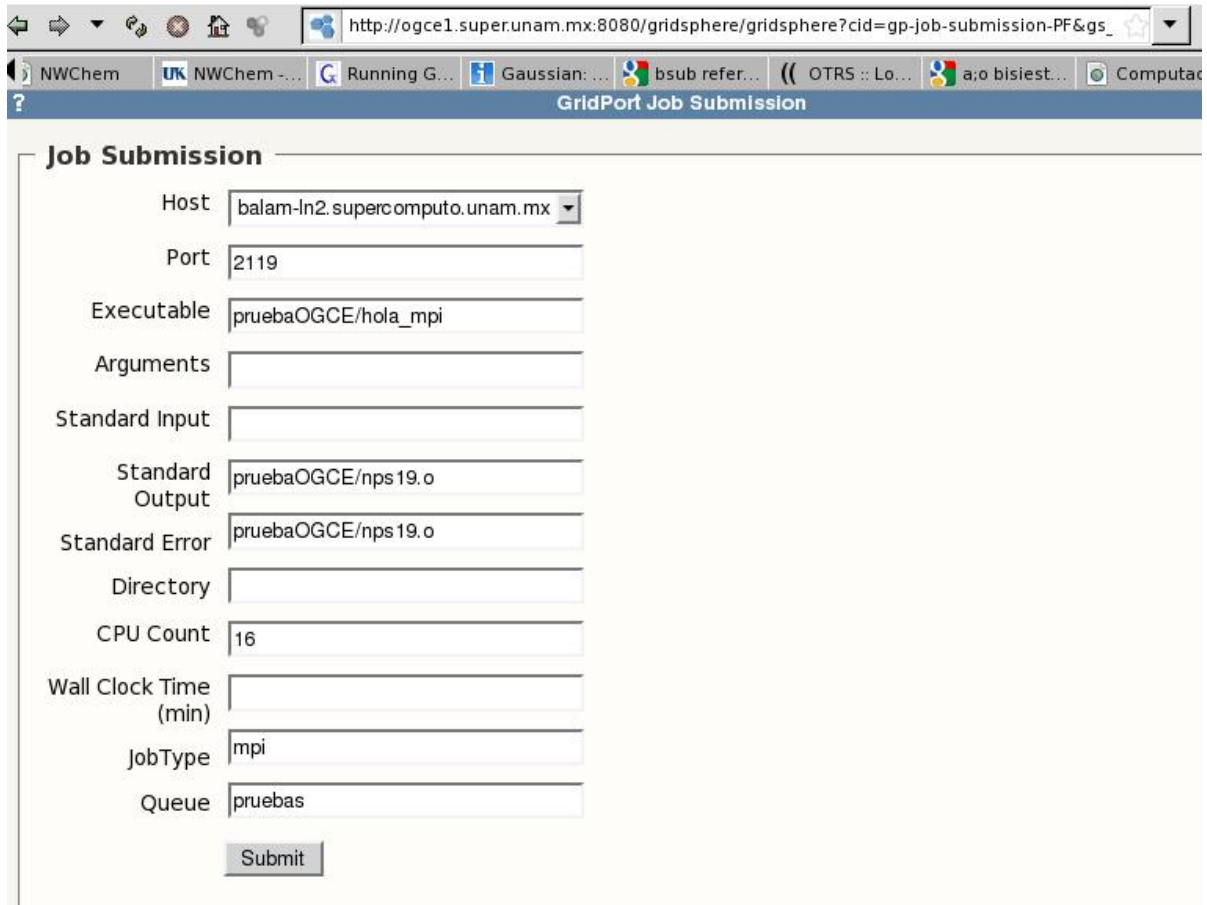
El código fuente de un programa en mpi es el siguiente:

```
#include<stdio.h>
#include<mpi.h>

int main(int argc, char *argv[])
{
int rank,nproc,namelen;
char nombre_proc [MPI_MAX_PROCESSOR_NAME];

MPI_Init(&argc, &argv);
MPI_Comm_size(MPI_COMM_WORLD,&nproc);
MPI_Comm_rank(MPI_COMM_WORLD,&rank);
MPI_Get_processor_name(nombre_proc,&namelen);
printf("Hola proceso %d de %d en %s \n",rank, nproc,nombre_proc);
MPI_Finalize();
return 0;
}
```

Los parámetros para solicitar la ejecución de este programa de mpi son los siguientes:



The screenshot shows a web browser window with the URL `http://ogce1.super.unam.mx:8080/gridsphere/gridsphere?cid=gp-job-submission-PF&gs_`. The browser tabs include "NWChem", "Running G...", "Gaussian:...", "bsub refer...", "OTRS :: Lo...", "a;o bisiest...", and "Computac". The page title is "GridPort Job Submission". The main content area is titled "Job Submission" and contains the following fields:

|                       |                                |
|-----------------------|--------------------------------|
| Host                  | balam-ln2.supercomputo.unam.mx |
| Port                  | 2119                           |
| Executable            | pruebaOGCE/hola_mpi            |
| Arguments             |                                |
| Standard Input        |                                |
| Standard Output       | pruebaOGCE/nps19.o             |
| Standard Error        | pruebaOGCE/nps19.o             |
| Directory             |                                |
| CPU Count             | 16                             |
| Wall Clock Time (min) |                                |
| JobType               | mpi                            |
| Queue                 | pruebas                        |

At the bottom of the form is a "Submit" button.

Figura 4.24: Parámetros para la ejecución de un job de MPI

- En el campo *Executable* se proporciona el programa de mpi a ejecutar.
- El campo *CPU count* puede ser variable y depende de la cola seleccionada.
- En el campo *Job Type* se proporciona el valor mpi.

#### 4.5.4. OpenMP

Para ejecutar correctamente OpenMP en Kanbalam, fue necesario hacer modificaciones en el archivo `$GLOBUS_LOCATION/lib/perl/Globus/GRAM/JobManager/lsf.pm`, las modificaciones fueron las siguientes:

```
if($description->count() lt '4')
  {print JOB "export OMP_NUM_THREADS=", $description->count(), "\n";}
else
  {print JOB "export OMP_NUM_THREADS=4 \n";}
```

El código fuente del programa que utiliza openmp es el siguiente:

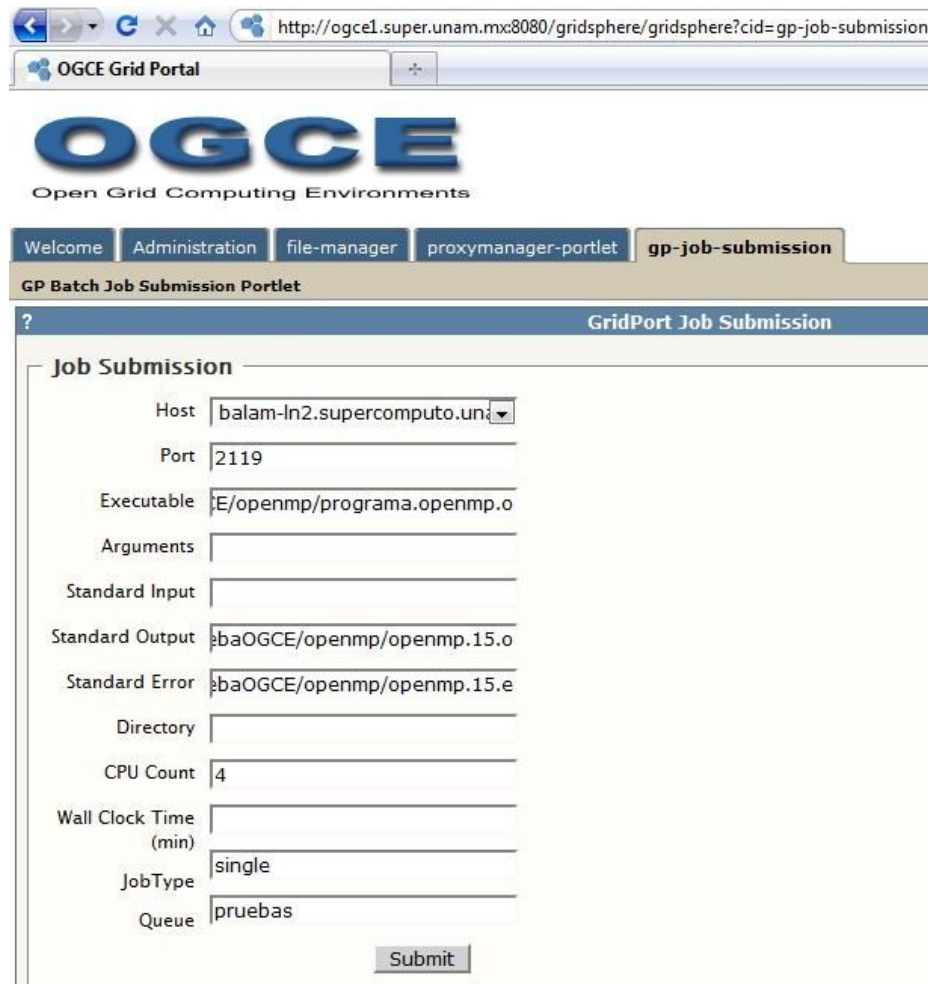
```
#include <omp.h>
#define N      1000000

main ()
{
  int i,tid,c[N];

  #pragma omp parallel shared(c) private(i,tid)
  {

    #pragma omp for schedule(dynamic) nowait
    for (i=0; i <= N; i++)
      {
        c[i] = i*i;
        tid = omp_get_thread_num();
        printf("i = %d, thread = %d, c[i] = %d\n", i, tid, c[i]);
      }
    } /* end of parallel section */
}
```

Para utilizar openmp mediante el portal grid, se tienen que utilizar los siguientes parámetros:



The screenshot shows a web browser window with the URL `http://ogce1.super.unam.mx:8080/gridsphere/gridsphere?cid=gp-job-submission`. The page title is "OGCE Grid Portal" and the logo "OGCE Open Grid Computing Environments" is displayed. The navigation menu includes "Welcome", "Administration", "file-manager", "proxymanager-portlet", and "gp-job-submission". The main content area is titled "GP Batch Job Submission Portlet" and "GridPort Job Submission". The "Job Submission" form contains the following fields:

|                       |                             |
|-----------------------|-----------------------------|
| Host                  | balam-ln2.supercomputo.unam |
| Port                  | 2119                        |
| Executable            | E/openmp/programa.openmp.o  |
| Arguments             |                             |
| Standard Input        |                             |
| Standard Output       | baOGCE/openmp/openmp.15.o   |
| Standard Error        | baOGCE/openmp/openmp.15.e   |
| Directory             |                             |
| CPU Count             | 4                           |
| Wall Clock Time (min) |                             |
| JobType               | single                      |
| Queue                 | pruebas                     |

A "Submit" button is located at the bottom right of the form.

Figura 4.25: Parámetros para la ejecución de un job de OpenMP

- En el campo *Executable* se proporciona el nombre del programa de openmp.
- En el campo *Arguments* se proporciona los argumentos que utilice el programa.
- En el campo *CPU count* sólo se pueden proporcionar hasta 4 procesadores debido a que sólo existen 4 procesadores utilizando memoria compartida por nodo.

Y finalmente se da click en *submit* para enviar el job.

## 4.6. Estadísticas del Portal Grid

Se utilizó la herramienta JMeter [26] con el objetivo de realizar pruebas de estrés en el portal grid OGCE y verificar su correcto funcionamiento bajo estas condiciones. Mediante esta herramienta se simuló el ingreso de varios usuarios al portal y de esta manera se caracterizó el comportamiento con los diferentes componentes del portal grid.

A continuación en la Fig. 4.26 la duración (eje x) y el porcentaje de carga del procesador (eje y) para distintos números de usuarios, mientras que en la Fig. 4.27 se muestra la carga en memoria:

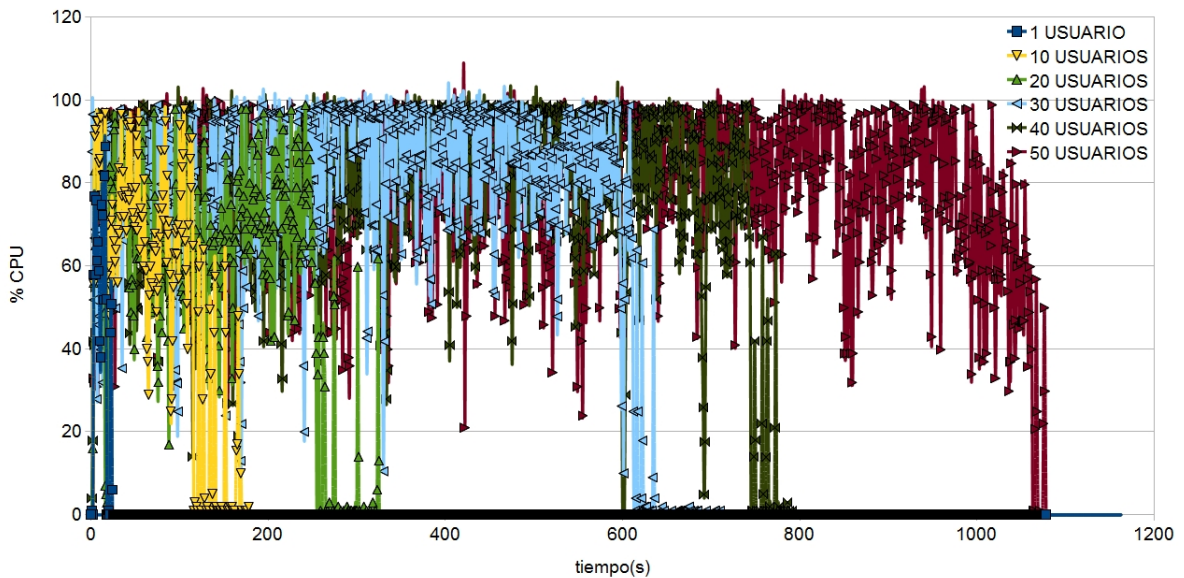


Figura 4.26: Carga del CPU al variar los usuarios conectados en el portal

De la gráfica 4.26 se puede apreciar que conforme se aumenta la cantidad de usuarios en el portal se aumenta el tiempo de ejecución de la prueba. El porcentaje de uso del procesador para todos los casos es aproximadamente 100 %. Para un usuario la prueba dura 20 segundos aproximadamente y cuando el portal atiende simultáneamente a 50 usuarios el tiempo de la prueba se alarga a cerca de los 19 minutos.

De la figura 4.27 no se aprecia un aumento de carga en memoria considerable al aumentar el número de usuarios, el uso de la memoria fue menor de 30 porcientos en todas las pruebas realizadas.

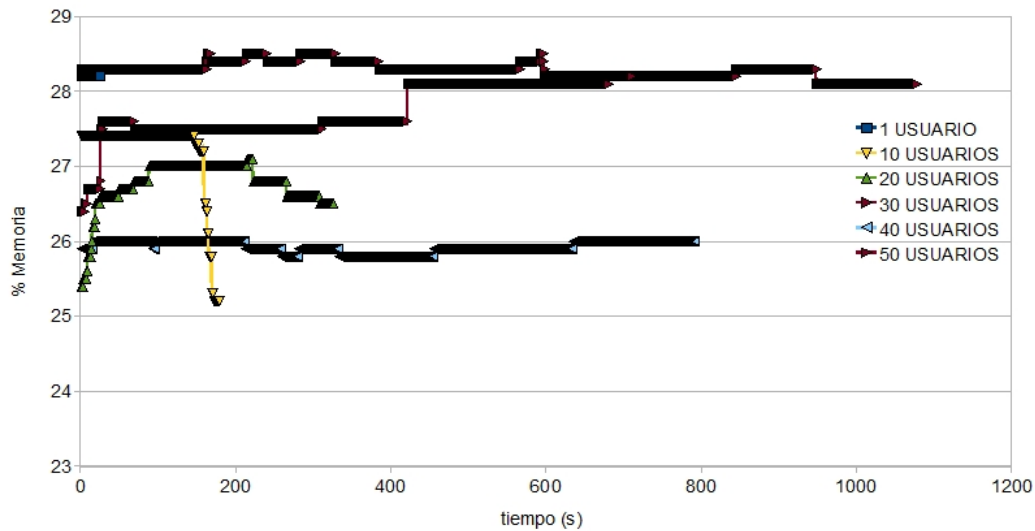


Figura 4.27: Porcentaje de memoria utilizado al incrementar la cantidad de usuarios conectados

A continuación se presentan los tiempos de respuesta obtenidos al variar la cantidad de usuarios que utilizan los difentes componentes del portal:

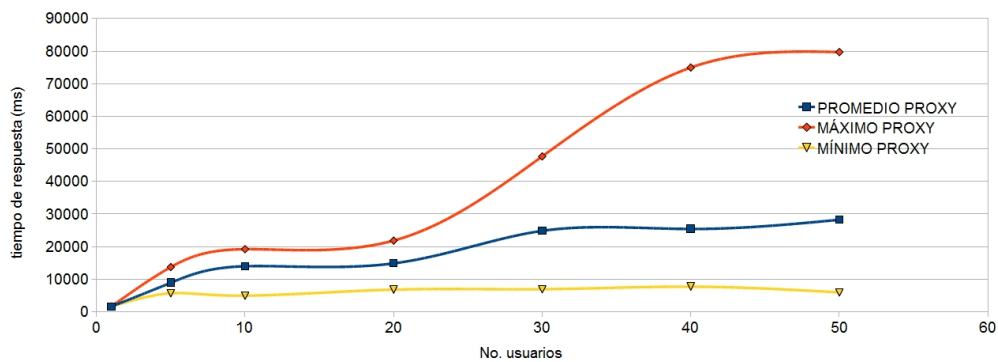


Figura 4.28: Carga al solicitar un certificado proxy

En la figura 4.28 se muestra que para una carga de 50 usuarios el tiempo promedio para obtener un certificado es de 30 segundos y el tiempo máximo que puede llegar a demorarse es de 1 minuto con 20 segundos aproximadamente.



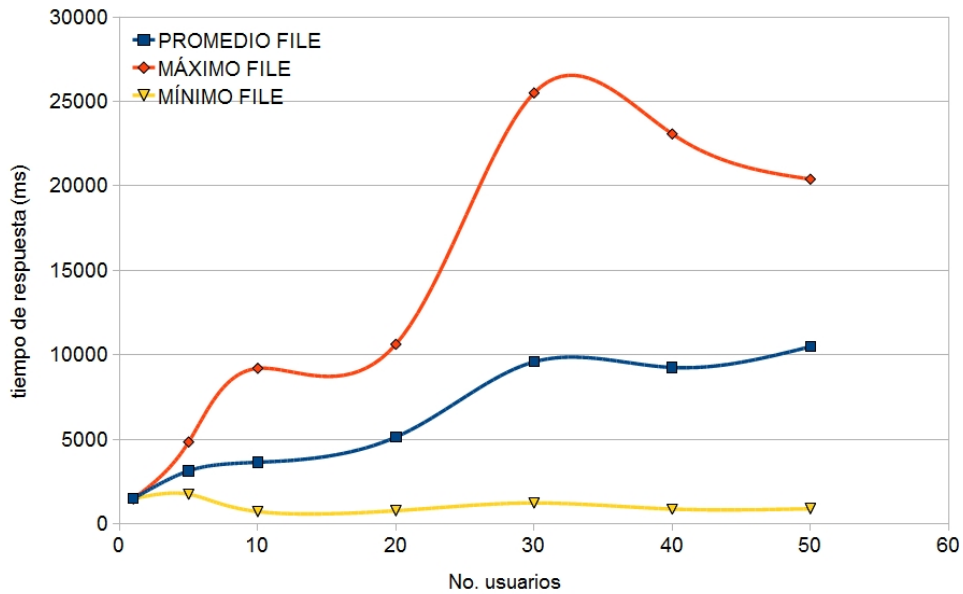


Figura 4.29: Carga al entrar al portlet de manejo de archivos

De la figura 4.29 se puede apreciar que el tiempo promedio de respuesta para entrar al portlet de manejo de archivos es de 11 segundos aproximadamente cuando se encuentran 50 usuarios utilizando el portal y un tiempo máximo de 25 segundos aproximadamente para 30 usuarios.

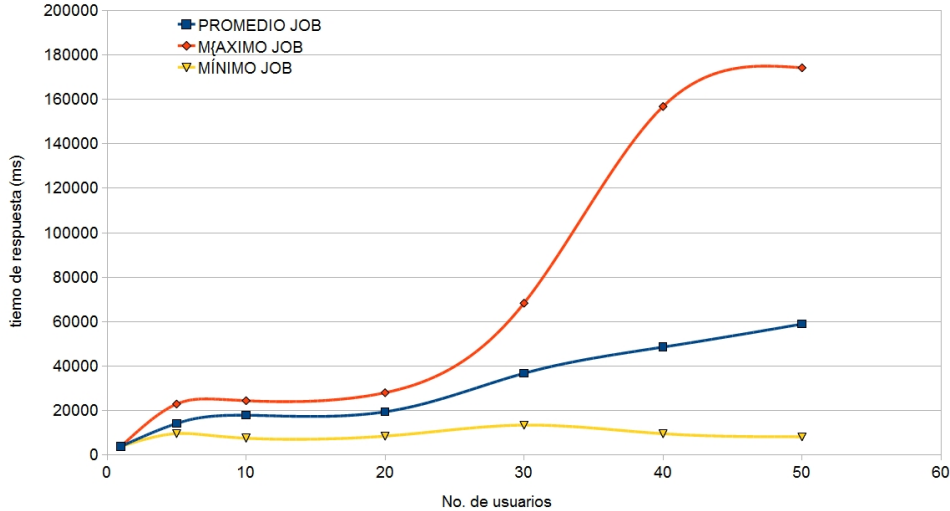


Figura 4.30: Carga al entrar al portlet de envío de jobs

En la figura 4.30 se muestra que el tiempo promedio de respuesta es de 1 minuto al acceder al portlet de envío de jobs cuando la carga es de 50 usuarios; y el tiempo máximo de respuesta de este portlet es de casi 3 minutos para una carga de 50 usuarios.

Los tiempos de respuesta en los principales componentes del portal grid son muy grandes al tener una carga mayor a 30 usuarios. Esto puede explicarse por las características del equipo donde se instaló el portal grid, puesto que se encuentra en una máquina virtual con un procesador y dicha máquina fue adquirida hace más de cinco años. Para tener mejores tiempos de respuesta en los diferentes componentes del portal grid, se recomienda instalar el portal grid en un equipo más potente, con varios procesadores y preferentemente en un servidor dedicado.

## 4.7. Resumen

En este capítulo se describieron las características en hardware y en software de los equipos donde se instaló el portal grid, el servidor de certificados, el middleware de grid y los recursos de grid (*fabric*).

También se describió cómo se llevó a cabo la instalación del portal grid 2.5, la configuración de los diferentes componentes del portal, la modificación del código fuente del portal para poder interactuar adecuadamente con la parte de envío de jobs a una supercomputadora, y así mismo se mostraron los archivos de configuración de aplicaciones específicas en el middleware de grid utilizado.

Se mostró el funcionamiento del portal, al probar sus diferentes componentes: manejo de certificados proxy, manejo de archivos, pruebas de ejecución de diferentes aplicaciones (gaussian, nwchem, mpi y openmp).

Y finalmente se realizaron pruebas de carga al utilizar una herramienta que permite simular a usuarios dentro del portal grid. Esta herramienta y los resultados obtenidos, nos permitirán determinar las características en hardware que debe tener una máquina de producción del portal grid para una determinada cantidad de usuarios a atender.

