

Servicio General de Informática aplicada a la investigación

Cálculo Científico

◀ **Calcular en el SGI-IZO** ▶

Dr. Txema Mercero

Dr. Edu Ogando

SGIker

<http://www.ehu.es/SGI>



Introducción

● Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

Monitorización web

¡Socorro!

Introducción



Introducción

Qué es este curso

Las máquinas del SGI-IZO tienen unas particularidades que es preciso o muy recomendable conocer para trabajar con ellas. Por otro lado el uso de máquinas de cálculo requiere de una metodología especial y el uso de programas como gestores de colas.

En este curso se explicara la composición y estructura de las máquinas del SGI-IZO, como calcular en ellas y como hacerlo de una forma eficiente.

[Introducción](#)[● Introducción](#)[Arquitecturas](#)[Compilar](#)[Paralelización](#)[Configuración](#)[Colas](#)[Monitorización web](#)[¡Socorro!](#)



- Introducción
- Arquitecturas**
 - Tipos de arquitecturas
- Compilar
- Paralelización
- Configuración
- Colas
- Monitorización web
- ¡Socorro!

Arquitecturas



Tipos de arquitecturas

Introducción

Arquitecturas

● Tipos de arquitecturas

Compilar

Paralelización

Configuración

Colas

Monitorización web

¡Socorro!

- Itanium2 - ia64
- Opteron - x86_64 (Xeon, core2duo)
- Pentium - x86

- Mac - ppc
- Sun - sparc



Tipos de arquitecturas (Itanium2)

Introducción

Arquitecturas

● Tipos de arquitecturas

Compilar

Paralelización

Configuración

Colas

Monitorización web

¡Socorro!

- Itanium2 - ia64
Compiladores de Intel (ifort, icc, icpc)
- Opteron - x86_64 (Xeon, core2duo)
- Pentium - x86

- Mac - ppc
- Sun - sparc



Tipos de arquitecturas (Opteron)

Introducción

Arquitecturas

● Tipos de arquitecturas

Compilar

Paralelización

Configuración

Colas

Monitorización web

¡Socorro!

- Itanium2 - ia64
- Opteron - x86_64 (Xeon, core2duo)
Compiladores de PGI (pgf77, pgf90, pgcc, pgCC)
Alternativamente compiladores de Intel y GNU
- Pentium - x86
- Mac - ppc
- Sun - sparc



Tipos de arquitecturas (Pentium)

Introducción

Arquitecturas

● Tipos de arquitecturas

Compilar

Paralelización

Configuración

Colas

Monitorización web

¡Socorro!

- Itanium2 - ia64
- Opteron - x86_64 (Xeon, core2duo)
- Pentium - x86 (Pendulo)
Compiladores de Intel
Alternativamente compiladores de PGI y GNU
- Mac - ppc
- Sun - sparc



- Introducción
- Arquitecturas
- Compilar**
 - Compilar
- Paralelización
- Configuración
- Colas
- Monitorización web
- ¡Socorro!

Compilar



Compilar

Introducción

Arquitecturas

Compilar

● Compilar

Paralelización

Configuración

Colas

Monitorización web

¡Socorro!

■ Compilar en Arina

```
ifort inputfile.f90 -o outputfile
```

```
ifort inputfile.f90 -o outputfile_ia64
```

■ Compilar en Maiz

```
pgf90 inputfile.f90 -o outputfile
```

```
pgf90 inputfile.f90 -o outputfile_x86_64
```

■ Compilar en Péndulo

```
ifort inputfile.f90 -o outputfile
```

```
ifort inputfile.f90 -o outputfile_i686
```

```
ifort hola.f90 -o hola_ia_64  
./hola_ia_64  
./hola_$(arch)
```



Cargar entorno

Los compiladores y librerías los actualizamos regularmente. No siempre se mantiene la compatibilidad hacia atrás. Hay dos soluciones:

1. Recompilar el programa. Para programas sencillos es más seguro y aprovechamos las nuevas ventajas de la actualización.

2. Cargar el entorno, librerías con las que se compiló.

```
source /opt/intel/fc/10.1.017/bin/ifortvars.sh  
source  
/opt/intel/mkl10.3/tools/environment/mklvars64.sh
```

Introducción

Arquitecturas

Compilar

● Compilar

Paralelización

Configuración

Colas

Monitorización web

¡Socorro!



Introducción

Arquitecturas

Compilar

Paralelización

- Paralelización
- Paralelización - OpenMP
- Paralelización - MPI
- Paralelización - Rendimiento

Configuración

Colas

Monitorización web

¡Socorro!

Paralelización



Paralelización

Es el uso de varios procesadores para realizar un único trabajo, ejecutándose de forma paralela en todos.

Es necesario programarlo específicamente.

- OpenMP
- MPI

Es el futuro

Introducción

Arquitecturas

Compilar

Paralelización

● Paralelización

● Paralelización - OpenMP

● Paralelización - MPI

● Paralelización - Rendimiento

Configuración

Colas

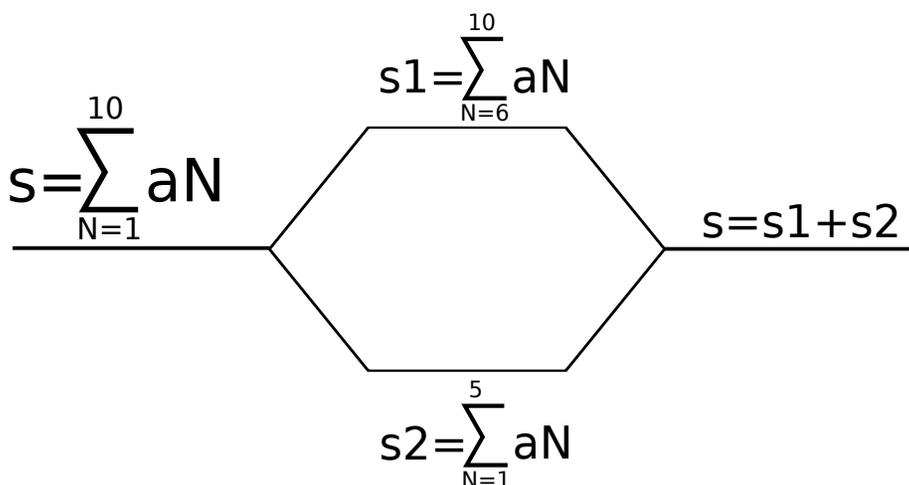
Monitorización web

¡Socorro!



Paralelización - OpenMP

- Más fácil de implementar.
- Menos eficiente en general.
- Sólo puede usar un nodo.

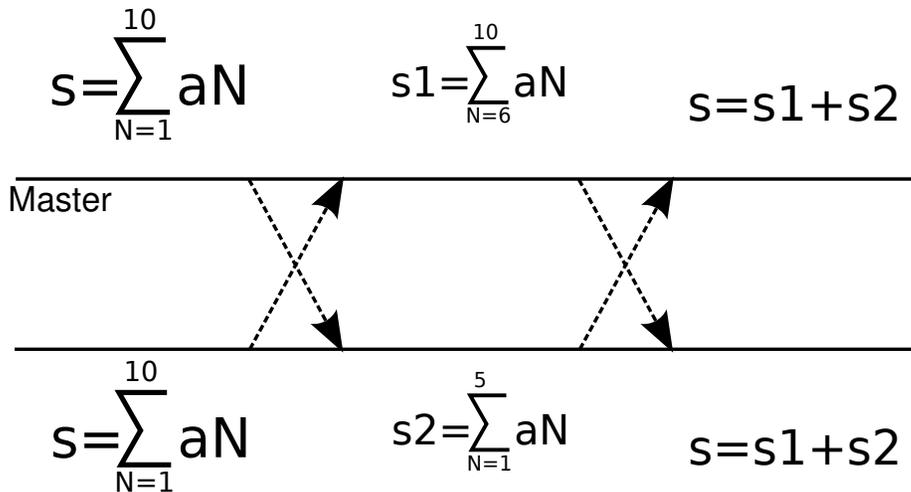


La variable a es la misma para los dos hilos.

Paralelización - MPI

- Introducción
- Arquitecturas
- Compilar
- Paralelización**
 - Paralelización
 - Paralelización - OpenMP
 - **Paralelización - MPI**
 - Paralelización - Rendimiento
- Configuración
- Colas
- Monitorización web
- ¡Socorro!

- Más difícil de implementar.
- Más eficiente en general.
- Permite usar muchos nodos.



La variable a es diferente para cada proceso.



Paralelización - Rendimiento

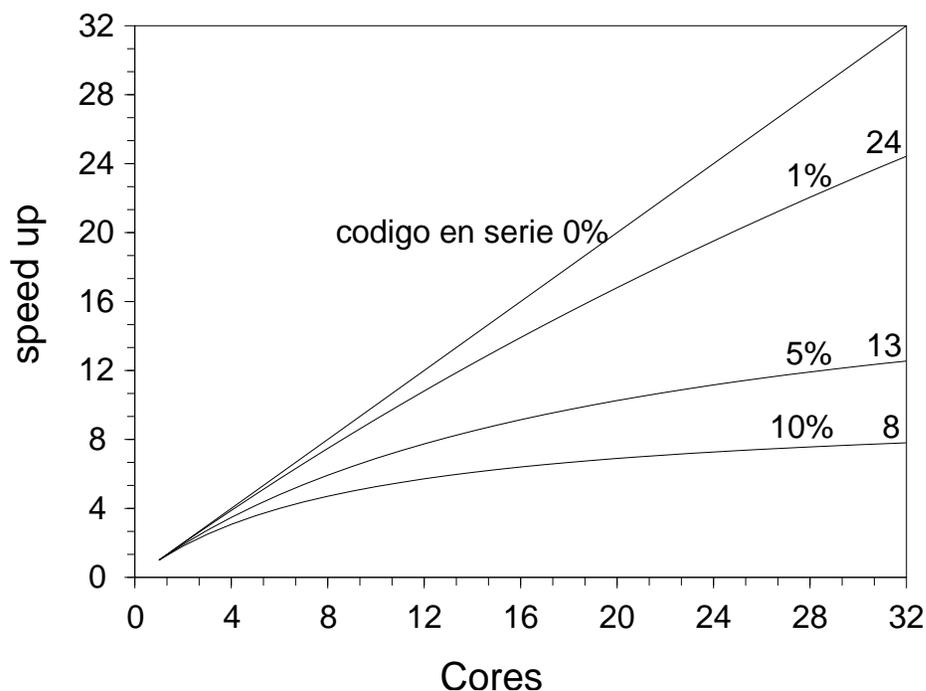
- Introducción
- Arquitecturas
- Compilar
- Paralelización**
 - Paralelización
 - Paralelización - OpenMP
 - Paralelización - MPI
 - **Paralelización - Rendimiento**
- Configuración
- Colas
- Monitorización web
- ¡Socorro!

“La ley de Amdahl”

$$T_0 = T_s + T_p(N = 1)$$

$$T = T_s + T_p(N)$$

$$speedup = \frac{T_0}{T} = \frac{N}{1+s(N-1)}$$





Introducción

Arquitecturas

Compilar

Paralelización

Configuración

- Configuración
- Servidores
- Nodos de Cálculo
- Almacenamiento
- Red
- Red
- Esquema
- Galería

Colas

Monitorización web

¡Socorro!

Configuración



Configuración

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

● Configuración

● Servidores

● Nodos de Cálculo

● Almacenamiento

● Red

● Red

● Esquema

● Galería

Colas

Monitorización web

¡Socorro!

- Servidores de conexión
- Nodos de cálculo.
- Almacenamiento.
- red.



Servidores

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

● Configuración

● **Servidores**

● Nodos de Cálculo

● Almacenamiento

● Red

● Red

● Esquema

● Galería

Colas

Monitorización web

¡Socorro!

■ Servidores de conexión.

Arina (Itanium2).

Maiz (Opteron) Más rápido para aplicaciones gráficas.

Péndulo (Pentium).

Arinaz (Antiguo servidor Itanium2).

■ Nodos de cálculo.

■ Almacenamiento.

■ Red.



Nodos de Cálculo

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

● Configuración

● Servidores

● Nodos de Cálculo

● Almacenamiento

● Red

● Red

● Esquema

● Galería

Colas

Monitorización web

¡Socorro!

- Servidores de conexión.
- Nodos de cálculo.
320 cores en 42 nodos con 4-8 CPUs.
4-128 GB de RAM por nodo.
Arquitecturas ia64 y x86_64.

160 cores i686 en Péndulo.

- Almacenamiento.
- Red.



Almacenamiento

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

● Configuración

● Servidores

● Nodos de Cálculo

● Almacenamiento

● Red

● Red

● Esquema

● Galería

Colas

Monitorización web

¡Socorro!

■ Servidores de conexión.

■ Nodos de cálculo.

■ Almacenamiento.

`/home.`

`/software (/software/bin).`

`/scratch (36-550 GB.)`

`/gscratch (4.2 TB.)`

■ Red.



Red

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

- Configuración
- Servidores
- Nodos de Cálculo
- Almacenamiento

● Red

- Red
- Esquema
- Galería

Colas

Monitorización web

¡Socorro!

- Servidores de conexión.
- Nodos de cálculo.
- Almacenamiento.
- Red.
Ethernet (Interna).
Infiniband (Cálculo).



Red

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

- Configuración
- Servidores
- Nodos de Cálculo
- Almacenamiento
- Red

● Red

- Esquema
- Galería

Colas

Monitorización web

¡Socorro!

- Servidores de conexión.
- Nodos de cálculo.
- Almacenamiento.
- Red.

NOTA

Péndulo usa PCs de las aulas de docencia por ello los nodos de cálculo solo están disponibles 11.5 horas entre semana y 58 horas los fines de semana.

Esquema

Introducción

Arquitecturas

Compilar

Paralelización

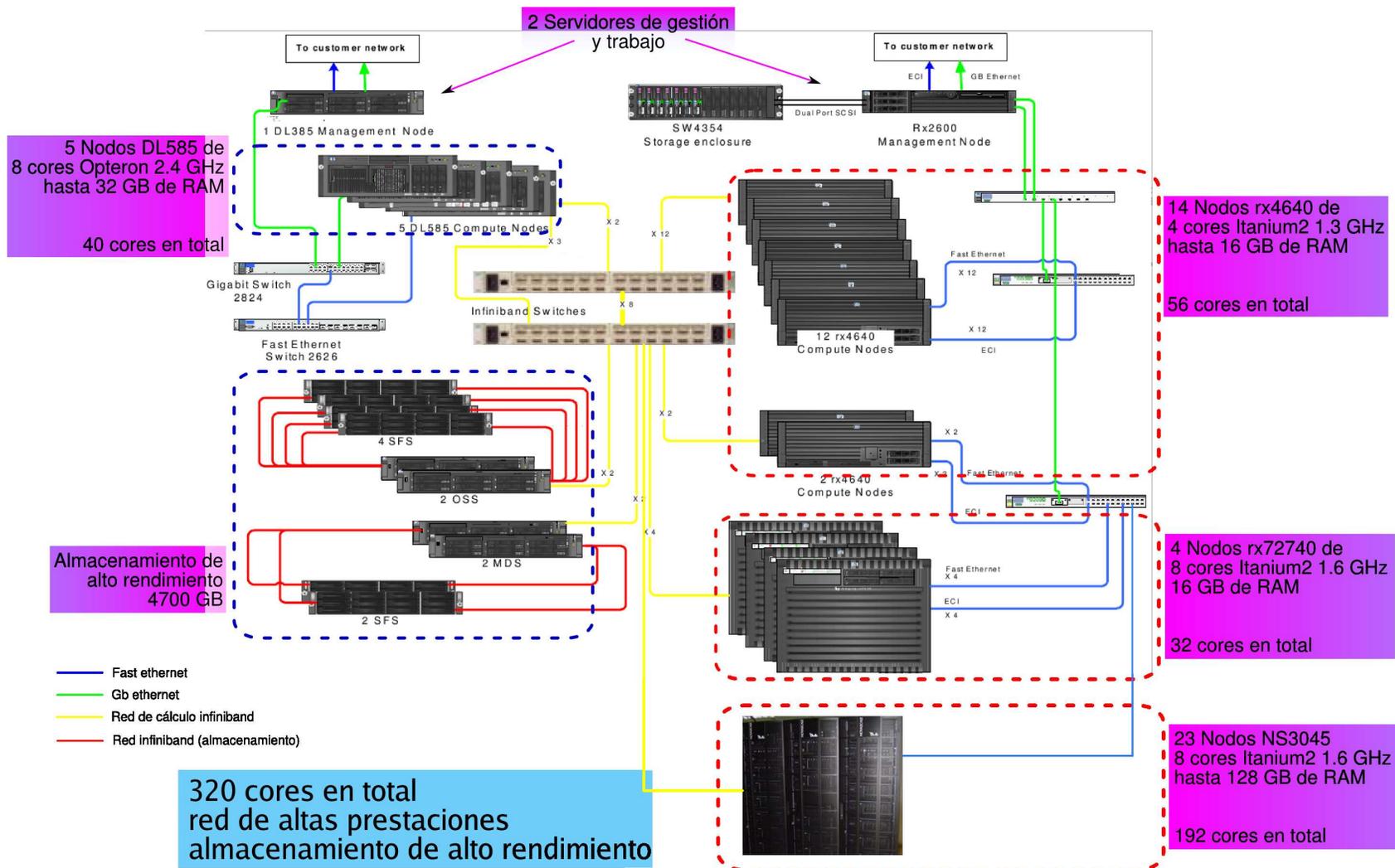
Configuración

- Configuración
- Servidores
- Nodos de Cálculo
- Almacenamiento
- Red
- Red
- Esquema
- Galería

Colas

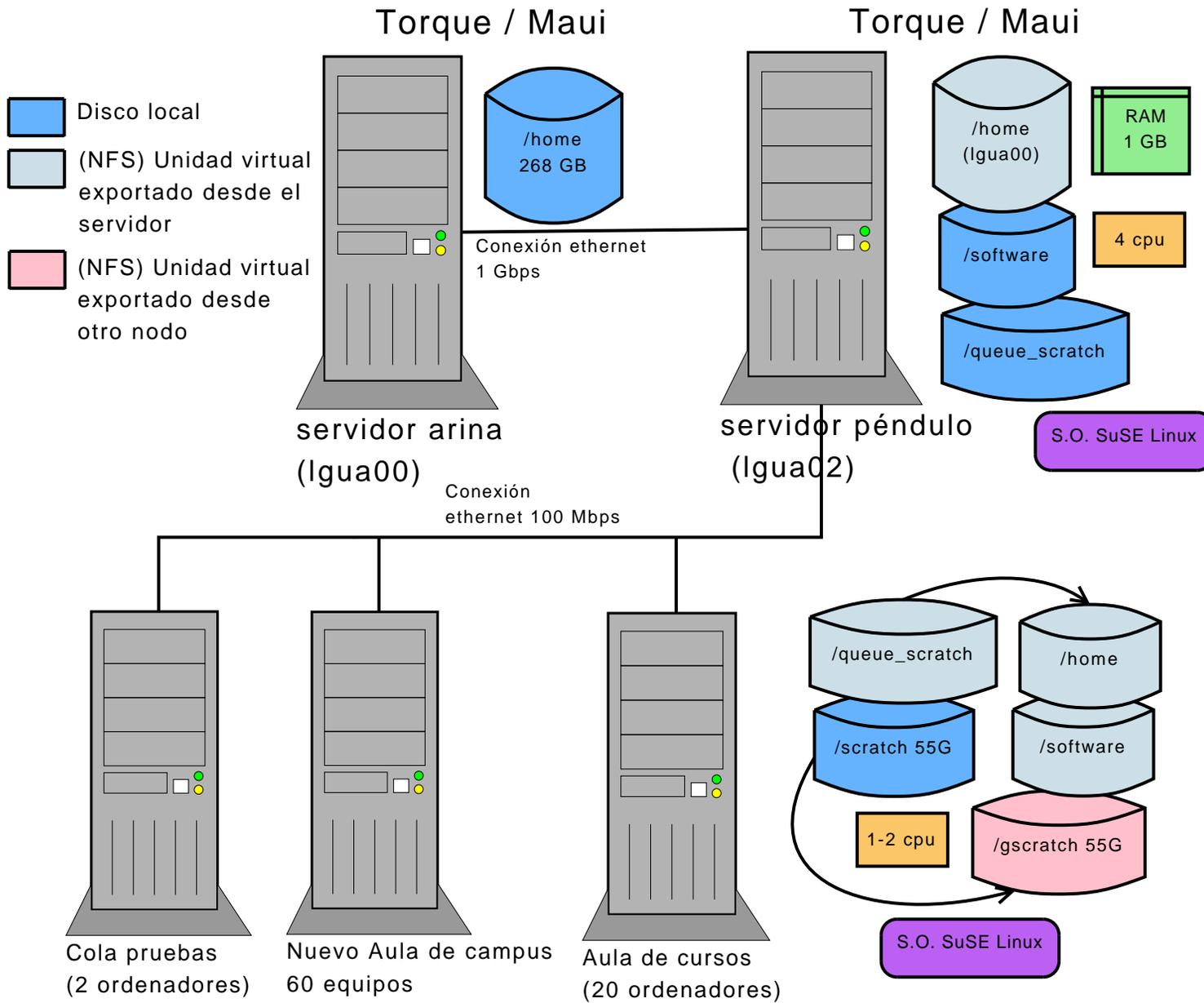
Monitorización web

¡Socorro!



Esquema

- Introducción
- Arquitecturas
- Compilar
- Paralelización
- Configuración**
 - Configuración
 - Servidores
 - Nodos de Cálculo
 - Almacenamiento
 - Red
 - Red
- Esquema**
- Galería
- Colas
- Monitorización web
- ¡Socorro!





Esquema

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

- Configuración
- Servidores
- Nodos de Cálculo
- Almacenamiento
- Red
- Red

● Esquema

● Galería

Colas

Monitorización web

¡Socorro!

Nuestra trabajo es que toda esta complejidad desaparezca y el uso sea transparente.



Galería

Introducción

Arquitecturas

Compilar

Paralelización

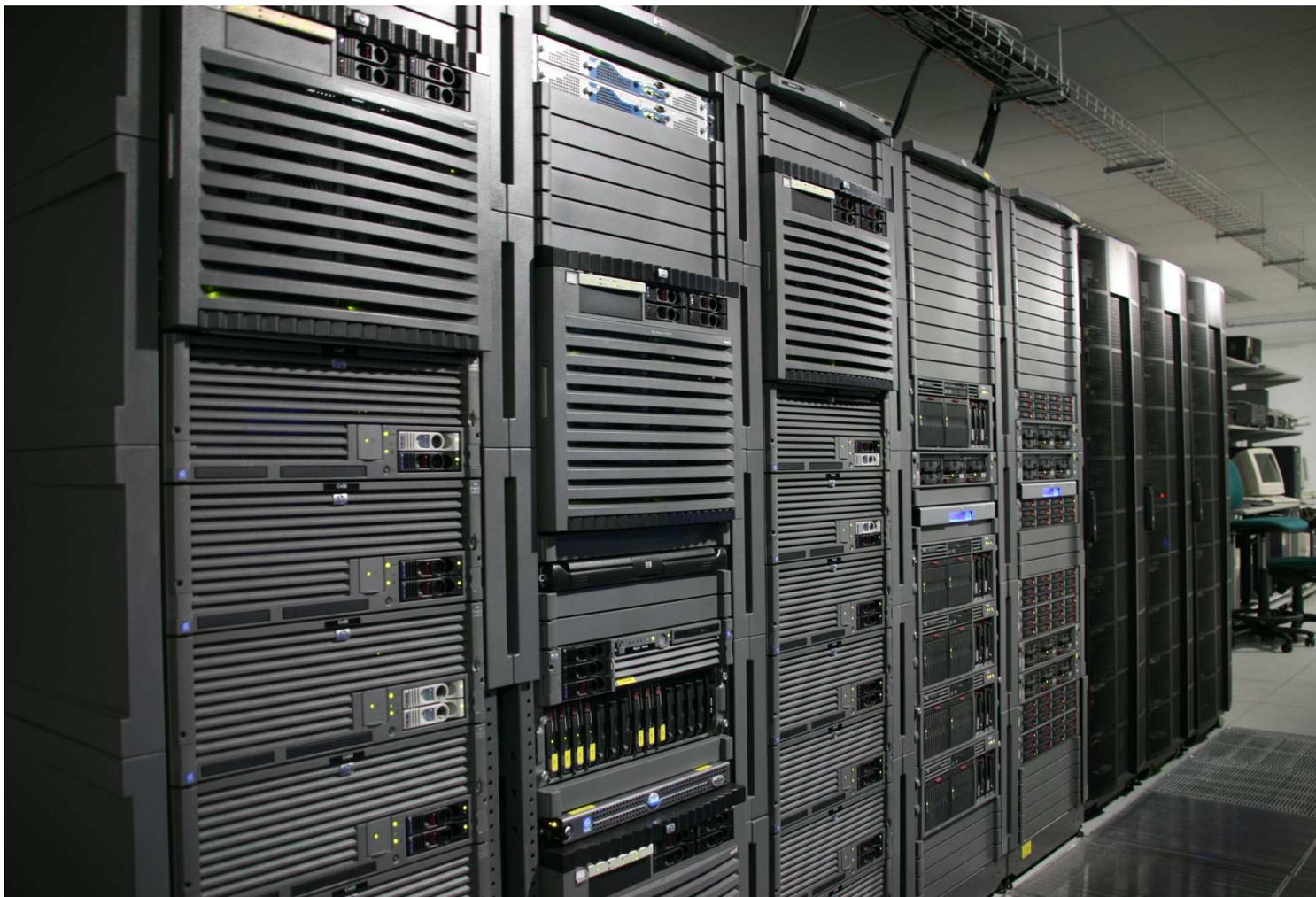
Configuración

- Configuración
- Servidores
- Nodos de Cálculo
- Almacenamiento
- Red
- Red
- Esquema
- Galería

Colas

Monitorización web

¡Socorro!





Galería

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

- Configuración
- Servidores
- Nodos de Cálculo
- Almacenamiento
- Red
- Red
- Esquema
- Galería

Colas

Monitorización web

¡Socorro!





[Introducción](#)

[Arquitecturas](#)

[Compilar](#)

[Paralelización](#)

[Configuración](#)

Colas

- Gestores: Torque + Maui
- Acciones básicas
- qsub
- qstat
- qdel
- Maui
- Comandos interesantes
- Ejercicios
- Script torque
- qsub específicos

[Monitorización web](#)

[¡Socorro!](#)

Colas



Gestores: Torque + Maui

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● qsub

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!

Torque gestiona los trabajos, conoce sus propiedades y las de los nodos. El scheduler Maui se encarga de gestionar el orden de los trabajos.

- 1 proceso un core, es lo más eficiente.
- Selecciona los nodos adecuados.
- Establece prioridades de ejecución.

Hay dos gestores de colas, uno para Arina y otro para Péndulo.

En los servidores sólo se pueden correr de forma interactiva o directa pequeños tests o programas.



Acciones básicas

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● qsub

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!

1. Mandar trabajos

qsub

2. Mostrar estado

qstat

3. Borrar trabajos

qdel



qsub

qsub es el comando para enviar un trabajo al sistema de colas.

```
qsub script.pbs
```

Otras variantes

qsub:	Realiza preguntas de modo interactivo
qsub_arina:	Manda los trabajos a Arina
qsub_pendolo:	Manda los trabajos a Péndulo
qsub_all:	Manda los trabajos a ambos clusters

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● **qsub**

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!



qsub

qsub_all manda los trabajos a ambos clusteres pero se ejecuta allí donde primero se encuentran recursos disponibles.

qsub_all traduce automáticamente por ejemplo

```
#PBS -l nodes=1:ppn=8
```

que es adecuado para Arina a

```
#PBS -l nodes=4:ppn=2
```

adecuado para los PCs con dos cores de Péndulo.

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● **qsub**

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!



Ejemplo qsub

[Introducción](#)[Arquitecturas](#)[Compilar](#)[Paralelización](#)[Configuración](#)[Colas](#)

- Gestores: Torque + Maui

- Acciones básicas

- **qsub**

- qstat

- qdel

- Maui

- Comandos interesantes

- Ejercicios

- Script torque

- qsub específicos

[Monitorización web](#)[¡Socorro!](#)

```
qsub
```

```
##entramos en modo interactivo
```

```
0
```

```
sleep 60
```

```
job.pbs
```

```
1:00:00
```

```
250mb
```

```
1
```

```
1
```

```
0
```

```
0
```

```
n
```

```
resultados
```

```
1
```



Ejemplo qsub

Hemos lanzado un trabajo a la cola y el scrip que ha utilizado está en nuestro directorio.

```
qstat -a  
ls  
head -4 job.pbs
```

Veamos las directrices de PBS

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● qsub

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!



qsub -l

-l nodes nos permite definir las propiedades como el número y la distribución de cores que vamos a usar.

- #PBS -l nodes=1 : ppn=1 1 nodo un core
- #PBS -l nodes=1 : ppn=4 1 nodo 4 cores por nodo
- #PBS -l nodes=2 : ppn=4 2 nodos 4 cores por nodo
- #PBS -l nodes=1 : ppn=8 1 nodos 8 cores por nodo
- #PBS -l nodes=2 : ppn=8 2 nodos 8 cores por nodo

Estas combinaciones son las más efectivas, reducen el tiempo de espera en cola y maximizan el uso del cluster.

Usar 1 cpu o múltiplos de 4.

- Introducción
- Arquitecturas
- Compilar
- Paralelización
- Configuración
- Colas**
 - Gestores: Torque + Maui
 - Acciones básicas
 - qsub**
 - qstat
 - qdel
 - Maui
 - Comandos interesantes
 - Ejercicios
 - Script torque
 - qsub específicos
- Monitorización web
- ¡Socorro!



qsub -l

Etiquetas

Las usamos para escoger el tipo de nodos que queremos usar.

- `#PBS -l nodes=1:ppn=1:itanium` (a la arquitectura itanium)
- `#PBS -l nodes=1:ppn=1:opteron` (a la arquitectura opteron)
- `#PBS -l nodes=1:ppn=1:itanium4` (a los nodos itanium de 4 cores)
- `#PBS -l nodes=1:ppn=1:itanium8` (a los nodos itanium de 8 cores)

http://www.ehu.es/SGI/Cluster_Arina_tf.html

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● **qsub**

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!



qsub -l

-l mem nos permite definir la memoria RAM.

- #PBS -l mem=100mb (100 MB)
- #PBS -l mem=2gb (2 GB)

No nos reserva memoria a nivel de SO, es sólo gestión.

Si está cerca del límite la cambiamos.

Es mejor ajustar para tener más posibilidades.

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● qsub

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!



qsub -l

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● qsub

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!

-l walltime selecciona el tiempo real de cálculo.

■ #PBS -l walltime=12:00:00
Solicitamos 12 horas (HH:MM:SS)

Es mejor ajustar para tener más posibilidades para permitirnos una mejor administración.

Podemos aumentar el walltime.



qsub -l

- Introducción
- Arquitecturas
- Compilar
- Paralelización
- Configuración
- Colas**
 - Gestores: Torque + Maui
 - Acciones básicas
 - qsub**
 - qstat
 - qdel
 - Maui
 - Comandos interesantes
 - Ejercicios
 - Script torque
 - qsub específicos
- Monitorización web
- ¡Socorro!

El walltime determina la prioridad con que se le asigna a nuestro trabajo. Es independiente de la cola.
Hay dos formas de aumentar la prioridad.

1. Proporcionalmente al tiempo que lleva en cola.
2. Proporcionalmente al tiempo que lleva en la cola respecto del walltime solicitado.
3. Como tiempo de referencia están puestas 36 horas.

Ejemplo:

walltime=100:00:00 y 50 horas en cola.

$$\text{Suma } 50 + \frac{50}{100} 36 = 68 \text{ puntos.}$$

walltime=2:00:00 y 4 horas en cola.

$$\text{Suma } 4 + \frac{4}{2} 36 = 76 \text{ puntos.}$$



qsub -M -m

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● qsub

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!

■ #PBS -M edu.ogando@ehu.es
Definimos nuestra dirección de correo

■ #PBS -m be

Nos manda un trabajo cuando el programa entra en ejecución *b* y cuando termina *e* y nos manda cierta información, podemos escoger una de las acciones o las dos.



qsub -W

[Introducción](#)[Arquitecturas](#)[Compilar](#)[Paralelización](#)[Configuración](#)[Colas](#)

- Gestores: Torque + Maui

- Acciones básicas

- **qsub**

- qstat

- qdel

- Maui

- Comandos interesantes

- Ejercicios

- Script torque

- qsub específicos

[Monitorización web](#)[¡Socorro!](#)

■ `#PBS -W group_list=grupo`

Nuestro trabajo es lanzado como si pertenecieramos al grupo señalado. Permite que el trabajo use la “cuenta” de otro grupo. Es necesario que el usuario pertenezca a ese otro grupo.



qstat

- Nos da información del status de los trabajos.
 - Tiene varios flags, -u nos permite filtrar un usuario.
 - Accesos rápidos
- Trabajos en estado queued

qqstat

Trabajos en estado running

rqstat

rrqstat

Otras variantes

qstat_arina: Estado de los trabajos a Arina

qstat_pendolo: Estado de los trabajos a Péndulo

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● qsub

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!

[Introducción](#)[Arquitecturas](#)[Compilar](#)[Paralelización](#)[Configuración](#)[Colas](#)

- Gestores: Torque + Maui
- Acciones básicas
- qsub
- qstat
- **qdel**
- Maui
- Comandos interesantes
- Ejercicios
- Script torque
- qsub específicos

[Monitorización web](#)[¡Socorro!](#)

- Nos permite eliminar trabajos de la cola.
- Si está en estado running se interrumpe la ejecución del script, no se devuelven los ficheros.

Otras variantes

qdel_arina:	Elimina los trabajos a Arina
qdel_pendolo:	Elimina los trabajos a Péndulo
qdel_all:	Elimina los trabajos a ambos clusters



Maui

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

- Gestores: Torque + Maui
- Acciones básicas
- qsub
- qstat
- qdel

● Maui

- Comandos interesantes
- Ejercicios
- Script torque
- qsub específicos

Monitorización web

¡Socorro!

■ showq

Nos da información sobre los trabajos. Flags -b, -u, -r, -i.
Tiene varios flags, -u nos permite filtrar un usuario.

■ showstart

Nos da una estimación de cuando entrará nuestro trabajo en ejecución.



Comandos interesantes

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

- Gestores: Torque + Maui
- Acciones básicas
- qsub
- qstat
- qdel
- Maui

● Comandos interesantes

- Ejercicios
- Script torque
- qsub específicos

Monitorización web

¡Socorro!

■ now

Nos da información sobre el estado de los nodos.

■ myjobs

Nos da información sobre nuestros trabajos en ejecución.

■ gastua

Nos da el gasto realizado desde la última convocatoria.



Ejercicios

[Introducción](#)[Arquitecturas](#)[Compilar](#)[Paralelización](#)[Configuración](#)[Colas](#)

- Gestores: Torque + Maui
- Acciones básicas
- qsub
- qstat
- qdel
- Maui
- Comandos interesantes
- Ejercicios
- Script torque
- qsub específicos

[Monitorización web](#)[¡Socorro!](#)

Miramos el estado del cluster, editamos el script para usar 2 cpus y mandamos el trabajo a Arina

```
now  
showq  
gedit job.pbs &  
qsub job.pbs
```

Buscamos nuestro trabajo y lo eliminamos

```
qstat -u $USER  
qdel (jobid)  
qstat -u $USER
```



Ejercicios

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

● Gestores: Torque + Maui

● Acciones básicas

● qsub

● qstat

● qdel

● Maui

● Comandos interesantes

● Ejercicios

● Script torque

● qsub específicos

Monitorización web

¡Socorro!

Mandamos el trabajo a Péndulo y lo eliminamos

```
qsub_pendolo job.pbs  
qstat_pendolo  
qdel_pendolo (jobid)
```

Mandamos el trabajo a ambos clusters y lo eliminamos

```
qsub_all job.pbs  
qstat_pendolo -u $USER  
qstat -u $USER  
qdel_all (jobid)  
qstat_pendolo -u $USER
```



Script torque

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

- Gestores: Torque + Maui
- Acciones básicas
- qsub
- qstat
- qdel
- Maui
- Comandos interesantes
- Ejercicios
- Script torque
- qsub específicos

Monitorización web

¡Socorro!

Veamos que hace este script.

```
gedit job.pbs &
```



qsub específicos

Existen comandos específicos para facilitar el envío de trabajos de ciertos programas.

send_adf	Manda trabajos adf.
send_cp2k	Manda trabajos cp2k.
send_cpmd	Manda trabajos cpdm.
send_gamess	Manda trabajos gamess.
send_gauss	Manda trabajos gauss.
send_jaguar	Manda trabajos jaguar.
send_mmodel	Manda trabajos macromodel.
send_nwchem	Manda trabajos nwchem.
send_qsite	Manda trabajos qsite.
send_siesta	Manda trabajos siesta.

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

- Gestores: Torque + Maui
- Acciones básicas
- qsub
- qstat
- qdel
- Maui
- Comandos interesantes
- Ejercicios
- Script torque
- qsub específicos

Monitorización web

¡Socorro!



qsub específicos

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

- Gestores: Torque + Maui
- Acciones básicas
- qsub
- qstat
- qdel
- Maui
- Comandos interesantes
- Ejercicios
- Script torque
- qsub específicos

Monitorización web

¡Socorro!

La sintaxis es en todos similar, por ejemplo

```
send_gauss h2o 10:00:00 4
```

Para lanzar un trabajo de Gaussian cuyo fichero de entrada es h2o.com, con un walltime de 10 horas y a 4 cores.

Para detalles sobre este y otros comandos ver

http://www.ehu.es/SGI/Como_encolar_trabajos_tf.html



- Introducción
- Arquitecturas
- Compilar
- Paralelización
- Configuración
- Colas
- Monitorización web**
 - Monitorizaci3n Web
- ¡Socorro!

Monitorización web



Monitorización Web

- Introducción
- Arquitecturas
- Compilar
- Paralelización
- Configuración
- Colas
- Monitorización web**
 - Monitorización Web
- ¡Socorro!

Se pueden monitorizar

- Los trabajos.
- El estado del cluster.
- El accounting (pluggins para firefox <http://mycroft.mozdev.org/>).

Se actualiza cada 5 minutos salvo el accounting que se actualiza cada 24 horas.



[Introducción](#)

[Arquitecturas](#)

[Compilar](#)

[Paralelización](#)

[Configuración](#)

[Colas](#)

[Monitorización web](#)

[¡Socorro!](#)

● [Ayuda](#)

¡Socorro!



Ayuda

Introducción

Arquitecturas

Compilar

Paralelización

Configuración

Colas

Monitorización web

¡Socorro!
● Ayuda

<http://www.ehu.es/sgi>
<http://www.google.com>

No dudéis en preguntarnos

Txema Mercero

Edu Ogando

Gracias por vuestra amable atención.