Colas Batch
- Actualizado (22.11.2010)

Consulte aquí a nosa Política de Acceso.
Sistema de colas dos servidores de cálculo

Para traballos que requiran máis recursos (tempo de cálculo, memoria ou espazo en disco) que os limitados polo shell interactivo, deberase utilizar o sistema de colas. Este sistema é o Sun Grid Engine. O modo de enviar traballos implica coñecer de antemán os valores máximos de cantidade de tempo, número de procesadores, memoria e espazo en disco que vai requirir o cálculo. Isto permite ademais un mellor aproveitamento dos recursos do sistema por parte de todos os usuarios.
O comando para enviar traballos ao sistema de colas nos servidores de cálculo é qsub, seguido por unha lista dos recursos que necesita o traballo. Por exemplo, supoñamos que queremos enviar un traballo Gaussian que non precisa máis de 2 gigabytes de memoria e 10 gigabytes de scratch, e estimamos un tempo de execución aproximado non superior a 24 horas, utilizando 1 procesador. Se o ficheiro de entrada para Gaussian chamase script.com e se atopa no directorio "pruebas", a forma de enviar este traballo a cola é:

```
qsub -l num_proc=1,s_rt=24:00:00,s_vmem=2G,h_fsize=10G
cd $HOME/pruebas
g98 < script.com
control+D
```

Se non se produce ningún erro, obteremos unha mensaxe deste tipo:

```
Your job 492 ("STDIN") has been submitted
```

O número 492 é o que se chama identificador de traballo ou JOBID e permítenos identificar o noso traballo dentro do sistema de colas (por exemplo, para utilizar co comando qstat e saber se se está a executar, encolado, etc.). Tamén é importante indicar este número ao facer consultas telefónicas ou por correo electrónico co persoal de sistemas.
A forma de especificar recursos é coa directiva -l, seguida dos recursos que se solicitan separados cunha ",". É imprescindible deixar un espazo en branco entre a opción "-l" e a lista de recursos. Os recursos deben ser:

Recurso

Significado

Unidades

Valor mínimo

Valor máximo

SVG

HPC320

Superdome

num_proc
Número de CPUs (procesadores) requeridos polo traballo

1

16
(sólo con MPI)

4

16

s_rt
Máximo cantidade de tempo real que pode durar un traballo

Tempo

-

200:00:00

360:00:00

300:00:00

s_vmem
Cantidade total de memoria RAM requerida polo traballo

Tamaño

112M-SVGD
112M-HPC
550M-SD

4G

16G

64G

h_fsize
Máximo espazo requerido por un único ficheiro creado polo traballo

Tamaño

-

120G

30G

400G

Debemos ter en conta que:
É moi importante deixar un espazo en branco entre a opción "-l" e o seguinte recurso, mentres que despois non deberá haber ningún outro espazo en branco separando os recursos.

1. Estes valores son máximos, polo que non se poderán superar. Esto quere dicir que se cremos que o noso traballo durará unhas 23 horas, debemos poñer como s_rt=24:00:00 para asegurarnos de deixar unha marxe ao traballo. Despois de 24 horas o sistema finalizará o traballo automaticamente, aínda que este non rematase.
2. Canto máis axustados sexan estes valores aos recursos que realmente consume o traballo, maior prioridade para entrar en execución terá o traballo.

Se estes recursos non son suficientes para o traballo, este abortará por falla de recursos e será necesario indicar os valores adecuados. En xeral, recomendamos solicitar recursos o máis próximos, por riba, aos valores que se estimen necesarios. O motivo é que cantos menos recursos se soliciten, con maior prioridade entrará o traballo en execución.
O formato das unidades para os recursos é o seguinte:
Tempo: especifica o período de tempo máximo durante o que se pode utilizar un recurso. O formato é o seguinte: [[horas:]minutos:]segundos, por exemplo:

• 30:00 son 30 minutos
• 100:00:00 son 100 horas
• 1200 son 1200 segundos (20 minutos)

Tamaño: especifica o tamaño máximo en Megabytes. Exprésase na forma enteiro[sufixo]. O sufixo actúa como un multiplicador definido na seguinte táboa:

k
kilo (1024) bytes

m
Mega (1,048,576) bytes

g
kmgGiga (1,073,741,824) bytes

Resumindo, imos poñer uns exemplos para distintos traballos:

1. Para un traballo que require pouco consumo de memoria e tempo de execución:
qsub -l num_proc=1,s_rt=10:00,s_vmem=100M,h_fsize=100M traballo.sh
2. Traballo que require moito tempo de execución (80 horas) e pouca memoria (é suficiente con 256mb):
qsub -l num_proc=1,s_rt=80:00:00,s_vmem=256M,h_fsize=10M traballo.sh
3. Un traballo con grandes requirimentos de memoria (4 Gigabytes) pero pouco tempo de execución:
qsub -l num_proc=1,s_rt=30:00,s_vmem=4G,h_fsize=10M traballo.sh
4. Un traballo que xera un ficheiro grande (ata 20 Gigabytes) de resultados:
qsub -l num_proc=1,s_rt=30:00:00,s_vmem=500M,h_fsize=20g traballo.sh
5. Un traballo que consume 100 horas de CPU, 2 Gigabytes de memoria e xera un ficheiro de 5 gigabytes:
qsub -l num_proc=1,s_rt=100:00:00,s_vmem=2G,h_fsize=5G traballo.sh
6. Un traballo paralelo con 8 procesadores e 10 horas de tempo de execución total, 8 Gigabytes de memoria total e xera

un ficheiro de 10 gigabytes:
qsub -l num_proc=8,s_rt=10:00:00,s_vmem=8G,h_fsize=10G traballo.sh

Se se precisa utilizar valores superiores aos límites destes recursos, débese solicitar o acceso á cola especial, enviando un mail ao enderezo de correo
Unha vez que executamos o comando qsub, e obtemos o identificador para o traballo, este pasa a unha cola apropiada para a súa execución. O traballo esperará a súa vez ou o momento en que estean dispoñibles os recursos solicitados, para pasar a execución, e finalmente o traballo rematará e desaparecerá da cola.

Chequeando o estado dos traballos
Para comprobar o estado en que se atopan os traballos, pódese utilizar o comando qstat

qstat

Obteremos unha saída como a seguinte:

Job id

prior

name

user

state

at

Queue

master

489
0
carlosf
r
12/29/2003
19:49:05
Cola1
MASTER

O significado dos campos é o seguinte:
Job id: 489 é o valor do JOB-ID que lle asignou o sistema de colasPBS. O JobID é un identificador único para cada traballo e permite realizar o seguimento do mesmo.
Prior: Indica a prioridade coa que se está a executar o traballo
Name: STDIN é o nome do traballo que se enviou á cola. Se se enviou un traballo dende a entrada estándar (é dicir, escribindo os comandos ao enviar o traballo), aparecerá STDIN. No caso de ser un script, aparecerá o nome do script.
User: carlosf é o login do usuario que enviou o traballo á cola
State: "r" é o estado no que se atopa o traballo e indica que está en execución (running). Os outros posibles estados dun traballo son:

• t: transferíndose o traballo para comezar a súa execución. R indica que o traballo está en execución.
• s: suspendido temporalmente para executar traballos máis prioritarios.
• w: o traballo está encolado en espera de que haxa suficientes recursos para ser executado ou debido a que se excederon os límites por usuario.

Submit/start at: Data e hora na que o traballo foi enviado á cola ou entrou en execución.
Queue: cola 1 é o nome da cola á que se enviou o traballo. A cola destino dependerá dos recursos que se solicitaran.
Master: indica o host dende o que se enviou o traballo.

Execución de traballos paralelos no SVG
O Clúster SVG está formado por 80 nodos monoprocesador con interconexión Gigabit-ethernet e outros 16 nodos monoprocesador con interconexión Myrinet.
A execución de traballos paralelos está limitada a estes últimos nodos para aproveitar a rede Myrinet (en caso de querer executar traballos paralelos de tipo SMP, OpenMP, ... ou MPI con nodos máis potentes, aconsellamos utilizar o Superdome).
Para enviar traballos paralelos a estes nodos, emprégase o mesmo comando qsub engadindo a opción -pe mpi número_de_nodos, da seguinte forma:

```
qsub -l num_proc=1,s_rt=1:00:00,s_vmem=128M,h_fsize=1G -pe mpi 2 script.sh
```

Neste exemplo, estanse solicitando dous nodos (con 1 procesador por nodo, especificado pola opción num_proc=1) para o traballo paralelo mpi "script.sh". O número de nodos debe estar entre 1 e 16.
Execución de traballos paralelos no HP Superdome
O Cluster Integrity Superdome está formado por 2 nodos de 64 CPUs cada un. É posible, polo tanto, utilizar modelos de paralelización de memoria compartida (do estilo OpenMP) para a comunicación dentro dun nodo, coma modelos de memoria distribuída (do tipo MPI) para a comunicación entre-nodos e tamén dentro dun nodo. Nos dous casos, o número de CPUs máximo que se pode utilizar está limitado a 16 e sempre se utilizarán dentro dun mesmo nodo.