Universidad Autónoma de Madrid

excelencia UAM CSIC

Escuela Politécnica Superior

Proyecto básico: Gestión de colas

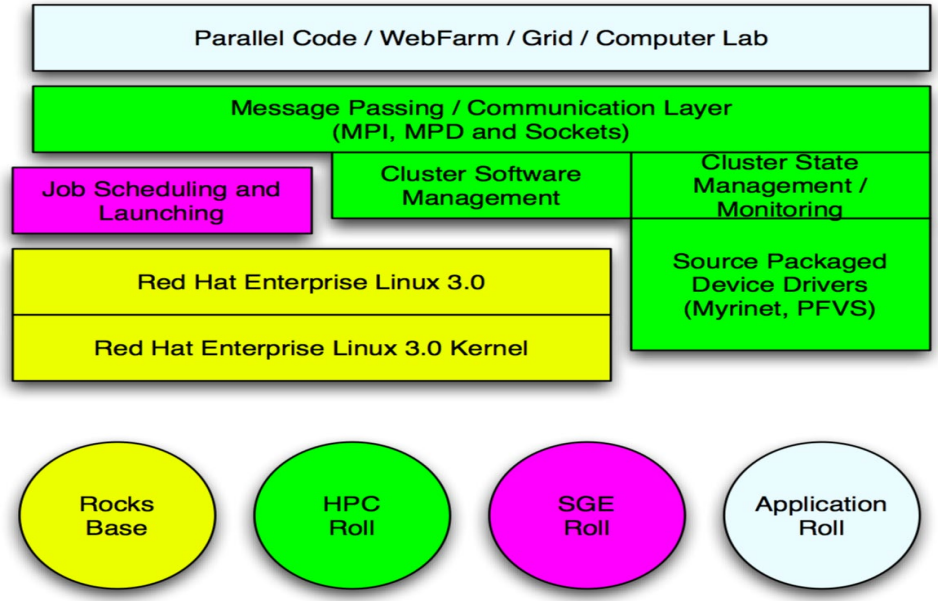
Francisco Javier Gómez Arribas

Agenda

- Cluster de ordenadores
  - Cluster multicomputador: Arquitectura y Componentes.
  - Tipos de cluster y dominio de aplicación: HPC, Bigdata
  - Construir y desplegar un cluster
  - Administración básica, planificación y balanceo de tareas.
  - Gestor de tareas SGE
    - Script de Lanzamiento
    - Envío de tareas a colas
    - Creación de colas

Gestor de tareas: Sun Grid Engine (SGE)

- El Sun Grid Engine (SGE) maneja y controla todos las tareas que se ejecutan en el cluster, incluyendo el balanceo de las diferentes tareas entre las máquinas disponibles
  - Se asegura de que todas las tareas se ejecutan en máquinas con suficiente memoria y número de CPUs para ejecutar la tarea

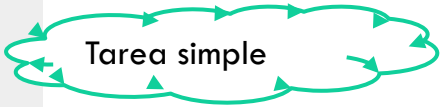


Sun Grid Engine (SGE)

- Una tarea SGE se va a definir como una secuencia de uno o varios de comandos que lanzan un programa (aplicación comercial o ejecutable realizado por el usuario), y que se va a implementar como un script.
  - En muchos casos sigue la misma secuencia de comandos que se escriben en una shell cuando se de realizar el lanzamiento de un programa.
  - Posteriormente se envía la tarea o script a SGE junto con la lista de requisitos (uso de memoria, número de CPUs, tiempo de ejecución, etc.) y SGE encontrará la máquina o máquinas necesarias para ejecutar la tarea
  - Gestión mediante **colas** y **entornos paralelos**

SGE : Ejemplo de script que defien una tarea

```
#----------------------------Start program.job-----------------------
#!/bin/bash

# The name of the job, can be whatever makes sense to you
#$ -N ProgramName_EstimatedHoursToFinish_MiscComment

# The job should be placed into the queue 'all.q'.
#$ -q default

# Redirect output stream to this file.
#$ -o sge_output.dat

# Redirect error stream to this file.
#$ -e sge_error.dat

# The batchsystem should use the current directory as working directory.
# Both files (output.dat and error.dat) will be placed in the current
# directory. The batchsystem assumes to find the executable in this directory.
#$ -cwd

# This is my email address for notifications. I want to have all notifications
# at the master node of this cluster.
#$ -M username@domain.name

# Send me an email when the job is finished.
#$ -m e

# This is the file to be executed.
echo $PATH

time genesis ./CLIMB9.g > ./sge.out
#----------------------------End program.sge-----------------------
```



Colas en SGE

- En el cluster SGE permite la configuración de varias colas que diferencian como utilizar los recursos.
- Un ejemplo de colas disponibles en un cluster:
  - ***all.q***: Incluye a todos los equipos del cluster con todos los cores disponibles.
    - Es la configuración estándar.
  - ***all_1.q***: Incluye a todos los equipos del cluster pero solo usa un core por equipo.
  - ***all_amd.q***: Incluye a todos los equipos AMD con todos los cores disponibles.
  - ***all_gpu.q***: Incluye a todos los equipos con GPUs. Para realizar pruebas con GPUs.
  - ***all_intel.q***: Incluye a todos los equipos Intel con todos los cores.

Entornos paralelos

El entorno paralelo permite decidir el modo de funcionamiento de la cola. Ejemplos de entornos paralelos que se pueden definir:

- orte: Es el entorno por defecto. Reparte los procesos de la tarea asignando todos los posibles a cada máquina. Si quedan procesos sin asignar entonces salta a otra máquina. Así hasta completar el número de procesos lanzados.
- mpi: Este entorno está configurado para lanzar tareas MPI. Reparte los procesos de manera equitativa entre todos los equipos (disponibles) del clúster.
- openmp: Este entorno está configurado para lanzar tareas OpenMP. Todos los procesos se ejecutan en una única máquina. Este entorno requiere asegurarse de que existe al menos un equipo en el clúster que tienen tantos procesadores/*cores* como procesos se quieren lanzar.

Creación de tareas para su ejecución en una cola

Suponga un script (con nombre simple.q) con el siguiente contenido

```
#!/bin/bash
cd /home/usuario/ejemplo
myprog
```

- SGE elegirá la máquina para esta tarea y entonces la tarea cambiará de directorio y ejecutará el programa "myprog" en la máquina seleccionada
- En este caso no se especifican ficheros de entrada y salida
  - Se asume que cualquier fichero de entrada se debe encontrar en el directorio de ejecución o que el programa sabe donde encontrarlo
  - Del mismo modo, los ficheros de salida se generarán en el mismo directorio
- La primera línea "# !", identifica el tipo de shell que se usará como interprete de comandos - en este caso es la shell "bash"
  - El uso de "# !" en la primera línea es una convención estándar de Linux
- Otras líneas que empiezan por "#" son líneas de comentarios y serán ignoradas durante la ejecución del script

Envío de tareas a SGE

- Para enviar una tarea a SGE se usa el comando 'qsub' y el nombre del script:

```
> qsub simple.q
```

donde 'simple.q' es el nombre del fichero que incluye los comandos anteriores

- Cuando se envía una tarea es posible indicar ciertos parámetros que indican como debe ejecutarse:

```
> qsub -o simple.out simple.q
```

  - La opción '-o simple.out' indica que cualquier salida debe ser redireccionada al fichero 'simple.out'
  - SGE provee un número de opciones que pueden ser usadas para identificar como la tarea se debe ejecutar y que tipo de recursos son necesarios para que la tarea se ejecute correctamente

Añadiendo opciones de SGE

- Se pueden añadir las opciones directamente al script:

```
#!/bin/bash
#
#$ -S /bin/bash
#$ -cwd
#$ -o simple.out
#$ -j y
cd /home/usuario/ejemplo
./myprog
```

- La opción '**-cwd**' le dice a SGE que ejecute la tarea en el mismo directorio desde el que se ejecutó el comando qsub - por ejemplo, SGE se moverá al directorio de trabajo antes de ejecutar el script de la tarea
- La opción '**-o**' se emplea para dirigir la salida a un fichero
- La opción '**-S**' es otro modo de indicar a SGE el tipo de shell (tcsh, bash, or sh).
- La opción '**-j y**' es un modo de decir al SGE que combine la salida estándar-error con la salida estándar, de modo que toda la salida del programa irá al fichero 'simple.out'

Enviando tareas MPI a SGE

- Un script sencillo para una tarea MPI es el siguiente:

```
#!/bin/bash
#
#$ -S /bin/bash
#$ -cwd
#$ -o mpirun.out
#$ -j y
/opt/openmpi/bin/mpirun -np $NSLOTS ./mpiprogram
```

- En este caso MPI debe estar instalado en /opt/openmpi
- El número de procesos a lanzar se encuentra almacenado en la variable $NSLOTS cuyo valor se asignara cuando se envíe la tarea SGE:

```
> qsub -pe orte 4 mpisimple.q
```

En este caso, se lanzar el script 'mpisimple.q' solicitando 4 procesos en el entorno paralelo 'orte' (Open Run-Time Environment)

Enviando tareas MPI a SGE

- Es posible añadir esta información dentro del script:

```
#!/bin/bash
#
#$ -S /bin/bash
#$ -cwd
#$ -o mpirun.out
#$ -j y
#$ -pe orte 4
/opt/openmpi/bin/mpirun -np $NSLOTS ./mpiprogram
```

- No es necesario modificar el script para cambiar el número de procesos. Si lanzamos el script con otro número de procesos, se toma el nuevo valor:

```
> qsub -pe orte 8 mpisimple.q
```

Gestionar tareas en SGE

- Una vez enviada una tarea mediante 'qsub' a SGE, es posible conocer el estado de la tarea mediante el comando 'qstat', que además nos muestra información de la tarea:

```
> qsub -pe orte 4 run.sh
Your job 9 ("run.sh") has been submitted

> qstat
job-ID prior name user state submit/start at queue slots ja-task-ID
-----------------------------------------------------------------------------
8 0.55500 run_mpi_ri jose r 03/15/2010 11:35:45 all.q@compute-0-6.local 8
9 0.00000 run.sh jose qw 03/15/2010 11:35:45 4
```

Gestionar tareas en SGE

- Este caso, además de la nueva tarea con ID 9, existe una tarea previa con ID 8, llamada 'run_mpi'
  - La tarea 8 se está ejecutando en 8 procesadores (número de slots) ya que su estado (state) es 'r'
  - La tarea 9 llamada 'run.sh' está esperando en la cola ya que el estado de la misma (state) es 'qw'
    - Eso significa que SGE todavía no le ha asignado los recursos necesarios

- Es probable que la nueva tarea se ejecute en paralelo con la otra tarea ya que hay procesadores disponibles

- En otras ocasiones es posible que tengamos que esperar a que otras tareas terminen antes de ejecutar la nueva tarea

Gestionar tareas en SGE

- Para eliminar una tarea hacemos lo siguiente:

```
> qdel 14
jose has deleted job 14
```

- Como se puede apreciar, el parámetro que se pasa a 'qdel' es el ID de la tarea

Uso de colas en SGE

- Para utilizar una cola particular solo es necesario indicarlo durante la llamada a "qsub" mediante el parámetro '-q' (también se puede añadir dentro del script):

```
> qsub -pe orte 2 -q all_1.q run_mpi_ring.sh
```

  - En este caso ejecutamos el script 'run_mpi_ring.sh' con dos procesos en el entorno paralelo 'orte' usando la cola 'all_1.q'

- Si hacemos "qstat" podemos comprobar que la tarea se está ejecutando en la cola 'all_1.q' como indica el campo queue:

```
% qstat
job-ID prior name user state submit/start at queue slots ja-task-ID
-----------------------------------------------------------------------------
92 0.55500 run_mpi_ri pepe r 03/30/2010 15:05:53 all_1.q@compute-0-1.local 2
```

Gestión de colas SGE : ver configuración

```
Comandos para ver configuración de SGE

% Comandos para gestionar COLAS

% qconf -sql      para listar colas
% qconf -sq mpi.q   para ver las características de una cola llamada mpi.q

 Comandos para gestionar ENTORNOS DE EJECUCION PARALELA

% qconf -spl     para listar los entornos paralelos disponibles
% qconf -sp mpi       para ver la definición del entorno denominado

Comandos para gestionar Grupos de hosts.

% qconf -shgrpl     para listar los grupos de hosts disponibles
```

Gestión de colas SGE : Crear grupo de equipos

Un grupo de equipos se define con un nombre y un listado de nodos.

Por defecto el grupo @allhosts contiene todos los nodos.

```
$ qconf -shgrp @allhosts
```

Creamos nuestro propio grupo de equipos de la siguiente forma:

```
$ qconf -shgrp @allhosts > /nuestra/ruta/ejemplo/a/MyHostGroup.txt
```

Se edita el campo group name (por ejemplo, llamamos a este nuevo grupo @MyHostGroup), y lo añadimos a los gestionados por SGE) con los siguientes comandos:

Nota:El proceso de añadir el grupo de host necesita ser root pero listarlo no.

```
$ su
# qconf -Ahgrp /nuestra/ruta/ejemplo/a/MyHostGroup.txt
# exit
$ qconf -shgrpl
```

Gestión de colas SGE : Crear Cola

- La cola all.q se proporciona por defecto y contiene siempre a todos los nodos del cluster.
- Puede interesar crear colas con un subconjunto de los nodos que compartan ciertas características, o bien modificar cualquiera de las muchas opciones de la cola de trabajo.

El proceso sería siendo root:

1. Ver todas las colas de trabajo gestionadas por SGE,
2. Mostrar en detalle la configuración de una cola en concreto
3. Copiar la configuración de la cola en un archivo.
4. Después se añade a las colas gestionadas por SGE y se modifican el resto de campos que se deseen.

Gestión de colas SGE : Crear Cola

- Los pasos anteriores se realizan con los siguientes comandos:

$ **qconf -sql**

$ qconf -sq all.q

$ qconf -sq all.q >/nuestra/ruta/ejemplo/MyCola.txt

Se debe modificar con un editor de textos

- el campo qname por otro nombre que no sea all.q, por ejemplo MyCola.q y
- también modificar los campos hostlist con el grupo de nodos que corresponda y
- slots para que sea coherente con el campo hostlist.

- Después se añade a las colas gestionadas por SGE y se modifican el resto de campos que se deseen con los dos siguientes comandos:

$ qconf -Aq / nuestra/ruta/ejemplo/MyCola.txt

y se puede modificar con:

$ **qconf -mq MyCola.q**

Bibliografía y enlaces de interés

- Bibliografía
  - High Performance Cluster Computing: Architectures and Systems by Rajkumar Buyya Ed Prentice Hall PTR; ISBN: 0130137847; 1st edition (1999)
  - Linux Cluster Architecture by Alex VreniosSams;

    ISBN: 0672323680; (2002)
  - Linux Clustering: Building and Maintaining Linux Clusters by Charles Bookman

    Ed:New Riders Publishing; ISBN: 1578702747; 1st edition (2002)
- Enlaces
  - http://www.rocksclusters.org/
  - http://www.hispacluster.org/
  - http://www.openclustergroup.org/
  - http://www.beowulf.org/ y http://www.beowulf-underground.org/