Queueing system
PBS Pro 使用介紹

報告人：高鈺盛

勁智數位科技股份有限公司
客服專線： 07-5228611# 32
客服信箱：
yusengao@infowrap.com.tw

勁智數位科技
Leading @ Extreme Computing

Agenda

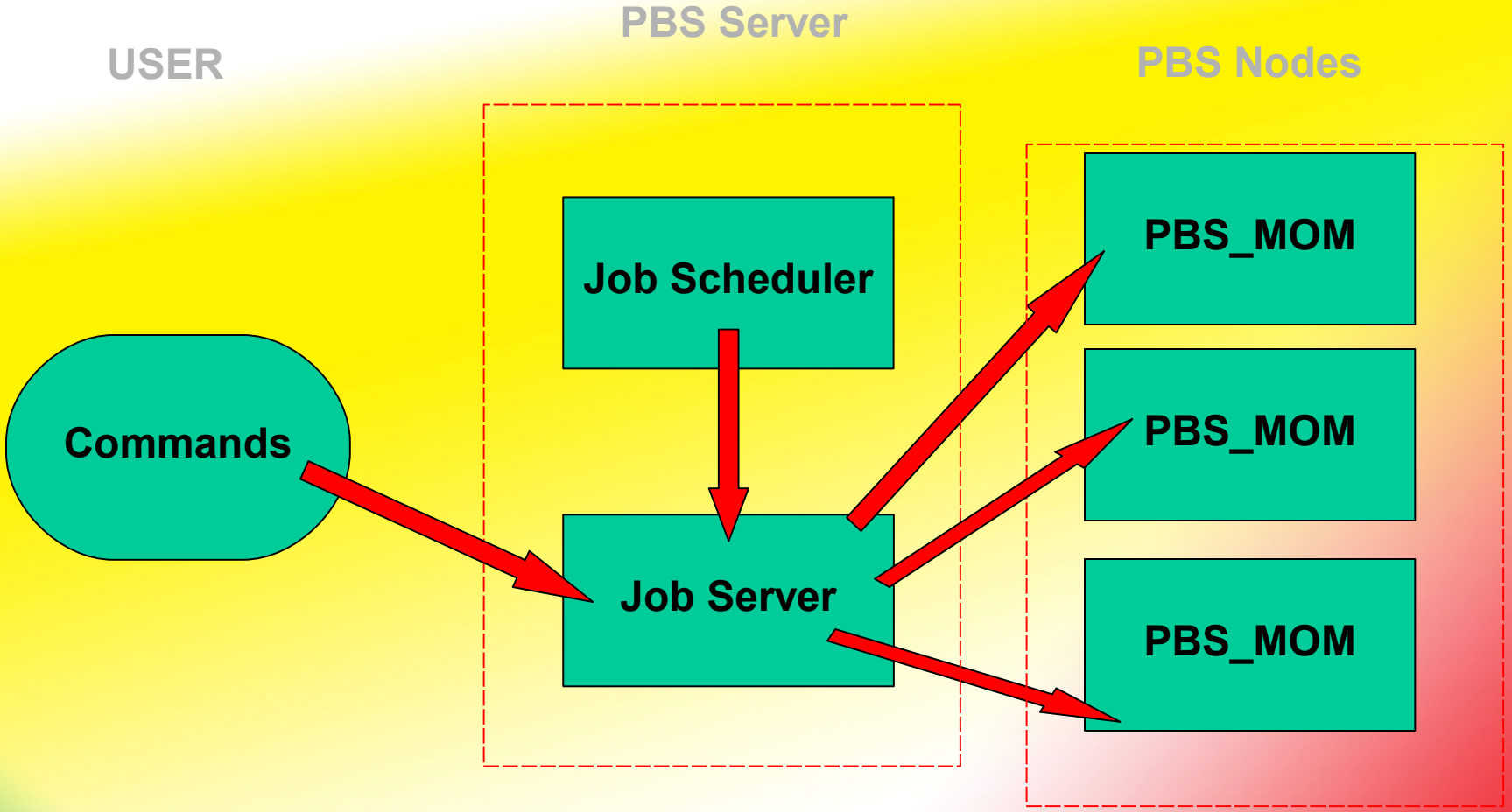
- Queueing System 概述
- PBSpro 概述
  - 組成元件
  - 常用指令說明
- PBS script 撰寫
- PBSpro 環境變數
- PBSpro script 範例說明
  - searial job
  - OpenMP
  - MPI1 & MPI2
- 常見問題

Queuing System 概述

- 公平分配電腦資源，使得資源可以被充分利用
- 根據使用者的屬性分配可用資源
- 監控 node 執行狀態與資源利用狀態
- 排程變更、平衡負載等

PBS Pro 組成元件

- **Commands**
  以命令的方式讓使用者可以 Submit 、 Monitor 、 Modify 和 Delete 欲執行的工作
- **Job Server**
  接收、產生、管理及保護使用者的 Job
- **Job Executor (PBS_MOM)**
  接收 Job Server 給予的 Job ，呼叫對應的程式來執行，完成後將結果回報給 Job Server
- **Job Scheduler**
  負責排程工作、資源分配和 Node 管理



PBS 常用指令 – Control Job

- Submit Job – **qsub**
  Usage ： qsub job_script
- Delete Job – **qdel**
  Usage ： qdel job_id
- Query Job Statue – **qstat**
  Usage ： qstat

PBS 常用命令說明 – qsub (1)

- qsub 將 script 送交 PBS Pro 處理.
- 執行後系統會回傳 job ID 供使用者參考.
- 常用參數:
  - -N name : 指定 job 的名稱.(預設值為 script 檔名)
  - -l resource=xxx : 指定需要的系統資源,包括 CPU 個數,最大執行時間,… 等.
  - -o outfile : 標準輸出檔(就是正常執行時顯示在螢幕的輸出),預設值為 {jobname}.o{jobid},這裡 {jobname} 是 job 的名稱, {jobid} 是 job id .
  - -e errfile : 錯誤輸出檔,預設檔名是 {jobname}.o{jobid}
  - -j (oe|eo) : 合併兩個輸出檔, oe 合併到標準輸出檔, eo 合併到錯誤輸出檔.

PBS 常用命令說明 qsub (2)

- 常用參數（續）：

-q 指定使用的佇列，預設是使用 short。
short : 3days (72 hours) , MAX 16cores
long : 7days (168 hours), MAX 64cores

- 範例：

```
[guest@ge1 guest]$ qsub -N test -o out.txt -e
err.txt -l walltime=1:00:00 -l select=4:ncpus=4
run.sh

3131.ge1   ← 3131 是 Job ID，ge1 是上傳 job 的機器名稱
```

將 run.sh 送入系統預設的佇列，指定輸出檔以及錯誤輸出檔，最長執行時間 1 小時（超過會強行中止），使用 4 台機器，每個機器都需要 4 CORES。

PBS 常用命令說明 qsub (3)

- 有關系統資源選項的設定部份，簡介一些比較常用的格式：
  - walltime=hh.mm.ss 是限制程式最大執行時間，如果超過的話會強制停止。預設值是 3 天。
  - select=N:ncpus=M 選擇 N 台機器，每一台機器需要 M 個 CPU core。預設是只用一台機器的一個 CPU core。
  - select=N:ncpus=M:mpiprocs=M 和前一個例子相同，一樣是選擇 N 個機器，每個機器 M 個 CPU。唯一的差別是 $PBS_NODEFILE 這個環境變數指向的檔案內容，如果沒有使用 mpiprocs 的參數，檔案裡面每台機器只會列一次，如果有 mpiprocs 的參數，每台機器會重覆 M 次，這是用來配合一些 MPI 程式執行參數的使。

PBS 常用命令說明：qstat

- qstat 用來顯示 PBS Pro 相關狀態，有三種模式：
  - Job 模式： qstat <job ID>
  - Queue 模式： qstat –Q <queue>
  - Server 模式： qstat –B <server>
- 三種模式都可以加上 – f 參數來顯示詳細資訊．
- 各模式如果沒有使用 <> 的參數，將顯示所有的資訊，例如單獨執行 qstat 會顯示所有正在執行以及等待的工作．

PBS 常用命令說明 qdel

- qdel 用來刪除 PBS Pro 中正在等待或是執行的工作，執行方式是

  qdel <jobID>

- 可以同時指定超過一個以上的 job ID ，

```
[guest@i90 guest]$ qdel 3131
```

常用指令（LAB）

1. $vim pbs-ping.sh
   #PBS -N [座號]
   ping localhost

2. $qsub pbs.sh

3. $qstat
   $qstat -ans
4. $qdel [job-id]

PBS Script 撰寫

- PBS Script 基本說明：
  - PBS script 就是一般的 shell script
  - #PBS 開始的每一行內容都會被 qsub 取出來當成 qsub 的執行參數。（在原 sge 環境，則是 #$ 開頭）
  - 當 qsub 命令行參數與 PBS script 裡面描述的參數衝突時，以 qsub 命令行參數為優先．
  - PBS Pro 開始執行 job 時，將會在使用者的家目錄執行這個 script ．執行之前會先設定一些環境變數．

PBS Script 撰寫（續）

- PBS Script 基本規則（續）：
  - 當指定使用超過 1 個 CPU 時，PBS script 只在第一個 CPU 啟動，如果要能使用到其他的 CPU 資源，需要執行 MPI 或是 OpenMP 這些能夠提供平行運算的程式庫，否則即使佔用了許多 CPU 資源，實施用到的仍只有一個．
  - 由於 PBS Pro 執行的 script 一般都是以批次執行的方式，因此盡量避免執行需要使用者當場輸入資料的程式．

PBS Script 撰寫（續）

- PBS Script 基本規則（續）：
  - 當執行 OpenMP 或是 MPI 平行程式時，請務必確認實際使用的 CPU 數目與 qsub 標示的 CPU 資源相符，如果使用的 CPU 數目比標示的少，會造成 CPU 資源的浪費，如果使用的 CPU 比標示的少，則無法執行。

PBS Pro 預設的環境變數

- PBS Pro 在執行 PBS script 前，會先設定一些環境變數，使用者可以依需要在 script 中使用，底下列出一些比較常用的變數：
  - PBS_O_WORKDIR : 使用者送出 job 時所在的目錄
  - PBS_O_HOST : 使用者送出 job 時的主機名稱
  - PBS_O_HOME : 使用者的家目錄
  - PBS_JOBID : Job ID
  - PBS_NODEFILE : 一個檔案名稱，檔案內容為 PBS Pro 指定給這個 job 的機器列表

PBS Script 範例 (serial)

- `#!/bin/bash`
- `#PBS -l nodes=1:ncpus=1` ← PBS 資源使用設定
- `#PBS -l walltime=72:00:00`
- `#PBS -l mem=400mb`
- `#PBS -j oe` ← PBS 產生的兩個 Output 檔案整合
- `#PBS -o out.txt` ← PBS 輸出檔名稱
- `#PBS -q short` ← PBS 使用的 Queue
- `cd $PBS_O_WORKDIR`
- `./a.out`

PBS Script LAB (serial)

- `$vim pbs.sh`

  cd $PBS_O_WORKDIR

  ./cpi
- `$qsub pbs.sh`
- `$cat pbs.sh.o[jobid]`
- `$cat pbs.sh.e[jobid]`

PBS Script 範例 (OpenMP)

- #!/bin/bash
- #PBS -l nodes=1:ncpus=4 ← PBS 資源使用設定
- #PBS -l ompthreads=4 ← PBS OMP_NUM_THREADS 環境變數設定
- cd $PBS_O_WORKDIR
- ### export OMP_NUM_THREADS=4
- ./a.out

PBS Script LAB (OpenMP)

- `$vim pbs-omp.sh`

  #PBS -l nodes=1:ncpus=4:ompthreads=4
  #PBS -N [座號]
  cd $PBS_O_WORKDIR
  ./cpi-omp
  ./hello-omp

- `$qsub pbs-omp.sh`
- `$cat pbs.sh.o[jobid]`
- `$cat pbs.sh.e[jobid]`

PBS Script LAB (OpenMP)

- `$vim pbs-omp.sh`

```
#PBS -l nodes=1:ncpus=4
#PBS -N [ 座號 ]
cd $PBS_O_WORKDIR
export OMP_NUM_THREADS=4
./cpi-omp
./hello-omp
```

- `$qsub pbs.sh`
- `$cat pbs.sh.o[jobid]`
- `$cat pbs.sh.e[jobid]`

PBS Script 範例 (MPI1)

- `#!/bin/bash`
- `#PBS -l select=4:ncpus=4:mpiprocs=4`
- `#PBS -l walltime=1:00:00`
- `#PBS -N test_job` ← Job 的名稱
- `cd $PBS_O_WORKDIR`
- `cat $PBS_NODEFILE` ← 列出 $PBS_NODEFILE 內容
- `mpirun -np 16 -machinefile $PBS_NODEFILE ./a.out`

PBS Script LAB (MPI1)

- `$vim pbs-mpi.sh`

  #PBS -l select=2:ncpus=4:mpiprocs=4
  #PBS -N [座號]
  cd $PBS_O_WORKDIR
  cat $PBS_NODEFILE
  mpirun -np 8 -machinefile $PBS_NODEFILE ./cpi-mpi
  mpirun -np 8 -machinefile $PBS_NODEFILE ./hello-mpi

- `$qsub pbs-mpi.sh`
- `$cat pbs.sh.o[jobid]`
- `$cat pbs.sh.e[jobid]`

PBS Script 範例 (MPI2)

- `#!/bin/bash`
- `#PBS -l select=4:ncpus=4`
- `#PBS -l walltime=1:00:00`
- `#PBS -n test_job` ← Job 的名稱
- `cd $PBS_O_WORKDIR`
- `cat $PBS_NODEFILE` ← 列出 $PBS_NODEFILE 內容
- `mpdboot -n 4 --rsh=/usr/bin/ssh --file=$PBS_NODEFILE`
- `mpiexec -np 16 ./a.out`
- `mpdallexit` ← 釋放所有 mpd 呼叫的資源

PBS Script LAB (MPI2)

- $vim ~/.bashrc
  source /ap/****.sh
- $vim pbs-mpi.sh
  #PBS -l select=2:ncpus=4:mpiprocs=4
  #PBS -N [座號]
  cd $PBS_O_WORKDIR
  cat $PBS_NODEFILE
  mpdboot *******
  mpirun -np 8 ./hello-mpi
  mpdallexit
- $qsub pbs-mpi.sh
- $cat pbs.sh.o[jobid]
- $cat pbs.sh.e[jobid]

常見問題

Q:我的工作派送出去一下就結束，沒有正常工作？

A:請瀏覽 [job-name].o[job-id] 與 [job-name].e[job-id]，了解錯誤原因。常見為script內容打字錯誤、執行（資料）路徑打字錯誤。

常見問題

Q:我已經檢查script、工作資料、參數內容無誤，為何仍不能正常執行？

A:若是使用windows編輯再複製到linux cluster作業環境，有時會有檔案格式判斷問題，請執行

dos2unix [file-name]

轉換格式。

Unix2dos [file-name] 可轉換為windows能判讀的格式。

常見問題

Q:我的工作派送後等待很久都未執行？

A:請執行 qstat -ns [job-id] 與 qstat -f [job-id]，查看comment項目，了解工作等待狀態。若是因為資源不足的關係，需等待其他執行工作完成。

常見問題

Q:我的工作似乎已經結束，但是qstat顯示仍存在此工作，無法完成output輸出？

A:若qstat -ns [job-id] 顯示 state 'E' 請稍待5分鐘，若仍未結束，請聯絡計中。

若qstat -ns [job-id] 顯示 state 'R'，但您確認工作已經結束，請聯絡計中。

若qstat -ns [job-id] 顯示 state 'H' 一般為程式本身執行出現錯誤，無法繼續執行。請檢查程式執行步驟是否有問題。確認沒問題而仍無法執行，請聯絡計中。

INFOWRAP

Thanks!

勁智數位科技
Leading @ Extreme Computing