

PBSPro users and nodes

AJ Smit

Table of contents

1 Which nodes are associated with a user?	1
2 How to delete or terminate a user's specific job	2
3 How to terminate a multi-node compute job	3
4 How many jobs are ahead of mine?	4
Bibliography	5

1 Which nodes are associated with a user?

To find active nodes associated with a user in PBSPro, you'd typically use the `qstat` command followed by some parsing to extract the node information. The method to extract node information can vary based on your setup and specific requirements.

Here's a general approach:

1. Fetch Jobs Associated with a User:

Use `qstat` to get a list of jobs associated with a user:

```
qstat -u <username>
```

This will give you a list of jobs associated with the user `<username>`.

2. Parse Job IDs:

From the output, you'd want to extract the Job IDs of interest (typically, the running ones).

3. Get Detailed Information for Each Job:

For each of those Job IDs, you'd then get more detailed information:

```
qstat -f <JobID>
```

This will provide a lot of information, including the nodes the job is running on.

4. Extract Node Information:

From the detailed output, you'd look for an entry that specifies nodes. This might look something like:

```
exec_host = node_name/0+node_name/1+...
```

You'd then parse this to get the list of nodes.

5. Script It:

You can wrap this up in a script for convenience. Here's a simple Bash script that automates the above steps:

```
#!/bin/bash

USERNAME=$1

# Fetch jobs for user
JOBS=$(qstat -u $USERNAME | awk 'NR>5 {print $1}' | grep '^[0-9]')

for JOB in $JOBS; do
    # Fetch node information for each job
    NODES=$(qstat -f $JOB | grep 'exec_host' | awk -F'=' '{print $2}')
    echo "Job: $JOB runs on nodes: $NODES"
done
```

You can save this script, make it executable using `chmod guo=rwx` (on Arch Linux on Lengau) `chmod +x script_name.sh` (on other Linux distributions), and then run it:

```
./script_name.sh <username>
```

Note: The specifics, especially the parsing bits, can vary based on one's exact PBSPro setup and version. Adjust the script accordingly. Always refer to the official PBSPro documentation or local documentation provided by your HPC center for precise details.

2 How to delete or terminate a user's specific job

1. Find the user's running jobs:

Use the `qstat` command to fetch the list of jobs associated with a user:

```
qstat -u <username>
```

2. Extract the Job IDs and delete them:

You can use the `qdel` command to delete specific jobs.

```
qdel -W force <job.name>
```

3 How to terminate a multi-node compute job

Terminating a multi-node compute cluster in a PBSPro environment typically involves deleting or cancelling the job that has been submitted:

1. Using Job ID:

- First, identify the Job ID of the running job. You can use the `qstat` command to list all the jobs you have submitted:

```
qstat -u <your_username>
```

- Once you have identified the Job ID of the job you wish to terminate, you can use the `qdel` command to delete it:

```
qdel <Job_ID>
```

This will terminate the job and free up the resources that were being used across the multiple nodes.

2. Using Job Name:

- If you know the name of the job, you can also use it to delete the job:

```
qdel -N <Job_Name>
```

3. Deleting All User Jobs:

- If you want to delete all the jobs you have submitted, you can use the following command:

```
qdel `qselect -u <your_username>`
```

4. Interactive Mode:

- If you are in an interactive session, you can simply type `exit` or press `CTRL+D` to terminate the session, and the job will be terminated.

The above assumes you have the necessary permissions to delete the job. You should ensure that there is no unsaved work as terminating the job will discard all the unsaved progress. Always be cautious and double-check the Job ID or Job Name to avoid terminating the wrong job. Different clusters might have slightly different configurations, so it might be best

to consult the specific documentation of your environment or get in touch with the system administrator for detailed guidance.

4 How many jobs are ahead of mine?

To find out where in the queue your job is located, you can use the `qstat` command with various options to filter and view the status of the jobs:

1. List all jobs in the queue:

```
qstat
```

This will display all jobs in the queue, showing their IDs, names, usernames, time used, and their statuses (e.g., Running, Queued, etc.).

2. Find a specific job in the queue:

Once you know the Job ID of one's submitted job, you can use this command to display information specifically for that job:

```
qstat <job_id>
```

Replace `<job_id>` with your job ID.

3. Find the position of a specific job in the queue:

You can use the `qstat` command with some Unix commands to find out the position of your job:

```
qstat | grep -B 10 "<job_id>" | head -n 10
```

This command will show you the 10 jobs before your own job in the queue, giving an idea of your job's position relative to others.

4. Customise the display to make the position more apparent:

You can customise the `qstat` output using various options to display specific columns or order the jobs in a particular manner to make it easier to locate your job and infer its position.

For example, to display jobs in the order they are queued:

```
qstat -a
```

The -a option displays more details, and you can then manually locate your job to determine its position in the queue.

The position in the queue might not always accurately represent when your job will start executing because the actual scheduling and execution of jobs depend on various factors, such as job priorities, requested resources, and the scheduler's configuration and policies.

Bibliography