Job Array, large number of (nearly)identical jobs?

How to run job arrays, useful for running groups of (semi) identical jobs?

Very often you might need to run a large amount of jobs that are almost identical, as they maybe doing the exact same analysis on different input data, or running the exact same simulation maybe starting from different initial conditions, or performing a parameter scanning in evaluating a model.

Instead of submitting these runs as individual independent jobs, it's much more convenient - both for you and for the scheduler - to use job arrays.

When you submit a job array, you are giving the scheduler a single script, and indicating a range of indexes that will be used to differentiate the single jobs - called “tasks” - in the array. The indexes are provided in the form of ‘t N1-N2:step’. This will create the series task N1 , task N1+step , task N1+2*step, ..... task N2. If step is omitted the default is 1.

The typical usage is simply ‘t 1-N ’ which will create N tasks with indexes from 1 to N.

At runtime, in the execution environment the system will define for you some environmental variables that can be used to distinguish the tasks in the array and differentiate (if needed) their work. In particular

<table>
<thead>
<tr>
<th>Variable</th>
<th>meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$SGE_TASK_ID</td>
<td>index that differentiate the tasks</td>
</tr>
<tr>
<td>$SGE_TASK_FIRST</td>
<td>index of the first task</td>
</tr>
<tr>
<td>$SGE_TASK_LAST</td>
<td>index of the last task</td>
</tr>
<tr>
<td>$SGE_TASK_STEPSIZE</td>
<td>step size</td>
</tr>
</tbody>
</table>

The standard output and error of array job tasks will be written into different files with the default location $JOB_NAME.o$JOB_ID.$SGE_TASK_ID and $JOB_NAME.e$JOB_ID.$SGE_TASK_ID

If you need to operate on a single task in the queue (for example to delete it), you have to address it using $JOB_ID.$SGE_TASK_ID for example

```bash
[testuser@login-node-1-0 basic-test]$ qsub simple_job_array.txt
Your job-array 2334051.1-10:2 (*job_array_example*) has been submitted
### where the script looks like this
[testuser@login-node-1-0 basic-test]$ cat simple_job_array.txt
#$ -N job_array_example      ### sets the jobname
#$ -t 1-10:2                 ### creates tasks 1, 3, 5, 7, 9
#$ -cwd                      ### starts the job from the submission folder.
#$ -ckpt reloc               ### the task job will be relocated upon suspension.
#$ -q all.q                  ### sets the queue
./run_my_simulation.x < input_run_${SGE_TASK_ID}  ### each task will use a
different input file

[testuser@login-node-1-0 basic-test]$ qdel 2334051.3  ### this will delete only task
3
testuser has deleted job-array task 2334051.3
[testuser@login-node-1-0 basic-test]$ qdel 2334051     ### this will delete the whole
array
testuser has deleted job 2334051
```