Scripting for Advanced Workflows

Jack Deslippe and Zhengji Zhao

Presentation URL: http://goo.gl/s6M8x

Outline

- Scripting basics
- Shell scripting examples for launching multiple MPI jobs
- Chaining jobs
- Managing multiple serial jobs on Hopper and Carver

Presentation URL: http://goo.gl/s6M8x

Scripting Basics

Scripting is a rich and deep topic and can refer to any of the following:

SHELL (BASH, CSH ...)
PERL
Python
PHP
...

Is PERL easier than a random language?

"Perl users were unable to write programs more accurately than those using a language designed by chance." - http://www.cs.siue.edu/~astefik/papers/StefikPlateau2011.pdf

Scripting Basics

Even BASH scripting itself is too rich and deep a topic to cover here.

Scripting Basics

Even BASH scripting itself is too rich and deep a topic to cover here.

Commands to become familiar with for scripting:

awk - Great for manipulating data in columns.
bc - Floating point calculator.
csplit - Split a file into many files by columns.
grep - Search for a phrase in a file.
sed - A file editor. Good for find and replace etc...
...

Scripts for Multiple MPI Jobs

Many applications have a trivially parallelizable layer.

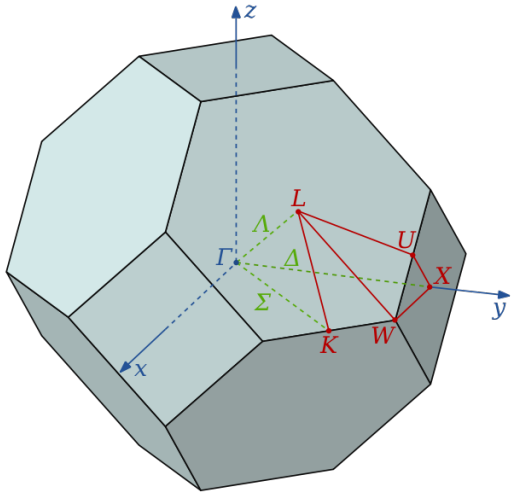
-This layer be exploited by multiple runs of the same executable with slightly different input.

Scripts for Multiple MPI Jobs

Many applications have a trivially parallelizable layer.

-This layer be exploited by multiple runs of the same executable with slightly different input.

For example: Material science codes are often trivially parallelizable over k-points (like s,p,d electrons).



Scripts for Multiple MPI Jobs


```
#!/bin/bash -l

njobs=32

for (( i = 0; i < $njobs ; i++  ))
do

    echo "starting $i"

    mkdir job_$i
    cd job_$i

    # Application Specific Input Modifications
    cp -f ../input ../CD ../*DAT .
    shift=$(echo "($i) / ($njobs)" | bc -l)
    echo "k_grid_shift 0.0 0.0 $shift" >> input

    sed 's/^.*-N.*$/\#PBS -N job-'$i'/' ../qscript_template > tmp
    mv tmp qscript_$i
    qsub qscript_$i

    cd ..

done
```

PARATEC requires the following input files:

input - contains job parameters, including k-points
CD, POT.DAT - Other application input files

```
% cat qscript_template

#PBS -q regular
#PBS -l mppwidth=120
#PBS -l walltime=10:00:00
#PBS -N paratec-template
#PBS -j eo

cd $PBS_O_WORKDIR
aprun -n 120 paratec.x
```

Scripts for Multiple MPI Jobs


```
#!/bin/bash -l

njobs=32

for (( i = 0; i < $njobs ; i++  ))
do

    echo "starting $i"

    mkdir job_$i
    cd job_$i

    # Application Specific Input Modifications
    cp -f ../input ../CD ../*DAT .
    shift=$(echo "($i) / ($njobs)" | bc -l)
    echo "k_grid_shift 0.0 0.0 $shift" >> input

    sed 's/^.*-N.*$/\#PBS -N job-'$i'/' ../qscript_template > tmp
    mv tmp qscript_$i
    qsub qscript_$i

    cd ..

done
```

PARATEC requires the following input files:

input - contains job parameters, including k-points
CD, POT.DAT - Other application input files

```
% cat qscript_template

#PBS -q regular
#PBS -l mppwidth=120
#PBS -l walltime=10:00:00
#PBS -N paratec-template
#PBS -j eo

cd $PBS_O_WORKDIR
aprun -n 120 paratec.x
```

Need to delete those 100 jobs you just created? Try:

```
%  qstat -u <user> | grep sdb | awk '{print $1}' | xargs qdel
```

Scripts for Multiple MPI Jobs

You can pack your multiple jobs into one larger calculation.

(Hopper Only!)

```
% cat qscript_template

#PBS -q regular
#PBS -l mppwidth=120
#PBS -l walltime=10:00:00
#PBS -j eo

cd $PBS_O_WORKDIR
```

```
#!/bin/bash -l

njobs=32
nprocs_per_job=24
let total_procs=($njobs * $nprocs_per_job)

sed 's/^.*width.*$/\#PBS -l mppwidth='$total_procs'/' qscript_template > tmp
mv tmp qscript

for (( i = 0; i < $njobs ; i++  ))
do
   mkdir job_$i
   cd job_$i

   # Application Specific Input Modifications
   cp -f ../input ../CD ../*DAT .
   shift=$(echo "($i) / ($njobs)" | bc -l)
   echo "k_grid_shift 0.0 0.0 $shift" >> input
   cd ..

   echo "cd job_$i" >> qscript
   echo "aprun -n $nprocs_per_job paratec.x &" >> qscript
   echo "cd .." >> qscript
done

echo "wait" >> qscript
qsub qscript
```

Scripts for Multiple MPI Jobs

On Carver, and other systems with mpirun instead of aprun. You need a hostfile.

```
% cat qscript_template

#PBS -q regular
#PBS -l nodes=4:ppn=8
#PBS -l walltime=10:00:00
#PBS -j eo

cd $PBS_O_WORKDIR
```

```
#!/bin/bash -l

njobs=32
nodes_per_job=4 ; PPN=8
let nprocs_per_job=($nodes_per_job * $PPN)
let tot_nodes=($njobs* $nodes_per_job)
sed 's/^.*nodes.*$/\#PBS -nodes='$tot_nodes':ppn='$PPN'/' qscript_template > tmp
mv tmp qscript

for (( i = 0; i < $njobs ; i++  ))
do
   mkdir job_$i
   cd job_$i

   cp -f ../input ../CD ../*DAT . # Application Specific Section
   shift=$(echo "($i) / ($njobs)" | bc -l)
   echo "k_grid_shift 0.0 0.0 $shift" >> input
   cd ..

   echo "cd job_$i" >> qscript
   let lstart=${i}*${nprocs_per_job}+1
   let lend=${lstart}+${nprocs_per_job}-1
   echo "sed -n ${lstart},${lend}'p' < \$PBS_NODEFILE > nodefile" >> qscript
   echo "mpirun -np $nprocs_per_job paratec.x -hostfile nodefile &" >> qscript
   echo "cd .." >> qscript
done
echo "wait" >> qscript
qsub qscript
```

Scripts for Multiple MPI Jobs

Job Arrays (Carver Only):

Arrays of separate jobs that can be controlled by a single mother job.

```
% cat qscript

#PBS -q regular
#PBS -l nodes=4
#PBS -l walltime=10:00:00
#PBS -N myjob
#PBS -j eo

cd $PBS_O_WORKDIR
cd job_$PBS_ARRAYID
mpirun -np 32  paratec.x
```

```
#!/bin/bash -l

njobs=32
let njobs_minus_one=($njobs - 1)

for (( i = 0; i < $njobs ; i++  ))
do

    echo "starting $i"

    mkdir job_$i
    cd job_$i

    # Application specific section
    cp -f ../input ../CD ../*DAT .
    shift=$(echo "($i) / ($njobs)" | bc -l)
    echo "k_grid_shift 0.0 0.0 $shift" >> input

    cd ..

done

qsub -t 0-$njobs_minus_one qscript
```

Scripts for Multiple MPI Jobs

Job Arrays:

```
% qstat -u jdeslip

1793543[].cvrsvc    jdeslip  reg_med  myjob          --    20  20    --  00:10 Q   --

% qstat -t -u jdeslip

1793543[0].cvrsv    jdeslip  reg_med  myjob-0        --    20  20    --  00:10 Q   --
1793543[1].cvrsv    jdeslip  reg_med  myjob-1        --    20  20    --  00:10 Q   --
1793543[2].cvrsv    jdeslip  reg_med  myjob-2        --    20  20    --  00:10 Q   --
1793543[3].cvrsv    jdeslip  reg_med  myjob-3        --    20  20    --  00:10 Q   --
1793543[4].cvrsv    jdeslip  reg_med  myjob-4        --    20  20    --  00:10 Q   --
1793543[5].cvrsv    jdeslip  reg_med  myjob-5        --    20  20    --  00:10 Q   --
...

% qdel 1793543[5]

% qdel 1793543[]
```

Chaining Jobs

Each job starts only after completion of previous job

```
% cat qscript_template

#PBS -q regular
#PBS -l mppwidth=120
#PBS -l walltime=10:00:00
#PBS -N myjob
#PBS -j eo

cd $PBS_O_WORKDIR
aprun -n 120 paratec.x
```

```
#!/bin/bash -l

njobs=32

for (( i = 0; i < $njobs ; i++  ))
do
   mkdir job_$i
   cd job_$i

   #Application specific section
   cp -f ../input ../CD ../*DAT .
   shift=$(echo "($i) / ($njobs)" | bc -l)
   echo "k_grid_shift 0.0 0.0 $shift" >> input

   sed 's/^.*-N.*$/\#PBS -N myjob-'$i'/' ../qscript_template > qscript_$i

   if [ -n $last_jobID ]
   then
      jobID=`qsub -W depend=afterok:$last_jobID qscript_$i`
   else
      jobID=`qsub qscript_$i`
   fi

   last_jobID=${jobID%%.*}
   #last_jobID="${last_jobID}.sdb@sdb" # uncomment this line on hopper
   cd ..
done
```

Scripts for Multiple Serial Jobs

Carver:

A serial queue exists. You can submit multiple serial jobs to this queue.

Hopper:

No serial queue exists. If you submit a serial job with "aprun -n 1 command" you will be charged for a whole node.

aprun is limited so that each "aprun -n 1 command &" in a batch script will require a new empty node.

Scripts for Multiple Serial Jobs

Fortran MPI Job Wrapper.

Each MPI task executes serial "executable" in directory run.rank

```
program mwrapper
include 'mpif.h'
integer :: size, rank, ierr, nargs, i
character (len=50) :: rundir, stdoutfile, executable, args, tmp

args=''
call getarg(1,executable)
do i = 2, iargc()
  call getarg(i,tmp)
  args=trim(args)//' '//trim(tmp)
enddo

call mpi_init(ierr)
call mpi_comm_size(mpi_comm_world,size,ierr)
call mpi_comm_rank(mpi_comm_world,rank,ierr)

write(rundir,"('run.',i3.3)") rank
write(outfile,"('stdout.',i3.3)") rank

call system ('mkdir '//rundir)
call system ('cd '//rundir//';'//trim(executable)//' '//args//' >'//outfile)

call mpi_finalize(ierr)
end program mwrapper
```

Scripts for Multiple Serial Jobs

Fortran Wrapper Example Usage:

```
% ftn mwrapper.f90 -o mwrapper.x

% qsub -I -V -q interactive -l mppwidth=48

% aprun -n 48 mwrapper.x /global/homes/j/jdeslip/paratec.x arg1 arg2
```

Scripts for Multiple Serial Jobs

Fortran Wrapper Example Usage:

```
% ftn mwrapper.f90 -o mwrapper.x

% qsub -I -V -q interactive -l mppwidth=48

% aprun -n 48 mwrapper.x /global/homes/j/jdeslip/paratec.x arg1 arg2
```

-Works well on both Cray systems and Carver.
-Fortran wrapper has very little overhead.
-Can be used with threaded applications.

Scripts for Multiple Serial Jobs

Python MPI Job Wrapper

Each MPI task executes serial "executable" in directory run.rank

```
from mpi4py import MPI
from subprocess import call
import sys

exctbl = sys.argv[1]
args = ""
i=0
for arg in sys.argv:
   if (i > 1):
      arguments = args+sys.argv[i]+" "
   i += 1

comm = MPI.COMM_WORLD
rank = comm.Get_rank()

myDir = "run."+str(rank)
outfile = "out."+str(rank)

cmd = "mkdir "+myDir+" ; cd "+myDir+" ; "+exctbl+" "+args+" > "+outfile

sts = call(cmd,shell=True)

comm.Barrier()
```

Scripts for Multiple Serial Jobs

Example python script and limitations

Usage:

```
% qsub -I -V -q interactive -l mppwidth=48

% mpirun -np 48 python mwrapper.py /global/homes/j/jdeslip/paratec.x arg1 arg2
```

Scripts for Multiple Serial Jobs

Example python script and limitations

Usage:

```
% qsub -I -V -q interactive -l mppwidth=48

% mpirun -np 48 python mwrapper.py /global/homes/j/jdeslip/paratec.x arg1 arg2
```

Limitations:

- Best used on Carver.
- On Cray systems, loading python modules can be slow if using many processors.

Summary

Scripting can improve your productivity using NERSC

Scripting can be used to manage multiple serial, threaded or MPI jobs.

Scripting is too big a topic to fully cover in 30 minutes.

- http://tldp.org/LDP/abs/html/
- http://linux.die.net/Bash-Beginners-Guide/
- http://docs.python.org/tutorial/

Presentation URL: http://goo.gl/s6M8x