DS 7347
High-Performance Computing (HPC) and Data Science
Session 10

Robert Kalescky
Adjunct Professor of Data Science
HPC Research Scientist

May 26, 2022

Research and Data Sciences Services
Office of Information Technology
Center for Research Computing
Southern Methodist University

Outline

Session Question

Slurm

- Resource Selection
- Dependencies
- Arrays

Readings and Assignments

Session Question

Session Question

Why do we use Slurm?

Slurm

Resource Selection

There are many options for selecting specific resources.

- Nodes
- Cores
- Memory
- Process distribution
- Accelerators

Resource Selection

```
#!/bin/bash
#SBATCH -J gpu
#SBATCH -o gpu_%j.out
#SBATCH -p development
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --cpus-per-task=1
#SBATCH --gres=gpu:1
#SBATCH --mem=6G

nvidia-smi

```

Resource Selection

```
#!/bin/bash
#SBATCH -J mpi
#SBATCH -o mpi_%j.out
#SBATCH -p development
#SBATCH --nodes=3
#SBATCH --ntasks-per-node=1
#SBATCH --cpus-per-task=1
#SBATCH --mem=6G

# Build executable ahead of job submission
# module purge
# module load intel/oneAPI-2021
# mpicc -o mpi_get_hostnames mpi_get_hostnames.c
```

Resource Selection

```
14
15  # Setup environment
16  module purge
17  module load intel/oneAPI-2021
18
19  # Run executable
20  srun mpi_get_hostnames
21
```

Resource Selection

```
#!/bin/bash
#SBATCH -J mpi
#SBATCH -o mpi.out
#SBATCH -p development
#SBATCH --nodes=3
#SBATCH --ntasks-per-node=1
#SBATCH --cpus-per-task=2
#SBATCH --mem=6G
#SBATCH --exclude=k001

# Build executable ahead of job submission
# module purge
# module load intel/oneAPI-2021
# mpicc -fopenmp -o mpi_get_hostnames_hybrid mpi_get_hostnames_hybrid.c
```

Resource Selection

```
16  # Setup environment
17  module purge
18  module load intel/oneAPI-2021
19  export OMP_NUM_THREADS=${SLURM_CPUS_PER_TASK}
20
21  # Run executable
22  srun mpi_get_hostnames_hybrid
23
```

Dependencies

- Jobs can depended on others via the `--dependency` flag.
- There are numerous options for how this is done.
- `after` and `afterok` are common.

Resource Selection

```
#!/usr/bin/env bash

for step in {1..10}; do
  if [ ${step} -eq 1 ]; then
    dependency_flag=""
    previous_id="no job"
  else
    dependency_flag="--dependency=afterany:${dependency_id}"
  fi
  dependency_id=($(sbatch ${dependency_flag} --mem=1G --wrap "sleep 60"))
  dependency_id=${dependency_id[-1]}
  printf "Job %s is dependent upon %s.\n" ${dependency_id} "${previous_id}"
  previous_id=$(printf "job %s" ${dependency_id})
done

```

Arrays

- Launch many similar jobs in parallel
- Execute over files, directories, data, etc. via shell scripting

Arrays

```
#!/bin/bash
#SBATCH -J data
#SBATCH -o data_import_%A-%a.out
#SBATCH --array=0-2492              # Loops over all XML files
#SBATCH -p htc
#SBATCH --mem=6G
#SBATCH --cpus-per-task=1

# Load Python3
module purge
module load python
```

Arrays

```
13  # Builds array of all original XML files, which are read-only
14  data_dir="$WORK/data/original_xml/"
15  readarray xml_files < <(ls ${data_dir}*.xml | sort)
16  f=${xml_files[${SLURM_ARRAY_TASK_ID}]}
17  fb=$(basename ${f} .xml)
18
19  # Converts XML to TSV and SQLite3 files
20  export OMP_NUM_THREADS=${SLURM_CPUS_PER_TASK}
21  python3 data_import.py ${f} ${fb}
22
```

Readings and Assignments

Readings and Assignments

Readings

None

Readings and Assignments

Project

- Provide proposals for the following:
  1. Data source
  2. Analysis workflow
  3. Tools for implementing the workflow
  4. Possible performance optimization targets
- Commit to your class repo:
  1. `project/proposal.md`.
- Due 12:00 AM Central, Thursday, June 2, 2022