

Introduction to Sunray

The Local Computer Cluster



Contents

1 Introduction

This small guide will help you access the computational resources that is available to you. If you are new to running calculations on a cluster system, **read this document thoroughly**. Especially section "basic workflow".

WARNING: If you run a job, and allocate more resources than the program is using, that job will be deleted **without warning**, as this is huge waste of time and resources.

2 The Queue System

The Queues

There are number of nodes (computers) available to you, and the way we use them are submitting "jobs"/"calculations" to the queue system. Using this queue system, your jobs are added to a queue so everyone shares the available resources. The sunray cluster is running a job-manager called **SLURM**. In SLURM a queue is called a partition.

There are currently 3 partitions on sunray.

- **coms** Primary queue of about 60 nodes, usable by all.
Intel(R) Xeon(R) 8x CPU X5550 @ 2.67GHz. 48GB RAM
- **sauer** Queue for Stephan Sauer. Consists of 2 Nodes.
Intel(R) Xeon(R) 12x CPU E5-2640 0 @ 2.50GHz. 126GB RAM
- **thinclients** The workstation in C317, mostly for single core calculations for comchem students.

The Queue system

The following SLURM commands will provide you with useful information. Use them.

```
1 sinfo                # Overview of all the partitions
2 squeue               # Overview of all queues
3 squeue -p <part>     # Overview of jobs on specific partition
4 squeue -u <user>     # queue for a specific user
5 scancel <id>         # cancel the job
```

Submission scripts for different programs can be found in `/opt/bin/submit/` for the different programs.

It is very important, if you have jobs running a long time to check it and see that it is behaving like it should. Otherwise it is a waste of computational resources. Especially make sure that the resources the job will use (from the programs input file) is the same as allocated by the queue system (from the submit script).

3 Essential Terminal Commands

You will without a doubt be using a commandline interface when working with sunray. If you are new to using Linux and a command shell here is a list of the essential commands you cannot live without. A good idea is also to get a copy of a commandline cheatsheet for linux.

- **cd** change directory. Change the directory by writing

```
1 cd directory_name
```

if you want to move up on directory the command is

```
1 cd ..
```

- **ls** list current directory. Prints out a list of all the items in the current folder.

- **grep** search for text in file/files. Prints out the search results. If for example you want to find all the optimization steps in a GAMESS output file the command is

```
1 grep "NSERCH: " filename.log
```

or if you have a set of GAMESS calculations running and you want to know which are done

```
1 grep "exited:" *.log
```

- **tail** prints out the last lines of a file. If you are running a calculation and you want to check how far the calculations has gone you can print out the last lines with

```
1 tail filename.log
```

- **less** read current file. If you want to read a log file, sometimes this is not possible by normal text editors because the log files are so big. For this you can use less and navigate the file via terminal.

```
1 less filename.log
```

you exit by pressing "q".

- **vim** if you want to edit files via the terminal you can use vim.

```
1 vi filename.inp
```

using vim like a boss, you'll need a lot of time, but the most important things to remember is to be in insert mode when you want to write (press "i"), and to exit you need to be in command-mode (press "esc" to exit insert mode) and write `:wq` + enter.

4 Accessing Sunray

If you want to work from home the ssh address for sunray is

```
1 sunray.theory.ki.ku.dk
```

4.1 Installing X Server on OS X

To use the X Server on a Mac with OS X 10.8 or newer, you need to install *XQuartz*. You can download it at <http://xquartz.macosforge.org/>. At the time of writing (July, 2015) the current version is XQuartz-2.7.7. Download and install that. Finally you need to restart your computer for the changes to take effect.

5 Folder structure and available software

Some of the available software (most of the software needed for calculations, such as gaussview) is installed in the /opt/ folder. For advanced users; please add the /opt/bin/ folder to your \$PATH.

List users

To get a list of all users on the sunray system write the following in the terminal;

```
1 sunset_user_view
```

Depending on who your supervisor is (be it in the Computational Chemistry course or otherwise), you are going to carry out your calculations in different programs. The different programs requires different syntax and keywords in the input file.

Avogadro

Avogadro is an open source software tool which can prepare input files for many of the programs you'll encounter and can also visualize the molecular structure of the molecule. You will find Avogadro under Application → Science. It is easy to draw molecules (and minimize while you draw using the AutoOpt tool) and to prepare input files using the built in editors.

gaussview

You will use gaussview a lot during this course. It works similar to Avogadro but contains a lot more advanced tools and will only prepare input files for gaussian. To start gaussview open a terminal a write

```
1 gaussview &
```

the "&" character makes sure the terminal is not locked to the program, so you can use it for other things as well.

Inputfile preparation

Avogadro and gaussview are great at drawing the molecules, but if you want to export a structure to another program, or if you have the structure in a XYZ format, you can easily convert it to a GAMESS input file by using **OpenBabel**.

```
1 babel -ixyz filename.xyz -ogamin filename.inp
```

to get the list for input and output formats write

```
1 babel -L formats
```

Only for advanced users: If you are working with a lot of files that needs to be converted from XYZ to a specific input format, you can have a header file **gamess.opt** and convert all XYZ-files in a a folder like this;

```
1 for x in *.xyz; do babel -ixyz $x -ogamin ${x%.*}.inp -xf gamess_opt; done
```

6 Basic Work Flow

Basically; Think Before Submission

Here is a recommended work flow when working with computations in general. Follow this, and the number of recalculations you have to make is guaranteed lower than one who does not follow these recommendations.

A very first important point is to realize what you want to do. You've probably been assigned a molecule so what properties are you interested in? Is it a basis set analysis? Do you need to see the effect of using correlation? Once you have that in place, you can start to think about in which order the calculations will be carried out. A general rule of thumb is that you work your way through one molecule in a serial manner, i.e. no more than one calculation at a time.

Until you have a clear idea about what is going on from start all the way to finish, there is no point to have a lot of calculations running because there will be errors and you'll have to correct them. You don't need to start with the final molecule of interest. Make a small example system so the calculations will run faster and you can check the output and correct errors that may arise. Say you have 10 jobs, each taking 50 hours and all of them fail after 40 because you forgot a keyword. That is 400 hours (17 days!!) of wasted computer time, that others could have used.

A second important point is that you need to be organized. Make folders, sub folders so you are always able to find your data files – it happens that you need to find it later on. Keep a log (be it a document on the computer or a logbook like in a real lab) to help you.

Lastly, when you have all your calculations ready for computation, be considerate about other students. You might need to finish 10000 calculations, but do you need to process them all right now? The case is usually, that no matter how many calculations one have, it is always the collection of data/writing down stuff for the report which take the longest time to finish. Find a sweet spot between what you can process and you make the rest of the crowd pleased.

7 Steno

When to submit on which queue on steno

Queues

- **Kemi3** Intel hello hello hello hello
-