Tutorial on Linux and SUGAR access

Authors: Yunming Zhang, Vivek Sarkar

Resource Summary

Course wiki: `https://wiki.rice.edu/confluence/display/PARPROG/COMP322`

Staff Email: comp322-staff@mailman.rice.edu

Coursera Login: visit `http://rice.coursera.org` and log in via Shibboleth

Clear Login: ssh *your-netid*@ssh.clear.rice.edu and then login with your password

Sugar Login: ssh *your-netid*@sugar.rice.edu and then login with your password

Linux Tutorial visit `http://www.rcsg.rice.edu/tutorials/`

NOTE: This handout contains important information for the rest of the semester. Be sure to refer to it when doing future labs and homeworks.

1 Linux Filesystem Basics

pwd Show the current directory

mkdir DIRECTORY Create a new directory

cd DIRECTORY Change current directory to DIRECTORY

ls List file information in current directory

emacs FILE Use EMACS text editor to create or edit a FILE

cp SRCFILE DESTFILE Copy SRCFILE to DESTFILE

cp -r SRCDIR DESTDIR Copy SRCDIRECTORY to DESTDIRECTORY

mv SRCFILE DESTDIR Move SRCFILE to DESTDIRECTORY

mv SRCDIR DESTDIR Move SRCDIRECTORY to DESTDIRECTORY

rm FILE Delete the FILE

rm -r DIRECTORY Delete the DIRECTORY

2 Other Useful Bash Commands

CTRL+A Go to beginning of the line

CTRL+E Go to end of the line

CTRL+C End a running program and return to prompt

CTRL+R Search command history

CTRL+D Log out or Exit

ArrowUp and ArrowDown Browse history of commands

Tab Auto complete

3 Read the SUGAR FAQ

Before going any further, it is important that you familiarize yourself with the SUGAR system by reading the FAQ at `http://rcsg.rice.edu/sugar/faq`. In particular, it is important that you read the following links:

- **Getting Started on SUG@R.** Click on this FAQ and scroll down to the section on "Login Nodes" to understand the difference between login nodes and compute nodes. Make special note of the following comment:

  > "Any user running intensive computational tasks directly on the login node risks disciplinary action up to and including the loss of their access privileges."

- **Getting a Compute Node.** To request a dedicated *compute node*, you should use the following command (as usual) from a SUGAR login node:
  qsub -q commons -I -V -l nodes=1:ppn=8,walltime=00:30:00
  When successful, it will give you a command shell on a dedicated 8-core compute node for your use for 30 minutes at a time. Your home directory is the same on both the login and compute nodes. NOTE: If you are unable to get a node with the above command, please try remove the "-q commons" option and try again.

- **How do use DrHJ with SUGAR?** You cannot. However, you can run DrHJ on your local computer, and transfer files to SUGAR when you need to run them there for performance timings.

4 HJ Setup on SUGAR

Run the following command on SUGAR to setup the environment for executing HJ and Java programs:
source /users/COMP322/hjsetup.txt

5 Compiling and Running HJ programs on SUGAR

The simplest way to compile and execute HJ programs on SUGAR is via the command-line interface. To compile an HJ program, Foo.hj, type "*hjc [options] Foo.hj*". The following command-line options are currently available for hjc (type "*hjc -help*" for a summary):

-racedet Enable race-detection when program executes (off by default, only works for basic async and finish constructs)

-dcg Output dynamic computation graph as a dot file when program executes (off by default, only works for basic async and finish constructs)

-rt s Compile program for work-**s**haring runtime (on by default, supports all HJ constructs)

-rt h Compile program for work-stealing runtime with **h**elp-first policy (off by default, only works for basic async and finish constructs)

-rt w Compile program for work-stealing runtime with **w**ork-first policy (off by default, only works for basic async and finish constructs)

-classpath ⟨**path**⟩ Search path for class files

-sourcepath ⟨**path**⟩ Search path for hj source files (must include -classpath if this option is used)

-destdir ⟨**path**⟩ Set the location where output classes from hjc should be placed

-version Print version number of the HJ compiler. Please include this version number when reporting any problems to comp322-staff.

To execute a compiled HJ program, Foo.hj, type "*hj [options] Foo [args]*", where *args* are the command-line arguments for your program'. The following command-line options are currently available for hj (type "hj -help" for a summary):

-places ⟨**p**⟩**:**⟨**w**⟩ Set number of places and workers per places. The default value is $p = 1$ and $w =$ number of processors in the system.

For now, you will be working with 1 place, so you can use *-places 1:n* to run an HJ program with n workers. Since a SUGAR compute node has 8 cores, the best value for n will usually be $n = 8$ (which is the default for SUGAR).

-perf=true Output abstract execution metrics when program executes (off by default, only works for basic async and finish constructs)

-fj Use Fork-Join variant of work-sharing runtime (off by default, assumes that program has been compiled for work-sharing execution)

-version Print version number of the HJ runtime system. Please include this version number when reporting any problems to comp322-staff.

-mx ⟨**size**⟩ set max heap size, e.g., -mx 8000M sets the max heap size to 8GB

-classpath ⟨**path**⟩ Search path for class files

-J⟨**arg**⟩ Pass ⟨arg⟩ directly to Java runtime (use with caution)

To download files on SUGAR, you have two options:

1. Download the file using a web browser on any computer, and then transfer (via sftp or scp) the file to your SUGAR account.
2. Use the *wget* command. If you type the command "*wget URL*" in SUGAR, it will retrieve the file from URL into your local directory e.g.,
   wget `http://www.cs.rice.edu/~vs3/downloads/nqueens.hj`

Whenever possible, we will try to make copies of files locally available on SUGAR in /users/COMP322.