SciLifeLab
UPPSALA

Quality of Life

a better terminal experience

2022-09-27

Martin Dahlö
martin.dahlo@nbis.se

Enabler for Life Sciences

SciLifeLab UPPSALA

Terminal improvements

- There are a lot of small tips that will improve your experience greatly.

- This lecture will cover some of them:
  - Navigating the terminal
  - Finding files and contents of files

- Also covered: How to transfer files to/from Rackham

Command history

- Up & down arrow to step through history

SciLifeLab UPPSALA

Command history

- Up & down arrow to step through history
- Ctrl+r to search through previous commands

SciLifeLab UPPSALA

Command history

- Up & down arrow to step through history

- Ctrl+r to search through previous commands
  - Ctrl+r again to search further back

SciLifeLab
UPPSALA

Quicker cursor

- Navigating the command line can be done quicker than with just left and right arrows

SciLifeLab UPPSALA

Quicker cursor

- Navigating the command line can be done quicker than with just left and right arrows

  ctrl+a and ctrl+e places the cursor at the beginning and end of command line respectively

SciLifeLab UPPSALA

Quicker cursor

- Navigating the command line can be done quicker than with just left and right arrows

  ctrl+a and ctrl+e places the cursor at the beginning and end of command line respectively

  alt+**b** to go **b**ack a word, alt+**f** to go **f**orward

Cursor position

- No need to be at end of line when pressing enter
- Only determines where you type or erase

```
$ echo “Position not important”
```

Cursor position

- No need to be at end of line when pressing enter
- Only determines where you type or erase

```
$ echo “Position not important”
```

SciLifeLab UPPSALA

Previous directory

- Remembers previous directory

```
user@rackham5 ~/ $
```

Previous directory

- Remembers previous directory

```
user@rackham5 ~/ $ cd /proj/g2020009
```

Previous directory

- Remembers previous directory

```
user@rackham5 ~/ $ cd /proj/g2020009
user@rackham5 /proj/g2020009 $
```

Previous directory

- Remembers previous directory

```
user@rackham5 ~/ $ cd /proj/g2020009
user@rackham5 /proj/g2020009 $ cd -
```

Previous directory

- Remembers previous directory

```
user@rackham5 ~/ $ cd /proj/g2020009
user@rackham5 /proj/g2020009 $ cd -
user@rackham5 ~/ $
```

Previous directory

- Remembers previous directory

```
user@rackham5 ~/ $ cd /proj/g2020009
user@rackham5 /proj/g2020009 $ cd -
user@rackham5 ~/ $ cd -
```

Previous directory

- Remembers previous directory

```
user@rackham5 ~/ $ cd /proj/g2020009
user@rackham5 /proj/g2020009 $ cd -
user@rackham5 ~/ $ cd -
user@rackham5 /proj/g2020009 $
```

SciLifeLab
UPPSALA

Rsync & SCP

- Copy files between computers
- Similar syntax as cp

```
rsync user@host:/path/to/file  /local/path
```

SciLifeLab
UPPSALA

Rsync & SCP

- Copy files between computers
- Similar syntax as cp

```
rsync user@rackham.uppmax.uu.se:/home/user/t.txt   .
```

SciLifeLab UPPSALA

Rsync & SCP

- Copy files between computers
- Similar syntax as cp
- rsync: -a save modification time, -P show progress

```
rsync -aP user@rackham.uppmax.uu.se:/home/user/t.txt   .
```

SciLifeLab UPPSALA

Rsync & SCP

- Copy files between computers
- Similar syntax as cp
- rsync: -a save modification time, -P show progress
- Copy files either direction

```
rsync -aP user@rackham.uppmax.uu.se:/home/user/t.txt   .

rsync -aP t.txt user@rackham.uppmax.uu.se:/home/user/
```

SciLifeLab UPPSALA

Rsync & SCP

- Copy files between computers
- Similar syntax as cp
- rsync: -a save modification time, -P show progress
- Copy files either direction

```
rsync -aP \
user@rackham.uppmax.uu.se:/sw/courses/ngsintro/linux/qol/aa.fa \
.
```

SciLifeLab UPPSALA

Find

- Find files based on name

```
$ find /path/to/look/ -name nametolookfor.txt
```

SciLifeLab UPPSALA

Find

- Find files based on name

```
$ find /path/to/look/ -name nametolookfor.txt
$ find . -name “*.txt”
```

SciLifeLab UPPSALA

Grep

- Searches content of files

```
$ grep texttofind filetolookin.txt
```

SciLifeLab UPPSALA

Grep

- Searches content of files

```
$ grep texttofind filetolookin.txt

$ grep “text to find” /path/to/files/*.txt
```

SciLifeLab UPPSALA

Grep

- Searches content of files

```
$ grep texttofind filetolookin.txt

$ grep “text to find” /path/to/files/*.txt

$ cd /sw/courses/ngsintro/linux/qol
$ grep MYNAME protein_seq.fa
```

SciLifeLab UPPSALA

Grep

- Searches content of files

```
$ grep texttofind filetolookin.txt

$ grep “text to find” /path/to/files/*.txt

$ cd /sw/courses/ngsintro/linux/qol
$ grep MYNAME protein_seq.fa

$ grep -r “found” filetree/*
```

SciLifeLab UPPSALA

Ctrl+c

- Kills whatever your terminal is currently running
  - Sometimes we want to end programs prematurely, for whichever reason.

```
$ cd /sw/courses/ngsintro/linux/qol
$ python friendly_counter.py
```

SciLifeLab UPPSALA

Ctrl+c

- Kills whatever your terminal is currently running
  - Destroys hanged or non-executing commands
  - Cancel commands that are running for too long
  - Cancel commands that you realize are incorrect

UPPMAX cheat sheet

- https://uppmax.uu.se/support/getting-started/uppmax-cheat-sheet/

SNIC-UPPMAX CHEAT SHEET

www.uppmax.uu.se

Logging in

`ssh -AX user@host`	Connect to host with ssh

Getting Help

`man command`	Read manual for command
`apropos keyword`	Find commands related to keyword

File commands

`ls`	List contents of current dir
`ls -al`	Detailed listing with hidden files
`cd dir`	Go to directory (if dir is not given, go to home dir)
`pwd`	Show the current directory
`mkdir dir`	Create directory
`rm file`	Remove file
`rm -r dir`	Recursively remove directory
`rm -f file`	Force remove file
`rm -rf dir`	Force recursively remove directory
`cp -i file1 file2`	Copy file1 to file2
`cp -r dir1 dir2`	Recursively copy directory
`mv -i file1 file2`	Rename or move file/directory (if file2 is a directory, it places file1 inside it)
`ln -s file link`	Create a symbolic link
`touch file`	Create file or update timestamp of existing one
`command > file`	Write output of a command to a file
`command >> file`	Append (add to end) output to file
`less file`	Show contents of file, with scrolling (quit with 'q')
`head file`	Show 10 first lines of file
`tail file`	Show 10 last lines of file
`tail -f file`	Show file as it grows, starting with 10 last lines
`nano file`	Edit file with a simple command line text editor

File Permissions

`chmod permission file`	Change file permission
`chmod -R perm dir`	Recursively change permission for dir

Ex: Allow read/write/exec for user, r/w for group and r for others:

```
chmod u=rwx,g=rw,o=r file
```

`chmod a-x file`	Ex: Remove execute permission for all

Searching

`grep pattern files`	Find lines in files, containing pattern
`grep -r pattern dir`	Recursively do same as above in dir
`command | grep pattern`	Run grep on command output
`find dir | grep pattern`	Find filepaths matching pattern

Compression

`tar cf file.tar files`	Create tar archive, adding files
`tar xf file.tar`	Extract tar archive
`tar czf file.tar.gz files`	With gzip compression
`tar xzf file.tar.gz`	Extract gzip compressed tar archive
`gzip file`	Compress file with gzip
`gunzip file.gz`	Decompress file with gzip

UPPMAX modules

`module avail`	List available modules
`module load modulename`	Load module

Showing user and project info

`uquota`	Show current user's disk usage
`projinfo`	Show used core hours for current user's projects

View details of a specific project:

```
egrep '^b2011999' /etc/slurm/grantfile
```

Running jobs with the Slurm resource manager

`jobinfo`	Show all running and waiting jobs in the queue
`jobinfo -u user`	Show jobs for specific user
`interactive -A project`	Start interactive job

Start batch job (see user guide on the web for more info):

```
sbatch -A project -t d-hh:mm:ss -n cores \
 -p partition jobscript_file
```

Ex: Running for **7** days on **16** cores (2 nodes) on **node** partition:

```
sbatch -A b2011999 -t 7-00:00:00 -n 16 \
 -p node my_jobscript_file
```

`scancel jobid`	Cancel a single job
`scancel -i -u user`	Interactively cancel all jobs for user

Logging out

```
exit
```

www.snic.vr.se

Using multiple terminals

- Launch and use several terminals for better overview of your work
  - Write your scripts in one terminal, run your scripts in another, check the results in a third…
- Just like you might have several tabs in your browser or have several documents open at once