

Getting faster with Unix/Linux

These are some very useful tips on getting started modeling within a unix/linux framework. It's by no means comprehensive (I hope to keep addition to it), but I think it might help you know what's available, so you know what to search for.

Use <tab>

When you start writing a command or entering a directory, pressing the <tab> button will cause the shell to try to complete the command for you. This can save you a lot of time as you shift around directories.

aliases

Use aliases for commands that you have to type over and over again. I provided a sample .bashrc file that shows how to do this for the "bourne shell" (see [guide_to_start_running_models.pdf](#)), to help you get started. There are also ways to do this in our shells, such as the c-shell. The idea is to create a custom command. For example instead of always typing "ssh -Y -l epg2 hpc.es.its.nyu.edu" I have this alias in my .bashrc file `alias hpc='ssh -Y -l epg2 hpc.es.its.nyu.edu'`. To get to hpc login node, I mere type `> hpc`

Moving data from one computer to another

Here the commands `scp`, `scp -r` and `rsync -a` come in very handy. Use `man` (for manual pages) and look online on how to use these. `scp` allows you to copy from one machine to another. The `-r` option is for directories recursively. `rsync -a` preserves the dates, permissions, etc, and will also not copy things that have already been copied.

Create new xterms

A rookie mistake I often see is that people who'd like to work in multiple terminals will login multiple times. All you need to do is type `xterm &` once you've logged in once. (Make sure to `ssh -Y`, otherwise you can't do this.) This gives you new terminals to work from. In my sample.bashrc I have commands for "nicer looking" xterms: `t1`, `t2`, `t2`, and `t4`.