This PDF file contains an embedded LibreOffice Document, so it can be opened and edited in LibreOffice on Linux

Purpose of System

The system's purpose is to give us a platform for performing memory and processor intensive bioinformatics tasks like denovo assembly and Genotype by Sequencing as well as be a location where the large amount of data can be stored.

Hardware

The system is based on a HPX XS4-2440 system from Thinkmate. It was customised to have a large amount of RAM (96 GB) and Hard drive (4TB) storage. This system has 6x16 GB of RAM for a total of 96 GB. It also has 2 x 4-TB internal HDD x 2-TB external usb HDDs, and a single 1-TB SSD. The drives are set up using 3 RAID arrays, which will be described in more detail under “Data Locations”

HPX XS4-2440 $3,292.00 1 $3,292.00

RAID Configuration
Supermicro SuperWorkstation 7037A-iL - 4x SATA/SAS - 6x DDR3 - 500W [SATA 3, 4 ports] No RAID (*OS) -> 1 x 500GB SATA 6.0Gb/s 7200RPM - 3.5" - Western Digital RE4 WD5003ABYZ
Supermicro SuperWorkstation 7037A-iL - 4x SATA/SAS - 6x DDR3 - 500W [SATA 3, 4 ports] RAID 1 -> 2 x 4.0TB SATA 6.0Gb/s 7200RPM - 3.5" - Western Digital RE WD4000FYYZ

- Intel C602 Chipset - Dual 1-Gigabit Ethernet - Mid Tower - 500W Single Power Supply
- 2 x Quad-Core Intel Xeon Processor E5-2403 v2 1.80GHz 10MB Cache (80W)
- 6 x 16GB PC3-14900 1866MHz DDR3 ECC Registered DIMM
- 500GB SATA 6.0Gb/s 7200RPM - 3.5" - Western Digital RE4 WD5003ABYZ
- 2 x 4.0TB SATA 6.0Gb/s 7200RPM - 3.5" - Western Digital RE WD4000FYYZ
- Samsung 24x DVD+/-RW Dual Layer (SATA)
- NVIDIA Quadro 410 512MB DDR3 (1xDVI-DL, 1xDP)
- Logitech Keyboard K120 (USB)
- Logitech B100 Optical Mouse (USB)
- No Windows Operating System (Hardware Warranty Only, No Software Support)
- Ubuntu Linux 14.04 LTS Desktop Edition (No Media) (Community Support) (64-bit)
- Thinkmate Three Year Depot Warranty (Return System to Depot for Repair)

Barebone	
Memory Technology	DDR3 ECC Reg
North Bridge	Intel C602
Form Factor	Mid-Tower
Color	Black
Memory Slots	6x 240-pin DIMM Socket
Audio	RealTek ALC889 7.1 High Definition Audio with S/PDIF header
Ethernet	Intel 82574L Dual Port Gigabit Ethernet
Power	500W AC power supply w/ PFC
External Bays	4x 3.5" Hot-swap SAS/SATA Drive Bays
Expansion Slots	1x PCI Express 3.0 x16 3x PCI Express 3.0 x8 1x PCI Express 3.0 x4 (in x8) 1x PCI-32
I/O	Nuvoton W83627
Front Panel	Power On/Off button Power LED Hard drive activity LED Network activity LED System Overheat LED
Dimensions (WxHxD)	7.6" (193mm) x 16.7" (424mm) x 20.68" (525.3mm)
Gross Weight	40 lbs (18.1 kg)
SATA 3Gbps AHCI Ports	4
SATA 3Gbps AHCI RAID Levels	0, 1, 5, 10
SATA 6Gbps AHCI Ports	2
Processor	
Product Line	Xeon E5-2400 v2
Socket	LGA1356 Socket
Clock Speed	1.80 GHz

Software

The system is running Ubuntu 14.04 LTS, installed using the BioLinux distro, which has a number of useful bioinformatics packages installed. Some important ones are listed below:

vcftools - Filters VCF files to various specifications.

samtools - SNP call program what works with SAM, BAM and outputs VCF files

bowtie2 - Aligns illumina reads to a reference genome and outputs a sam file GBS scripts have been written to automate this task.

R - Used for statistical processes. We use it for linkage analysis with the use of the onemap library.

python - Used for various scripts that analyze or change the format of files.

velvet - Used for denovo assembly of illumina reads, used with oasis

oasis - Used for denovo assembly of illumina reads, used with velvet

Tassel - Alternate GBS pipeline that doesn't require a reference sequence. Also used for Linkage Disequilibrium analysis with data from association mapping.

http://ngsutils.org/installation/

FastQC - Used to perform quality statistics on GBS data. Used in the GBS pipelines.

fastStructure - Used for determining the population structure of a data set for Association Analysis.

tophat/cufflinks - This software package does the alignments for RNA-Seq analysis. See http://cole-trapnell-lab.github.io/cufflinks/

Data Locations

As described under the “Hardware” section, BIOnic has 2x4-TB internal HDDs, 2x2TB USB HDDs, and 1x1-TB SSD.

RAID

The 4TB HDDs are both divided into two partitions, and partitions from either drive are combined together in two RAID0 arrays. These arrays have no data-redundancy, but because they span two drives, they can transfer data twice as fast (because both drives can write at the same time). Additionally, the two 2-TB USB HDDs are also combined into a RAID0 array. These RAID arrays are called m0, m1 and m2, but you don't have to worry about them unless something goes wrong.

Project Folders

All of our drives are mounted in the /mnt/ directory, which is where you'll spend most of your time. There are 4 imporant folders here, each with a special purpose:

WorkingDir: this is a folder on the SSD where we will do the majority of our work. This folder is used because the SSD is much faster for reading/writing data than the other drives. While the other folders in /mnt/ are actually mounted drives, this folder is just a symlink to a space on the same drive the rest of the system is running on.
Inside the WorkingDir, I typically make a different folder for each plant type we are currently working on, and then different folders for each analysis we are running. As soon as results are completed, I clear them out to make room on the limited drive.
ProjectArchives: this folder is one of the RAID0 arrays from the internal HDDs. It contains 4TB of space. This drive is used to store entire projects when they are not being worked on any more. Typically, projects can be moved directly from the WorkingDir to project archives. The folder structure should be set up similarily tow WorkingDir.
DataArchives: this folder is the other RAID0 from the internal HDDs. It contains the raw data from previous projects. Data files for reference genomes and sequencing files should be stored here in a compressed (.tar.gz) format for later use. It is important that new data is saved here immediately to ensure it is saved, even for projects that are still being processed in the WorkingDir.
backups: this folder is the RAID0 array for the external USB HDDs. Ideally, the goal was to make this folder hold a backup of the ProjectArchives and DataArchives folders. Unfortunately, because the drive is the same size as the drives it wants to back up, this was not set up. At the time of writing, there is no automated backup solution. Instead, I selectively added what I believe to be the most important files. Fixing this to create a long-term solution is a high priority.

Git Repo

Important/Reusable scripts are saved to a git repo in /mnt/workingDir/git_repo. This repo has scripts organized into different folders for different use cases. The repo is also stored on github at the following URL: https://github.com/chibbargroup/CentralRepository

Processes.md

This file in the git repo deserves a special note. Processes.md contains the notes I made for each process I have had to perform. I have attempted to take through, step-by-step notes in case anything needs to be reproduces, or in case we need to include any steps in a report. The file is saved as a markdown .md file. It can be edited in any plain text editor (Atom has a good markdown extension, along with vim), and can be viewed as a user-friendly html page. To display the information in a browser, use the grip command (grip /mnt/WorkingDir/git_repo/Processes.md)

SSH

We access BIOnic by SSHing in through the client computer. BIOnic's current IP address is 10.65.78.37.

SSH Tips and Tricks

Because BIOnic is run as a headless system, we do all commands using SSH connections. Here is some advice that makes things much easier

Tmux

tmux is terminal multiplexing software. It's main benefit is that you can start a terminal session on BIOnic, run some long-lasting programs, and then disconnect and go home for the weekend. Ordinarily, any disconnection would terminate the entire program. With tmux, you can re-attach on Monday like you never left. It also has some other useful functions, like split panes. It is pretty ugly at first, but there are a lot of tutorials out there that explain how you can customize it into something usable.

Terminal Alises

I used a number of terminal aliases to help run common commands. For example, I added an alias to automatically log me in to the remote system, by typing BIOnic. You will likely find some other common tasks that can be turned into aliases.

X11 Forwarding

Even though we are accessing BIOnic through an SSH terminal, you can still run full GUI programs through X11 forwarding. If you run ssh with the -X parameter, you can start programs from the command line, and they will forward the interface onto the ssh client. For example, you can run “firefox” to launch a browser window that will download files onto BIOnic, or you can run Pycharm or other code editors to edit files on BIOnic with a nice interface.

Mounting the Filesystem

You can use sshfs to mount BIOnic's file system on your client computer, so you can more easily navigate it's files. I would mount it into /mnt/BIOnic. Alternatively, you can use FileZilla to view and transfer files.

SSH From Home

BIOnic's IP address isn't acessible from the outside internet, but it is still possible to log in to the system from outside the schools' network if you need to check on a process over the weekend. The trick that I would use is to log in to one of the University's student servers (ex, nsid@tuxworld.usask.ca), and from there you can SSH into BIOnic's IP. In effect, you are forwarding your home computer's actions to tuxworld, which then forwards them to BIOnic from inside the network.

Df and Du

I have found df and du to be two very useful functions for tasks here. “du -hd 1” will list the file sizes of all files/folders in the current directory. “df -h” lists the size and free space on all drives attached to the system. Because we deal so often with large files, you may need to use these commands to track down unneeded files to save space (although the new drives we recently attached may make this less necessary).