Next-Generation Sequencing Applications at GACRC

Georgia Advanced Computing Resource Center

University of Georgia

Suchitra Pakala

pakala@uga.edu

Overview

- It's a brave new world – NGS and its Applications
- Hardware, Software, Databases available at GACRC
- NGS project: Logistics and resource considerations
- Best practices, common mistakes, troubleshooting and getting help from GACRC

It's a Brave New World

- First Human genome with annotation – 2006.
  - Price tag – more than $3 Billion
- Within one decade, efforts underway:
  - Genomics England's – 100,000 Genomes Project
  - BGI – Three Million Genomes Projects: "Million Plant and Animal Genomes Project," "Million Human Genomes Project" and "Million Micro-Ecosystem Project."

So, how did this explosion happen?

- A lot of things came together… Rapid advances in:
  - Sequencing technologies
  - Software
  - Hardware

… So, how did this explosion happen?

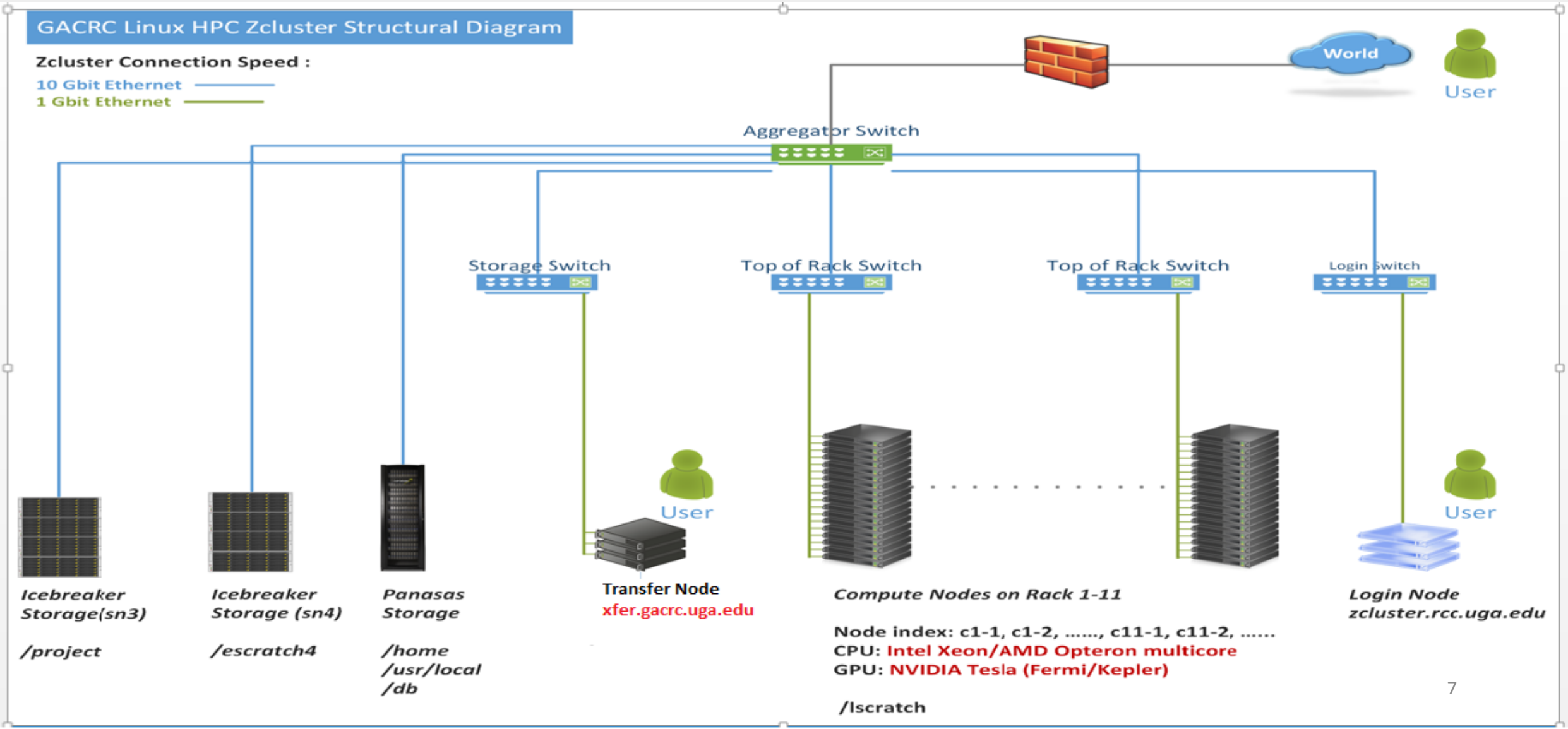
- Next Generation Sequencing (NGS)
  - Illumina, PacBio, etc.
  - Cheap, Fast, High throughput, Accurate, Accessible
- "Large" applications, initiatives, and necessary software
  - Cancer genomics, Microbiome, etc.
  - Rapidly evolving software for every application
- Infrastructure development
  - High Performance Compute Clusters (HPCs), Cloud computing
  - Cheaper and faster options for storage and computation
  - Increasing bandwidth for data transfers

Infrastructure at GACRC

- Two High Performance Compute Clusters
  - Zcluster
  - Sapelo

- Hundreds of Software and Databases installed and actively maintained

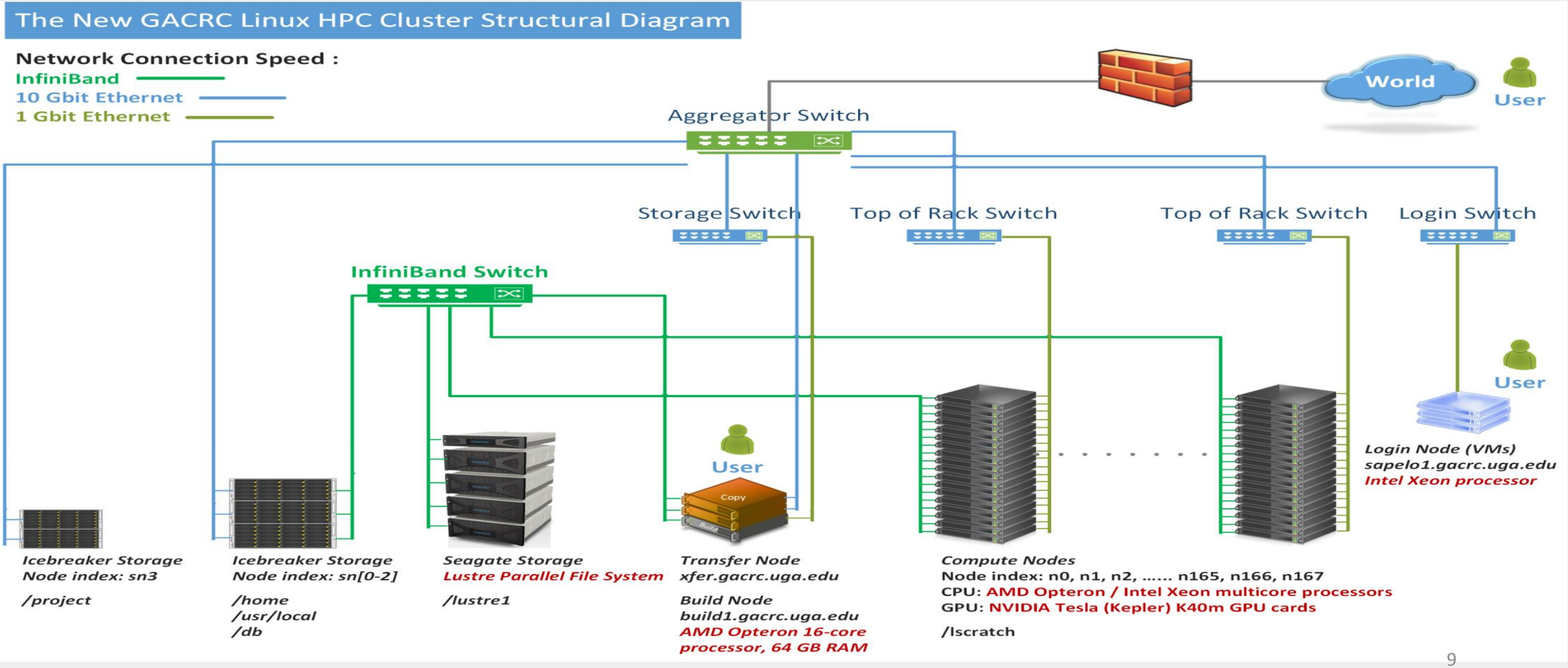
Zcluster (Separate training sessions available)



Zcluster resources

- 230 compute nodes (2600 compute cores), 32 with InfiniBand connectivity.
- Four 8-core, 192GB high-memory compute nodes
- Ten 12-core, 256GB high-memory compute nodes
- Two 32-core, 512GB high-memory compute nodes
- Six 32-core, 64GB high-memory compute nodes
- One NVIDIA Tesla S1070 with four GPU cards (4 x 240 = 960 GPU cores)
- One NVIDIA Tesla (Fermi) C2075 GPU processor (448 GPU cores)
- Nine NVIDIA Tesla (Fermi) M2070 GPU cards (9 x 448 = 4032 GPU cores).
- 32 NVIDIA Tesla (Kepler) K20X GPU cards (32 x 2688 = 86016 GPU cores).

Sapelo (dedicated training sessions available)



Sapelo resources

- 112 compute nodes with AMD Opteron processors (48 cores and 128GB of RAM per node)
- Four 48-core 256GB RAM nodes with AMD Opteron processors
- Six 48-core 512GB RAM nodes with AMD Opteron processors
- One 48-core 1TB RAM node with AMD Opteron processors
- Two 16-core 128GB RAM nodes with Intel Xeon processors and 8 NVIDIA K40m GPU cards each (n48, n49)

Zcluster vs Sapelo

- **Zcluster** – Legacy system. Lesser computational power. But, has most Bioinformatics software installed and ready to use.

- **Sapelo** – New system. More computational power. But, only a fraction of the Bioinformatics software are installed. More in progress.

Software available on GACRC clusters

https://wiki.gacrc.uga.edu/wiki/Software

GACRC

Navigation

Main page
Software
Software by Category
KB (internal)
FAQ
Help

Toolbox

Special pages
Printable version

Software

Installed Software List

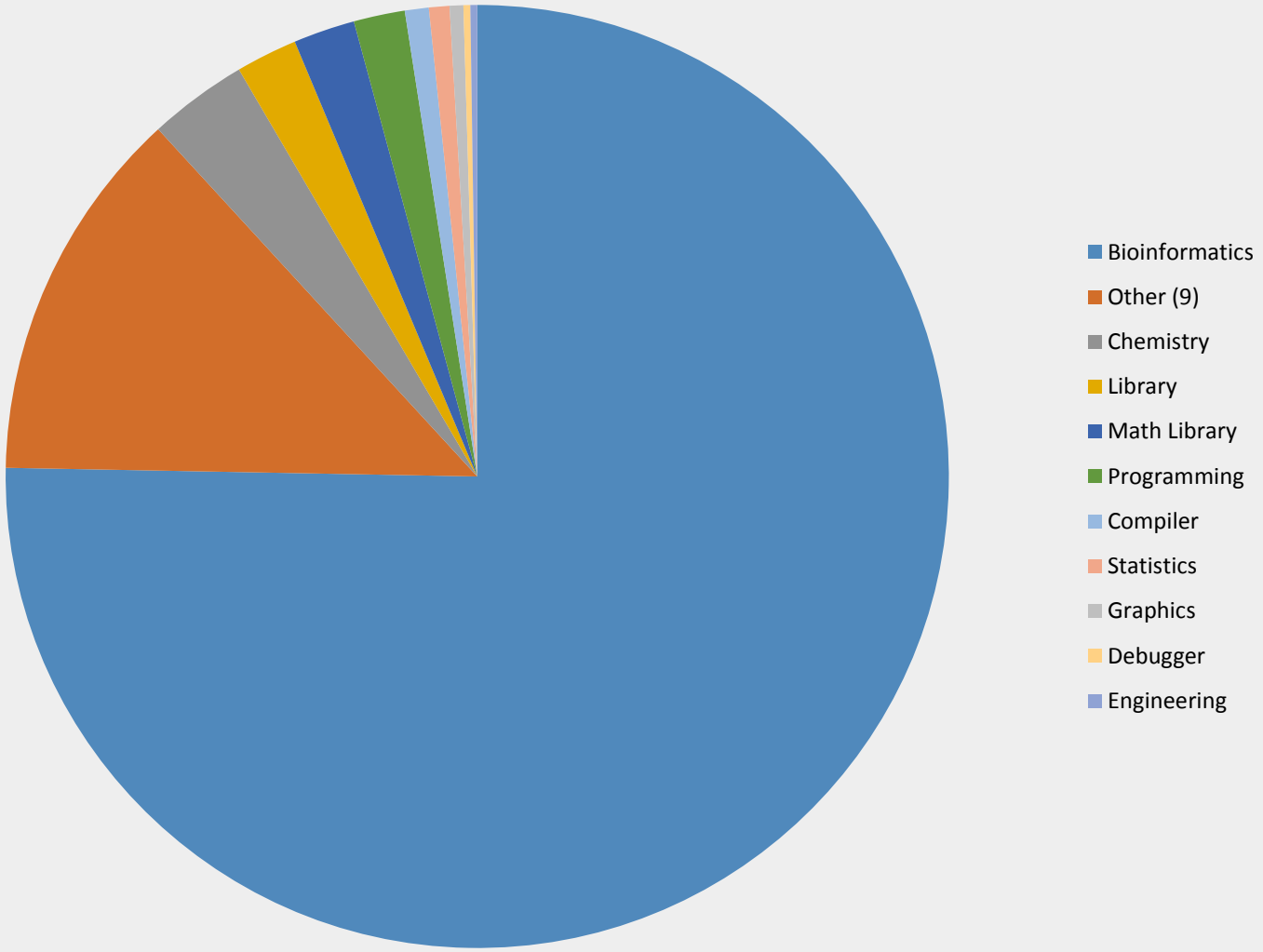
Last updated **2016-10-20**.

Note: Click on the icons to the right of the column headings to sort the table.

0-9 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Name	Version	Category	Cluster
454 Software		Bioinformatics	Zcluster
a5-miseq	20140604	Bioinformatics	Zcluster
aaarf	1.0.0	Bioinformatics	Zcluster
ABINIT	6.12.1	Chemistry	Zcluster
ABySS	1.9.0	Bioinformatics	Sapelo
ABySS	1.9.0	Bioinformatics	Zcluster
ACT	03/27/2013	Bioinformatics	Zcluster
ADMIXTOOLS	1.1	Bioinformatics	Zcluster
Admixture	1.22	Bioinformatics	Zcluster
AFNI	2013.04.26	Other	Zcluster
Allmaps	0.6.6	Bioinformatics	Sapelo
ALLPATHS-LG	47907	Bioinformatics	Zcluster

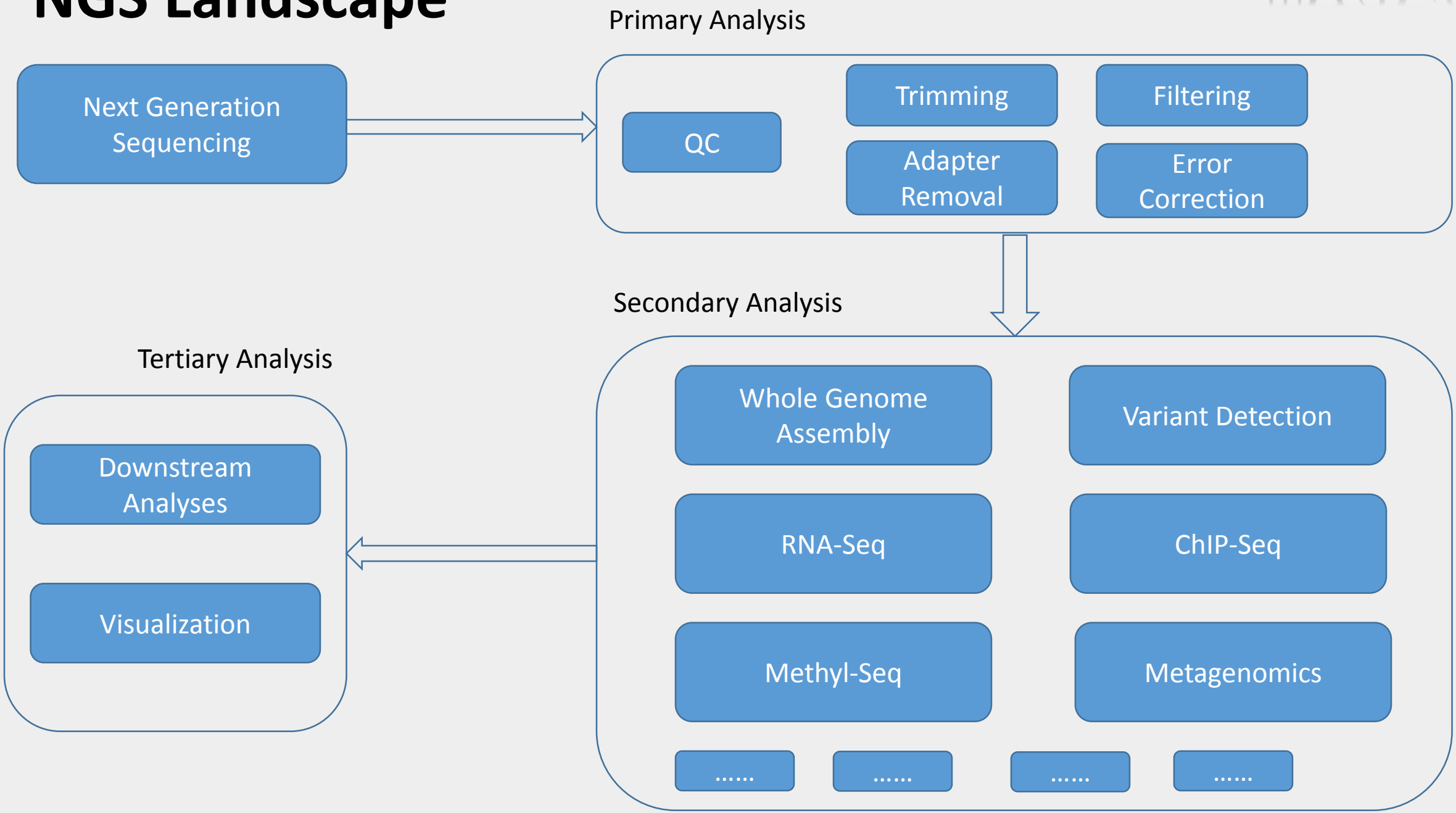
Lion’s share: Bioinformatics software



NGS related software on GACRC clusters

- Zcluster – 474 software in the “Bioinformatics” category
- Sapelo – 169 software in the “Bioinformatics” category
- NGS related software, loosely classified by their function:
  - Pre-processing of sequence data (Primary Analysis)
  - Application specific processing (Secondary Analysis)
  - Analysis and Visualization (Tertiary Analysis)

NGS Landscape



NGS related software on GACRC clusters

1. **Pre-processing of sequence data**

   - **Quality Control:** FastQC
   - **Trimming / Filtering:** Trimmomatic, FastX-Toolkit, TrimGalore, Cutadapt
   - **Error Correction:** proovRead
   - **K-mer analysis:** Jellyfish, KmerGenie
   - **Read merge:** FLASH
   - **Sequence Utilities:** Bamtools, bcl2fastq, BEDops, BEDTools, CGAT, deepTools, gatk, GDC-client, GenomeTools, htseq, IGVTools, MrDemuxy, NGSUtils, NxTrim, picard, PRINSEQ, pybedtools, RSeQC, samtools, SRAToolkit, UCSC, vcftools

NGS related software on GACRC clusters

2. Application specific processing

- **Alignment**: BBMap, Blat, Bowtie2, bwa, DIAMOND, HISAT2, SHORE, STAR, Tophat
- **Whole Genome Assembly**: ABySS, Allmaps, cap3, circlator, gam-ngs, HipMer, IDBA, Meraculous2, Ray, sga, SOAPdenovo2, SPAdes, Velvet, VelvetOptimizer
- **Metagenomics**: Graftm, Kraken, MaxBin, MEGAHIT, MetaVelvet, MOCAT, MOTHUR, QIIME, USEARCH, VSearch
- **Annotation**: Prokka, TransDecoder
- **RNA-Seq**: Cufflinks, DEPICT, Kallisto, READemption, RSEM, Salmon, Scripture, StringTie, Trinity
- **ChIP-Seq, Methly-Seq, Other**: Bismark, HOMER, MethPipeline, Methylpy, MOABS, PeakRanger
- **Variant Analysis**: BreakDancer, FreeBayes, GCTA, Haploview, platypus
- **Validation**: BUSCO, QUAST, Transrate
- **Pac-Bio related**: canu, CluCon, FALCON, PBSuite, SMRT-Analysis, wgs

NGS related software on GACRC clusters

3. Analysis and Visualization

- AStalavista, CytoScape, IGV, MEGAN
- Several tools listed under “Sequence Utilities”

- Most of the software listed above are available on both clusters

- Additionally, many more software are available on Zcluster:
  - Pre-Process: Quake, NGS QC Toolkit, ngopt
  - Alignment: Stampy, MapSplice2
  - Assembly: PASA, PANDASeq, PacBioToCA, Oases, MOSAIK, MIRA 3, MaSuRCA, MAQ
  - Annotation: SNAP, RATT, MAKER
  - Analysis and Visualization: Mauve, Circos

Databases Available

- NCBI Blast databases
- Uniprot DB
- Bowtie Indices
- RNA-Seq reference sequences and annotation files
- Human, Mouse, Dog DB
- Pfam
- … etc. under /db and https://wiki.gacrc.uga.edu/wiki/Bioinformatics_Databases

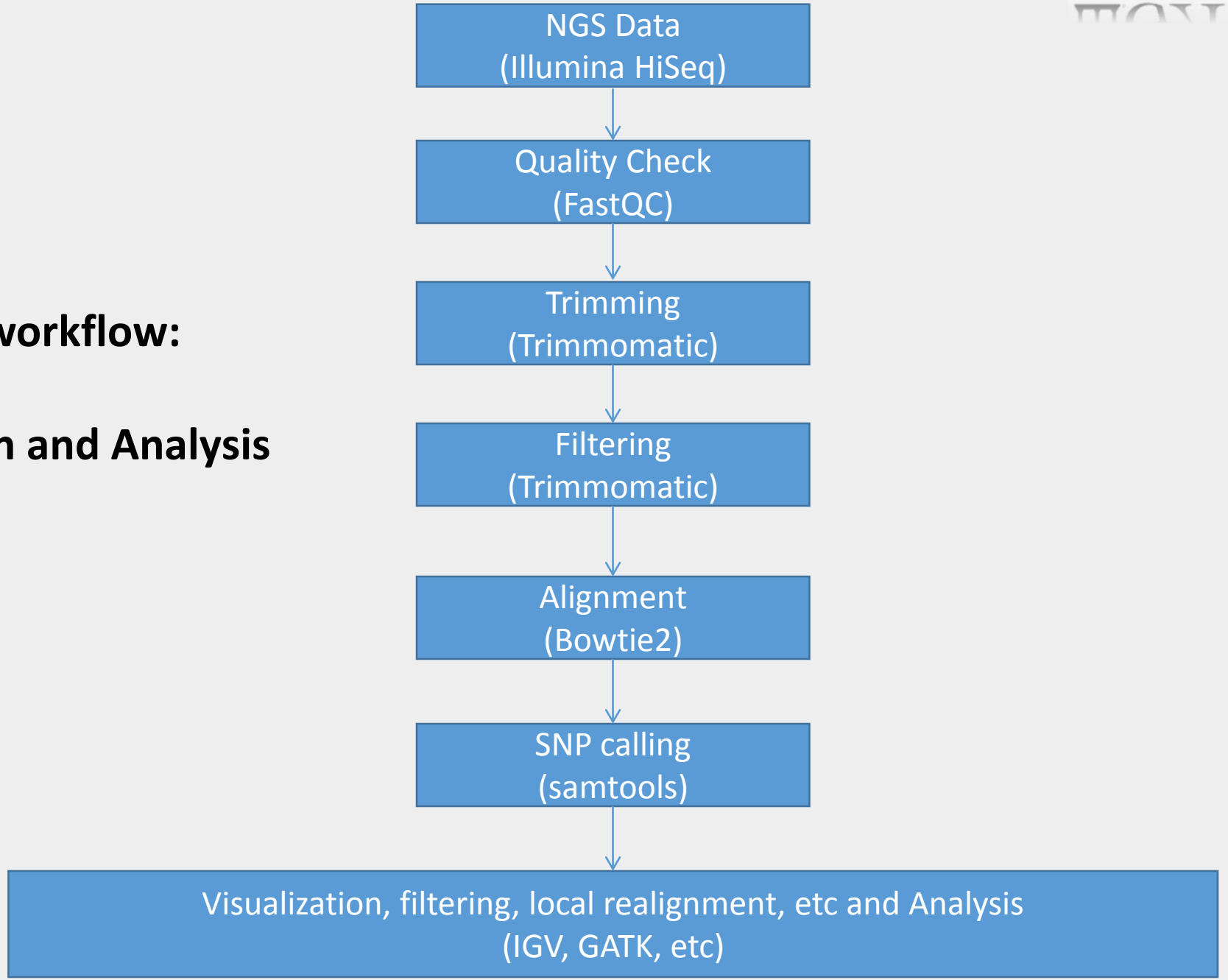
Installation and Maintenance

- Software/Databases will be installed/hosted or updated upon receiving request from users of the cluster(s)

- Database updates are done periodically. For example, Blast databases are updated once in two months.

An example workflow:

SNP detection and Analysis



NGS Project: Logistics and resource considerations

Due to the size of NGS data, several issues need to be considered during the life of a project :

- Data transfer from external machines to GACRC clusters
- Storage space
- Memory needed for various stages of data processing
- Number of compute nodes needed
- Time to completion

NGS Project: Logistics and resource considerations

- Data transfer (think tens to hundreds of GB of data):
  - Speed – your computer to cluster (in/out campus)
  - Data integrity – number of files, checksums (eg. md5sum)
  - Designated nodes to be used for transfer

- Data Storage:
  - Space – how much do you need?
    - Through the life of the project
    - Factor in several runs. Varying parameters. Competing software.
  - Location – home or project area / scratch
  - Duration – different locations have different time limits
    - Expect to run, review, repeat several times before satisfactory results
    - Write up and publication; deposit to public repository

NGS Project: Logistics and resource considerations

- Memory – how much is needed?
  - What software will be used?
    - Do they have established memory profiles/guidelines?
    - If not, use a small subset and observe consumption
  - Need this amount continuously?
    - Some programs have a very high peak memory usage at a certain stage in its workflow. Can that step be isolated?

- Number of nodes (in combination with memory)
  - Few nodes with high memory
  - Large number of nodes with little memory
  - Large number of nodes with high memory… hmmm….

NGS Project: Logistics and resource considerations

- Time to completion
  - Think about overall processing time needed when a job is submitted (will jobs be terminated if they go over?)
  - Remember, these are shared resources. Hundreds of UGA students and staff access them on a daily basis

NGS Project: Logistics and resource considerations

- What happens if estimated storage / memory / nodes / time are wrong?
  - Over estimated ? Wastage of limited resources that are shared by a lot of people
  - Under estimated? Job may hang due to insufficient resources allocated (no automatic dynamic expansion of resources)
  - It is important to estimate all required resources within a reasonable range

NGS Project: Logistics and resource considerations

- So, how do we estimate resources in a reasonable way?
  - Think about the overall project workflow and break it down into:
    - Primary Analysis
    - Secondary Analysis
    - Tertiary Analysis
  - Estimate resources needed for each step

- Run those steps separately, by submitting jobs to appropriately configured resources

Resource estimation

- **Primary Analysis** (Read quality control and pre-processing)
  - Trimming, filtering, de-duplication, etc
  - Typically, these need large storage, large number of nodes for processing, and require low memory.

- **Secondary Analysis** (Application specific data processing)
  - Genome assemblies, variant detection and comparative genomics, etc
  - Typically, these can easily multiply the amount of storage needed, some steps may need large number of nodes with low memory requirements, and some steps may need few nodes with high memory.
  - Identify each step and appropriate tool to be used, and estimate resources separately.

- **Tertiary Analysis**
  - Statistics. Visualization. Science.
  - Typically, no additional storage is needed, may need low to medium memory for applications such as visualizing genomes with several tracks for reads mapped, etc.

- Check GACRC wiki page of each software and review notes regarding running the program

Resource estimation

Examples of important notes available on GACRC wiki pages regarding running programs:

- Velvet
  - “velvet is compiled in multi-thread (compiled with 'BIGASSEMBLY=1' 'LONGSEQUENCES=1' 'MAXKMERLENGTH=99' 'CATEGORIES=62' 'OPENMP=1'). some long reads causes segment fault with high categories (e.g. CATEGORIES=99), we suggest using the fitting categories and kmer version for less memory.”
  - “Velvet needs large memory to run.”
- Oases
  - “Oasis uses velvet program, and velvet needs a large memory. Please refer to Velvet for details and contact us to get permission to run on a large memory queue.”

Resource estimation

Examples of important notes available on GACRC wiki pages regarding running programs:

- Trinity
  - “Mostly trinity needs to run at large memory queue, large memory queues are permission only. Contact us to get permission”
  - “Here is a post for memory estimates. For a 4 billion base mouse, it uses about 50 GB memory at peak.”
  - “Do not ask for more than 12 CPUs at the command and double the quantity of requesting CPU from queue. e.g. at the following, command ask for 8 CPU and at the header, it asks for ppn=16.”
- STAR
  - “Due to memory limits, <Nthreads> in runThreadN suggests be 2 or 4”
- MIRA
  - “MIRA uses lots of memory for the job. Please run following to estimate how much memory needed for the job”

Tying it all together

- Identify storage space needed through life of project
- Determine location on cluster
- Ensure availability

- Transfer data to Cluster
- Use designated transfer node
- Verify checksums

- Identify application workflow
- Review GACRC wiki pages and estimate resource requirements

- Review GACRC wiki pages and prepare job submission scripts
- Obtain permission to use resources beyond default limits

- Submit jobs
- Monitor progress of jobs
- Monitor resource utilization

- Seek GACRC help if errors related to compute environment or if resource utilization exceeds expectations

Limitations

- Is there a limit to the number of jobs that can be submitted or resources that can be utilized by a single user? How about a lab?
  - Storage space: Yes. Limits imposed. Different for project area (lab based) and scratch area.
  - 75 – maximum number of jobs that can be submitted.
  - Resource utilization – first come first served (as soon as requested resources become available)

- Follow ethics and best practices

Ethics and Best Practices

- On cluster, you are not alone... Each user is sharing finite resources, e.g., CPU cycles, RAM, disk storage, network bandwidth, with other researchers. *What you do may affect other researchers on the cluster*.

- 6 rules of thumb to remember:
    - NO jobs running on login node
    - NO multi-threaded job running with only 1 core requested
    - NO large memory job running on regular nodes
    - NO long job running on interactive node
    - NO small memory job running on large memory nodes
    - Use the copy node for file transfer and compression

Tips / Common mistakes / Troubleshooting

(assuming a new user to linux, script writing, job submission to cluster, etc):

- Guard against unintended spaces, line breaks in script
- Avoid spaces in file names
- When referencing locations of files, use full paths. Check if files are readable from location where command/script was run
- Check permissions on files and directories. Make sure they are writable
- Practice with small datasets and estimate requirements. Any trouble due to under-estimating storage, time, memory, cpu needed? Seek help from GACRC
- In case of errors, review error files and log files generated by the software

Getting help

GACRC installs and maintains software. No explicit support for data processing and analysis. If software throws errors such as missing libraries or if there are questions regarding the environment:

- Please send a descriptive message about the issue. Make sure to include the following:
  - Full command line used. If script, send full path to the specific script
  - Error message. If software produced an error log, send path to that file. Else, provide descriptive text of the error.
  - Working directory from where command or script was run

- Make sure that input files that the command or script reference are available

Relevant training sessions and resources

- Zcluster
- Sapelo
- NCBI Blast
- Linux I and II
- Python I and II

- https://wiki.gacrc.uga.edu/wiki/Training
- https://wiki.gacrc.uga.edu/wiki/Software
- https://wiki.gacrc.uga.edu/wiki/Systems
- https://wiki.gacrc.uga.edu/wiki/Bioinformatics_Databases

- To you! For your patience.
- To GACRC team:
  - Yecheng Huang
  - Shan-Ho Tsai
  - Zhuofei Hou