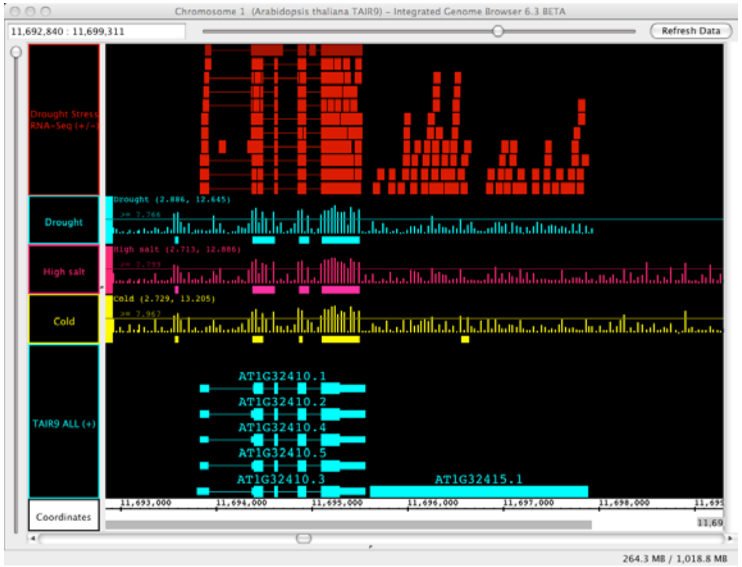
BIOTECHNOLOGY
NC STATE UNIVERSITY

BIT 815: Analysis of Deep DNA Sequencing Data

Overview:

This course covers methods for analysis of data from high-throughput DNA sequencing, with or without a reference genome sequence, using free and open-source software tools with an emphasis on the command-line Linux computing environment and some practice with CLC Genomics Workbench



Lecture Topics:

- Types of samples and experiments
- Experimental design and analysis
- Data formats and conversion tools
- Alignment, de-novo assembly, gene expression analysis, and chromatin studies
- Computing needs and available resources
- Annotation, summarizing and visualizing results

Website: www4.ncsu.edu/~rosswhet/BIT815/

Labs:

The class meets in a computing lab, and each class meeting combines lecture with exercises intended provide students with hands-on experience in managing and analyzing datasets from high-throughput sequencing instruments. Computing exercises will use a Linux machine image on the Virtual Computing Laboratory with pre-installed datasets and software tools, or CLC Genomics Workbench. Alternatives for access to adequate computing resources are discussed and demonstrated.

Introduction to the course and to each other

- background in biology, computing, and sequencing
- experiments of interest to participants

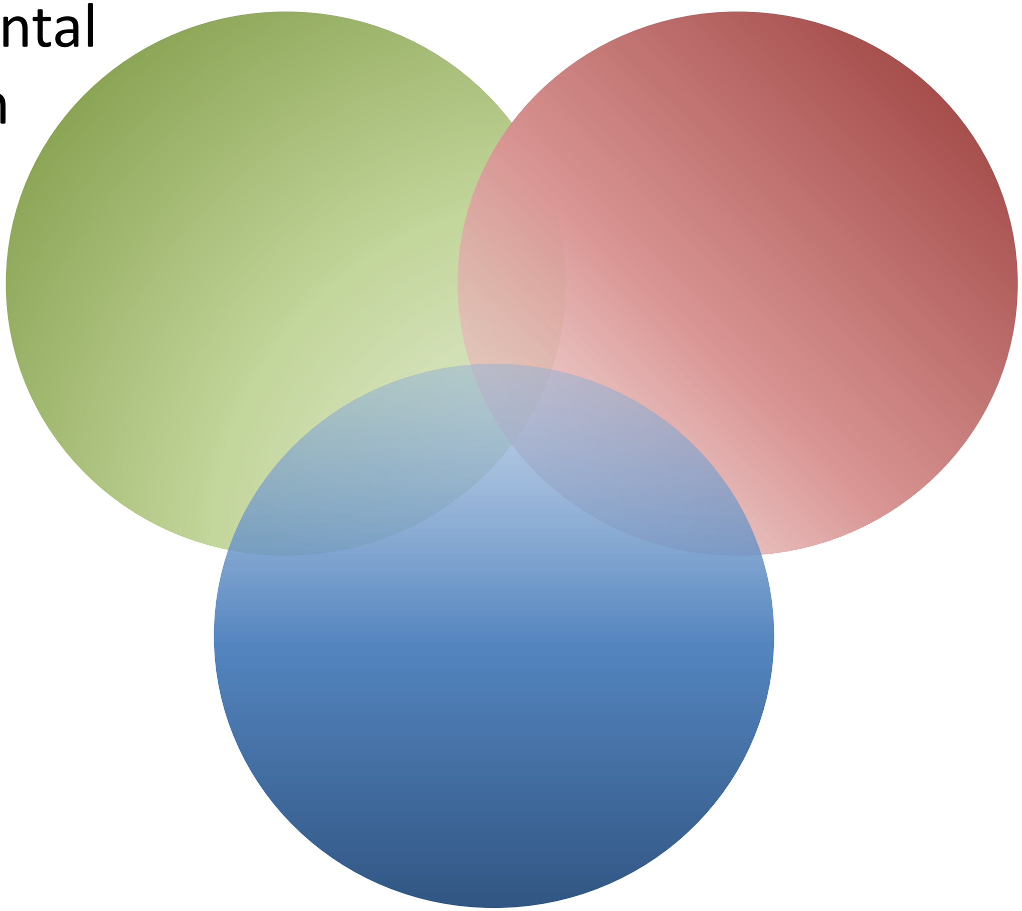
Course structure

- 3 two-hour blocks per week (Mon, Wed, Friday)
  - ~ 40 min lecture/discussion
  - ~ 70 min lab exercises
- some assigned reading – Biostar Handbook, others
- participation in classroom discussion is expected
- no exams

Course Objective

- to teach you how to teach yourself

What contributes to successful data analysis?

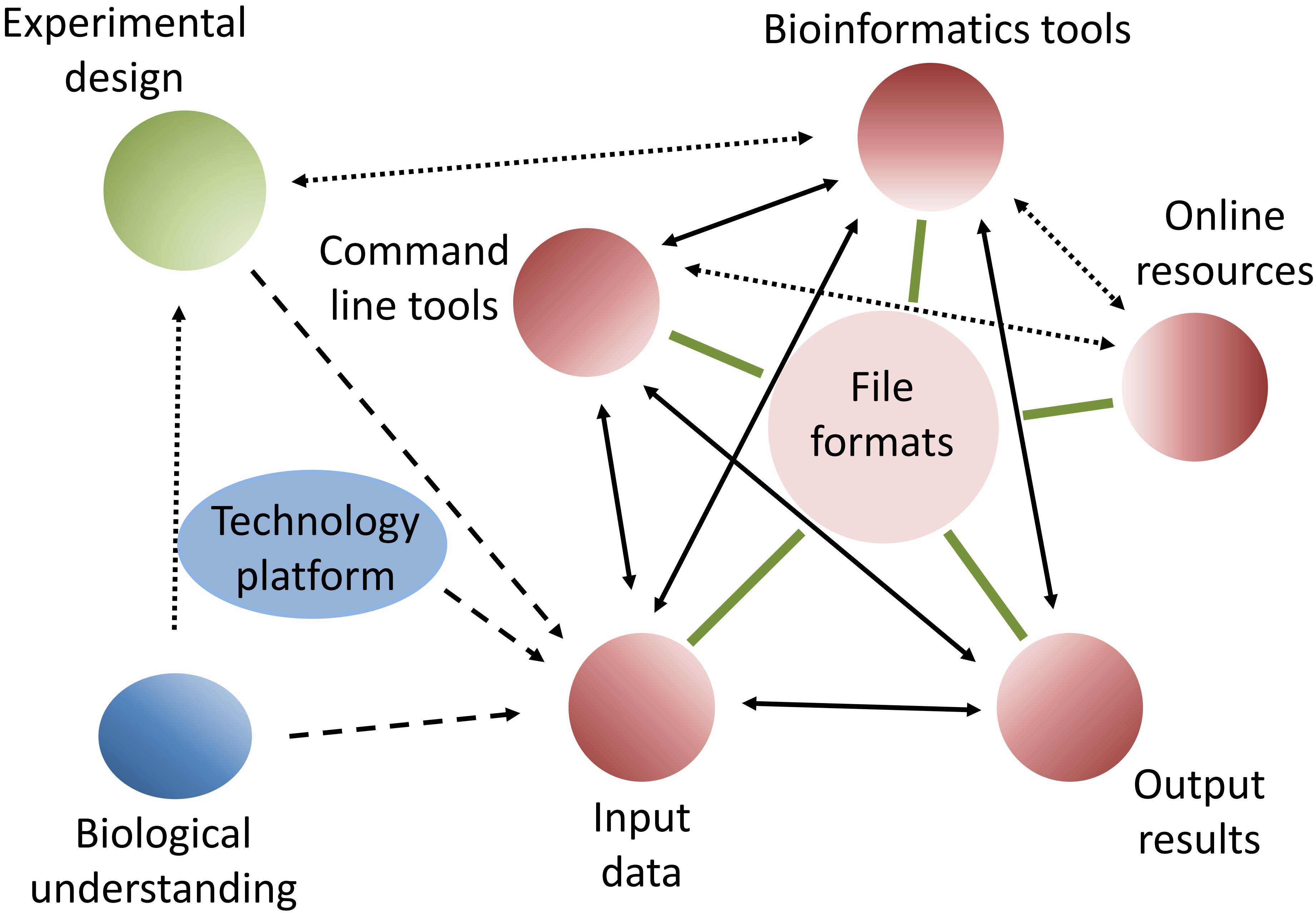


Experimental design

Computing skills

Biological knowledge

A higher-resolution view



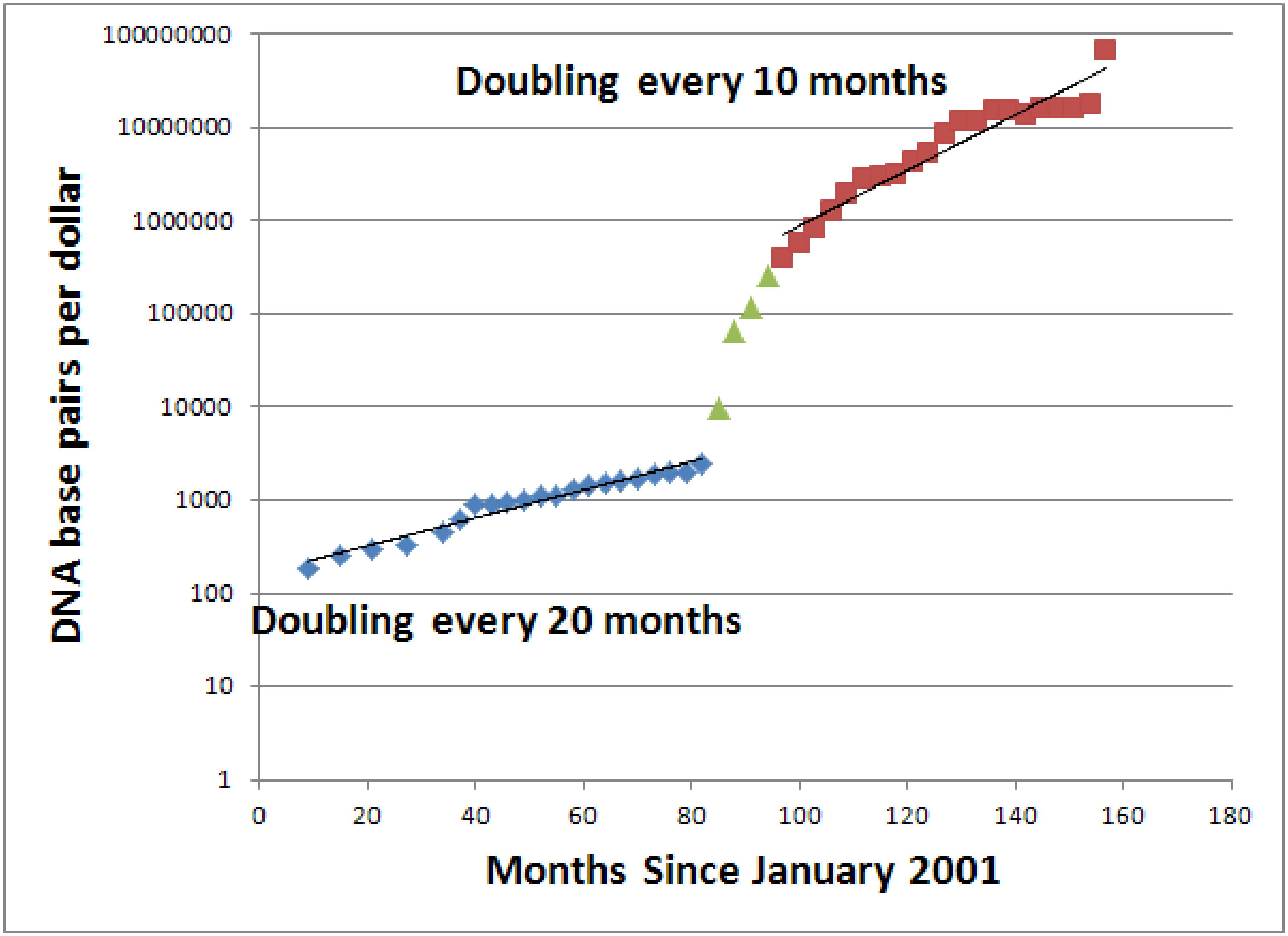
Sequence data analysis is changing rapidly

- relatively few methods are completely static
- much of the software is still under active development
- new methods and tools are reported every month
- staying on the learning curve is essential
- Biostar Handbook is updated monthly – the online version is likely to be the most current

Why is the pace of change so fast?

- by necessity – the rate of data acquisition is increasing faster than the growth of computing power and storage space
- opportunity abounds – technology and theory are advancing rapidly and in parallel

DNA sequence data per dollar



Based on data from https://www.genome.gov/sequencingcosts/

Strategy for sequencing data analysis

- Decompose complex tasks into simple and systematic steps
- Identify key aspects of those simple steps as patterns
- Find a general solution for each pattern or step
- Put those solutions together into a ‘pipeline’ for data analysis

Decomposition -> Abstraction -> Generalization -> Algorithm Development

Example

- Find the sum of all integers from 1 to 200

  - without a calculator or written calculations

  - in thirty seconds or less

Example

- Find the sum of all integers from 1 to 200

  - without a calculator or written calculations

  - in thirty seconds or less

1	2	3	4	5	.....	98	99	100
$+\underline{200}$	$+\underline{199}$	$+\underline{198}$	$+\underline{197}$	$+\underline{196}$	.....	$+\underline{103}$	$+\underline{102}$	$+\underline{101}$
201	201	201	201	201	.....	201	201	201

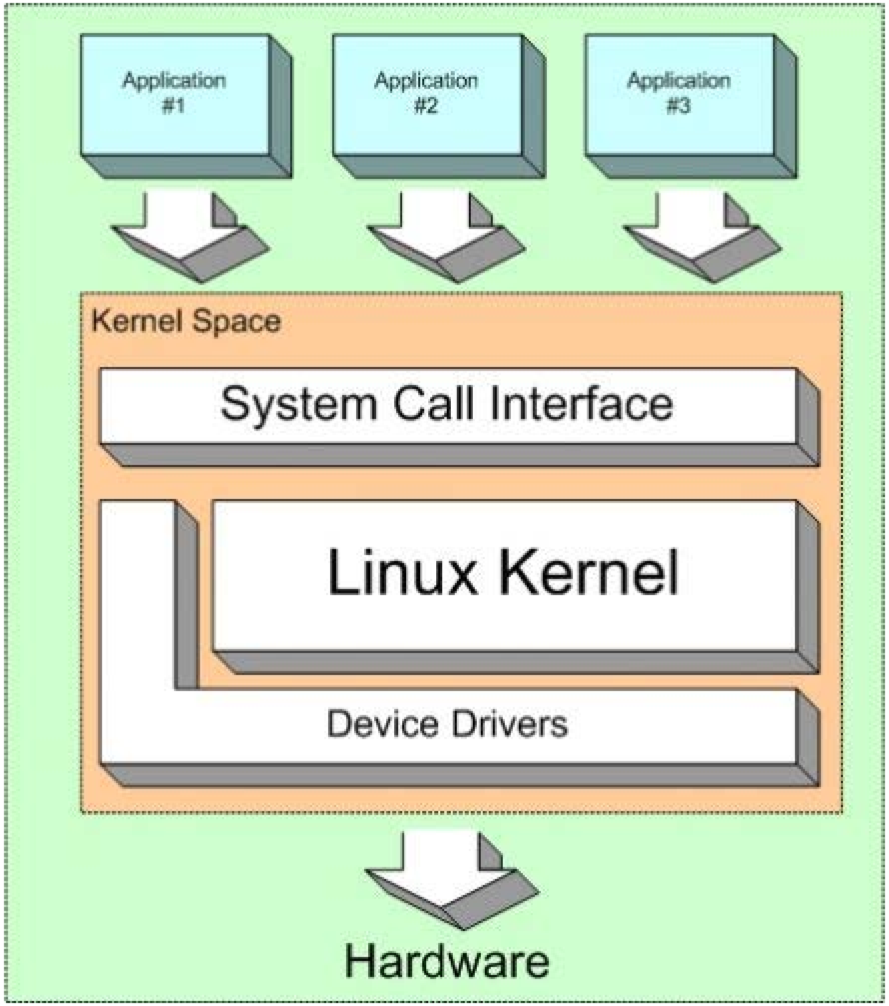
Solution: 100 * 201 = 20,100

Why use Linux for sequencing data analysis?

- It is well-suited to the task
  - Composed of thousands of tools that each do simple tasks
  - Preferred development platform for open-source software
  - Free, as in free speech and as in free beer
  - However … it's built for speed, not for comfort

- Some alternatives exist
  - Java programs can often run on any major operating system (Mac, Windows, or Linux)
  - Mac OS X is essentially Linux with a very nice graphical user interface; the Mac Terminal is very similar to the Linux command-line environment
  - Commercial software packages exist for data analysis in Windows – CLC Genomics Workbench is one such program

Modular design in Linux – a ‘toolbox’ approach



- Individual components of the Linux operating system are written as separate programs
- Different programs can have similar functions
- A Linux “distribution” is a collection of programs that work together as an operating system
- Users have the power to add new programs, or take away existing programs that are not being used, to optimize system performance

Why is modularity an advantage?

- Separate tools, each with a single simple function, can be combined in an infinite variety of ways to solve novel problems
- A monolithic integrated software solution can be very good for what the designer intended, but useless for any problem the designer did not anticipate
- Adding new tools to the toolbox is easy; redesigning a monolithic integrated system is difficult

There is always more than one way to do it

- Some sequence analysis tasks have matured to stability
- Most have not, and are still changing
- ‘Best practices’ are also changing, and subject to dispute

Linux distributions

- collections of ‘tools’ targeted to different user groups
- some are commercial, most are not
- five or six distributions account for most of the users
- many dozens of variants available, mostly of minor interest

Which to use for sequencing data analysis?

We will use different varieties of Ubuntu

- A widely-used distribution with good hardware support
- Base for Cyverse machine images, with pre-installed bioinformatics packages
- Lubuntu is Ubuntu with the Lightweight X Desktop Environment (LXDE) instead of the Gnome desktop

Cloud computing

- Amazon Web Services (AWS) is a commercial resource for computing infrastructure; Cyverse is a free resource for academic users; the Virtual Computing Lab (VCL) is free for NC State users
- Provide access to ‘virtual machines’, or VMs, for users
- A VM is an ‘instance’ of a ‘machine image’
- The underlying image is the same for every instance
- User-generated files are lost when the instance terminates

Using Cyverse for data analysis

- Machine images with bioinformatics programs pre-installed and configured are available
- A laptop or desktop is used as a terminal
- Connections
    - Secure SHell (SSH), using PuTTY
    - Graphical interface through X2Go Client
- Data storage in the Data Store – 100 Gb allotment to start

Command-line utilities

- Lubuntu16.04_Intro_BIT815.pdf is a tutorial
- Organized around the Linux system on the USB drive
- Introduces important concepts and common commands
- Work through this tutorial once per week until May or until all the commands become second nature
- CONTINUED PRACTICE IS ESSENTIAL

Commands introduced in tutorial

- Directory commands: ls, mkdir, cd, pwd, rmdir
- File commands: cp, mv, rm, cat, less, head, tail, wc, file, sort, cut, uniq, chmod, gzip, zcat, zgrep, diff
- Stream commands and redirection: >, >>, <, |, rev, tr
- System commands: man, whatis, apropos, ps, sleep, bg, jobs, fg, kill, df, du, find, free, …

How to read man pages

- The ‘man <command>’ command invokes the ‘less’ page viewer
- The terminal prompt disappears and the screen shows only text from the man page
- To advance to the next page of text: hit the ‘f’ key or space bar
- To go back to the previous page: hit the ‘b’ key
- To exit from the viewer and return to a terminal prompt, hit ‘q’
- The information on the screen will disappear and you will be back at the terminal prompt; your previous command history will be visible in the lines above the current prompt.
- To keep the man page open while you type at an active prompt, open another terminal window to view the man page

Format conventions on man pages

- The command name is followed by one or more [options] …
- Mandatory arguments if any are UPPERCASE
- Subsequent lines explain what each option does

  sometimes in great detail, sometimes in cryptic jargon

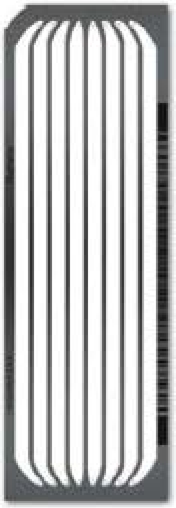
Sequencing technology overview

- Two different systems at the GSL: Illumina, PacBio
- Illumina has higher throughput in terms of numbers of reads, but length is in range of 50 to 300 nt
- PacBio is a long-read single-molecule platform that produces fewer reads, but read lengths of tens of thousands of bases are possible

Similarities

- DNA molecules are fragmented and ligated to adaptors
- individual DNA molecules are immobilized on a surface
- a series of nucleotide addition reactions are carried out
- the nucleotide added is detected after each addition
- a data file is produced containing the DNA sequences of many fragments

Sequencing technology overview – Illumina



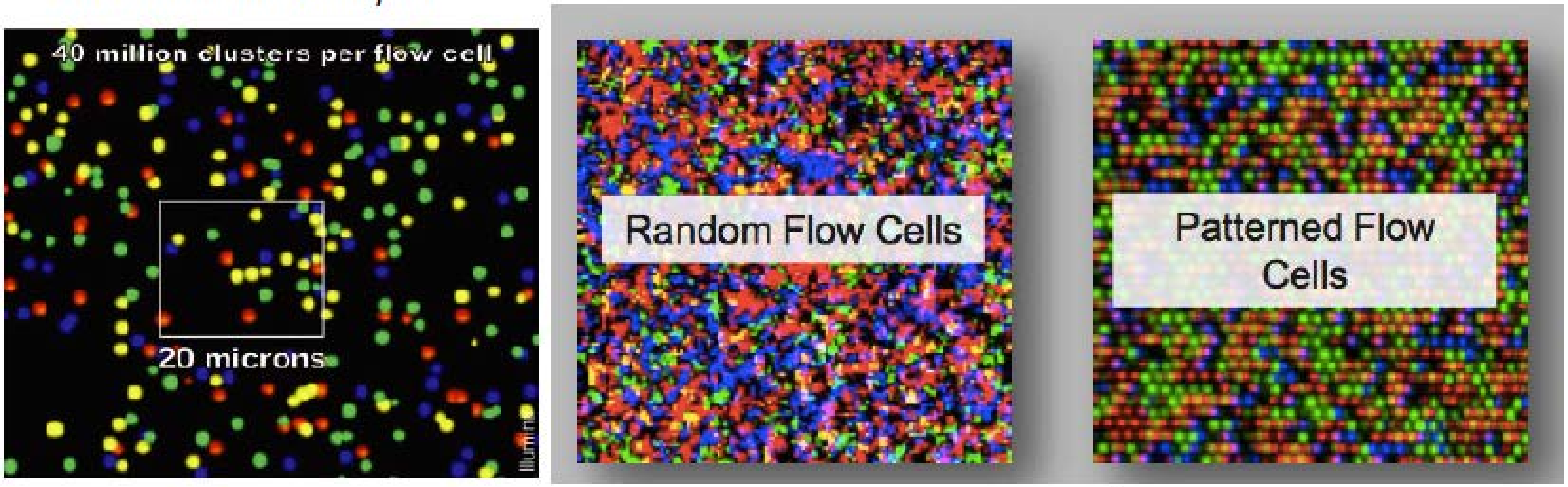
Illumina uses a glass ‘flowcell’, about the size of a microscope slide, with 8 separate ‘lanes’.

Increases in throughput have come in part from increasing the density of “clusters” (DNA molecules being sequenced) on the flowcell. The original Genome Analyzer scanned only one surface of the lane; the HiSeq instruments scan both upper and lower surfaces of each flowcell lane. The patterned flow cells used in Hiseq3000 and 4000 instruments provide higher density and better separation of clusters, to improve both data yield and quality.

2006 Genome Analyzer

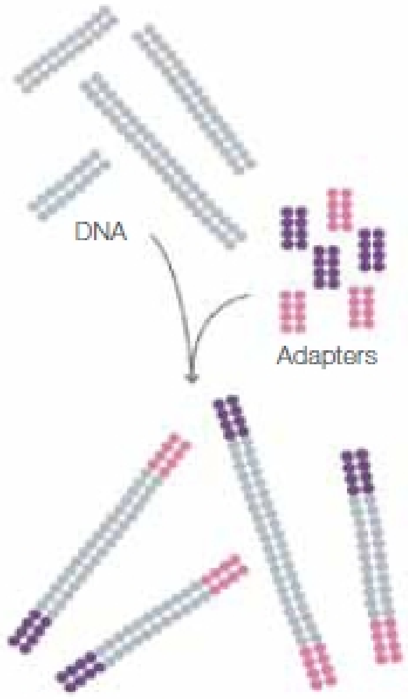
2010 – Hiseq2000

2015 – Hiseq 3000/4000



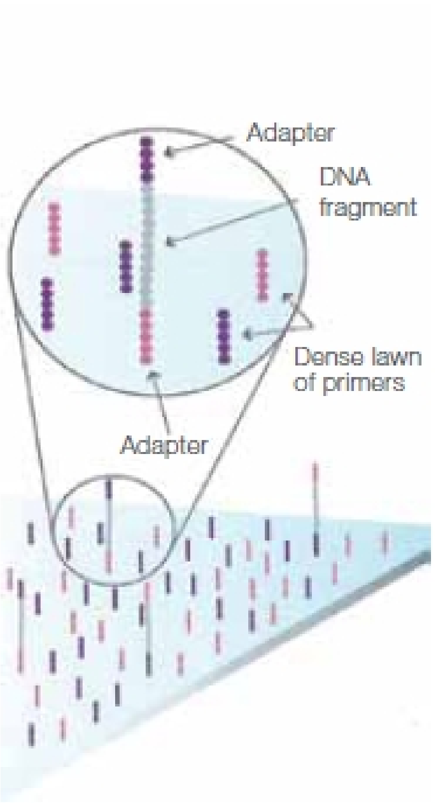
Sequencing technology overview – Illumina

Figure 2: Prepare Genomic DNA Sample



Randomly fragment genomic DNA and ligate adapters to both ends of the fragments.

Figure 3: Attach DNA to Surface

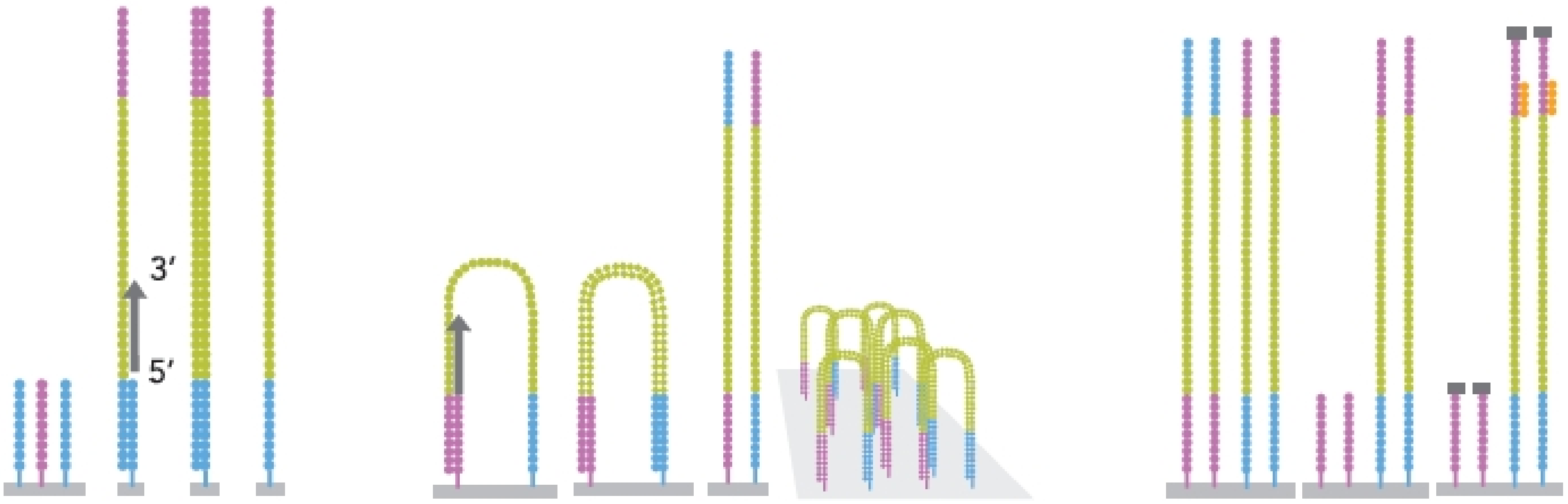


Bind single-stranded fragments randomly to the inside surface of the flow cell channels.

Fragment DNA, ligate adaptor oligos

Single-stranded DNA binds to flowcell surface

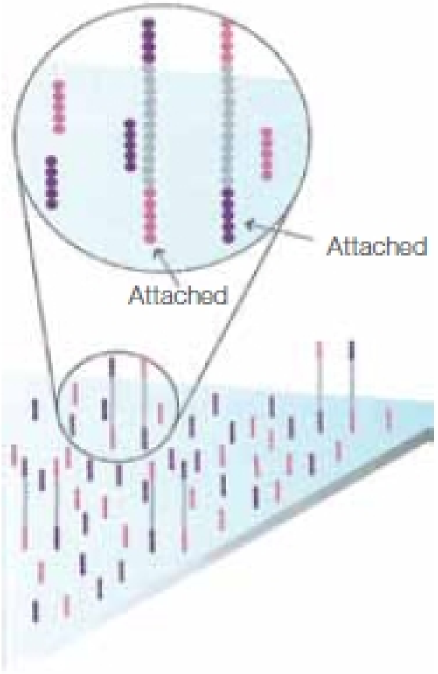
Sequencing technology overview – Illumina



Surface-bound primers are extended by DNA polymerase across annealed ssDNA molecules, the DNA is denatured back to single strands, and the free ends of immobilized strands anneal again to oligos bound on surface of flowcell. This ‘bridge PCR’ continues until a cluster of ~ 1000 molecules is produced on the surface of the flowcell, all descended from the single molecule that bound at that site. After PCR, the free ends of all DNA strands are blocked.

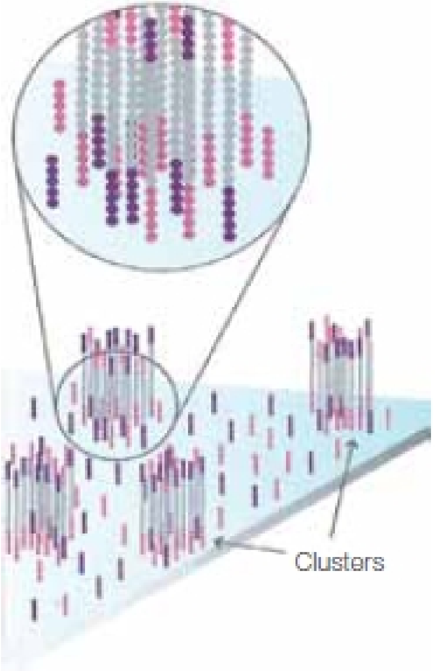
Sequencing technology overview – Illumina

Figure 6: Denature the Double-Standed Molecules



Denaturation leaves single-stranded templates anchored to the substrate.

Figure 7: Complete Amplification

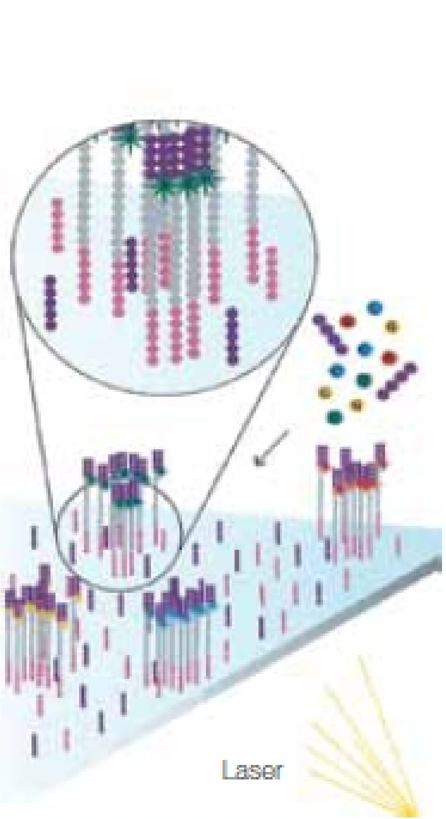


Several million dense clusters of double-stranded DNA are generated in each channel of the flow cell.

Another perspective of the amplification process, showing the clusters of products

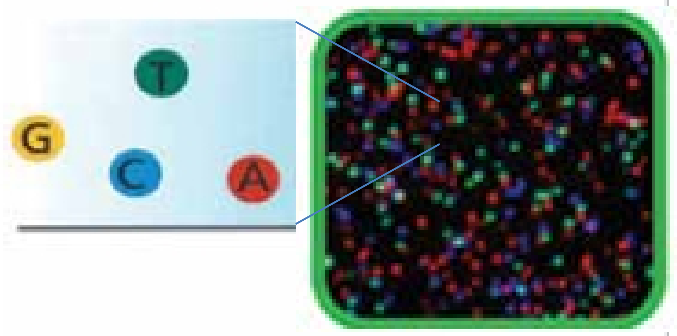
Sequencing technology overview – Illumina

Figure 8: Determine First Base



The first sequencing cycle begins by adding four labeled reversible terminators, primers, and DNA polymerase.

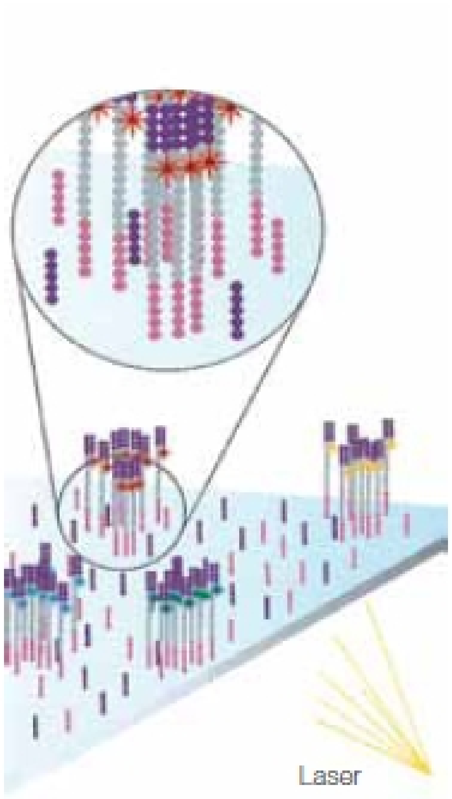
Figure 9: Image First Base



After laser excitation, the emitted fluorescence from each cluster is captured and the first base is identified.

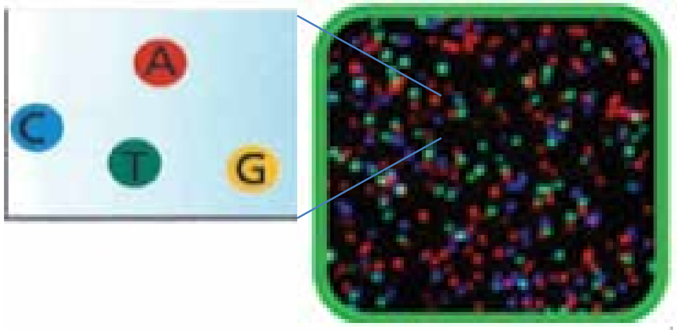
Sequencing technology overview – Illumina

Figure 10: Determine Second Base



The next cycle repeats the incorporation of four labeled reversible terminators, primers, and DNA polymerase.

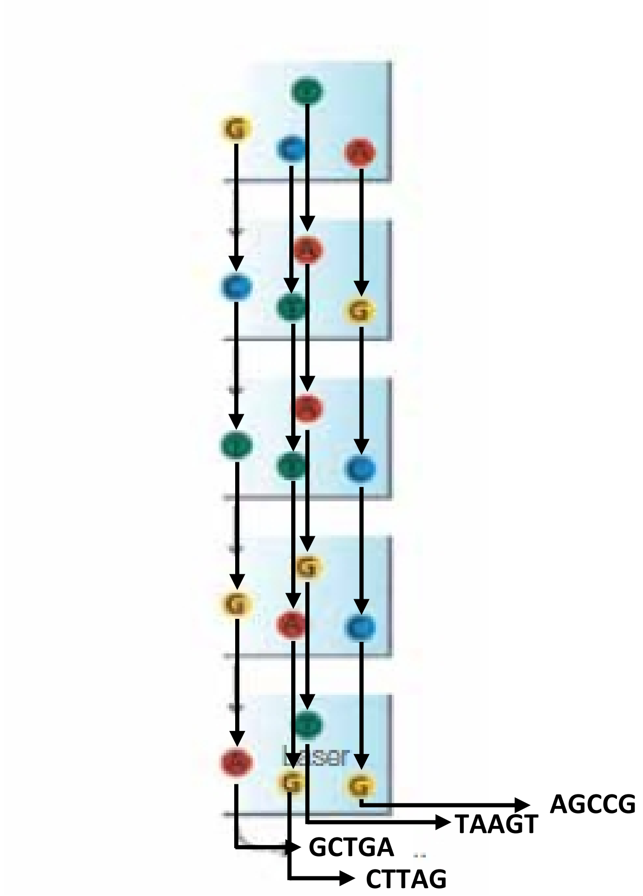
Figure 11: Image Second Chemistry Cycle



After laser excitation, the image is captured as before, and the identity of the second base is recorded.

Sequencing technology overview – Illumina

Figure 12: Sequencing Over Multiple Chemistry Cycles

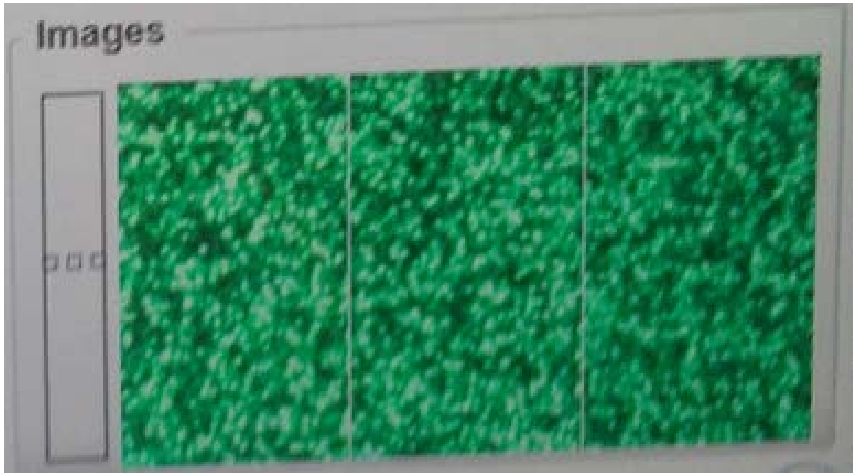


The sequencing cycles are repeated to determine the sequence of bases in a fragment, one base at a time.

Although four different colors are used for the fluorescent nucleotides, only two lasers are used to excite the fluorescence. The fluorescent labels are grouped in pairs - labels on A and G are excited by one laser, and labels on C and T are excited by the other laser.

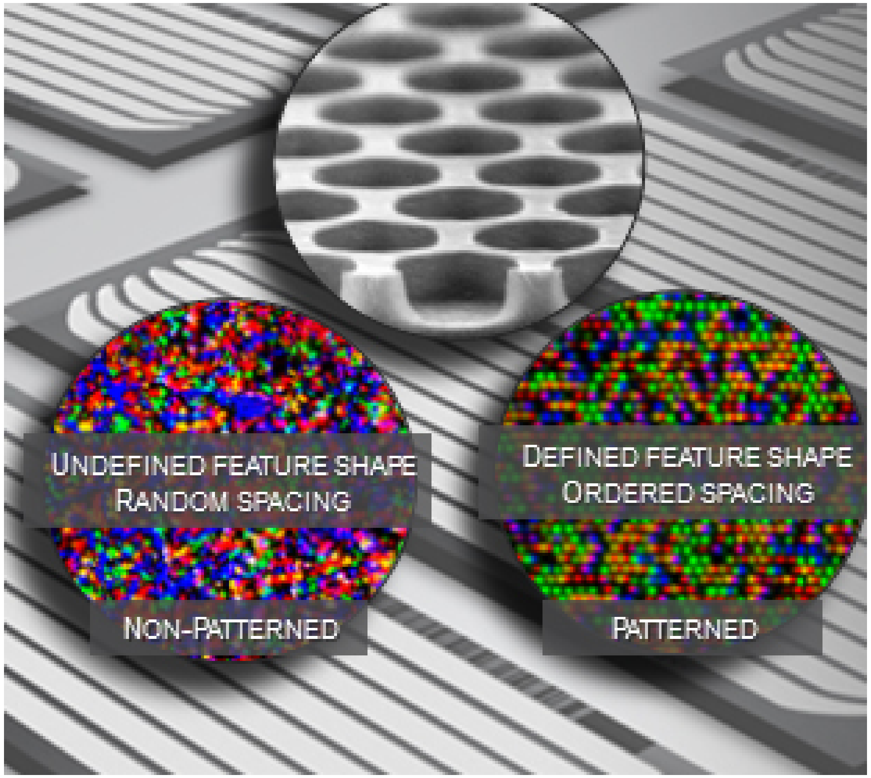
This means that distinguishing between the A signal and the G signal is more difficult for the instrument than A versus C or A versus T. Base substitution errors are the most common type of sequencing error for Illumina instruments.

Sequencing technology overview – Illumina

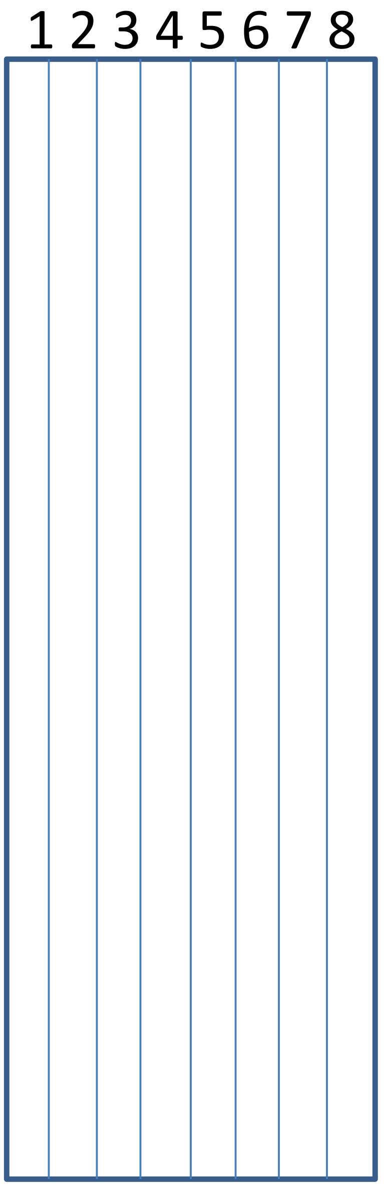


An image from the Hiseq2500 during a run

New “patterned flow cell technology” on Hiseq3000, 4000, X5 and X10



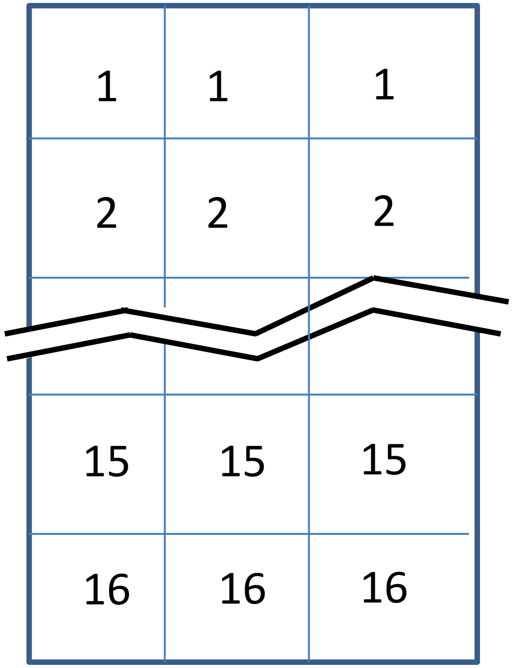
Illumina flowcell geometry (HiSeq)



A flowcell has 8 lanes, which are physically separated. Each surface (upper and lower) of each lane is imaged during each cycle of sequencing in 3 separate "swaths", and 16 images or 'tiles', are collected from each swath, for a total of 96 tiles per lane. The swaths and tiles are not physically separated.

Tiles within a lane are numbered from 1 to 16 down (from outflow end to inflow end), and swaths are numbered from left to right.

The top surface is 1, and the bottom surface is 2. Each tile ID is expressed as a 4-digit number, organized as Surface-Swath-Tile
[12]   [123]   [01..16]



http://seqanswers.com/forums/showthread.php?t=19259

Sequence data output format

- DNA sequence data are typically provided with “quality scores”, either as paired files or combined in a FASTQ file
- In separate files, DNA sequences are in FASTA format and quality scores are numbers from 0 to 40

```
>FQSOZHZ01ASD8U rank=0159502 x=206.0 y=1164.5 length=65
TACCTCTCCGCGTAGGCGCTCGTTGGTCCAGCAGAGGCGGCCGCCTTCGTCGCGAGCAGAATAGG
```

and

```
>FQSOZHZ01ASD8U rank=0159502 x=206.0 y=1164.5 length=65
37 28 28 28 37 37 37 28 28 28 37 39 36 33 33 33 37 37 40 40 39 39
39 39 39 39 40 40 39 39 39 39 39 39 40 38 37 37 35 35 35 33 33 23
23 23 19 17 19 21 21 17 17 17 19 17 19 14 14 12 12 14 16 12 12
```

- In a FASTQ file, DNA sequences look similar, but quality scores are encoded as single text characters rather than as numbers

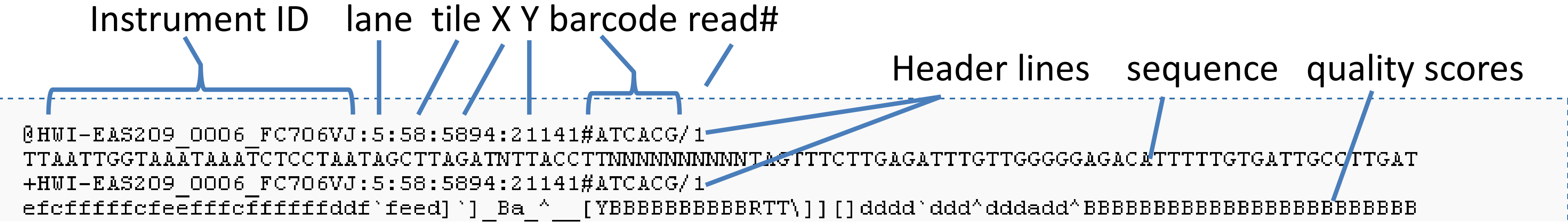
```
@FQSOZHZ01ASD8U rank=0159502 x=206.0 y=1164.5 length=65
TACCTCTCCGCGTAGGCGCTCGTTGGTCCAGCAGAGGCGGCCGCCTTCGTCGCGAGCAGAATAGG
+
F===FFF===FHEBBBFFIIHHHHHHIIHHHHHHIGFFDDDBB88842466222424//--/1--
```

http://en.wikipedia.org/wiki/FASTA_format and http://en.wikipedia.org/wiki/FASTQ_format

Understanding FASTQ format

or “what do all these symbols mean?”

See http://en.wikipedia.org/wiki/FASTQ_format for more details

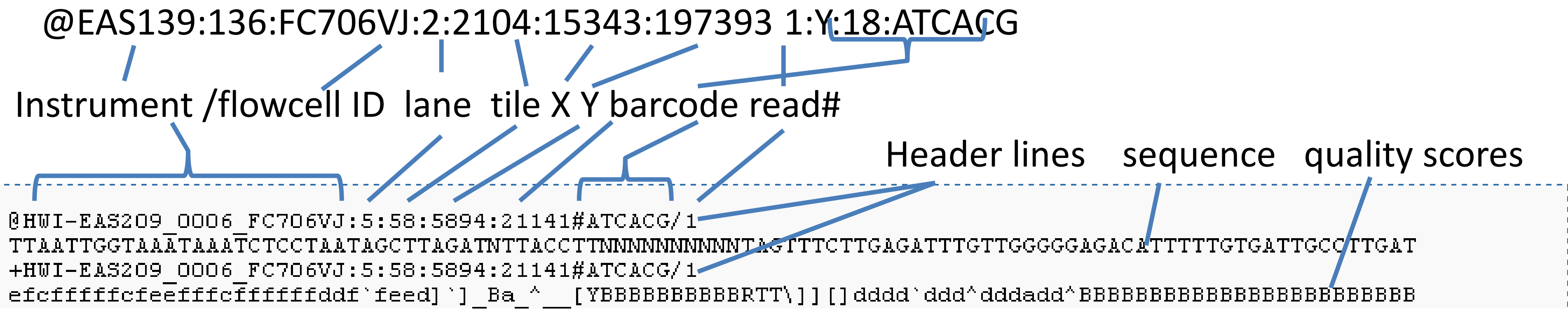


```
@HWI-EAS209_0006_FC706VJ:5:58:5894:21141#ATCACG/1
TTAATTGGTAAATAAATCTCCTAATAGCTTAGATNTTACCTTNNNNNNNNNNTAGTTTCTTGAGATTTGTTGGGGGAGACATTTTTGTGATTGCCTTGAT
+HWI-EAS209_0006_FC706VJ:5:58:5894:21141#ATCACG/1
efcfffffcfeefffcffffffddf`feed]`]_Ba_^__[YBBBBBBBBBBRTT\]][]dddd`ddd^dddadd^BBBBBBBBBBBBBBBBBBBBBBBB
```

- Quality scores are numbers that represent the probability that the given base call is an error.
- These probabilities are always less than 1, so the value is given as -10 x log(10) of the probability
- For example, an error probability of 0.001 ($1x10^{-3}$) is represented as a quality score of 30.
- The numbers are converted into text characters so they occupy less space – a single character is as meaningful as 2 numbers plus a space between adjacent values

Understanding FASTQ format

Illumina v1.8 header version:



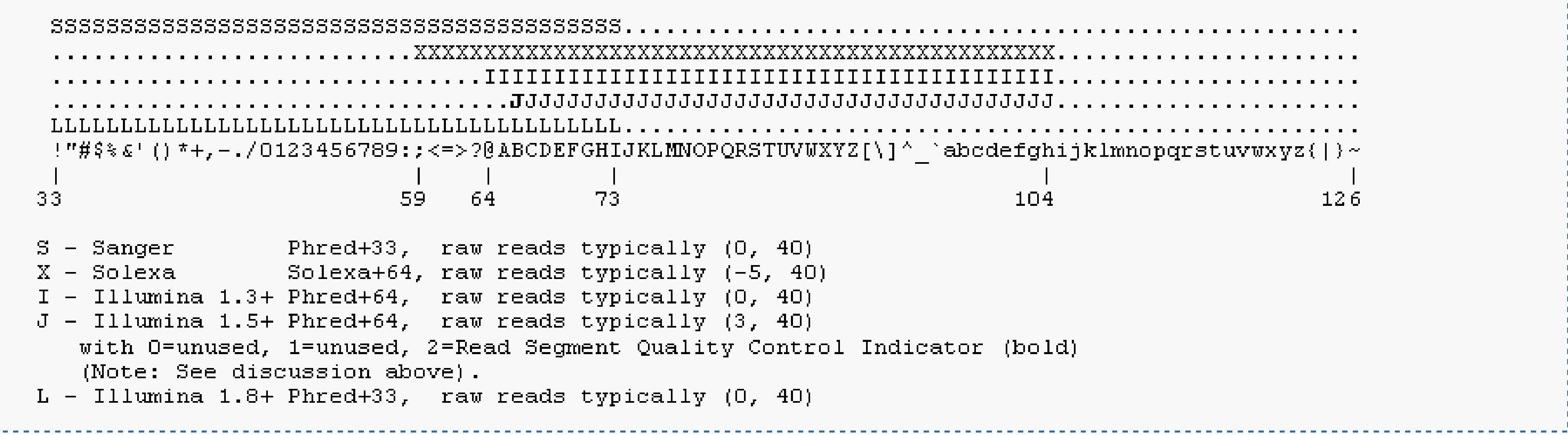
@EAS139:136:FC706VJ:2:2104:15343:197393 1:Y:18:ATCACG

Instrument /flowcell ID lane tile X Y barcode read#

Header lines   sequence   quality scores

```
@HWI-EAS209_0006_FC706VJ:5:58:5894:21141#ATCACG/1
TTAATTGGTAAATAAATCTCCTAATAGCTTAGATNTTACCTTNNNNNNNNNNNTAGTTTCTTGAGATTTGTTGGGGGAGACATTTTTGTGATTGCCTTGAT
+HWI-EAS209_0006_FC706VJ:5:58:5894:21141#ATCACG/1
efcfffffcfeefffcffffffddf`feed]`]_Ba_^__[YBBBBBBBBBBRTT\]][]dddd`ddd^dddadd^BBBBBBBBBBBBBBBBBBBBBBBB
```

Unfortunately, at least four different ways of converting numbers to characters have been used, and header line formats have also changed, so one aspect of data analysis is knowing what you have.



```
  SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....................................................
  ..........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX......................
  ...............................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII......................
  .................................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ......................
  LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL.....................................................
  !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
  |                         |    |        |                              |                     |
 33                        59   64       73                            104                   126

 S - Sanger        Phred+33,  raw reads typically (0, 40)
 X - Solexa        Solexa+64, raw reads typically (-5, 40)
 I - Illumina 1.3+ Phred+64,  raw reads typically (0, 40)
 J - Illumina 1.5+ Phred+64,  raw reads typically (3, 40)
    with 0=unused, 1=unused, 2=Read Segment Quality Control Indicator (bold)
    (Note: See discussion above).
 L - Illumina 1.8+ Phred+33,  raw reads typically (0, 40)
```

Computer Vocabulary

- RAM – random-access memory
  - Holds data for immediate access
  - Information is lost when machine is shut down
- Disk Storage Space – hard drive or equivalent
  - Holds data stored on physical surface
  - Stable to power shutdown
- Central Processing Unit (CPU), processors, cores
  - What actually does the computation
  - Most computers now have more than one
  - Each processor has a “thread” of computational tasks, “hyper-threading” means > 1 thread per core

Computational Resources

- Typical desktop computers often lack enough RAM to analyze sequence datasets
  - 32-bit operating systems cannot address more than 4 Gb
  - 64-bit operating systems can address up to 2000 Gb
  - 64-bit Linux is the platform of choice for open-source software packages; this is what we will use
- Alternatives to desktop computers
  - Virtual Computing Lab (http://vcl.ncsu.edu)
  - HPC (http://hpc.ncsu.edu)
  - Cyverse.org cloud computing (NSF-funded)

Computational Resources

- Resources on the NC State campus
  - High Performance Computing center (http://hpc.ncsu.edu)
    - Primarily used by engineers and computer scientists
    - Staff have not been very familiar with bioinformatics programs
    - Users are expected to know what they want to do and how to accomplish it with available software; support staff help with getting software installed and working
  - Bioinformatics Consulting and Service Core (http://brc.ncsu.edu/consulting/)
    - A fee-for-service facility
    - Provides analytical services, some access to computer hardware

The Unix Shell

“A **Unix shell** is a command-line interpreter or shell that provides a traditional user interface for the Unix operating system and for Unix-like systems. Users direct the operation of the computer by entering commands as text for a command line interpreter to execute or by creating text scripts of one or more such commands.” - *Wikipedia*

Things to Keep in Mind

- *There is no ‘undelete’*
- Shell commands & filenames are case-sensitive (unlike Windows)
- Many characters have special meanings to the shell, so it is safest to use only letters, numbers, _, and . in filenames. Special characters are #;& "\/',`:<>|*?$(){}[] and space
- File or directory names containing these characters must be quoted so the shell does not assign the character its special meaning

Things to remember

- The computer does what you tell it to do, not what you want it to do
- Prototype, then optimize

  Start small, with something that works, then expand its capabilities until you reach your goal
- Worry about speed and efficiency only if you must

  If trial runs are slow, optimize; otherwise let the computer do more work and you do less