Read mapping

Michael Schatz

Sept 21, 2022

Lecture 7: Applied Comparative Genomics

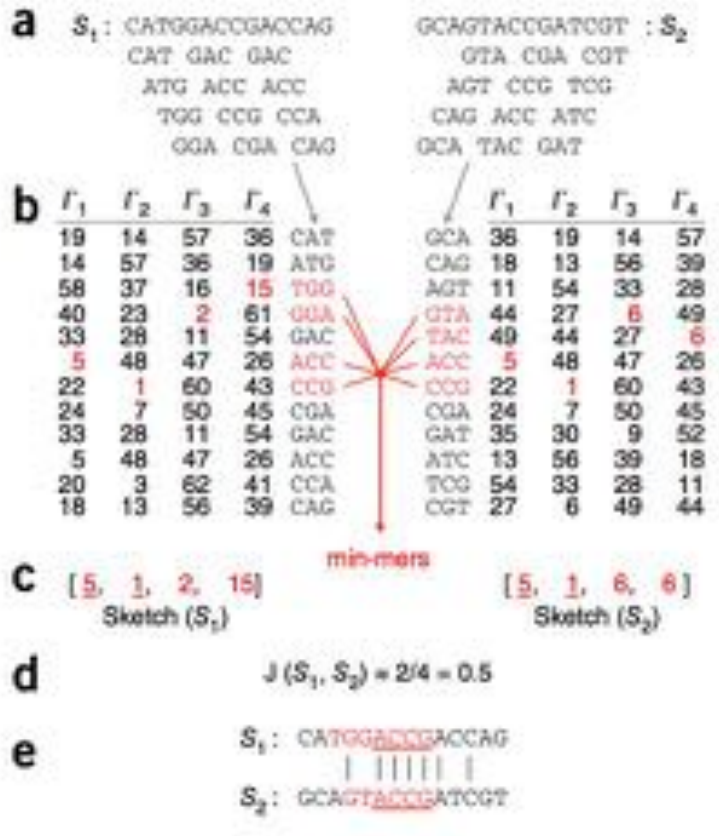
Assignment 2: Genome Assembly
Due Monday Sept 19 by 11:59pm

https://github.com/schatzlab/appliedgenomics2022/tree/main/assignments/assignment2
Check Piazza for questions!

Very fast approximate overlapping

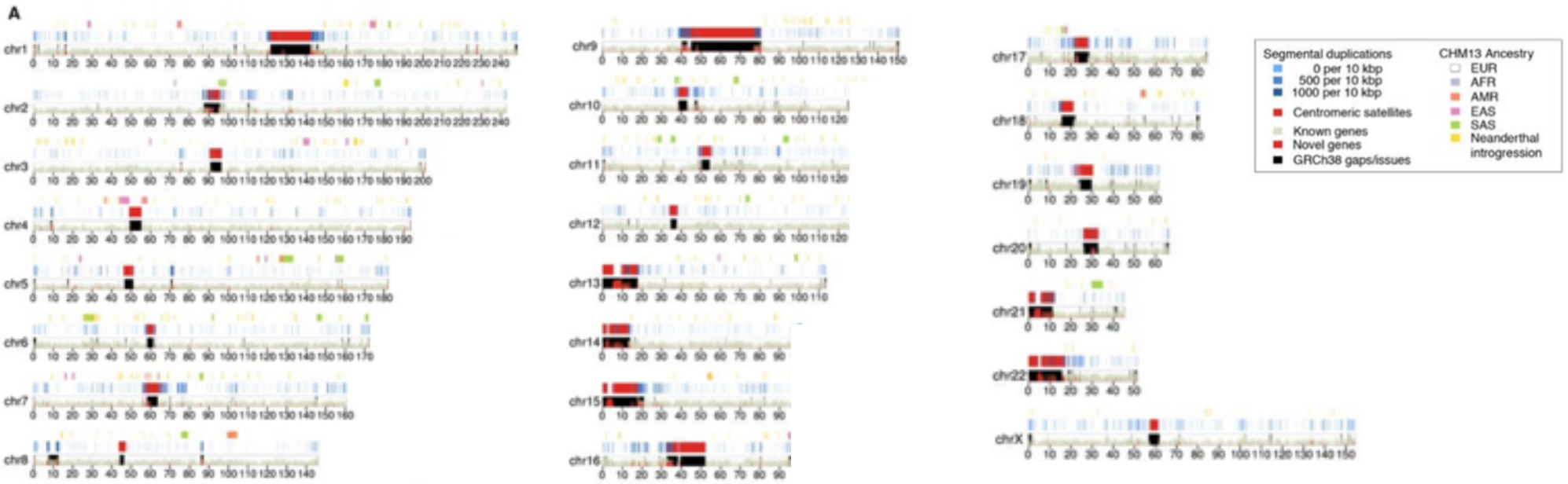
Maybe we don't need to compute the exact identity of the overlap region, just approximate it

- If two reads overlap, they should share many of the same kmers: Their Jaccard coefficient should be high: |intersection| / |union|
- But tracking all of the kmers for a read is a lot of overhead
- Instead, compare the "sketch" of the reads: a small fraction of kmers carefully chosen
- LSH: Find the sketch by applying N hash functions to the kmers, and keeping the minimum hash values reported from each (N=4 in example)
- This forms a nice "random" sample of the reads, and the Jaccard coefficient is a good approximation of the sequence similarity



Assembling large genomes with single-molecule sequencing and locality-sensitive hashing
Berlin et al (2015) *Nature Biotechnology*

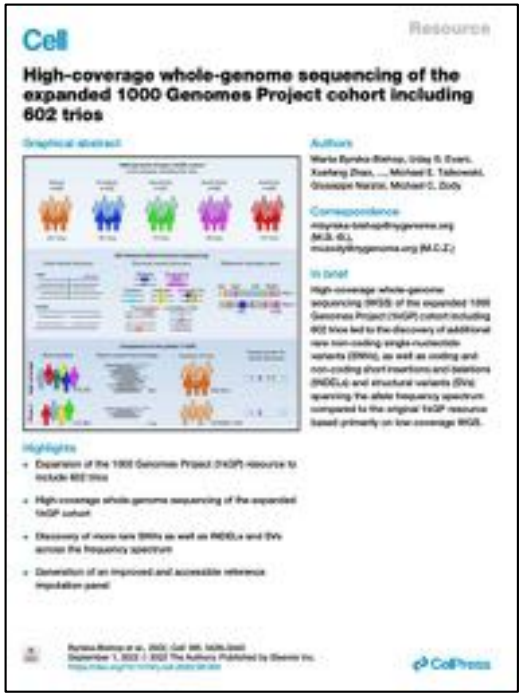
The *complete* sequence of a human genome



CHM13v1.1 genome size is **3.057 Gbp with zero Ns**
Every chromosome is telomere-to-telomere, quality estimated >Q70
~190 Mbp (~8%) of new sequence vs. GRCh38, fixes thousands of errors

(Nurk et al. Science, 2022)

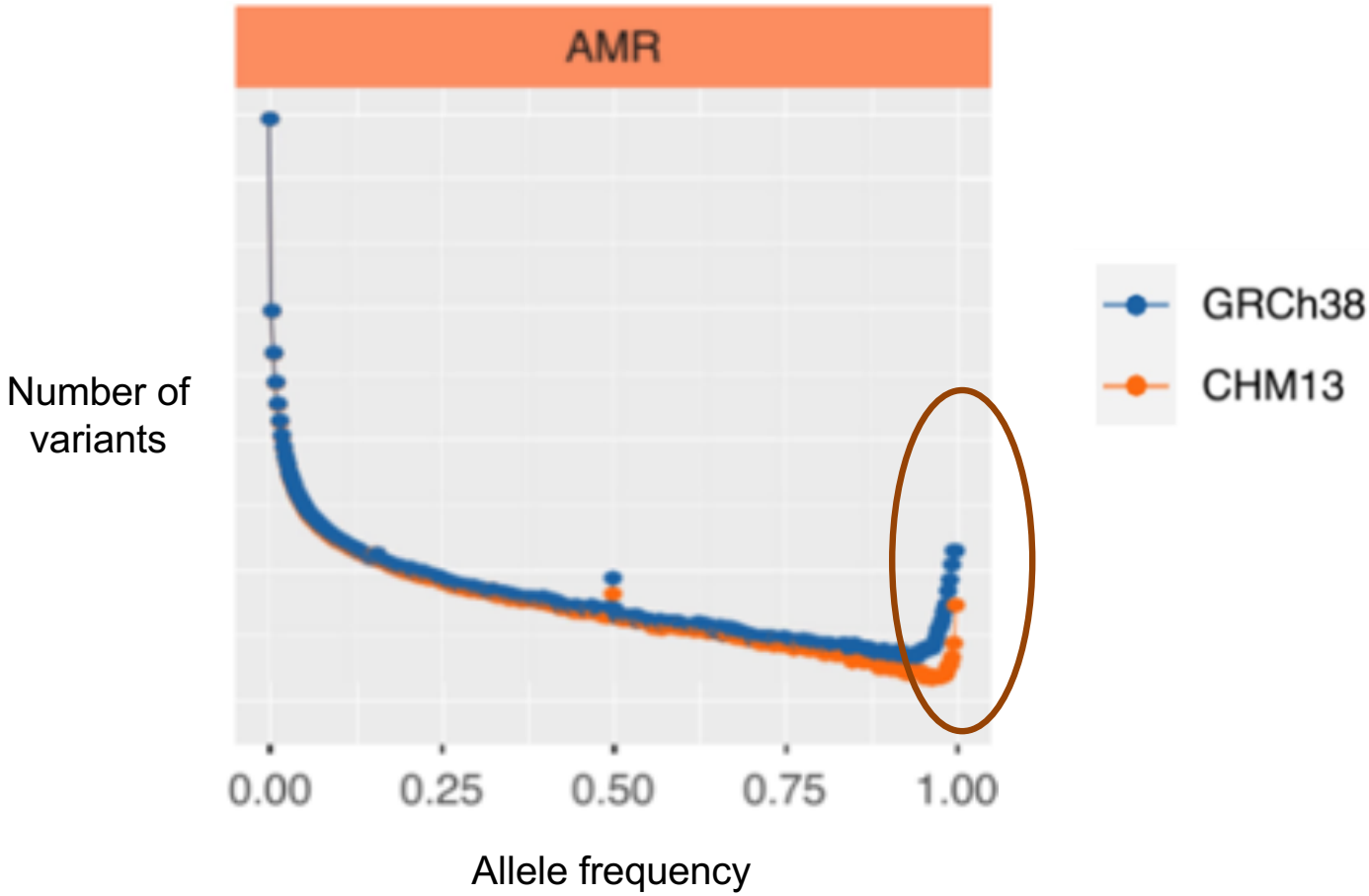
T2T Variants: 1000 Genomes Project



26 populations from 5 superpopulations (continental regions)
3202 samples (2504 core genomes + 698 offspring)

3202 samples x 30Gb = 96Tb input data | >5Pb of intermediate data | >>1M core hours

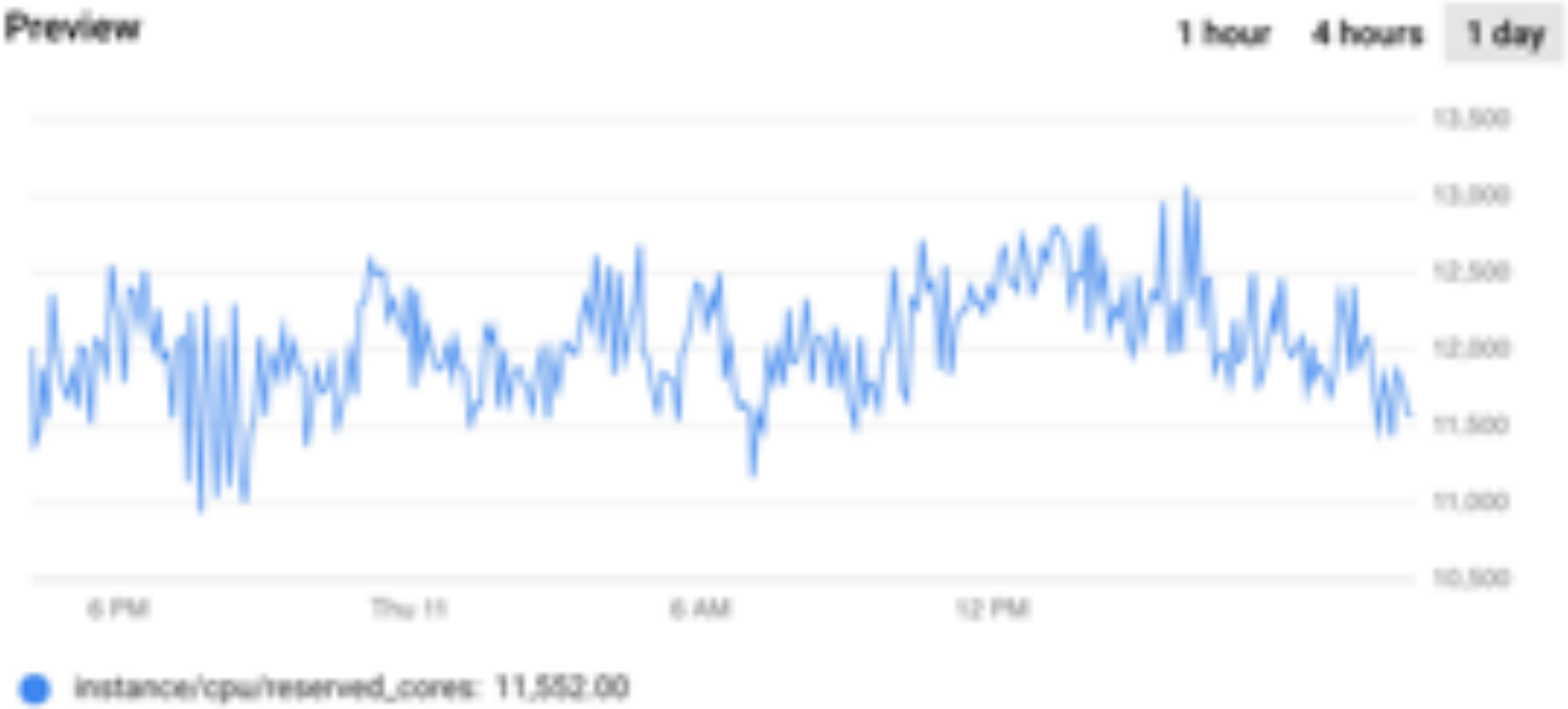
Explaining Decreased Per-Sample Count




Earth's heart of iron begins to yield its secrets p. 35
Microglia in chronic pain recovery and relapse pp. 13 & 84
Particle acceleration in a nova explosion p. 77
Science
$15
1 APRIL 2022
SPECIAL ISSUE
AAAS
FILLING THE GAPS
Closing in on a complete human genome p. 47


Science
COMPLETING THE HUMAN GENOME
Segmental duplications and their variation in a complete human genome
Complete genomic and epigenetic maps of human centromeres
From telomere to telomere: The transcriptional and epigenetic state of human repeat elements
A complete reference genome improves analysis of human genetic variation
Epigenetic patterns in a complete human genome
The complete sequence of a human genome

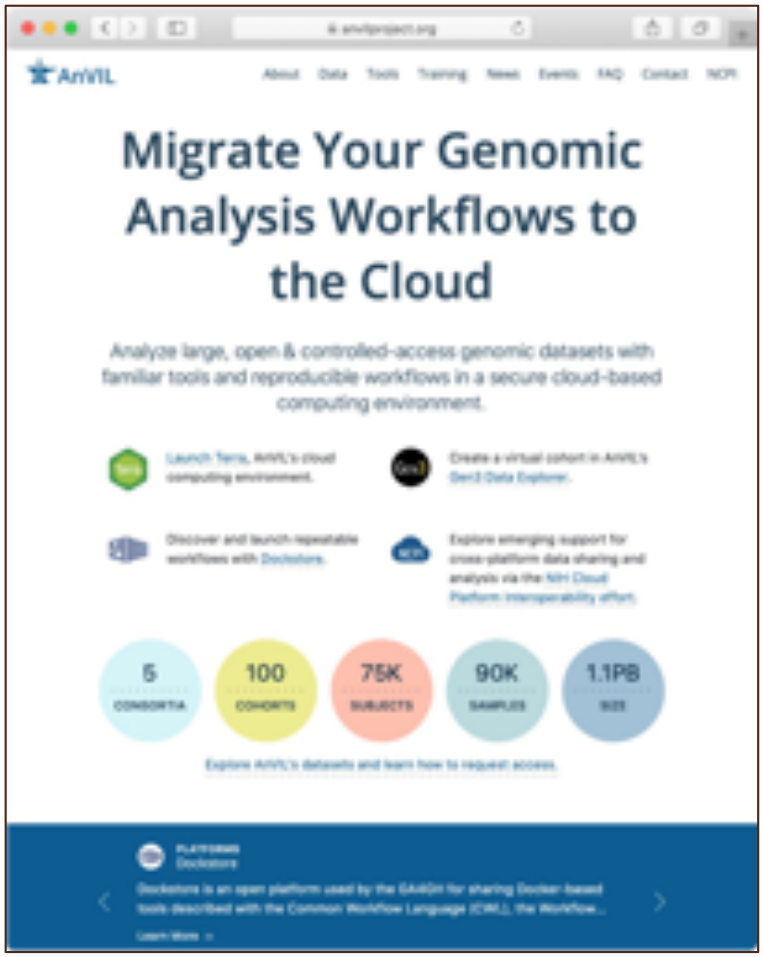
Core usage over 24 hours



What is the AnVIL?

Scalable and interoperable computing resource for the genomics scientific community

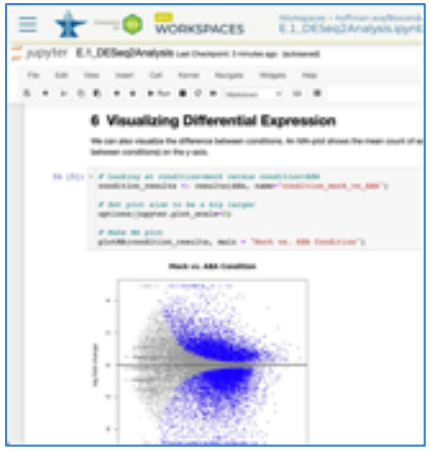
- Cloud-based infrastructure
  - Highly elastic; shared analysis and computing environment
- Data access and security
  - Genomic datasets, phenotypes and metadata
  - Large datasets generated by NHGRI programs, as well as other initiatives / agencies
  - dbGaP Authenticated sharing of primary and derived datasets
- Collaborative computing environment for datasets and analysis workflows
  - Storage, scalable analytics, data visualization
  - Security, training & outreach, with new models of data access
  - ...for both users with limited computational expertise and sophisticated data scientist users



https://anvilproject.org

Schatz, Philippakis et al. (2022) *Cell Genomics*
doi: https://doi.org/10.1016/j.xgen.2021.100085

AnVIL Analysis Platforms



jupyter

+ Code, text and plots in one document
+ Supports coding in Python or R
- Least scalable, not a complete IDE

Galaxy

+ Graphical interface for thousands of tools and workflows
+ Highly accessible and reproducible
- Tools must be preconfigured to use

RStudio®

+ Feature rice IDE for programming in R
+ Rich statistics, ML, and visualizations
- Limited support for other programming languages

{wdl}

+ Extremely scalable and flexible
- Most technically demanding
- Unpredictable and potentially large costs

My perspective

jupyter

+ Code, text and plots in one document
+ Supports coding in Python or R
- Least scalable, not a complete IDE

Galaxy

+ Graphical interface for thousands of tools and workflows
+ Highly accessible and reproducible
- Tools must be preconfigured to use

RStudio®

+ Feature rice IDE for programming in R
+ Rich statistics, ML, and visualizations
- Limited support for other programming languages

{wdl}

+ Extremely scalable and flexible
- Most technically demanding
- Unpredictable and potentially large costs

Parallel Algorithm Spectrum

Embarrassingly Parallel

Each item is Independent

Loosely Coupled

Independent-Sync-Independent

Tightly Coupled

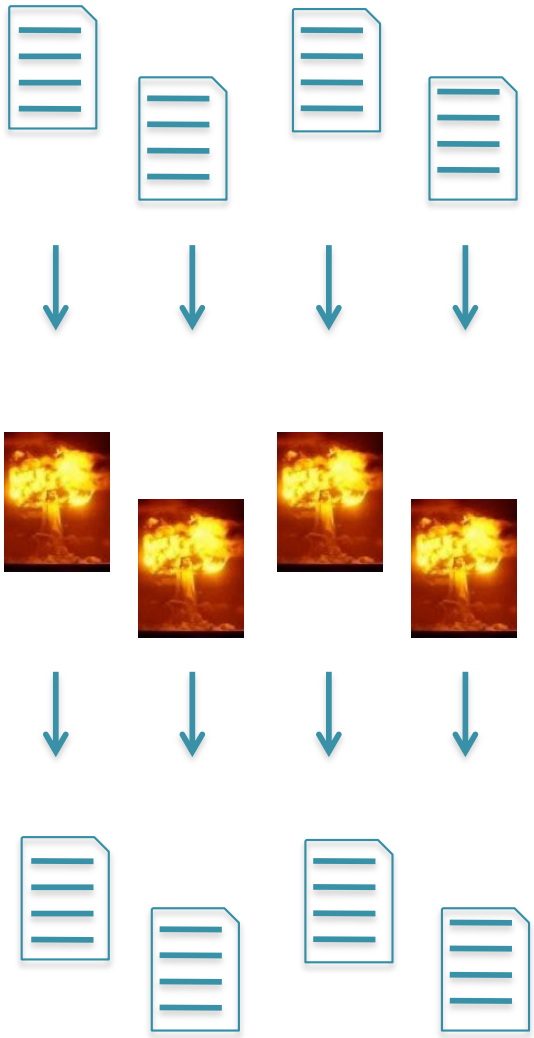
Constant Sync

Embarrassingly Parallel

aka Elementary School Dance

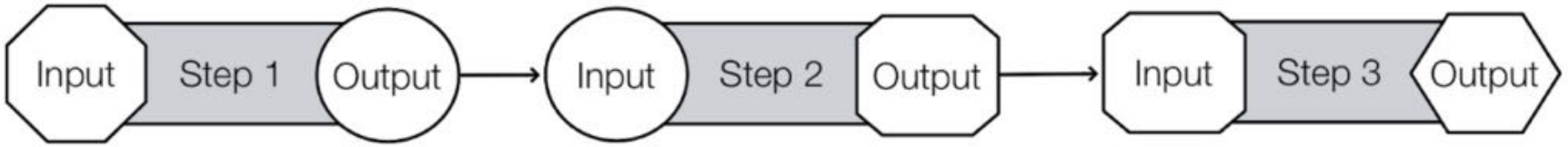
Embarrassingly Parallel

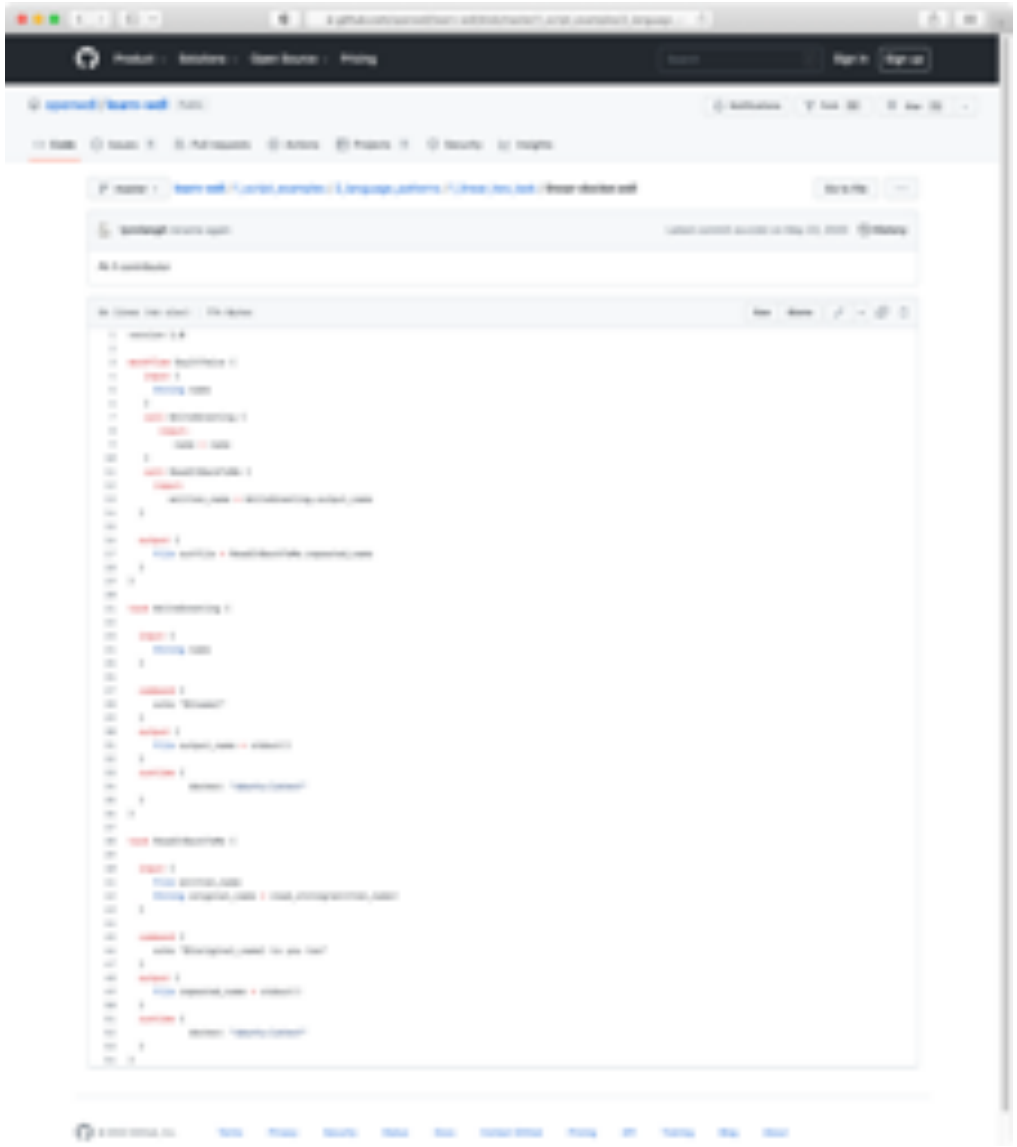
- Batch computing
  - Each item is independent
  - Split input into many chunks
  - Process each chunk separately on a different computer
- Challenges
  - Distributing work, load balancing, monitoring & restart
- Technologies
  - Condor, Sun Grid Engine, WDL
  - Amazon Simple Queue



3202 Genomes to go…

Samantha Zarate





SayItTwice.input

↓

WriteGreeting

↓

ReadGreeting

↓

SayItTwice.outfile

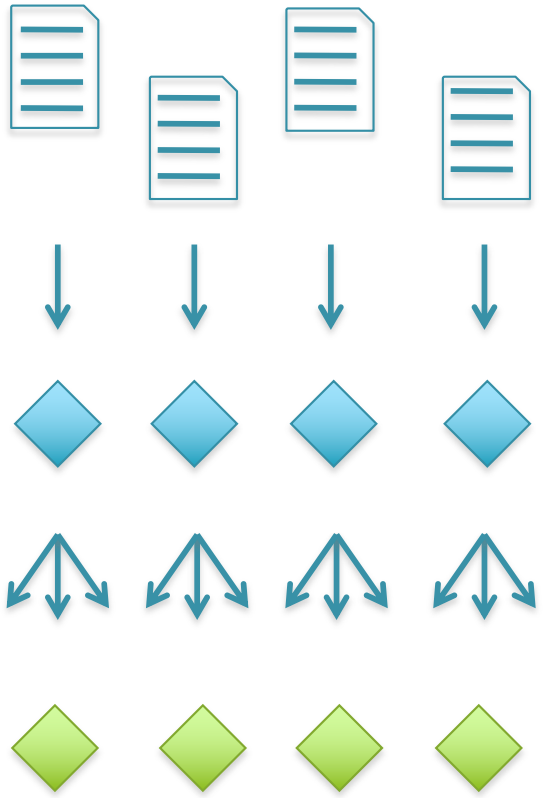
https://github.com/openwdl/learn-wdl/blob/master/1_script_examples/2_language_patterns/1_linear_two_task/linear-docker.wdl

Loosley Coupled

aka the High School Dance

Loosely Coupled

- Divide and conquer
  - Independently process many items
  - Group partial results
  - Scan partial results into final answer
- Challenges
  - Batch computing challenges
  - + Shuffling of huge datasets
- Technologies
  - Hadoop, MapReduce, Dryad, WDL
  - Parallel Databases

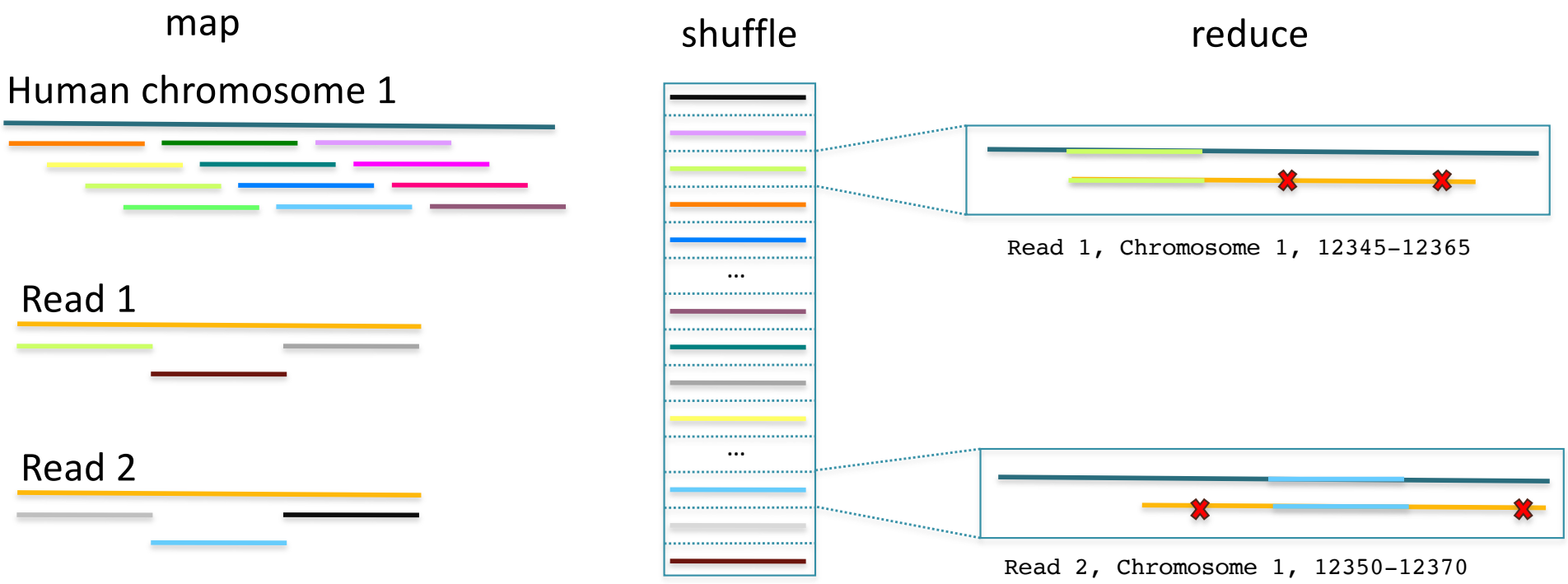


CloudBurst

1. Map: Catalog K-mers
   - Emit k-mers in the genome and reads

2. Shuffle: Collect Seeds
   - Conceptually build a hash table of k-mers and their occurrences

3. Reduce: End-to-end alignment
   - If read aligns end-to-end with ≤ k errors, record the alignment

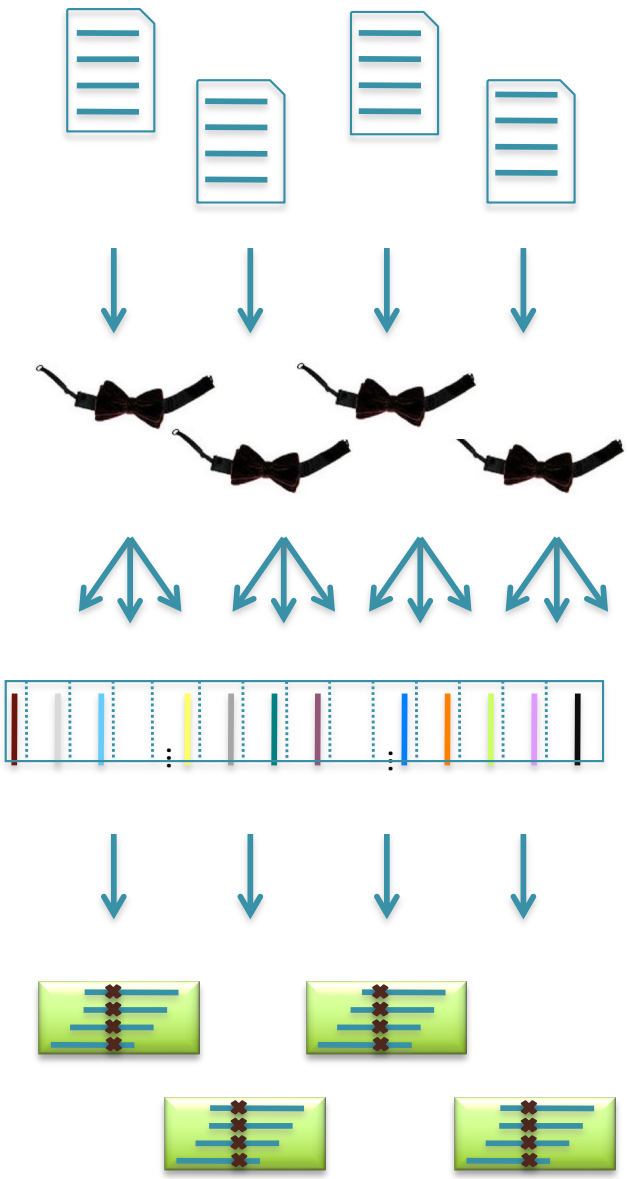


Schatz, MC (2009) CloudBurst: Highly Sensitive Read Mapping with MapReduce. *Bioinformatics*. 25:1363-1369

Crossbow

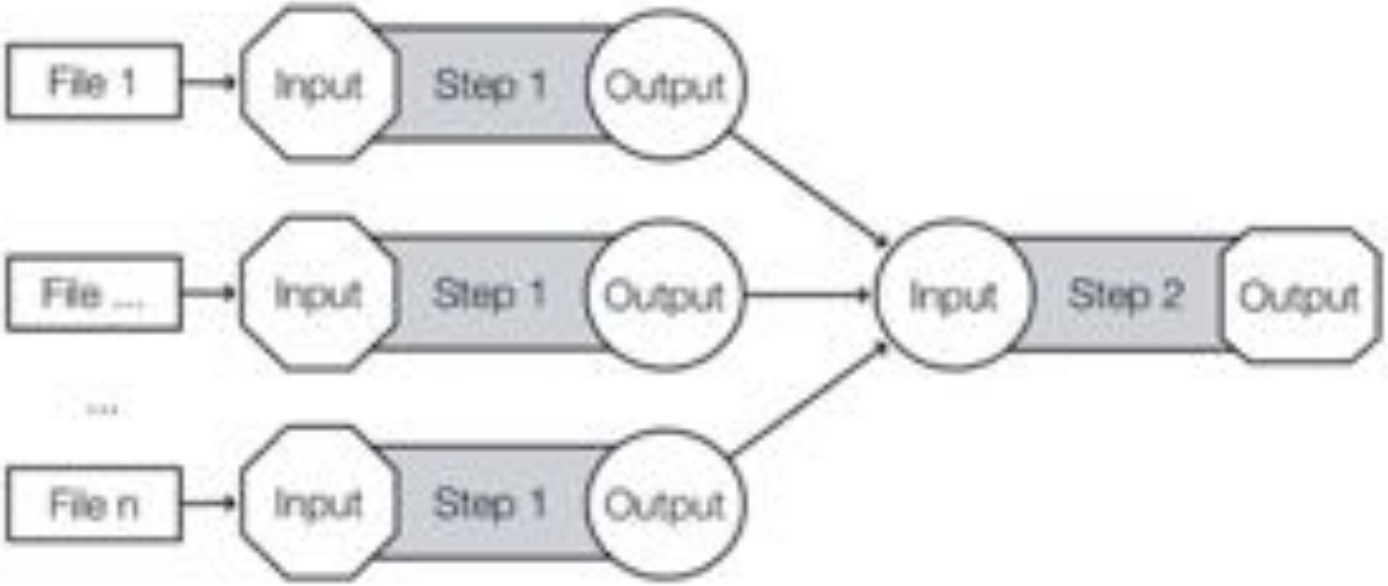
Align billions of reads and find SNPs

- Map: Bowtie (Langmead *et al.*, 2009)
  - Find best alignment for each read
  - Emit (chromosome region, alignment)
- Shuffle: Hadoop
  - Group and sort alignments by region
- Reduce: SOAPsnp (Li *et al.*, 2009)
  - Scan alignments for divergent columns
  - Accounts for sequencing error, known SNPs



Searching for SNPs with Cloud Computing.
Langmead B, Schatz MC, Lin J, Pop M, Salzberg SL (2009) *Genome Biology*. **10**:R134

Scatter-Gather




```
nextflow.enable.dsl=2

process sayHello {
    input:
        val x
    output:
        stdout

    """
    sleep 2
    echo "Hello $x!"
    """
}

workflow {
    Channel.of('Bonjour', 'Ciao', 'Hello', 'Hola') | sayHello | view
}
```

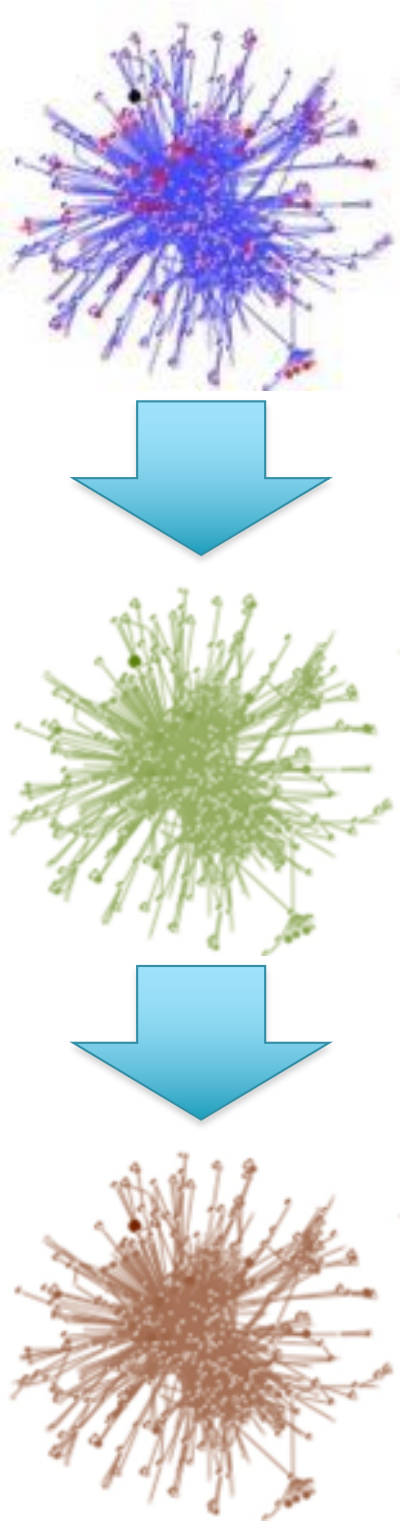
```
version 1.0

workflow HelloWorld {
  scatter(i in range(5)) {
    call WriteGreeting
  }
}

task WriteGreeting {
  command {
    echo "Hello Scattered World"
  }
  output {
    File output_greeting = stdout()
  }
}
```

Tightly Coupled

aka the Ballroom Dance

Tightly Coupled

- Computation that cannot be partitioned
  - Graph Analysis
  - Molecular Dynamics
  - Population simulations
- Challenges
  - Loosely coupled challenges
  - + Parallel algorithms design
- Technologies
  - MPI
  - MapReduce, Dryad, Pregel

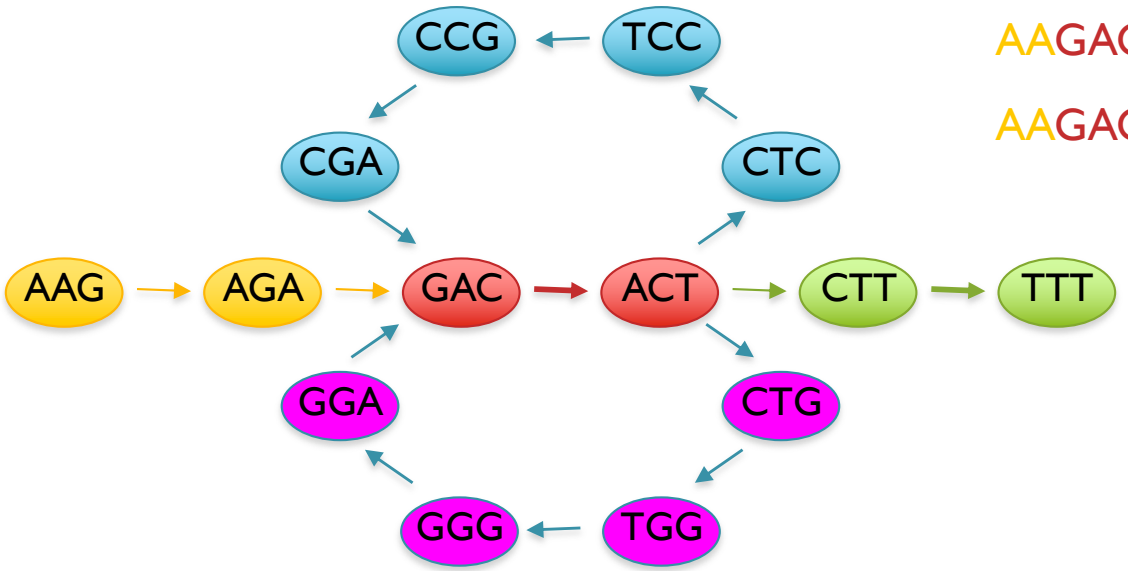


Parallel Genome Assembly

Reads

AAGA
ACTT
ACTC
ACTG
AGAG
CCGA
CGAC
CTCC
CTGG
CTTT
…

de Bruijn Graph



Potential Genomes

AAGACTCCGACTGGGACTTT

AAGACTGGGACTCCGACTTT

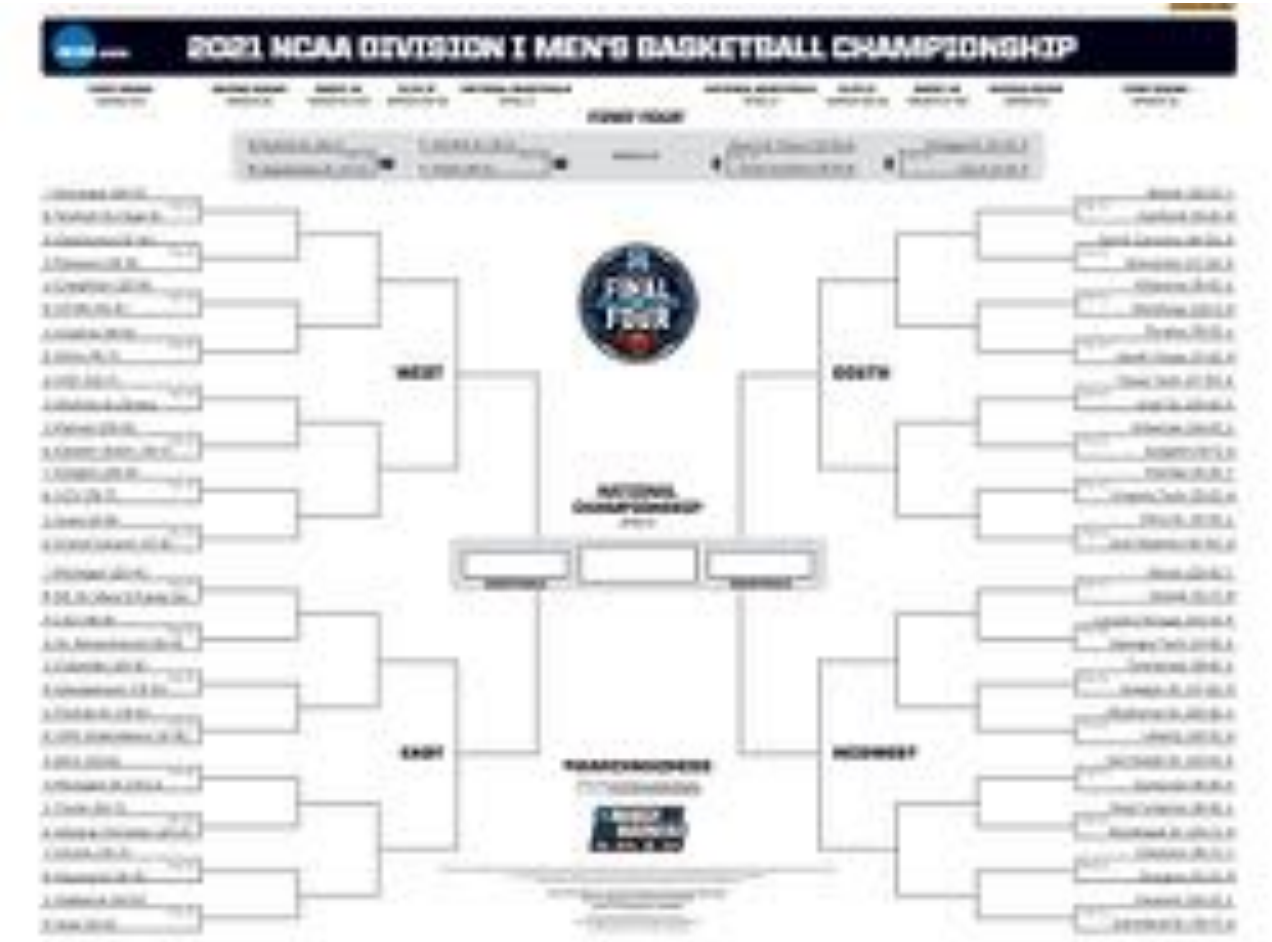
The de Bruijn graph for a human genome requires billions of nodes and edges, which may not fit on a single computer

- Assume there is only one node per computer, but each computer can send messages to its neighbors in the graph
- How can each node decide what to do to collapse the graph?

Warmup Exercise

Who here was born closest to Sept 21?

– You can only compare to 1 other person at a time

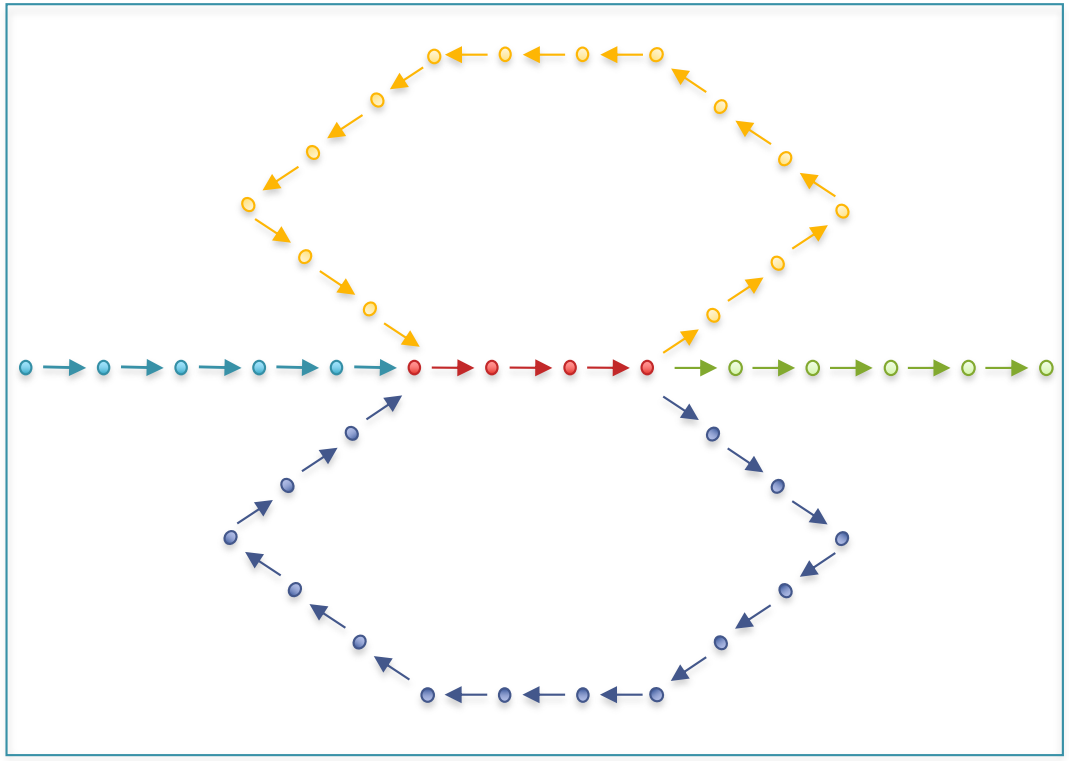


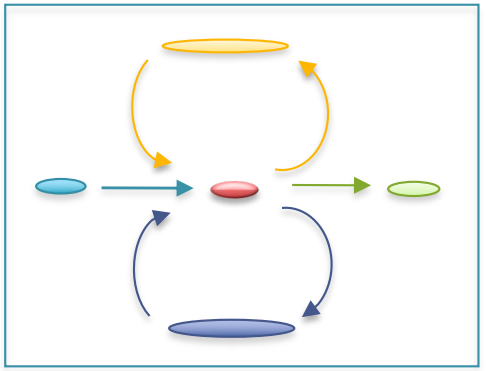
Find winner among 64 teams in just 6 rounds

Why 6? $\log_2(64) = 6$ (aka $2^6 = 64$)

Parallel Graph Compression

- After construction, many edges are unambiguous
  - Merge together compressible nodes
  - Graph physically distributed over hundreds of computers





Design Patterns for Efficient Graph Algorithms in MapReduce.
Lin, J., Schatz, M.C. (2010) Workshop on Mining and Learning with Graphs Workshop (KDD-2010)

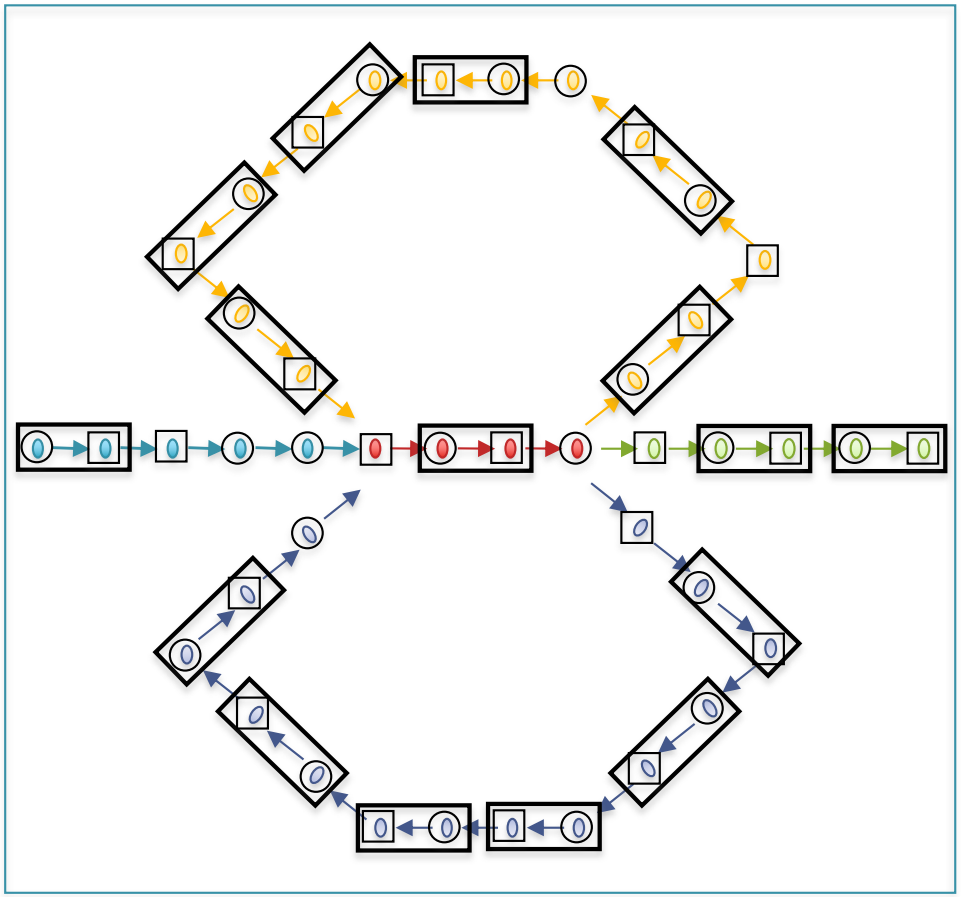
Fast Path Compression

Challenges

- Nodes stored on different computers
- Nodes can only access direct neighbors

Randomized List Ranking

- Randomly assign (H)/ [T] to each compressible node
- Compress (H)→[T] links



Initial Graph: 42 nodes

Randomized Speed-ups in Parallel Computation.
Vishkin U. (1984) *ACM Symposium on Theory of Computation. 230-239.*

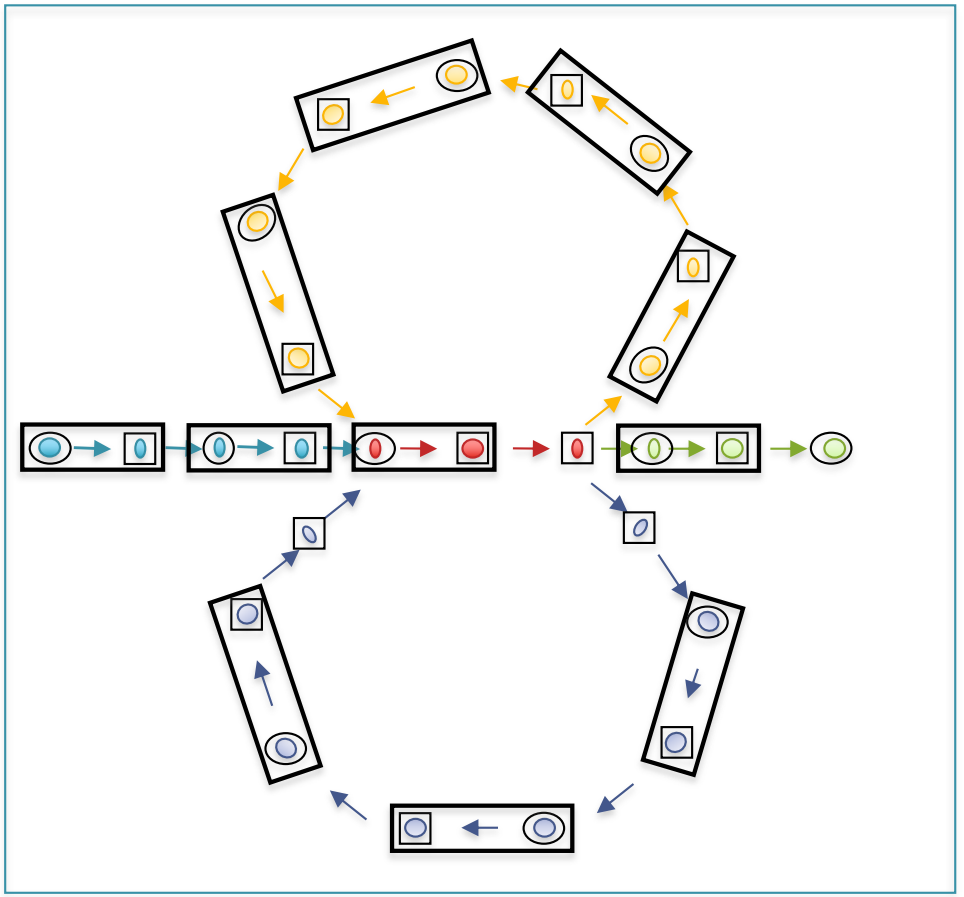
Fast Path Compression

Challenges

- Nodes stored on different computers
- Nodes can only access direct neighbors

Randomized List Ranking

- Randomly assign Ⓗ/ T to each compressible node
- Compress Ⓗ→T links



Round 1: 26 nodes (38% savings)

Randomized Speed-ups in Parallel Computation.
Vishkin U. (1984) *ACM Symposium on Theory of Computation. 230-239.*

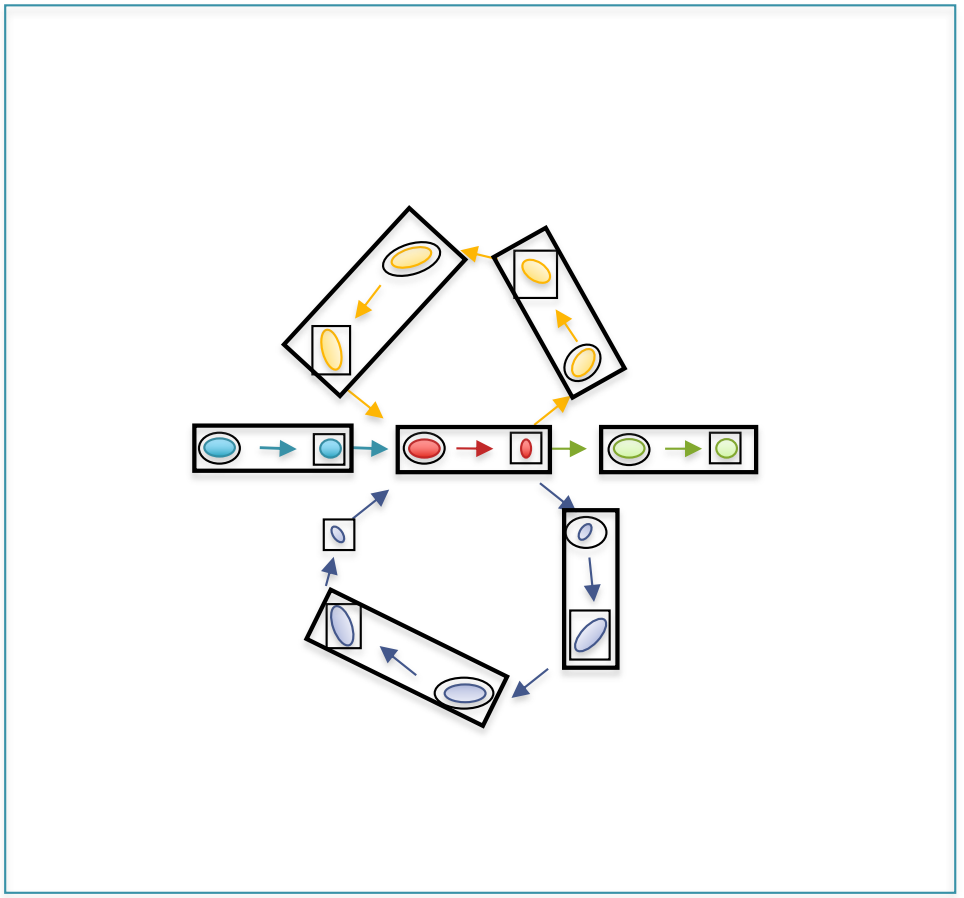
Fast Path Compression

Challenges

- Nodes stored on different computers
- Nodes can only access direct neighbors

Randomized List Ranking

- Randomly assign (H)/ [T] to each compressible node
- Compress (H)→[T] links



Round 2: 15 nodes (64% savings)

Randomized Speed-ups in Parallel Computation.
Vishkin U. (1984) *ACM Symposium on Theory of Computation. 230-239.*

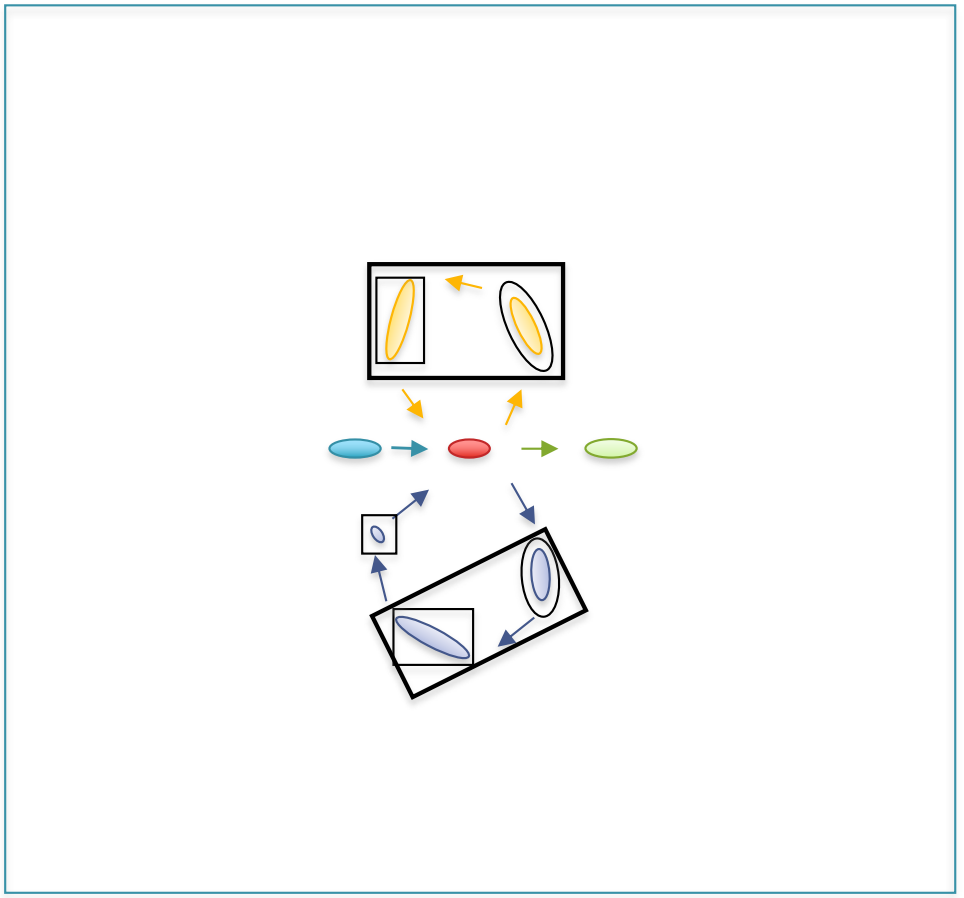
Fast Path Compression

Challenges

- Nodes stored on different computers
- Nodes can only access direct neighbors

Randomized List Ranking

- Randomly assign (H)/ [T] to each compressible node
- Compress (H)→[T] links



Round 2: 8 nodes (81% savings)

Randomized Speed-ups in Parallel Computation.
Vishkin U. (1984) *ACM Symposium on Theory of Computation. 230-239.*

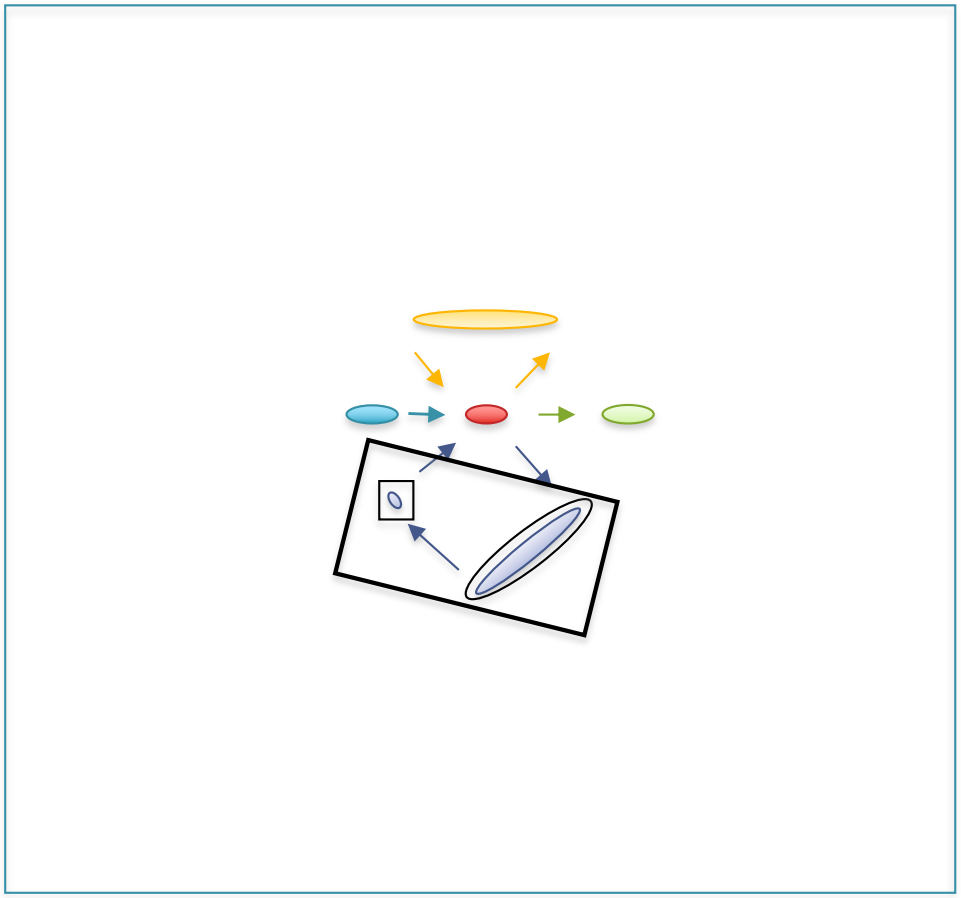
Fast Path Compression

Challenges

- Nodes stored on different computers
- Nodes can only access direct neighbors

Randomized List Ranking

- Randomly assign Ⓗ/ T to each compressible node
- Compress Ⓗ→T links



Round 3: 6 nodes (86% savings)

Randomized Speed-ups in Parallel Computation.
Vishkin U. (1984) *ACM Symposium on Theory of Computation. 230-239.*

Fast Path Compression

Challenges

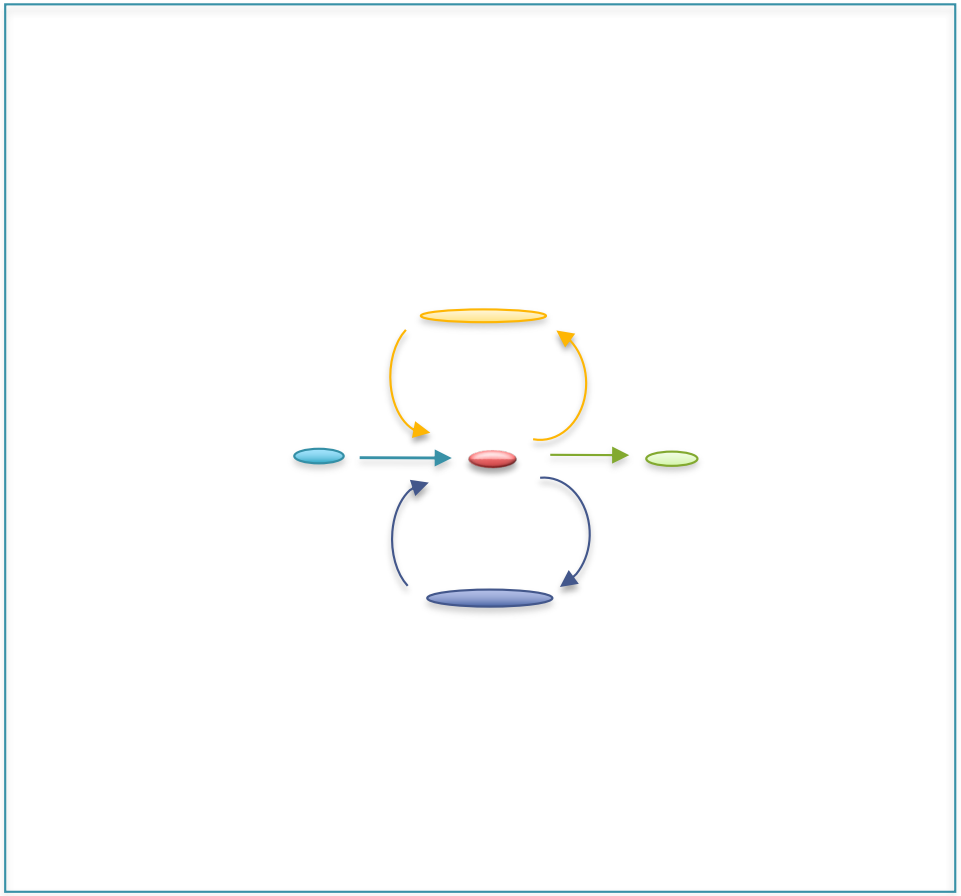
- Nodes stored on different computers
- Nodes can only access direct neighbors

Randomized List Ranking

- Randomly assign Ⓗ/ T to each compressible node
- Compress Ⓗ→T links

Performance

- Compress all chains in log(S) rounds



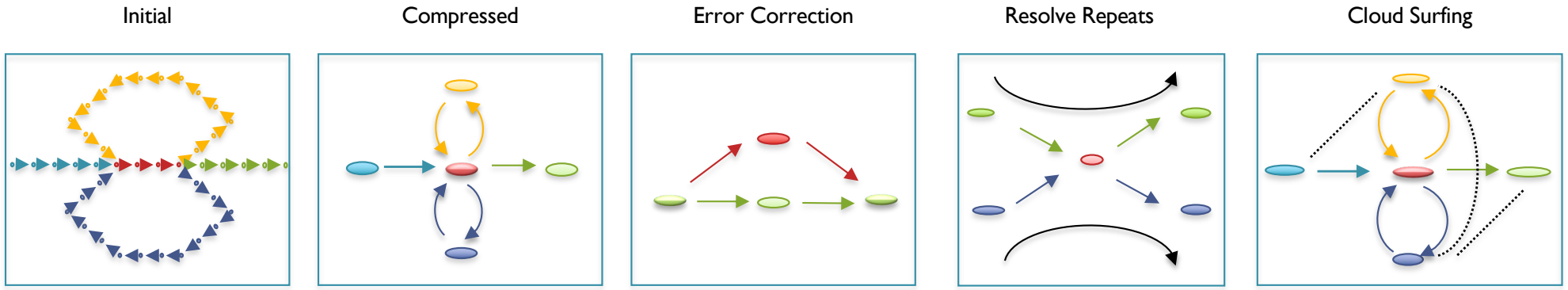
Round 4: 5 nodes (88% savings)

Randomized Speed-ups in Parallel Computation.
Vishkin U. (1984) *ACM Symposium on Theory of Computation. 230-239.*

Contrail

De novo Assembly of the Human Genome

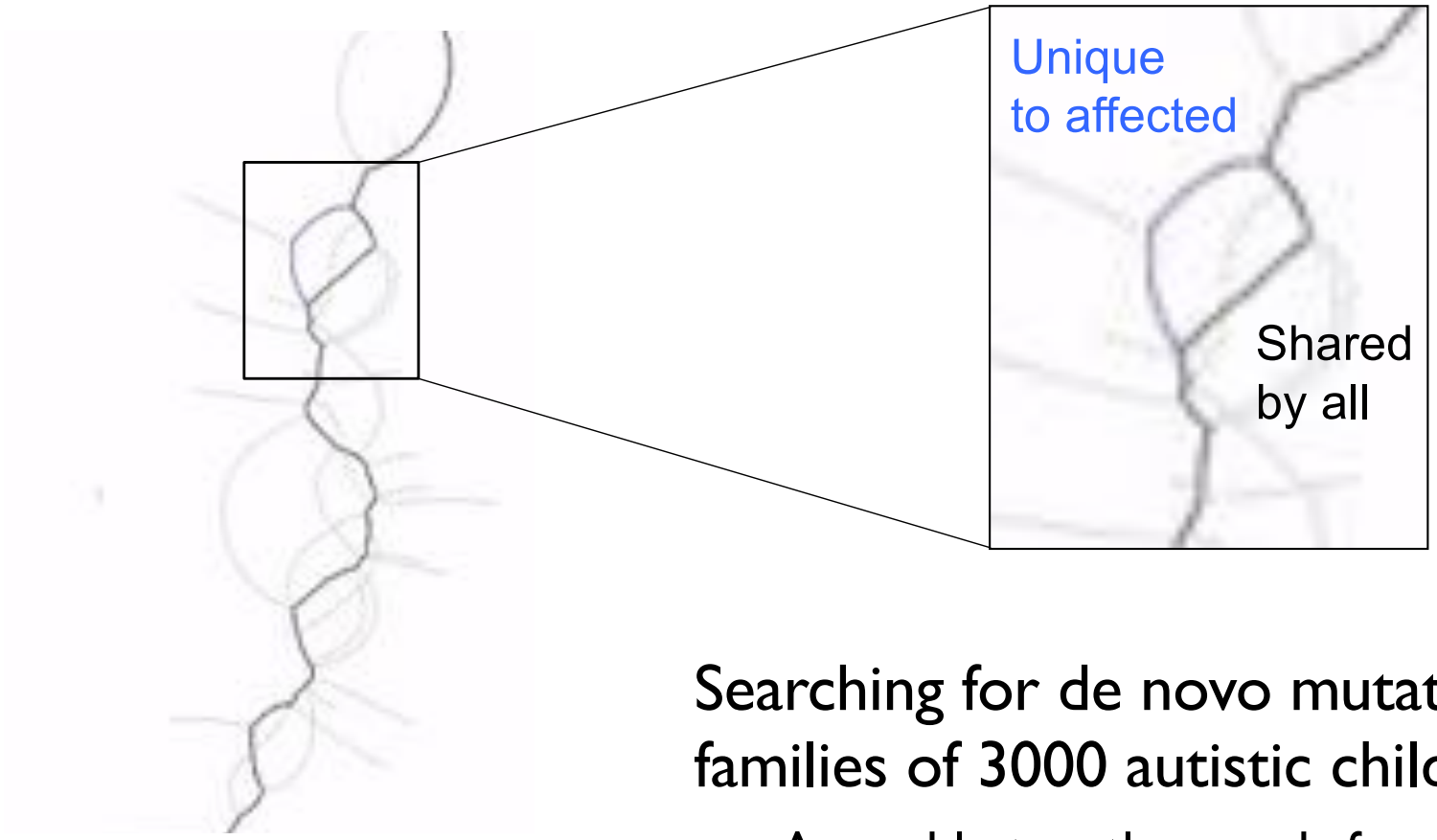
- *Genome:* African male NA18507 (SRA000271, Bentley *et al.*, 2008)
- *Input: 3.5*B 36bp reads, 210bp insert (~40x coverage)



N	>7 B	>1 B	4.2 M	4.1 M	3.3 M
Max	27 bp	303 bp	20,594 bp	20,594 bp	20,594 bp
N50	27 bp	< 100 bp	995 bp	1,050 bp	1,427 bp

Assembly of Large Genomes with Cloud Computing.
Schatz MC (2010) PhD Thesis. University of Maryland

De novo mutations and de Bruijn Graphs



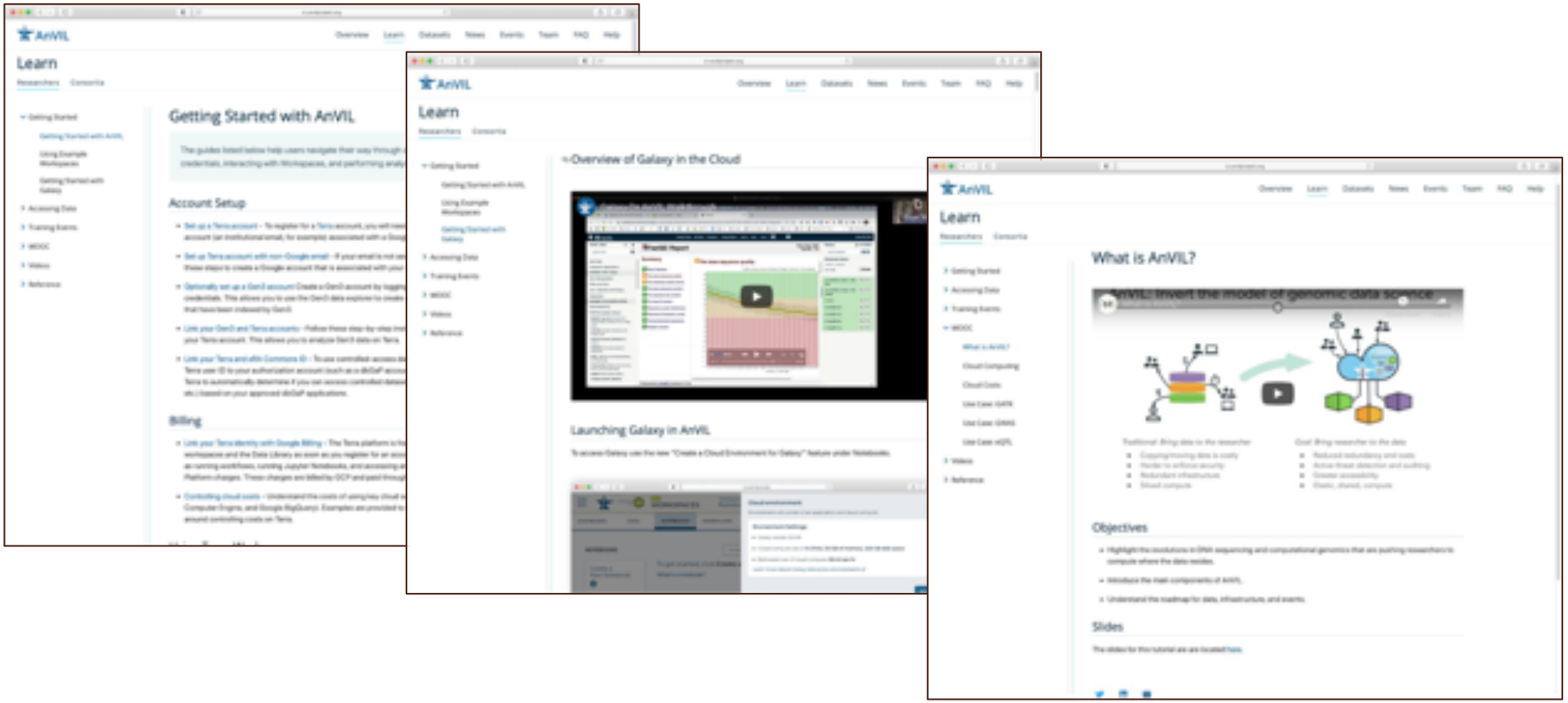
COLEC12
C->A

Searching for de novo mutations in the families of 3000 autistic children.

- Assemble together reads from mom, dad, affected & unaffected children
- Look for sequence paths unique to affected child

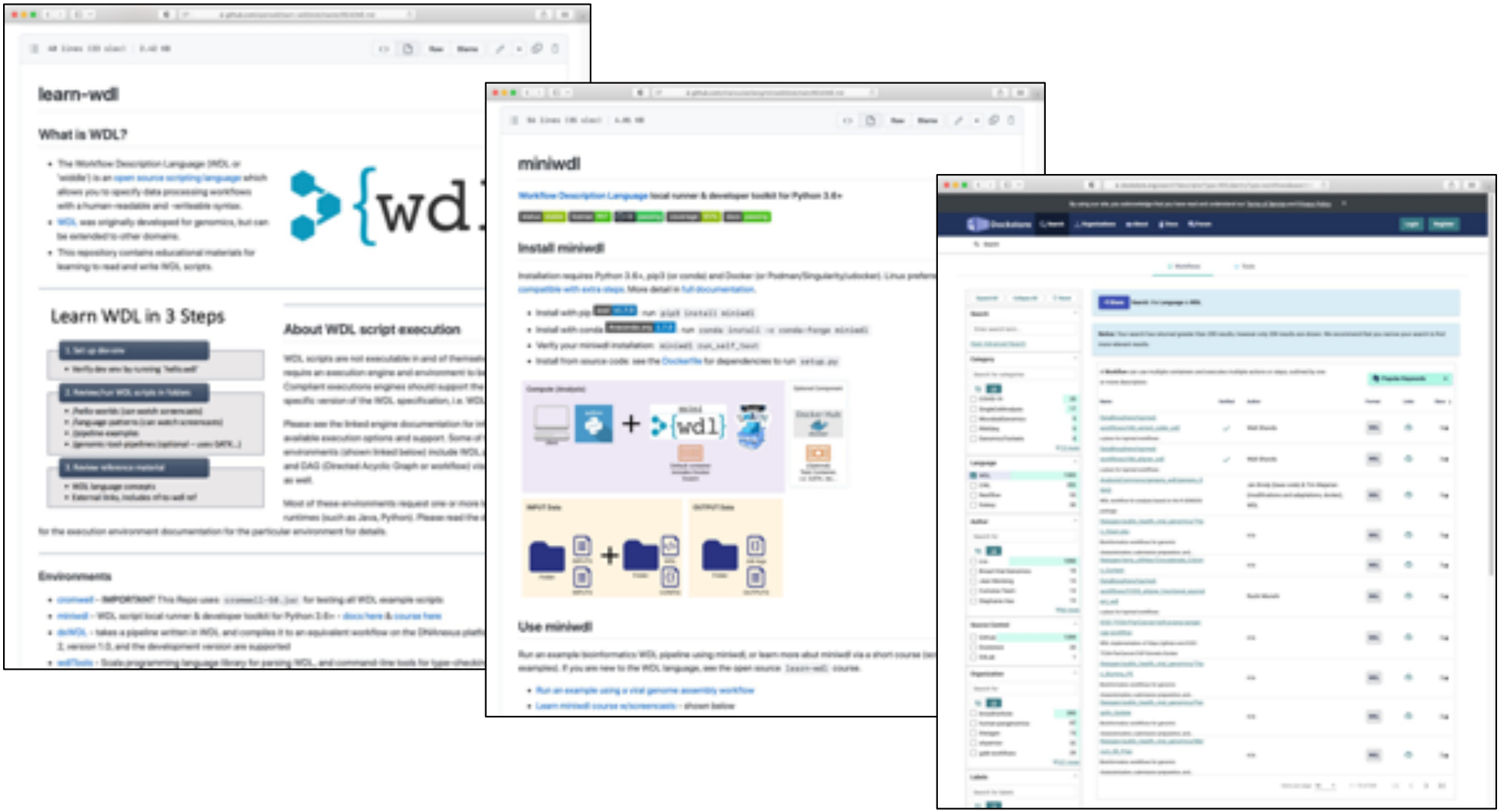
The contribution of de novo coding mutations to autism spectrum disorder
Iossifov *et al.* (2014) Nature doi:10.1038/nature13908

Learning More about AnVIL

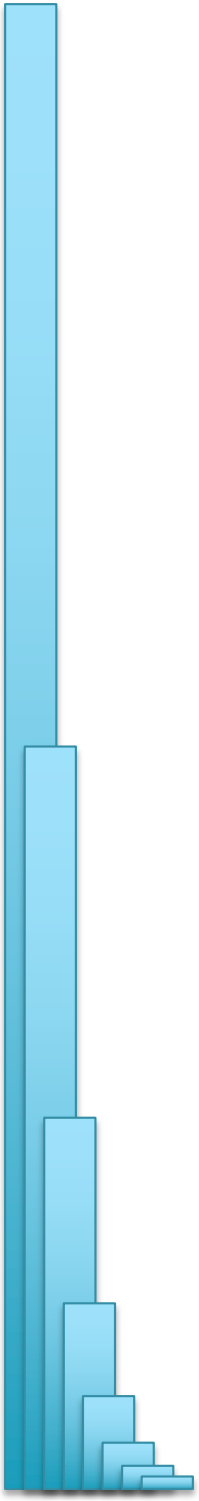


Portal: http://anvilproject.org/
Mailing List: help@lists.anvilproject.org
Discourse: http://help.anvil-project.org

Learning More about WDL



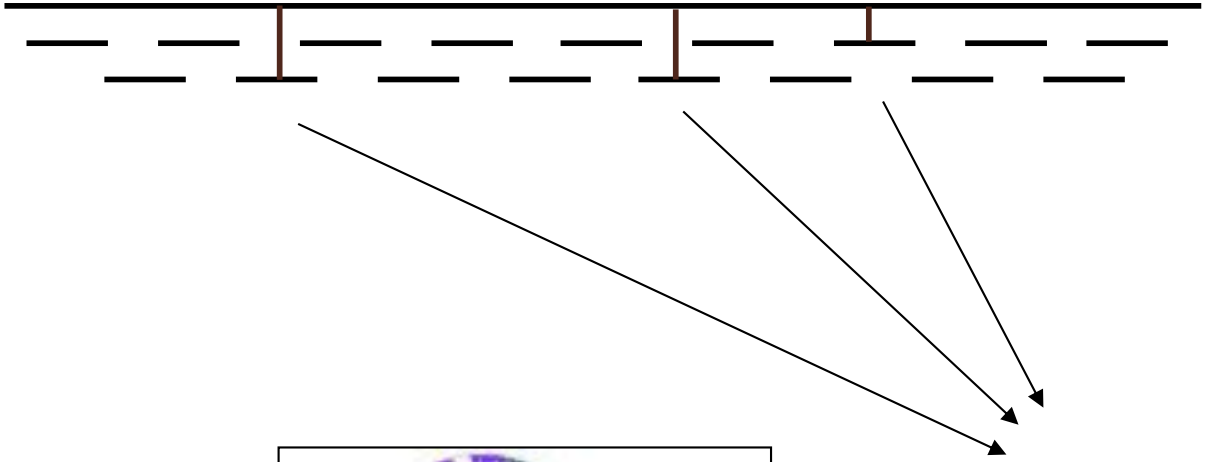
https://github.com/openwdl/learn-wdl/
https://github.com/chanzuckerberg/miniwdl/
https://dockstore.org/

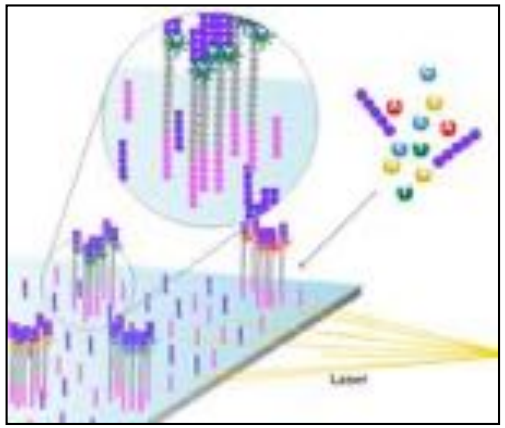


Read Mapping

Personal Genomics

How does your genome compare to the reference?



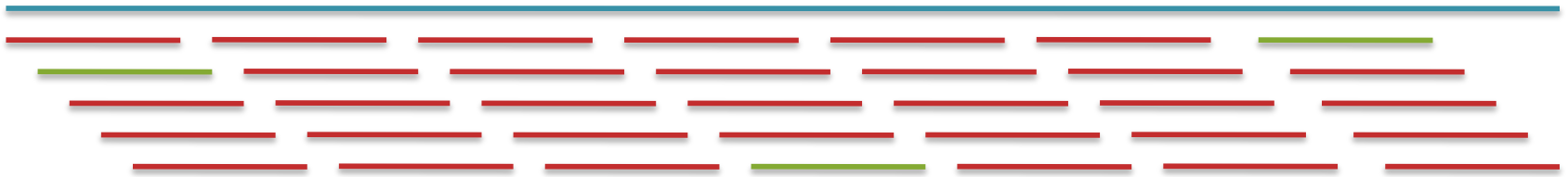


Heart Disease

Cancer

Presidential smile

Brute Force Analysis



- Brute Force:
  - At every possible offset in the genome:
    - Do all of the characters of the query match?

- Analysis
  - Simple, easy to understand
  - Genome length = n [3B]
  - Query length = m [7]
  - Comparisons: (n-m+1) * m [21B]

- Overall runtime: O(nm)

[How long would it take if we double the genome size, read length?]

[How long would it take if we double both?]

Brute Force Reflections

Why check every position?

– GATTACA can't possibly start at position 15 [WHY?]

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	…
T	G	A	T	T	A	C	A	G	A	T	T	A	C	C	…
								G	A	T	T	A	C	A	

– Improve runtime to O(n + m) [3B + 7]
  - If we double both, it just takes twice as long
  - Knuth-Morris-Pratt, 1977
  - Boyer-Moyer, 1977, 1991

– For one-off scans, this is the best we can do (optimal performance)
  - We have to read every character of the genome, and every character of the query
  - For short queries, runtime is dominated by the length of the genome