

Scalable Solutions for DNA Sequence Analysis


Daniel Sommer

April 13, 2010

University of Maryland



Outline

- 
- I. Genome Assembly by Analogy
 - DNA Sequencing and Genomics
 - MapReduce for Sequence Analysis
 - Genome Assembly
 - K-mer counting
 - Read Mapping & Genotyping

Shredded Book Reconstruction

- Dickens accidentally shreds the first printing of A Tale of Two Cities
 - Text printed on 5 long spools

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

Shredded Book Reconstruction

- Dickens accidentally shreds the first printing of A Tale of Two Cities
 - Text printed on 5 long spools

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

Shredded Book Reconstruction

- Dickens accidentally shreds the first printing of A Tale of Two Cities
 - Text printed on 5 long spools

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

- How can he reconstruct the text?
 - 5 copies x 138,656 words / 5 words per fragment = 138k fragments
 - The short fragments from every copy are mixed together
 - Some fragments are identical

Shredded Book Reconstruction

- Dickens accidentally shreds the first printing of A Tale of Two Cities
 - Text printed on 5 long spools

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, ...

- How can he reconstruct the text?
 - 5 copies x 138,656 words / 5 words per fragment = 138k fragments
 - The short fragments from every copy are mixed together
 - Some fragments are identical

Greedy Reconstruction

Greedy Reconstruction

It was the best of

age of wisdom, it was

best of times, it was

it was the age of

it was the age of

it was the worst of

of times, it was the

of times, it was the

of wisdom, it was the

the age of wisdom, it

the best of times, it

the worst of times, it

times, it was the age

times, it was the worst

was the age of wisdom,

was the age of foolishness,

was the best of times,

was the worst of times,

wisdom, it was the age

worst of times, it was

Greedy Reconstruction

It was the best of

age of wisdom, it was

best of times, it was

it was the age of

it was the age of

it was the worst of

of times, it was the

of times, it was the

of wisdom, it was the

the age of wisdom, it

the best of times, it

the worst of times, it

times, it was the age

times, it was the worst

was the age of wisdom,

was the age of foolishness,

was the best of times,

was the worst of times,

wisdom, it was the age

worst of times, it was

It was the best of

Greedy Reconstruction

It was the best of

age of wisdom, it was

best of times, it was

it was the age of

it was the age of

it was the worst of

of times, it was the

of times, it was the

of wisdom, it was the

the age of wisdom, it

the best of times, it

the worst of times, it

times, it was the age

times, it was the worst

was the age of wisdom,

was the age of foolishness,

was the best of times,

was the worst of times,

wisdom, it was the age

worst of times, it was

It was the best of

was the best of times,

Greedy Reconstruction

It was the best of

age of wisdom, it was

best of times, it was

it was the age of

it was the age of

it was the worst of

of times, it was the

of times, it was the

of wisdom, it was the

the age of wisdom, it

the best of times, it

the worst of times, it

times, it was the age

times, it was the worst

was the age of wisdom,

was the age of foolishness,

was the best of times,

was the worst of times,

wisdom, it was the age

worst of times, it was

It was the best of

was the best of times,

the best of times, it

Greedy Reconstruction

It was the best of

age of wisdom, it was

best of times, it was

it was the age of

it was the age of

it was the worst of

of times, it was the

of times, it was the

of wisdom, it was the

the age of wisdom, it

the best of times, it

the worst of times, it

times, it was the age

times, it was the worst

was the age of wisdom,

was the age of foolishness,

was the best of times,

was the worst of times,

wisdom, it was the age

worst of times, it was

It was the best of

was the best of times,

the best of times, it

best of times, it was

Greedy Reconstruction

It was the best of

age of wisdom, it was

best of times, it was

it was the age of

it was the age of

it was the worst of

of times, it was the

of times, it was the

of wisdom, it was the

the age of wisdom, it

the best of times, it

the worst of times, it

times, it was the age

times, it was the worst

was the age of wisdom,

was the age of foolishness,

was the best of times,

was the worst of times,

wisdom, it was the age

worst of times, it was

It was the best of

was the best of times,

the best of times, it

best of times, it was

of times, it was the

of times, it was the

Greedy Reconstruction

It was the best of

age of wisdom, it was

best of times, it was

it was the age of

it was the age of

it was the worst of

of times, it was the

of times, it was the

of wisdom, it was the

the age of wisdom, it

the best of times, it

the worst of times, it

times, it was the age

times, it was the worst

was the age of wisdom,

was the age of foolishness,

was the best of times,

was the worst of times,

wisdom, it was the age

worst of times, it was

It was the best of

was the best of times,

the best of times, it

best of times, it was

of times, it was the

of times, it was the

times, it was the worst

times, it was the age

The repeated sequence make the correct reconstruction ambiguous

- It was the best of times, it was the [worst/age]

Model sequence reconstruction as a graph problem.

de Bruijn Graph Construction

- $D_k = (V, E)$
 - $V =$ All length- k subfragments ($k < l$)
 - $E =$ Directed edges between consecutive subfragments
 - Nodes overlap by $k-1$ words

Original Fragment

It was the best of

Directed Edge

It was the best → was the best of

- Locally constructed graph reveals the global sequence structure
 - Overlaps between sequences implicitly computed

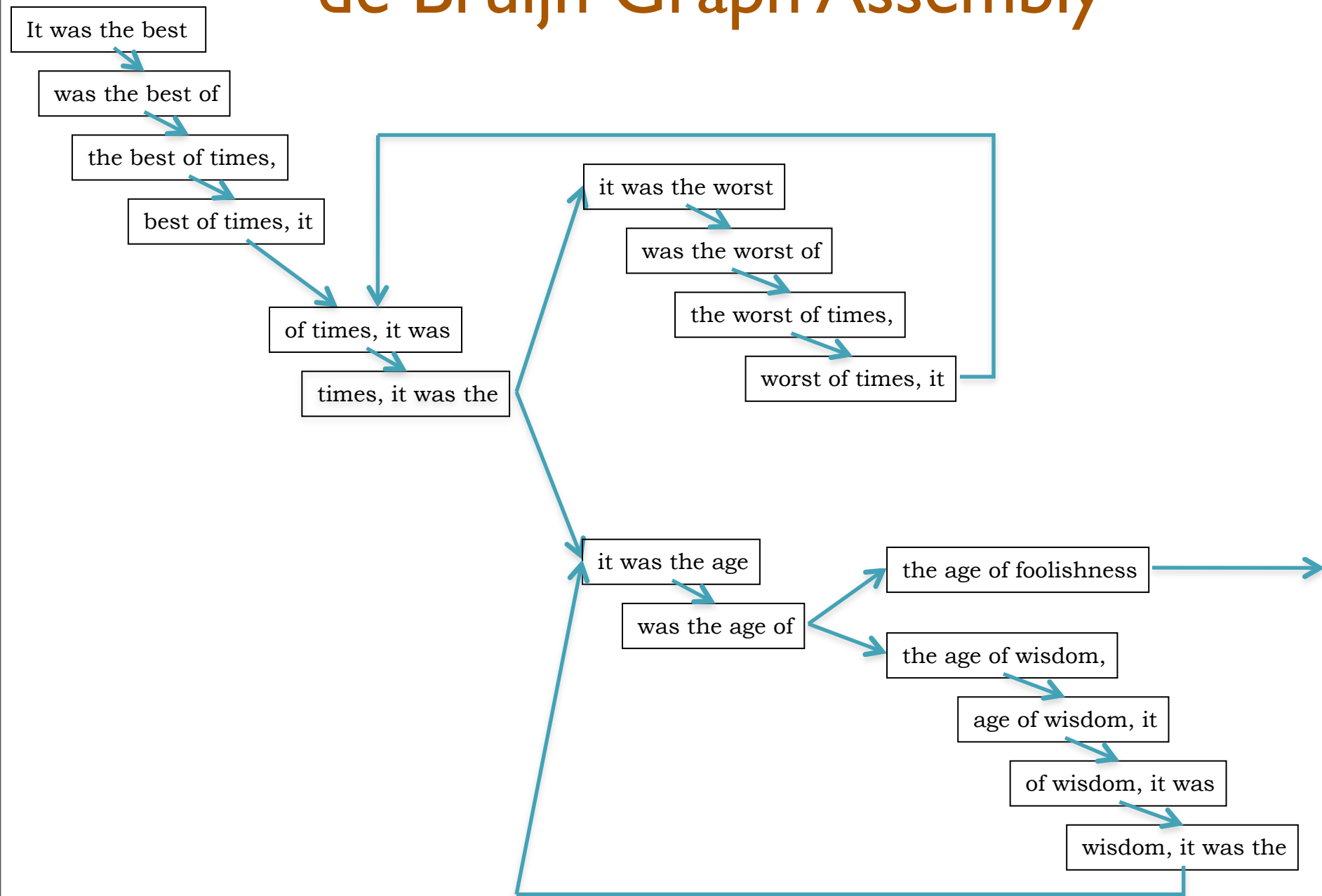
de Bruijn, 1946

Idury and Waterman, 1995

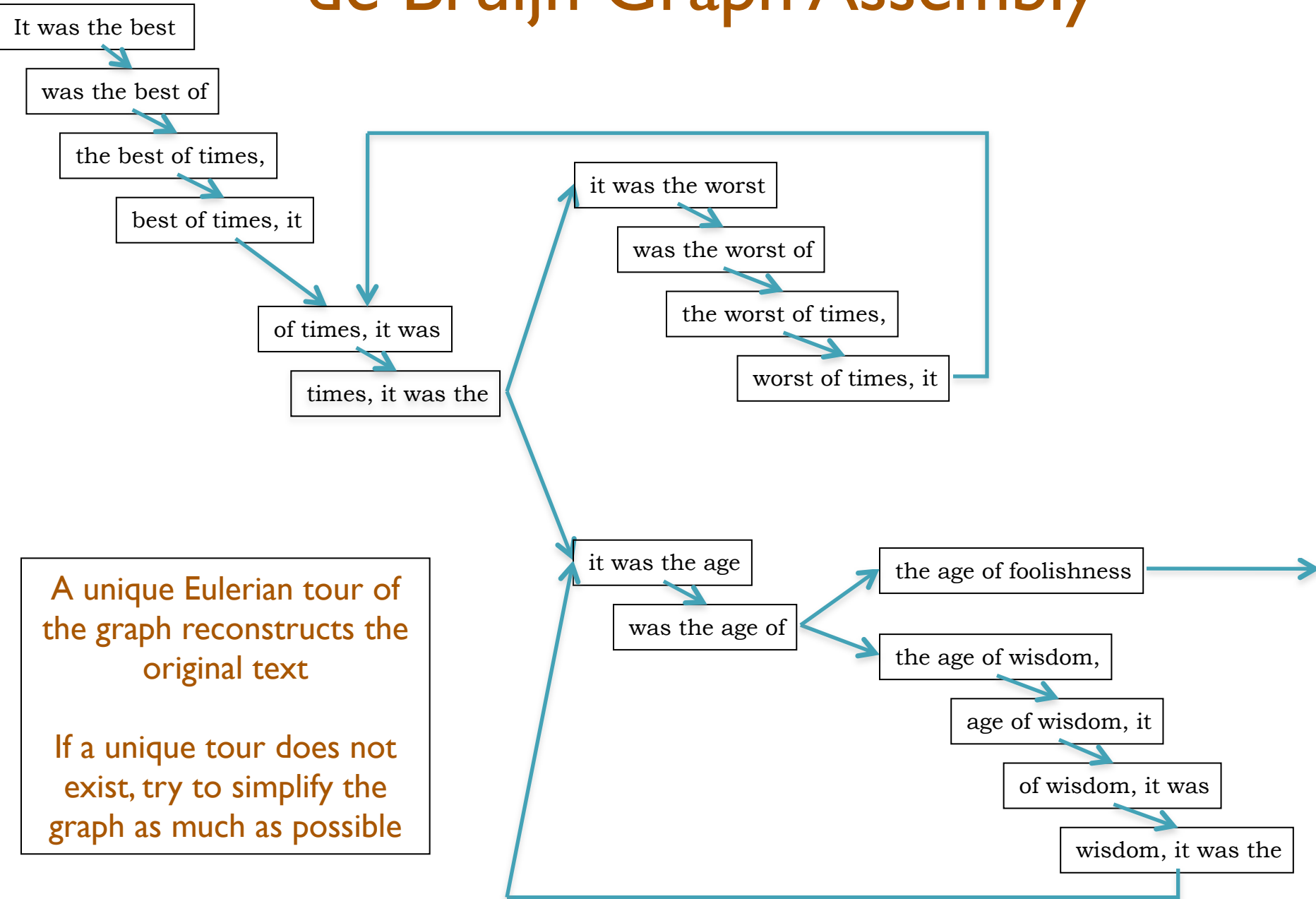
Pevzner, Tang, Waterman, 2001

de Bruijn Graph Assembly

de Bruijn Graph Assembly



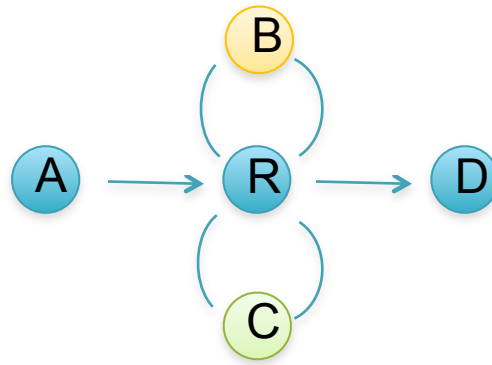
de Bruijn Graph Assembly



A unique Eulerian tour of the graph reconstructs the original text

If a unique tour does not exist, try to simplify the graph as much as possible

Counting Eulerian Tours



AR**B**RCRD
or
ARC**R**BRD

Generally an exponential number of compatible sequences

- Value computed by application of the BEST theorem (Hutchinson, 1975)

$$\mathcal{W}(G, t) = (\det L) \left\{ \prod_{u \in V} (r_u - 1)! \right\} \left\{ \prod_{(u,v) \in E} a_{uv}! \right\}^{-1}$$

$L = n \times n$ matrix with $r_u - a_{uu}$ along the diagonal and $-a_{uv}$ in entry uv

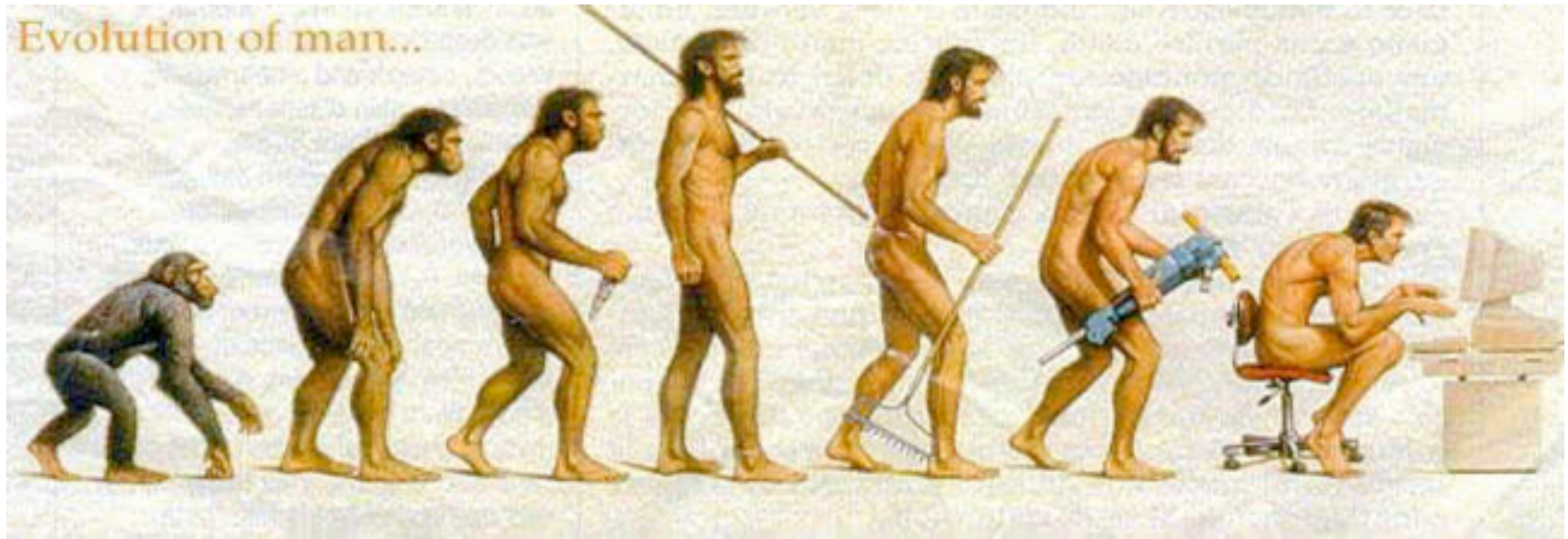
$r_u = d^+(u) + 1$ if $u=t$, or $d^+(u)$ otherwise

$a_{uv} =$ multiplicity of edge from u to v

Assembly Complexity of Prokaryotic Genomes using Short Reads.

Kingsford C, Schatz MC, Pop M (2010) *BMC Bioinformatics*.

Genomics



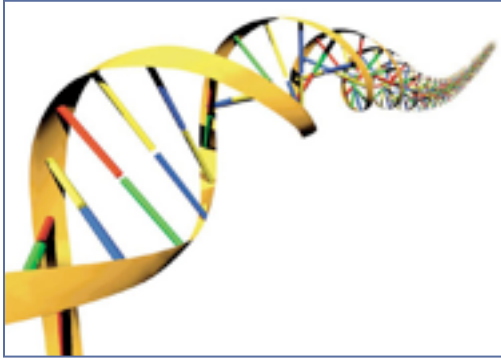
Your genome influences (almost) all aspects of your life

- Anatomy & Physiology: 10 fingers & 10 toes, organs, neurons
- Diseases: Sickle Cell Anemia, Down Syndrome, Cancer
- Psychological: Intelligence, Personality, Bad Driving

Your environment also influences your life

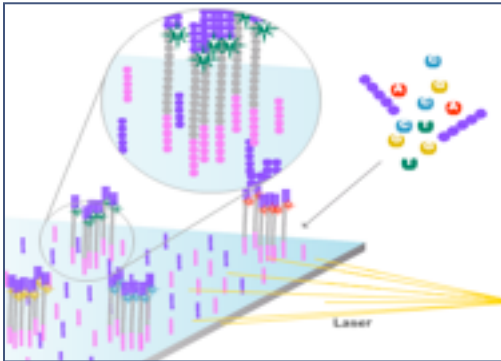
- Genome as a recipe, not a blueprint

DNA Sequencing



Genome of an organism encodes the genetic information in long sequence of 4 DNA nucleotides:ACGT

- Bacteria: ~3 million bp
- Humans: ~3 billion bp



Current DNA sequencing machines can generate 1-2 Gbp of sequence per day, in millions of short reads

- Per-base error rate estimated at 1-2% (Simpson *et al*, 2009)
- Sequences originate from random positions of the genome

ATCTGATAAGTCCCAGGACTTCAGT

GCAAGGCAAACCCGAGCCCAGTTT

TCCAGTTCTAGAGTTTCACATGATC

GGAGTTAGTAAAAGTCCACATTGAG

Recent studies of entire human genomes analyzed 3.3B (Wang, et al., 2008) & 4.0B (Bentley, et al., 2008) 36bp reads

- ~100 GB of compressed sequence data

The Evolution of DNA Sequencing

Year	Genome	Technology	Cost
2001	Venter <i>et al.</i>	Sanger (ABI)	\$300,000,000
2007	Levy <i>et al.</i>	Sanger (ABI)	\$10,000,000
2008	Wheeler <i>et al.</i>	Roche (454)	\$2,000,000
2008	Ley <i>et al.</i>	Illumina	\$1,000,000
2008	Bentley <i>et al.</i>	Illumina	\$250,000
2009	Pushkarev <i>et al.</i>	Helicos	\$48,000
2009	Drmanac <i>et al.</i>	Complete Genomics	\$4,400

(Pushkarev *et al.*, 2009)



Critical Computational Challenges: Alignment and Assembly of Huge Datasets

Hadoop MapReduce

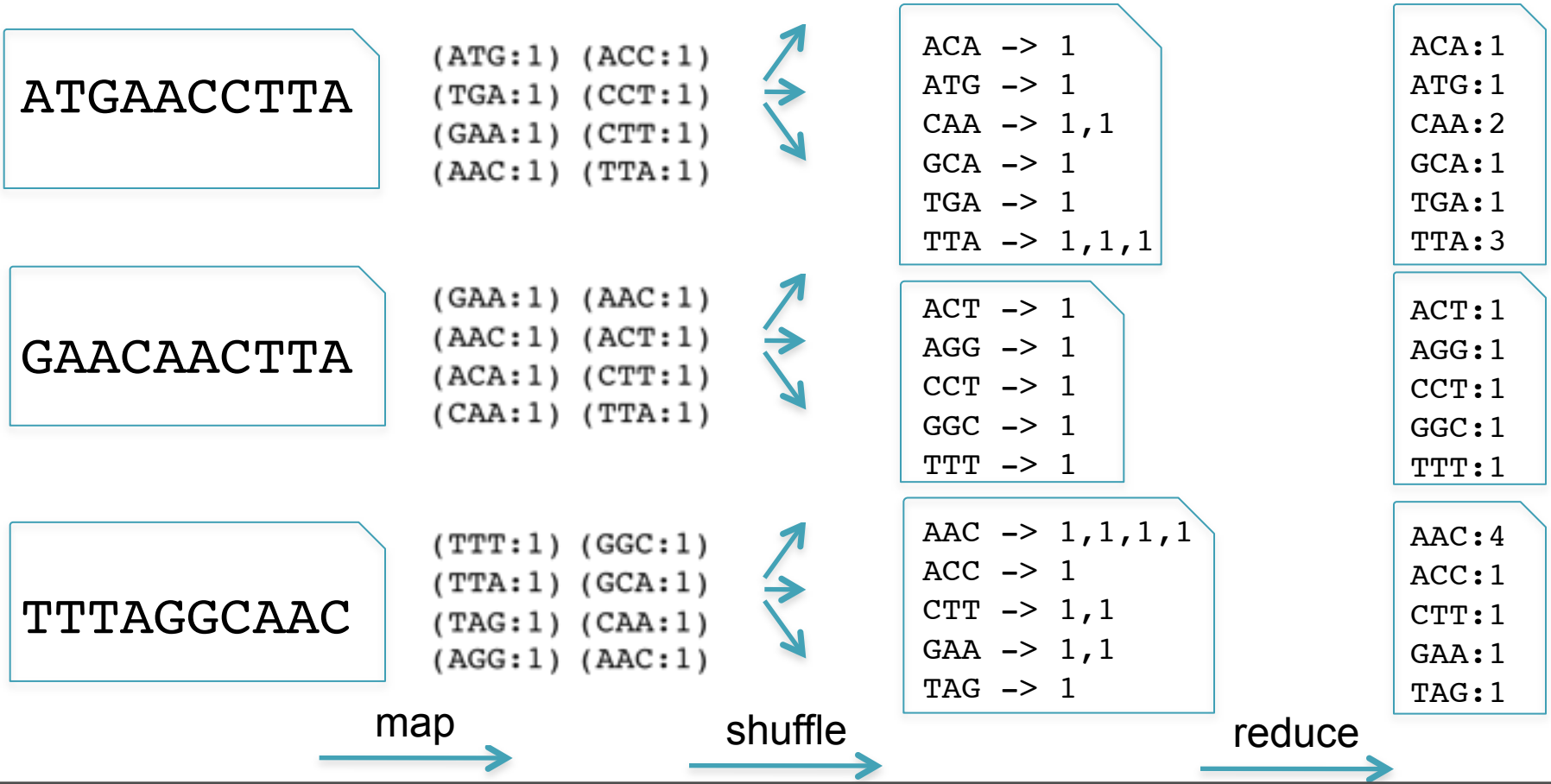
- MapReduce is the parallel distributed framework invented by Google for large data computations.
 - Data and computations are spread over thousands of computers, processing petabytes of data each day (Dean and Ghemawat, 2004)
 - Indexing the Internet, PageRank, Machine Learning, etc...
 - Hadoop is the leading open source implementation
- Benefits
 - Scalable, Efficient, Reliable
 - Easy to Program
 - Runs on commodity computers
- Challenges
 - Redesigning / Retooling applications
 - Not Condor, Not MPI
 - Everything in MapReduce



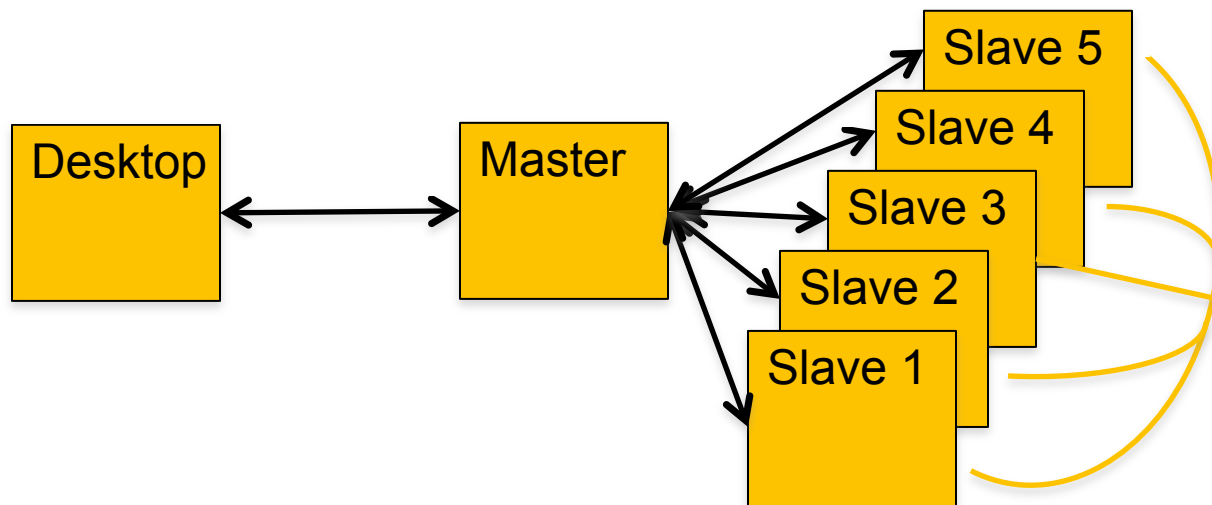
K-mer Counting

- Application developers focus on 2 (+1 internal) functions
 - **Map:** input \rightarrow key:value pairs
 - **Shuffle:** Group together pairs with same key
 - **Reduce:** key, value-lists \rightarrow output

Map, Shuffle & Reduce
All Run in Parallel

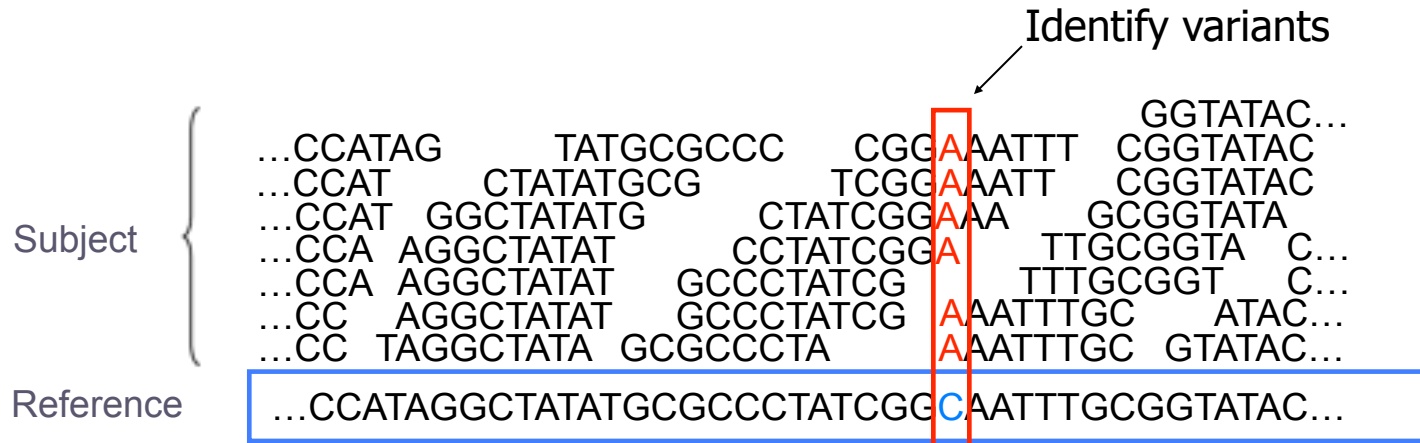


Hadoop Architecture



- Hadoop Distributed File System (HDFS)
 - Data files partitioned into large chunks (64MB), replicated on multiple nodes
 - NameNode stores metadata information (block locations, directory structure)
- Master node (JobTracker) schedules and monitors work on slaves
 - Computation moves to the data, rack-aware scheduling
- Hadoop MapReduce system won the 2009 GreySort Challenge
 - Sorted 100 TB in 173 min (578 GB/min) using 3452 nodes and 4x3452 disks

Short Read Mapping



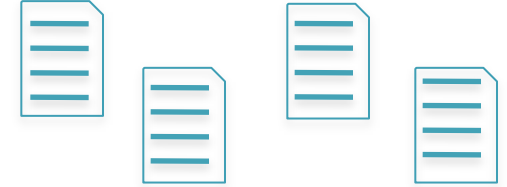
- Given a reference and many subject reads, report one or more “good” end-to-end alignments per alignable read
 - Find where the read most likely originated
 - Fundamental computation for many assays
 - Genotyping
 - RNA-Seq
 - Methyl-Seq
 - Structural Variations
 - Chip-Seq
 - Hi-C-Seq
- Desperate need for scalable solutions
 - Single human requires >1,000 CPU hours / genome

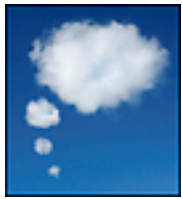


Crossbow

<http://bowtie-bio.sourceforge.net/crossbow>

- Align billions of reads and find SNPs
 - Reuse software components: Hadoop Streaming

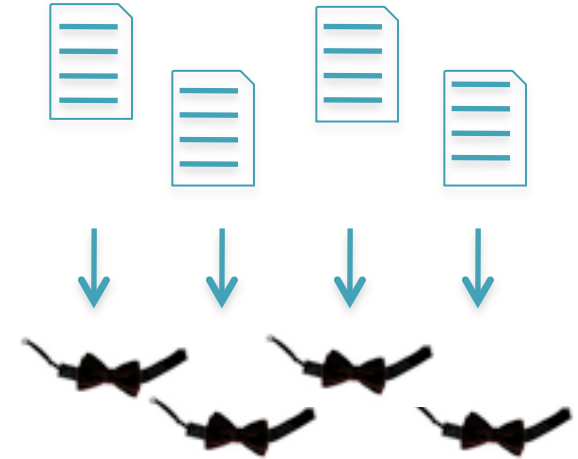


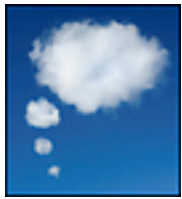


Crossbow

<http://bowtie-bio.sourceforge.net/crossbow>

- Align billions of reads and find SNPs
 - Reuse software components: Hadoop Streaming
- Map: Bowtie (Langmead *et al.*, 2009)
 - Find best alignment for each read
 - Emit (chromosome region, alignment)

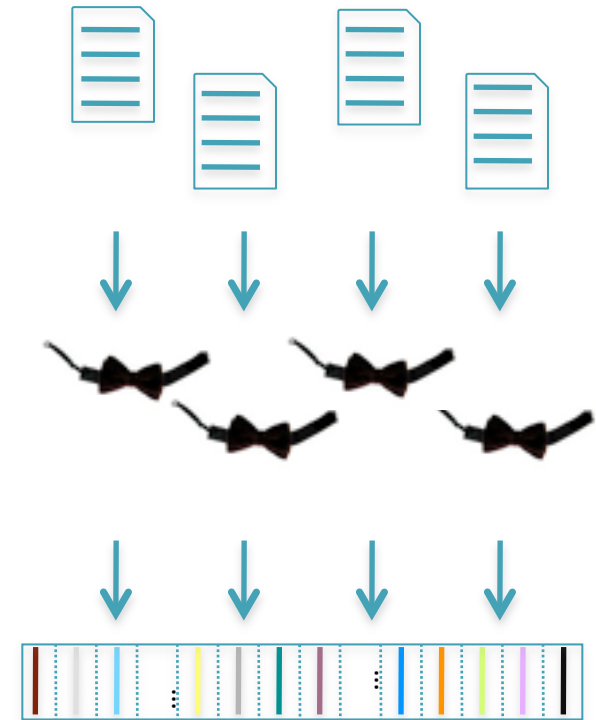


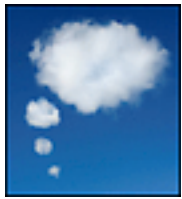


Crossbow

<http://bowtie-bio.sourceforge.net/crossbow>

- Align billions of reads and find SNPs
 - Reuse software components: Hadoop Streaming
- Map: Bowtie (Langmead *et al.*, 2009)
 - Find best alignment for each read
 - Emit (chromosome region, alignment)
- Shuffle: Hadoop
 - Group and sort alignments by region

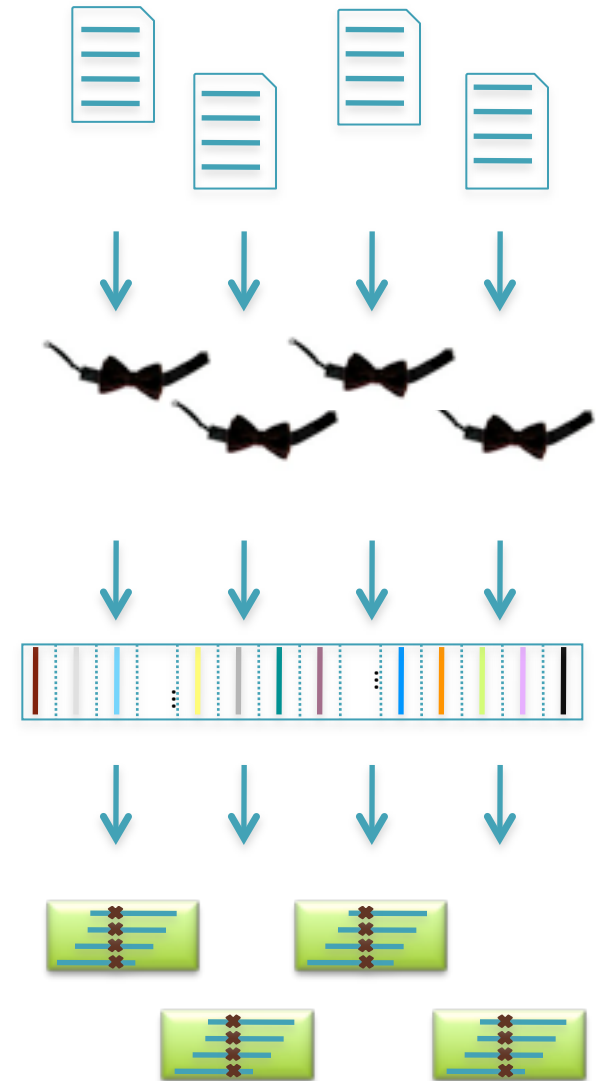




Crossbow

<http://bowtie-bio.sourceforge.net/crossbow>

- Align billions of reads and find SNPs
 - Reuse software components: Hadoop Streaming
- 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



Performance in Amazon EC2

<http://bowtie-bio.sourceforge.net/crossbow>

	Asian Individual Genome		
Data Loading	3.3 B reads	106.5 GB	\$10.65
Data Transfer	1h : 15m	40 cores	\$3.40
Setup	0h : 15m	320 cores	\$13.94
Alignment	1h : 30m	320 cores	\$41.82
Variant Calling	1h : 00m	320 cores	\$27.88
End-to-end	4h : 00m		\$97.69

Analyze an entire human genome for ~\$100 in an afternoon.
Accuracy validated at >99%

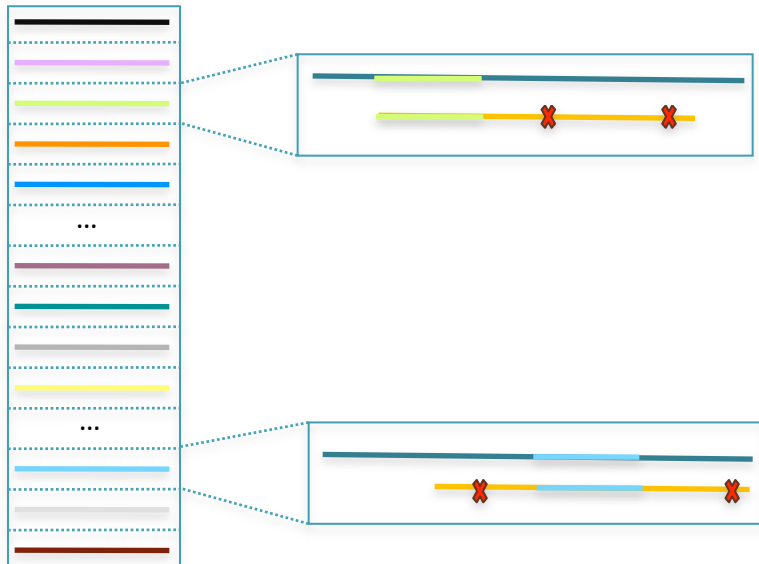
Searching for SNPs with Cloud Computing.

Langmead B, Schatz MC, Lin J, Pop M, Salzberg SL (2009) *Genome Biology*.

Related Approaches

CloudBurst

Highly Sensitive Short Read Mapping
with MapReduce

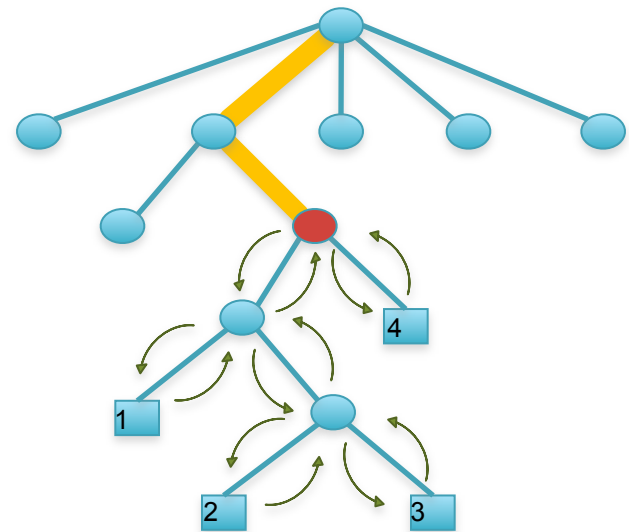


100x speedup on 96 cores @ Amazon

(Schatz, 2009)

MUMmerGPU

High Throughput Sequence Alignment
using GPGPUs

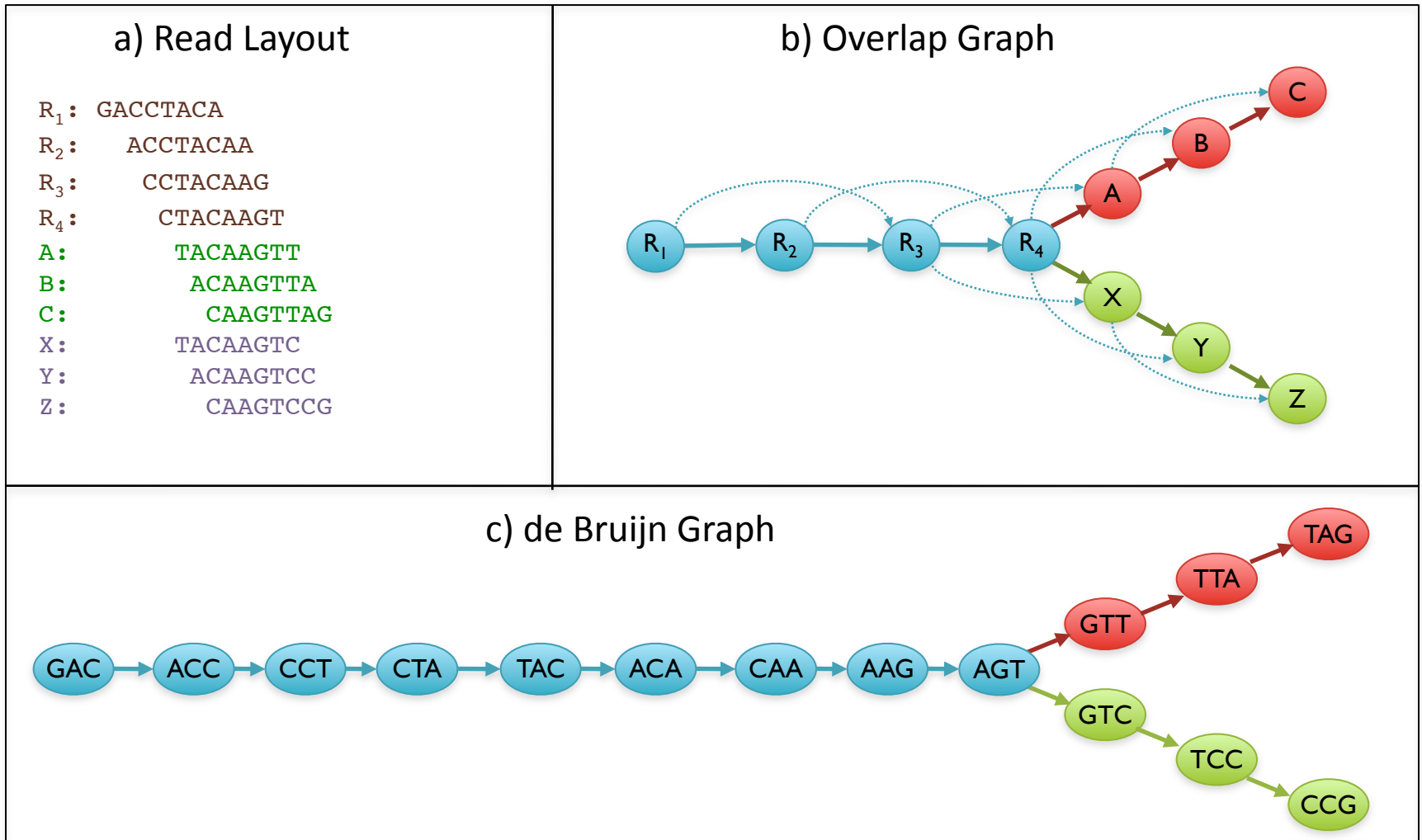


~10x speedup on nVidia GTX 8800

(Schatz, Trapnell, *et al.*, 2007)

(Trapnell & Schatz, 2008)

Two Paradigms for Assembly



Large-Scale Genome Assembly from Short Reads.

Schatz MC, Delcher AL, Salzberg SL (2010) *Manuscript Under Review.*

Short Read Assembly

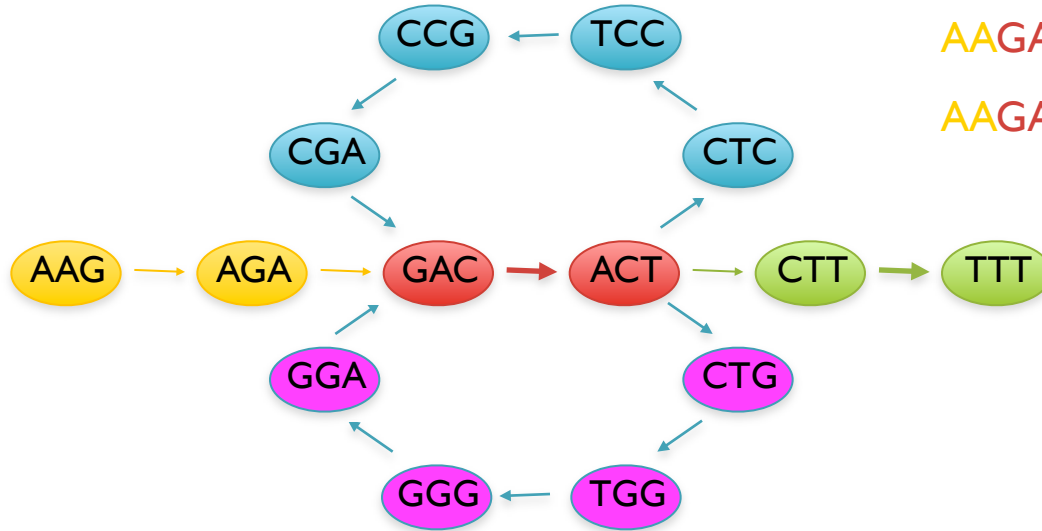
Reads

AAGA
ACTT

ACTC
ACTG
AGAG
CCGA
CGAC
CTCC
CTGG
CTTT

...

de Bruijn Graph



Potential Genomes

AAGACTCCGACTGGGACTTTT

AAGACTGGGACTCCGACTTTT

- Genome assembly as finding an Eulerian tour of the de Bruijn graph
 - Human genome: >3B nodes, >10B edges
- The new short read assemblers require tremendous computation
 - Velvet (Zerbino & Birney, 2008) serial: > 2TB of RAM
 - ABySS (Simpson *et al.*, 2009) MPI: 168 cores x ~96 hours
 - SOAPdenovo (Li *et al.*, 2010) pthreads: 40 cores x 40 hours, >140 GB RAM

Contrail

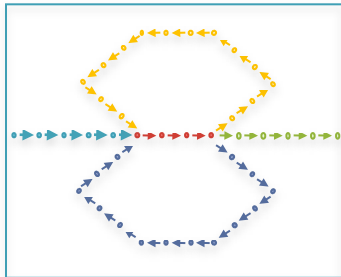
<http://contrail-bio.sourceforge.net>



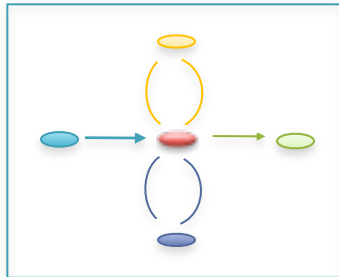
Scalable Genome Assembly with MapReduce

- *Genome: E. coli* 4.6Mbp bacteria
- *Input: 20M* 36bp reads, 200bp insert
- *Preprocessor: Quality-Aware Error Correction*

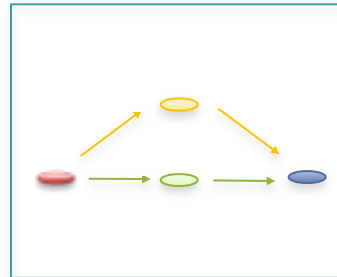
Initial



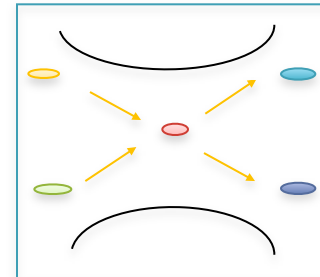
Compressed



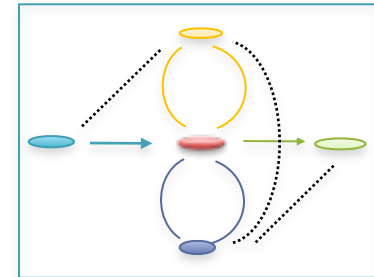
Error Correction



Resolve Repeats



Cloud Surfing



N	5.1 M	245,131	2,769	1,909	300
Max	27 bp	1,079 bp	70,725 bp	90,088 bp	149,006 bp
N50	27 bp	156 bp	15,023 bp	17,058 bp	54,807 bp

Assembly of Large Genomes with Cloud Computing.

Schatz MC, Sommer D, Kelley D, Pop M, et al. *In Preparation.*

Traditional Assembly on MapReduce

- How do you adapt the traditional overlap-layout-consensus assembler to the MapReduce parallel programming model?

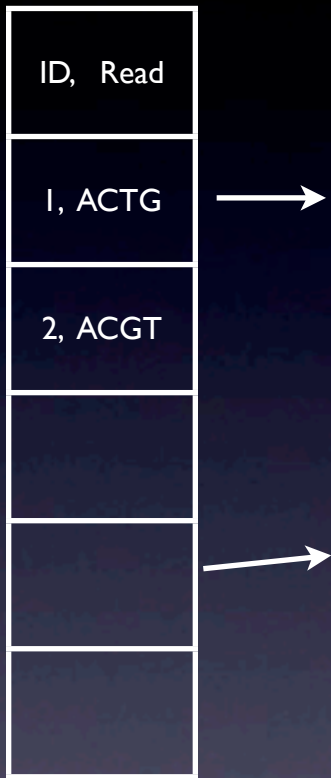
Overlap Stage

- Compute all pair wise alignments between reads
- Ideal for MapReduce because aligning two reads can be done independent of all other reads
- Use seed and extend algorithm that is currently used for the overlapper

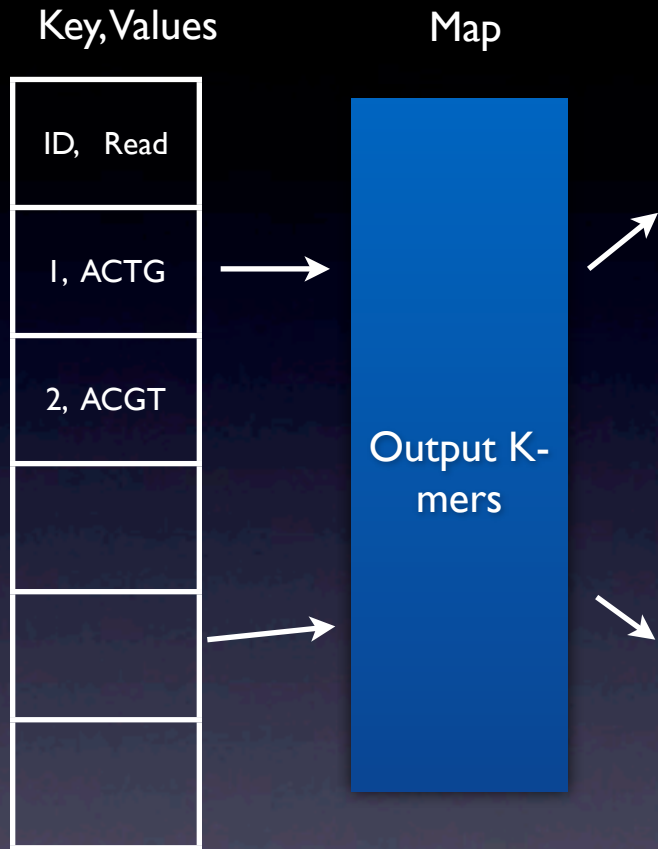
MapReduce Hash-Overlapper

MapReduce Hash-Overlapper

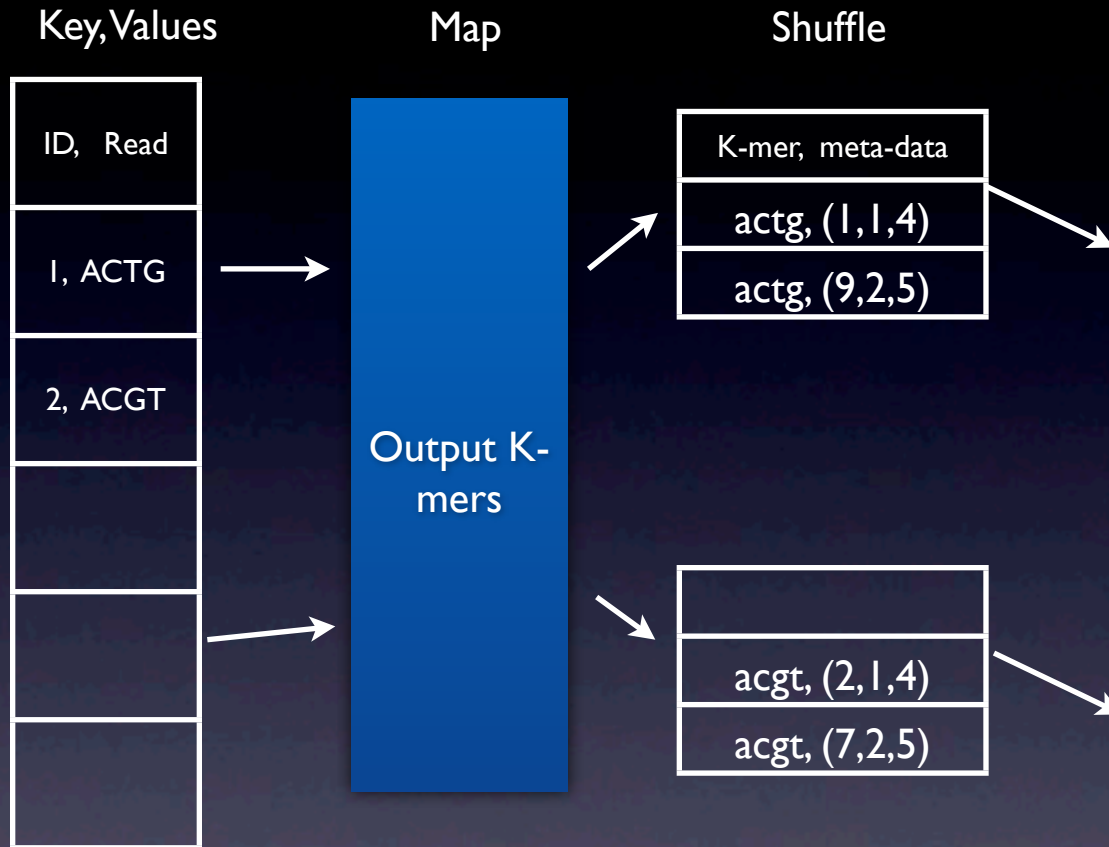
Key, Values



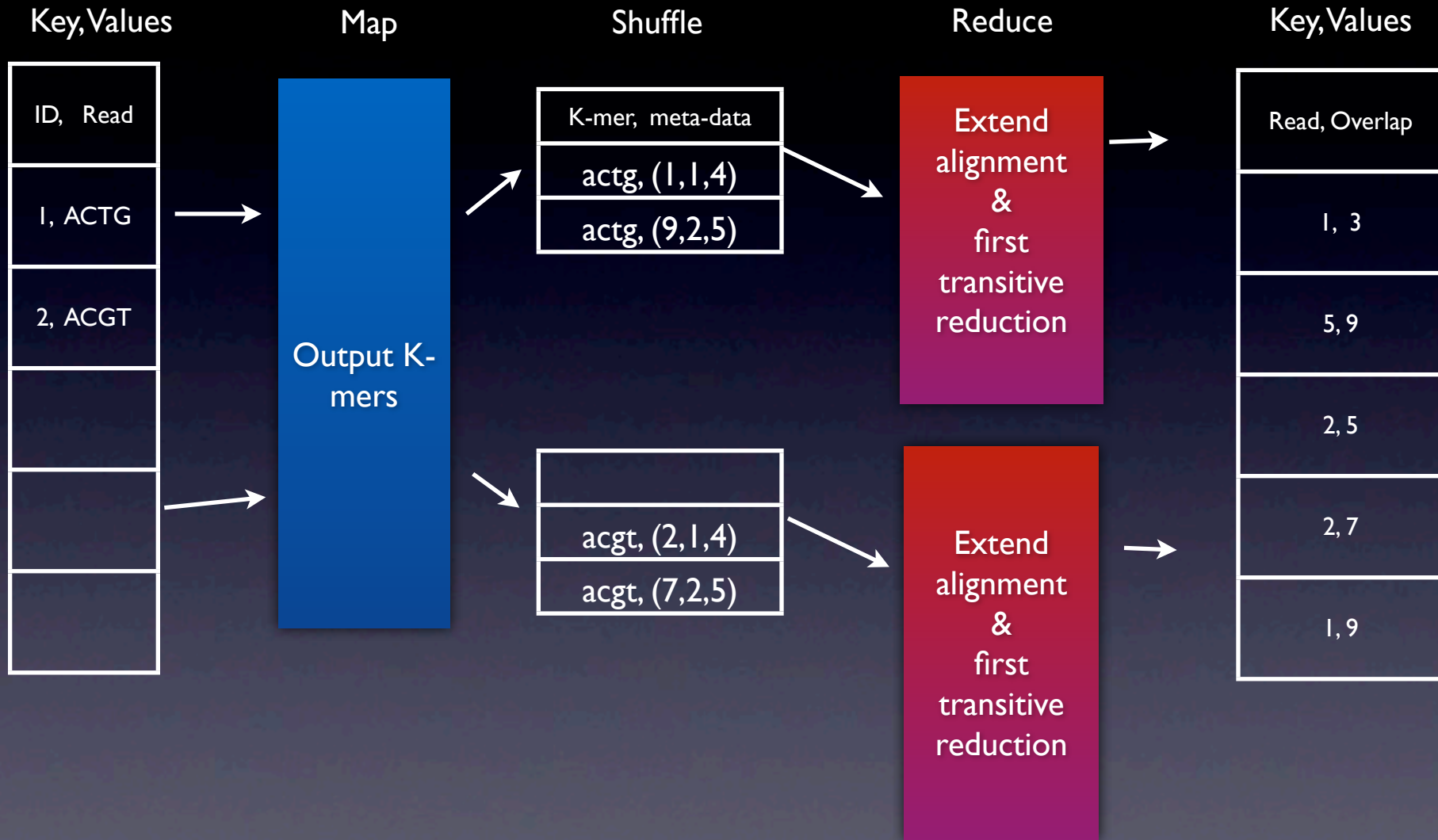
MapReduce Hash-Overlapper



MapReduce Hash-Overlapper

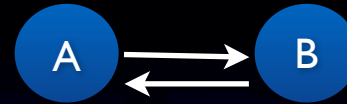


MapReduce Hash-Overlapper

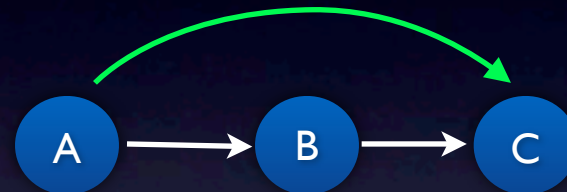


Overlap Graph Reduction Stages

- Remove contained reads



- Remove transitive edges



- Compress paths in the graph



Graphs and MapReduce

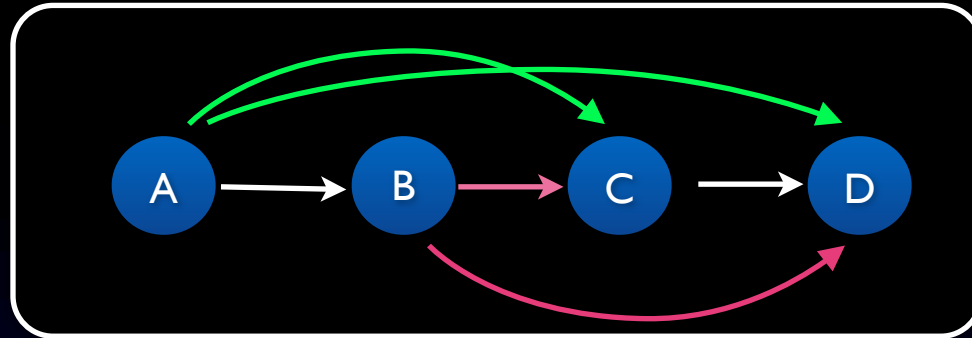
- How do we represent the overlap graph when using MapReduce?
- Large object oriented graph data structures do not work well in MapReduce
- Each Mapper and Reducer only has access to local copy of key, value data and do not have access to the entire graph data structure

Graphs and MapReduce

- Solution: Represent overlap graphs with node adjacency list
- Sort adjacency list by overlap size to effectively do transitive reduction step

Transitive Reduction

Graph $G =$



- Sorted Adjacency lists for graph G
 - A - B, C, D
 - B - C, D
- Compare lists and remove nodes from node A's list that are in node B's list
 - A - B
 - B - C, D

Transitive Reduction

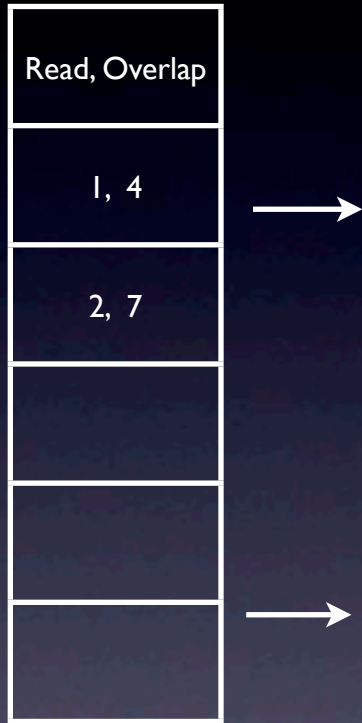
Step 1 : Sort adjacency lists

Transitive Reduction

Step I : Sort adjacency lists

Key, Values

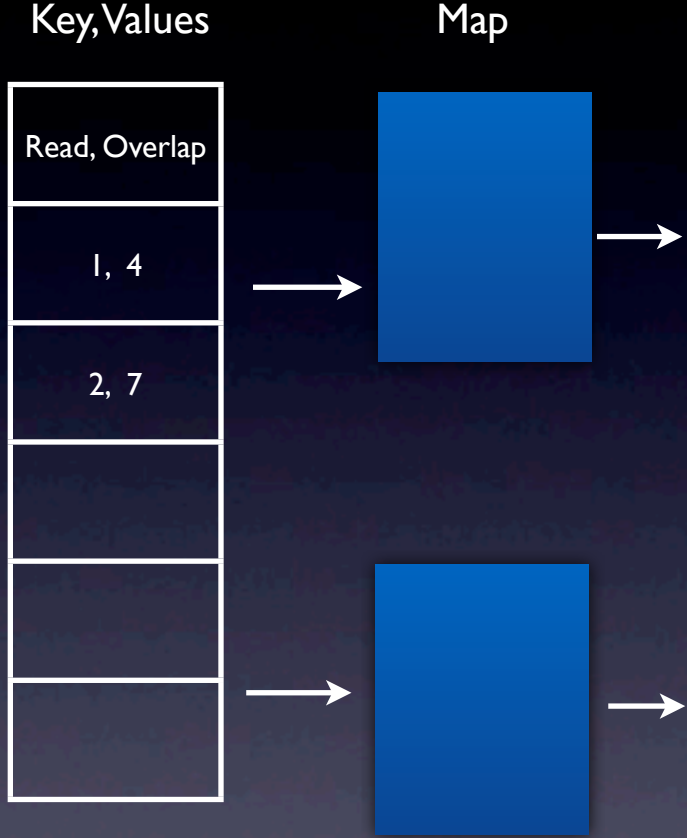
Read, Overlap
1, 4
2, 7



The diagram shows a vertical list of six rectangular boxes. The top box contains the text 'Read, Overlap'. The second box contains '1, 4'. The third box contains '2, 7'. The fourth, fifth, and sixth boxes are empty. A white arrow points from the right side of the second box to the right. Another white arrow points from the right side of the sixth box to the right.

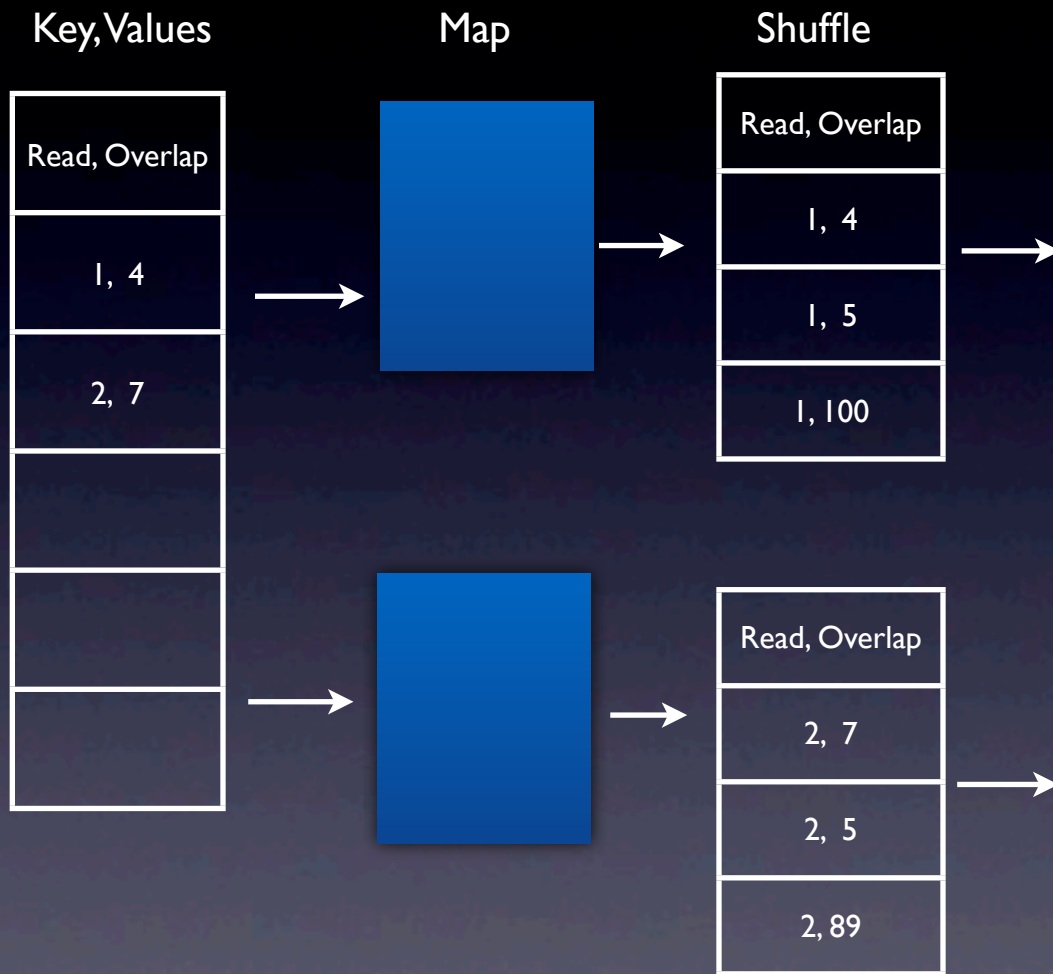
Transitive Reduction

Step I : Sort adjacency lists



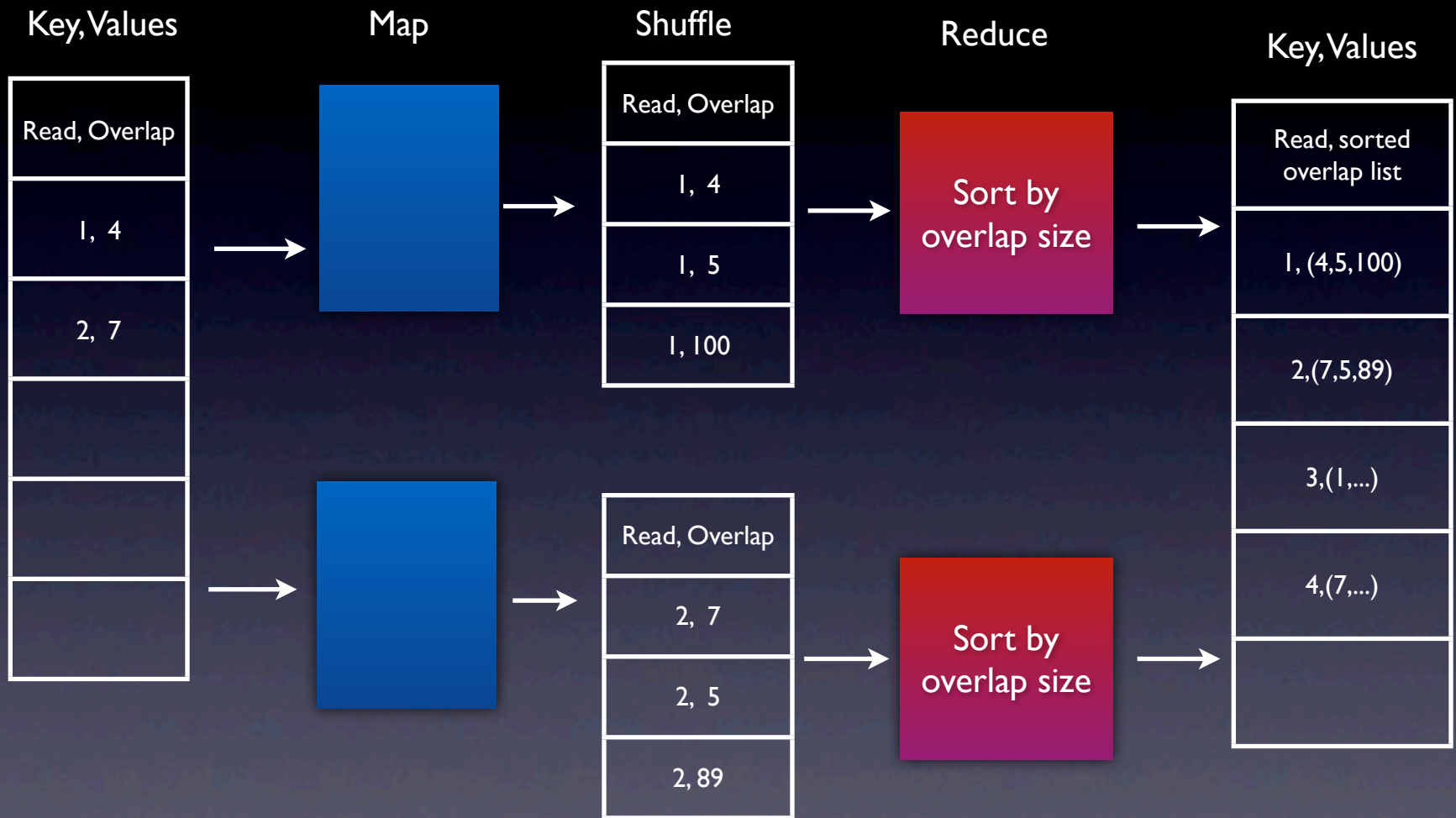
Transitive Reduction

Step I : Sort adjacency lists



Transitive Reduction

Step I : Sort adjacency lists



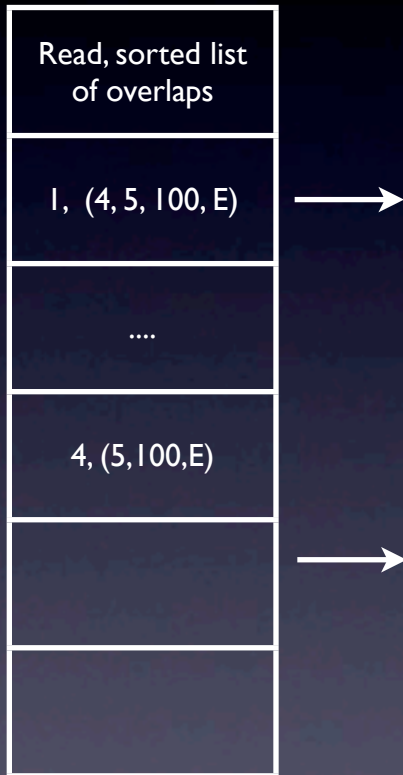
Transitive Reduction

Step 2: Compare lists

Transitive Reduction

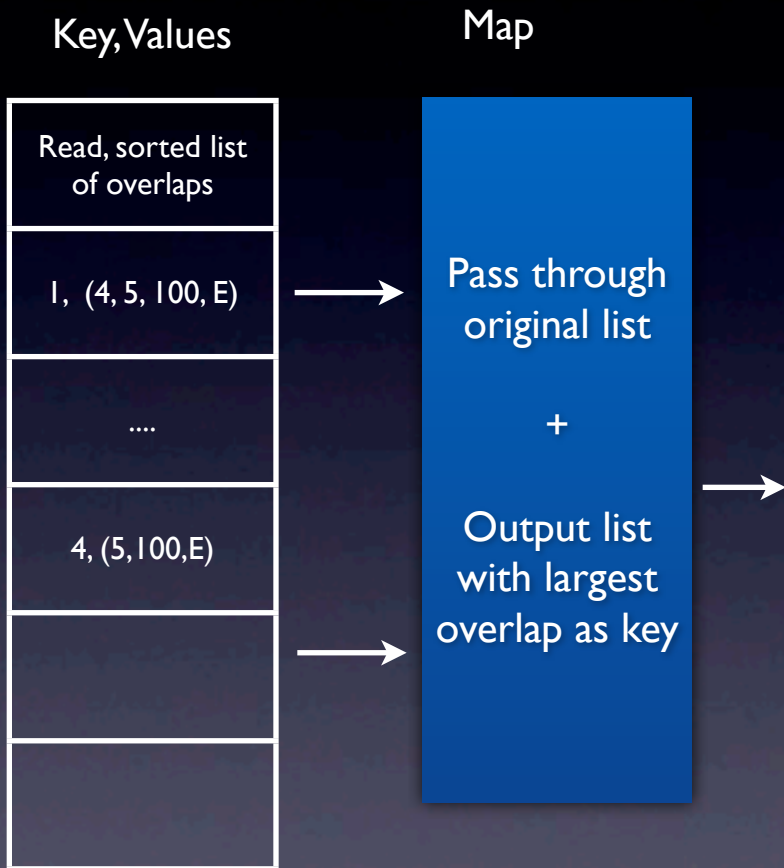
Step 2: Compare lists

Key, Values



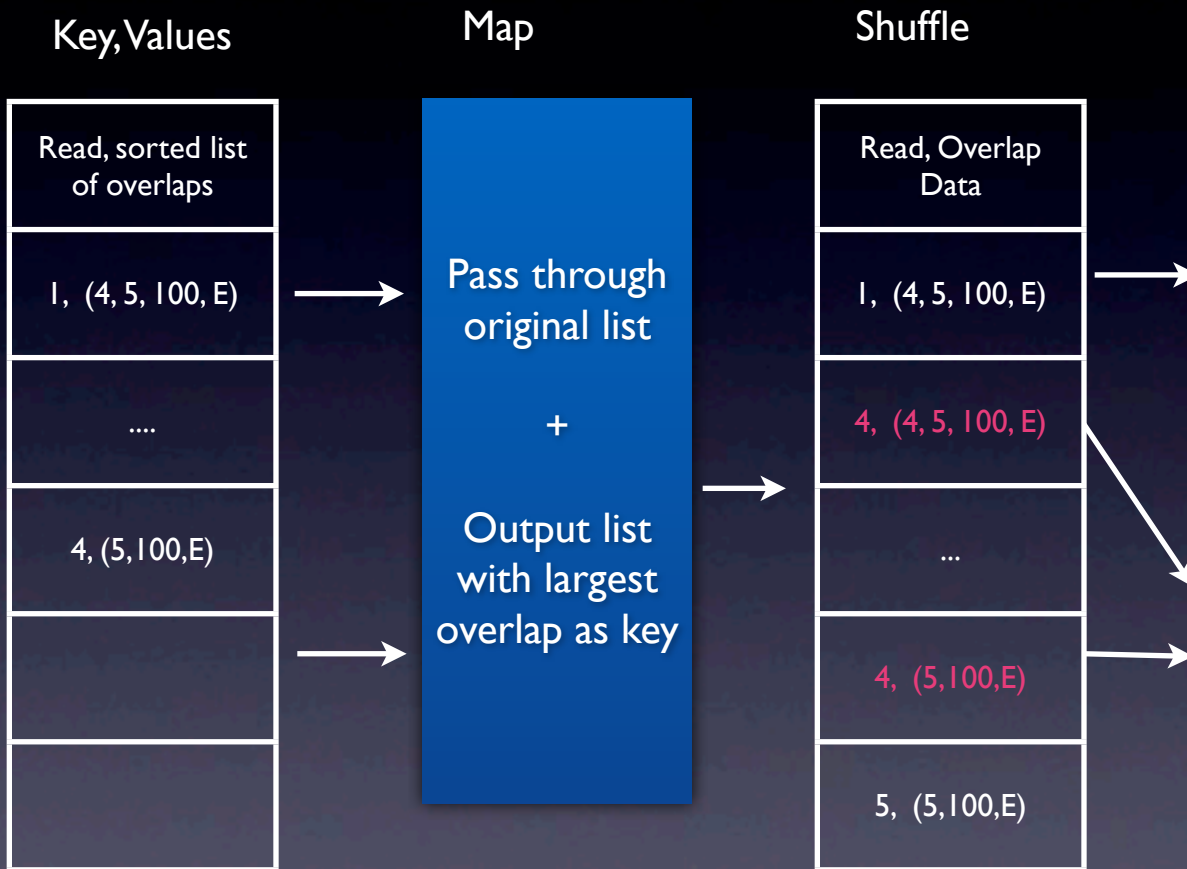
Transitive Reduction

Step 2: Compare lists



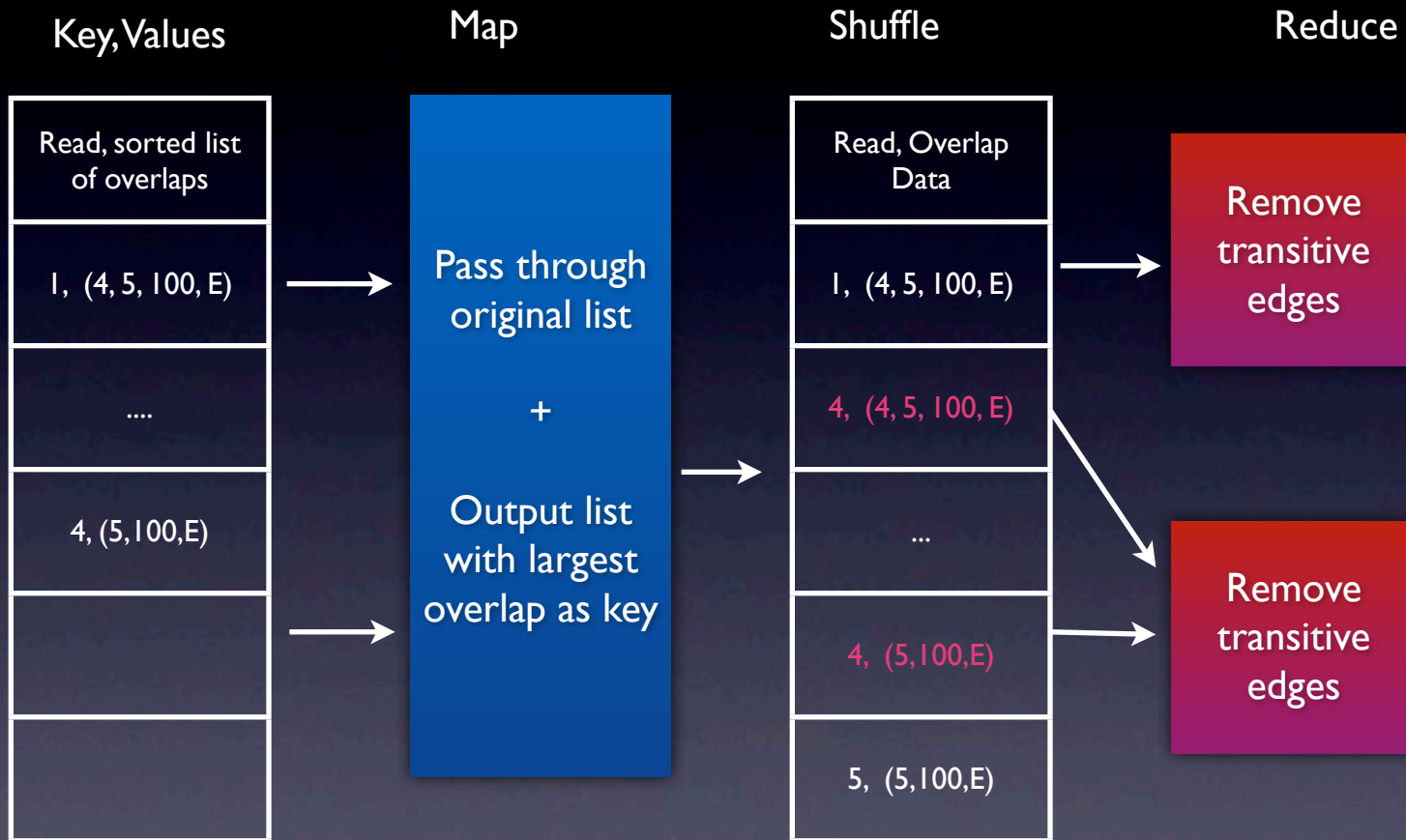
Transitive Reduction

Step 2: Compare lists



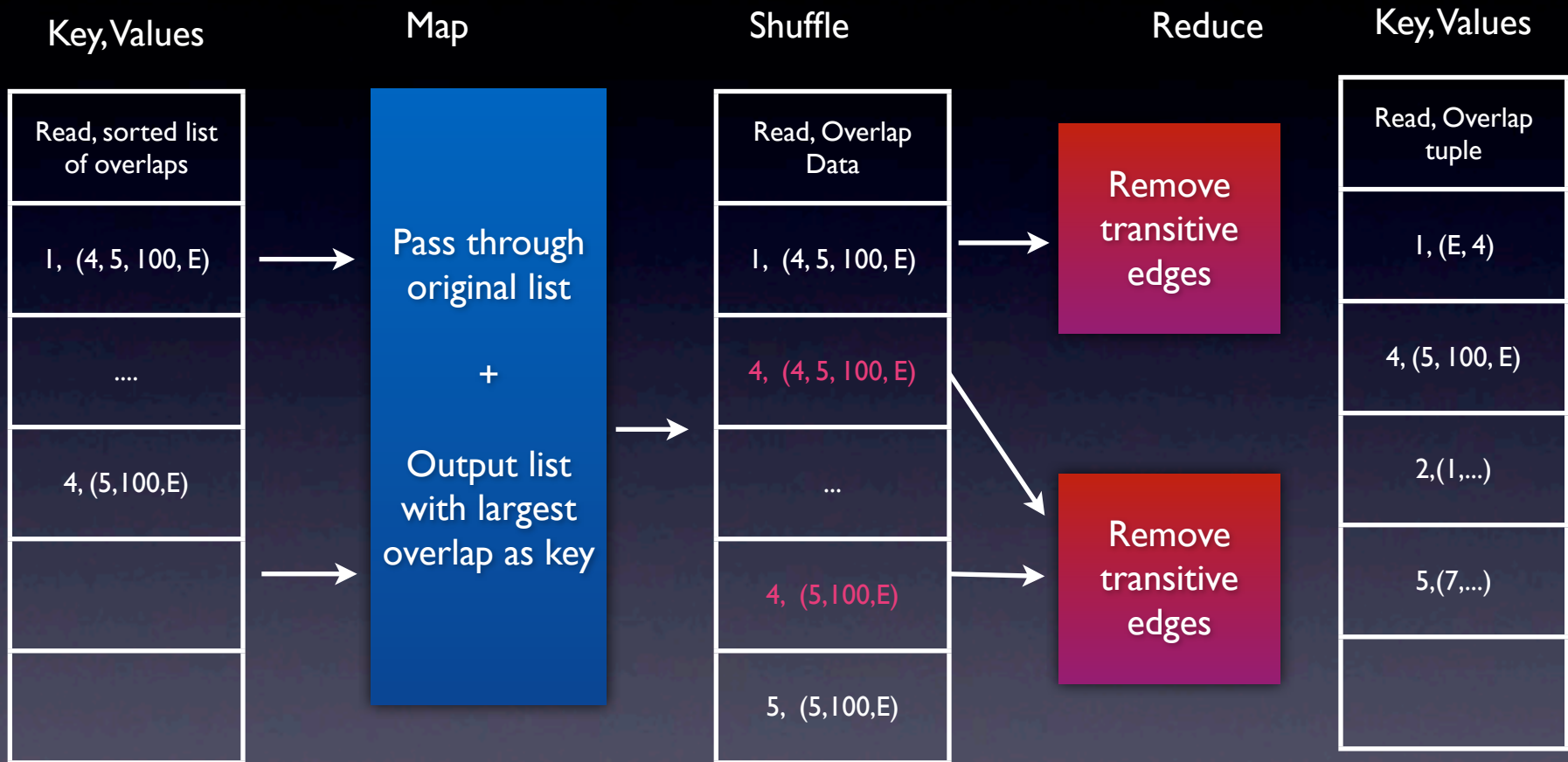
Transitive Reduction

Step 2: Compare lists

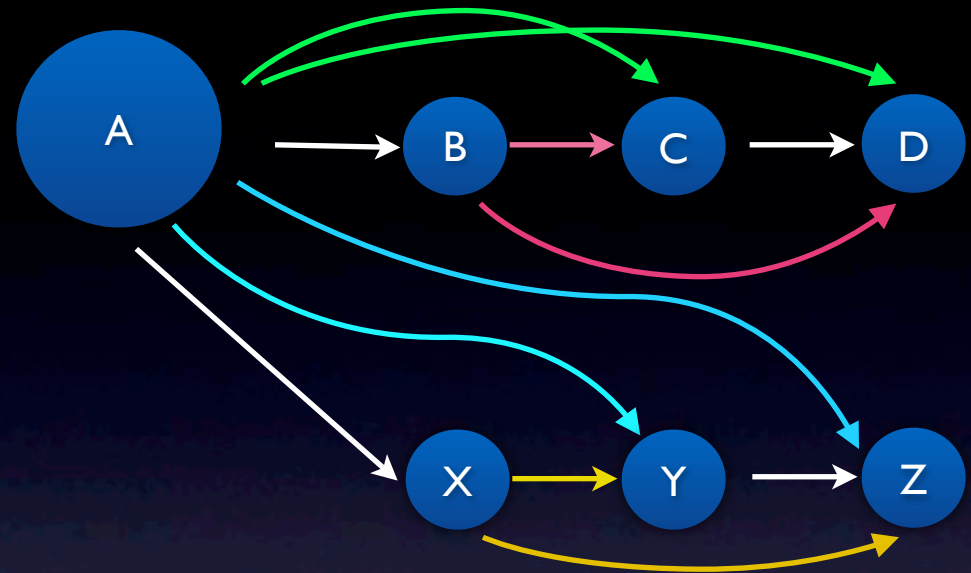


Transitive Reduction

Step 2: Compare lists



Transitive Reduction



- Each time through step 2 one irreducible edge is found
- Move irreducible edge to end of the adjacency list
- Loop through step 2 until end of lists are reached to remove all transitive edges

Summary



Summary

“NextGen sequencing has completely outrun the ability of good bioinformatics people to keep up with the data and use it well... We need a MASSIVE effort in the development of tools for ‘normal’ biologists to make better use of massive sequence databases.”

Jonathan Eisen – JGI Users Meeting –

3/28/09



Summary

“NextGen sequencing has completely outrun the ability of good bioinformatics people to keep up with the data and use it well... We need a MASSIVE effort in the development of tools for ‘normal’ biologists to make better use of massive sequence databases.”

Jonathan Eisen – JGI Users Meeting –

3/28/09

- Computational Biology
 - Make the problems of genotyping and assembly of large genomes from short reads feasible and accessible to individual researchers



Summary

“NextGen sequencing has completely outrun the ability of good bioinformatics people to keep up with the data and use it well... We need a MASSIVE effort in the development of tools for ‘normal’ biologists to make better use of massive sequence databases.”

Jonathan Eisen – JGI Users Meeting –

3/28/09

- Computational Biology
 - Make the problems of genotyping and assembly of large genomes from short reads feasible and accessible to individual researchers
- High Performance Computing
 - Developed Novel Parallel Algorithms for MapReduce and Multicore systems



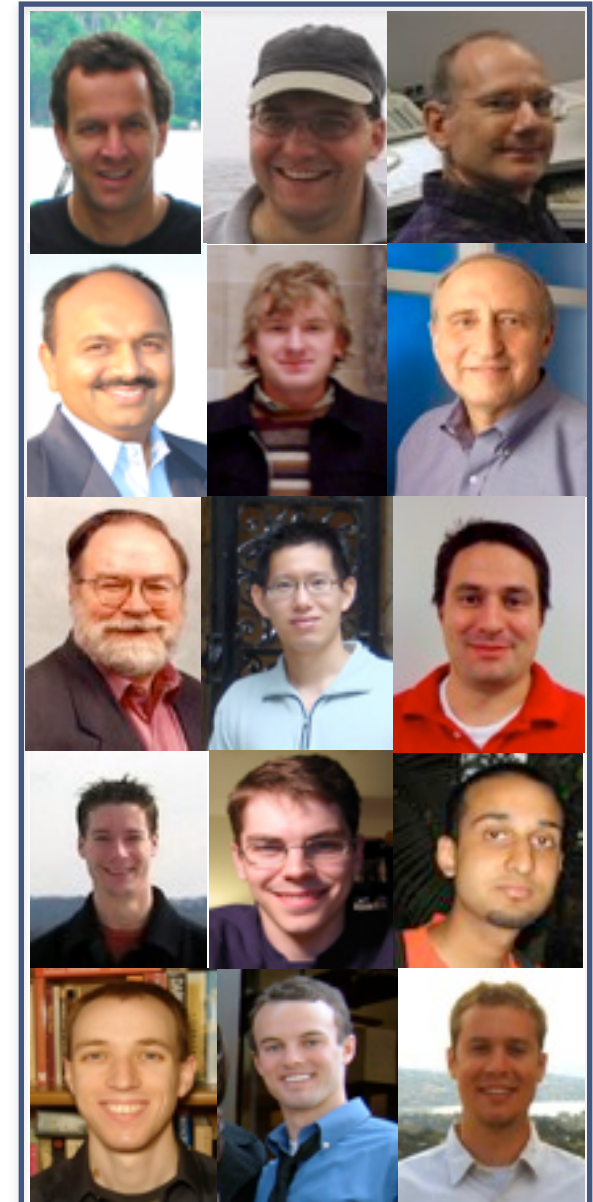
Acknowledgements

UMD Faculty

Steven Salzberg, Mihai Pop, Art Delcher,
Amitabh Varshney, Carl Kingsford, Ben
Shneiderman,
James Yorke, Jimmy Lin,

CBCB Students

Mike Schatz, Adam Phillippy, Cole Trapnell,
Saket Navlakha, Ben Langmead,
James White, David Kelley



Thank You!