Cloud Computing and the DNA Data Race

Michael Schatz, Ph.D.

June 16, 2010
Mayo Clinic Genomics Interest Group

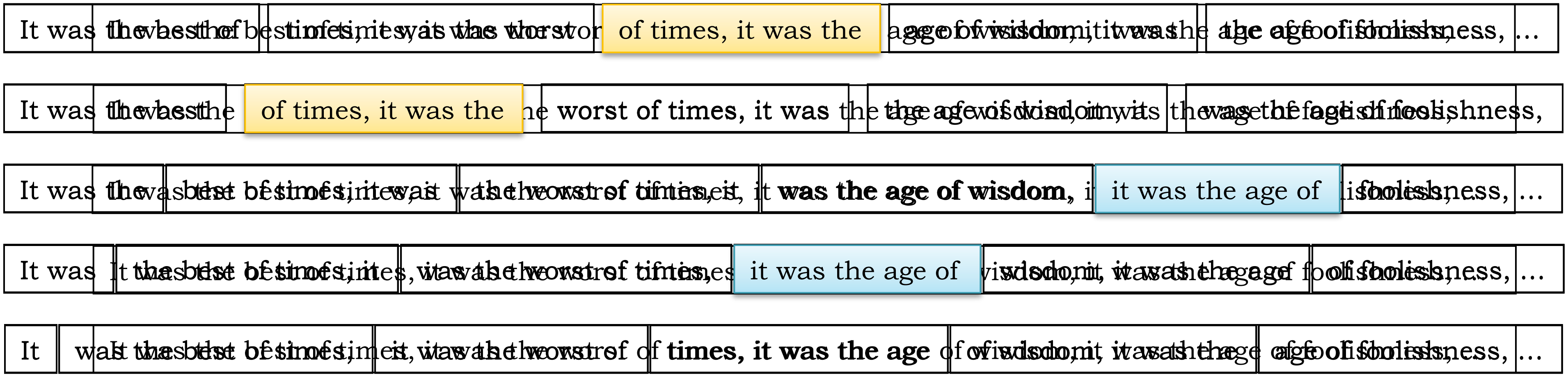
UNIVERSITY OF MARYLAND
18 56

Outline

1. Genome Assembly by Analogy
2. DNA Sequencing and Genomics
3. Sequence Analysis in the Clouds
    1. Sequence Alignment
    2. Mapping & Genotyping
    3. Genome Assembly

Shredded Book Reconstruction

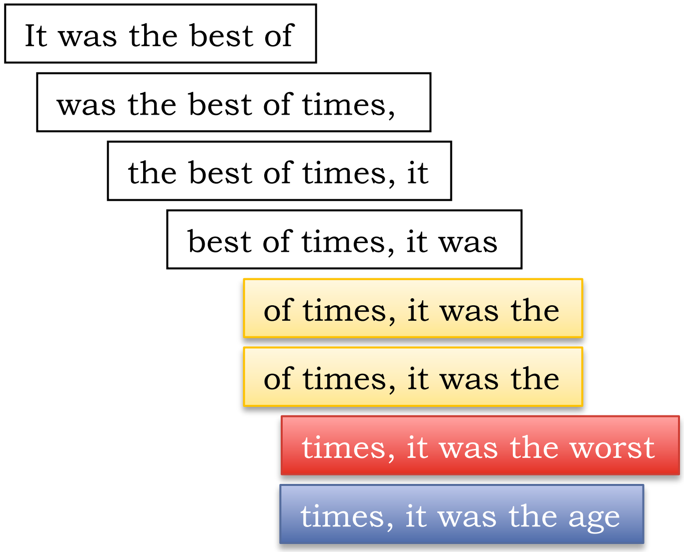
- Dickens accidentally shreds the first printing of <u>A Tale of Two Cities</u>
  - Text printed on 5 long spools



- 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

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

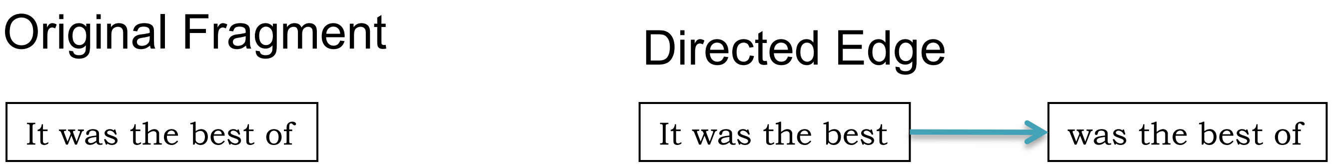
The repeated sequence make the correct reconstruction ambiguous

- 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

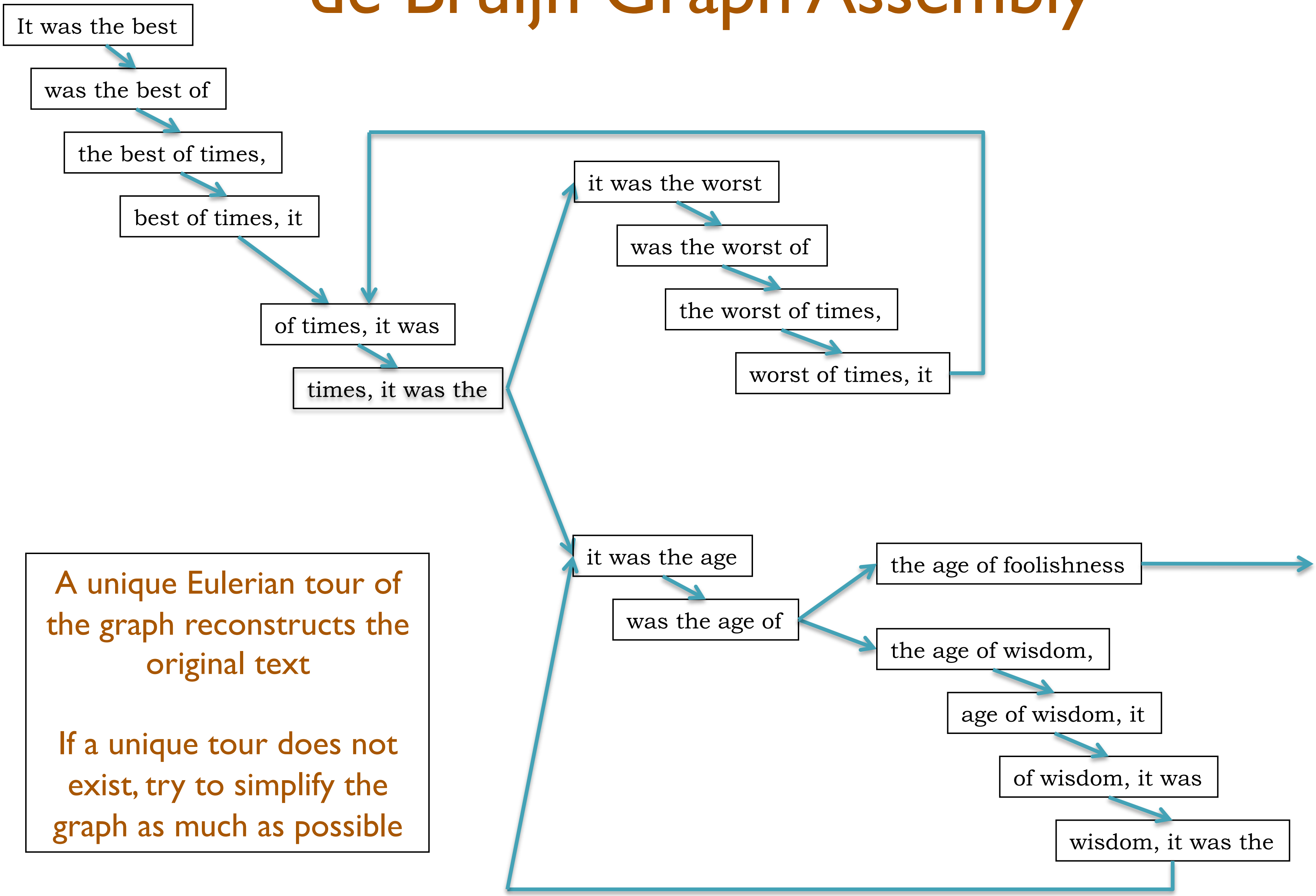
It was the best

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



It was the best

was the best of

the best of times,

best of times, it

of times, it was

times, it was the

it was the worst

was the worst of

the worst of times,

worst of times, it

it was the age

was the age of

the age of foolishness

the age of wisdom,

age of wisdom, it

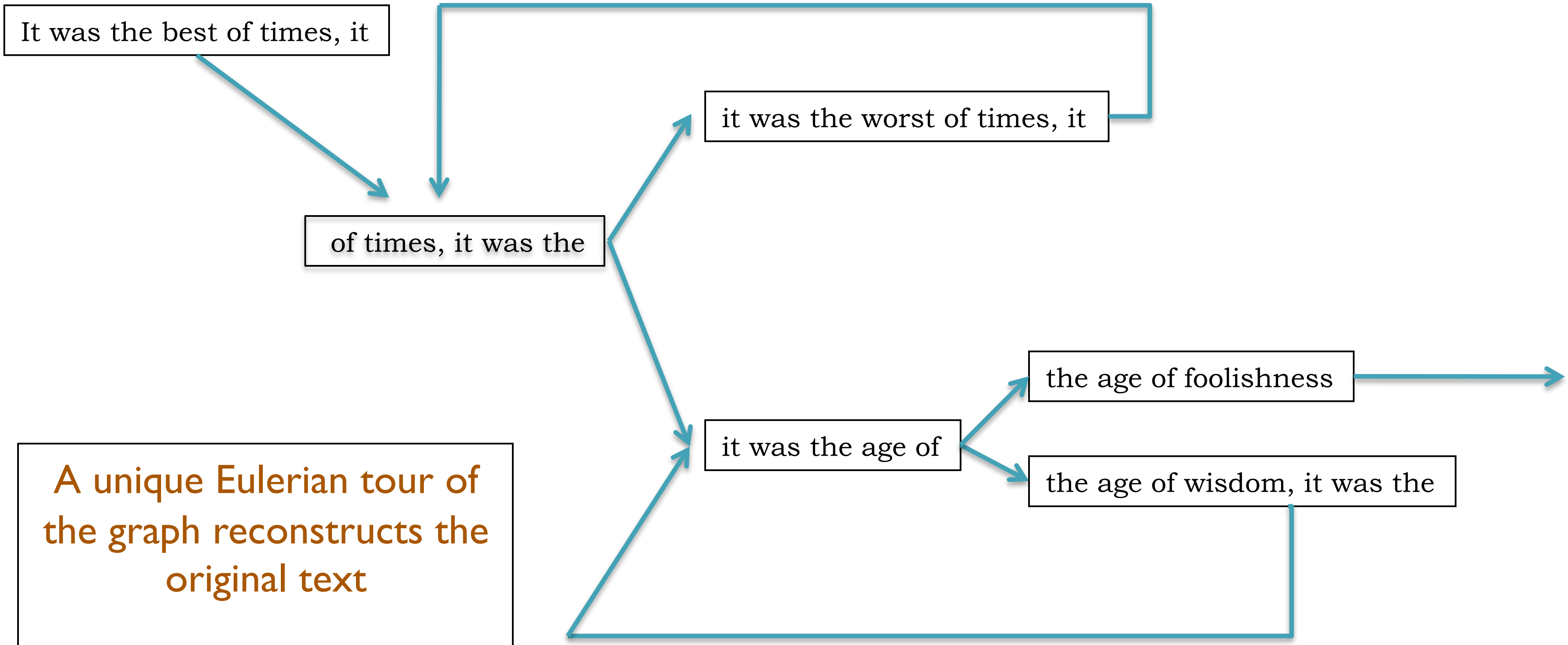
of wisdom, it was

wisdom, it was the

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

de Bruijn Graph Assembly



It was the best of times, it

of times, it was the

it was the worst of times, it

it was the age of

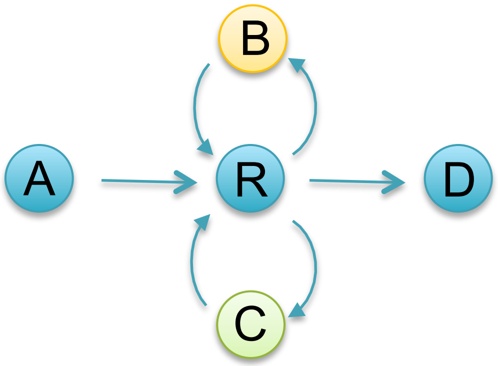
the age of foolishness

the age of wisdom, it was the

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



ARBRCRD
or
ARCRBRD

Generally an exponential number of compatible sequences

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

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

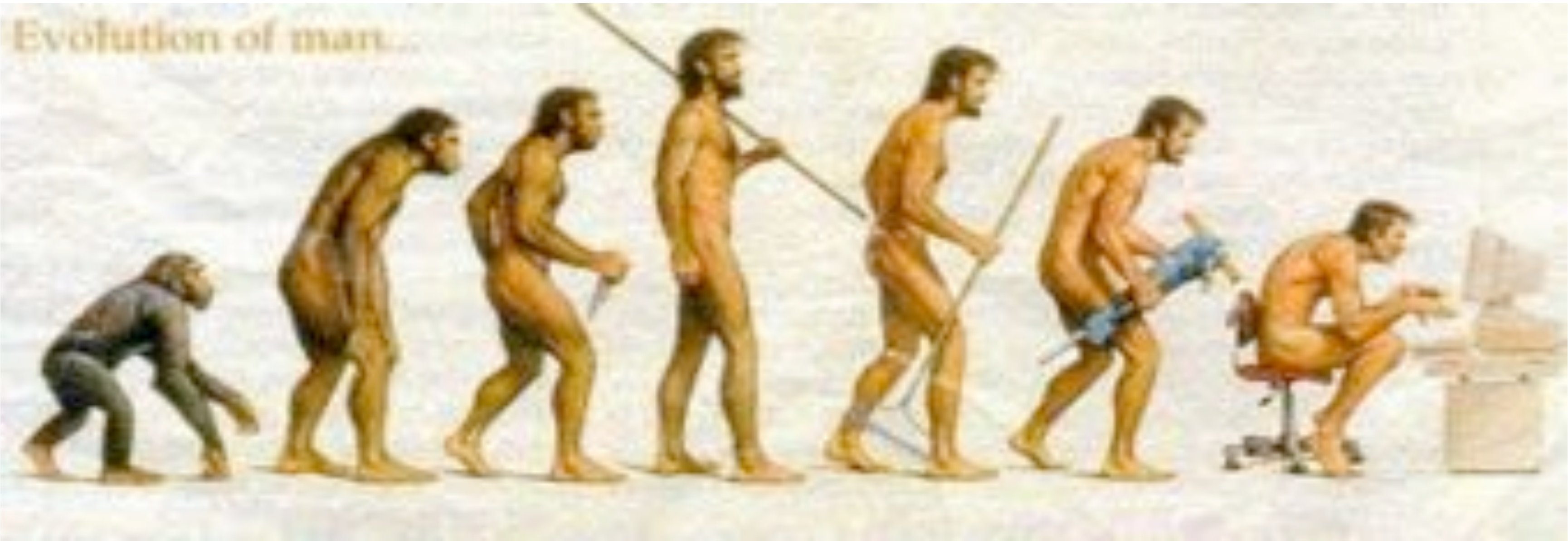
L = $n \ x \ n$ matrix with r_u-a_{uu} along the diagonal and -a_{uv} in entry uv

$r_u = d^+(u)+1$ 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 and Evolution



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
- Genome as a recipe, not a blueprint

Like Dickens, we can only sequence small fragments of the genome

The DNA Data Race

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

(Pushkarev *et al.*, 2009)

GENOME 10K

ENCODE

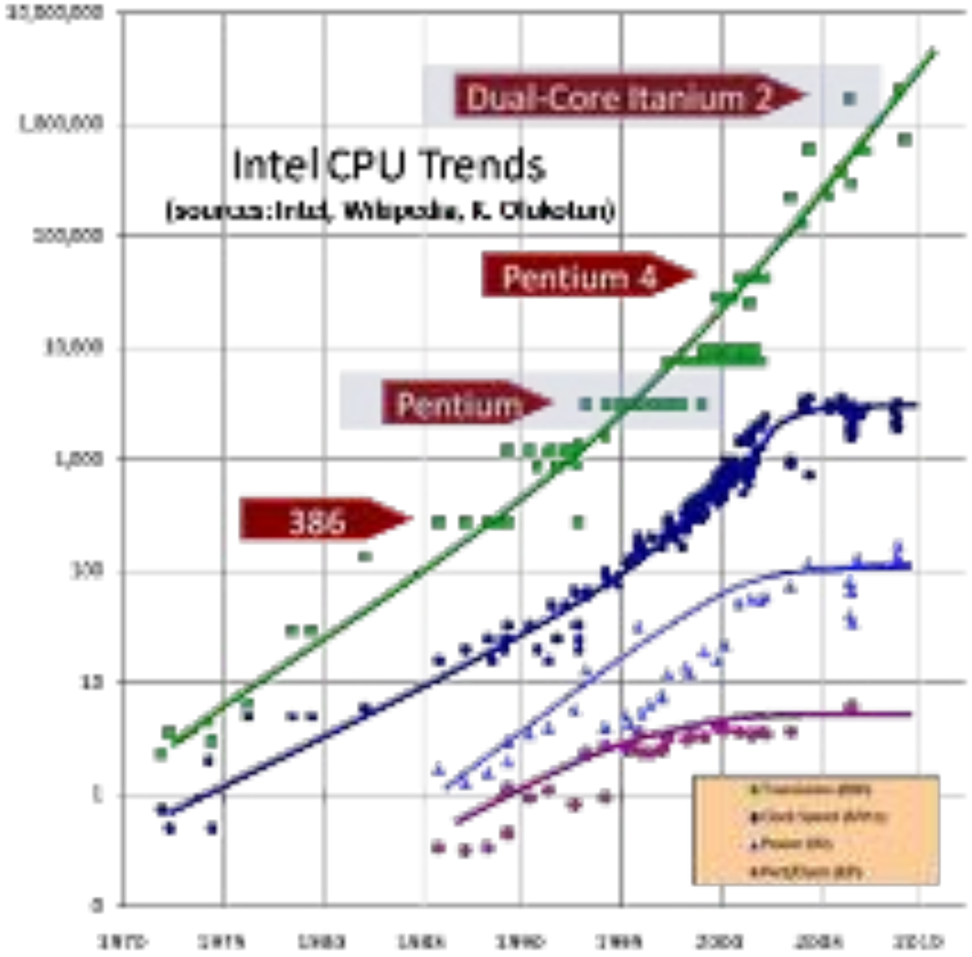
HMP

Critical Computational Challenges: Alignment and Assembly of Huge Datasets

HPC for Computational Biology

- Assembly and Genotyping require 1000s of CPU hours, 100s of GB of data per genome
  - Many cores
  - Many computers
  - Many disks

- Why HPC?
  - Computer speed is flat
  - Need results faster
  - Doesn't fit on one machine



The Free Lunch Is Over: A Fundamental Turn Toward Concurrency in Software
Herb Sutter, http://www.gotw.ca/publications/concurrency-ddj.htm

A Brief History of the Amazon Cloud

- Urban Legend
  - Additional capacity added every fall for the holiday shopping season, underutilized rest of the year…
- Official Story
  - Amazon is a technology company
    - Different divisions of Amazon share computation
  - Amazon Web Services is the 3rd Business Division
    - Retail & Seller Businesses

Amazon Web Services

http://aws.amazon.com

- All you need is a credit card, and you can immediately start using one of the largest datacenters in the world

- Elastic Compute Cloud (EC2)
  - On demand computing power
    - Support for Windows, Linux, & OpenSolaris
    - Starting at 8.5¢ / core / hour

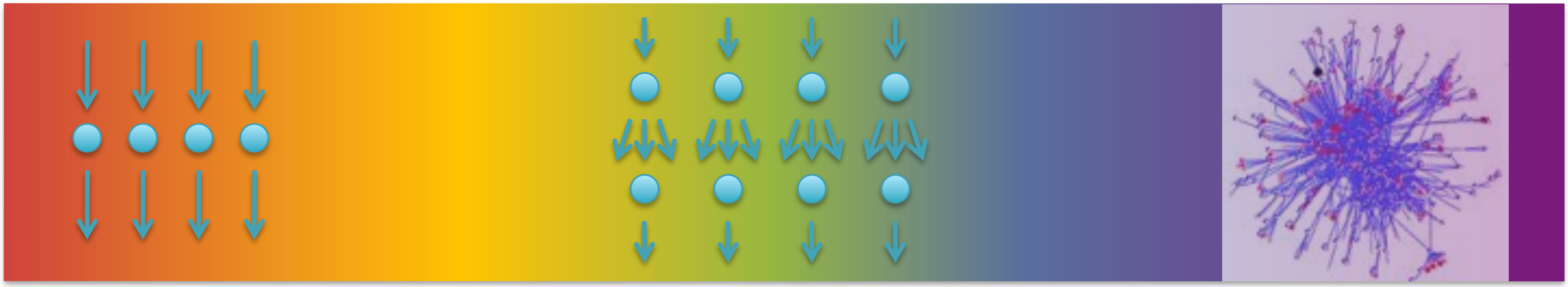
- Simple Storage Service (S3)
  - Scalable data storage
    - 10¢ / GB upload fee, 15¢ / GB monthly fee

Cloud Computing Spectrum

Embarrassingly
Parallel

Loosely
Coupled

Tightly
Coupled



Batch Computing

MapReduce/DryadLINQ

MPI/MapReduce/Pregel

Alignment
HMM Scoring

Genotyping
K-mer Counting

Graph Analysis
Genome Assembly

Scheduling +
Load Balance

Embarrassingly Parallel +
Parallel Communication

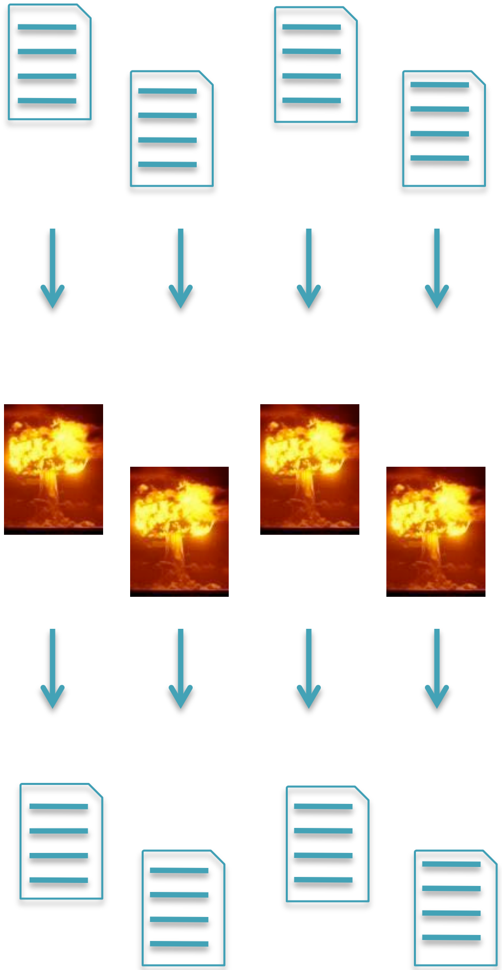
Loosely Coupled +
Parallel Algorithm Design

Embarrassingly Parallel

- Batch computing
  - Each item is independent
  - Split input into many chunks
  - Process each chunk separately on a different computer

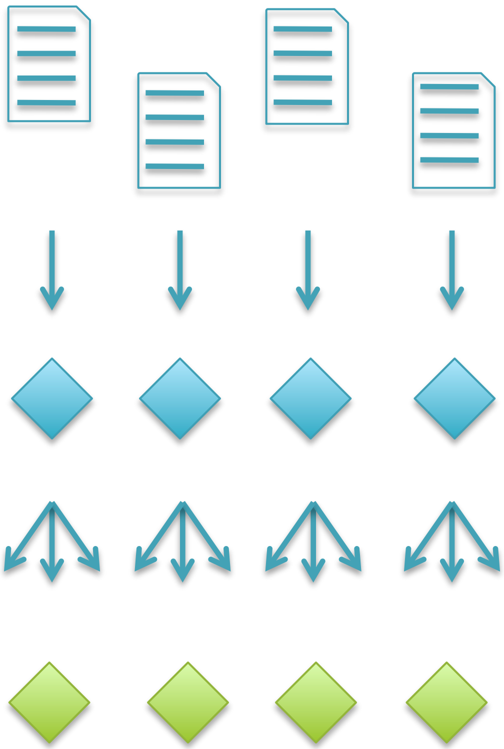
- Challenges
  - Distributing work, load balancing, monitoring & restart

- Technologies
  - Condor, Sun Grid Engine
  - Amazon Simple Queue



Elementary 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, Elastic MapReduce, Dryad
  - Parallel Databases

Junior High Dance

Hadoop MapReduce

http://hadoop.apache.org

- 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

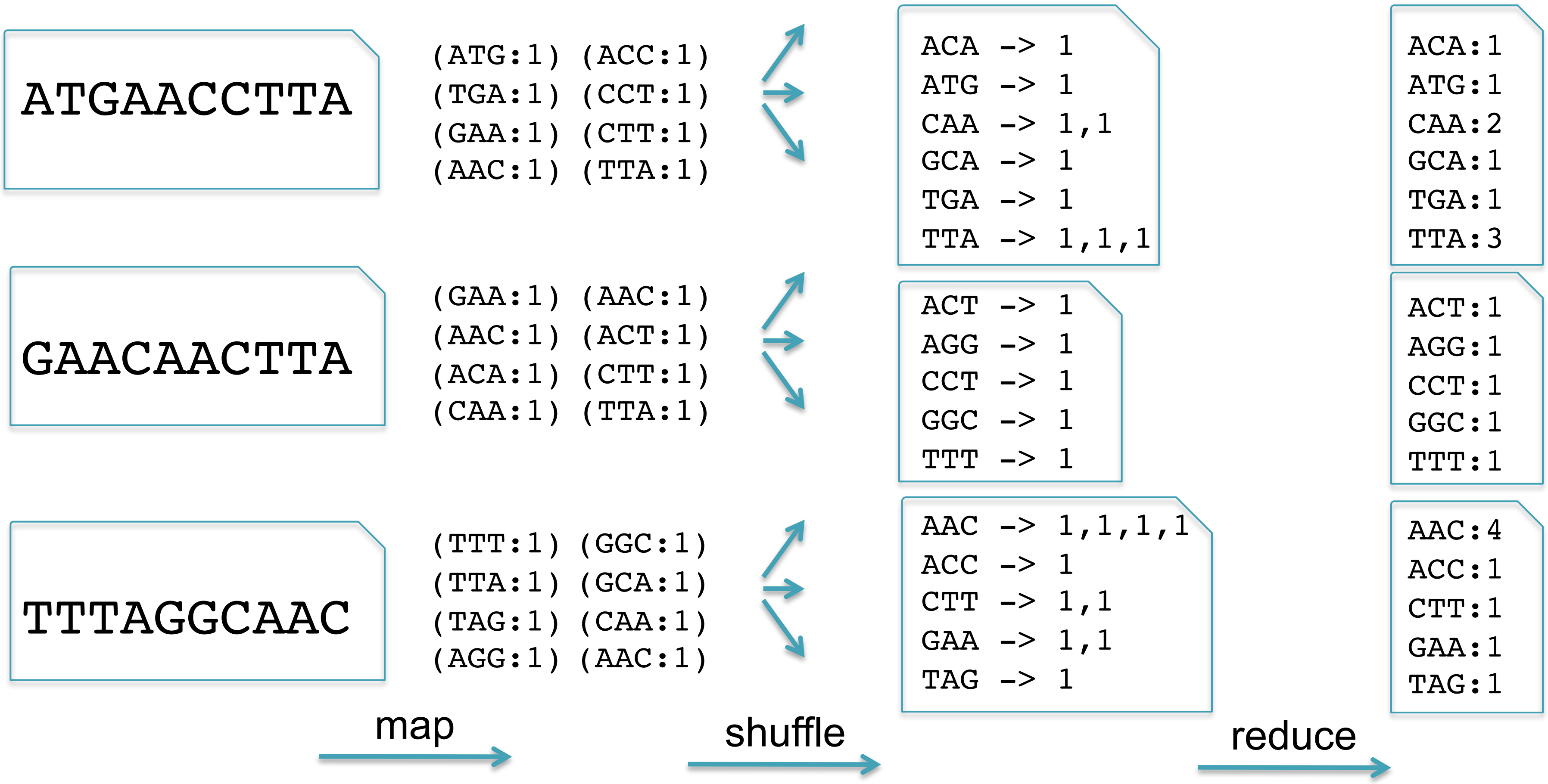
- Benefits
  - Scalable, Efficient, Reliable
  - Easy to Program
  - Runs on commodity computers

- Challenges
  - Redesigning / Retooling applications
    - Not Condor, Not MPI
    - Everything in MapReduce

K-mer Counting

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

Map, Shuffle & Reduce
All Run in Parallel



ATGAACCTTA

GAACAACTTA

TTTAGGCAAC

(ATG:1) (ACC:1)
(TGA:1) (CCT:1)
(GAA:1) (CTT:1)
(AAC:1) (TTA:1)

(GAA:1) (AAC:1)
(AAC:1) (ACT:1)
(ACA:1) (CTT:1)
(CAA:1) (TTA:1)

(TTT:1) (GGC:1)
(TTA:1) (GCA:1)
(TAG:1) (CAA:1)
(AGG:1) (AAC:1)

ACA -> 1
ATG -> 1
CAA -> 1,1
GCA -> 1
TGA -> 1
TTA -> 1,1,1

ACT -> 1
AGG -> 1
CCT -> 1
GGC -> 1
TTT -> 1

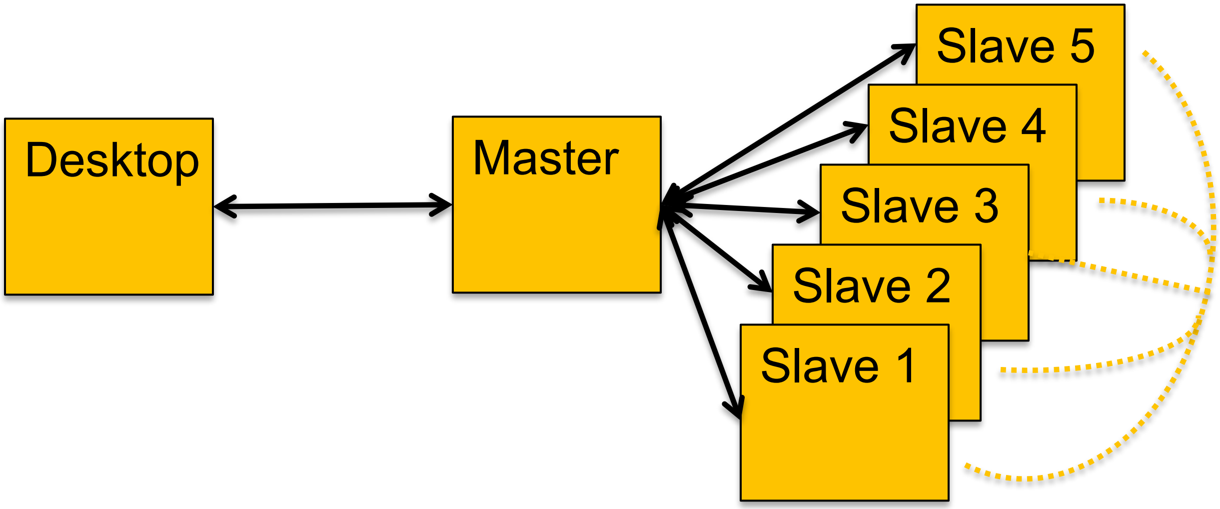
AAC -> 1,1,1,1
ACC -> 1
CTT -> 1,1
GAA -> 1,1
TAG -> 1

ACA:1
ATG:1
CAA:2
GCA:1
TGA:1
TTA:3

ACT:1
AGG:1
CCT:1
GGC:1
TTT:1

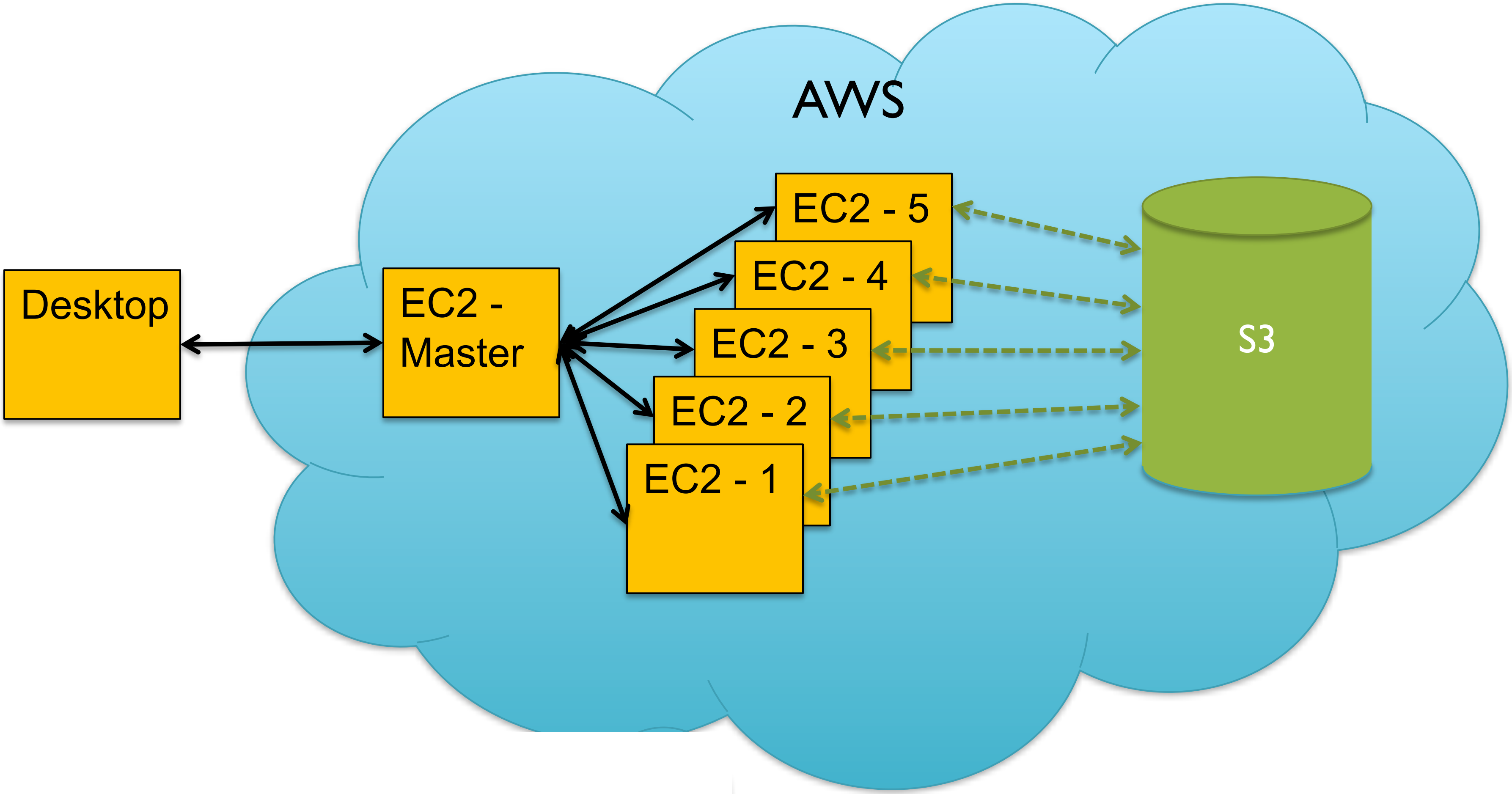
AAC:4
ACC:1
CTT:1
GAA:1
TAG:1

Hadoop Architecture



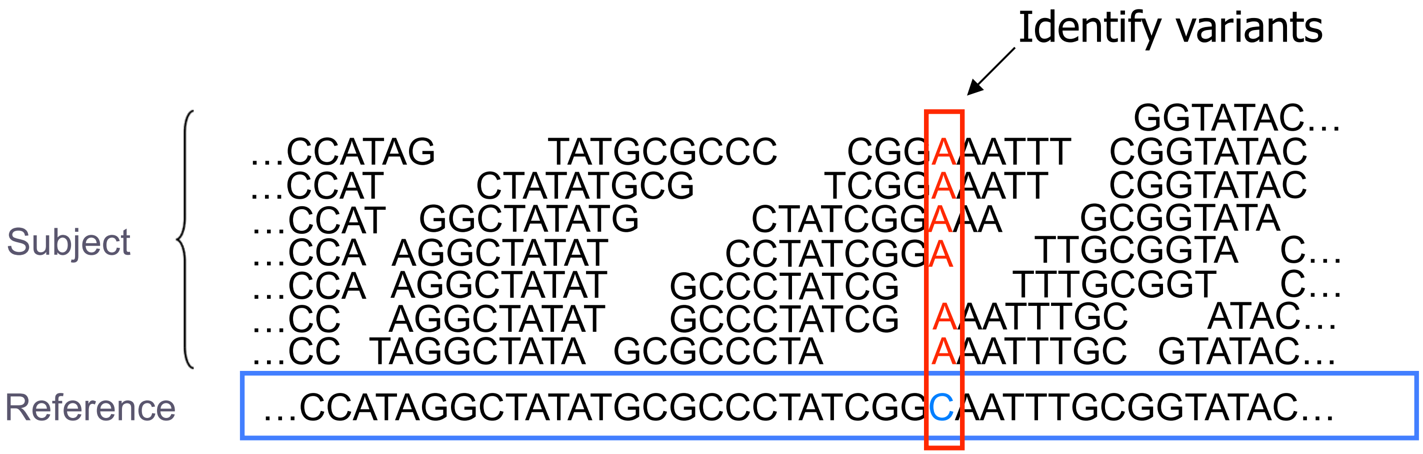
- Hadoop Distributed File System (HDFS)
  - Data files partitioned into large chunks (64MB), replicated on multiple nodes
  - Computation moves to the data, rack-aware scheduling
- Hadoop MapReduce system won the 2009 GreySort Challenge
  - Sorted 100 TB in 173 min (578 GB/min) using 3452 nodes and 4x3452 disks

Hadoop on AWS



- If you don’t have 1000s of machines, rent them from Amazon
  - After machines spool up, ssh to master as if it was a local machine.
  - Use S3 for persistent data storage, with very fast interconnect to EC2.

Short Read Mapping



Identify variants

```
                                                          GGTATAC…
          …CCATAG      TATGCGCCC     CGGAAATTT   CGGTATAC
          …CCAT      CTATATGCG      TCGGAAATT    CGGTATAC
          …CCAT   GGCTATATG      CTATCGGAAA    GCGGTATA
Subject   …CCA  AGGCTATAT      CCTATCGGA     TTGCGGTA   C…
          …CCA  AGGCTATAT    GCCCTATCG      TTTGCGGT    C…
          …CC   AGGCTATAT    GCCCTATCG  AAATTTGC      ATAC…
          …CC  TAGGCTATA  GCGCCCTA      AAATTTGC  GTATAC…

Reference  …CCATAGGCTATATGCGCCCTATCGGCAATTTGCGGTATAC…
```

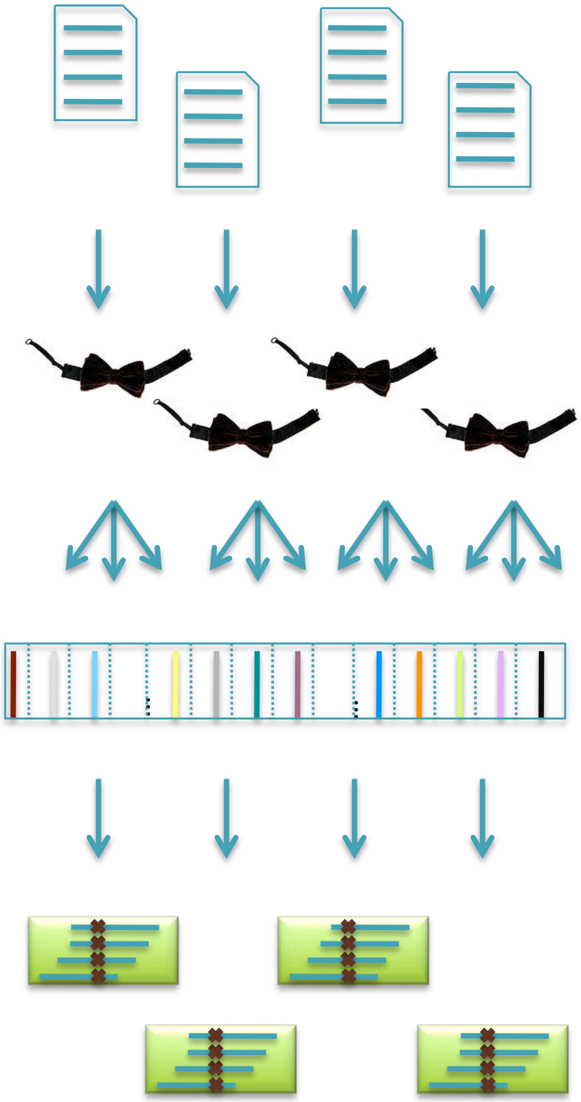
- Given a reference and many subject reads, report one or more "good" end-to-end alignments per alignable read
  - Find where the read most likely originated
  - Fundamental computation for many assays

• 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
- 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

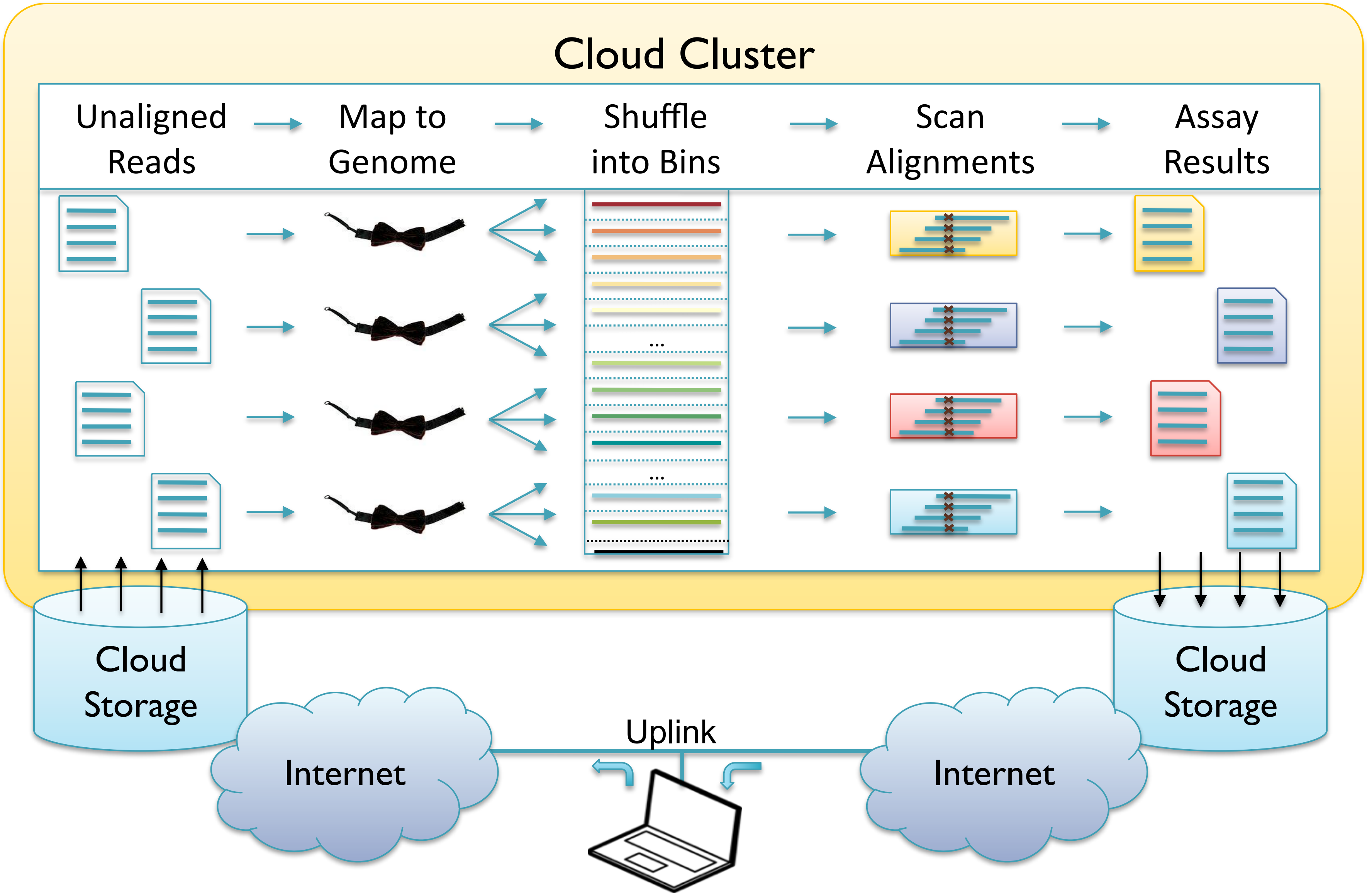
http://bowtie-bio.sourceforge.net/crossbow

	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.* **10**:R134

Map-Shuffle-Scan for Genomics

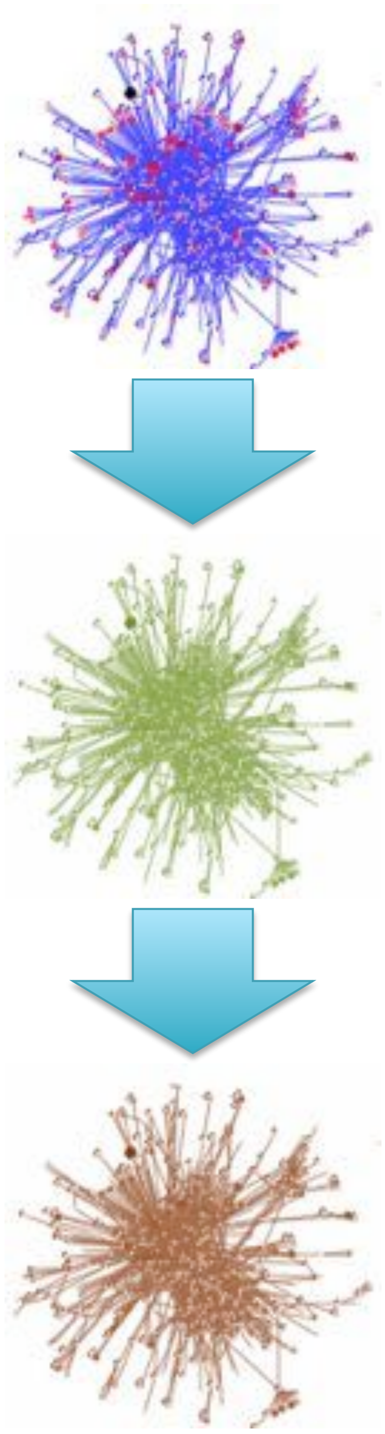


Tightly Coupled

- Computation that cannot be partitioned
  - Graph Analysis
  - Molecular Dynamics
  - Population simulations

- Challenges
  - Loosely coupled challenges
  - + Parallel algorithms design

- Technologies
  - MPI
  - MapReduce, Dryad, Pregel

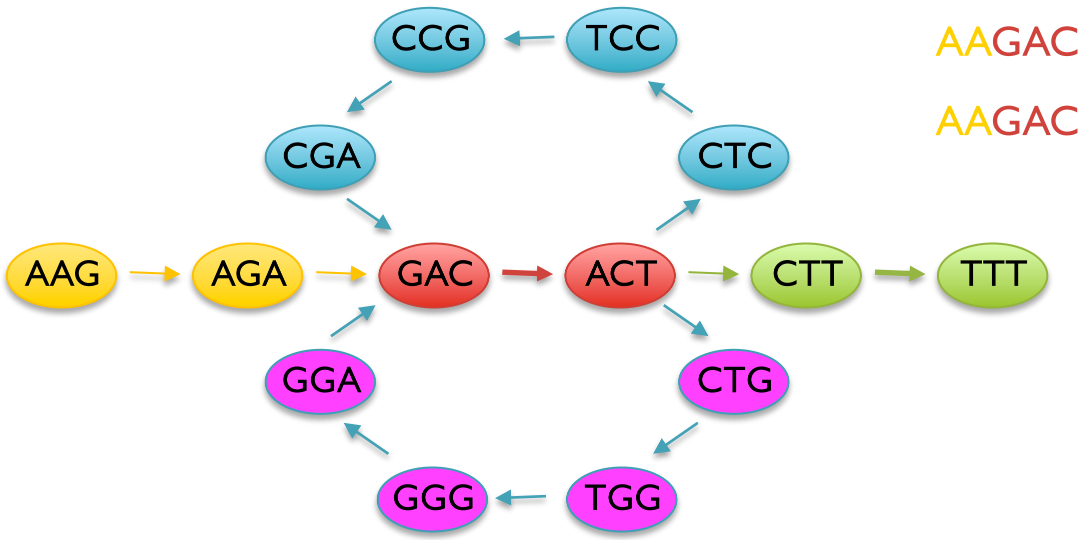


Short Read Assembly

Reads

AAGA
ACTT
ACTC
ACTG
AGAG
CCGA
CGAC
CTCC
CTGG
CTTT
…

de Bruijn Graph



Potential Genomes

AAGACTCCGACTGGGACTTT

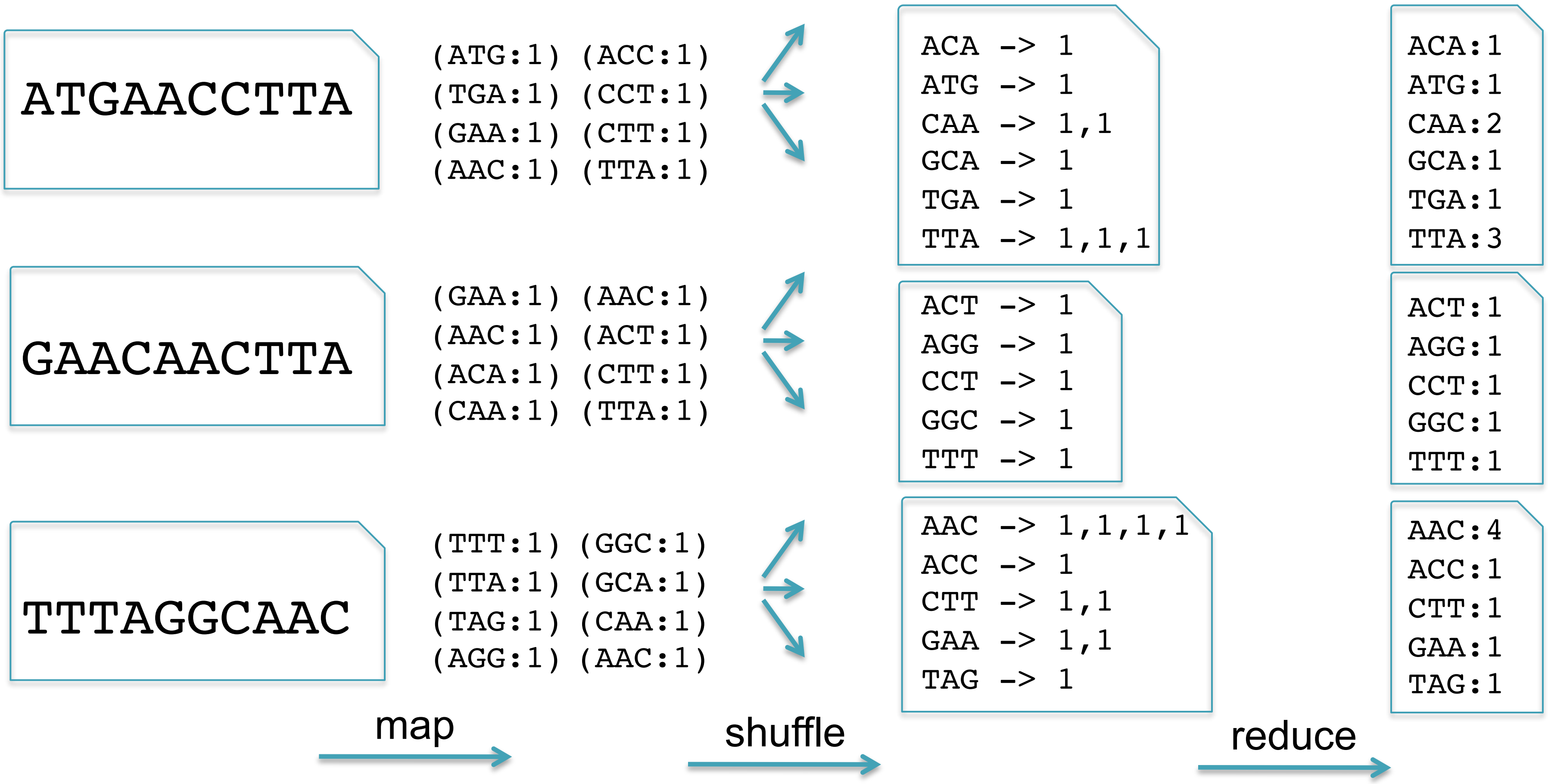
AAGACTGGGACTCCGACTTT

- 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

K-mer Counting

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

Map, Shuffle & Reduce
All Run in Parallel



```
ATGAACCTTA
```

```
GAACAACTTA
```

```
TTTAGGCAAC
```

```
(ATG:1) (ACC:1)
(TGA:1) (CCT:1)
(GAA:1) (CTT:1)
(AAC:1) (TTA:1)
```

```
(GAA:1) (AAC:1)
(AAC:1) (ACT:1)
(ACA:1) (CTT:1)
(CAA:1) (TTA:1)
```

```
(TTT:1) (GGC:1)
(TTA:1) (GCA:1)
(TAG:1) (CAA:1)
(AGG:1) (AAC:1)
```

```
ACA -> 1
ATG -> 1
CAA -> 1,1
GCA -> 1
TGA -> 1
TTA -> 1,1,1
```

```
ACT -> 1
AGG -> 1
CCT -> 1
GGC -> 1
TTT -> 1
```

```
AAC -> 1,1,1,1
ACC -> 1
CTT -> 1,1
GAA -> 1,1
TAG -> 1
```

```
ACA:1
ATG:1
CAA:2
GCA:1
TGA:1
TTA:3
```

```
ACT:1
AGG:1
CCT:1
GGC:1
TTT:1
```

```
AAC:4
ACC:1
CTT:1
GAA:1
TAG:1
```

map

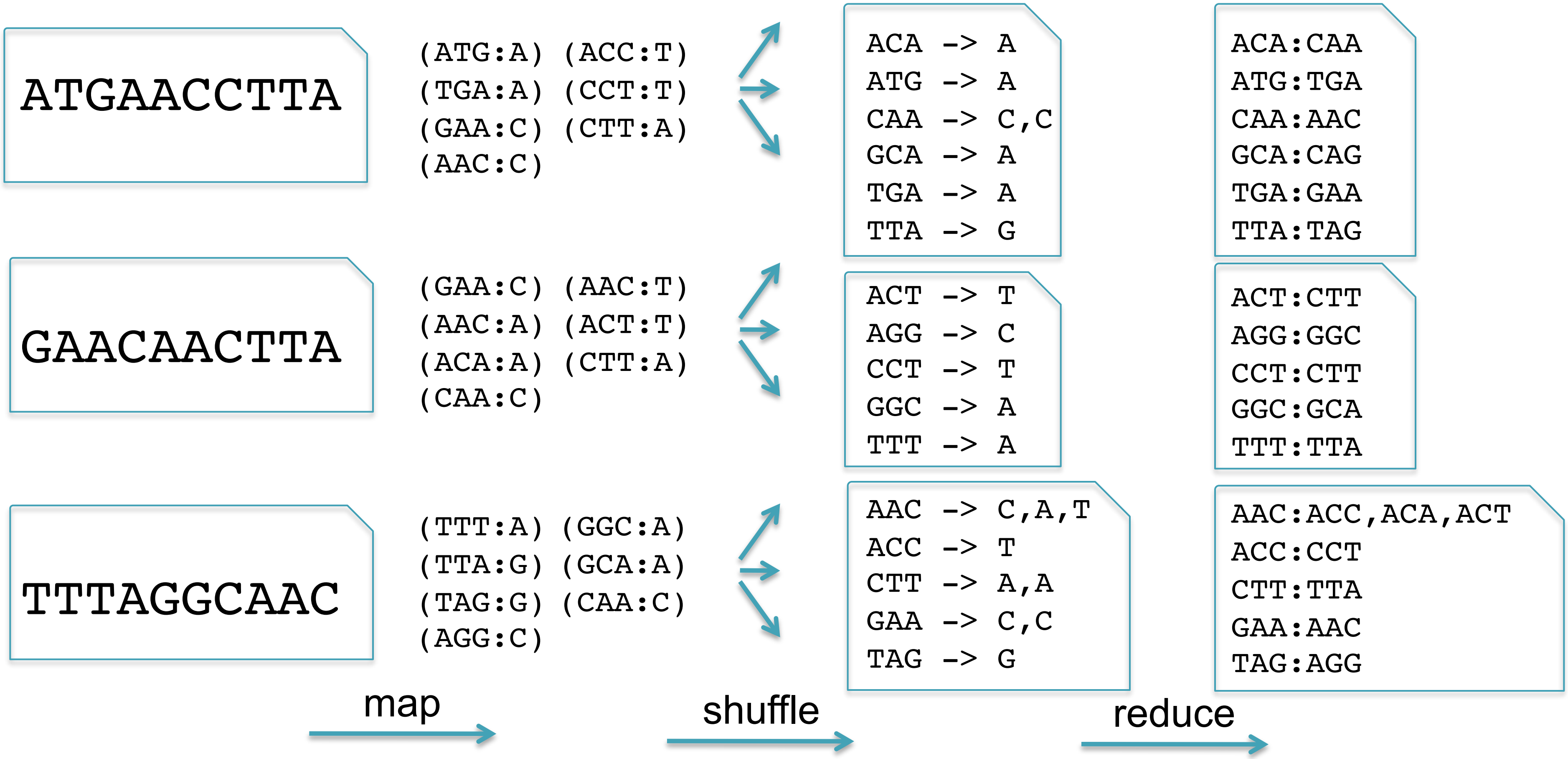
shuffle

reduce

Graph Construction

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

Map, Shuffle & Reduce
All Run in Parallel



ATGAACCTTA

GAACAACTTA

TTTAGGCAAC

```
(ATG:A) (ACC:T)
(TGA:A) (CCT:T)
(GAA:C) (CTT:A)
(AAC:C)

(GAA:C) (AAC:T)
(AAC:A) (ACT:T)
(ACA:A) (CTT:A)
(CAA:C)

(TTT:A) (GGC:A)
(TTA:G) (GCA:A)
(TAG:G) (CAA:C)
(AGG:C)
```

```
ACA -> A
ATG -> A
CAA -> C,C
GCA -> A
TGA -> A
TTA -> G

ACT -> T
AGG -> C
CCT -> T
GGC -> A
TTT -> A

AAC -> C,A,T
ACC -> T
CTT -> A,A
GAA -> C,C
TAG -> G
```

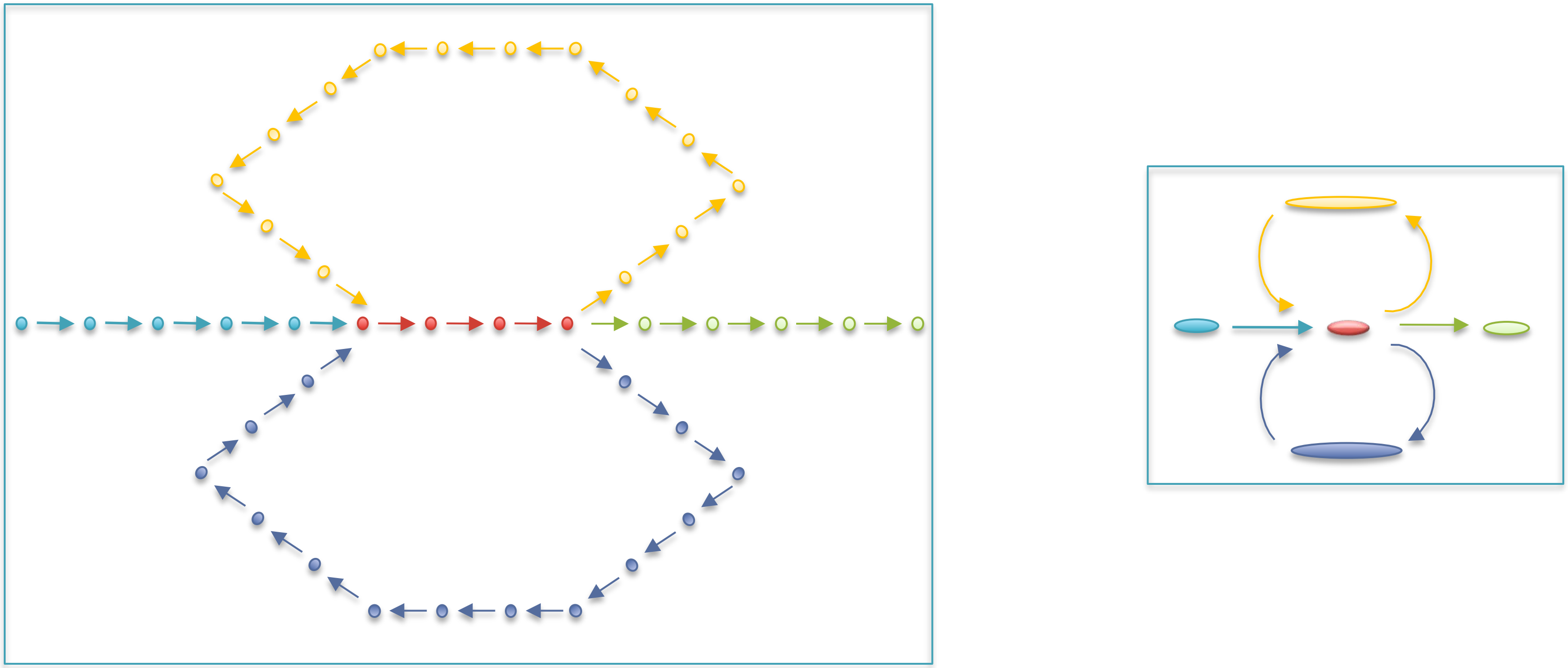
```
ACA:CAA
ATG:TGA
CAA:AAC
GCA:CAG
TGA:GAA
TTA:TAG

ACT:CTT
AGG:GGC
CCT:CTT
GGC:GCA
TTT:TTA

AAC:ACC,ACA,ACT
ACC:CCT
CTT:TTA
GAA:AAC
TAG:AGG
```

Graph Compression

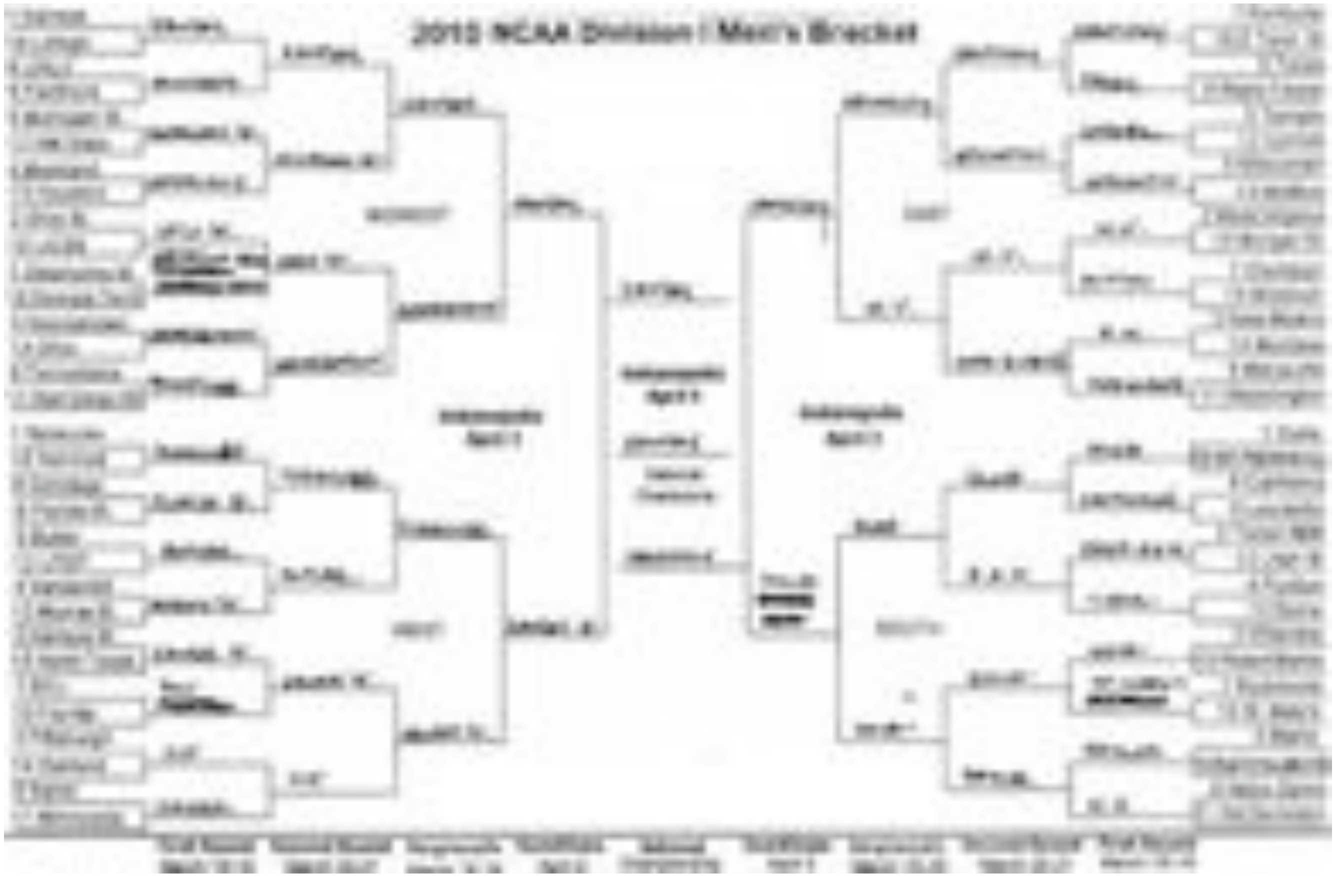
- After construction, many edges are unambiguous
  - Merge together compressible nodes
  - Graph physically distributed over hundreds of computers



High School Dance

Warmup Exercise

- Who here was born closest to June 16?
  - You can only compare to 1 other person at a time



Find winner among 64 teams in just 6 rounds

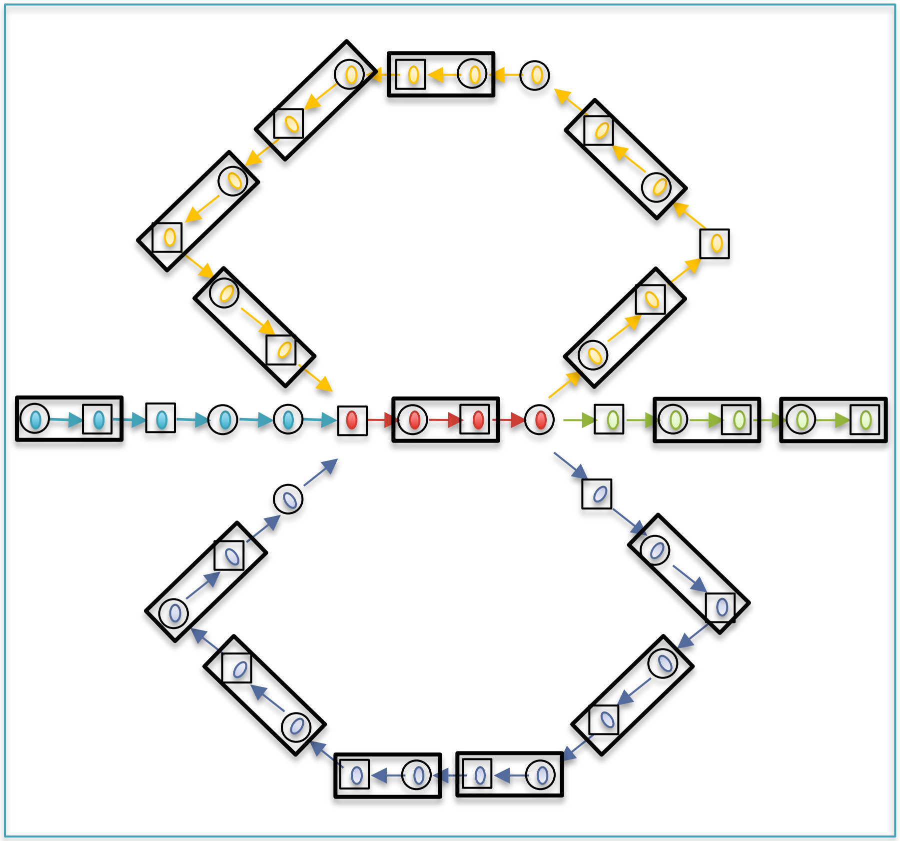
Fast Path Compression

Challenges

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

Randomized List Ranking

- Randomly assign Ⓗ/$\boxed{\text{T}}$ to each compressible node
- Compress Ⓗ➔$\boxed{\text{T}}$ links



Initial Graph: 42 nodes

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

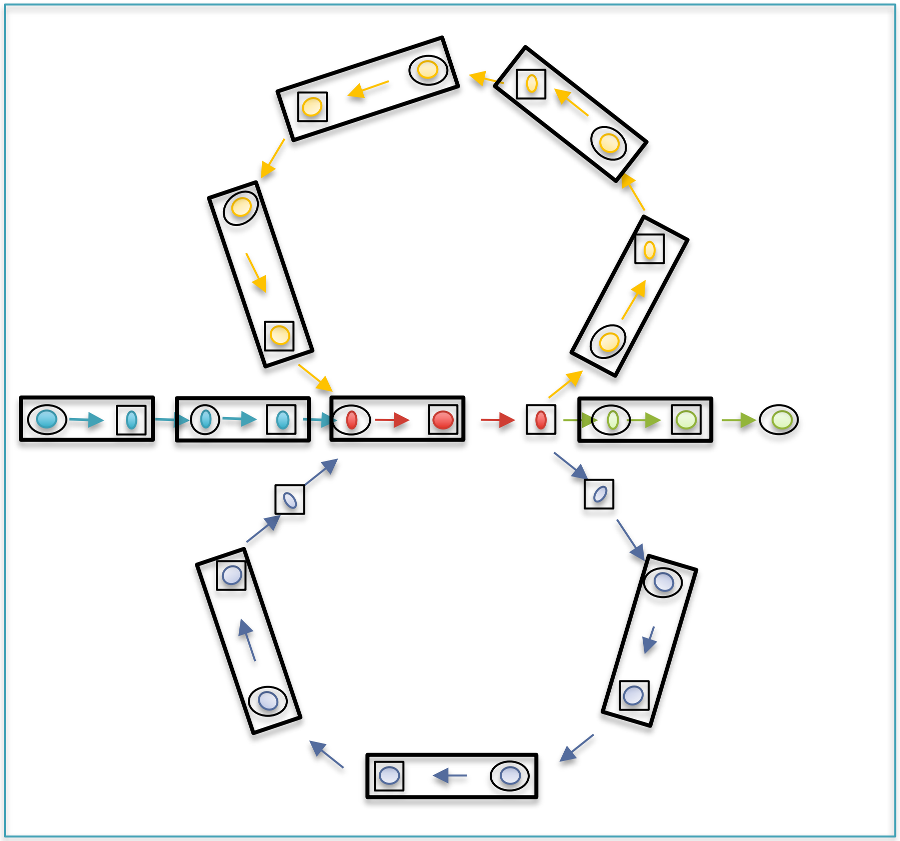
Fast Path Compression

Challenges

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

Randomized List Ranking

- Randomly assign Ⓗ/$\boxed{\text{T}}$ to each compressible node
- Compress Ⓗ→$\boxed{\text{T}}$ links



Round 1: 26 nodes (38% savings)

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

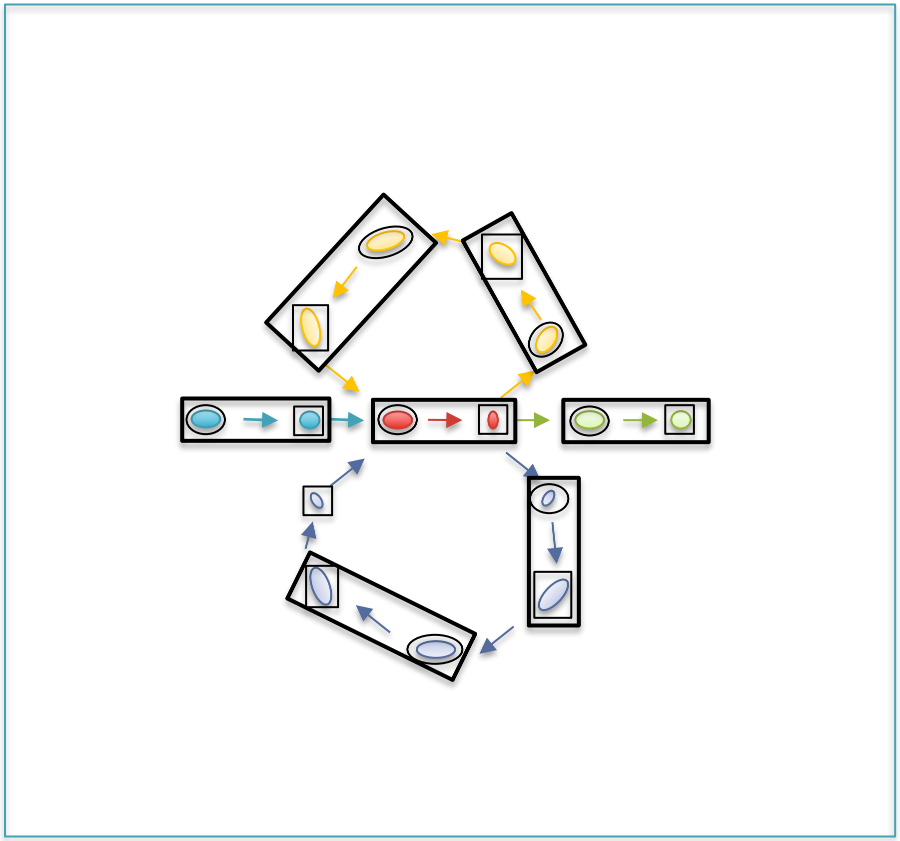
Fast Path Compression

Challenges

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

Randomized List Ranking

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

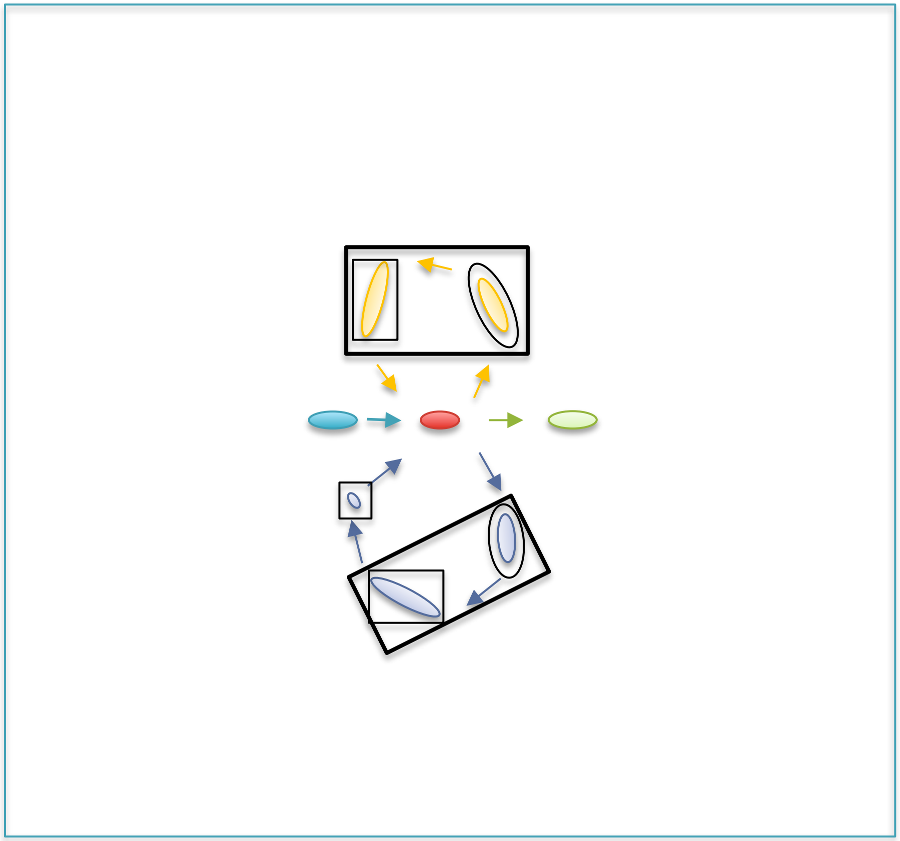
Fast Path Compression

Challenges

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

Randomized List Ranking

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



Round 2: 8 nodes (81% savings)

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

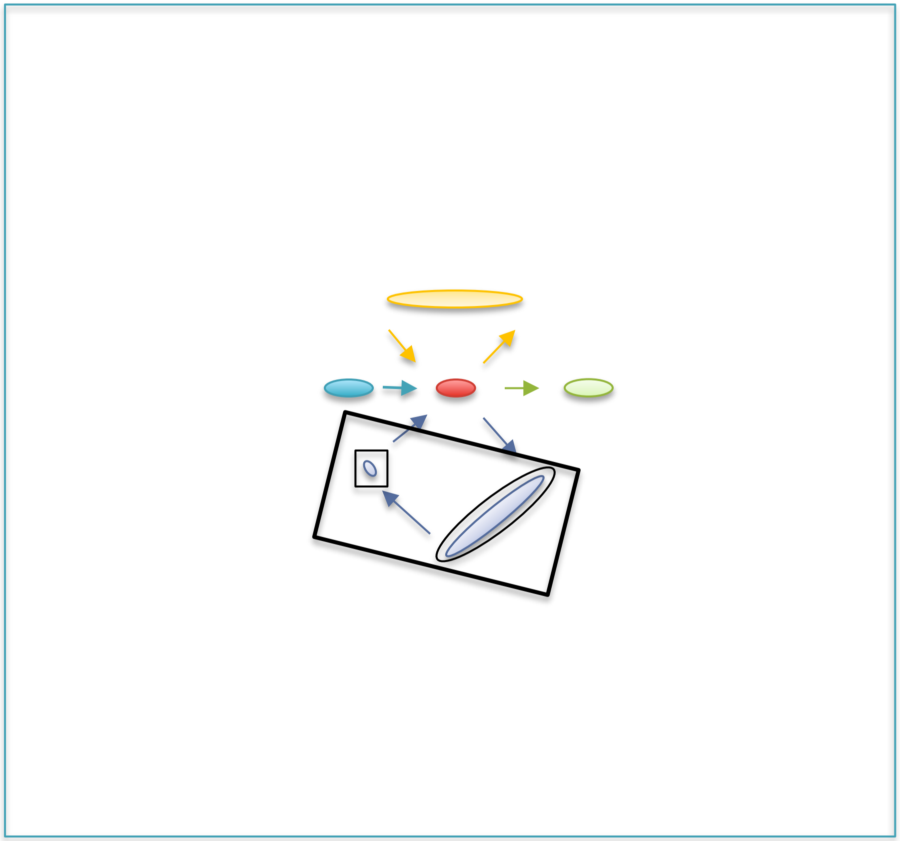
Fast Path Compression

Challenges

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

Randomized List Ranking

- Randomly assign Ⓗ/ $\boxed{T}$ to each compressible node
- Compress Ⓗ→$\boxed{T}$ 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

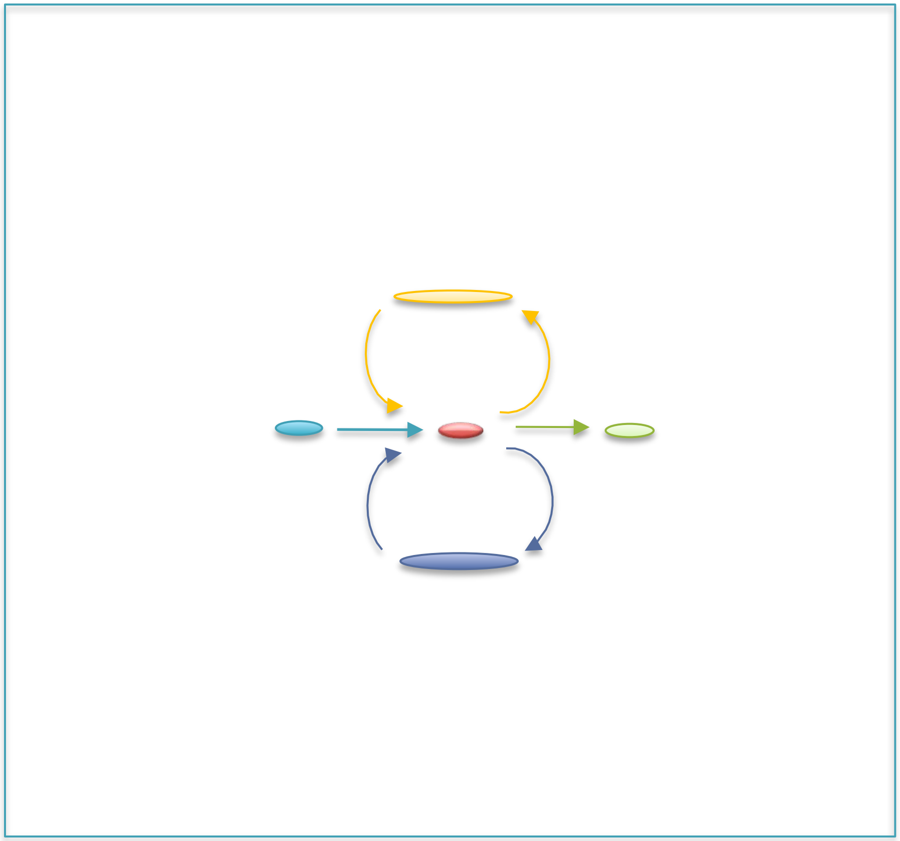
- Nodes stored on different computers
- Nodes can only access direct neighbors

Randomized List Ranking

- Randomly assign Ⓗ/ $\boxed{\text{T}}$ to each compressible node
- Compress Ⓗ→$\boxed{\text{T}}$ links

Performance

- Compress all chains in log(S) rounds
- If <1024 nodes to compress then assign them all to the same reducer
  - Save last 10 rounds



Round 4: 5 nodes (88% savings)

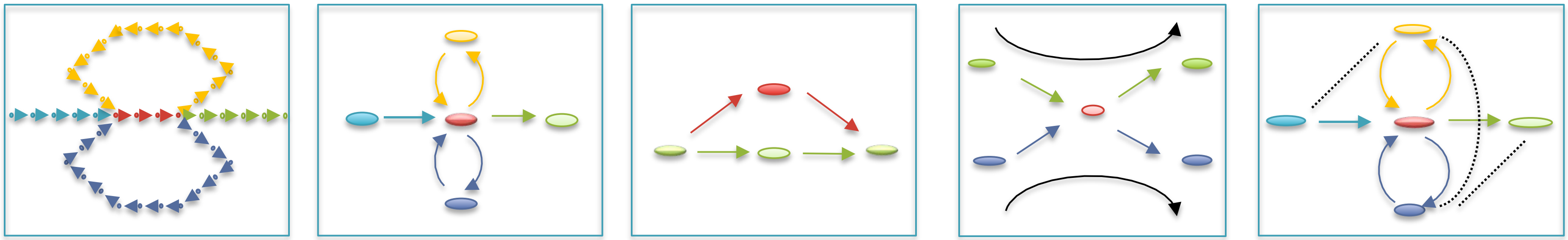
Randomized Speed-ups in Parallel Computation.
Vishkin U. (1984) *ACM Symposium on Theory of Computation. 230-239.*

Contrail

http://contrail-bio.sourceforge.net

Scalable Genome Assembly with MapReduce

- *Genome: E. coli* K12 MG1655, 4.6Mbp
- *Input:* 20.8M 36bp reads, 200bp insert (~150x coverage)
- *Preprocessor*: Quality-Aware Error Correction



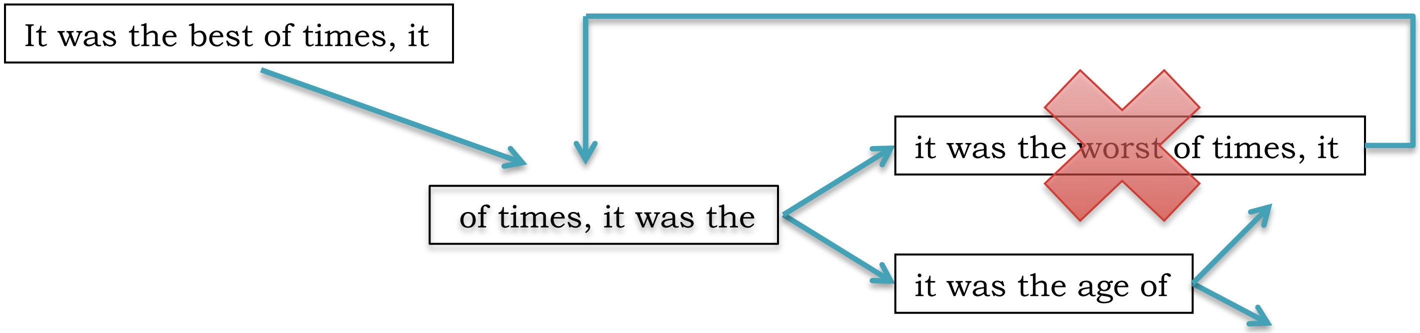
	Initial	Compressed	Error Correction	Resolve Repeats	Cloud Surfing
N	5.1 M	245,131	2,769	1,909	300
Max	27 bp	1,079 bp	70,725 bp	90,088 bp	149,006 bp
N50	27 bp	156 bp	15,023 bp	20,062 bp	54,807 bp

Assembly of Large Genomes with Cloud Computing.
Schatz MC, Sommer D, Kelley D, Pop M, *et al. In Preparation.*

E. coli Assembly Quality

Incorrect contigs: Align at < 95% identity or < 95% of their length

Assembler	Contigs ≥ 100bp	N50 (bp)	Incorrect contigs
Contrail PE	300	54,807	4
Contrail SE	529	20,062	0
SOAPdenovo PE	182	89,000	5
ABySS PE	233	45,362	13
Velvet PE	286	54,459	9
EULER-SR PE	216	57,497	26
SSAKE SE	931	11,450	38
Edena SE	680	16,430	6



It was the best of times, it

of times, it was the

it was the worst of times, it

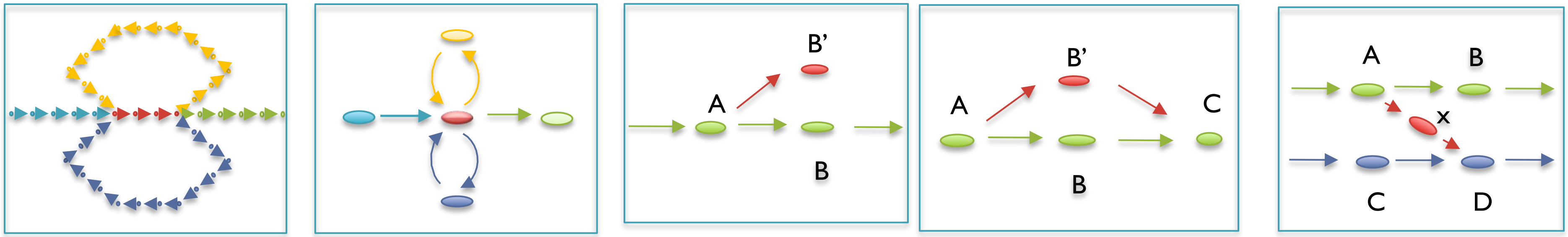
it was the age of

Contrail

http://contrail-bio.sourceforge.net

De Novo Assembly of the Human Genome

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



	Initial	Compressed	Clip Tips	Pop Bubbles	Chimeric Edges
N	>7 B	>1 B	5.0 M	4.2 M	
Max	27 bp	303 bp	14,007 bp	20,594 bp	In progress
N50	27 bp	< 100 bp	650 bp	923 bp	

Assembly of Large Genomes with Cloud Computing.
Schatz MC, Sommer D, Kelley D, Pop M, *et al. In Preparation.*

Scalable Solutions for DNA Sequence Analysis

Resource	Description
CloudBLAST	Scalable BLAST in the Clouds http://www.acis.ufl.edu/~ammatsun/mediawiki-1.4.5/index.php/CloudBLAST_Project
CloudBurst	Highly Sensitive Short Read Mapping http://cloudburst-bio.sf.net
Cloud BioLinux	Appliance for Genome Assembly and Alignment http://www.cloudbiolinux.com/
Cloud RSD	Reciprocal Smallest Distance Ortholog Detection http://roundup.hms.harvard.edu
CLoVR	Appliance for Genome and metagenome annotation and analysis http://clover.igs.umaryland.edu
Galaxy	Platform for interactive large-scale genome analysis http://galaxy.psu.edu
Myrna	Differential expression analysis of mRNA-seq http://bowtie-bio.sf.net/myrna/
Quake	Quality guided correction of short reads http://github.com/davek44/error_correction/

Cloud Solutions for DNA Sequence Analysis

- Rapid & elastic deployment of vast computation
  - Accessible, Reproducible, Reliable, Collaborative

- Why not?
  - Transfer time
  - Privacy / security requirements
  - Time and expertise required for development
  - Price
  - What are the alternatives?

Research Directions

- Scalable Sequencing
  - Genomes, Metagenomes, *-Seq, Personalized Medicine
  - How do we survive the tsunami of sequence data?
    - Efficient indexing & algorithms, multi-core & multi-disk systems

- Practically Parallel
  - Managing n-tier memory hierarchies, crossing the PRAM chasm
  - How do we solve problems with 1000s of cores?
    - Locality, Fault Tolerance, Programming Languages & Parallel Systems

- Computational Discovery
  - Abundant data and computation are necessary, but not sufficient
  - How do we gain insight?
    - Statistics & Modeling, Machine Learning, Databases, Visualization & HCI

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

- Surviving the data deluge means computing in parallel
  - Cloud computing is an attractive platform for large scale sequence analysis and computation
- Emerging technologies are a great start, but we need continued research
  - A word of caution: new technologies are new

Acknowledgements

Advisor

Steven Salzberg

UMD Faculty

Mihai Pop, Art Delcher, Amitabh Varshney,
Carl Kingsford, Ben Shneiderman,
James Yorke, Jimmy Lin, Dan Sommer

CBCB Students

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

Thank You!

http://www.cbcb.umd.edu/~mschatz

@mike_schatz