CloudBurst, Crossbow & Contrail: Scaling Up Bioinformatics with Cloud Computing

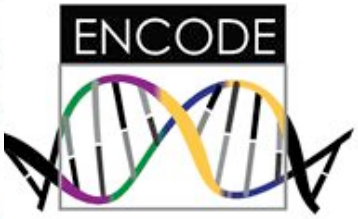
Michael Schatz

March 15, 2010
CHI XGen Congress

The Evolution of DNA Sequencing

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)



Critical Computational Challenges: Alignment and Assembly of Huge Datasets

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

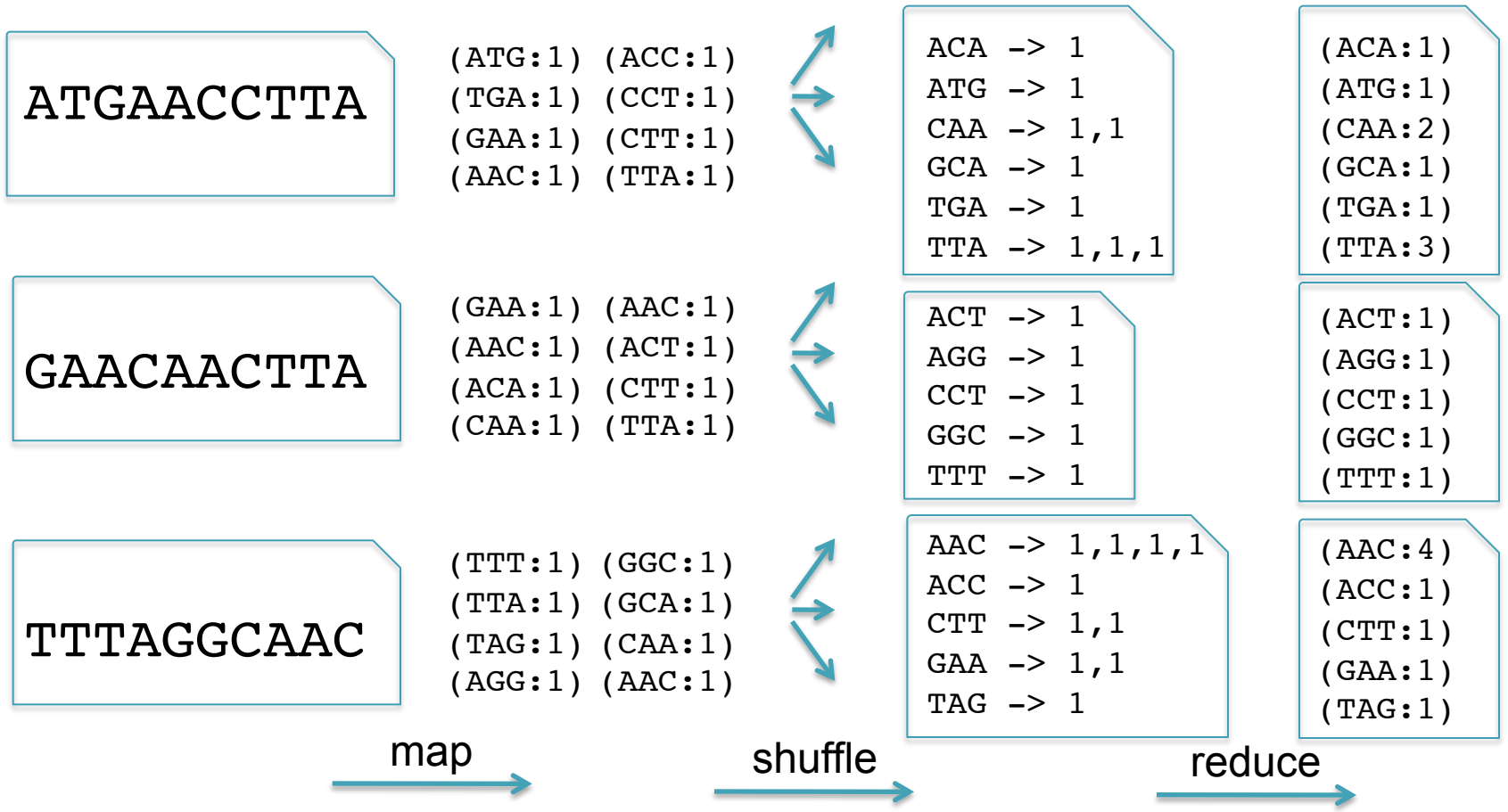
- Benefits
  - Scalable, Efficient, Reliable
  - Easy to Program
  - Runs on commodity computers

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

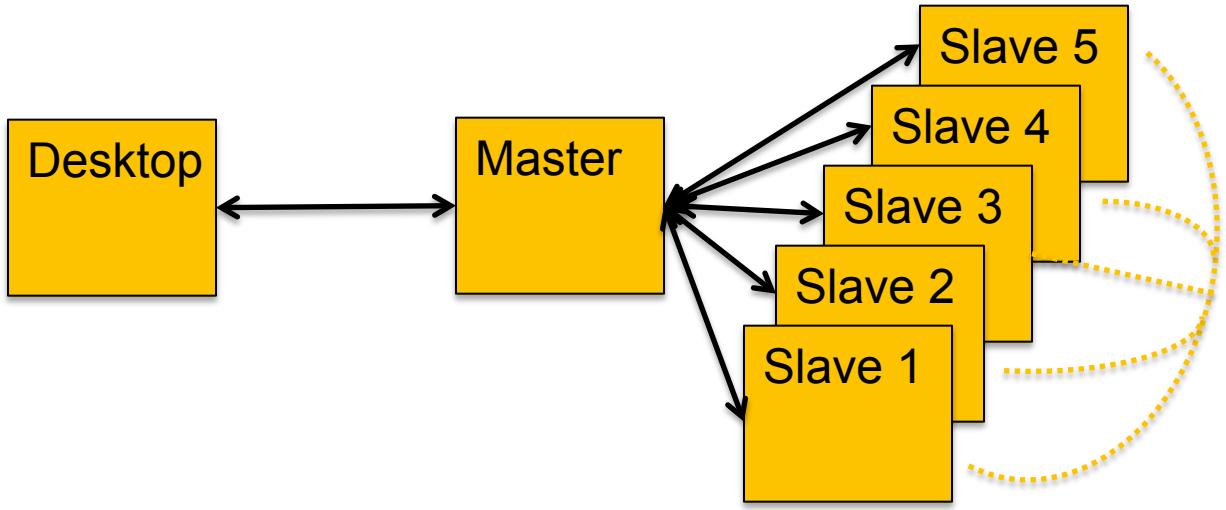
K-mer Counting

- 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



Hadoop Architecture



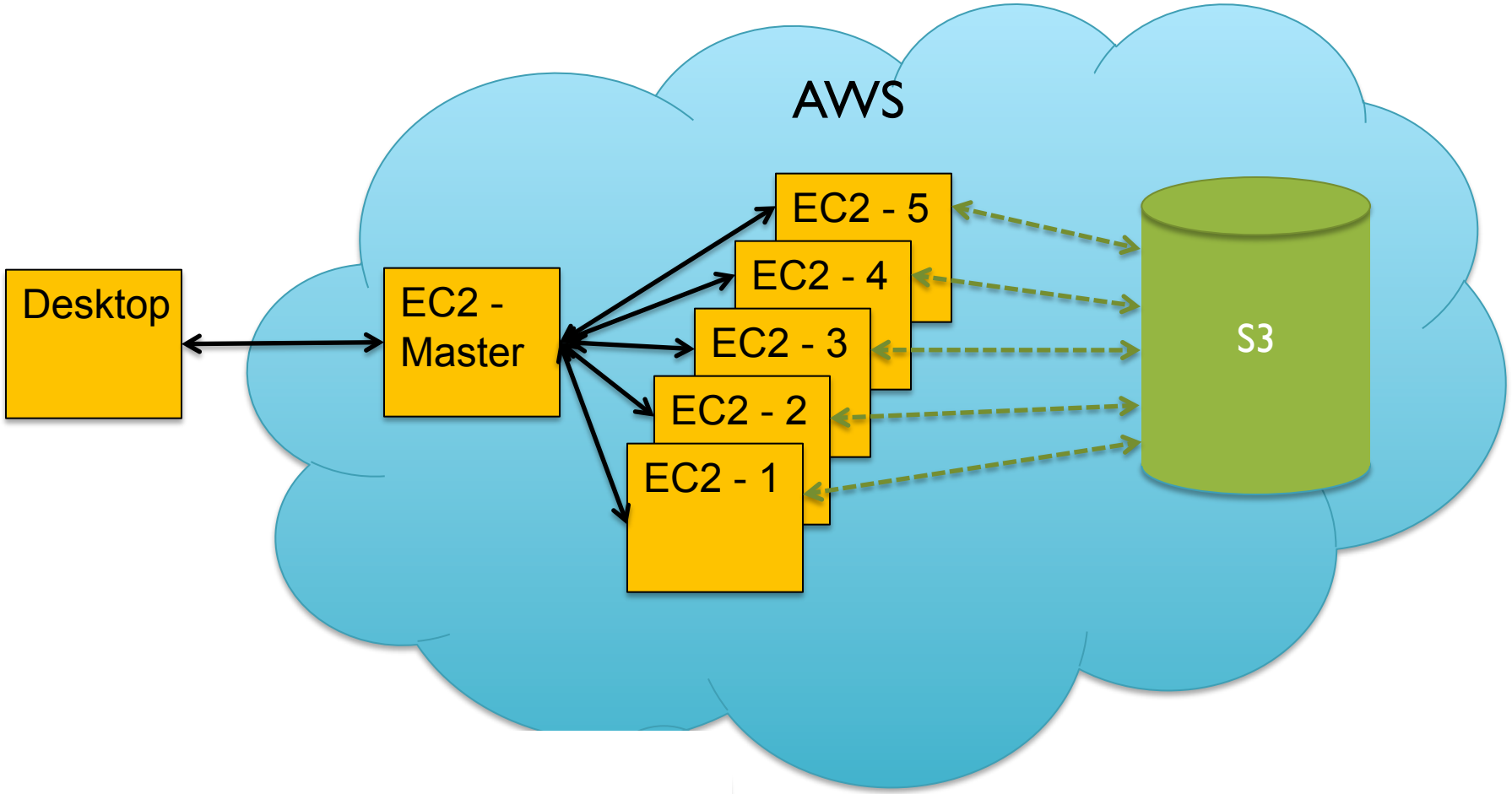
- Hadoop Distributed File System (HDFS)
  - Data files partitioned into large chunks (64MB), replicated on multiple nodes
  - NameNode stores metadata information (block locations, directory structure)
- Master node (JobTracker) schedules and monitors work on slaves
  - Computation moves to the data, 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

Amazon Web Services

http://aws.amazon.com

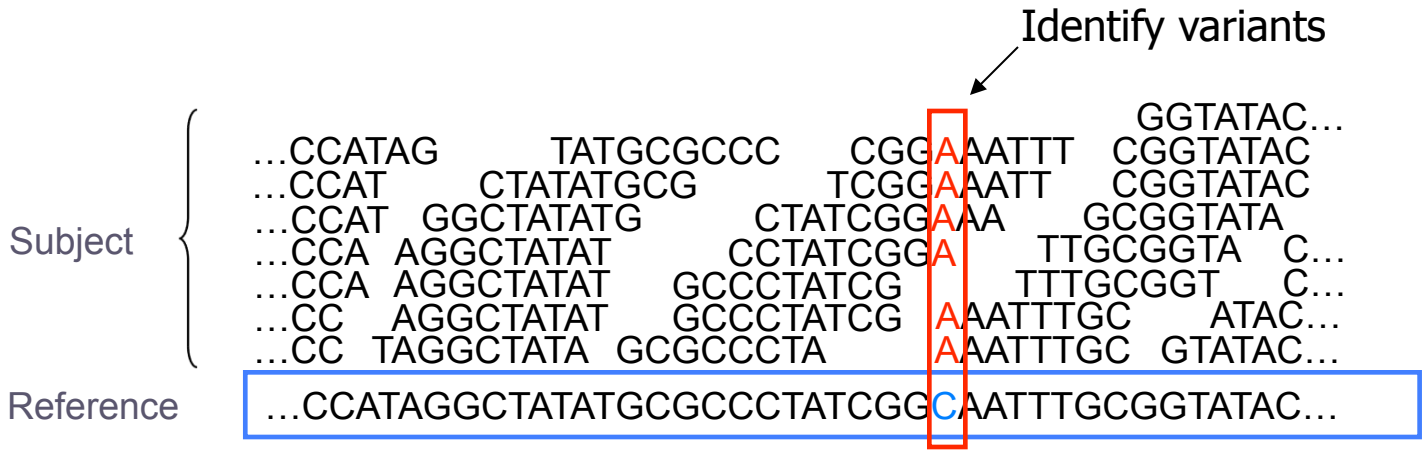
- Elastic Compute Cloud (EC2)
  - On demand computing power
    - Support for Windows, Linux, & OpenSolaris
    - Starting at 8.5¢ / core / hour
- Simple Storage Service (S3)
  - Scalable data storage
    - 10¢ / GB upload fee, 15¢ / GB monthly fee
- Elastic MapReduce (EMR)
  - Point-and-click Hadoop Workflows
    - Computation runs on EC2

Hadoop on AWS



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 with MapReduce

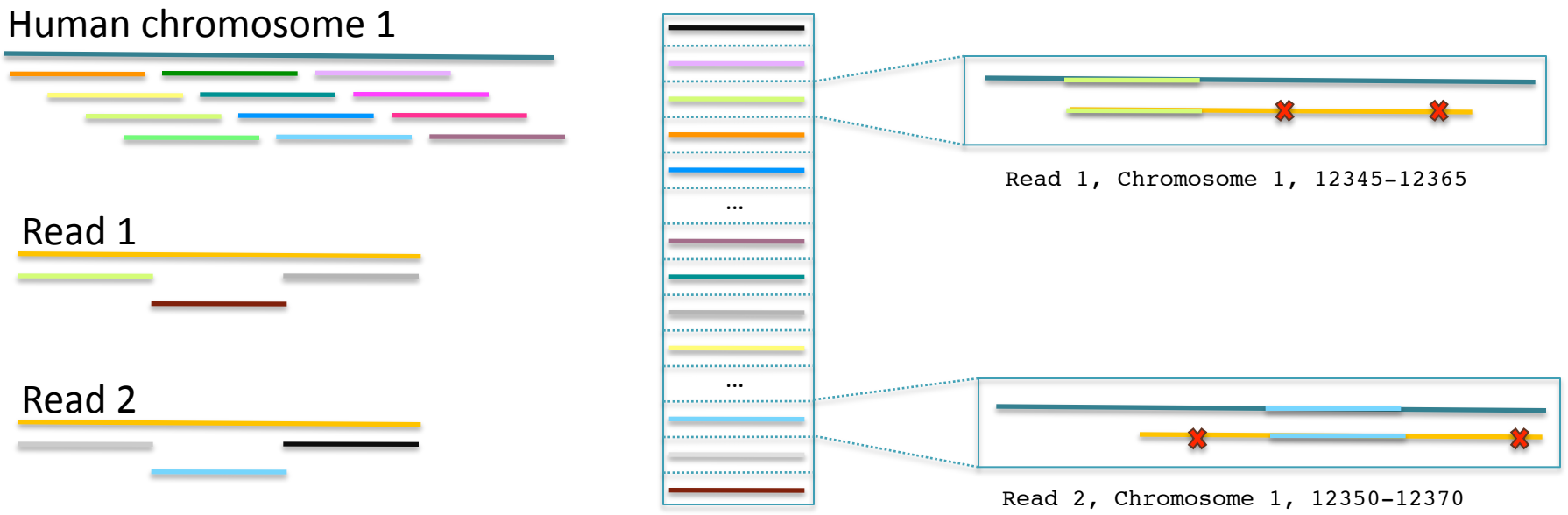


- Given a reference and many subject reads, report one or more "good" end-to-end alignments per alignable read
  - Finds where in the genome the read most likely originated
- Mapping of a whole human requires ~1,000 CPU hours
  - Alignments are "embarassingly parallel" by read
  - Variant detection is parallel by chromosome region

CloudBurst

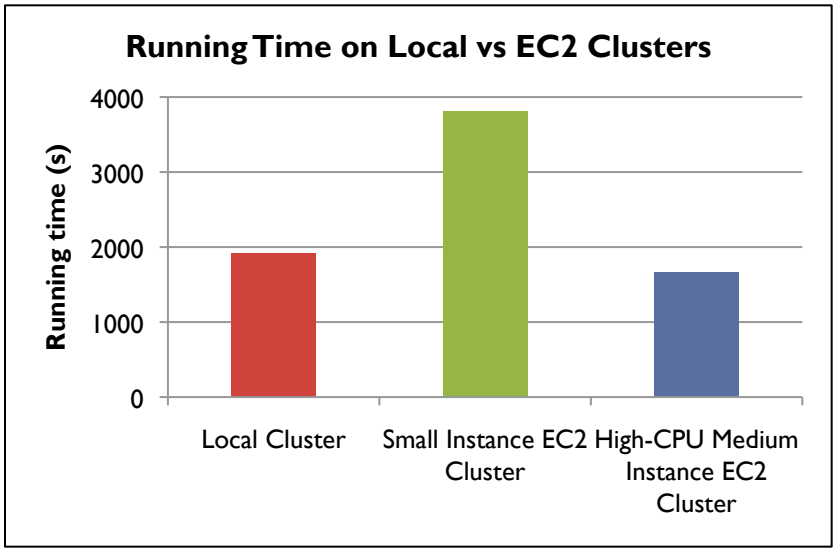
http://cloudburst-bio.sourceforge.net

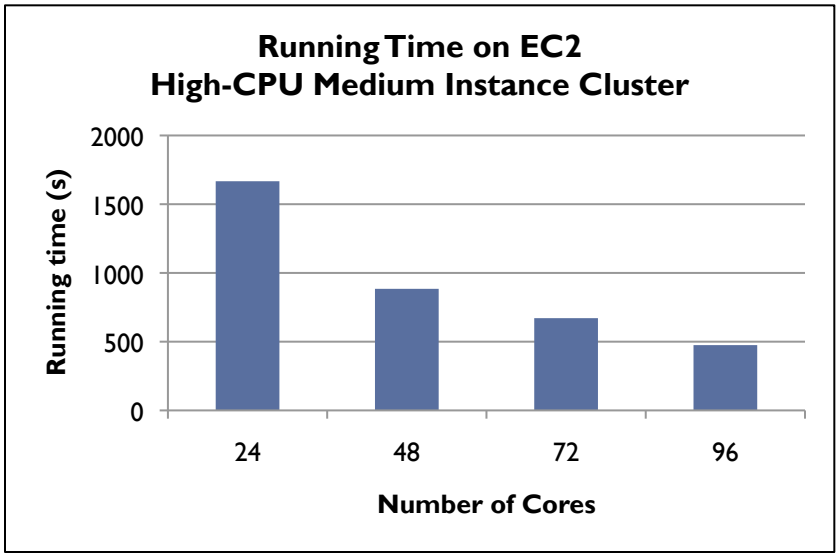
1. Map: Catalog K-mers
   - Emit k-mers in the genome and reads
2. Shuffle: Collect Seeds
   - Conceptually build a inverted index of k-mers
3. Reduce: End-to-end alignment
   - If read aligns end-to-end with ≤ k errors, record the alignment



EC2 Evaluation

http://cloudburst-bio.sourceforge.net





Evaluate mapping 7M reads to human chromosome 22 with at most 4 mismatches on a local and 2 EC2 clusters.

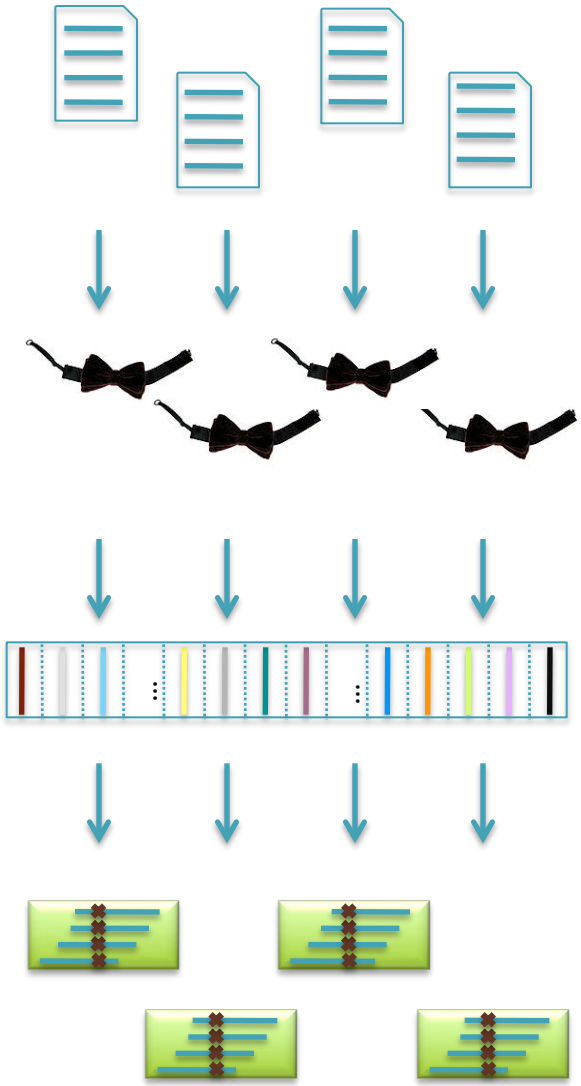
- 24-core High-CPU Medium Instance EC2 cluster is faster than 24-core local cluster.
- 96-core cluster is 3.5x faster than the 24-core, and 100x faster than serial RMAP.

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

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 CPUs	$3.40
Setup	0h : 15m	320 CPUs	$13.94
Alignment	1h : 30m	320 CPUs	$41.82
Variant Calling	1h : 00m	320 CPUs	$27.88
End-to-end	4h : 00m		$97.69

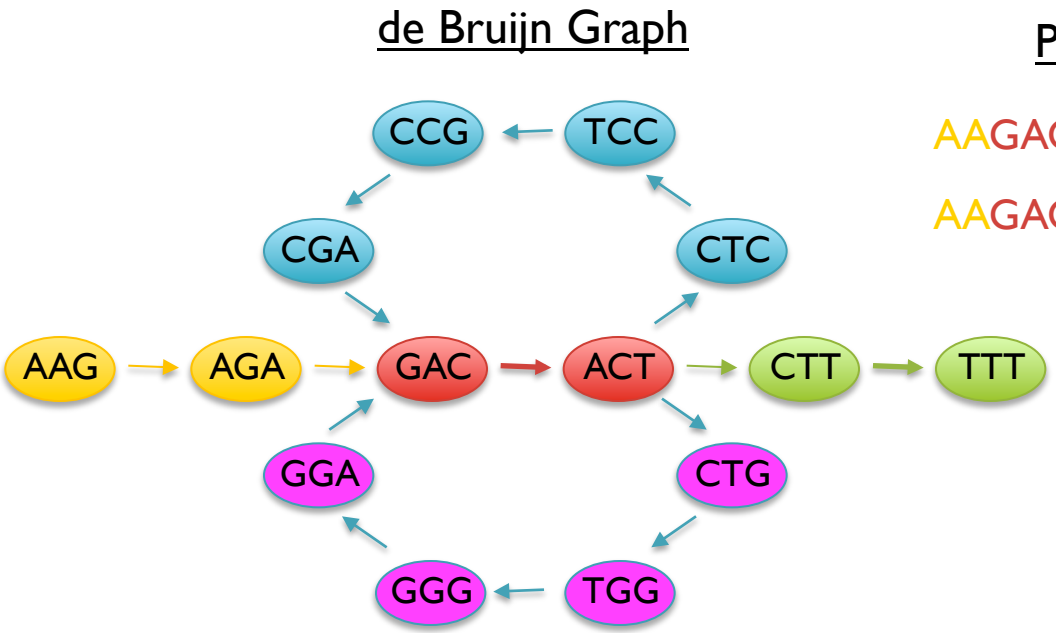
Analyze an entire human genome for < $100 in an afternoon.
Accuracy validated at > 99%

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

Short Read Assembly

Reads

AAGA
ACTT
ACTC
ACTG
AGAG
CCGA
CGAC
CTCC
CTGG
CTTT
…



Potential Genomes

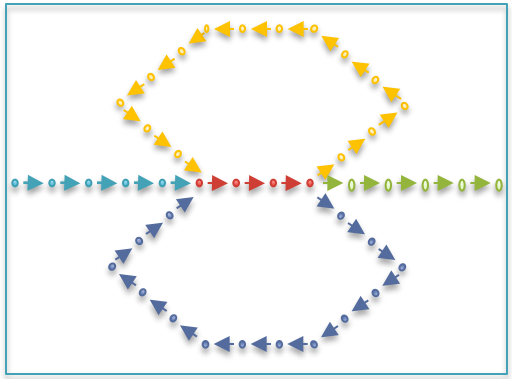
AAGACTCCGACTGGGACTTT

AAGACTGGGACTCCGACTTT

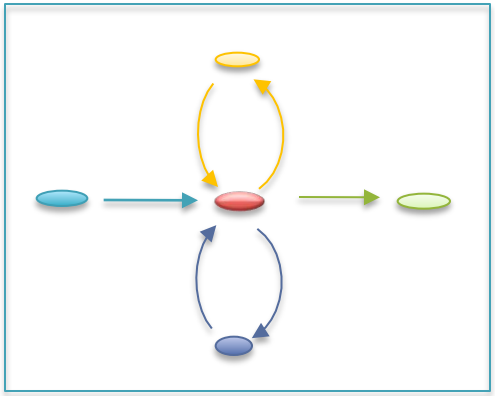
- Genome assembly as finding an Eulerian tour of the de Bruijn graph
  - Human genome: >3B nodes, >10B edges
- The new short read assemblers require tremendous computation
  - Velvet (Zerbino & Birney, 2008) serial: > 2TB of RAM
  - ABySS (Simpson *et al.*, 2009) MPI: 168 cores x ~96 hours
  - SOAPdenovo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

Genome Assembly with MapReduce

- Advantages
  - Proven system for processing huge datasets
    - PageRank: Significance of >1 trillion pages
    - CloudBurst: Highly Sensitive Alignment
    - Crossbow: Searching for SNPs in the Clouds
  - Simple programming model
    - Reliability, redundancy, scalability built-in



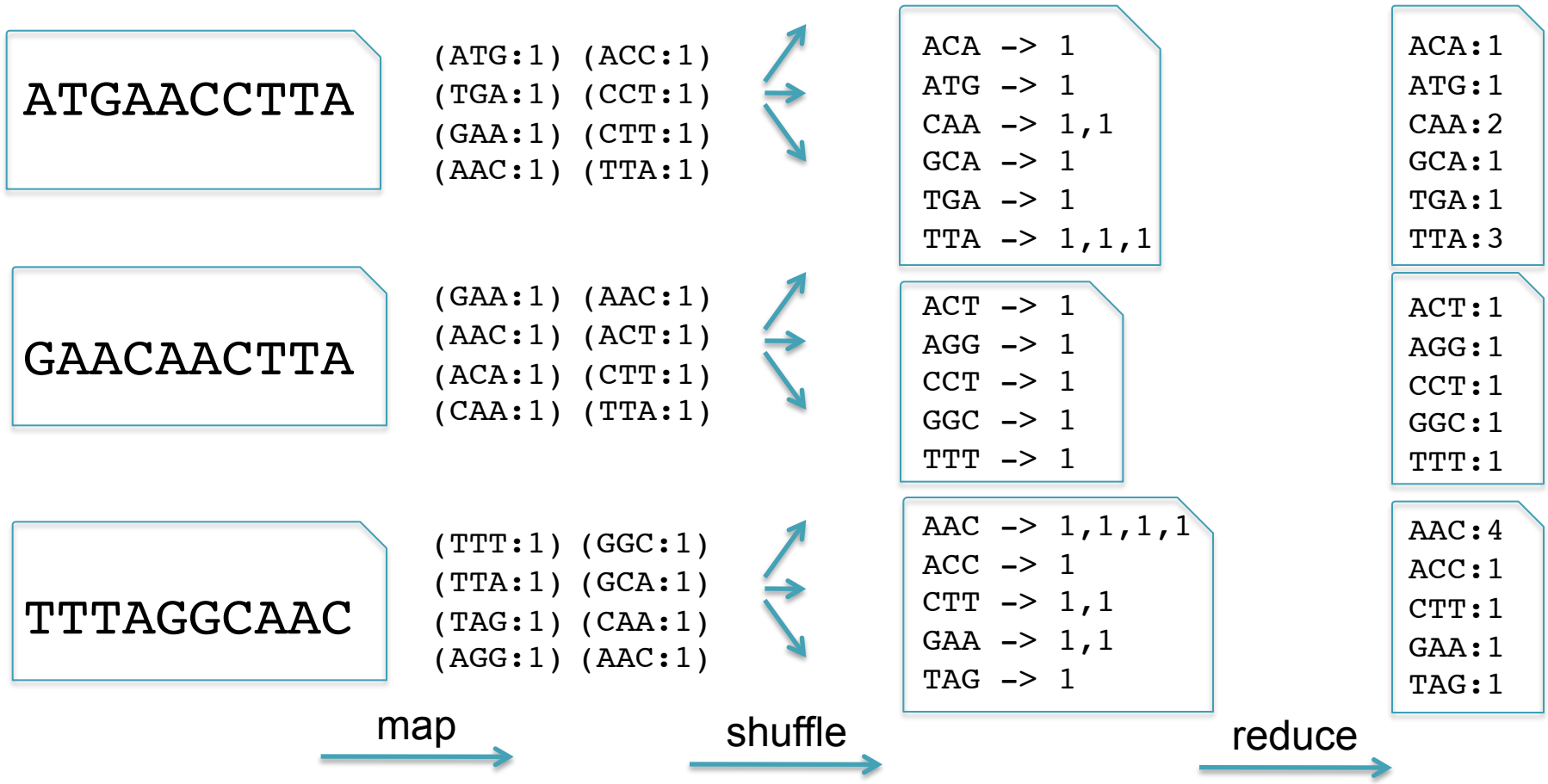
- Challenges
  - How to efficiently implement assembly graph algorithms when adjacent nodes are stored on different machines?
    - Restricted programming model (not Shared Memory, not MPI)



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

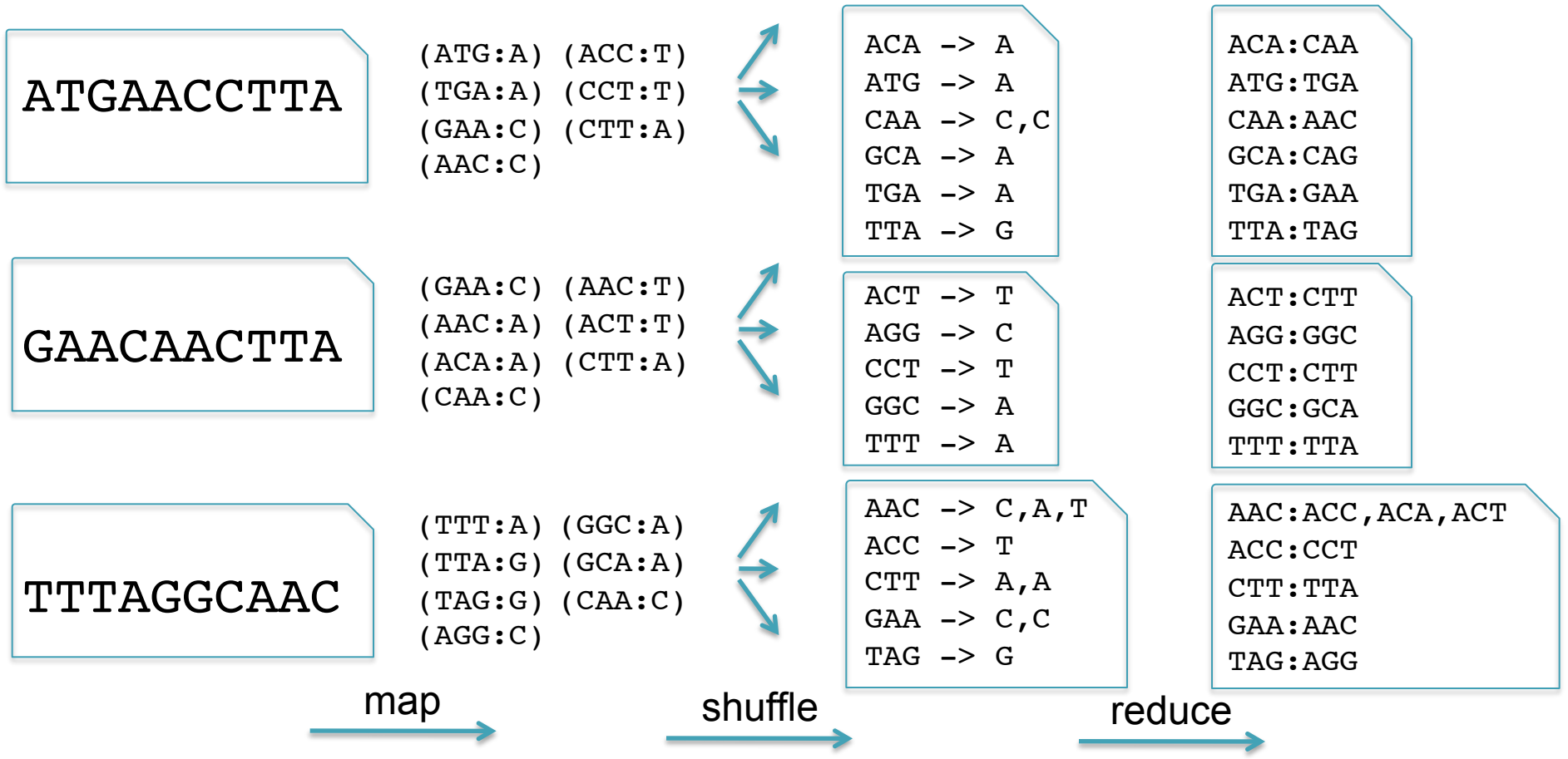
Map, Shuffle & Reduce
All Run in Parallel



Graph Construction

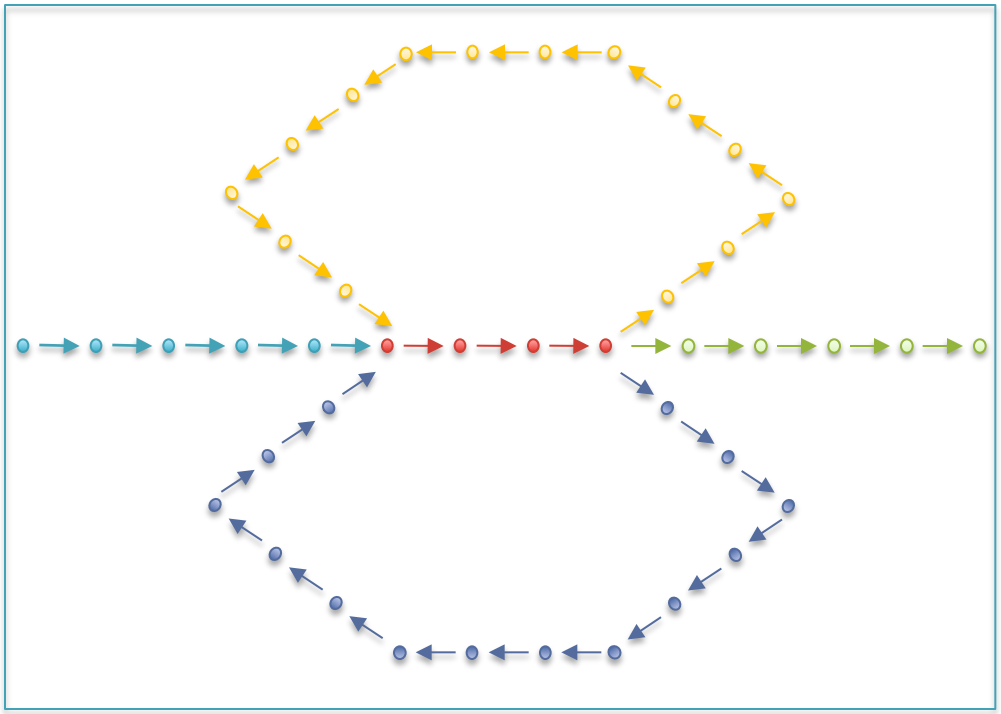
- Application developers focus on 2 (+1 internal) functions
  - Map: input ➔ key:value pairs
  - Shuffle: Group together pairs with same key
  - Reduce: key, value-lists ➔ output

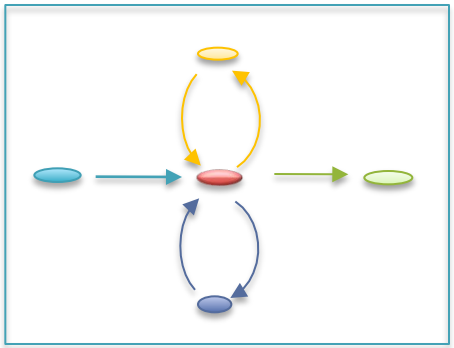
Map, Shuffle & Reduce
All Run in Parallel



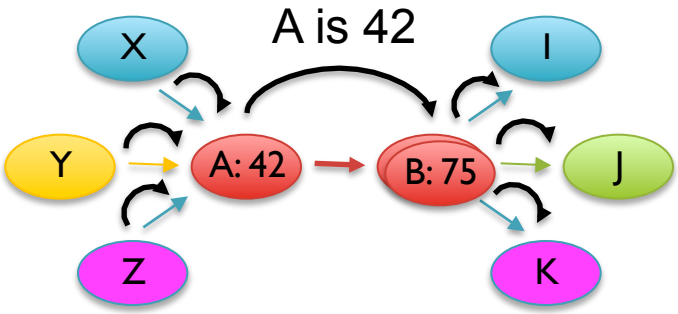
Graph Compression

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





Distributed Graph Processing



MapReduce
Message Passing

Input:

– Graph stored as node tuples

```
A: (N E:B W:42)
B: (N E:I,J,K W:33)
```

Map

– For all nodes, re-emit node tuple

– For all neighbors, emit value tuple

```
A: (N E:B W:42)
B: (V A 42)
B: (N E:I,J,K W:33)
…
```

Shuffle

– Collect tuples with same key

```
B: (N E:I,J,K W:33)
B: (V A 42)
```

Reduce

– Add together values, save updated node tuple

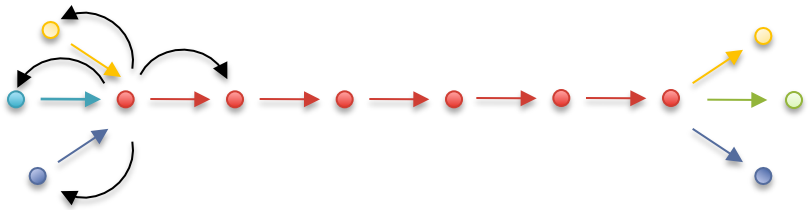
```
B: (N E:I,J,K W:75)
```

Iterative Path Compression

Iteratively identify and collapse the beginning of each chain

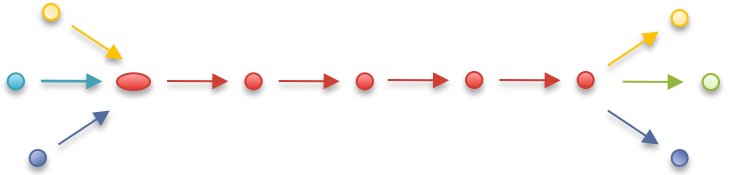
Map:

- Emit messages to the neighbors of the head of each chain



Reduce:

- Update links, node label



Requires S MapReduce cycles, where S is the length of the longest simple path

- *B. anthracis*: L=5.2Mbp S=268,925
- *H. sapiens* chr 22: L=49.6Mbp S=33,832
- *H. sapiens* chr 1: L=247.2Mbp S=37,172

Fast Path Compression

Challenges

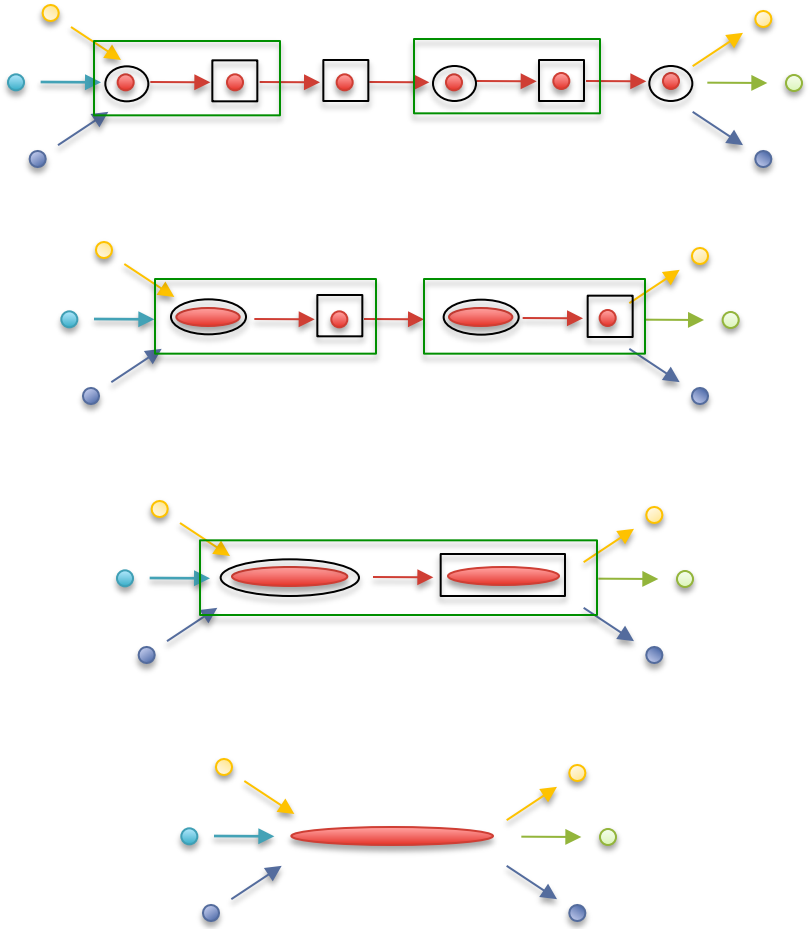
- Nodes stored on different computers
- Nodes can only access direct neighbors

Randomized List Ranking

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

Performance

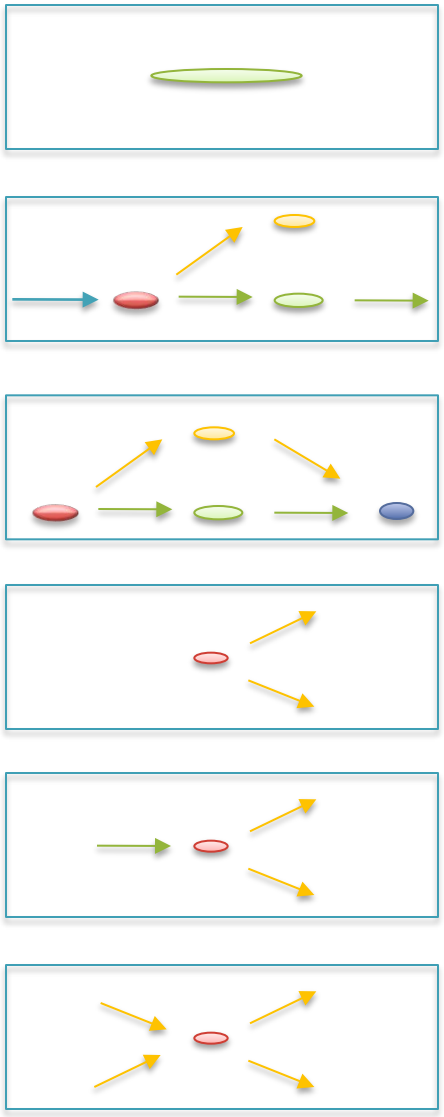
- Compress all chains in log(S) rounds (<20)
- If <1024 nodes to compress (from any number of chains), assign them all to the same reducer (save 10 rounds)



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

Node Types

(Chaisson, 2009)



Isolated nodes (10%)
- Contamination

Tips (46%)
- Clip short tips

Bubbles/Non-branch (9%)
- Pop bubbles

Dead Ends (.2%)
- Split forks

Half Branch (25%)
- Unzip

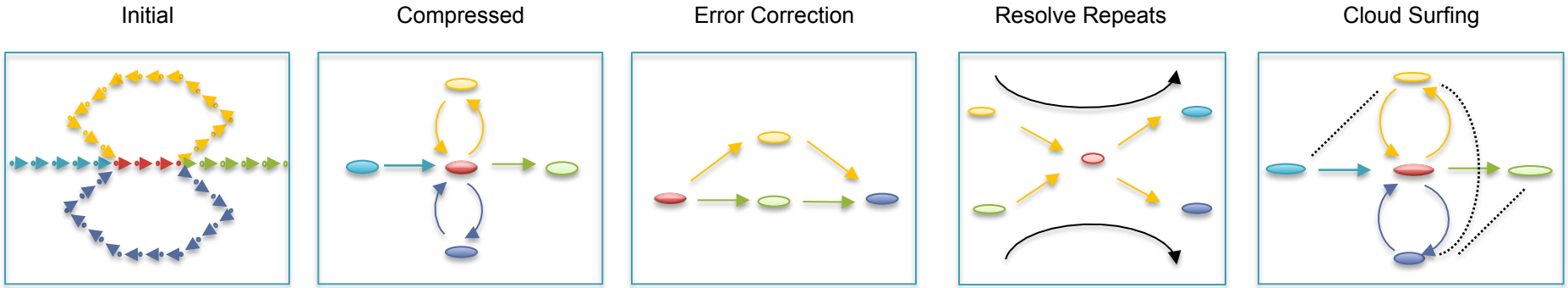
Full Branch (10%)
- Thread reads, cloud surfing

Contrail

http://contrail-bio.sourceforge.net

Scalable Genome Assembly with MapReduce

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



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.*

Summary

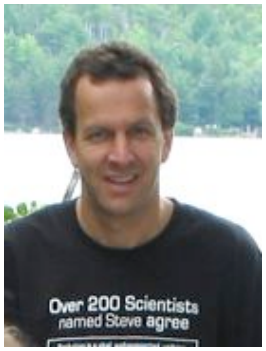
1. Hadoop is very well suited to analyzing very large next generation sequence datasets.

2. Hadoop Streaming for easy scaling of existing software.

3. Cloud computing is an attractive platform to augment resources.

4. Look for many cloud computing & MapReduce solutions this year.

Acknowledgements



Steven Salzberg

Mihai Pop

Jimmy Lin

Ben Langmead

Dan Sommer

David Kelley

amazon web services™

Thank You!

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

@mike_schatz