

Cloud Computing and the DNA Data Race

Michael Schatz

October 22, 2010
CSHL Advanced Sequencing Course



Outline

Part 1: Theory

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

Part 2: Practice

1. AWS Mini-Tutorial
2. Hadoop Mini-Tutorial



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

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

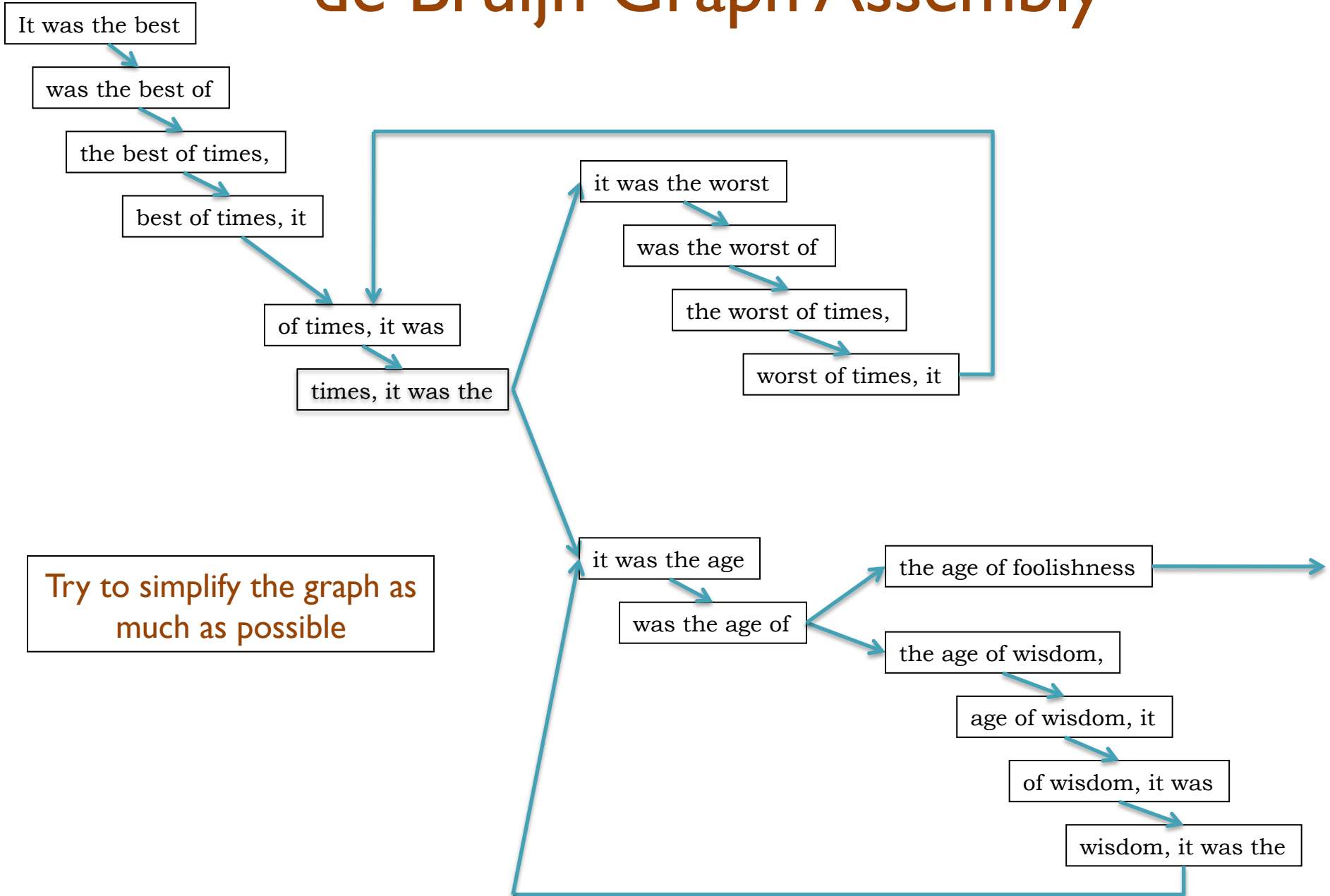
- Locally constructed graph reveals the global sequence structure
 - Overlaps between sequences implicitly computed

de Bruijn, 1946

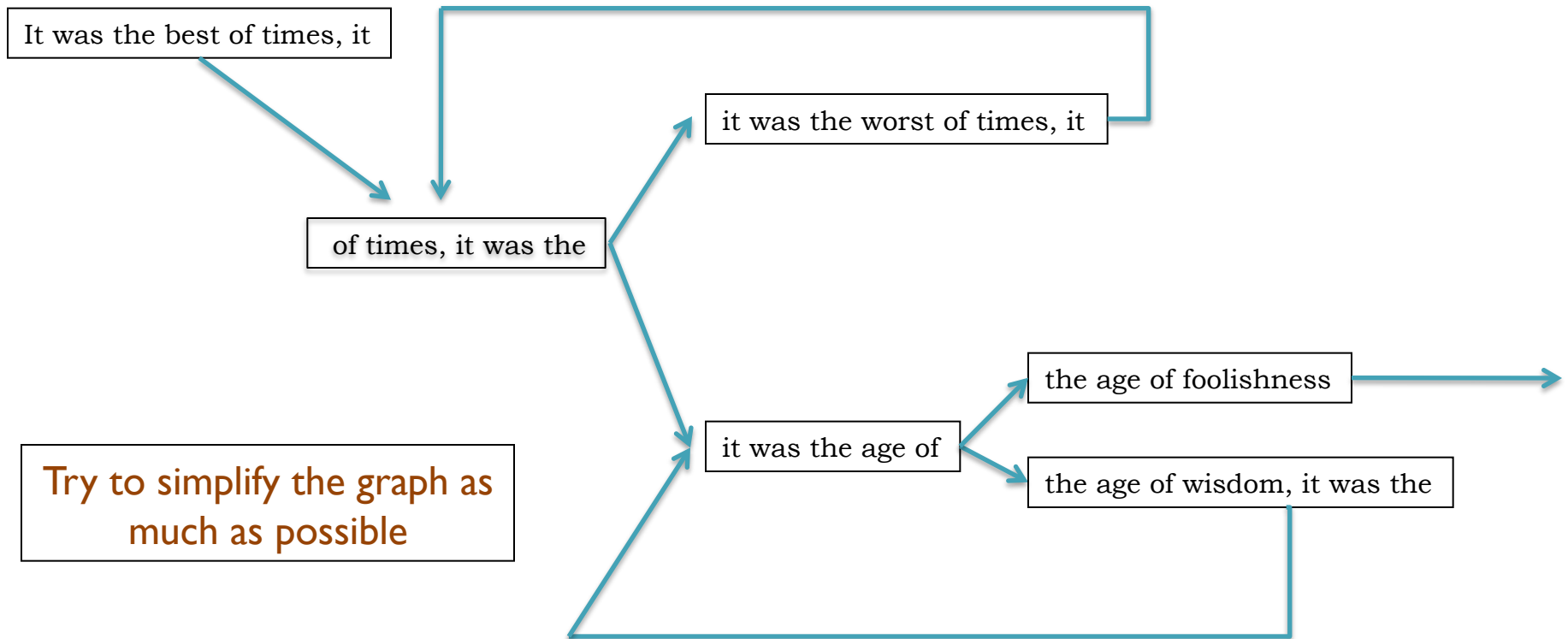
Idury and Waterman, 1995

Pevzner, Tang, Waterman, 2001

de Bruijn Graph Assembly



de Bruijn Graph Assembly

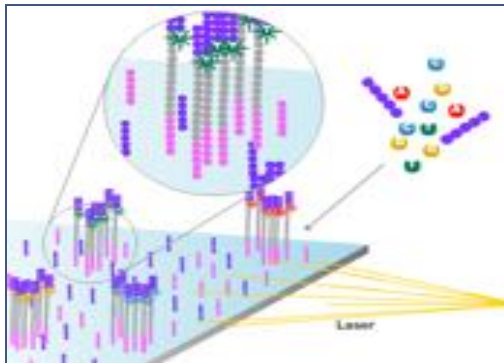


Molecular Biology & DNA Sequencing



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 sequence millions of short (25-500bp) reads from random positions of the genome

- Per-base error rate estimated at 1-2% (Simpson *et al*, 2009)

ATCTGATAAGTCCCAGGACTTCAGT

GCAAGGCAAACCCGAGCCCAGTTT

TCCAGTTCTAGAGTTTCACATGATC

GGAGTTAGTAAAAGTCCACATTGAG

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

- Use software to analyze the sequences
- Modern Biology requires Computational Biology

The DNA Data Race

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

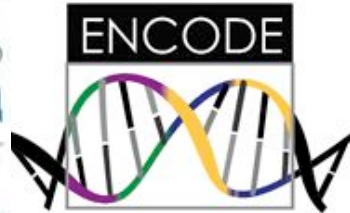
(Pushkarev *et al.*, 2009)

Sequencing a single human genome uses ~100 GB of compressed sequence data in billions of short reads.

~20 DVDs / genome



The DNA Data Tsunami



Use massive amounts of sequencing to explore the genetic origins of life



Our best (only) hope is to use many computers:

- Parallel Computing aka Cloud Computing
- Now your programs will crash on 1000 computers instead of just 1 😊



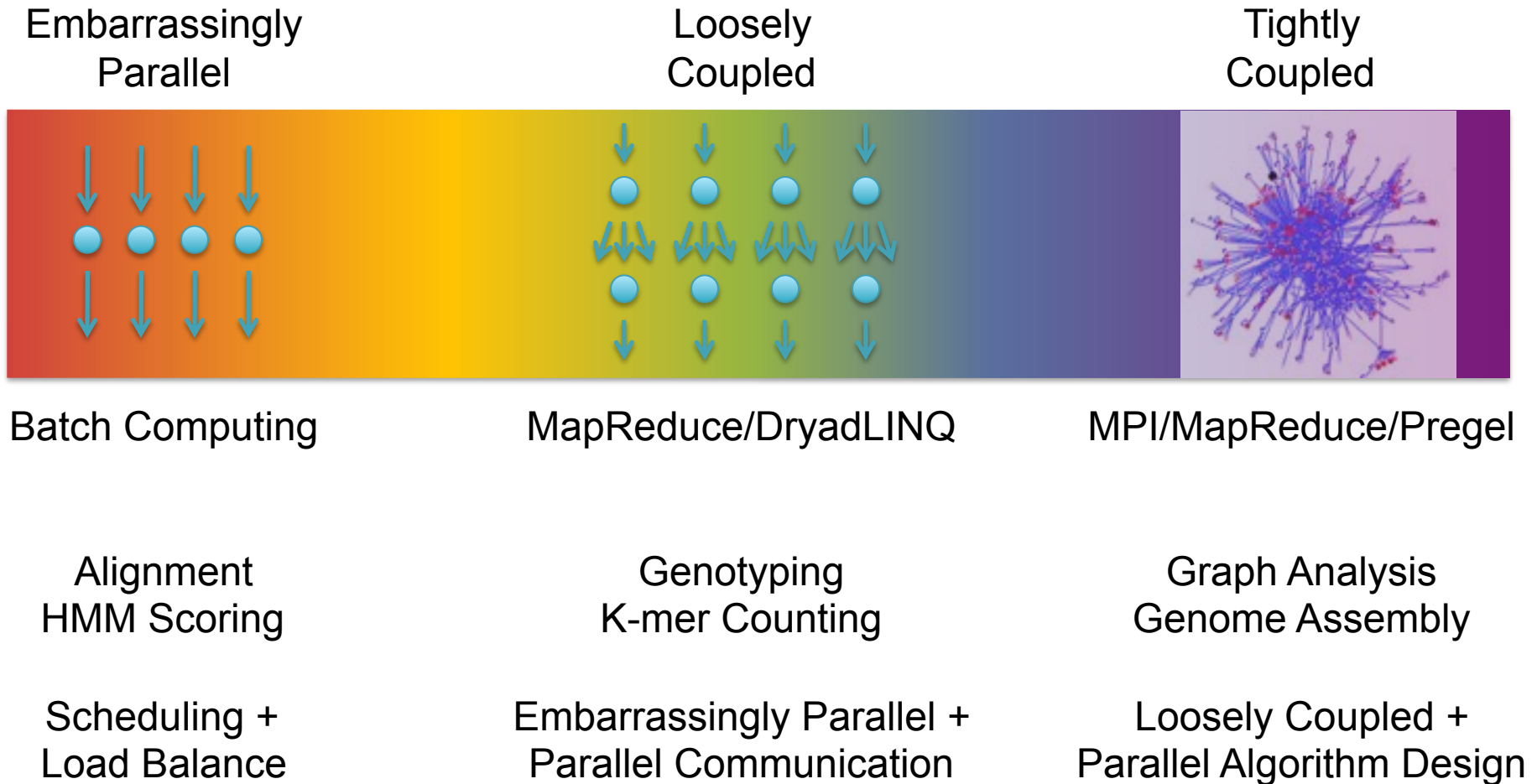
Amazon Web Services

<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

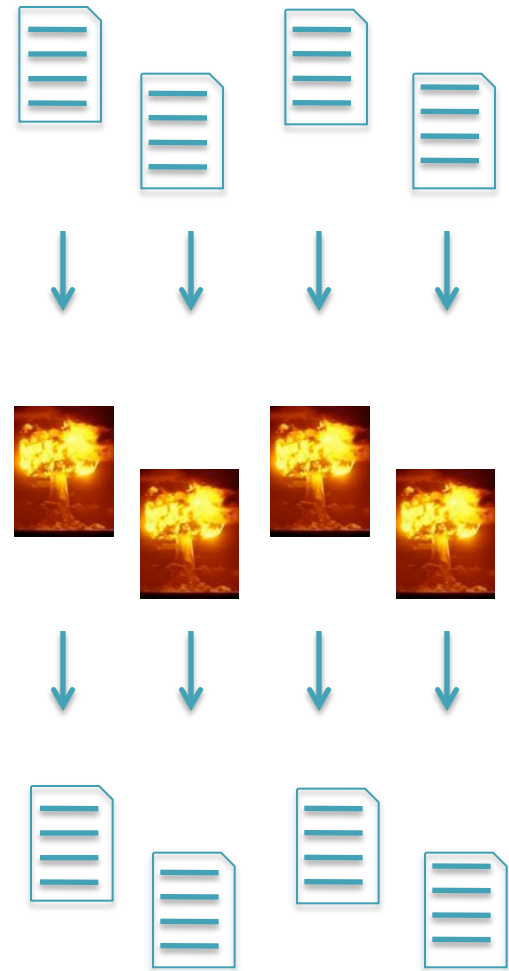


Cloud Computing Spectrum



Embarrassingly Parallel

- 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

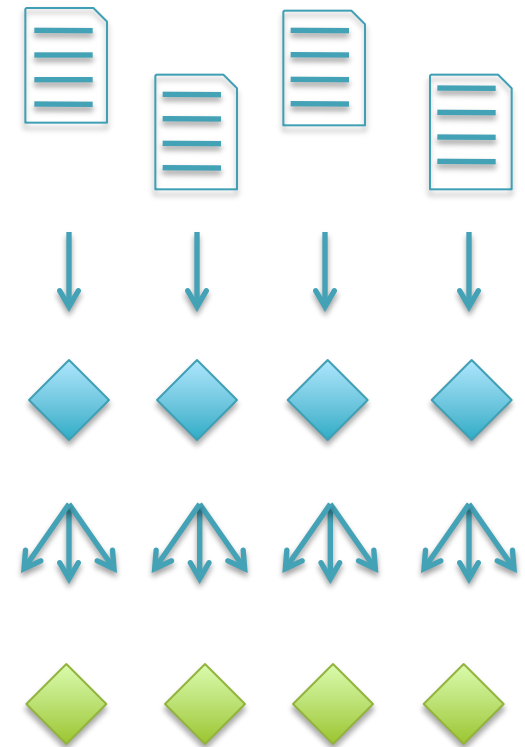


Elementary School Dance



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>

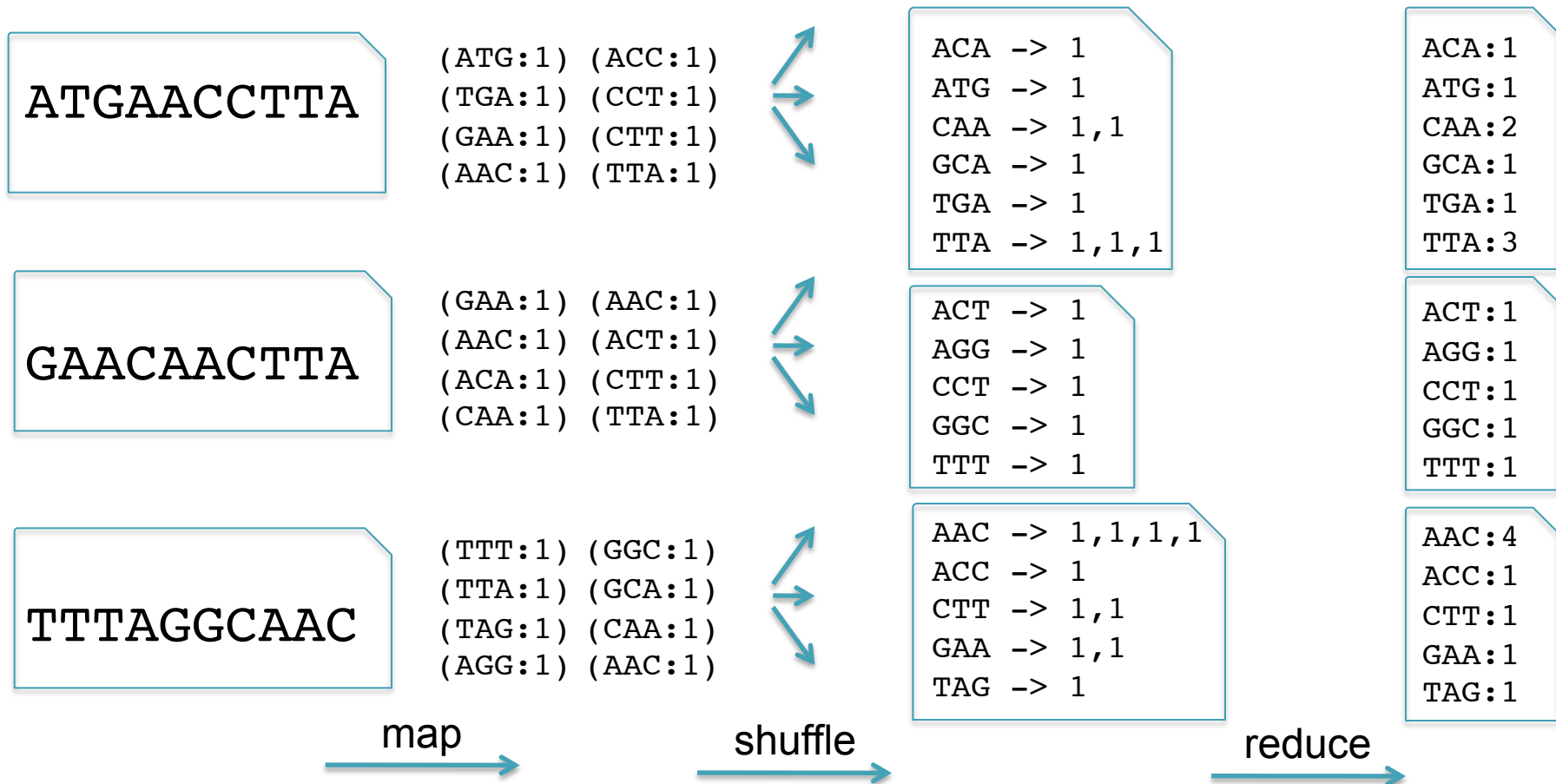
- MapReduce is the parallel distributed framework invented by Google for large data computations.
 - Data and computations are spread over thousands of computers, processing petabytes of data each day (Dean and Ghemawat, 2004)
 - Indexing the Internet, PageRank, Machine Learning, etc...
 - Hadoop is 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



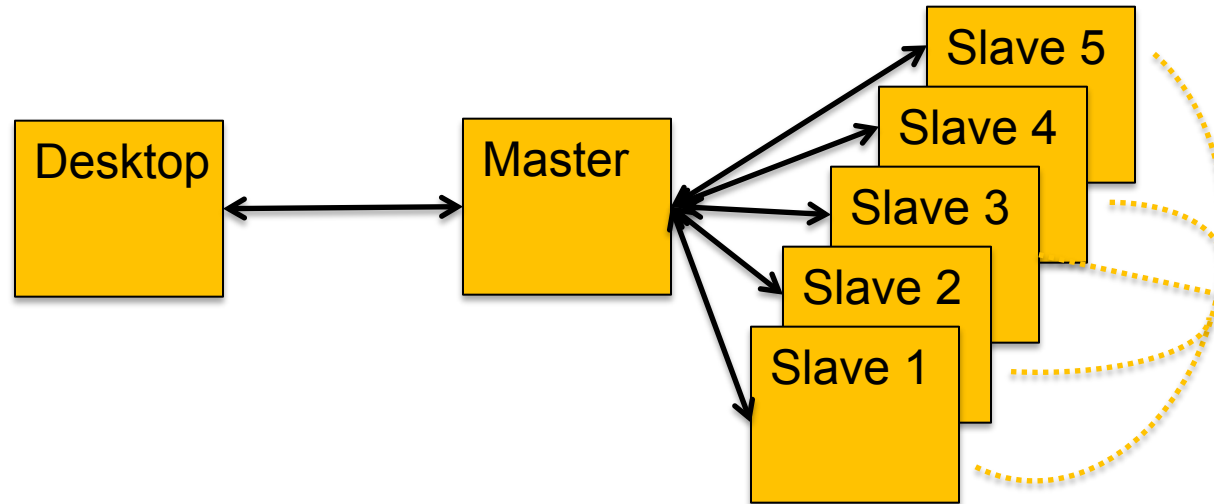
K-mer Counting

- 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

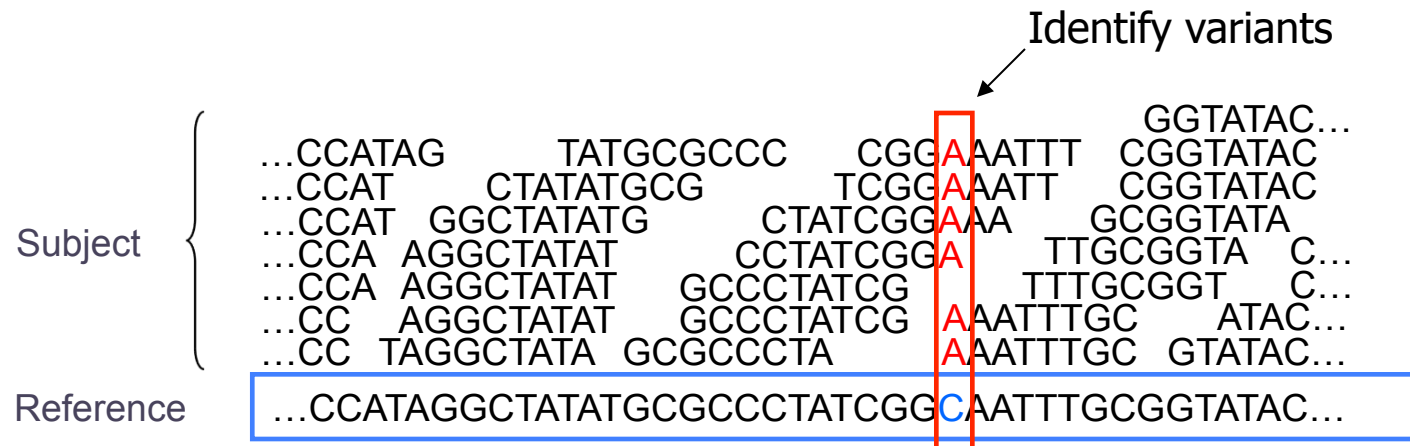


Hadoop Architecture



- 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

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

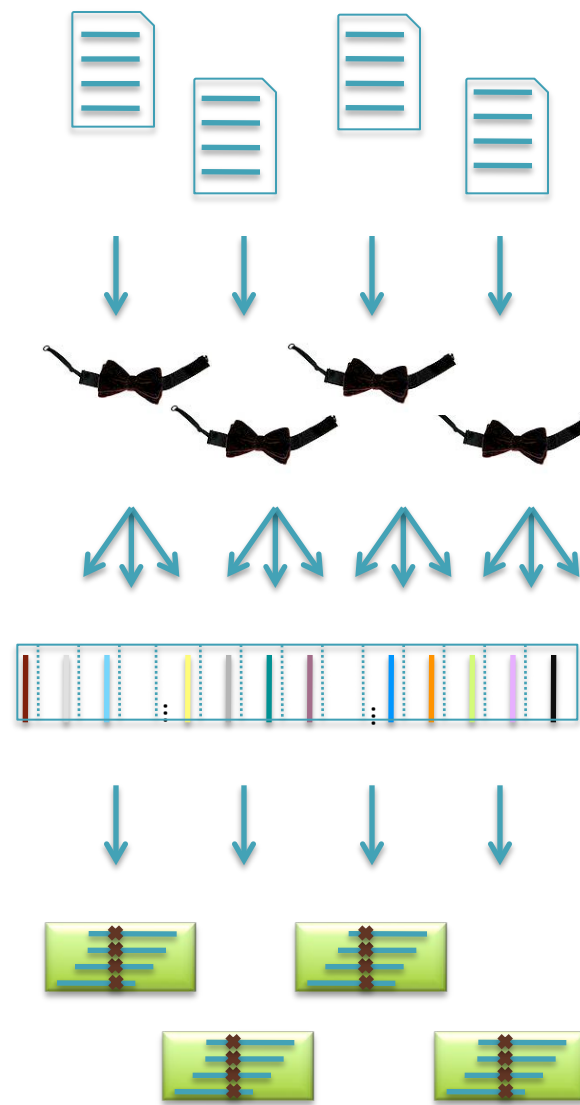
- Desperate need for scalable solutions
 - Single human requires >1,000 CPU hours / genome



Crossbow

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

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

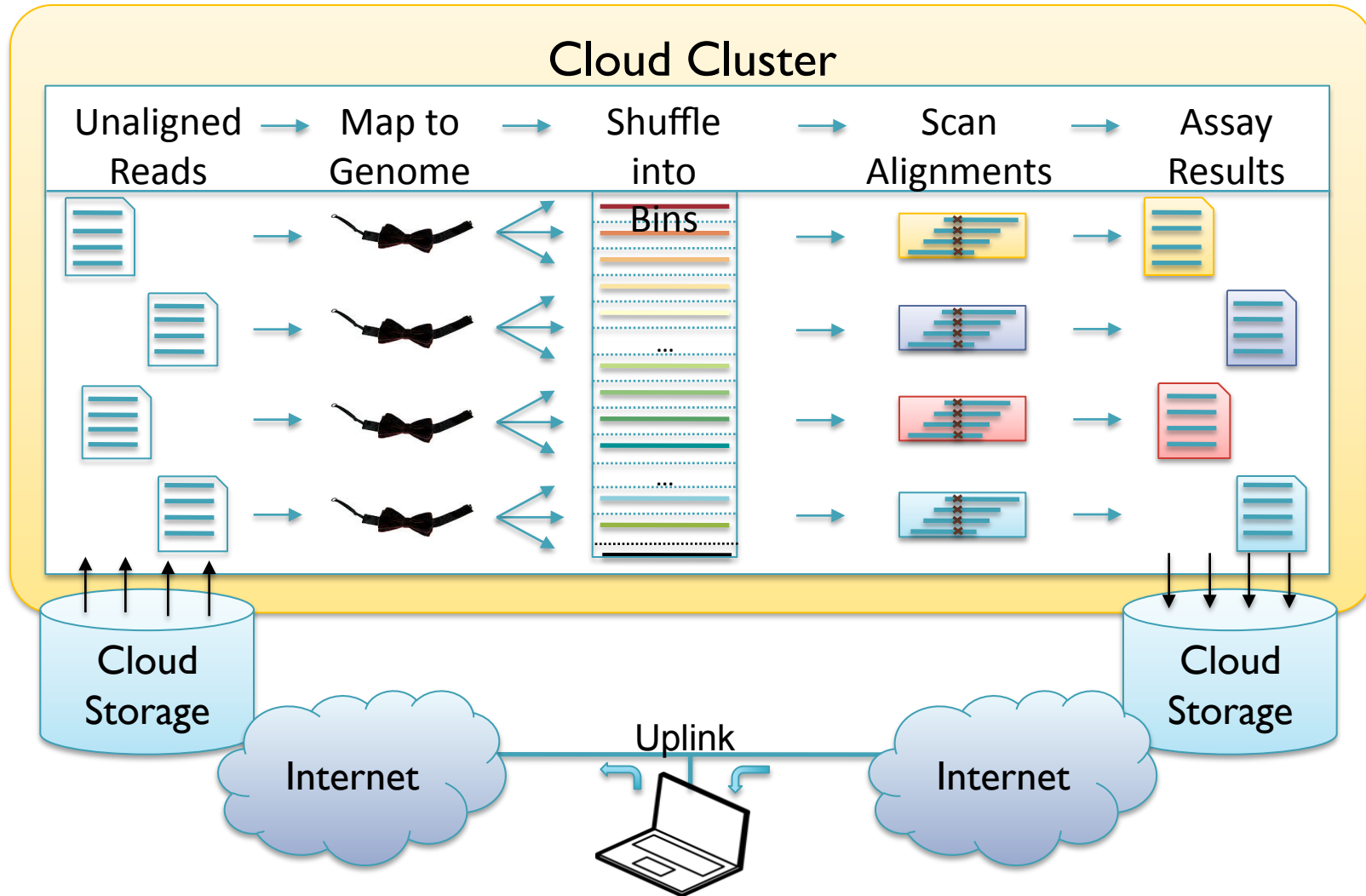
	Asian Individual Genome		
Data Loading	3.3 B reads	106.5 GB	\$10.65
Data Transfer	1h :15m	40 cores	\$3.40
Setup	0h : 15m	320 cores	\$13.94
Alignment	1h : 30m	320 cores	\$41.82
Variant Calling	1h : 00m	320 cores	\$27.88
End-to-end	4h : 00m		\$97.69

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

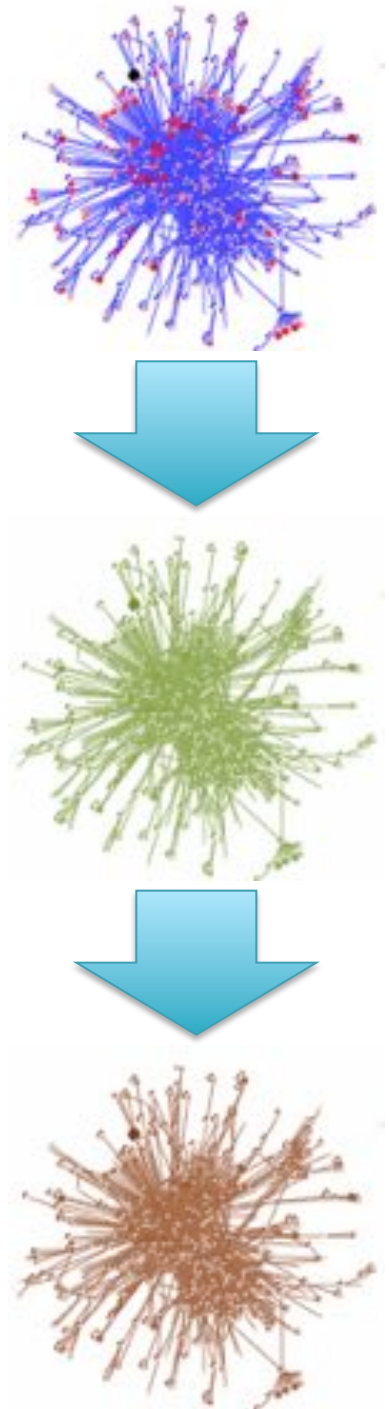


Cloud Computing and the DNA Data Race.

Schatz, MC, Langmead B, Salzberg SL (2010) *Nature Biotechnology* 28:691-693

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

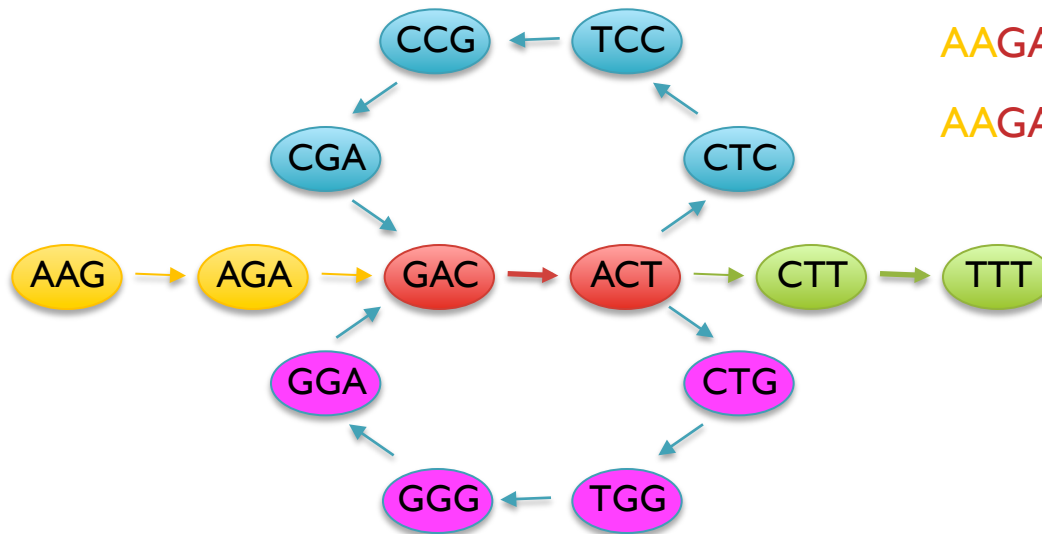


Short Read Assembly

Reads

AAGA
ACTT
ACTC
ACTG
AGAG
CCGA
CGAC
CTCC
CTGG
CTTT
...

de Bruijn Graph



Potential Genomes

AAGACTCCGACTGGGACTTT

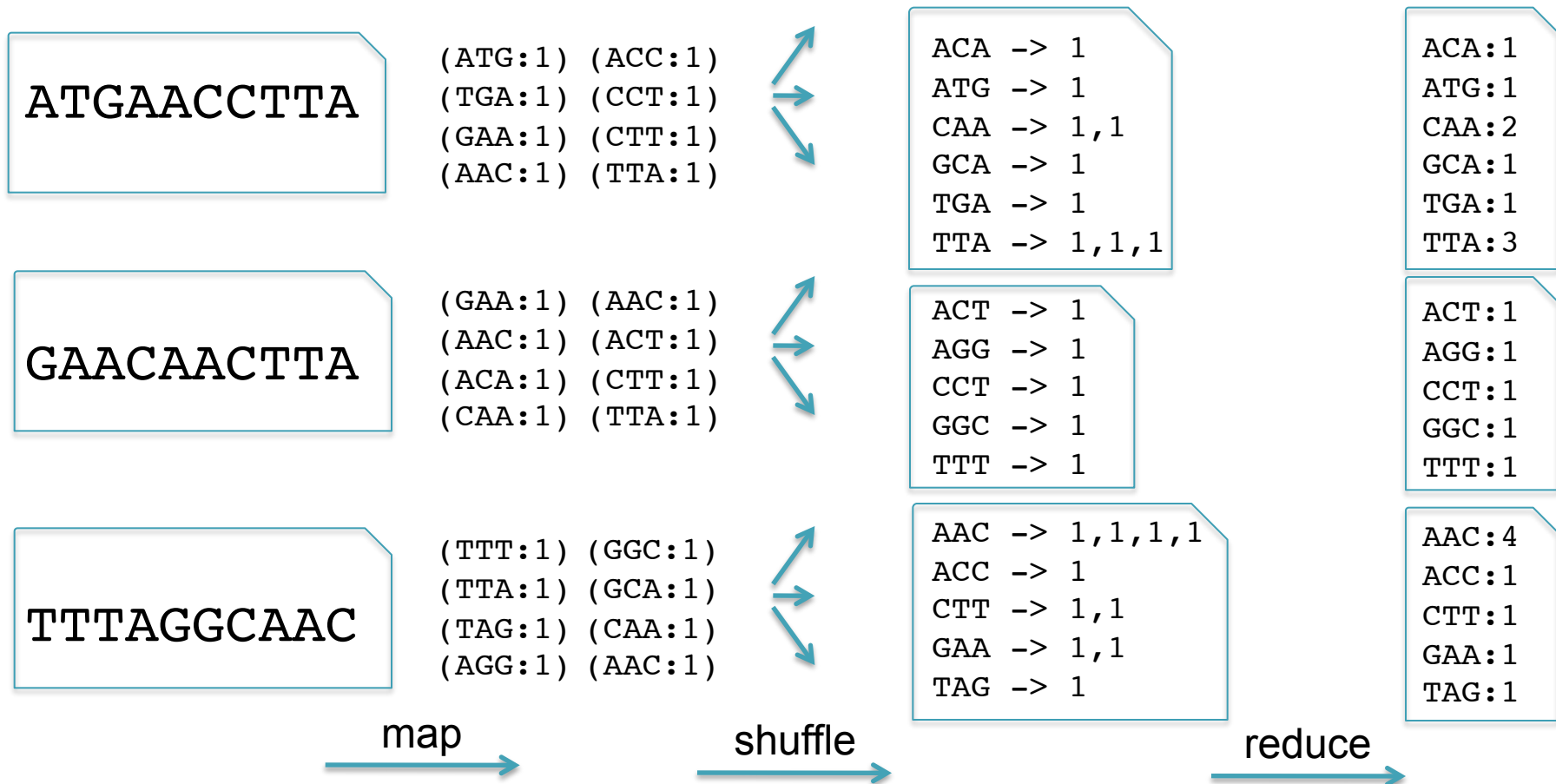
AAGACTGGGACTCCGACTTT

- 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

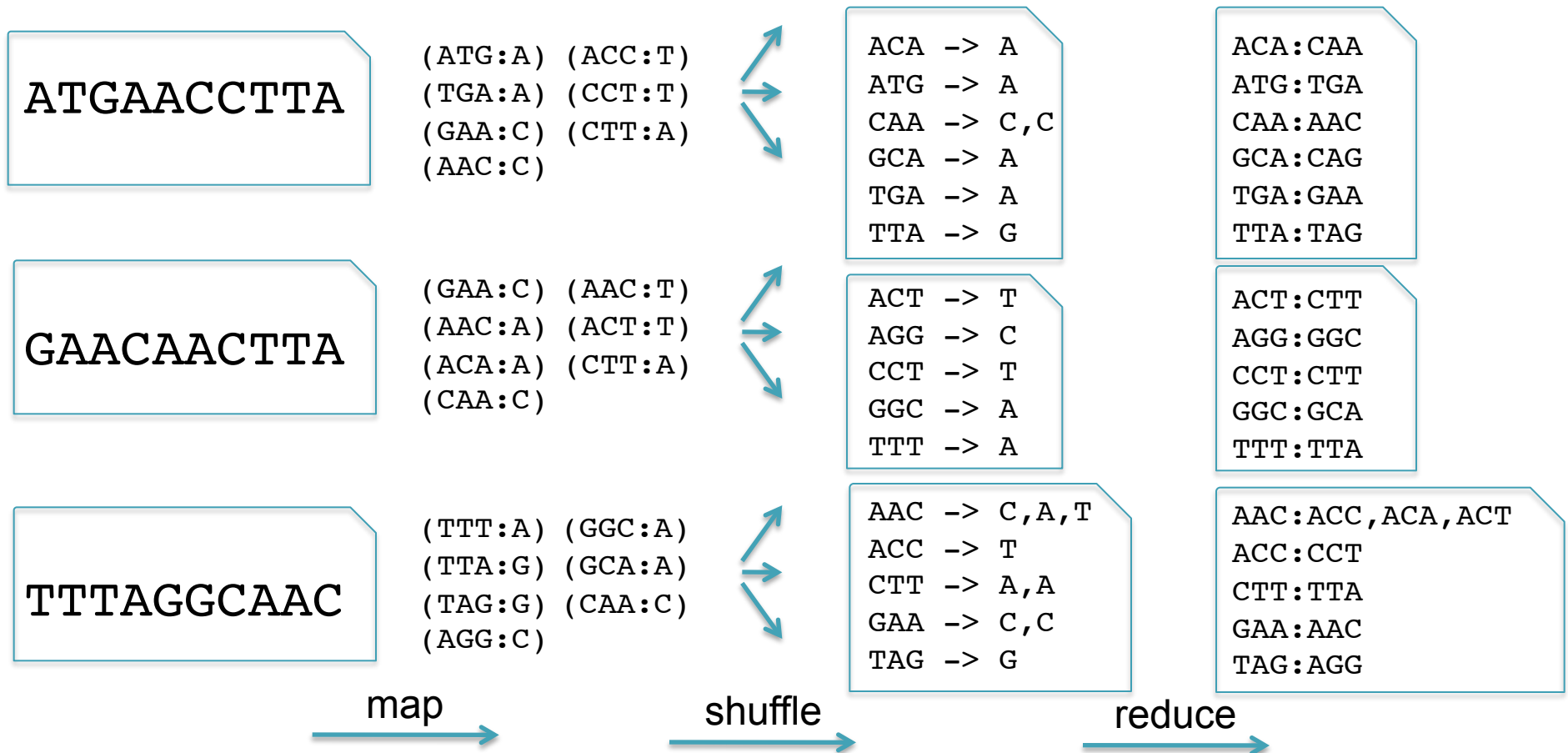
Map, Shuffle & Reduce
All Run in Parallel



Graph Construction

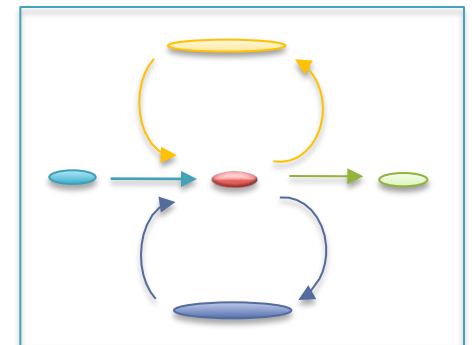
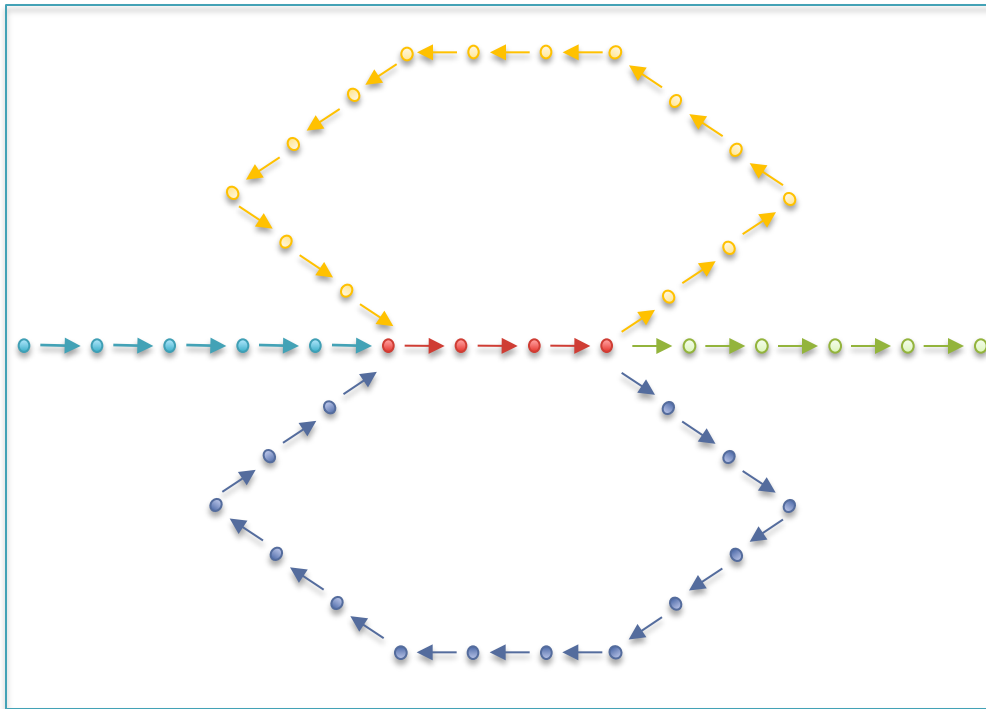
- Application developers focus on 2 (+1 internal) functions
 - **Map**: input → key:value pairs
 - **Shuffle**: Group together pairs with same key
 - **Reduce**: key, value-lists → output

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

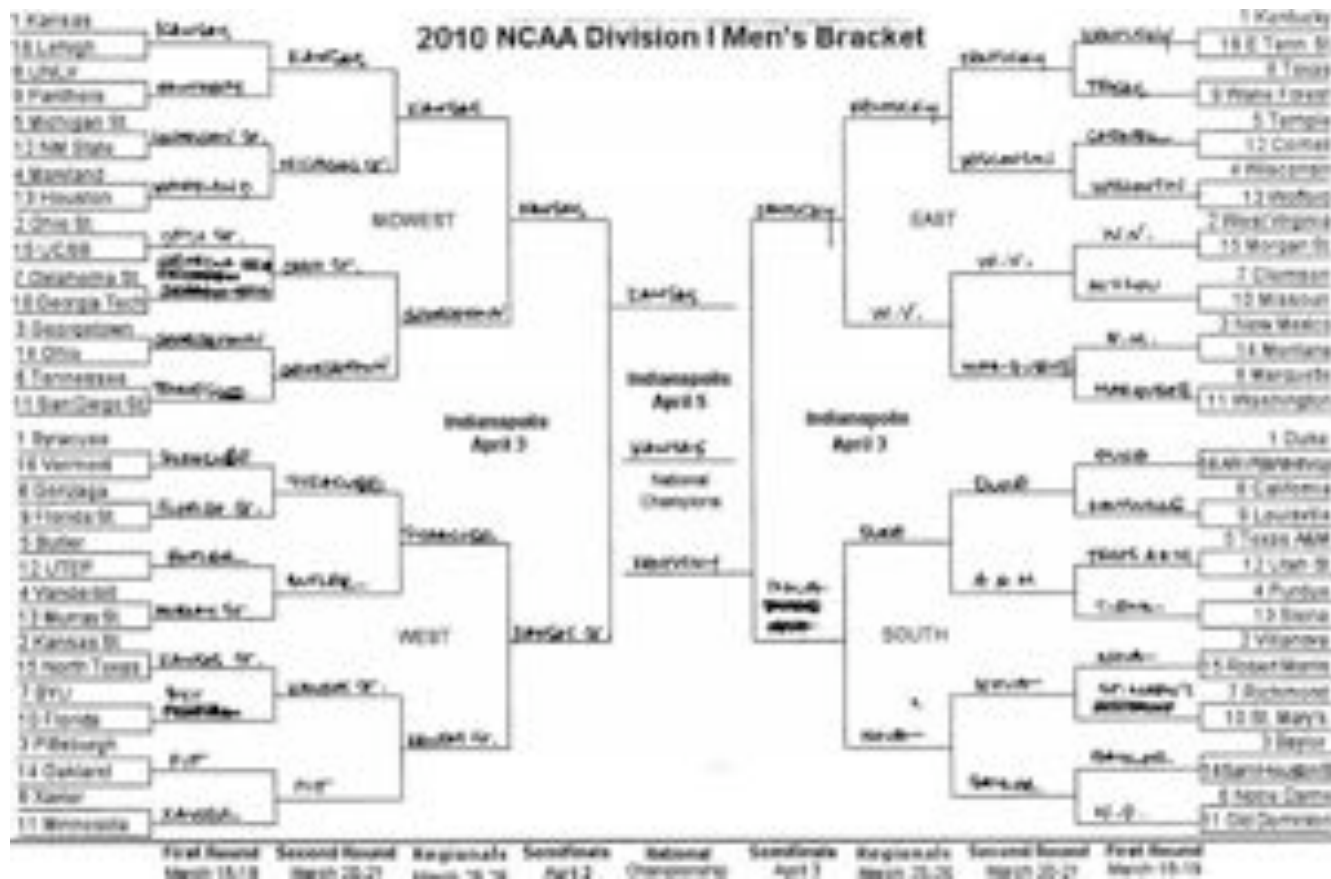


High School Dance



Warmup Exercise

- Who here was born closest to October 22?
 - You can only compare to 1 other person at a time



Find winner among 64 teams in just 6 rounds

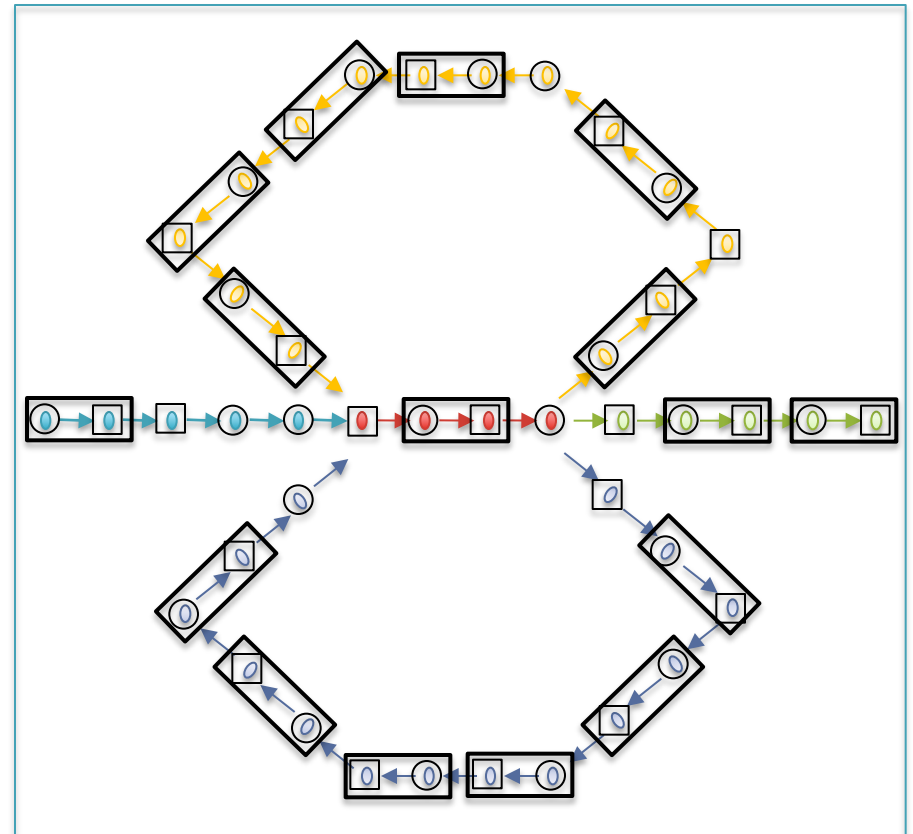
Fast Path Compression

Challenges

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

Randomized List Ranking

- Randomly assign \textcircled{H} / $\square T$ to each compressible node
- Compress $\textcircled{H} \rightarrow \square T$ links



Initial Graph: 42 nodes

Randomized Speed-ups in Parallel Computation.

Vishkin U. (1984) *ACM Symposium on Theory of Computation*. 230-239.

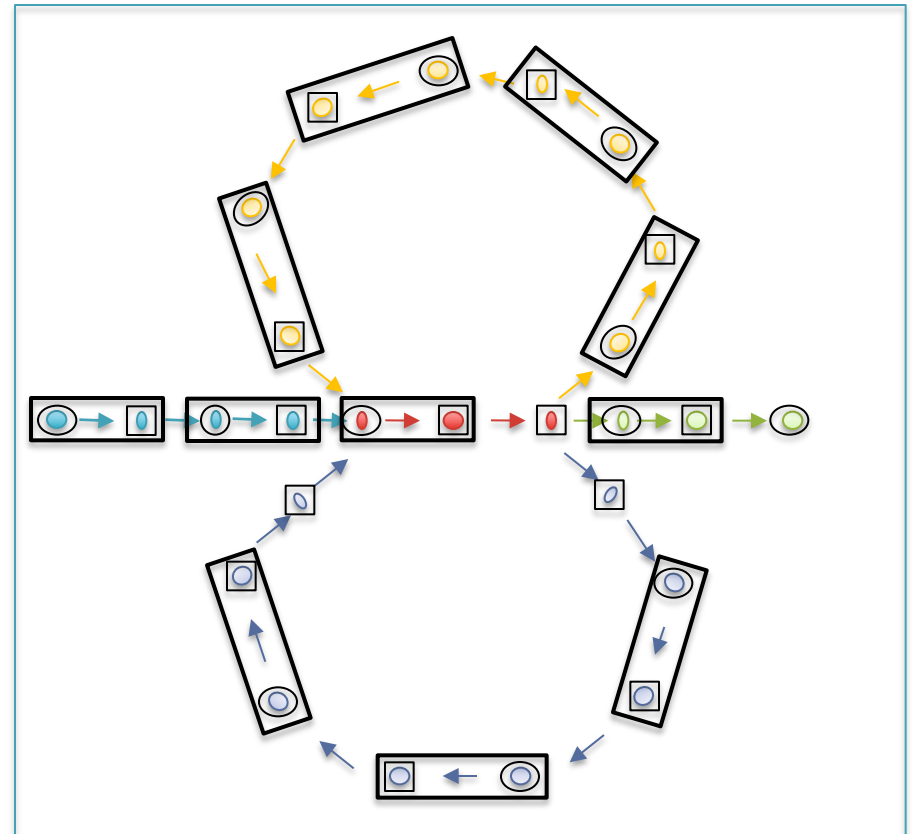
Fast Path Compression

Challenges

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

Randomized List Ranking

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



Round 1: 26 nodes (38% savings)

Randomized Speed-ups in Parallel Computation.

Vishkin U. (1984) *ACM Symposium on Theory of Computation*. 230-239.

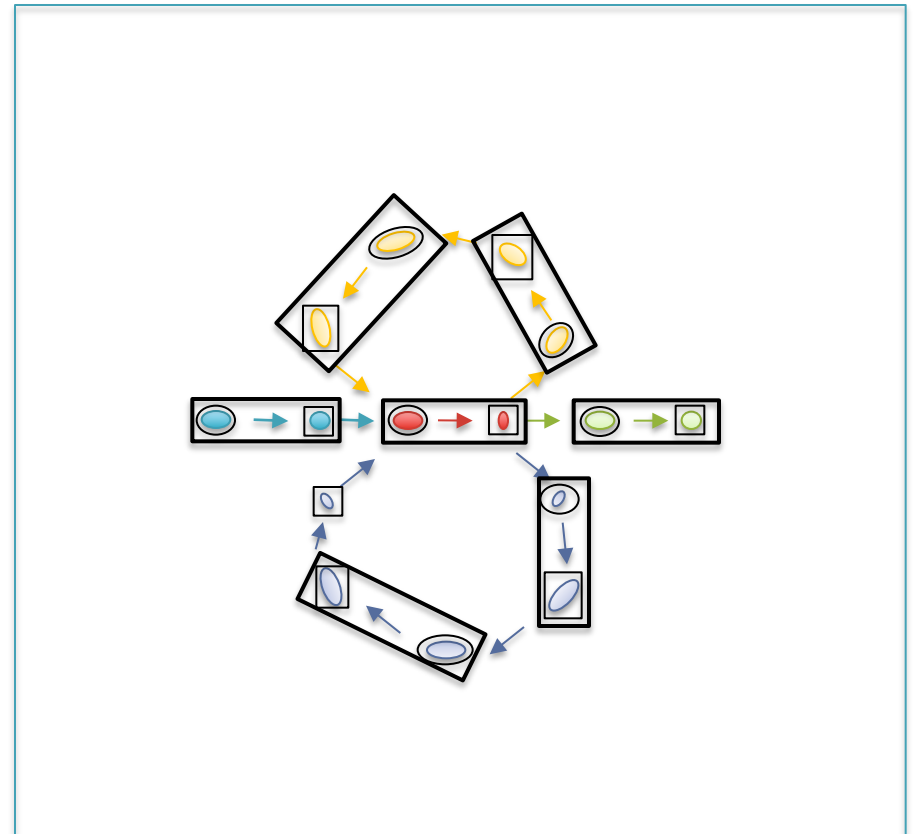
Fast Path Compression

Challenges

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

Randomized List Ranking

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



Round 2: 15 nodes (64% savings)

Randomized Speed-ups in Parallel Computation.

Vishkin U. (1984) *ACM Symposium on Theory of Computation*. 230-239.

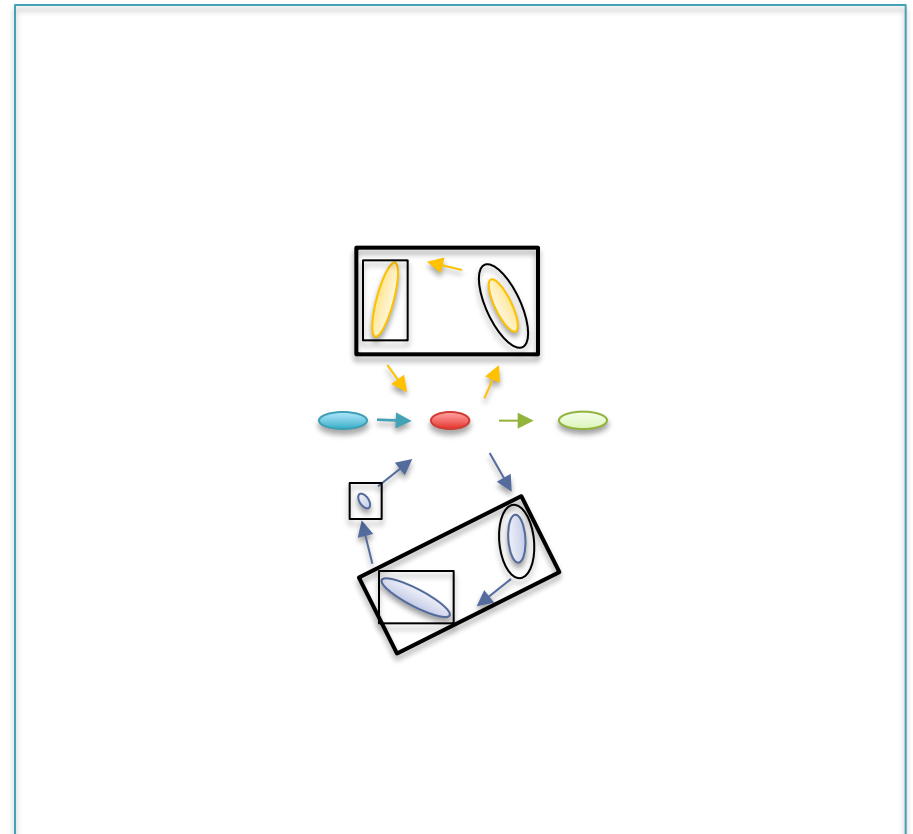
Fast Path Compression

Challenges

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

Randomized List Ranking

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



Round 2: 8 nodes (81% savings)

Randomized Speed-ups in Parallel Computation.

Vishkin U. (1984) *ACM Symposium on Theory of Computation*. 230-239.

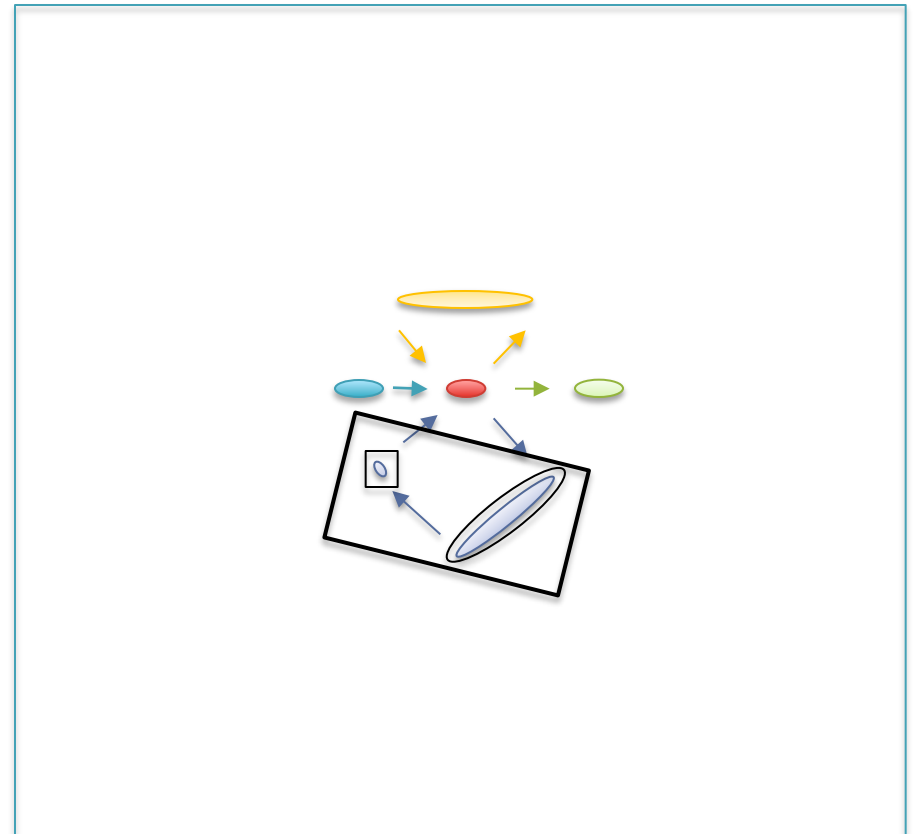
Fast Path Compression

Challenges

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

Randomized List Ranking

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

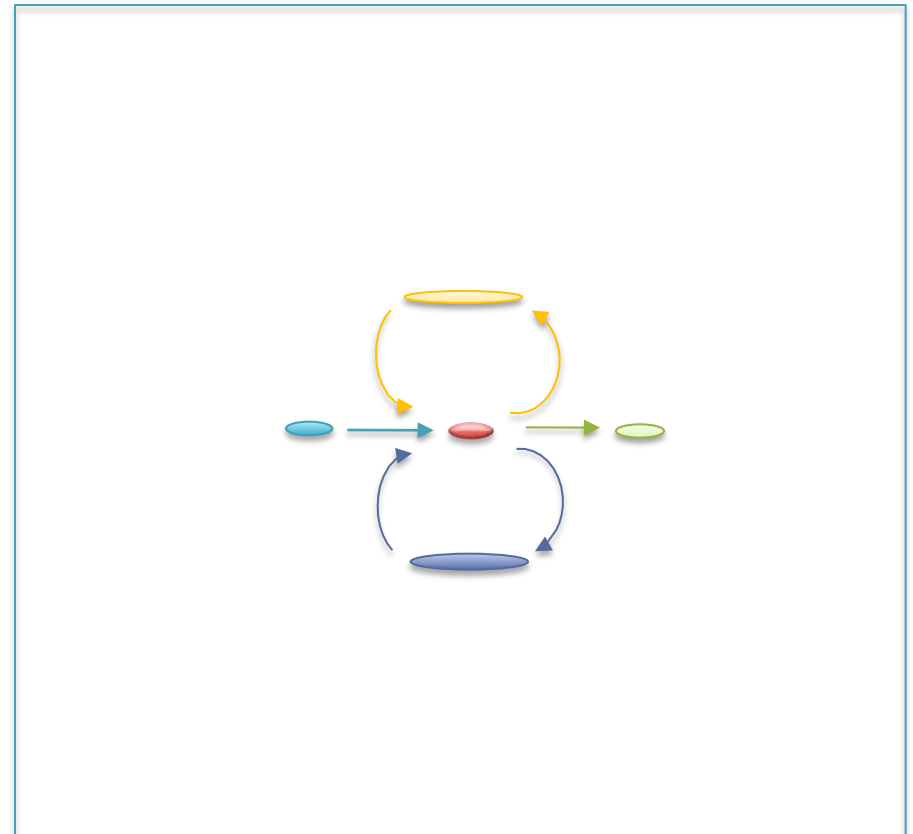
- Nodes stored on different computers
- Nodes can only access direct neighbors

Randomized List Ranking

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

Performance

- Compress all chains in $\log(S)$ rounds

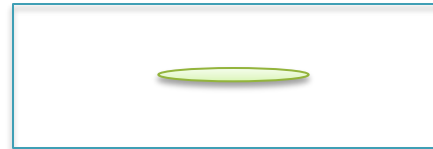


Round 4: 5 nodes (88% savings)

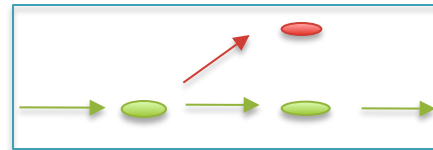
Randomized Speed-ups in Parallel Computation.

Vishkin U. (1984) *ACM Symposium on Theory of Computation*. 230-239.

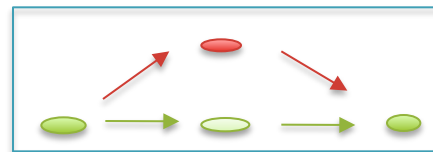
Node Types



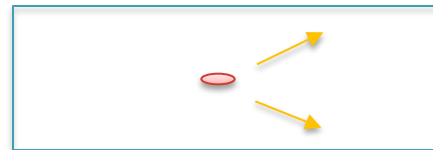
Isolated nodes (10%)



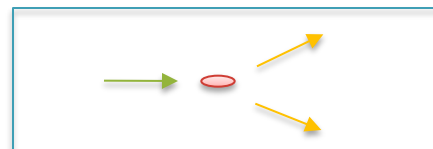
Tips (46%)



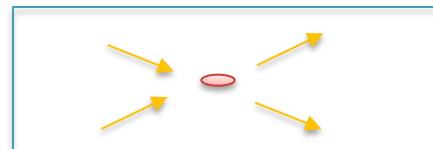
Bubbles/Non-branch (9%)



Dead Ends (.2%)



Half Branch (25%)



Full Branch (10%)

(Chaisson, 2009)

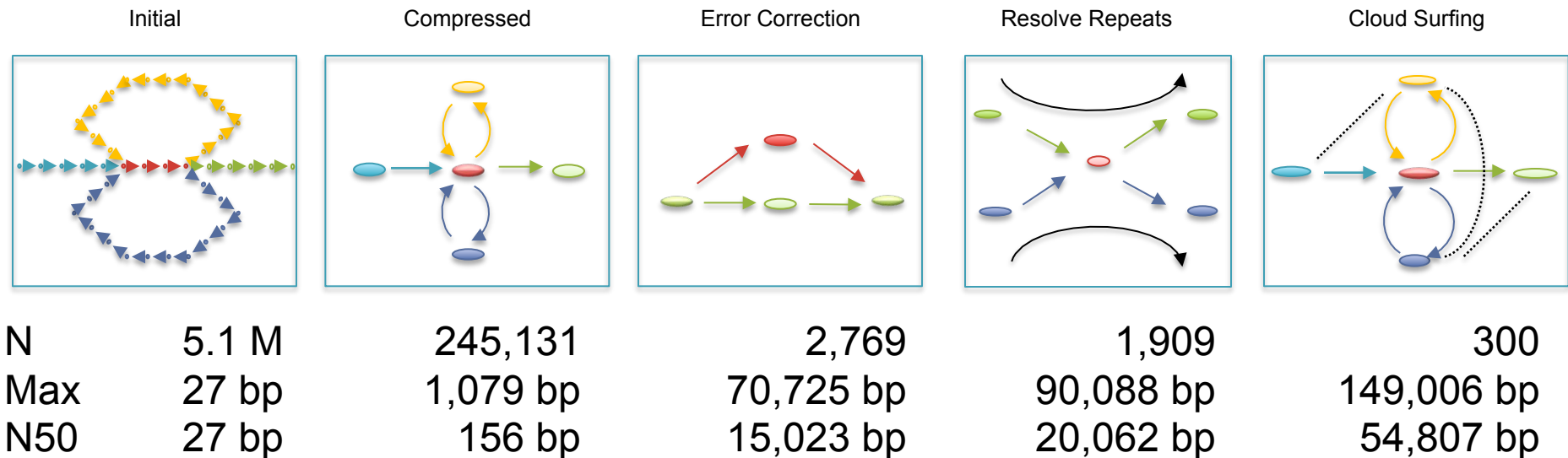
Contrail

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



Scalable Genome Assembly with MapReduce

- *Genome: E. coli* K12 MGI655, 4.6Mbp
- *Input: 20.8M* 36bp reads, 200bp insert (~150x coverage)
- *Preprocessor: Quality-Aware Error Correction*



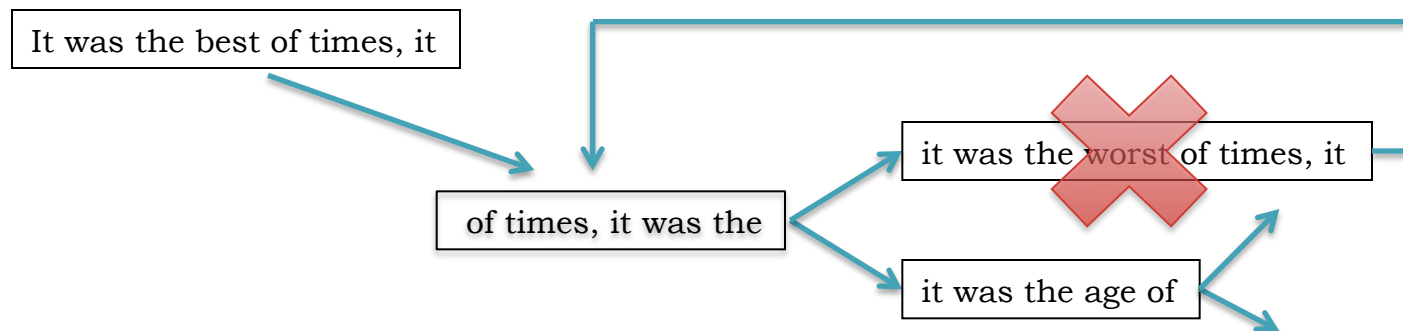
Assembly of Large Genomes with Cloud Computing.

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 \geq 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



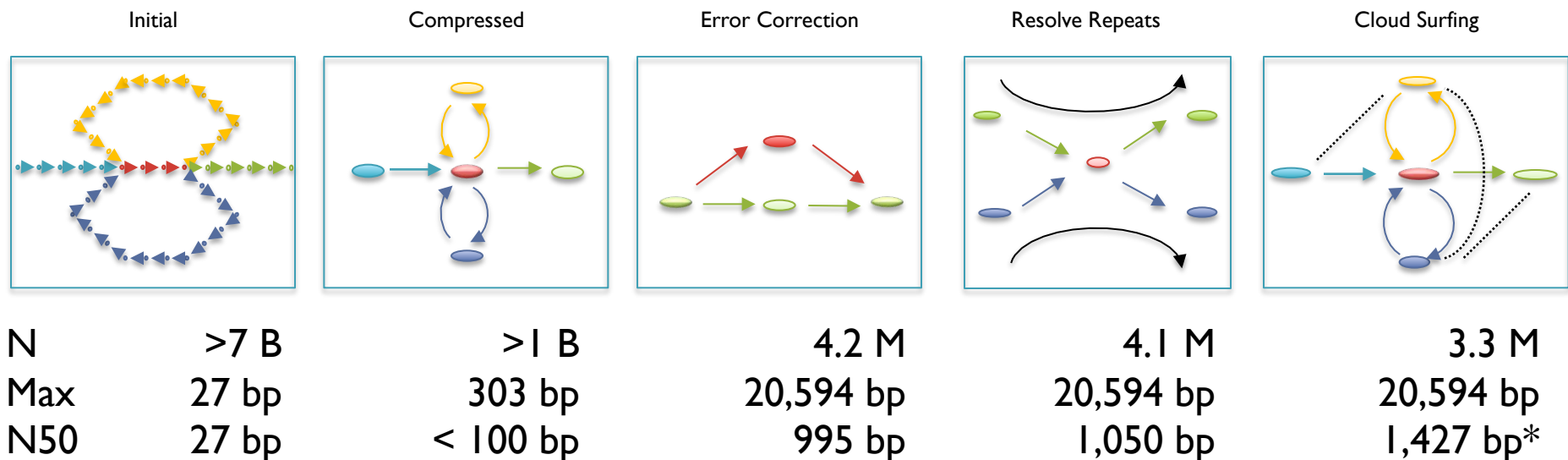
Contrail

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



De novo Assembly of the Human Genome

- *Genome*: African male NAI8507 (SRA000271, Bentley *et al.*, 2008)
- *Input*: 3.5B 36bp reads, 210bp insert (~40x coverage)

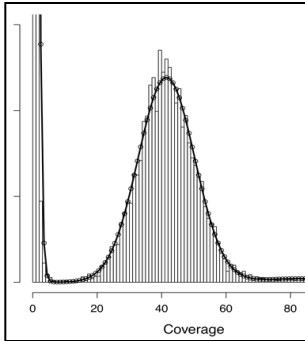


Assembly of Large Genomes with Cloud Computing.

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

Hadoop for NGS Analysis

Quake



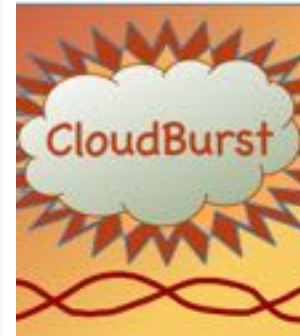
Quality-aware error correction of short reads

Correct 97.9% of errors with 99.9% accuracy

<http://www.cbcb.umd.edu/software/quake/>

(Kelley, Schatz, Salzberg, 2010*)

CloudBurst



Highly Sensitive Short Read Mapping with MapReduce

100x speedup mapping on 96 cores @ Amazon

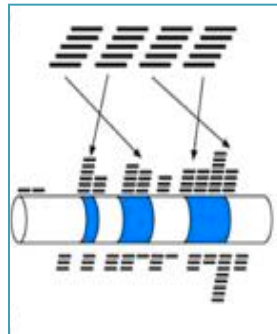
<http://cloudburst-bio.sf.net>

(Schatz, 2009)

Myrna

Cloud-scale differential gene expression for RNA-seq

Expression of 1.1 billion RNA-Seq reads in <2 hours for ~\$66



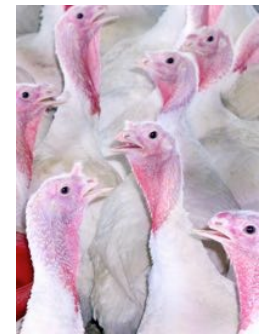
(Langmead, Hansen, Leek, 2010)

<http://bowtie-bio.sf.net/myrna/>

AMOS

Searching for SNPs in the Turkey Genome

Scan the de novo assembly to find 920k heterozygous alleles



(Dalloul et al, 2010)

<http://amos.sf.net>



Summary

- Surviving the data deluge means computing in parallel
 - Cloud computing is an attractive platform for large scale sequence analysis and computation
- Use the right tool for the job
 - Embarassingly parallel = Condor/Hadoop
 - Loosely coupled = Hadoop/Dyrad
 - Tightly coupled = MPI/Hadoop
- 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



Break

Outline

Part 1: Theory

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

Part 2: Practice

1. AWS Mini-Tutorial
2. Hadoop Mini-Tutorial



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



Other Services



A screenshot of the AWS website navigation menu. The top navigation bar includes links for AWS, Products, Developers, Community, Support, and Account. Below this, a grid of service categories is displayed, each with a list of specific services.

Category	Services
Compute	Amazon Elastic Compute Cloud (EC2) Amazon Elastic MapReduce Auto Scaling
Content Delivery	Amazon CloudFront
Database	Amazon SimpleDB Amazon Relational Database Service (RDS)
E-Commerce	Amazon Fulfillment Web Service (FWS)
Messaging	Amazon Simple Queue Service (SQS) Amazon Simple Notification Service (SNS)
Monitoring	Amazon CloudWatch
Networking	Amazon Virtual Private Cloud (VPC) Elastic Load Balancing
Payments & Billing	Amazon Flexible Payments Service (FPS) Amazon DevPay
Storage	Amazon Simple Storage Service (S3) Amazon Elastic Block Storage (EBS) AWS Import/Export
Support	AWS Premium Support
Web Traffic	Alexa Web Information Service Alexa Top Sites
Workforce	Amazon Mechanical Turk

close x

Everything you need to run a large scale service & analysis suite in the clouds.

Cloud Solutions for DNA Sequence Analysis

- Rapid & elastic deployment of vast computation
 - Accessible, Reproducible, Reliable, Collaborative
- Why not?
 - Transfer: 200 GBs takes 1 hr – 2 weeks
 - Privacy & security: Excellent... with care
 - Expertise: Computing on 1000 cores is complex
 - Price: The meter is always running
 - Who will be on the line for making it all work?
 - Psychological and Institutional paradigm shift

EC2 Architecture

- Very large pool of machines
 - Effectively infinite resources
 - High-end servers with many cores and many GB RAM
- Machines run in a virtualized environment
 - Amazon can subdivide large nodes into smaller instances
 - You are 100% protected from other users on the machine
 - You get to pick the operating system, all installed software



Instance Types

Type	Price / hr	CPU	Resources
Micro <i>Web service</i>	2¢	1 core @ 1 ECU	.6 GB RAM 10 GB Disk
Standard <i>Light Tasks</i>	8.5¢	1 core @ 1 ECU	1.7 GB RAM 160 GB Disk
Extra Large <i>Mapping w/BWA</i>	68¢	4 core @ 2 ECU	15 GB RAM 1.7 TB Disk
HighCPU XL <i>Mapping w/Crossbow</i>	68¢	8 cores @ 2.5 ECU	7 GB RAM 1.7 TB Disk
HighMem Quad XL <i>Assembly</i>	\$2.00	8 cores @ 3.25 ECU	68.4 GB RAM 1.7 TB Disk

ECU = EC2 Compute Unit. Approximately 1.0 – 1.2 GHz Intel Xeon from 2007

Reserved Instances make it cheaper for consistent use.
Pay for what you use, rounded UP to the next full hour

Amazon Machine Images



- A few Amazon sponsored images
 - Suse Linux, Windows
- Many Community Images & Appliances
 - Crossbow: Hadoop, Bowtie, SOAPsnp
 - CloudBioLinux.com: Appliance for Genomics
- Build you own
 - Completely customize your environment
 - You results could be totally reproducible

Amazon S3

- S3 provides persistent storage for large volumes of data
 - Very high speed connection from S3 to EC2 compute nodes
 - Public data sets include `s3://1000genomes`
- Tiered pricing by volume
 - Pricing starts at 15¢ / GB / month
 - 5.5¢ / GB / month for over 5 PB
 - Pay for transfer in and out of Amazon
- Import/Export service for large volumes
 - FedEx your drives to Amazon



Getting Started

<http://docs.amazonwebservices.com/AWSEC2/latest/GettingStartedGuide/>

The screenshot shows a web browser window titled "Amazon Elastic Compute Cloud" with the URL <http://docs.amazonwebservices.com/AWSEC2/latest/GettingStartedGuide/>. The page header includes the Amazon logo and "amazon web services". The main content area is titled "Amazon Elastic Compute Cloud Getting Started Guide (API Version 2010-08-31)". A left sidebar contains a table of contents with items like "Get Started with EC2", "Sign Up for EC2", "Launch an Instance", "Connect to Your Linux/UNIX Instance", "Connect to Your Windows Instance", "Terminate Your Instance", "Where Do I Go from Here?", "Please Provide Feedback", and "About This Guide". The main content area has a "Documentation Feedback" link and a "Welcome" message. The main heading is "Get Started with EC2". Below this is a paragraph explaining that Amazon Elastic Compute Cloud (Amazon EC2) is a web service for launching and managing Linux/UNIX and Windows server instances in Amazon's data centers, and that users should follow the tasks in the diagram using the AWS Management Console. A flowchart diagram shows the process: "Sign up for EC2" leads to "Launch instance", which then branches into "Connect to Linux/UNIX instance" and "Connect to Windows instance", both of which lead to "Terminate instance". Below the diagram is a paragraph stating that the guide walks through launching and connecting to the first Amazon EC2 instance, and a "Get Started" button with a right-pointing arrow.

Amazon Elastic Compute Cloud
Getting Started Guide (API Version 2010-08-31)

Documentation Feedback

Welcome

Get Started with EC2

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that enables you to launch and manage Linux/UNIX and Windows server instances in Amazon's data centers. You can get started with Amazon EC2 by following the tasks shown in the following diagram. You'll primarily use the AWS Management Console, a point-and-click web-based interface.

```
graph LR; A[Sign up for EC2] --> B[Launch instance]; B --> C[Connect to Linux/UNIX instance]; B --> D[Connect to Windows instance]; C --> E[Terminate instance]; D --> E;
```

This guide walks you through launching and connecting to your first Amazon EC2 instance. To start, click the following **Get Started** button.

[Get Started](#)

Signing Up

The screenshot shows the Amazon Web Services website in a browser window. The browser's address bar displays `http://aws.amazon.com/`. The page features the AWS logo and navigation links for **AWS**, **Products**, **Developers**, **Community**, **Support**, and **Account**. At the top right, there are links for **Sign in to the AWS Management Console** and **Create an AWS Account**, along with a language selector set to **English**.

The main content area includes a promotional banner for **Introducing Amazon Simple Notification Service**, with the text: "Enable applications, end-users, and devices to instantly send and receive notifications from the cloud." Below this is a "Learn More..." link and a diagram showing a cloud with arrows pointing to various devices and services.

On the right side, there is a prominent call to action: **Sign up for a free Amazon Web Services Account**, accompanied by a **Sign Up Now** button.

The **Get Started** section lists several resources for **Business Managers**, including:

- Solutions & Use Cases
- Security Center
- Economics Center
- Case Studies
- Service Health Dashboard
- Solution Providers

The **News & Events** section is divided into **What's New?**, **Media Coverage**, and **Upcoming Events**. The **What's New?** tab is active, showing a list of recent updates:

Date	Event/Update
Oct 14, 2010	Amazon Elastic Load Balancing Adds Support for HTTPS
Oct 12, 2010	AWS Management Console Adds Support for Amazon SNS
Oct 05, 2010	Amazon EC2 Running SUSE Linux Now Available
Oct 05, 2010	Announcing Read Replicas, Lower High Memory DB Instance Prices for Amazon RDS
Sep 29, 2010	Announcing the AWS SDK for PHP
Sep 21, 2010	Oracle Certifies Enterprise Software on Amazon EC2

At the bottom of the news section, there is an **RSS** feed icon and a **View all** link.

AWS Management Console

The screenshot displays the AWS Management Console interface for the Amazon EC2 service. The browser address bar shows the URL `https://console.aws.amazon.com/ec2/home`. The page header includes navigation links for AWS, Products, Developers, Community, Support, and Account, along with a user greeting "Welcome, CBCB" and links for Settings and Sign Out.

The main navigation bar features tabs for various AWS services: Amazon S3, Amazon EC2 (selected), Amazon VPC, Amazon Elastic MapReduce, Amazon CloudFront, Amazon RDS, and Amazon SNS.

The left sidebar contains a "Navigation" menu with a "Region" dropdown set to "US East". The menu items are categorized as follows:

- INSTANCES
 - EC2 Dashboard
 - Instances
 - Spot Requests
- IMAGES
 - AMIs
 - Bundle Tasks
- ELASTIC BLOCK STORE
 - Volumes
 - Snapshots
- NETWORKING & SECURITY
 - Elastic IPs
 - Security Groups
 - Placement Groups
 - Load Balancers
 - Key Pairs

The main content area is titled "Amazon EC2 Console Dashboard" and is divided into several sections:

- Getting Started:** A yellow box with the text "To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance." and a prominent "Launch Instance" button. A note below states: "Note: Your instances will launch in the US East (Virginia) region."
- My Resources:** A summary of resources in the US East (Virginia) region, including:
 - 0 Running Instances
 - 0 Elastic IPs
 - 0 EBS Volumes
 - 0 EBS Snapshots
 - 2 Key Pairs
 - 27 Security Groups
 - 0 Load Balancers
 - 0 Placement Groups
- Service Health:** A table showing the current status of the Amazon EC2 service in the US East - N. Virginia region.

Current Status	Details
	Amazon EC2 (US East - N. Virginia) Service is operating normally. View complete service health details

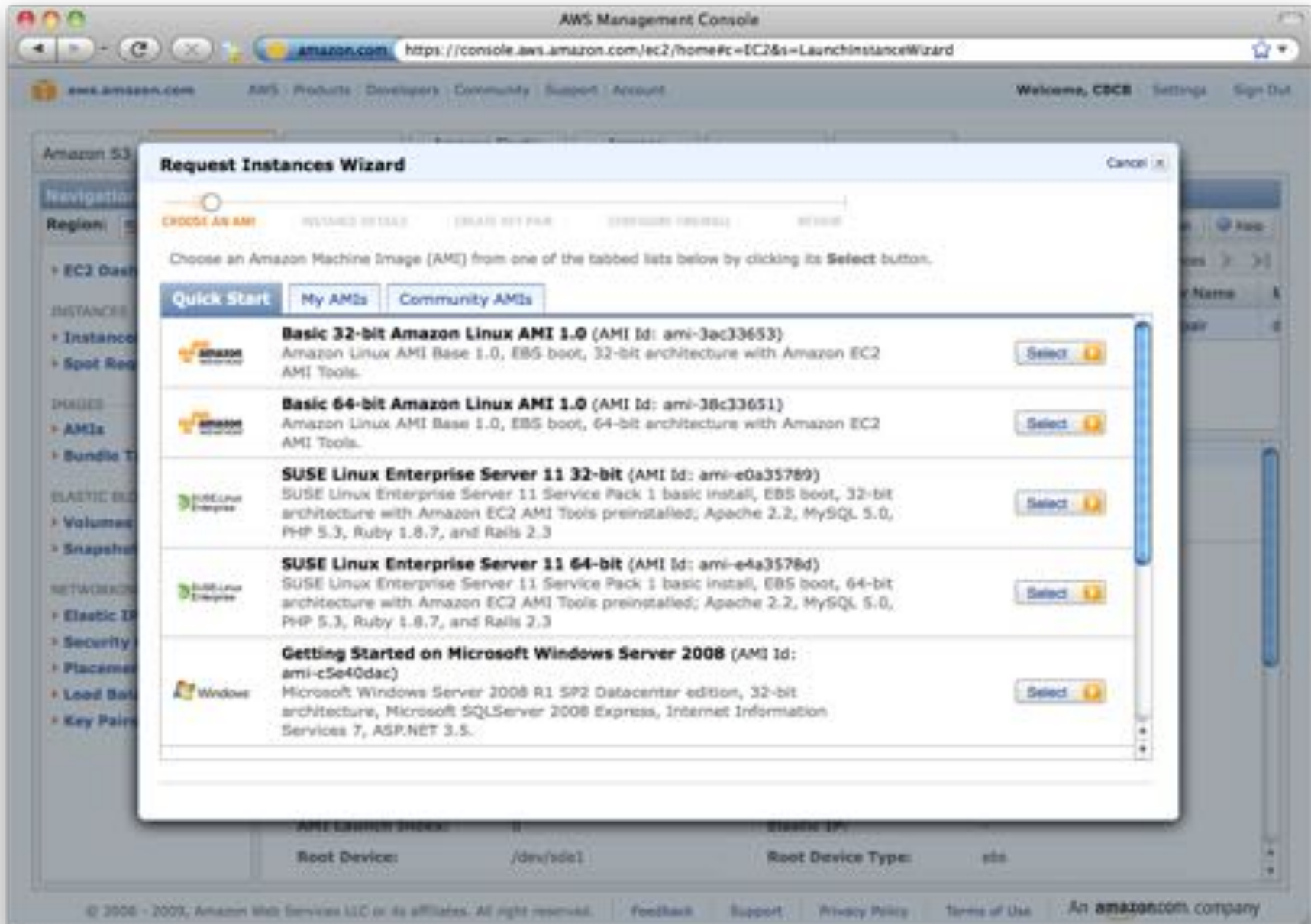
- Related Links:** Documentation, All EC2 Resources, Forums, Feedback, Report an Issue.

At the bottom of the page, the footer contains copyright information: "© 2008 - 2009, Amazon Web Services LLC or its affiliates. All rights reserved." and links for Feedback, Support, Privacy Policy, Terms of Use, and the Amazon logo with the text "An amazon.com company".

Running your First Cloud Analysis

1. Pick your AMI
 - Machine Image: Operating System & Tools
2. Pick your instance type & quantity
 - Micro - High-Memory Quadruple Extra Large
3. Pick your credentials
 - SSH Keys
4. Configure your Firewall
 - Protect your servers
5. Launch!

I. Pick your AMIs



The screenshot shows the AWS Management Console interface. The main window is titled "Request Instances Wizard" and is currently on the "CHOOSE AN AMI" step. The wizard has a progress bar at the top with five steps: "CHOOSE AN AMI", "INSTANCED PROFILE", "ELASTIC NET PAIR", "EBS VOLUME PROVISIONING", and "KEYPAIR".

Below the progress bar, there is a instruction: "Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its Select button." There are three tabs: "Quick Start", "My AMIs", and "Community AMIs". The "Quick Start" tab is selected.

The list of AMIs includes:

- Basic 32-bit Amazon Linux AMI 1.0** (AMI Id: ami-3ac33653)
Amazon Linux AMI Base 1.0, EBS boot, 32-bit architecture with Amazon EC2 AMI Tools.
- Basic 64-bit Amazon Linux AMI 1.0** (AMI Id: ami-38c33651)
Amazon Linux AMI Base 1.0, EBS boot, 64-bit architecture with Amazon EC2 AMI Tools.
- SUSE Linux Enterprise Server 11 32-bit** (AMI Id: ami-e0a35789)
SUSE Linux Enterprise Server 11 Service Pack 1 basic install, EBS boot, 32-bit architecture with Amazon EC2 AMI Tools preinstalled; Apache 2.2, MySQL 5.0, PHP 5.3, Ruby 1.8.7, and Rails 2.3
- SUSE Linux Enterprise Server 11 64-bit** (AMI Id: ami-e4a3578d)
SUSE Linux Enterprise Server 11 Service Pack 1 basic install, EBS boot, 64-bit architecture with Amazon EC2 AMI Tools preinstalled; Apache 2.2, MySQL 5.0, PHP 5.3, Ruby 1.8.7, and Rails 2.3
- Getting Started on Microsoft Windows Server 2008** (AMI Id: ami-c5e40dac)
Microsoft Windows Server 2008 R1 SP2 Datacenter edition, 32-bit architecture, Microsoft SQLServer 2008 Express, Internet Information Services 7, ASP.NET 3.5.

At the bottom of the wizard, there are fields for "AMI Launch Check" and "Instance Profile". The "AMI Launch Check" field shows "Root Device: /dev/sda1" and "Root Device Type: ebs".

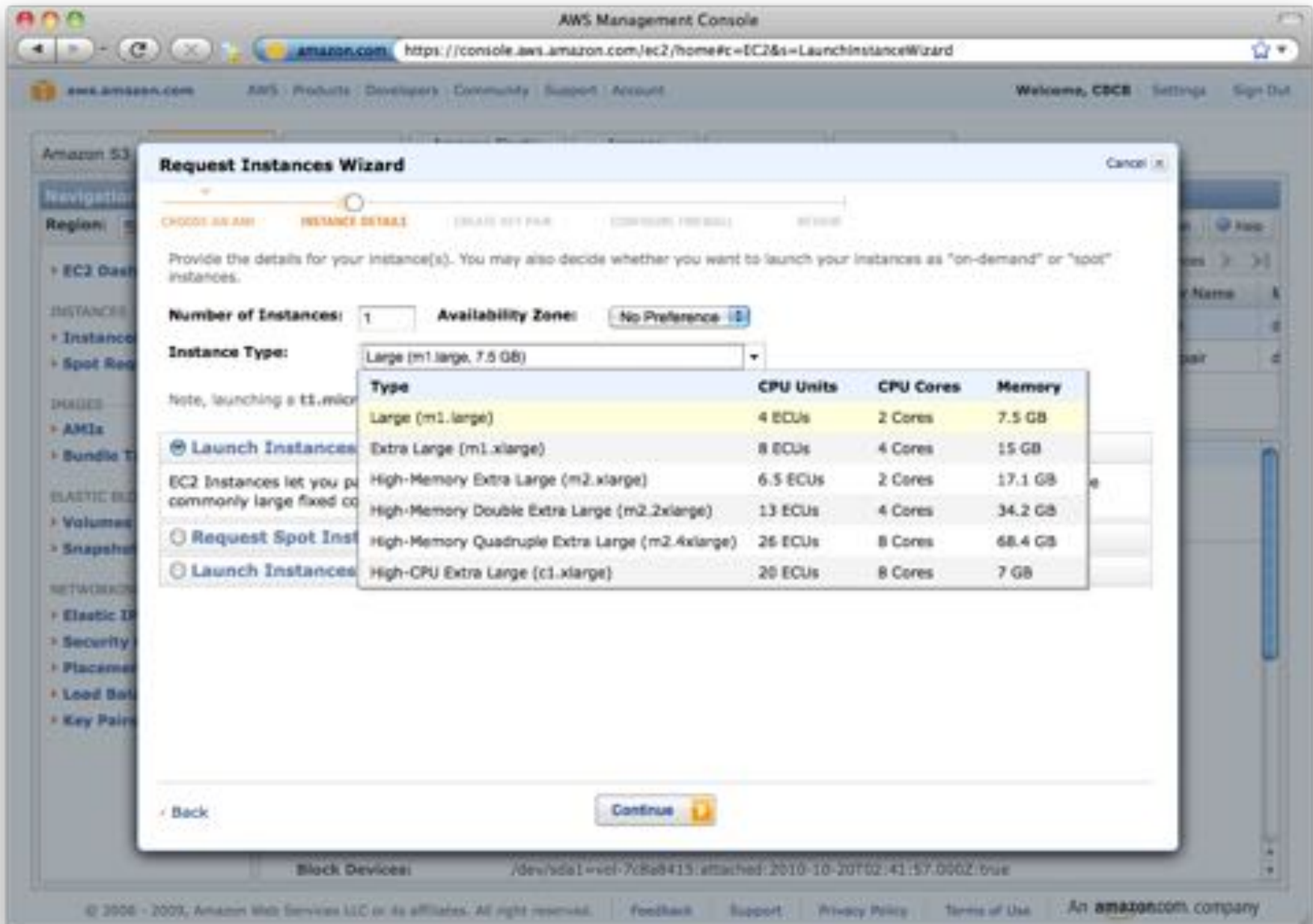
The footer of the console shows: "© 2006 - 2009, Amazon Web Services LLC or its affiliates. All rights reserved. Feedback Support Privacy Policy Terms of Use An amazon.com company"

CloudBioLinux

The screenshot shows the AWS Management Console interface. The main content area displays the 'Request Instances Wizard' for Amazon EC2. The wizard is currently on the 'CHOOSE AN AMI' step. Below the wizard title, there are four tabs: 'Quick Start', 'My AMIs', 'Community AMIs', and 'BYOS'. The 'Community AMIs' tab is selected. A search filter 'cloudbiolinux' is applied to the 'All Images' view. A table lists four CloudBioLinux AMIs, each with a 'Select' button.

AMI ID	Root Device	Manifest	Platform	
ami-0af91263	ebs	678711657553/CloudBioLinux 32bit 20100716	Other Linux	Select
ami-4e57a227	ebs	678711657553/CloudBioLinux 64bit 20100929	Other Linux	Select
ami-6953b200	instance-store	jcvi-cloudbiolinux/JCVI-Cloud-BioLinux.manifest.xml	Other Linux	Select
ami-879c75ee	ebs	678711657553/CloudBioLinux 20100507	Other Linux	Select

2. Pick your Instance Type



The screenshot shows the AWS Management Console's 'Request Instances Wizard' in the 'INSTANCE DETAILS' step. The 'Instance Type' dropdown menu is open, showing a list of instance types. The table below summarizes the visible options:

Type	CPU Units	CPU Cores	Memory
Large (m1.large, 7.5 GB)	4 ECUs	2 Cores	7.5 GB
Extra Large (m1.xlarge)	8 ECUs	4 Cores	15 GB
High-Memory Extra Large (m2.xlarge)	6.5 ECUs	2 Cores	17.1 GB
High-Memory Double Extra Large (m2.2xlarge)	13 ECUs	4 Cores	34.2 GB
High-Memory Quadruple Extra Large (m2.4xlarge)	26 ECUs	8 Cores	68.4 GB
High-CPU Extra Large (c1.xlarge)	20 ECUs	8 Cores	7 GB

At the bottom of the wizard, there is a 'Back' button and a 'Continue' button with a yellow arrow icon. The footer of the console shows the copyright notice: © 2006 - 2009, Amazon Web Services LLC or its affiliates. All rights reserved. and the text 'An amazon.com company'.

3. Pick your Credentials

The screenshot shows the AWS Management Console interface. The browser address bar displays `https://console.aws.amazon.com/ec2/home#ec2=LaunchInstanceWizard`. The page title is "Request Instances Wizard" with a "Cancel" button in the top right. A progress bar at the top indicates the current step is "CREATE KEY PAIR".

The main content area contains the following text and options:

Public/private key pairs allow you to securely connect to your instance after it launches. To create a key pair, enter a name and click **Create & Download your Key Pair**. You will then be prompted to save the private key to your computer. Note, you only need to generate a key pair once - not each time you want to deploy an Amazon EC2 instance.

- Choose from your existing Key Pairs
- Create a new Key Pair

1. Enter a name for your key pair:* (e.g., jdonkey)

2. Click to create your key pair:* [Create & Download your Key Pair](#)

Save this file in a place you will remember. You can use this key pair to launch other instances in the future or visit the Key Pairs page to create or manage existing ones.

Proceed without a Key Pair

At the bottom of the wizard, there is a "Back" link and a "Continue" button. Below the wizard, the "Root Device" is set to `/dev/sda1` and the "Root Device Type" is `ssd`.

© 2008 - 2009, Amazon Web Services LLC or its affiliates. All rights reserved. Feedback Support Privacy Policy Terms of Use An amazon.com company

4. Configure your Firewall

Request Instances Wizard

CHOOSE AN AMI | INSTANCE DETAILS | CREATE KEY PAIR | **CONFIGURE FIREWALL** | REVIEW

Security groups determine whether a network port is open or blocked on your instances. You may use an existing security group, or we can help you create a new security group to allow access to your instances using the suggested ports below. Add additional ports now or update your security group anytime using the Security Groups page. All changes take effect immediately.

Choose one or more of your existing Security Groups

Create a new Security Group

1. Name your Security Group:

2. Describe your Security Group:

3. Define allowed Connections

Application	Transport	Port	Source Network (IPv4 CIDR)	Actions
HTTP	TCP	80	All Internet	<input type="button" value="Remove"/>
SSH	TCP	22	All Internet	<input type="button" value="Remove"/>
Select...	-	-	All Internet Change	<input type="button" value="Add Rule"/>

Block Devices: /dev/sda1=vol-7c8a8415,attached:2010-10-20T02:41:57.000Z,true

© 2006 - 2009, Amazon Web Services LLC or its affiliates. All rights reserved. [Feedback](#) [Support](#) [Privacy Policy](#) [Terms of Use](#) An **amazon.com** company

5. Launch!

The screenshot shows the AWS Management Console interface with the 'Request Instances Wizard' modal open. The wizard is in the 'Review' step, indicated by a progress bar at the top. The configuration details are as follows:

- AMI:** Amazon Linux AMI ID ami-3ac33653 (1386)
Name: Basic 32-bit Amazon Linux AMI 1.0
Description: Amazon Linux AMI Base 1.0, EBS boot, 32-bit architecture with Amazon EC2 AMI Tools. [Edit AMI](#)
- Number of Instances:** 1
Availability Zone: No Preference
Instance Type: Micro (t1.micro)
Instance Class: On Demand [Edit Instance Details](#)
- Monitoring:** Disabled
Kernel ID: Use Default
RAM Disk ID: Use Default
User Data: [Edit Advanced Details](#)
- Key Pair Name:** mschatz [Edit Key Pair](#)
- Security Group(s):** basic [Edit Firewall](#)

At the bottom of the wizard, there is a 'Back' link and a prominent 'Launch' button with a lightning bolt icon. Below the wizard, the 'Root Device' is set to '/dev/sda1' and the 'Root Device Type' is 'efs'. The footer of the console includes copyright information and links for Feedback, Support, Privacy Policy, Terms of Use, and the Amazon logo.

Monitoring your Server

The screenshot shows the AWS Management Console interface. At the top, the browser address bar displays the URL `https://console.aws.amazon.com/ec2/home#ec2:instances`. The navigation bar includes the AWS logo, navigation links (Products, Developers, Community, Support, Account), and a user profile section with the name 'Welcome, CBCB' and links for 'Settings' and 'Sign Out'.

Below the navigation bar, there are tabs for various AWS services: Amazon S3, Amazon EC2 (selected), Amazon VPC, Amazon Elastic MapReduce, Amazon CloudFront, Amazon RDS, and Amazon SNS.

The main content area is divided into two sections. On the left is a 'Navigation' sidebar with a 'Region' dropdown set to 'US East'. The sidebar lists categories like INSTANCES, IMAGES, ELASTIC BLOCK STORE, and NETWORKING & SECURITY, with sub-items such as EC2 Dashboard, Instances, Spot Requests, AMIs, Bundle Tasks, Volumes, Snapshots, Elastic IPs, Security Groups, Placement Groups, Load Balancers, and Key Pairs.

The right section is titled 'My Instances'. It features a 'Launch Instance' button and a 'Reserved Instances' dropdown. Below this, there are filters for 'Viewing: All Instances' and 'All Instance Types'. A table lists the instances:

Name	Instance	AMI ID	Root Device	Type	Status	Security Groups	Key Pair Name	
<input type="checkbox"/>	empty	i-4aab6027	ami-3ac3853	ebs	t1.micro	running	crossbow45474681	gig-keypair

Below the table, a summary bar indicates '0 EC2 Instances selected' and prompts the user to 'Select an instance above'.

The footer contains copyright information: '© 2008 - 2009, Amazon Web Services LLC or its affiliates. All rights reserved.' and links for 'Feedback', 'Support', 'Privacy Policy', and 'Terms of Use'. It also states 'An amazon.com company'.

Connecting (I)

The screenshot displays the AWS Management Console interface. At the top, the browser address bar shows the URL `https://console.aws.amazon.com/ec2/home#ec2:instances`. The navigation bar includes links for Amazon S3, Amazon EC2 (selected), Amazon VPC, Amazon Elastic MapReduce, Amazon CloudFront, Amazon RDS, and Amazon SNS. The main content area is titled "My Instances" and shows a table with one instance named "empty" (ID: i-4aab6027) in a "running" state. A context menu is open over the instance, with the "Connect" option highlighted. Below the table, the instance details for "EC2 Instance: i-4aab6027" are displayed, including AMI ID, Security Groups, Status, VPC ID, Virtualization, Reservation, Platform, Kernel ID, AMI Launch Index, Root Device, Instance Management actions, Instance Lifecycle actions, and CloudWatch Monitoring options.

Name	Instance	AMI ID	Root Device	Type	Status	Security Groups	Key Pair Name
empty	i-4aab6027	ami-3ac33653	ebs	t1.micro	running	crossbow464746833350	gsg-keypair

1 EC2 Instance selected

EC2 Instance: i-4aab6027

Description | Monitoring | Tags

AMI ID: ami-3ac33653
Security Groups: crossbow464746833350-X-master
Status: running
VPC ID: -
Virtualization: para
Reservation: r-81f3c8b
Platform: -
Kernel ID: aki-407d9529
AMI Launch Index: 0
Root Device: /dev/sda1

Instance Management

- Connect
- Get System Log
- Create Image (EBS AMI)
- Add/Edit Tags
- Launch More Like This
- Dissociate IP Address

Instance Lifecycle

- Terminate
- Reboot
- Stop
- Start

CloudWatch Monitoring

- Enable CloudWatch
- Disable CloudWatch

Region: US East
Zone: us-east-1c
Type: t1.micro
Owner: 464746833350
Subnet ID: -
Placement Group: -
RAM Disk ID: -
Key Pair Name: gsg-keypair
Monitoring: disabled
Elastic IP: -
Root Device Type: ebs

© 2008 - 2009, Amazon Web Services LLC or its affiliates. All rights reserved. | Feedback | Support | Privacy Policy | Terms of Use | An amazon.com company

Connecting (2)

The screenshot shows the AWS Management Console interface. A modal dialog titled "Connect Help - Secure Shell (SSH)" is open, providing instructions for connecting to an EC2 instance. The instance ID is i-4aab6027. The dialog lists four steps for accessing the instance using any SSH client. An example command is provided in a text box: `ssh -i gsg-keypair.pem root@ec2-184-72-85-153.compute-1.amazonaws.com`. The background shows the "My Instances" page with a table of instances.

Name	Security Groups	Key Pair Name
empty	crsrbow46474683	gsg-keypair

Below the table, the details for the instance are visible:

- AMI ID: ami-407c9529
- Kernel ID: aki-407c9529
- AMI Launch Index: 0
- Root Device: /dev/sdc1
- Platform: Amazon Linux AMI
- Key Pair: gsg-keypair
- Monitor: Optional, RAM disk associated with the image. For more information, refer to the Amazon Elastic Compute Cloud Developer Guide.
- Elastic IP: -
- Root Device Type: ebs

Calling SNPs in the Cloud ☺

```
chmod 400 mschatz.pem
```

```
scp -r -i mschatz.pem data.tgz ubuntu@ec2-174-129-123-73.compute-1.amazonaws.com:  
ssh -i mschatz.pem ubuntu@ec2-174-129-123-73.compute-1.amazonaws.com
```

```
<remote>
```

```
ls
```

```
tar xzvf data.tgz
```

```
bowtie -S data/genomes/e_coli data/reads/e_coli_10000snp.fq ec_snp.sam
```

```
samtools view -bS -o ec_snp.bam ec_snp.sam
```

```
samtools sort ec_snp.bam ec_snp.sorted
```

```
samtools pileup -cv -f data/genomes/NC_008253.fna ec_snp.sorted.bam > snps
```

```
samtools index ec_snp.sorted.bam
```

```
samtools tview ec_snp.sorted.bam data/genomes/NC_008253.fna
```

```
exit
```

```
<local>
```

```
scp -i mschatz.pem ubuntu@ec2-174-129-123-73.compute-1.amazonaws.com:snps .
```

1000Genomes in the Cloud

```
s3cmd --configure
```

```
# cp data/.s3cfg .
```

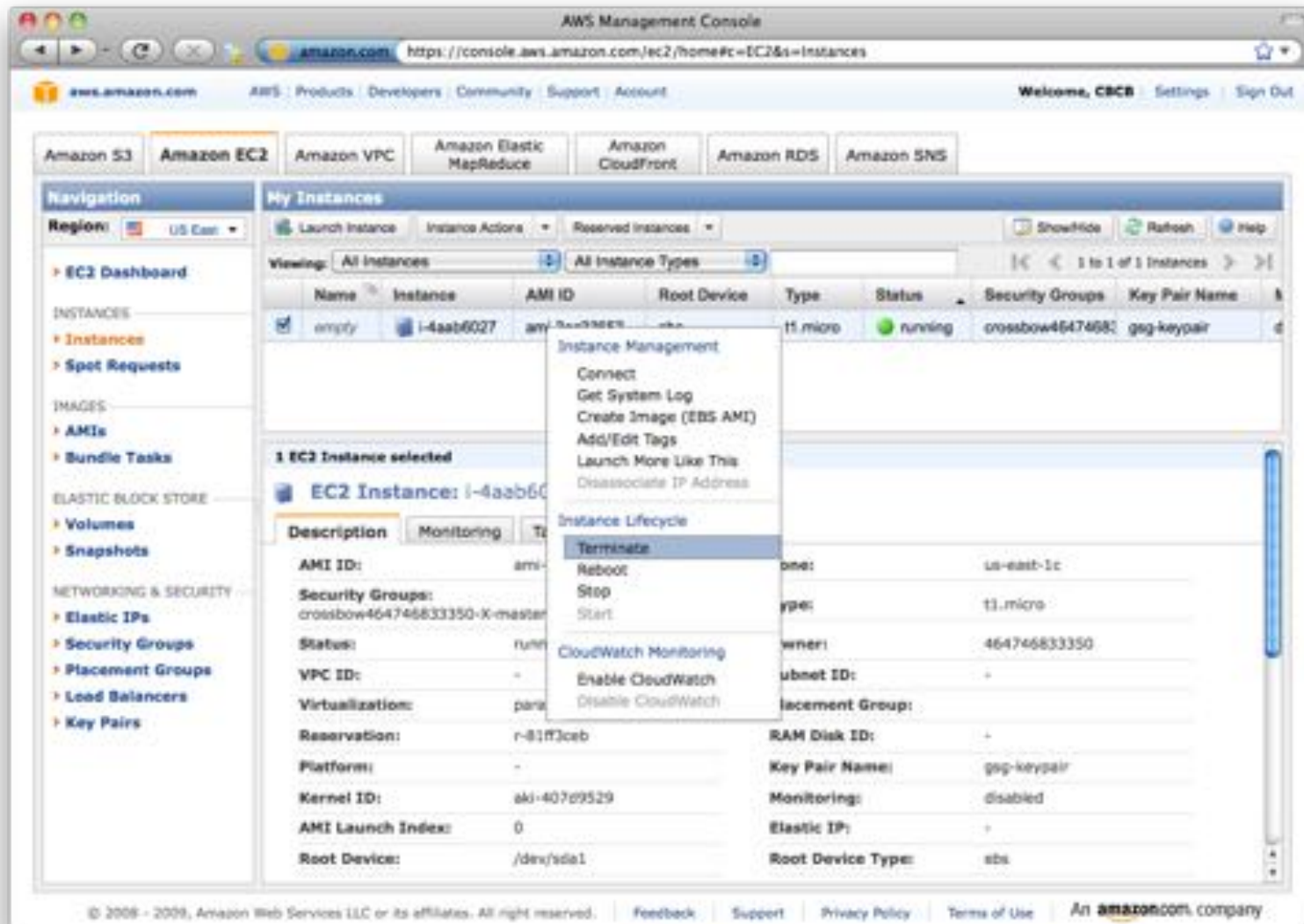
```
s3cmd ls s3://1000genomes
```

```
s3cmd ls s3://1000genomes/Pilots_Bam/NA20828/
```

```
s3cmd get s3://1000genomes/Pilots_Bam/NA20828/*chr22* .
```

```
samtools view NA20828.SLX.maq.SRP000033.2009_09.chr22_1_49691432.bam
```

Terminating

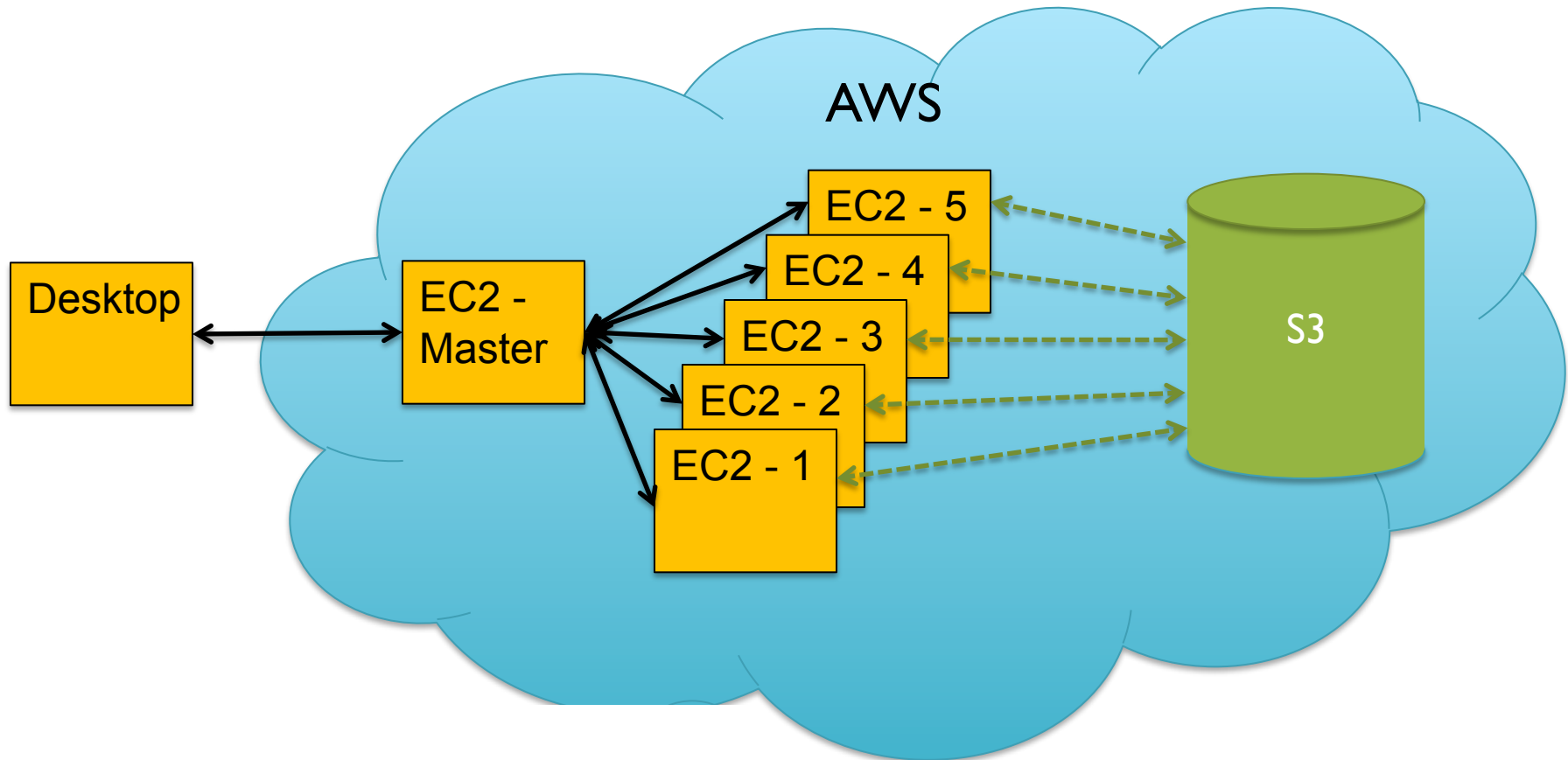


Total cost: 8.5¢

Reflections

- Launching and managing virtual clusters with the AWS Console is quick and easy
 - Entirely scriptable using ec2 tools
 - iPhone App also available
- Things get really interesting on 168 cores
 - 1 week CPU = 1 hour wall

Hadoop on AWS



Just 3 commands to bring up a 168 core (21 node) cluster & crunch terabytes:

```
$HADOOP/src/contrib/ec2/bin/hadoop-ec2 launch-cluster HADOOP 21
```

```
$HADOOP/src/contrib/ec2/bin/hadoop-ec2 <hadoop cmd> HADOOP
```

```
$HADOOP/src/contrib/ec2/bin/hadoop-ec2 terminate-cluster HADOOP
```

Kmer Code

kmer-map.pl

```
#!/usr/bin/perl

my $K = 15;

while (<>)
{
    for (my $i = 0;
         $i < length($_)-$K;
         $i++)
    {
        my $kmer = substr($_,$i,$K);
        print "$kmer\t1\n";
    }
}
```

kmer-reduce.pl

```
#!/usr/bin/perl

my $mer = "";
my $cnt = 0;

while (<>)
{
    chomp;

    my ($curmer, $curcnt) = split /\t/, $_;

    if ($curmer ne $mer)
    {
        print "$mer\t$cnt\n" if ($cnt > 0);
        $mer = $curmer; $cnt = 0;
    }

    $cnt += $curcnt;
}

print "$mer\t$cnt\n" if ($cnt > 0);
```


BashReduce

```
$ head -3 reads.txt
```

```
ATATTTTTTTCTTGTTTTTTTATATCCACAAACTCTTT  
CCACAAAATCAATACCTTGTGGAATAAAATTGTCCA  
TATTTTTTTCTTGTTTTTTTATATCCACAAACTCTTT
```

```
$ cat reads.txt | ./kmer-map.pl | head -3
```

```
ATATTTTTTTCTTGTT      1  
TATTTTTTTCTTGTTT     1  
ATTTTTTTCTTGTTTT     1
```

```
$ cat reads.txt | ./kmer-map.pl | sort \  
  | ./kmer-reduce.pl | head -3
```

```
AAAAAAAAGTAGCTA      44  
AAAAAAAAGTAGCTAT     44  
AAAAAAGCAAATGTG      17
```

Kmer Counting In Hadoop

```
#!/bin/sh
```

```
STREAMING=/usr/lib/hadoop-0.20/contrib/streaming/hadoop-streaming-0.20.2+320.jar
```

```
hadoop fs -mkdir /user/mschatz/kmertest/reads
```

```
hadoop fs -put reads.txt /user/mschatz/kmertest/reads
```

```
hadoop jar $STREAMING \  
  -input /user/mschatz/kmertest/reads \  
  -output /user/mschatz/kmertest/kmers \  
  -mapper ./kmer-map.pl \  
  -reducer ./kmer-reduce.pl \  
  -file ./kmer-map.pl \  
  -file ./kmer-reduce.pl \  
  -jobconf mapred.map.tasks=10 \  
  -jobconf mapred.reduce.tasks=1
```

```
hadoop fs -cat /user/mschatz/kmertest/kmers/part-* | head -3
```

```
hadoop fs -rmr /user/mschatz/kmertest
```

Hadoop Output

```
10/10/21 16:03:51 INFO mapred.FileInputFormat: Total input paths to process : 1
10/10/21 16:03:51 INFO streaming.StreamJob: getLocalDirs(): [/scratch0/hadoop/mapred/
    local]
10/10/21 16:03:51 INFO streaming.StreamJob: Running job: job_201009232028_2089
10/10/21 16:03:51 INFO streaming.StreamJob: To kill this job, run:
10/10/21 16:03:51 INFO streaming.StreamJob: /usr/lib/hadoop-0.20/bin/hadoop job -
    Dmapred.job.tracker=szhdname01.umiacs.umd.edu:8021 -kill job_201009232028_2089
10/10/21 16:03:51 INFO streaming.StreamJob: Tracking URL: http://
    szhdname01.umiacs.umd.edu:50030/jobdetails.jsp?jobid=job_201009232028_2089
10/10/21 16:03:52 INFO streaming.StreamJob: map 0% reduce 0%
10/10/21 16:03:58 INFO streaming.StreamJob: map 30% reduce 0%
10/10/21 16:04:01 INFO streaming.StreamJob: map 100% reduce 0%
10/10/21 16:04:07 INFO streaming.StreamJob: map 100% reduce 20%
10/10/21 16:04:16 INFO streaming.StreamJob: map 100% reduce 100%
10/10/21 16:04:19 INFO streaming.StreamJob: Job complete: job_201009232028_2089
10/10/21 16:04:19 INFO streaming.StreamJob: Output: /user/mschatz/kmertest/kmers
```

```
AAAAAAAAGTAGCTA      44
AAAAAAAAGTAGCTAT    44
AAAAAAGCAAATGTG     17
```

Crossbow Webform

<http://bowtie-bio.sf.net/crossbow/ui.html>

The screenshot shows a web browser window with the URL `http://ec2-184-73-43...cgi-bin/crossbow.pl`. The page contains a form with the following fields and options:

- AWS ID ***:
- AWS Secret Key ***:
- AWS Keypair Name**: [Look it up](#)
[Check credentials...](#)
- Job name**:
- Job type**: Crossbow Just preprocess reads
- Input URL ***:
[Check that input URL exists...](#)
- Output URL ***:
[Check that output URL doesn't exist...](#)
- Input type**: Preprocessed reads Manifest file
- Truncate length**: (if blank or 0, truncation is disabled)
 Skip reads shorter than truncate length
- Discard fraction**:
- Quality encoding**:
- Genome/Annotation**:
 Specify reference jar URL:

[Check that reference jar URL exists...](#)
- Bowtie options**:
- SOAPdenovo options**:
- Additional SOAPdenovo options for haplotigs**:
- Additional SOAPdenovo options for diploids**:

- Enter your account info, manifest file, reference info, and pipeline settings
 - List of URLs to fastq files
- Crossbow
 - Parallel ftp
 - Parallel map
 - Parallel SNPs

More Information

- Amazon Web Services
 - <http://aws.amazon.com>
 - <http://aws.amazon.com/free>
- Hadoop
 - <http://hadoop.apache.org>
- Crossbow & Bowtie
 - <http://bowtie-bio.sf.net>



Thank You!

<http://schatzlab.cshl.edu>

@mike_schatz