HPC for Genomic Data at Scale

Kathy Yelick
Associate Laboratory Director
Computing Sciences

Professor of Electrical Engineering and Computer Sciences, UC Berkeley

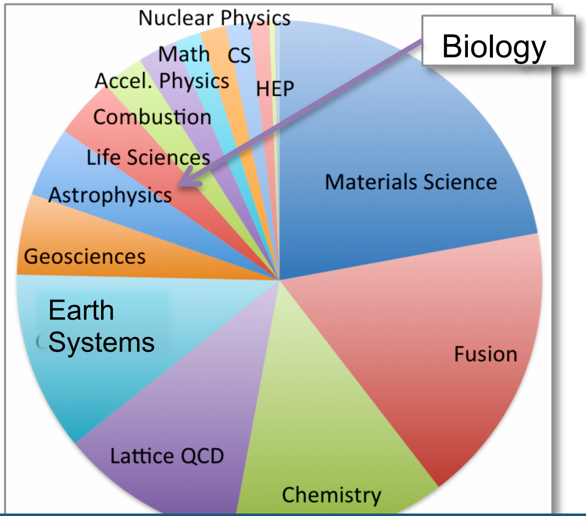
BERKELEY LAB
Lawrence Berkeley National Laboratory

U.S. DEPARTMENT OF ENERGY

DOE leadership in HPC

- **DOE leads High Performance Computing in the US**
- **Capabilities for science, engineering, and defense**
- **Expertise in mathematics, computer science, modeling and simulation and data analytics**

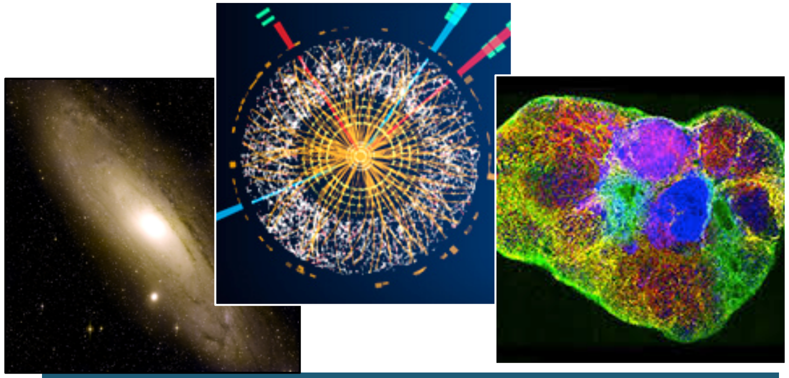
NERSC: Dedicated to DOE Science



> 7000 users, 700 application codes

> 2000 annual publications; 6 Nobels

Systems architected for science



History of data-intensive science

Myths of Genomics and HPC

(And a bit of computer science along the way)

Myth #1: Genomic assembly requires large shared memory machines

HPC systems can look like shared memory if you use the right algorithms and programming models

De Novo genome assembly problem

Input

reads
(input, typically 100-250 chars)

GCTACGGAATAAAACCAGGAACAACAGAGCCAGCAC

ATAAAACCAGGTACAACAGACCCAGCACGGATCCA

GC_ACGGAATACAACCAGGAACAACAGACCCAGCAC

GAACAACAGACCCAGCATGGATCCA

errors

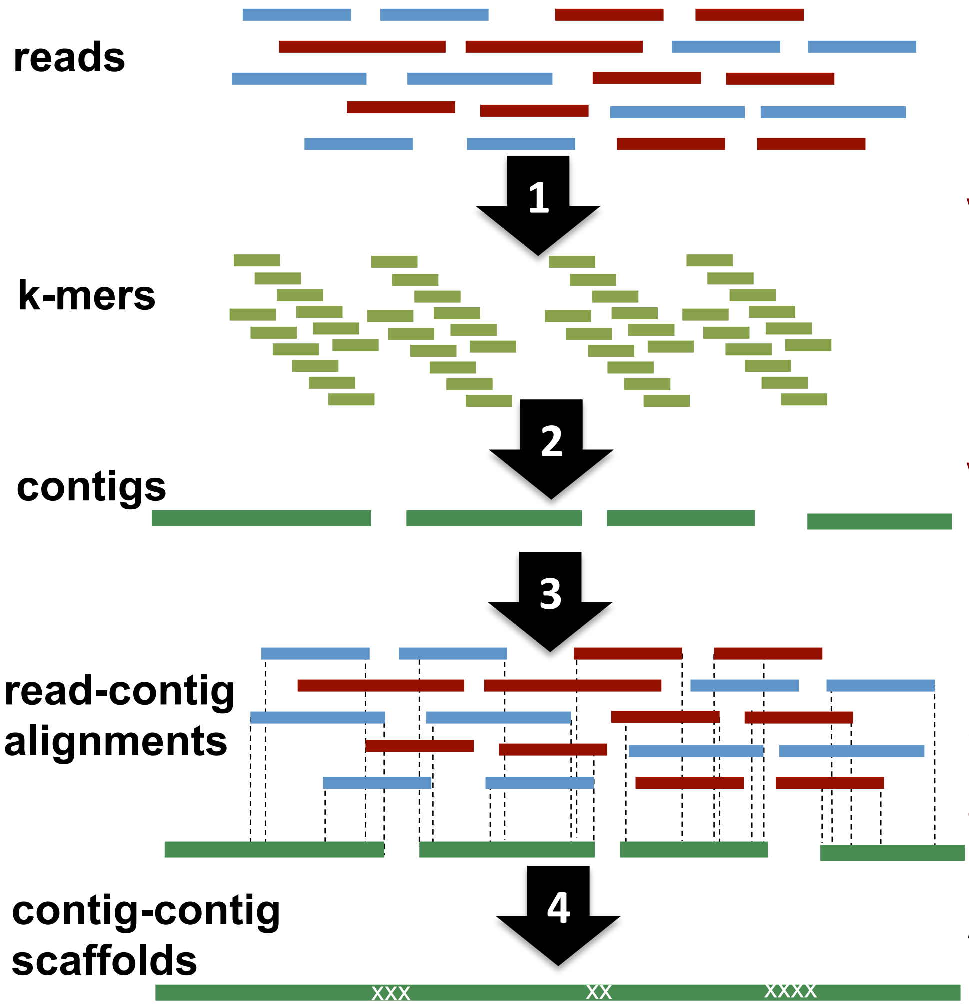
Multiple copies
(20x typical)

GCTACGGAATAAAACCAGGAACAACAGACCCAGCACGGATCCA

Output

The fully assembled genome (or 10s of Ks of bp fragments so we can find genes, which are typically longer than the reads)

HipMer genome assembly based on Meraculous



1) K-mer Analysis

Histogram fixed-length fragments with bloom filters

2) Contig Generation

Build **hash table** of k-mers and walk as graph

3) Alignment

Build a **hash table** of k-mers in contigs and map to reads (seed-and extend)

4) Scaffolding & Gap Closing

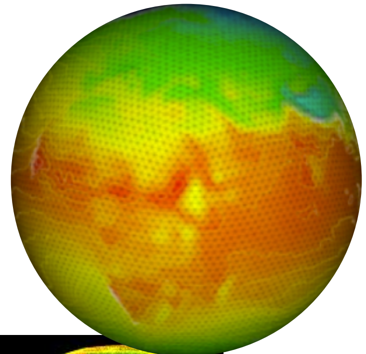
Build a **hash table** of contig pairs and merge them (local assembly)

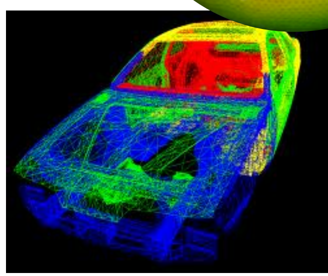
Chapman, Jarrod A., et al. "Meraculous: de novo genome assembly with short paired-end reads." PloS ONE 6.8 (2011)

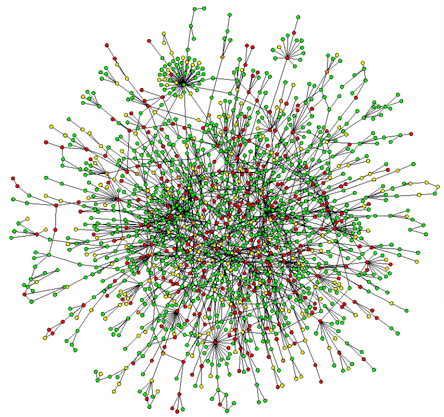
Using HPC for Large Memory Problems

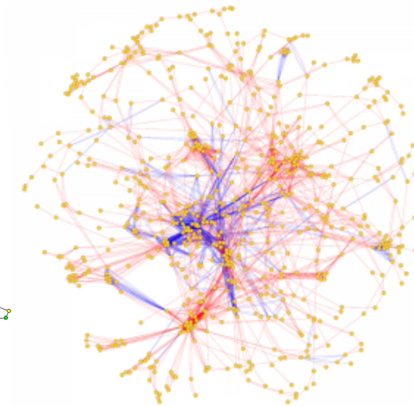
	GenePool (JGI) Large node	Cori Haswell	Cori KNL
Nodes	1	1630	9600
cores / node	80 cores	32 cores	68 cores
Memory / node	2 TB	128 GB	96 GB
Total memory	2 TB	299 TB	1060 TB
Storage	300 GB (local)	30PB (global)	30PB (global)
Interconnect	1 Gb/sec	80 Gb/s	80 Gb/s
Bisection Bandwidth		6 TB/s	45 TB/s

Shared Memory Thinking on Distributed Memory









<u>Message Passing Programming</u>

Divide up domain in pieces

Compute one piece and exchange

MPI, and many libraries

<u>Global Address Space Programming</u>

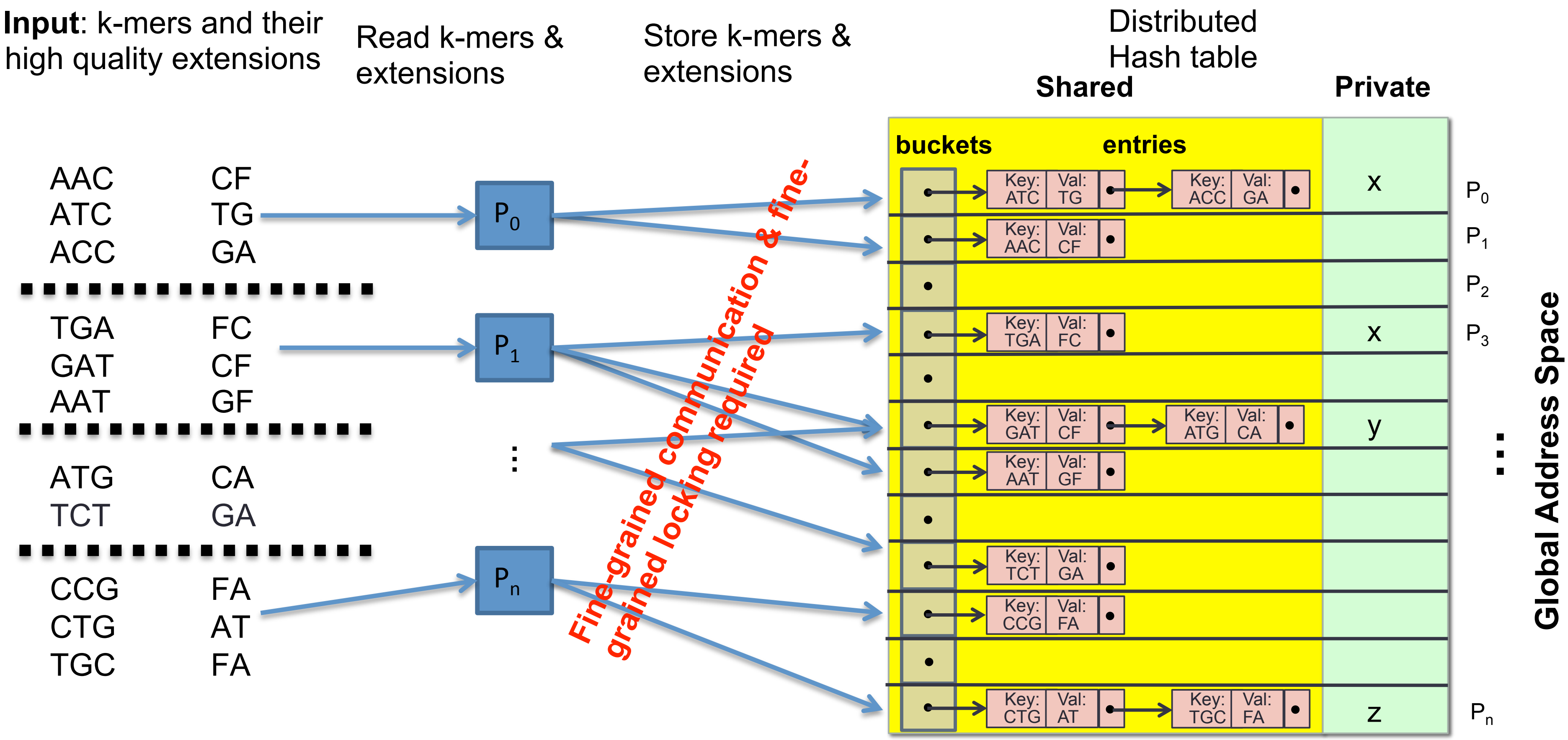
Each start computing

Grab whatever / whenever

UPC, UPC++, CAF, X10, Global Arrays, Chapel, and more

Graph algorithms (hash tables) in assembly

Graph construction, traversal, and all later stages are written in UPC to take advantage of its global address space



Using HipMer for first-ever science

Assembly of bread wheat genomes

- Wheat genome: 17 Gbp
- Assembled without chromosome sorting
- Over half of contigs > 7 kb and scaffolds > 20 kb

Chapman, Jarrod A., et al. " Genome biology (2015)

Twitchell Wetlands (preliminary)

- MetaHipMer uses k-mer lengths and new scaffolding approaches
- 21 libraries, 7.4B reads, 2.8 TB
- 34% reads assembled (vs < 10%)
- First whole assembly of 21 libraries -- largest of its kind?

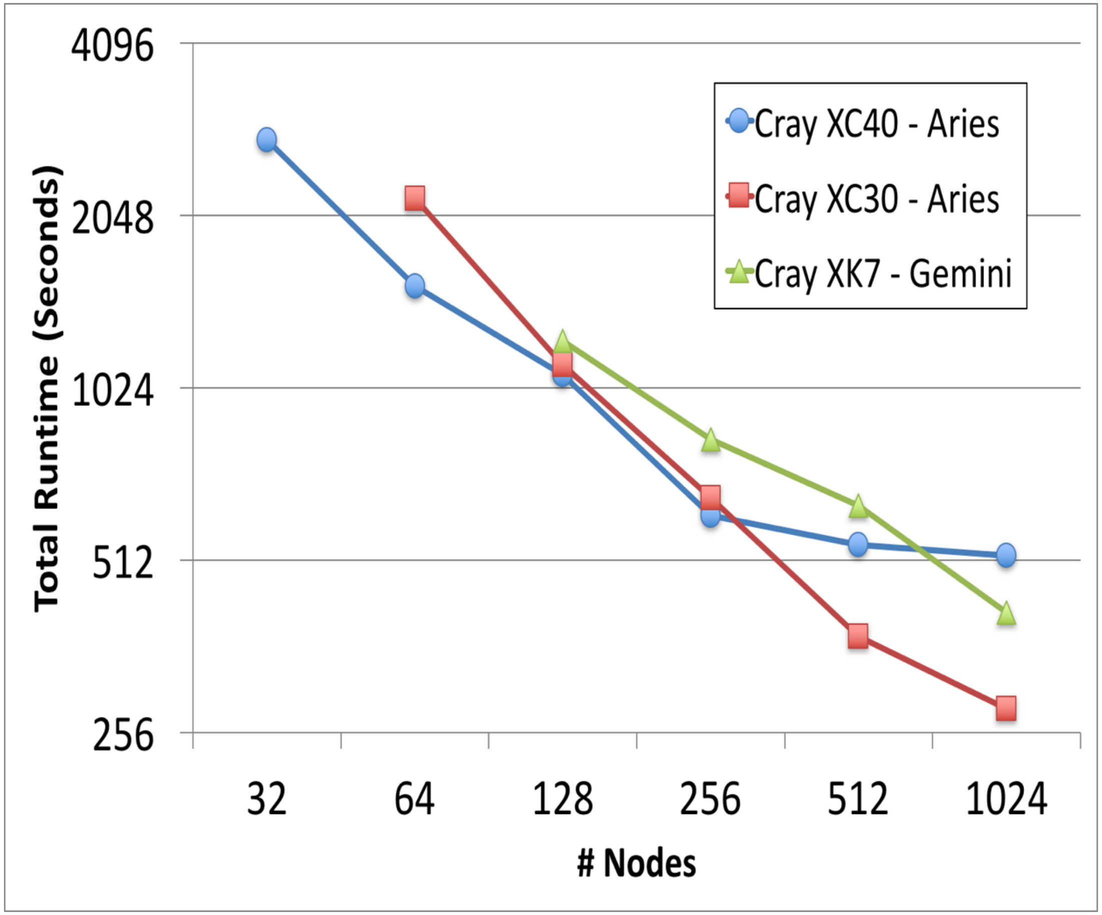
Myth #2: Genomic assembly needs large memory, but not large parallelism

Orders of magnitude speedup are possible through parallelism

Multi-Node Strong Scaling

HipMer scales to a thousand nodes (10Ks of cores) on a fixed modest sized problem (human genome)

- De Novo human assembly in 4 minutes
- Uses 1K nodes (24-32K cores)



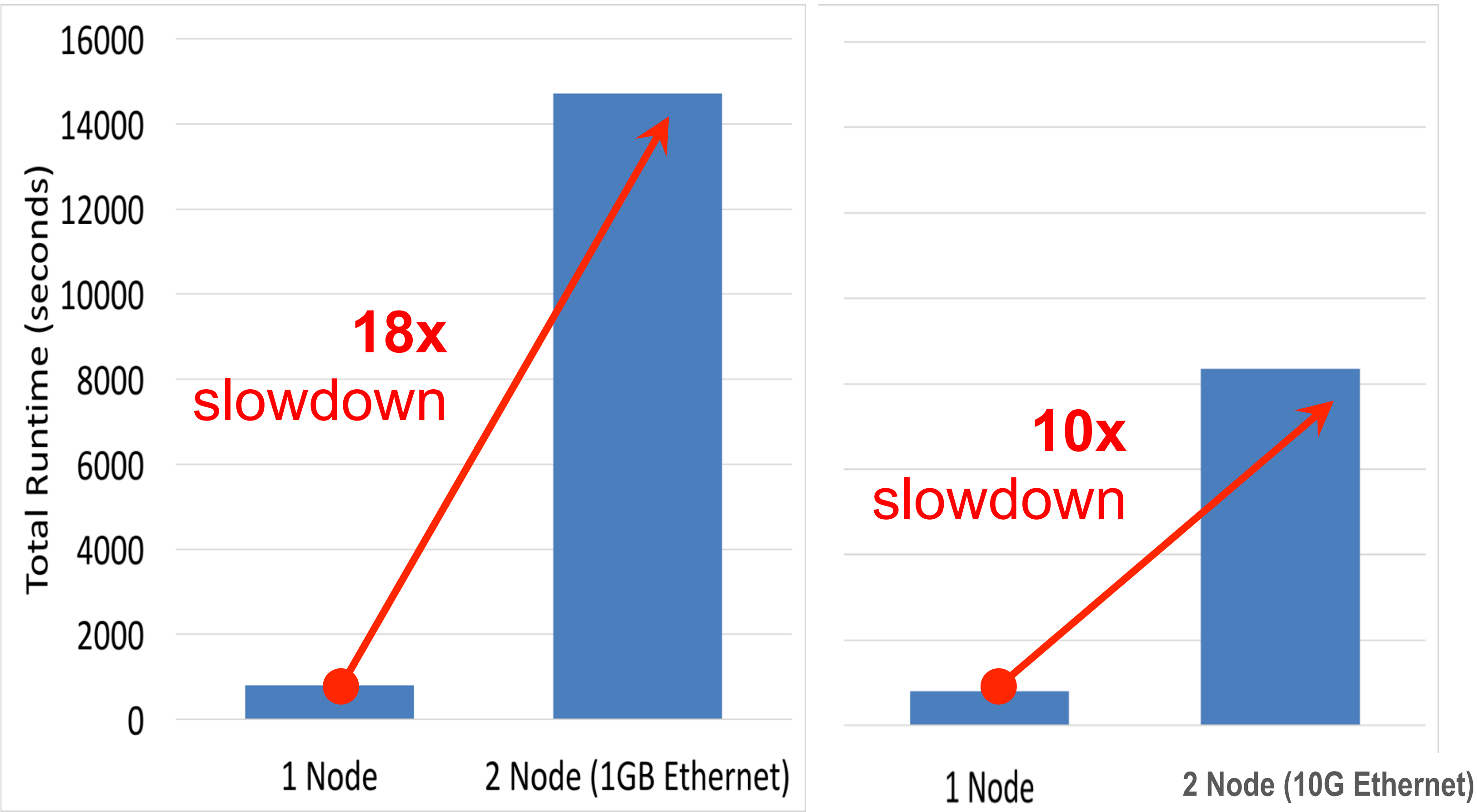
Shared HPC systems accelerate turnaround

M. Ellis, E. Georganas, R. Egan, S. Hofmeyr, A. Buluc, B. Cook, L. Oliker, K. Yelick, "Performance Characterization of De Novo Genome Assembly on Leading Parallel Systems" Euro Par 2017.

Don’t try this at home … you need an HPC network

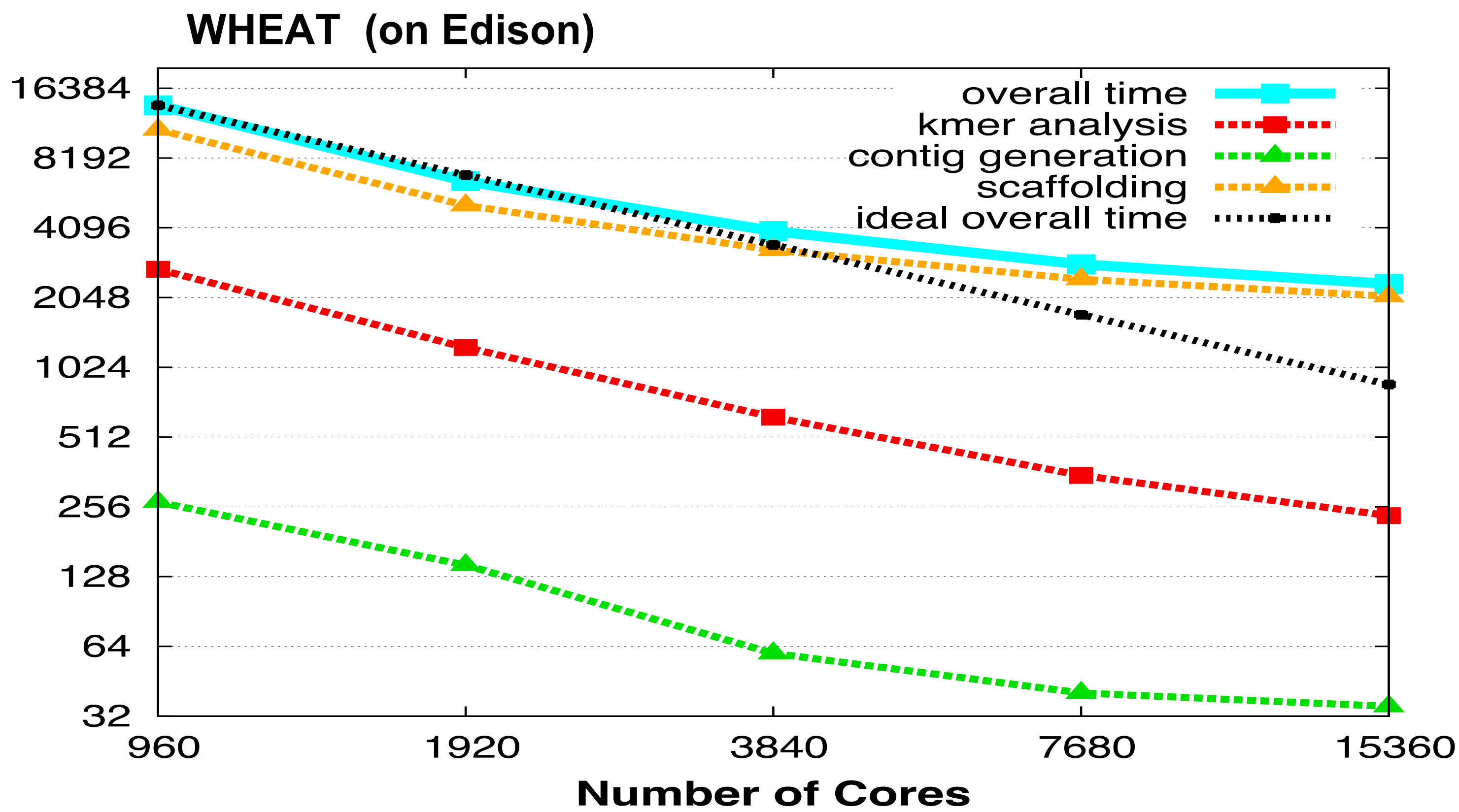
- Requires fast underlying network (e.g. **NOT** ethernet)

Ethernet slowdown 18x (on **1Gb switch**) and 10x (on **10Gb fiber optic patch**)



Reference: M. Ellis, E. Georganas, R. Egan, S. Hofmeyr, A. Buluc, B. Cook, L. Oliker, K. Yelick, “Performance Characterization of De Novo Genome Assembly on Leading Parallel Systems” Euro Par 2017.

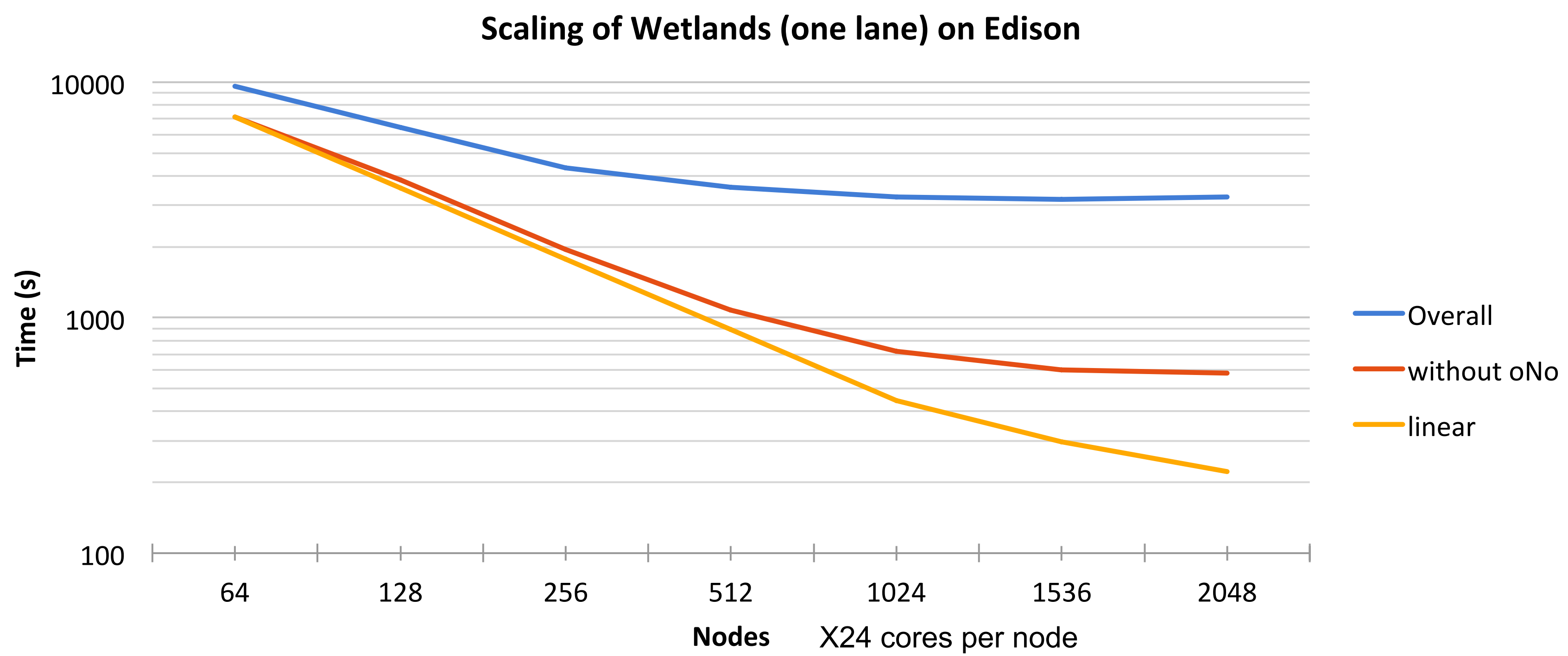
HipMer algorithms scale on and off nodes



Speedup for wheat genomes on up to 15K cores

Chapman, Jarrod A., et al. "Meraculous: de novo genome assembly with short paired-end reads." PloS ONE 6.8 (2011)

MetaHipMer at Scale: Amdahl's Law Strikes



- Demonstrated MetaHipMer scalability on 1-lane Wetlands (above) and multiple synthetic metagenome data sets
- New connected components oNo removes above bottleneck
- New HMM-based scaffolder aids ribosomal assembly

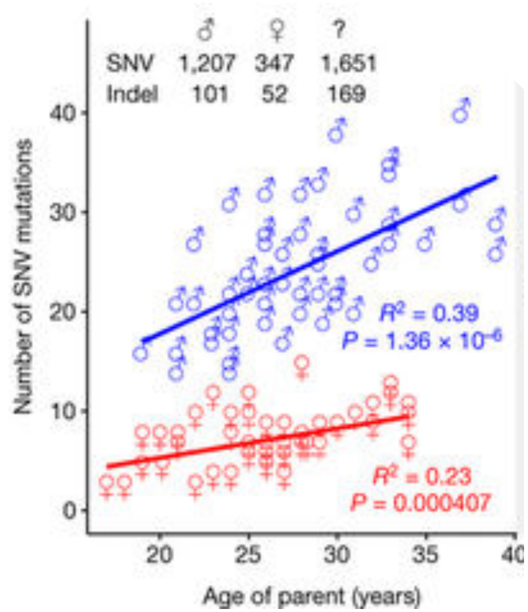
Myth #3: HPC is just about solving the same problems faster

The memory size and speed enables improved quality, new approaches, and new science

Pan-genome studies reveal intra-species diversity

150 de novo assemblies of individuals in Denmark

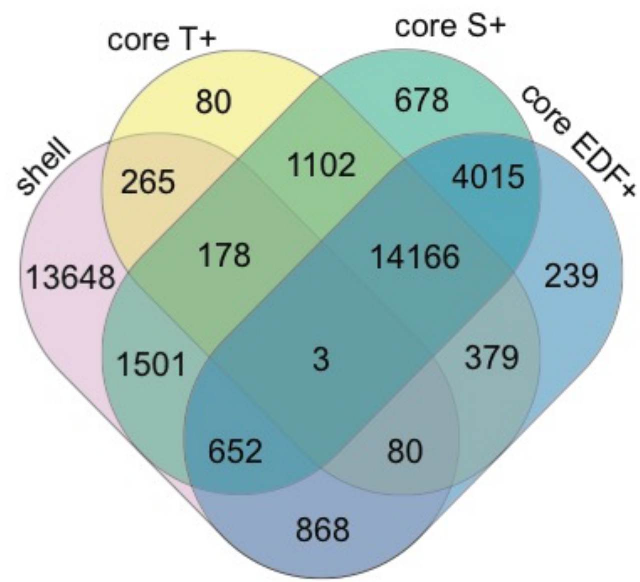
- Many genes not in reference
- 91.6% of insertions ≥50 bp were novel
- Reveals previous deletion bias



L Maretty *et al. Nature* 1–5 (2017)

54 de novo assemblies of the grass *Brachypodium distachyon*

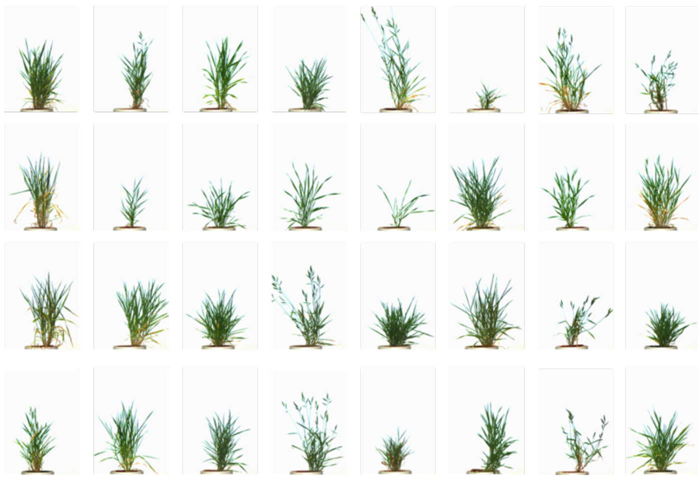
- Nearly 2x the number of genes found in any individual genome
- Many shell genes species-wide are core within a subpopulation.

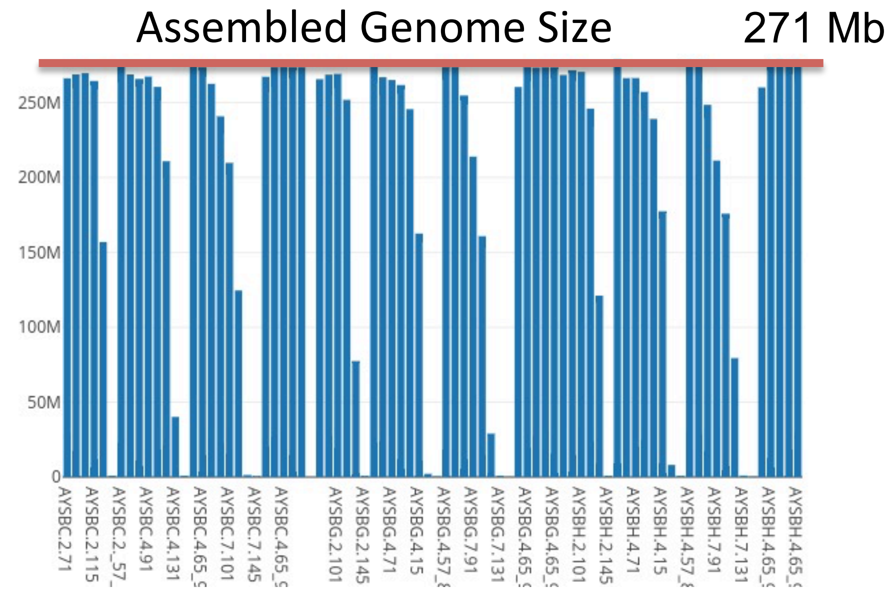


Gordon *et al.* Nature Communications, 2017

HipMer enables parameter exploration in large genomes

Brachypodium distachyon: Different populations grown under the same conditions differ in phenotype

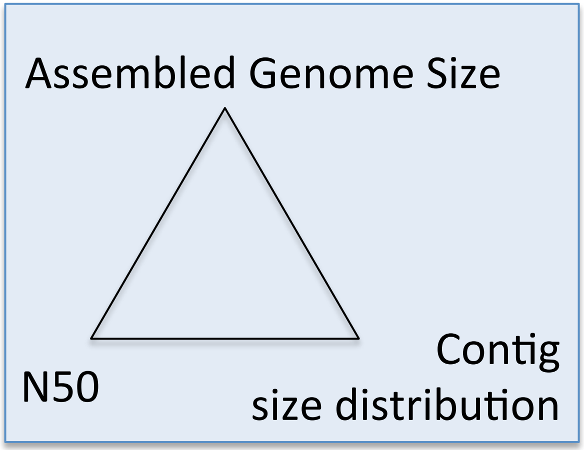




Different parameters over 10 samples

Summary from Sean Gordon (JGI, now Zymergen)

1. HipMer is faster using **fewer resources**
2. **Iterative kmer size** and **iterative scaffolding** improves assembly metrics
3. Combining several low depth, related samples, yields good assemblies

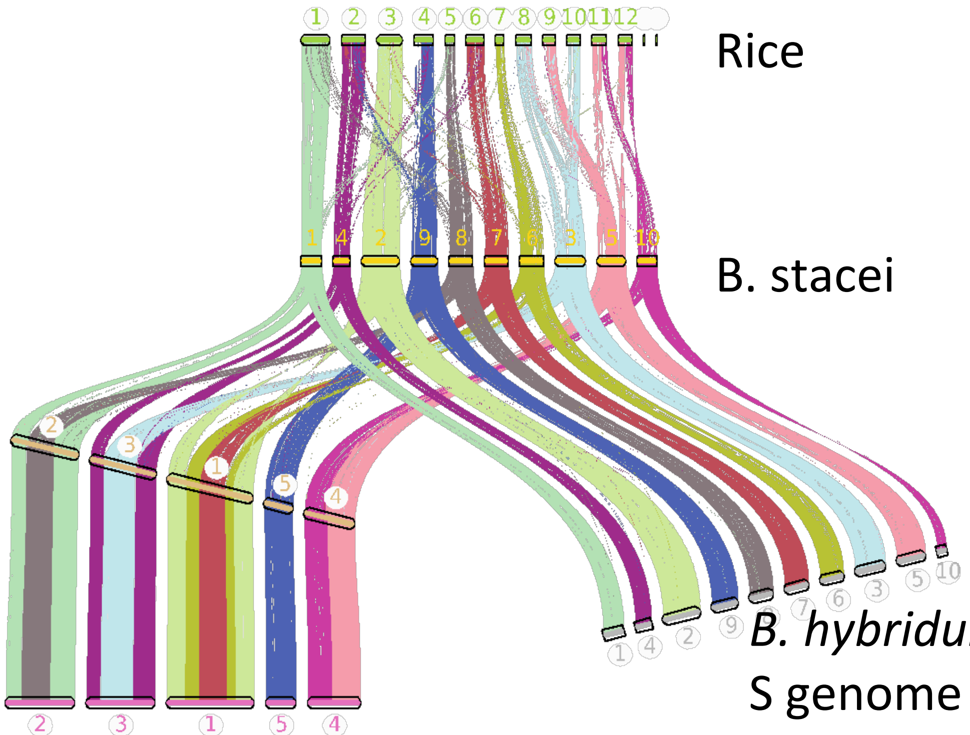


Pan-genomics needs high performance assembly

Assembling many genomes from different populations allows us to capture the majority of genes in the species

Problem: *De novo* assembly of hundreds of large genomes would take years to compute!

HipMer allows us to make 100s of assemblies

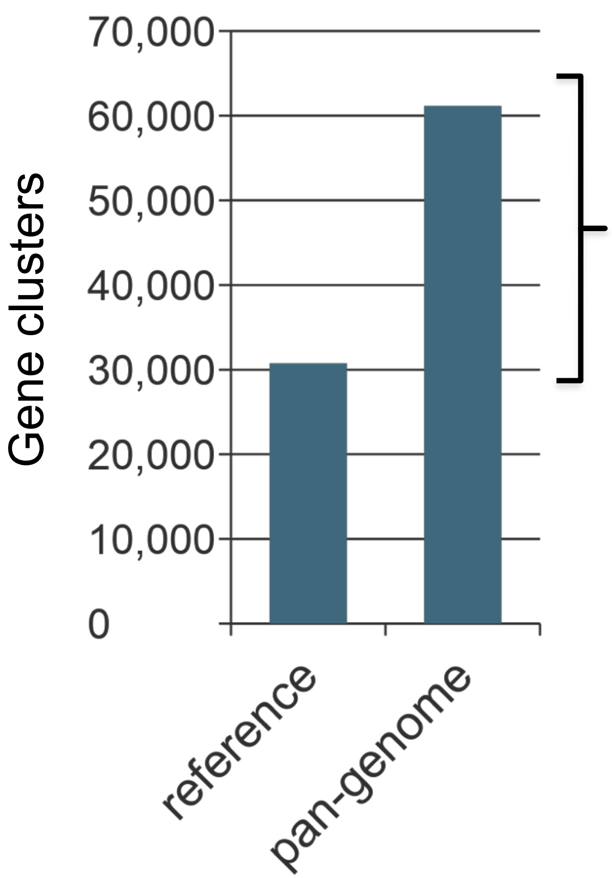


S. Gordon et al: Bits of the two subgenomes are lost over time in the hybrid -- can study this evolution

Gene-based pan-genome with clustering

CDS sequences from genes are clustered by an orthoMCL-like algorithm
61,155 pan-genome clusters

Pan-genes = representative sequence per cluster



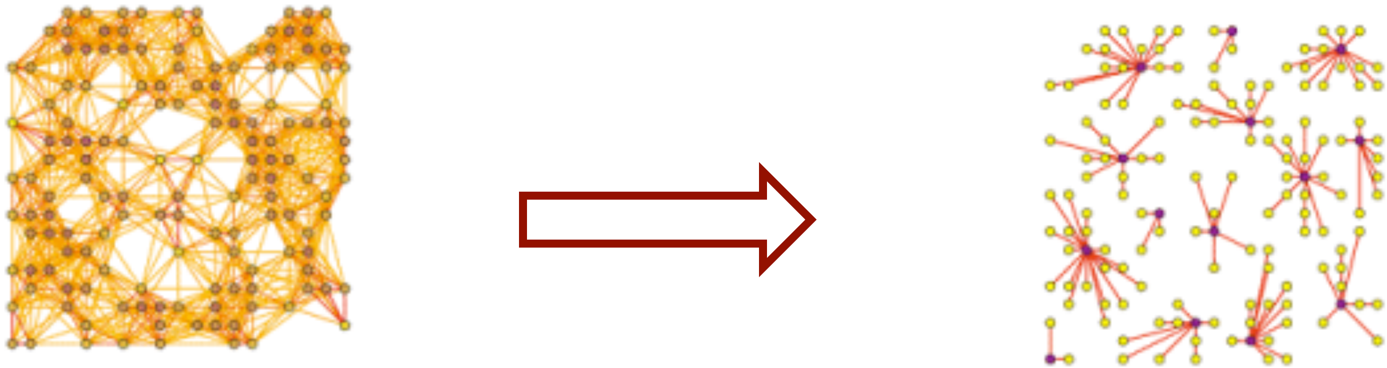
26,125 pan-genes that are not in the reference genome

Problem: Identify gene/protein families at scale

- A **protein family**: group of proteins with common evolutionary origin, reflected by similar functions, sequence or structure

Input: pairwise similarities between proteins (sparse)

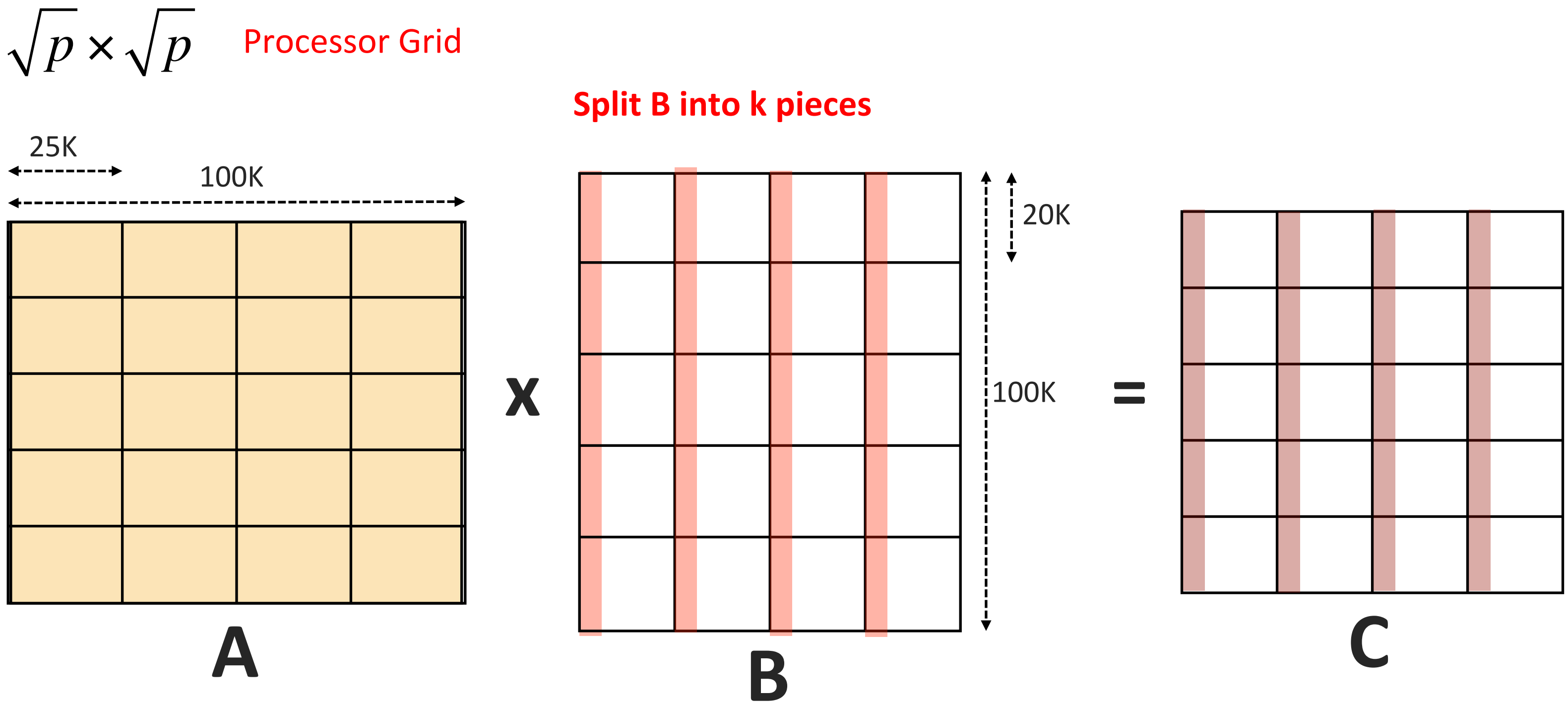
Output: clusters of similar proteins



- **Desired scale:** 10s of billions of genes/proteins, trillions of nonzero pairwise similarities ("all metagenomes")
- Today: 47M genes took 10 days before aborting (est. 45 days)

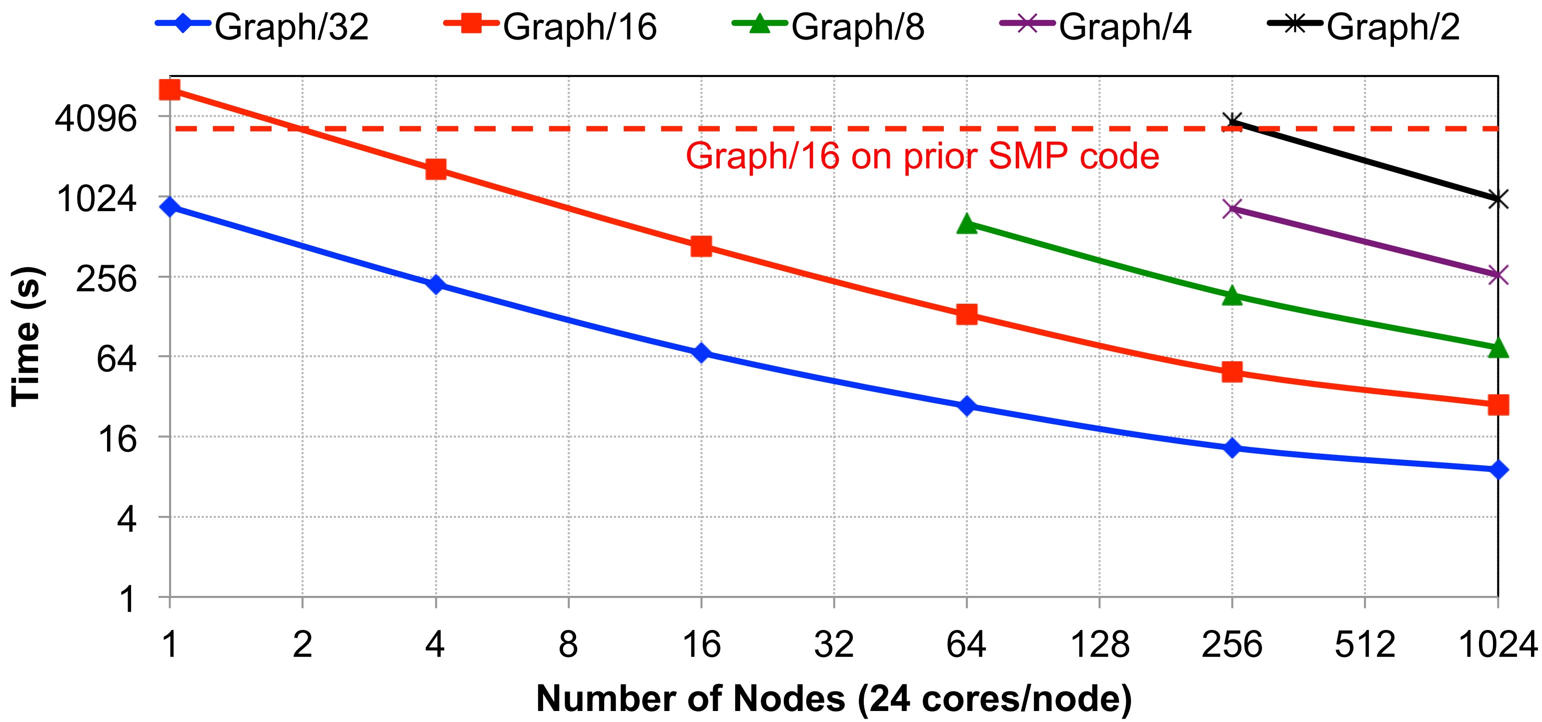
HipMCL work by Aydın Buluç (ECRP) and Ariful Azad

Scalable Distributed Memory, SpGEMM with Thresholding



- Parts of the result is produced and pruned
- Memory requirement can be significantly reduced by increasing k
- However, A is needed to be broadcasted k times
- With k=20: MCL ran on 64 nodes of Cori in about 20 minutes

HipMCL is highly scalable



- Full graph 47M genes (nodes), 10B nonzeros (edges), 1.8 PB
- Projected to take 47 days on previous shared memory code
- 1 hour with HipMCL – 1000x speedup!

Reference: A. Azad, G. Pavlopoulos, C. Ouzounis, N. Kyrpides, and A Buluç. HipMCL: A high-performance parallel implementation of the Markov clustering algorithm for large-scale networks. Nucleic Acids Research, 2018

Using HipMCL for first-ever science

Data	Proteins ($x10^6$)	Edges ($x10^9$)	Clusters ($x10^6$)	time (hr)	Cori KNL nodes
Isolates	70	68	2.9	2.4	2K
Meta-Clust50	282	37	41.5	3.2	2K

- **Science impact:** HipMCL can easily cluster protein similarity networks with 100 billion edges that were impossible to cluster with prior approaches, enabling unprecedented discovery in Biology.
- **HPC impact:** The computational need in biological clustering is reaching exascale.

Reference: A. Azad, G. Pavlopoulos, C. Ouzounis, N. Kyrpides, and A Buluç. HipMCL: A high-performance parallel implementation of the Markov clustering algorithm for large-scale networks. Nucleic Acids Research, 2018

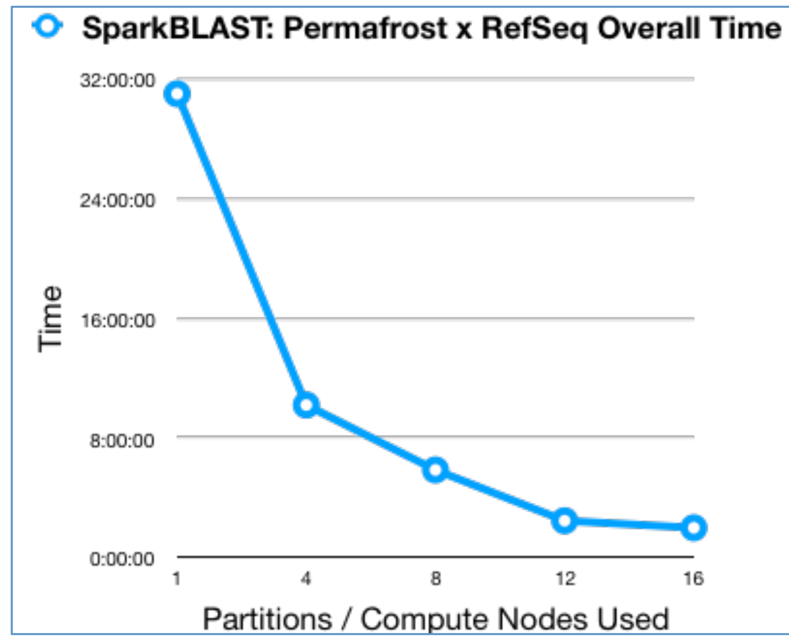
Myth #4: Genome alignment should be done in a cloud or a cheap cluster

Depends on the size of the data, especially the reference

Speeding up sequence comparison across nodes

- BLAST is ~41% of the JGI computing workload
- BLAST is *pleasingly parallel* – can be broken into independent chunks

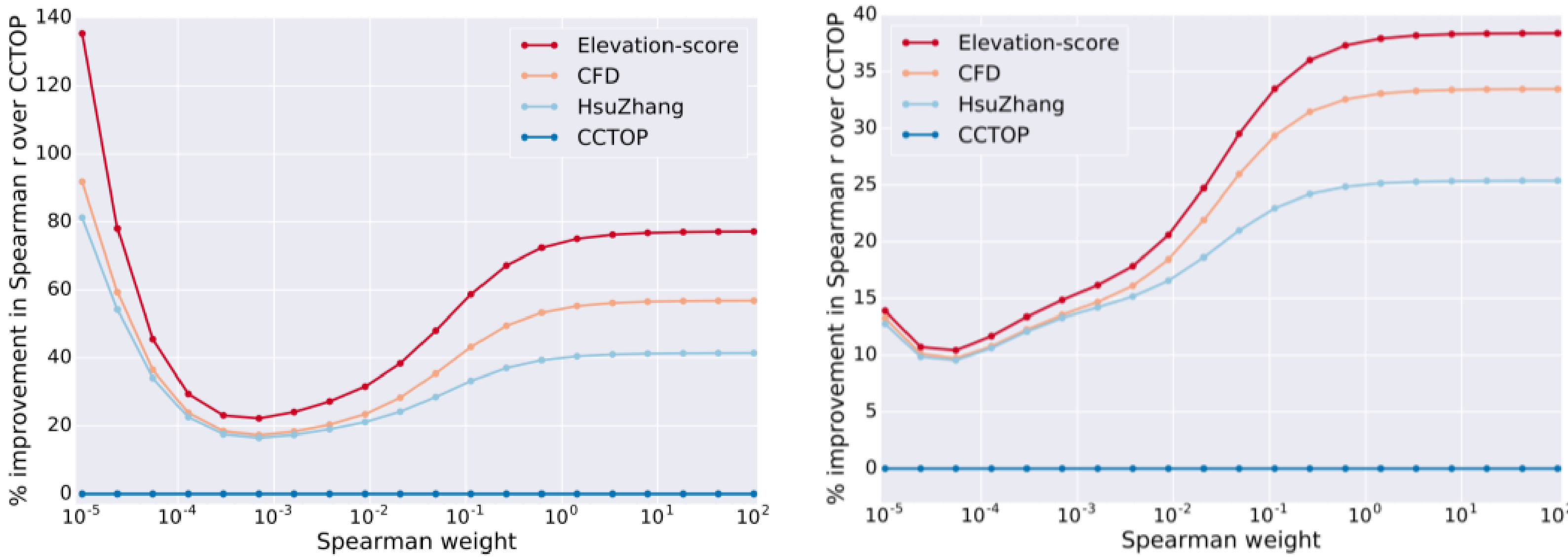
Widely-used tool for distributing compute tasks



SparkBLAST implementation by ***Chris Beecroft***, Data Management Group, Genomic Technologies

Figure: SparkBLAST is ~**15X faster** on 16 nodes than on a single core enabling ***significantly higher throughput***

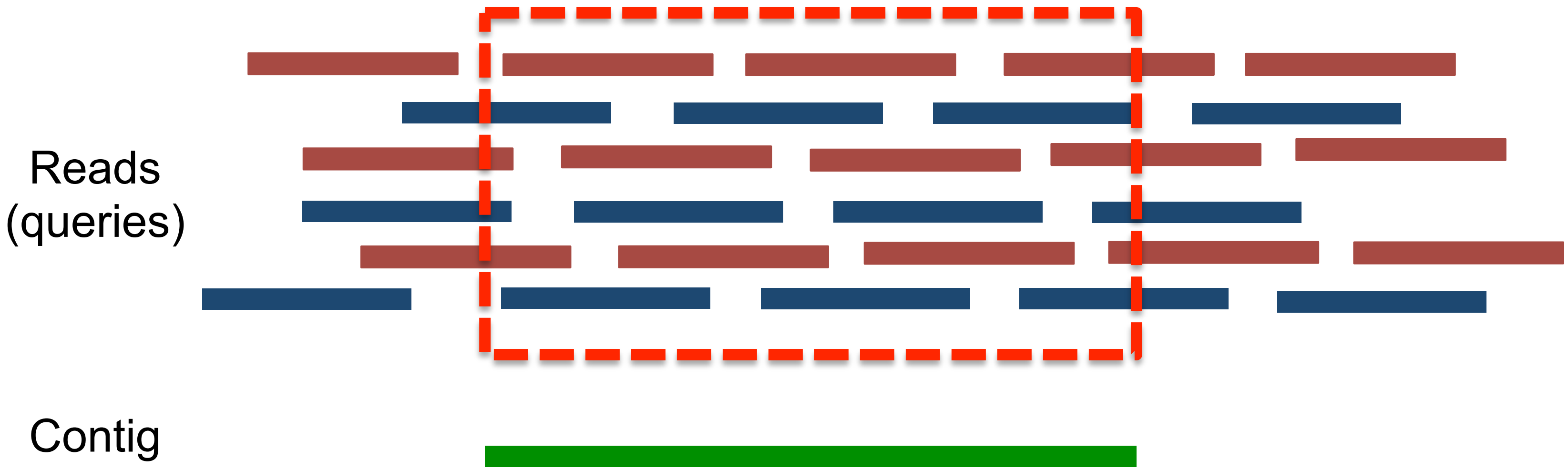
Target and Off-Target CRISPR guide analyses



1. Finds both desired targets and (unintentional) off-targets
2. Used seed-and-extend algorithm (also in HipMer)
   - *Build an index of them using fixed-length seeds*
   - *Locate matches (tandem in CRISPR work)*
   - *Extend (e.g. Smith-Waterman), which could run on GPUs*
3. They use 3 week on 15K cores for 1 human genome and single guide!

J. Listgarten *et al*, Nature Biomedical Engineering (2018)

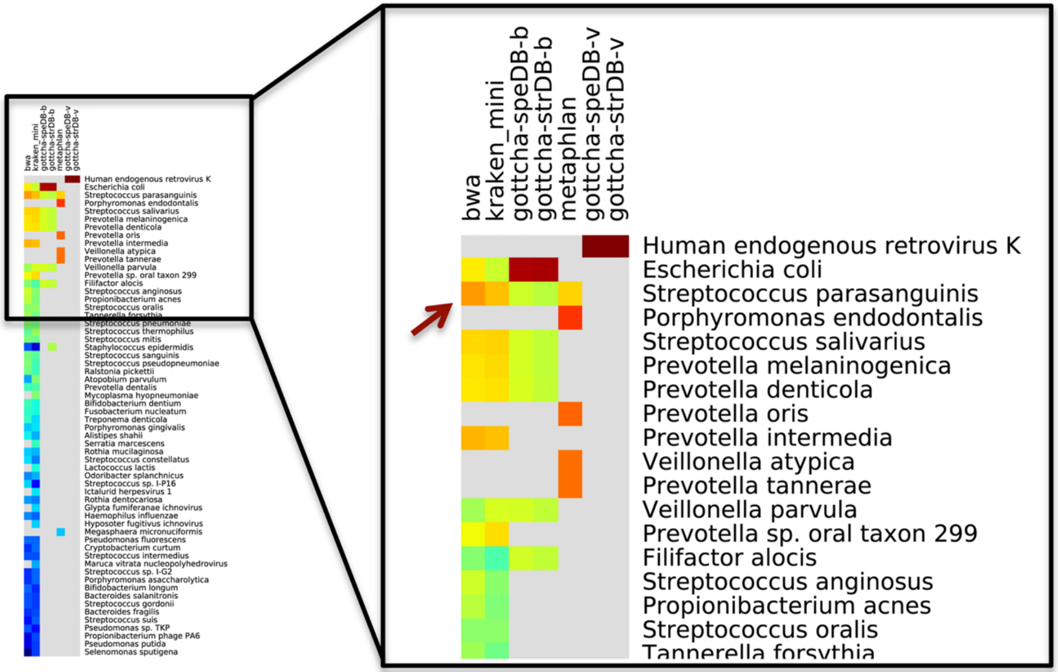
merAligner for large / dynamic references

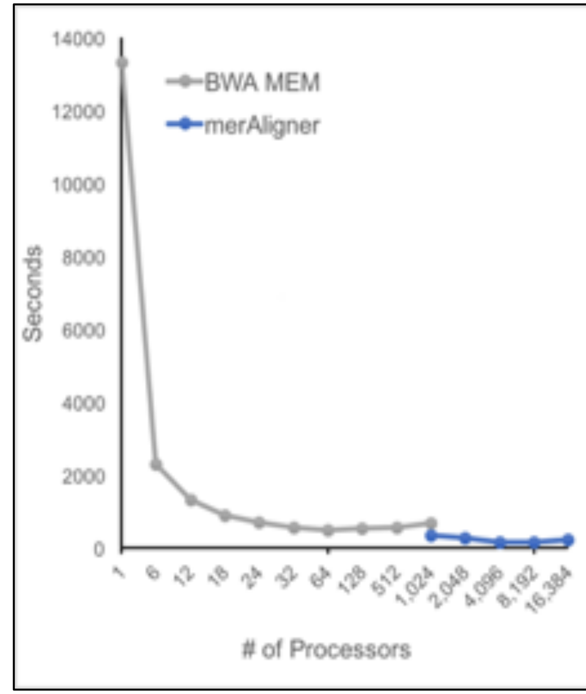


Design philosophy: *merAligner used in HipMer* parallelizes the end-to-end

1. Each processor is assigned a portion of the contigs (reference)
2. Processors build a *global distributed* seed index of the contigs in parallel
    - **Optimization:** Aggregating stores optimization.
3. Each processor is assigned one portion of the reads:
    - Extracts seeds and performs lookups in the *distributed* seed index.
    - Fetches candidate contigs and locally performs alignments

GOTTCHA Metagenome Comparison Tool





- Represents metagenome by taxonomy
  - Expensive all-to-all against database
  - Uses MerAligner (HPC aligner in HipMer)

Patrick Chain's group at LANL; MerAlinger integration by Migun Shakya (LANL) with Steve Hofmeyr (LBNL)

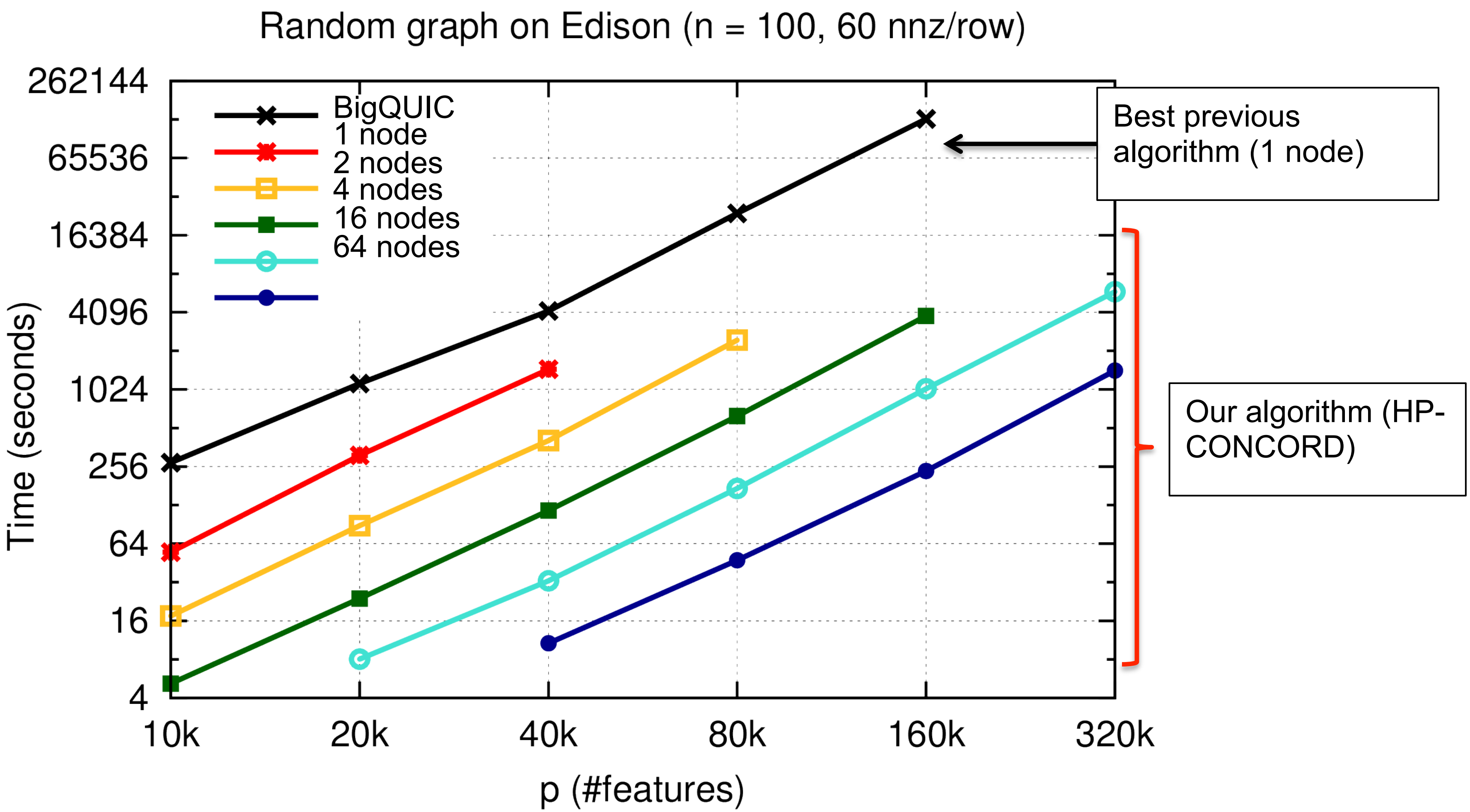
Myth #5: Machine Learning doesn't need HPC

Large-scale statistical models, including both deep and traditional learning can benefit

Learn the relationship between features with Graphical Model Estimator

Source: https://media4.s-nbcnews.com/i/newscms/2017_25/958456/150401-dna-strand-mn-1645_9d74198e59853eb79be3124a876ad4fd.jpg

Communication Avoiding “HP-CONCORD”

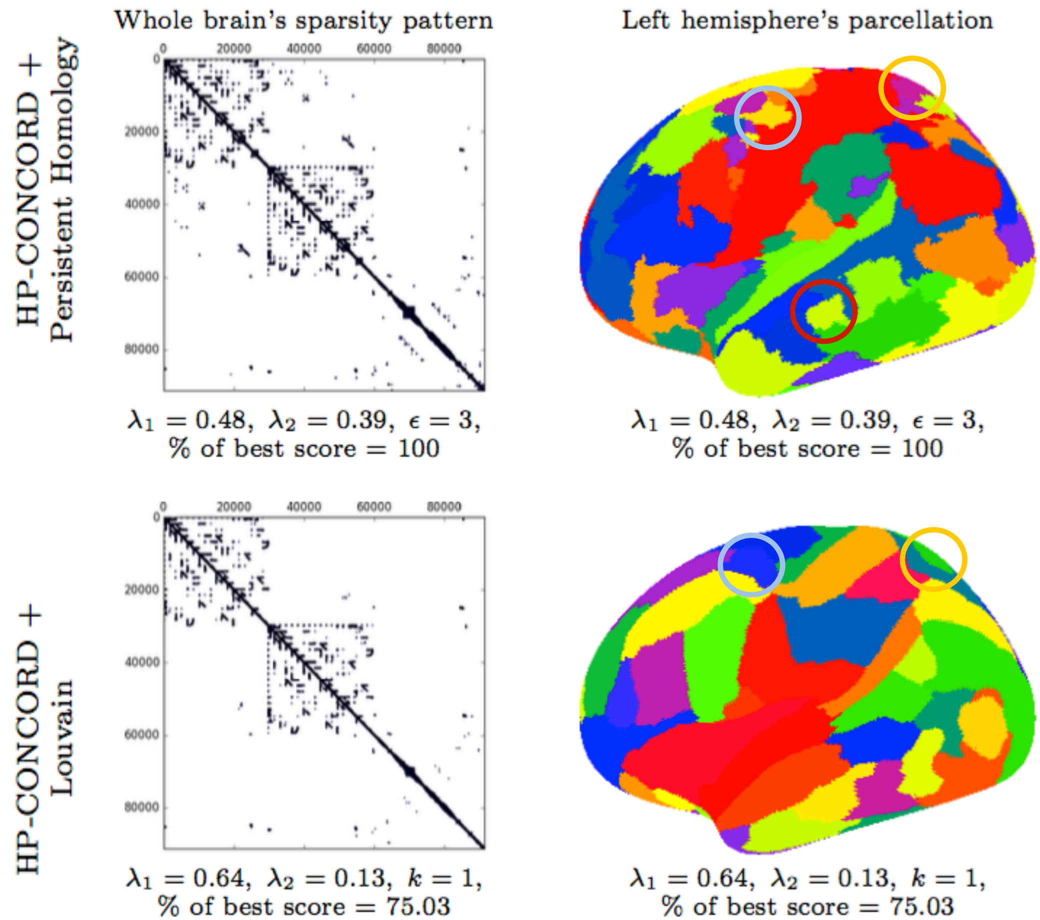


Solve previously intractable problems using clever algorithms and HPC

Discovering regions and co-region

Resting-state fMRI
91K x 91K Sample
Covariance matrix

- 91K data points (2mm x 2mm x 2mm cubes)
- 5K time points (every 0.7 sections for 2 hour)
- Averaged over 1,200 subjects

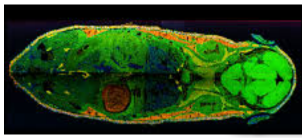


Koanantakool, et al, AISTAT 2018, to appear.

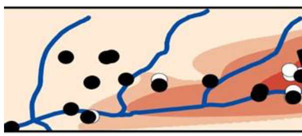
Deep Learning in Bioinformati

Deep Learning in Bioinformatic

Omics

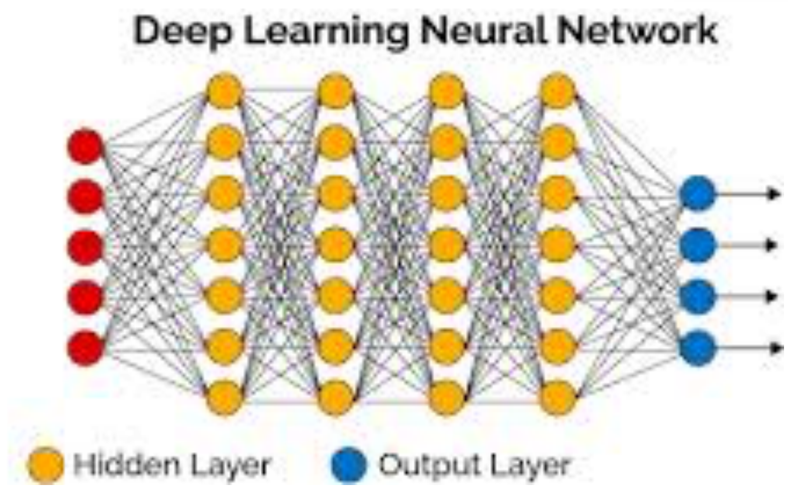


Bioimaging



Signal processing

Protein structure prediction
Gene expression regulation
Segmentation
Brain decoding
Anomaly classification



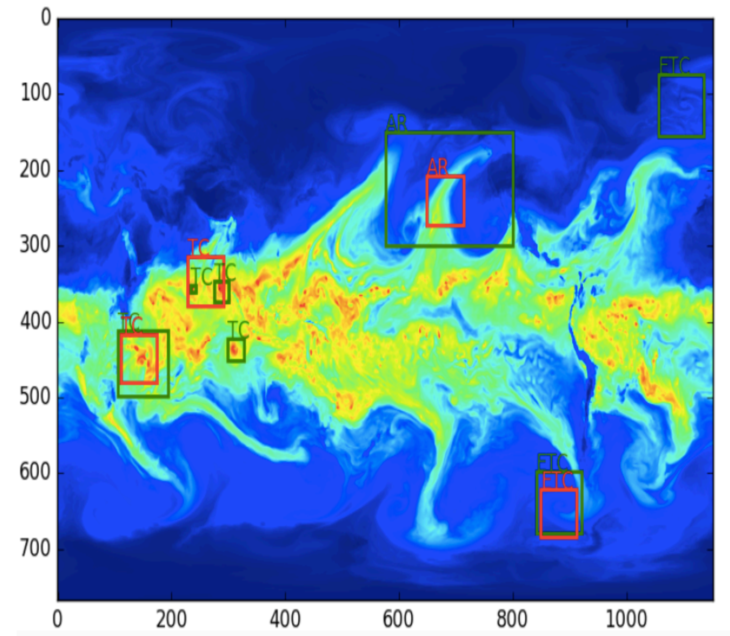
Derived from S. Min et al, Briefings in Bioinformatics

Deep Learning using HPC for Extreme Weather Events

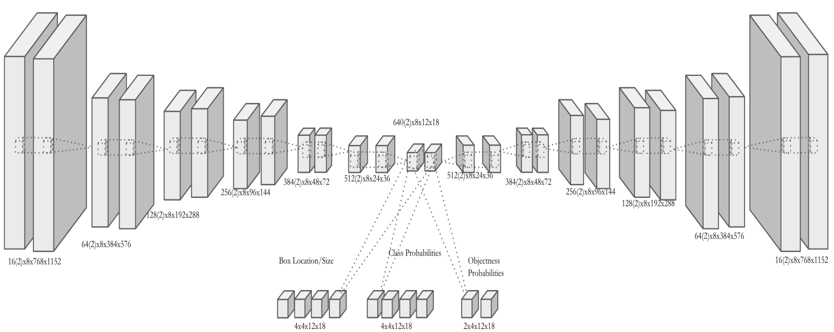
- First application of supervised and semi-supervised architectures for finding patterns in CAM5 data

- DL methods are capable of extracting weather patterns with 85-99% accuracy (NIPS'17 paper)

- Implementation scaled to 15PF on Cori Phase II (SC'17 paper)

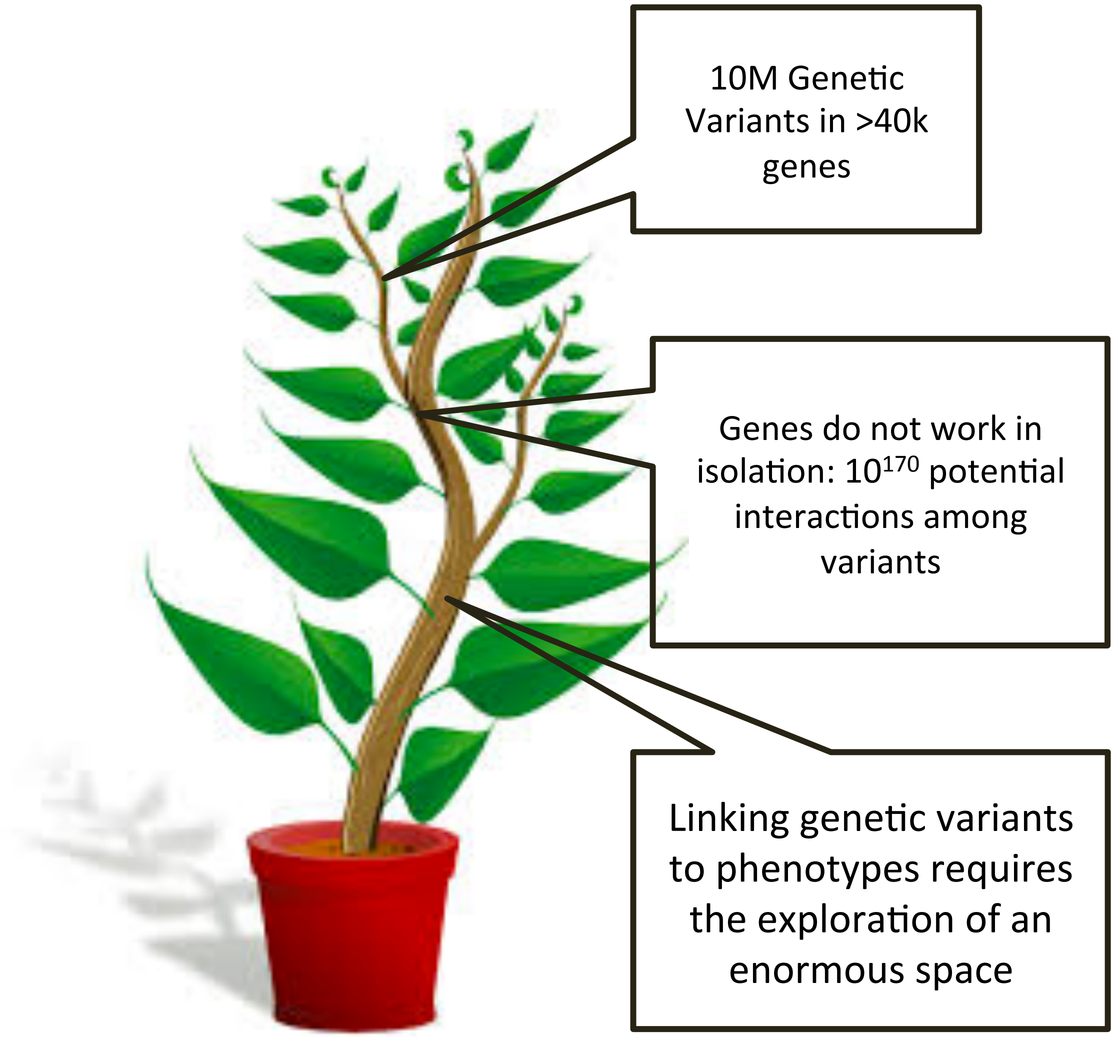


Ground Truth vs Prediction



Prabhat et al

Breaking the curse of dimensionality



To obtain accuracy and insight, we are developing procedures to detect interactions of any form or order at the same computational cost as main effects

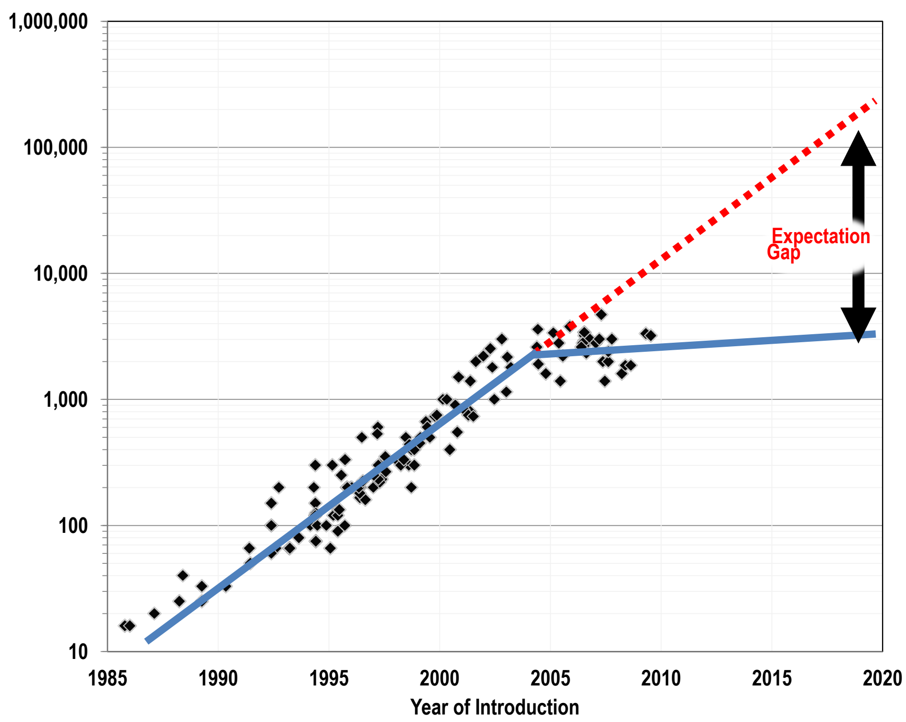
Explainable-AI

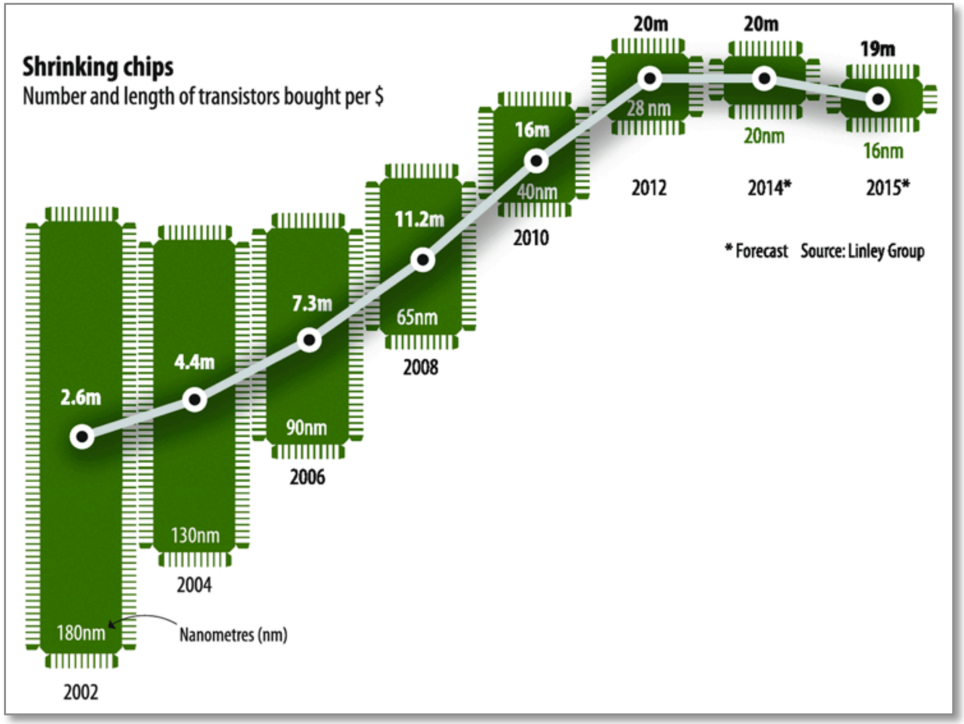
Dan Jacobson/Ben Brown
ORNL/LBNL

Myth #6: Exascale computing is only about building big machines

The Exascale Computing Project is developing novel applications and features, software, and hardware R&D

“Moore’s Law” is Running Out





Clock speed increases have ended

Transistor density is reaching its limit

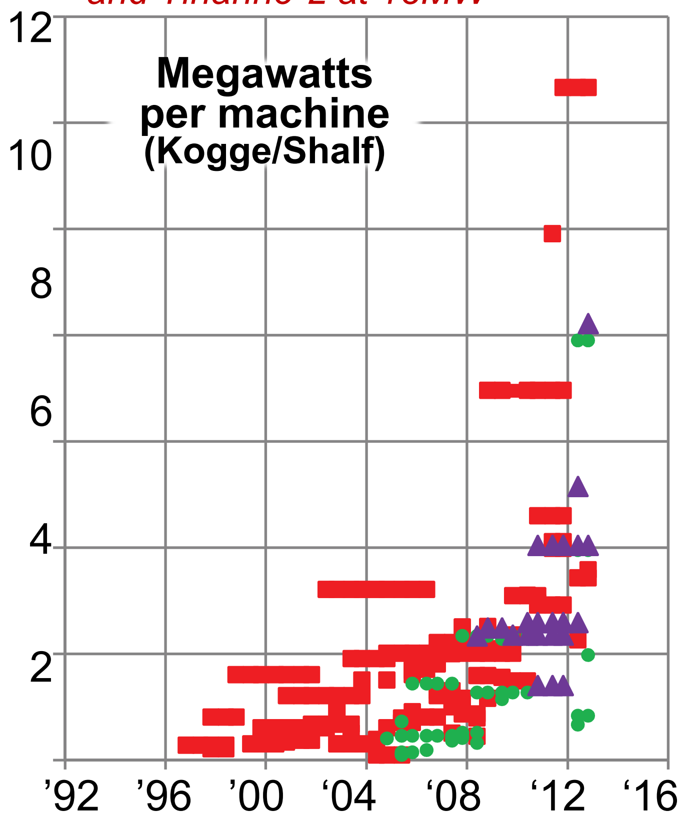
Power and cost of computing are no longer dropping at historic rates

Computing is energy-constrained

At ~$1M per MW, energy costs are substantial

- 1 petaflop in 2008 used 3 MW
- 1 exaflop in 2018 at 200 MW "usual chip scaling"

Missing TaihueLight at 15MW and Tihanhe-2 at 18MW



Goal: 1 Exaflop in 20 MW
= 20 pJ / operation

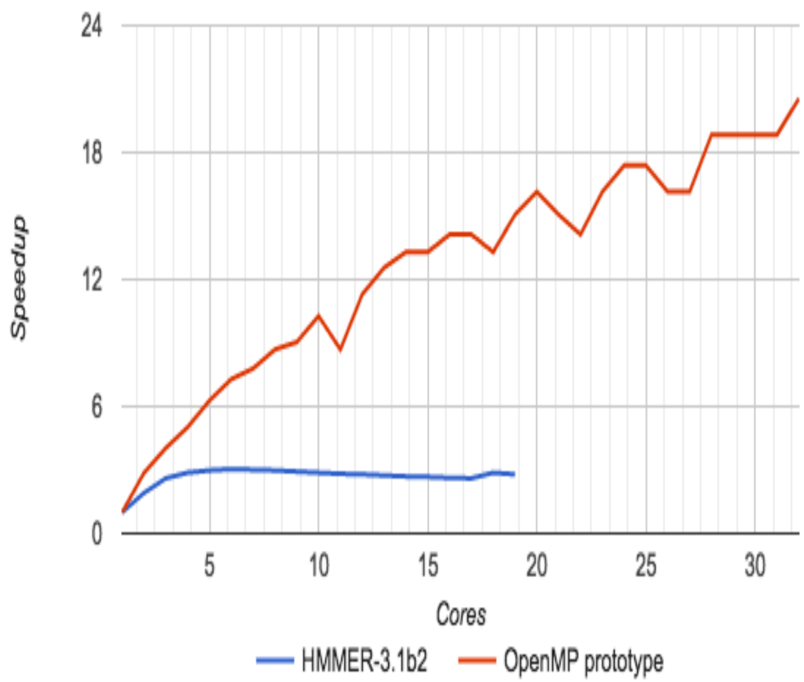
Note: The 20 pJ / operation is

- **Independent of machine size**
- **Independent of # cores used per application**
- **But "operations" need to be useful ones**

Energy Limits Computer Performance?

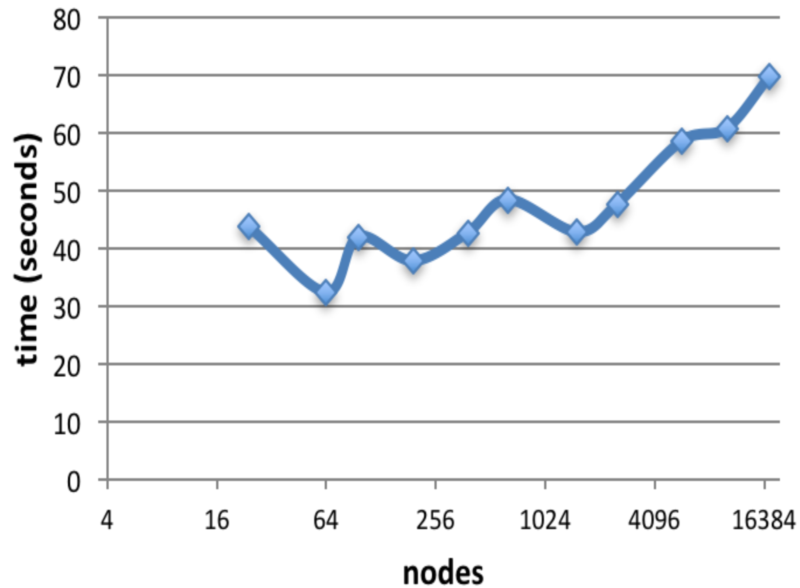
Use of manycore processors and accelerators

KNL manycore architecture for hmmsearch at JGI



Bill Arndt, NERSC

3-way similarity analysis Weak scaling (fixed size problem per node) on Titan w/ GPUs

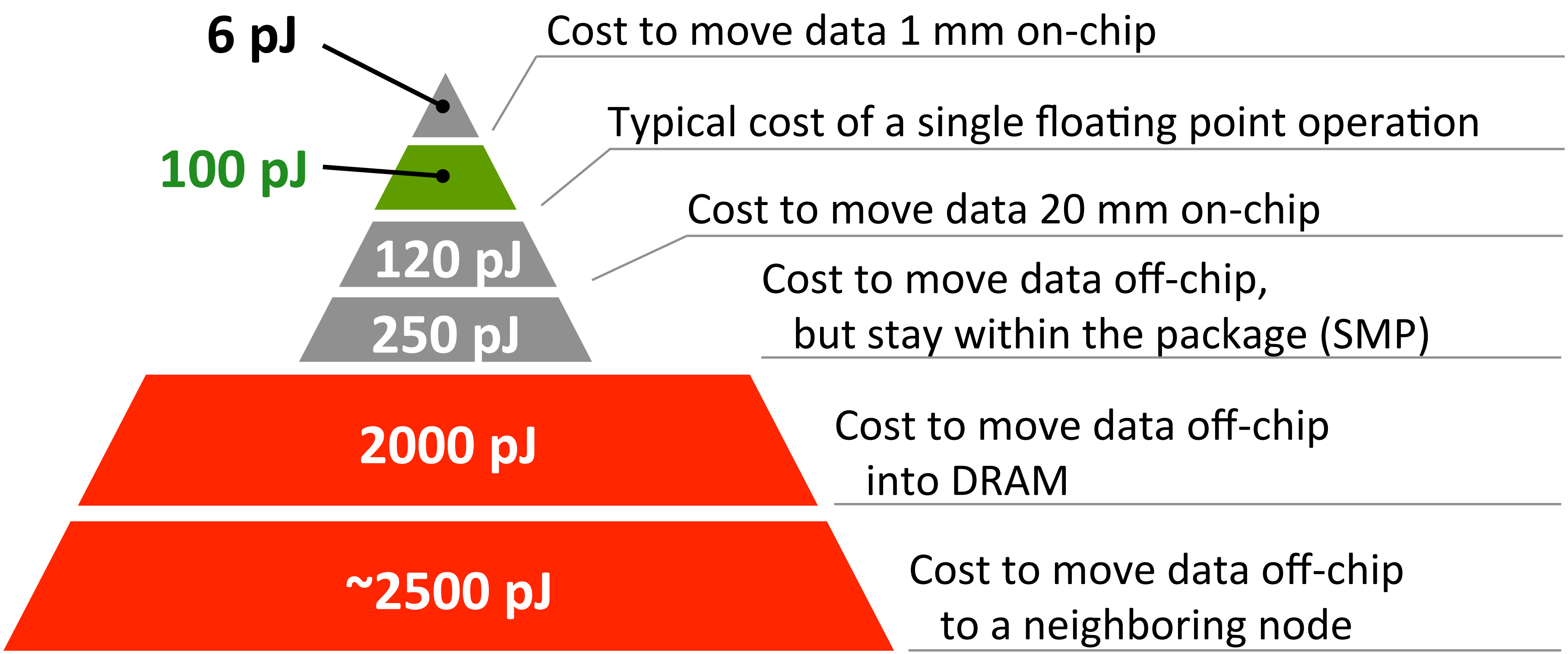


Joubert, Nance, Weighill, Jacobson, ORNL

- Mapping alignment, similarity calculations, etc., to energy efficient manycore and accelerator architectures

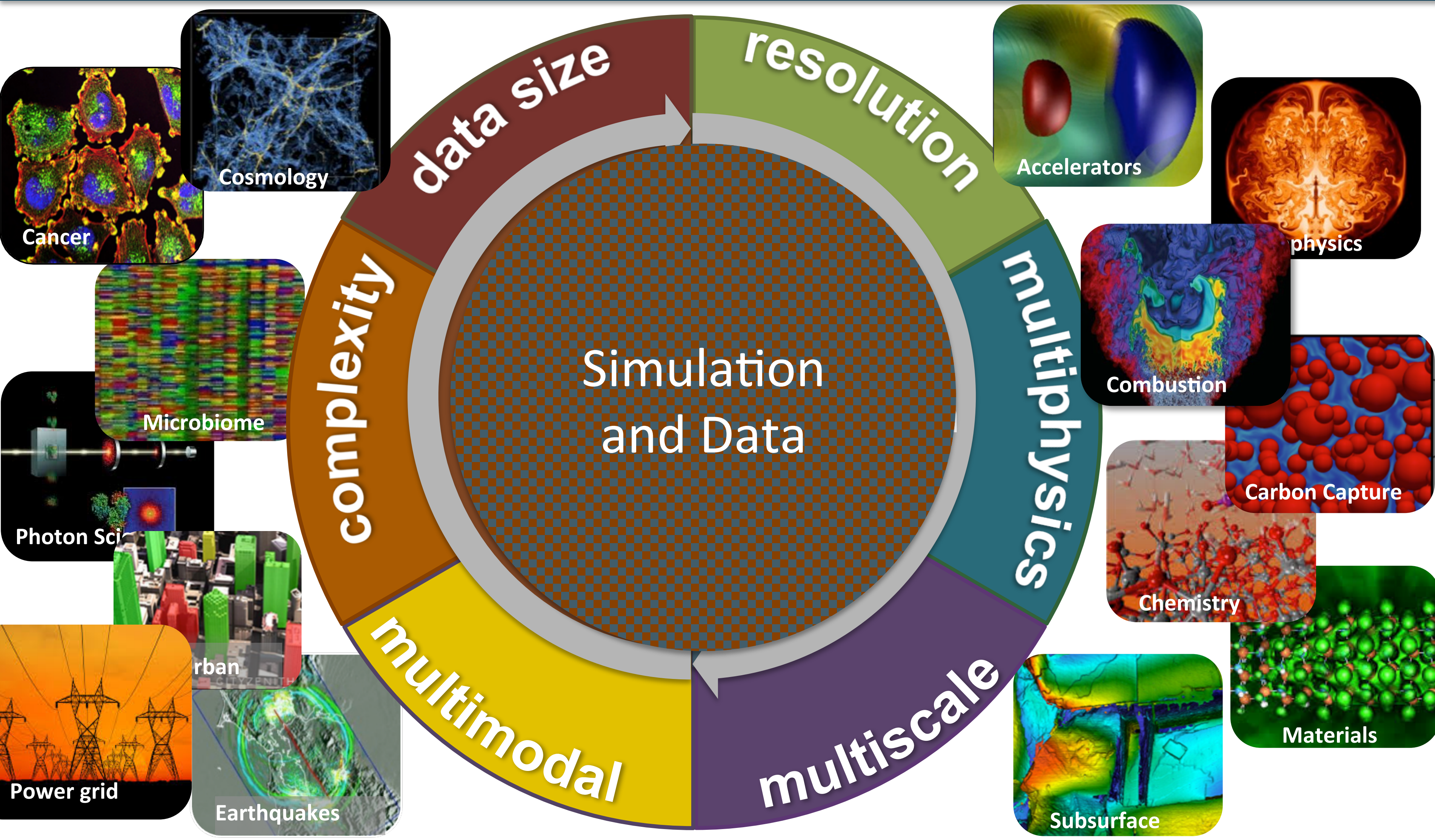
Data Movement is Expensive

Hierarchical energy costs.



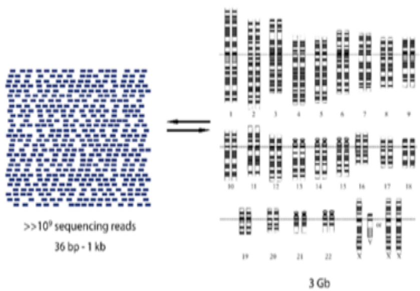
Source: http://slideplayer.com/slide/7541288/

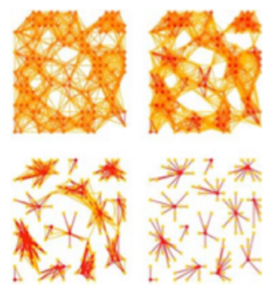
Breakthrough Science Challenges for Exascale

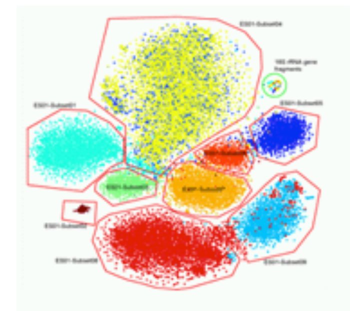


ExaBiome: Exascale Solutions to Microbiome Analysis

- Use HPC algorithms and systems for orders of magnitude speedup and to solve previously intractable problems



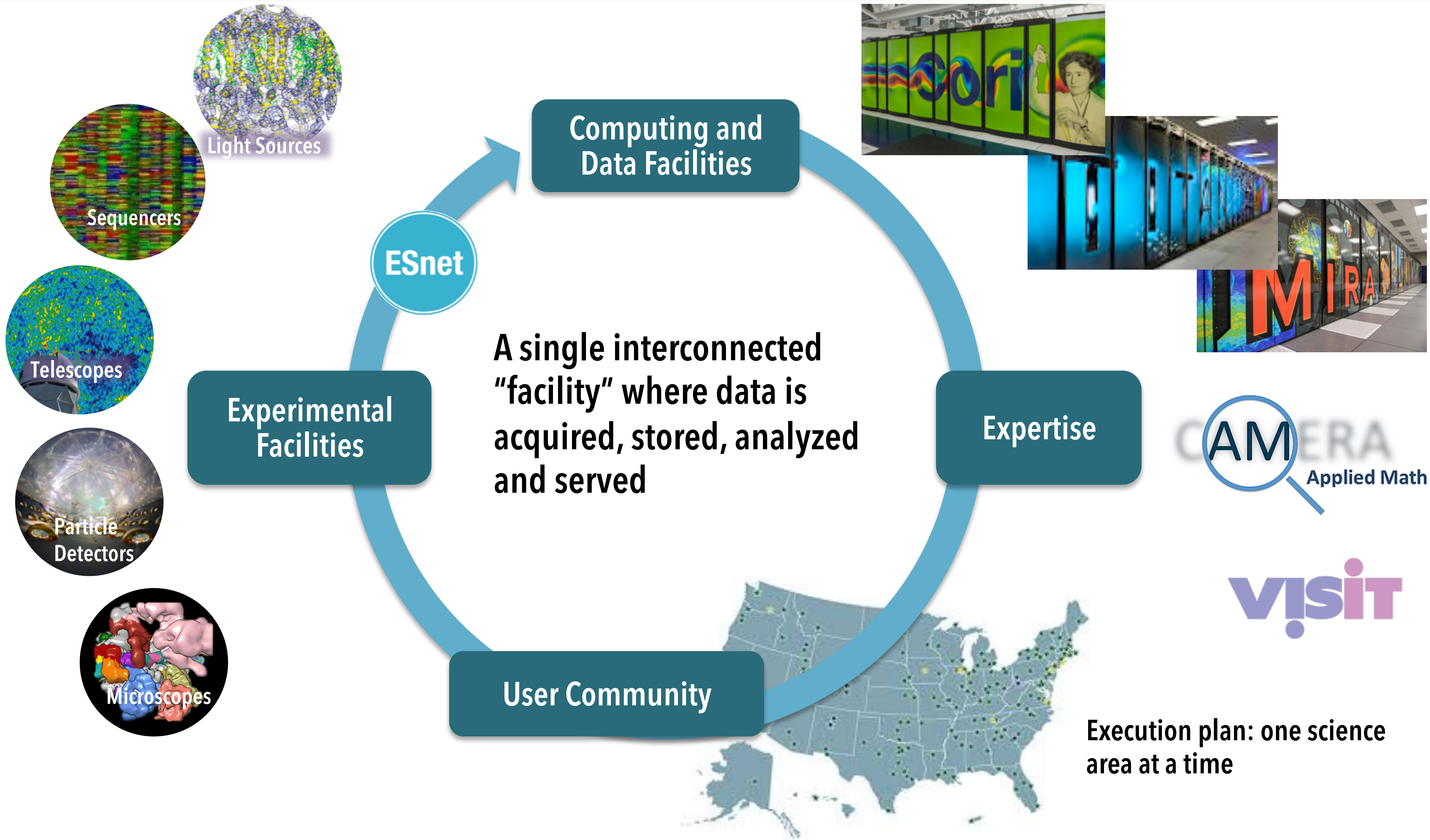




Problem Domain	**Metagenome Assembly**	**Protein Clustering**	**Comparative Analysis**
Exascale goal	Assemble millions of metagenomes from whole data	Cluster billions of proteins	Use fast alignment and annotation for time-sensitive analyses
Computing techniques	Graph algorithms, Hash Tables, alignment (Smith-Waterman)	Machine learning (clustering), sparse linear algebra / graphs	Alignment, Machine learning (dimensionality reduction), linear algebra

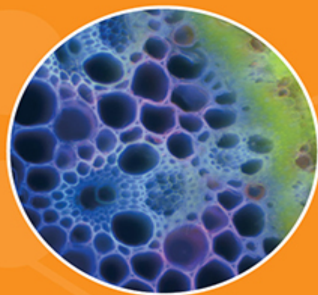
Superfacility: A vision for DOE Science Facilities and leveraging Expertise

Superfacility for Major Experimental Facilities



NERSC / Joint Genome Institute partnership

DOE Mission Areas


Bioenergy

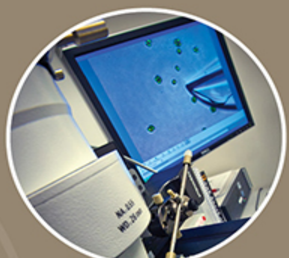

Carbon Cycling


Biogeochemistry

JOINT GENOME INSTITUTE

JGI Infrastructure


DNA Sequencing


Advanced Genomic Technologies

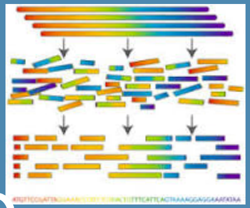

Computational Analysis

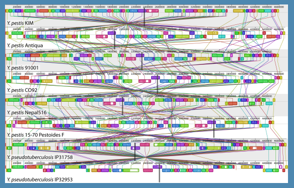

DNA Synthesis

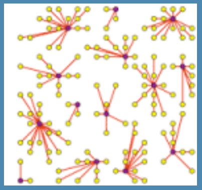
NERSC

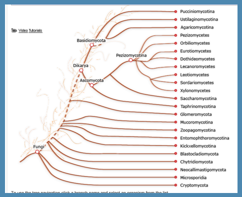
NERSC staff run JGI computing

HPC


Alignment, Assembly


Annotation, Search


Clustering

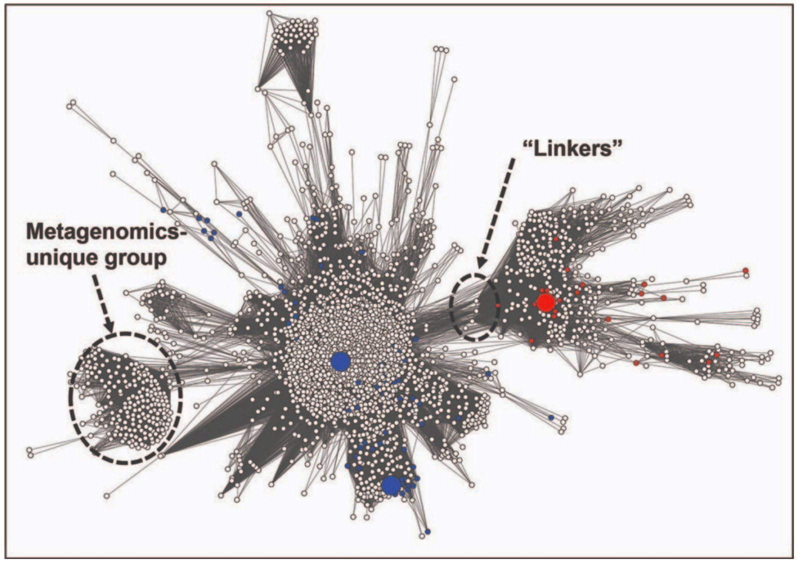

Phylogeny

JAMO
Archave, Metadata

JGI-NERSC Microbiome Data Science FICUS

JGI's metagenomic data and NERSC's Cori supercomputing

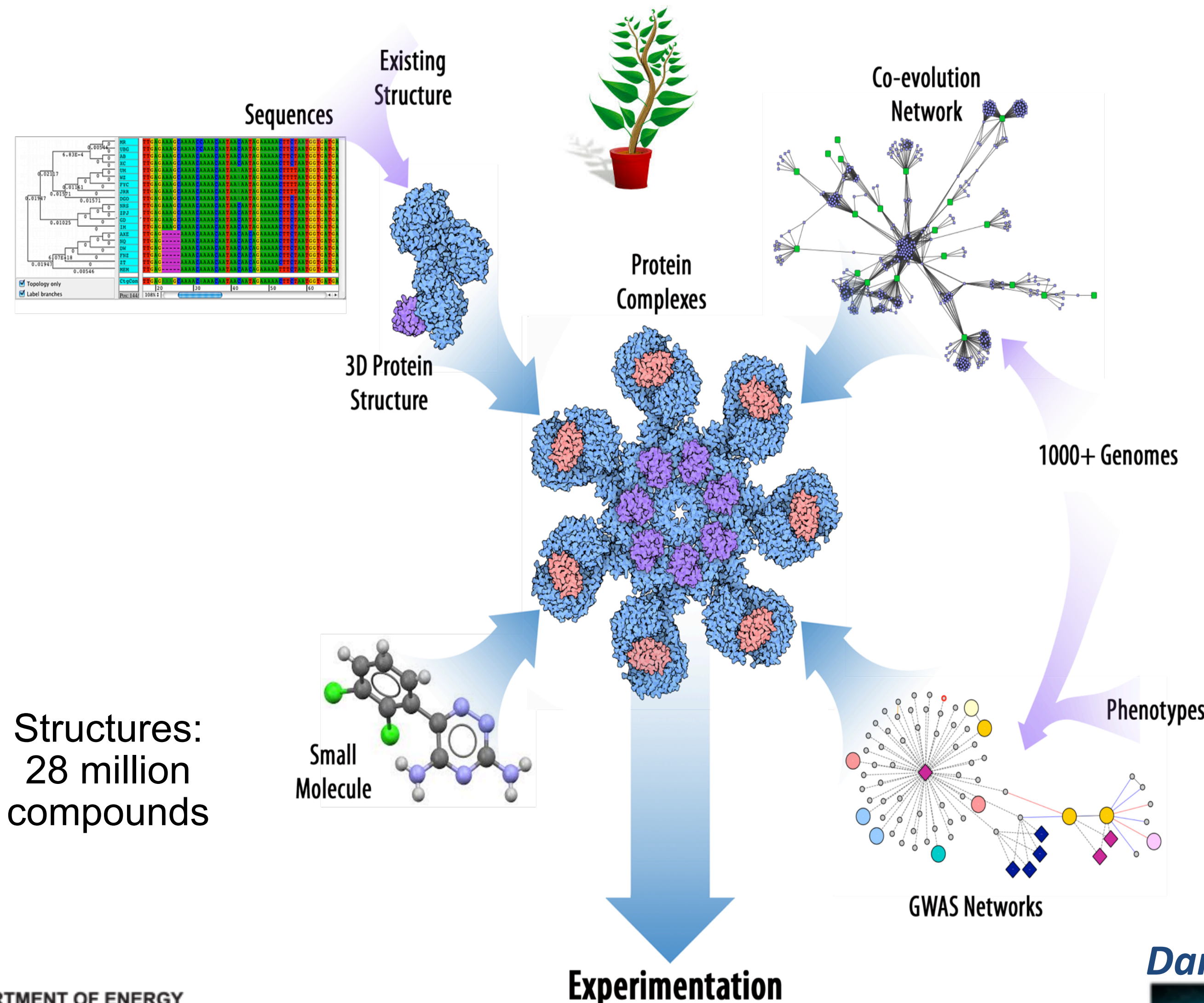
- 6 projects underway
  - Patricia Babbitt *UCSF*
  - David Baker *UW, Seattle*
  - Phillip Brooks *UC Davis*
  - Ed DeLong *UH Manoa*
  - Steve Hallam *UBC Vancouver*
  - Kostas Konstantinidis *Georgia Tech*



A sequence similarity network of a family of enzymes from the nitroreductase superfamily (some nitroreductases can reduce TNT, a significant soil contaminant). Source: Patsy Babbitt

FICUS
Facilities Integrating Collaborations for User Science

From Systems Biology to 3D Structural Interactions



Structures:
28 million
compounds

Dan Jacobson, ORNL

U.S. DEPARTMENT OF ENERGY INCITE LEADERSHIP COMPUTING

TITAN

Kbase: interface to collaborative, reproducible science

In KBase, you can create shareable, reproducible workflows called “Narratives” that include data, analysis steps, results, visualizations and commentary.

A short demonstration of an annotation of a field isolate from a radionuclide contaminated site

What's the problem?

The following is based on an article called "Use of immunomagnetic separation for the detection of Desulfovibrio vulgaris from environmental samples" that appeared in Journal of Microbial Methods in 2011.

Desulfovibrio vulgaris (Dv) is a well-characterized sulfate-reducer known to reduce metals, and has commonly been detected in DOE contaminated sites through genomic tools. D. vulgaris and closely related SRB have been routinely found at the uranium-contaminated groundwater at the Field Research Center (FRC) and the chromium- contaminated site at Hanford, WA (Chakraborty R. ncbi genome). To better comprehend the presence and activity of Dv or Dv- like microorganisms under these non-optimal conditions in-situ,it is imperative to examine the gene expression of these cells separated from their environment with minimal disruption or interference caused by cell processing. As part of our ongoing investigations on the stress and survival of SRB (namely Dv) in the environment (see more at Enigma), we developed and tested a non-destructive method that uses immunomagnetic separation (IMS) of the model sulfate-reducing bacterium, D.vulgaris. Our ultimate goal is to develop a field-deployable version of IMS that enables the detection of target microorganisms from often low biomass environmental samples to be then further processed in various -omics (e.g., transcriptomics and metabolomics) studies to better characterize the metabolic properties.

In this study, using an antibody raised against *Desulfovibrio vugaris Hildenborough* cells were pulled down from a Hanford Groundwater sample taken from the 100H region of the Hanford Reach National Monument.

The organism pulled down from the site using this method and immunostained looks like:

You can find more about Desulfovibrio vulgaris as a species by looking at Wikipedia. But it is a sulfate reducing bacteria, a motile, obligate anaerobe, with an extraordinary number of two-component systems. Here is the standard electron micrograph from Wikipedia.

Here's what I am going to do:

- Upload the genome
- Reannotate it for use in KBase.
- Annotate its domains for completeness
- Place it in a phylogenetic tree
- Compare it to the closest relative
- Try and understand its metabolic differences through comparing metabolic models.

Be aware though, I am not being rigorous here. Just giving a quick tour through KBase functionality for a realistic case.

Upload and examine the data.

I used the data browser upload tab to upload the RCH1 GenBank file to KBase. This creates two data types: The KBase Genome and KBase Contigs Objects. Uploading only took a few seconds and then I dragged the objects that were created from the data pabe to this Narrative to examine them.

Viewer Genome Desulfovibrio.RHC1.Genome 11:24:20, 2/14/2015

Overview Contigs Genes

KBase ID	287089
Name	Desulfovibrio vulgaris RCH1
Domain	Bacteria
Genetic code	11
Source	KBase user upload
Source ID	noid
GC	63.27 %
Taxonomy	
Size	3734357
Number of Contigs	2
Number of Genes	3223

HPC Transformative for Genomics

- Genome analysis is an HPC problem
  – De novo assembly for single genomes, metagenomes, and pan genomes
  – Protein clustering
  – Alignment
  – All-to-all comparisons
  – Statistical machine learning (traditional)
  – Deep learning
- Enables new approaches, new facility models, and new science