Why HPCs hate biologists (and what we're doing about it)

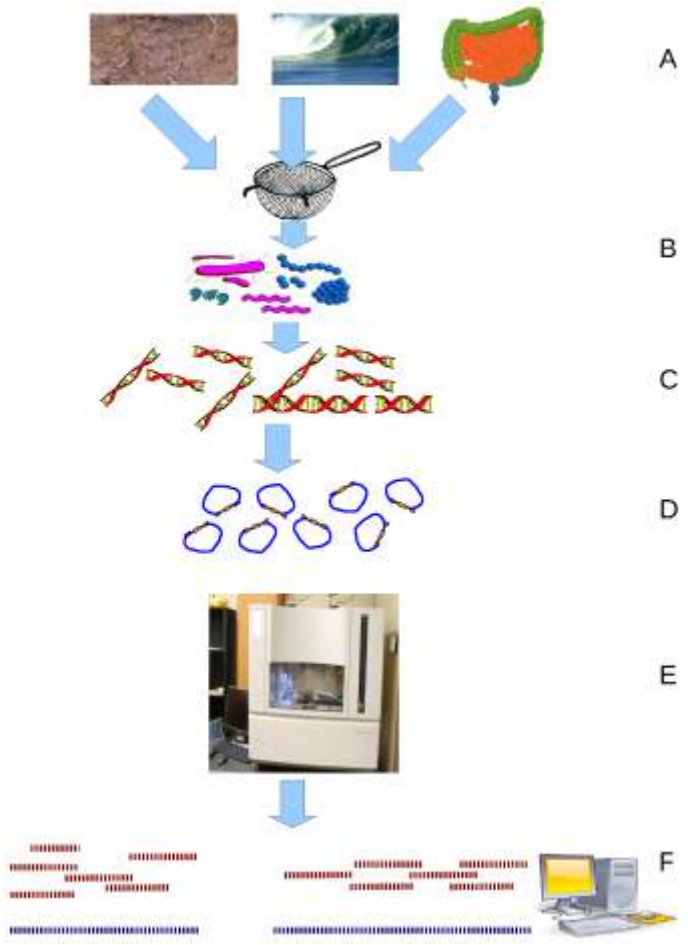
C. Titus Brown
Assistant Professor
CSE, MMG, BEACON
Michigan State University
September 2012
ctb@msu.edu

Outline

- The basic problem(s) – resequencing and assembly
- Data processing & data flow
- Compression approaches
- Some thoughts for the future

Shotgun genomics

- Collect samples;
- Extract DNA;
- Feed into sequencer;
- Computationally analyze.



Wikipedia: Environmental shotgun sequencing.png

Analogy: shredding books

It was the best of times, it was the wor

, it was the worst of times, it was the

isdom, it was the age of foolishness

mes, it was the age of wisdom, it was th

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

…but for lots and lots of fragments!

Sequencers also produce errors...

It was the Gest of times, it was the wor

, it was the worst of timZs, it was the

isdom, it was the age of foolisXness

, it was the worVt of times, it was the

mes, it was Ahe age of wisdom, it was th

It was the best of times, it Gas the wor

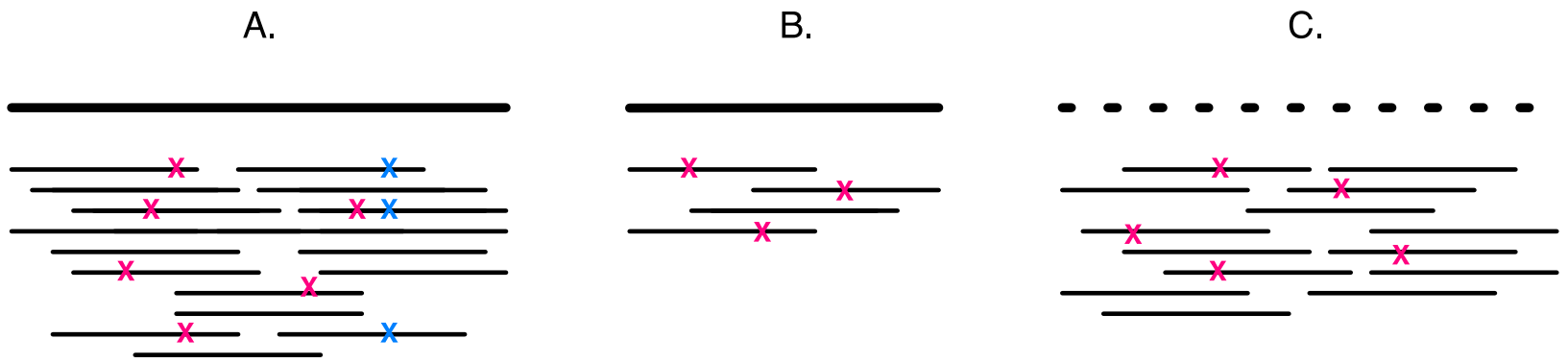
mes, it was the age of witdom, it was th

isdom, it was tIe 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

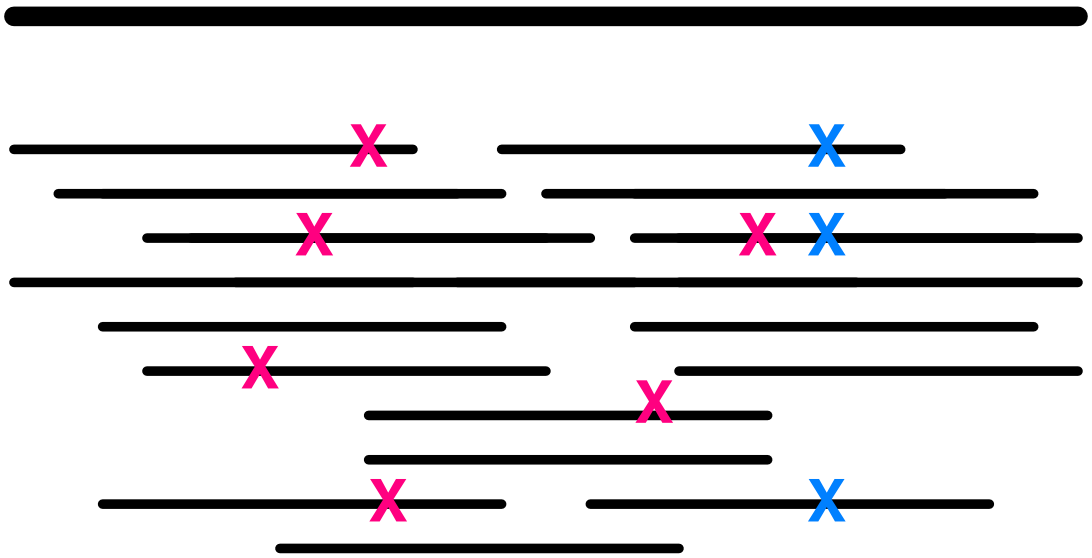
Three basic problems

Resequencing, counting, and assembly.



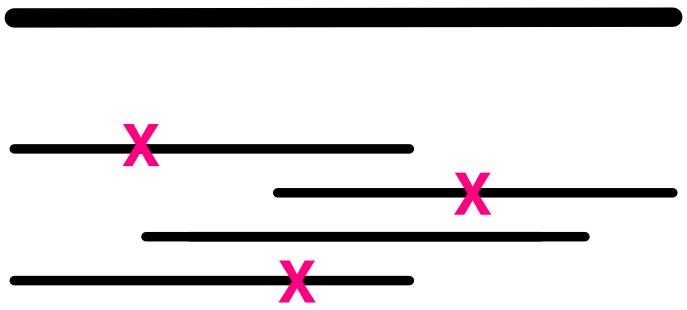
1. Resequencing analysis

We know a reference genome, and want to find *variants* (blue) in a background of errors (red)



2. Counting

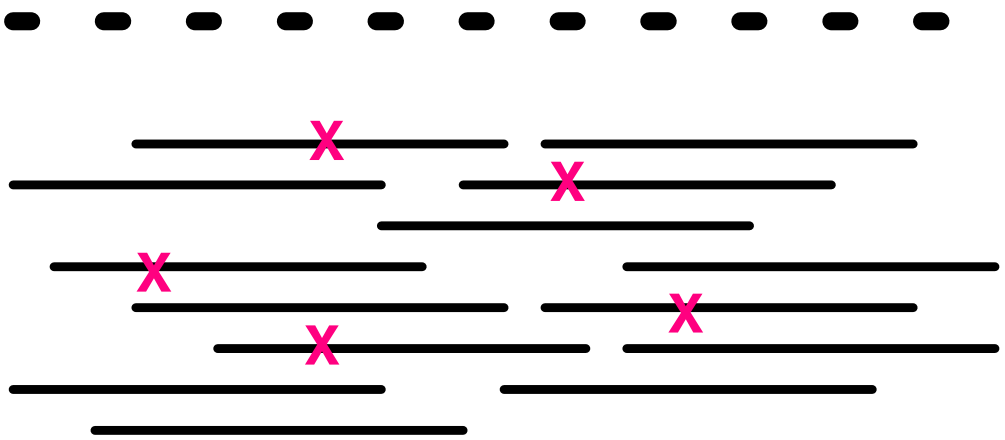
We have a reference genome (or gene set) and want to know how *much* we have. Think gene expression/microarrays.



3. Assembly

We don't have a genome or any reference, and we want to construct one.

(This is how all new genomes are sequenced.)



Noisy observations <-> information

It was the Gest of times, it was the wor

, it was the worst of timZs, it was the

isdom, it was the age of foolisXness

, it was the worVt of times, it was the

mes, it was Ahe age of wisdom, it was th

It was the best of times, it Gas the wor

mes, it was the age of witdom, it was th

isdom, it was tIe 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

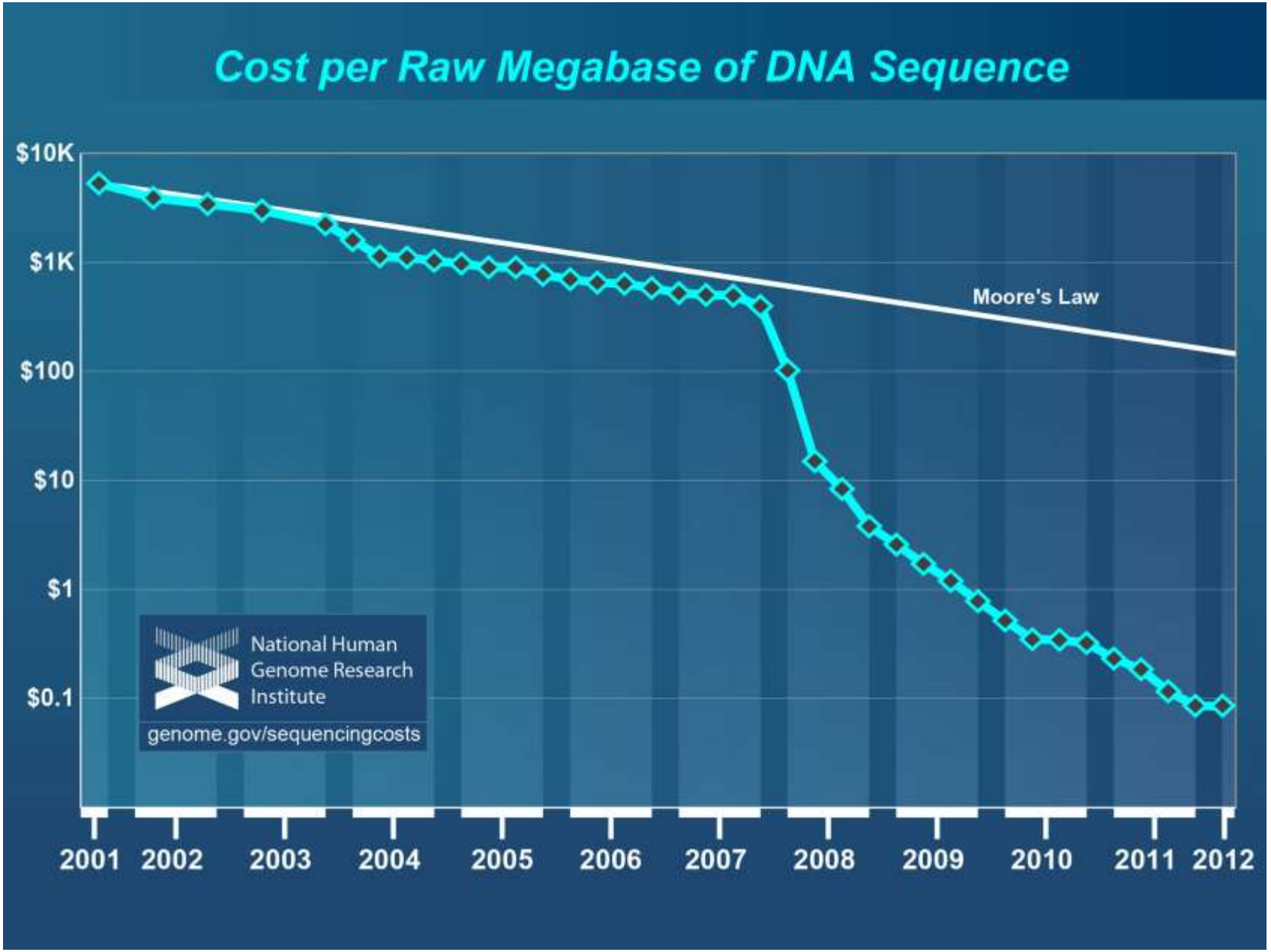
The scale of the problem is *stunning.*

- I estimate a worldwide capacity for DNA sequencing of 15 petabases/yr (it's probably larger).
- Individual labs can generate ~100 Gbp in ~1 week for $10k.
- This sequencing is at a *boutique* level:
  - Sequencing formats are semi-standard.
  - Basic analysis approaches are ~80% cookbook.
  - Every biological prep, problem, and analysis is different.
- **Traditionally, biologists receive no training in computation**
- …*and* our computational infrastructure is optimizing for high *performance* computing, not high *throughput*.

“Three types of data scientists.”

(Bob Grossman, U. Chicago, at XLDB)

1. Your data gathering rate is *slower* than Moore’s Law.
2. Your data gathering rate *matches* Moore’s Law.
3. Your data gathering rate *exceeds* Moore’s Law.



http://www.genome.gov/sequencingcosts/

“Three types of data scientists.”

1. Your data gathering rate is *slower* than Moore’s Law.

   => Be lazy, all will work out.

2. Your data gathering rate *matches* Moore’s Law.

   => You need to write good software, but all will work out.

3. Your data gathering rate *exceeds* Moore’s Law.

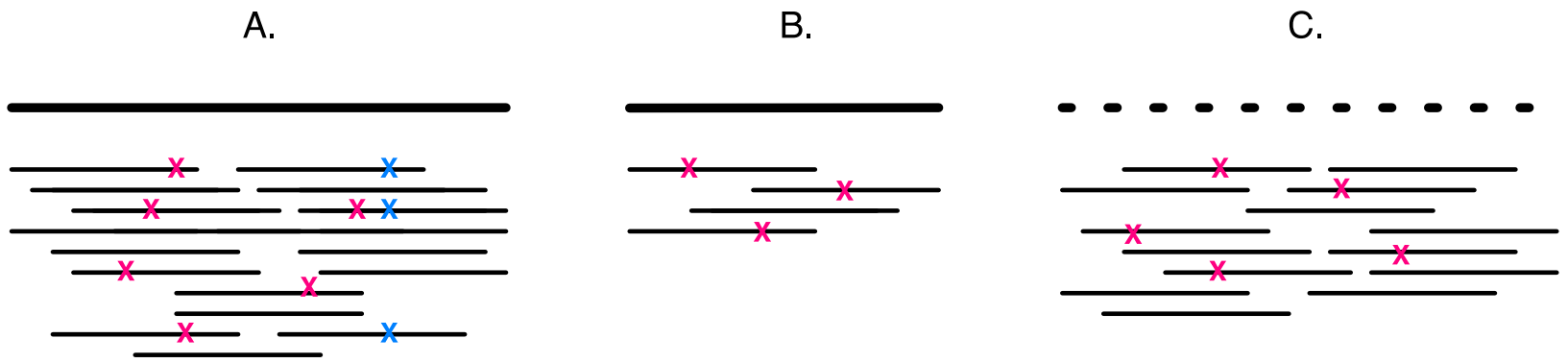
   => You need serious help.

A few use cases

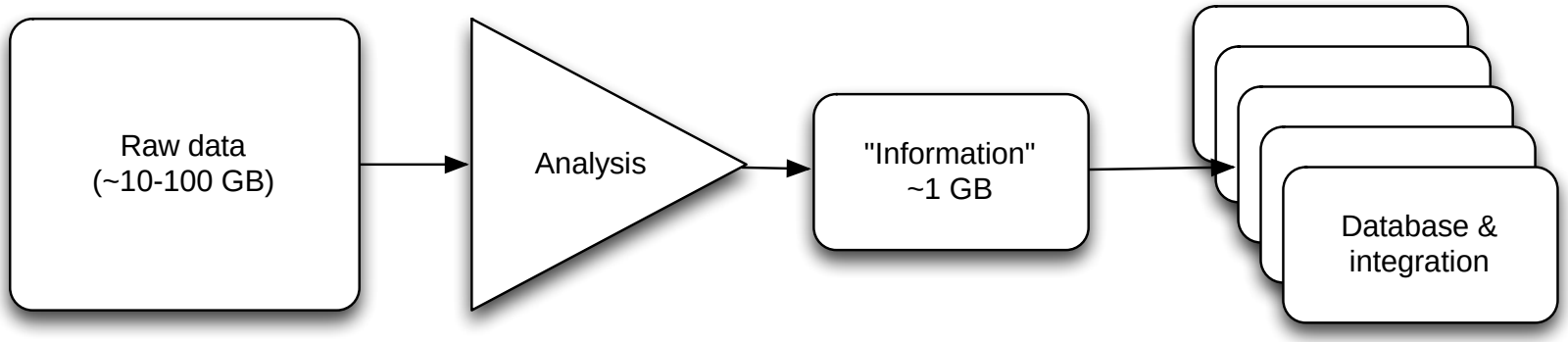
1. Real-time pathogen analysis
2. Cancer genome analysis => diagnosis & treatment regimen
3. Evolution of drug resistance in HIV
4. Gene expression analysis in agricultural animals
5. Examining microbial community change in response to agriculture and/or global climate change
6. Gene discovery & genome sequencing in non-model organisms

Three basic problems

Resequencing, counting, and assembly.



~2 GB – 2 TB of single-chassis RAM



A few interesting computational challenges:

- How do we store raw data for (1) provenance and (2) reanalysis?
- What kind of infrastructure (hardware/software) is the "right" approach?
- Can we reduce the cost of analysis?

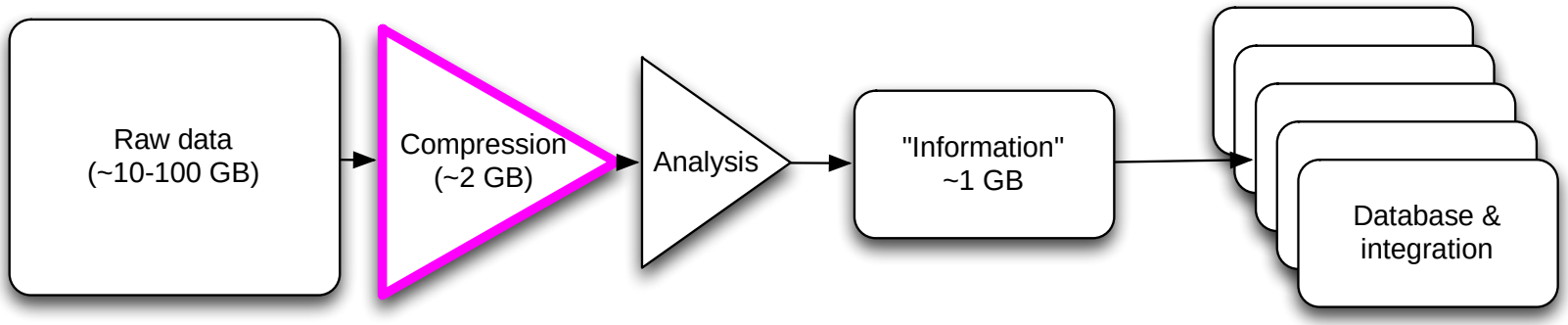
A few use cases

1. Real-time pathogen analysis
2. Cancer genome analysis => diagnosis & treatment regimen
3. Evolution of drug resistance in HIV
4. Gene expression analysis in agricultural animals
5. Examining microbial community change in response to agriculture and/or global climate change
6. Gene discovery & genome sequencing in non-model organisms

What kind of approaches?

- Hardware solutions may not be appropriate
  - *Everyone* in biology/biomedical informatics has or will have these data sets; need "commodity" solutions (=> ~cloud?)
  - …but current commodity hardware is optimized for processing power, while memory and I/O is expensive.

- Are there algorithmic approaches with which we can apply leverage?



A software & algorithms approach: can we develop ***lossy*** compression approaches that

1. Reduce data size & remove errors => efficient processing?
2. Retain all "information"? (think JPEG)

If so, then we can store only the compressed data for later reanalysis.

Short answer is: yes, we can.

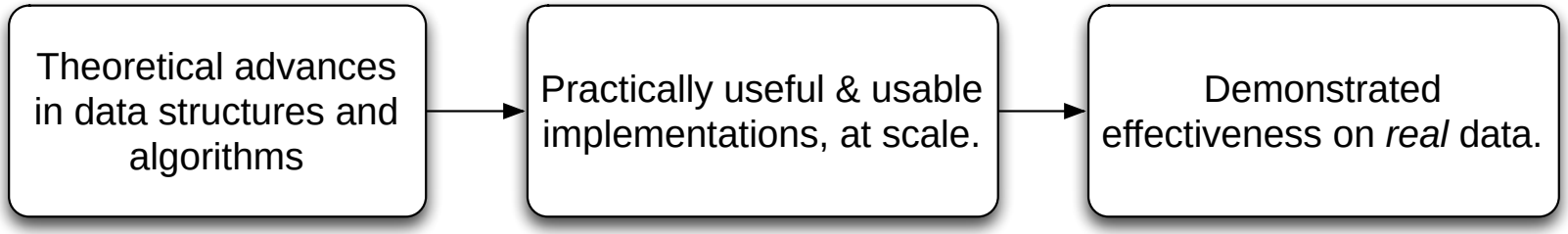
My research driven by *my* problems...

- Est ~50 Tbp to comprehensively sample the microbial composition of a gram of soil.
- Currently we have approximately 2 Tbp spread across 9 soil samples, for one project; 1 Tbp across 10 samples for another.

- Need 3 TB RAM on single chassis to do assembly of 300 Gbp.
- …estimate 500 TB RAM for 50 Tbp of sequence.

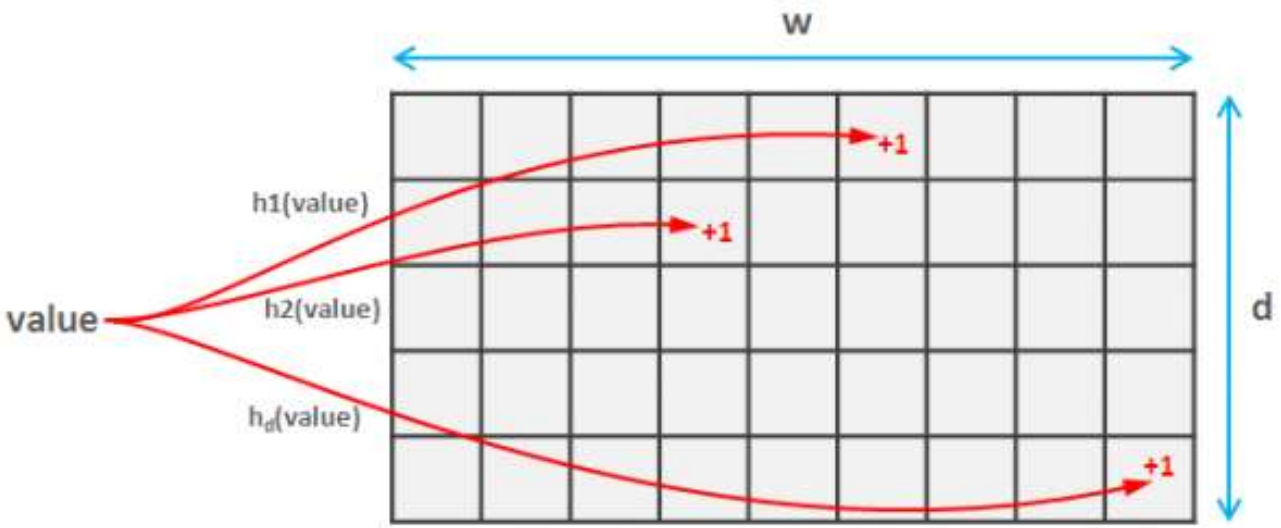
As it turns out, if we can solve that problem, we can solve the rest ☺

My lab at MSU:
Theoretical => applied solutions.



1. CountMin Sketch

To add element: increment associated counter at all hash locales
To get count: retrieve minimum counter across all hash locales



http://highlyscalable.wordpress.com/2012/05/01/probabilistic-structures-web-analytics-data-mining/

(NOVEL)

2. Online, streaming, lossy compression.

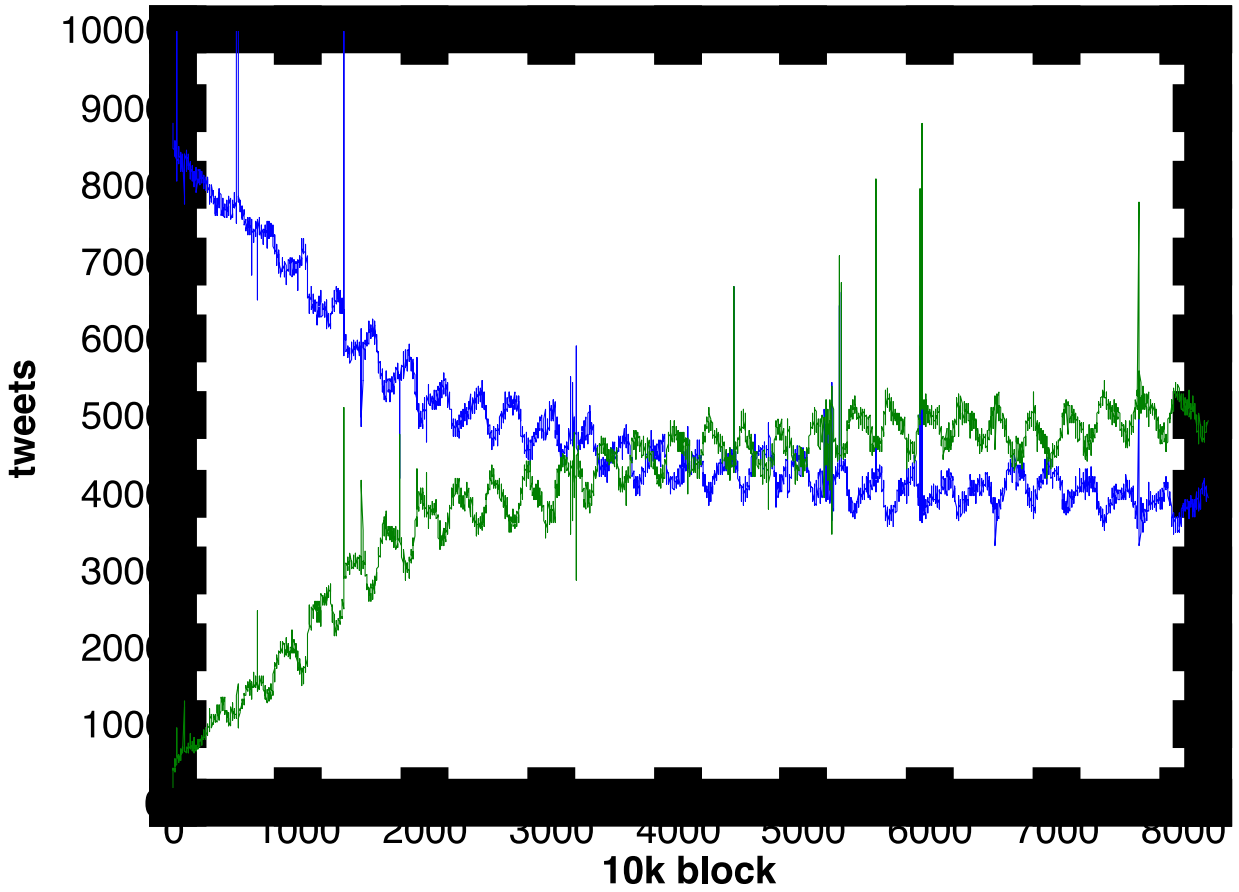
Table 3. Three-pass digital normalization reduces computational requirements for contig assembly of genomic data.

Data set	N reads pre/post	Assembly time pre/post	Assembly memory pre/post
E. coli	31m / 0.6m	1040s / 63s (16.5x)	11.2gb / 0.5 gb (22.4x)
S. aureus single-cell	58m / 0.3m	5352s / 35s (153x)	54.4gb / 0.4gb (136x)
Deltaproteobacteria single-cell	67m / 0.4m	4749s / 26s (182.7x)	52.7gb / 0.4gb (131.8x)

- Transcriptomes, microbial genomes incl MDA, and most metagenomes can be assembled in under 50 GB of RAM, with identical or *improved* results.

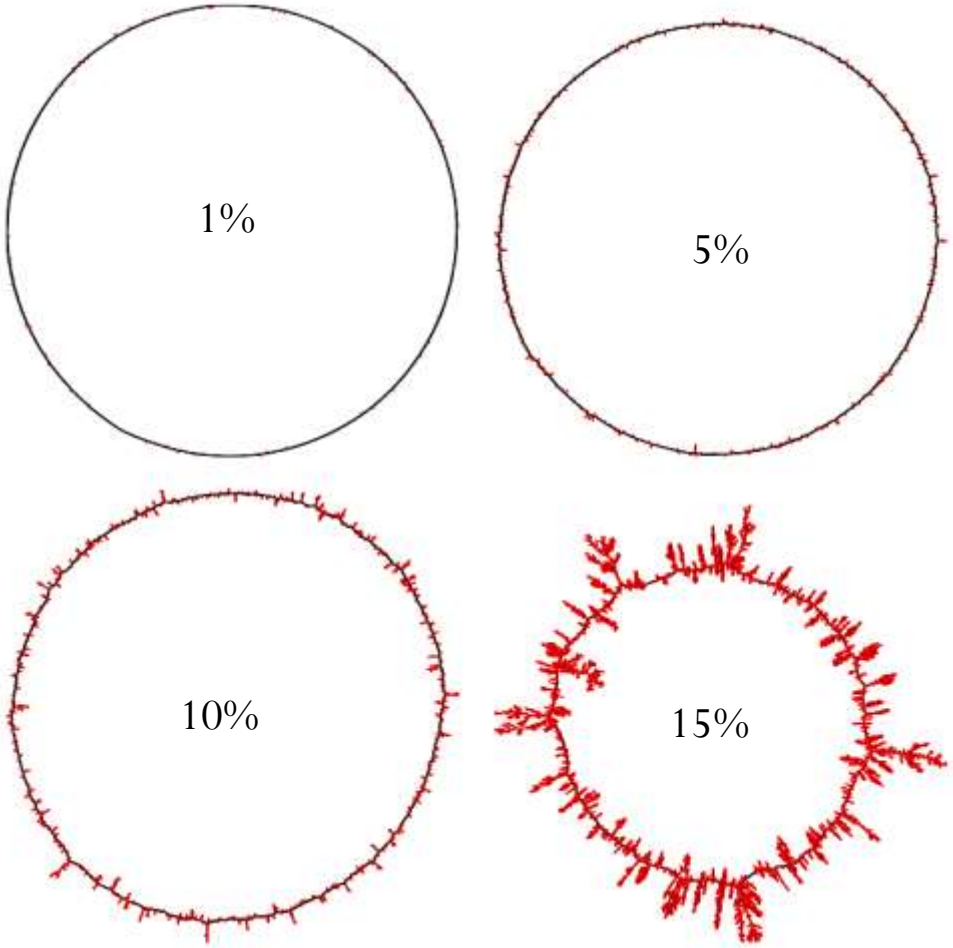
- Core algorithm is single pass, "low" memory.

Streaming Twitter analysis.



(NOVEL)

3. Compressible de Bruijn graphs



Jason Pell & Arend Hintze

Concluding thoughts

- Our approaches provide significant and substantial *practical* and *theoretical* leverage to some really challenging problems.
- They provide a path to the future:
  - Many-core compatible; distributable?
  - Decreased memory footprint => cloud computing can be used for many analyses.
- They are *in use*, ~dozens of labs using digital normalization.
- …although we're still in the process of publishing them.

There is nothing up my sleeves.

Everything discussed here:

- Code: github.com/ged-lab/ ; BSD license
- Blog: http://ivory.idyll.org/blog ('titus brown blog')
- Twitter: @ctitusbrown
- Grants on Lab Web site: http://ged.msu.edu/interests.html
- Preprints: on arXiv, q-bio:
  'diginorm arxiv'

…and I welcome e-mails, ctb@msu.edu

Thank you for the invitation!

Acknowledgements

Lab members involved

- **Adina Howe (w/Tiedje)**
- **Jason Pell**
- **Arend Hintze**
- **Rosangela Canino-Koning**
- **Qingpeng Zhang**
- Elijah Lowe
- Likit Preeyanon
- Jiarong Guo
- **Tim Brom**
- Kanchan Pavangadkar
- **Eric McDonald**

Collaborators

- Jim Tiedje, MSU
- Billie Swalla, UW
- Janet Jansson, LBNL
- Susannah Tringe, JGI

Funding

USDA NIFA; NSF IOS; BEACON.