Data-intensive computing in NGS

Luca Pireddu

Distributed Computing Group

June 6, 2013
Data-intensive computing

What is data-intensive computing?

- a class of parallel computing
- handling large quantities of data (frequently terabytes or more)
- typically data-parallel, performing the same operation to all data
  - kinda like a typical sequencing workflow
In recent years there has been a steady increase in the amount of digitized data available. More, cheaper capacity mainly thanks to improvement of technology and falling price of hardware. More, bigger, cheaper sources, sometimes enabled by technology itself; e.g.,
- larger scientific experiments, such as the LHC and sequencing
- Internet
  - WWW contents
  - Data recorded from users (anything “social”)
- RFIDs, consumer-level transactions (fidelity cards)
- etc.
Data-driven business

- This self-feeding cycle has brought about a number of well-known data-driven businesses.
- They collect data, process it and sell the result in some way.

- Google apparently processed 24 PB/day in 2009.
  - That’s about 20,000 Illumina run directories…per day!
How can do they do that?
The requirements of data-driven activities have spurred innovations in data-processing techniques

- Scale horizontally, using lots of machines
- Write software that accepts hardware failure
- Use commodity hardware
- Spread the data
  - split it into parts
  - distribute them on the processing nodes
- Move the computation to the data

*Note that this clashes with how sequencing data is typically processed*
But how is this useful for NGS?
But how is this useful for NGS?

- Actually, much of this technology has been released into the open
Hadoop ecosystem

- Hadoop is the most popular open source project for processing big data
- Includes distributed components for
  - storage
  - processing
  - query
  - workflows
- Not a second-class citizen: this is the system used by Facebook, Yahoo, LinkedIn, and others

But:
- software has to be written specifically for Hadoop
- Hadoop-based storage is not directly from regular programs
A number of projects have started applying this technology to NGS data. For instance:

- **Seal**
  - Convert Illumina bcl to qseq
  - Demultiplex
  - Align, remove duplicates, sort

- **Crossbow**: Align (with bowtie), find SNPs

- **Myrna**: RNA expression analysis

- **SeqPig**
  - Manipulate sequencing data (filter, sort, reformat)
  - Extract statistics

- **Pydoop**: write your Hadoop programs in Python

- **CloudGene**: point-and-click Hadoop workflows
Scalability

- Hadoop base typically makes these programs more easily scalable
- i.e., needs more speed? Add more nodes
- One can avoid shared parallel file system
  - $$$
  - Can become bottleneck
  - Single point of failure (ours at CRS4 is currently offline!)
At CRS4

CRS4 Sequencing and Genotyping Platform

- Currently the largest sequencing center in Italy
- Over 2000 samples sequenced since the end of 2010

Sequencing Equipment: 3 Illumina HiSeq2000, plus older sequencers
Sequencing Capacity: about 5 Tbases/month
Processing Capacity: 3200 cores, 4 PB of storage

- We invested in Hadoop by working on Pydoop and Seal
  - Both open source projects
- Hadoop used in our production NGS workflow since 2011
Was it worth it?

- Flowcell turn-around time: reduced by 50% given a constant number of nodes
- Scalability: we can reduce turn-around time by up to 80% by increasing the number of nodes
- Drastically reduced operator time
  - Software written to be robust to hardware problems
  - We had previously experienced much down time, especially due to file system problems
More advanced tools are constantly being added to the Hadoop ecosystem
- Has reached critical mass and enterprise support
- **Impala**: distributed SQL, interactive performance
  - Query 1 TB of data, with 20 computing, nodes in as little as 6s
  - More nodes → go faster
- **Parquet file format**
  - Columnar storage returns! This time, distributed
  - Read only the columns that your query requires
  - Better compression (similar data packed together)
Shortcomings

There are important obstacles to wider adoption

- Lack of software for NGS
  - There are pieces missing to implement a typical processing workflow
- Unfamiliarity with users makes it more difficult to approach
- Not quite compatible with typical HPC clusters
  - Difficult to share institutional computing resources
"BigData" technology has enabled data-based industries that process more data than us.

Seems reasonable to consider using their technology to deal with the sequencing data deluge.
  - At least the workflow/processing part

We’re trying it at CRS4.

Overall our results are positive.

Significant hurdles to more widespread adoption
  - Part of the ICT Challenges to be solved?
“BigData” technology has enabled data-based industries that process more data than us.

Seems reasonable to consider using their technology to deal with the sequencing data deluge.
- At least the workflow/processing part

We’re trying it at CRS4.

Overall our results are positive.

Significant hurdles to more widespread adoption:
- Part of the ICT Challenges to be solved?

Thanks for your attention!