Funded by the National Science Foundation, Grant #2127548

Changing Computational Model in Science

By Alex Szalay

NSF Ecosystem Lightning Talk

March 2, 2022
CI Compass Cyberinfrastructure for NSF Major Facilities Workshop

Changing Computational Model in Science

ALEX SZALAY

JOHNS HOPKINS

Institute for Data Intensive Engineering and Science

The Emergence of Big Science

- We are moving from the era of "manual production" of scientific data in small-scale experiments to the "industrial revolution" of Big Science
- 1920s : Small experiments done by individuals, small groups, slowly growing
- 1960- : Big Science projects, costing $1B+, take decades, very risk-adverse, data have much longer lifetimes

Van der Graaf -> Cyclotron -> Synchrotron -> National Labs

SSC ☹

LHC ☺

This is a big difference

- Past: Experiments rapidly followed one another, data sets had a short lifetime
- Today: Big Science experiments (LIGO, LHC, SKA, LSST, OOI, NEON,…) may not be surpassed by another in our life

The data is here to stay (for decades)…

Today’s Science Environment

- For a long time science was bimodal, small PI projects vs Big Science
- It is changing again today – more in the **middle**
  - NSF Mid-Scale projects, NIH U01, private collaborations,
    public-private partnerships (Sloan Digital Sky Survey, PFS, Human Genome …)
  - Typically: create a unique instrument, use cutting edge technology,
    take risks, push budgets to the limit (and beyond) to maximize science,
    generate large amounts (petabytes) of data
  - Enormous fresh energy liberated!
  - At the “sweet spot” for science
  - Do computations on a shoestring
  - Good example is the Event Horizon Telescope
  - Generally: **computations will be done whereever it is the cheapest**

In this new model computations are done opportunistically

The Open Storage Network

- Science needs a robust cyberinfrastructure with 3 pillars:
  - Computing Capabilities
  - Fast Networking
  - Data-Intensive Infrastructure (still largely missing)
- There is a serious **impedance mismatch**
  - In the storage layer disk-to-disk transfers are still much slower than the network bandwidth
  - OSN idea: build an inexpensive standard appliance
    - Open software stack centrally managed, built on microservices
    - Ultra-simple, industry standard API: S3 object store
    - Plug into the 100G Internet2 backbone in the DMZ
- 8 nodes deployed, extensive testing, system functional/in use

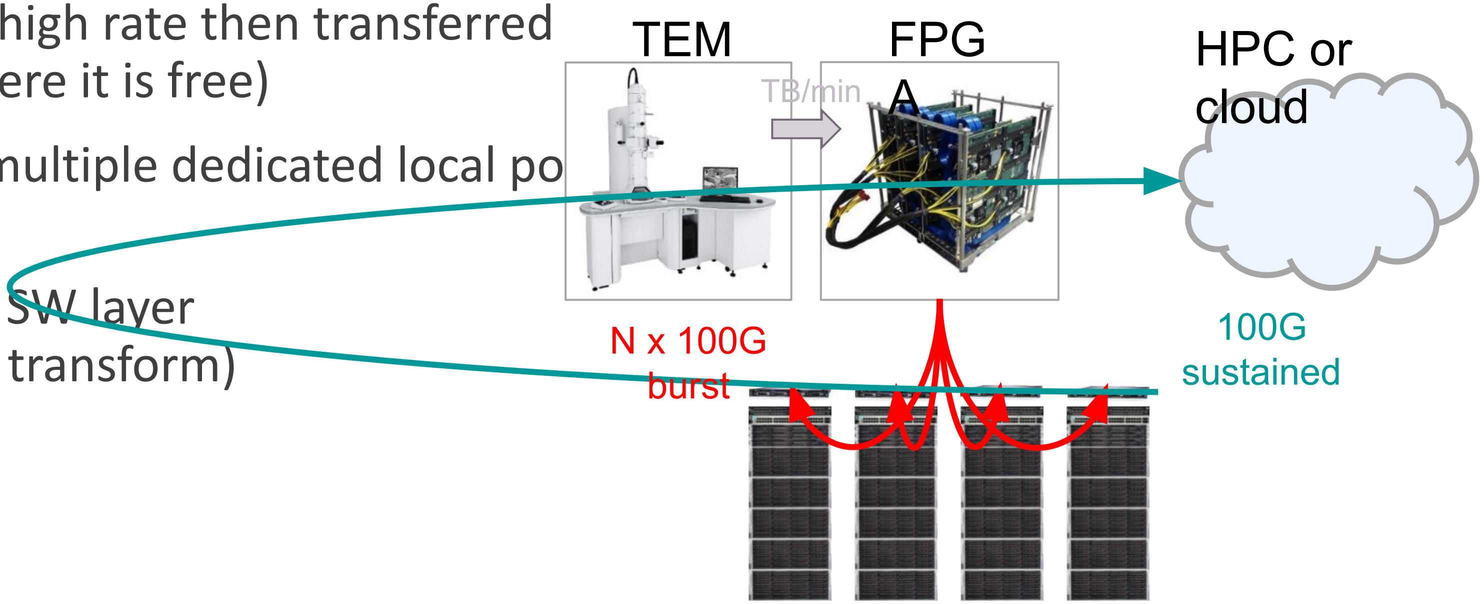
open storage network

OSN can solve the continental impedance problem

Midscale: the "First Meter" Problem

- High throughput instruments have an impedance problem
  - Both for data rate (few TB/min), and data volume (PB)
- Data needs to be first captured at a high rate then transferred to an HPC center for processing (where it is free)
- This can be beautifully solved with multiple dedicated local po
acting as high-capacity caches
- But: we need to add a customizable SW layer on the back end (compression, data transform)

  => Points to the NSDF



This task will be here even when everything will move to the cloud!

Evolving Data Analysis

- The evolution of the music industry is a good example:

LP/CD

=> iTunes

=> Spotify/Pandora

- What are the data equivalents?

Download all data

Send tapes, disk, sneakernet

=> Run queries at project servers

Astronomy archives, SkyServer, IVOA, MAST, NED,…

=> Run in the cloud, view the result

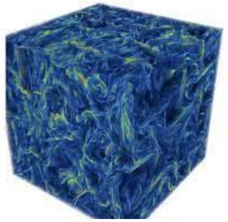
Google Colab, SciServer

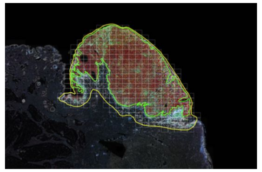
We are in between today, as keeping data in the cloud costs too much

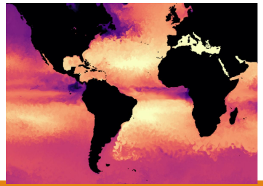
Recent Key Data Projects at JHU

Leapfrog – "non-incremental"

- (2001-) Sloan Digital Sky Survey (SDSS)
  4.2B web hits, 660M SQL queries, 10M casual users, 10K papers, 500K citations
- (2006-) JHU Turbulence database (JHTDB)
  ~0.75PB of data, 181 trillion data points delivered to the world
- (2016-) AstroPath – **1000-fold increase** in data for cancer immunotherapy,
  16 Trillion pixels, 500M cells in a spatial cell atlas
- (2017-) POSEIDON: building **the world's largest** ocean circulation model,
  2.5PB of data on its way







Building on SDSS we are able to create unique leapfrog projects over and over

Commonalities

- Data too large to store in multiple versions or multiple places (other than for redundancy)
- Need to support rapid failover
- Support multiple access patterns
  - on-the-fly visualizations
  - interactive Jupyter access
  - large scale-out computations
- Many data transformations and transfers present
  - quick look viz/Jupyter need low latency
  - data ingest and cloud scale-out must have high sequential streaming throughput

Many of these transformations can be implemented using the NSDF approach

Future-Proofing Our Computational Model

- We need to get on a future-proof trajectory (that can adopt over the years)
- We need to keep the main long-term data out of the cloud
  (for now, but be ready to overflow)
- Move enough data in **just in time**
  - AI training needs lots of cycles, but no long-term storage (may fit the cloud model)

  Eventually we will need to cross correlate between data in different clouds

  Need to move PB scale data sets – **fast!**
- There will always be many different access modes
  - We cannot afford to store the data in so many ways
  - Must be easy to perform on-the-fly conversions
- Don’t lose focus on preserving the valuable data sets