Table of Contents

pre-GDB Collaboration with Other Sciences, NIKHEF, March 10, 2015

Agenda

https://indico.cern.ch/event/319819/

Introduction - J. Templon

Co-organized with Netherlands eScience Center

Investigate possible topics for collaboration in the future

Background in Netherlands: BigGrid evolution into a new organization led by SURF and NWO, of which SARA is part (leads National e-Infrastructure) and Netherlands eScience Center is also a part, responsible for providing software engineering needed to enable new science.

eAstronomy - Rob van Nieuwpoort

Supported by eScience center: digitally enhanced science

- Funding projects with Universities and research labs
- eScience engineers to help sciences

Main challenges

- Efficient data handling
- Big data analytics
- Computing

eStep: eScience Technology Platform

- Coherent set of technologies to tackle the challenges
- Prevent fragmentation and duplication
- Easy access to data and e-infrastructures
- SW developed in projects generalized and transferred to eStep

Main "customers": LOFAR, Apertif, SKA

- LOFAR: distributed telescope made of 88K antennas around the world.
    - Hundreds of Gb/s: 200 Gb/s to supercomputer
- Next generation telescope: Apertif
    - Data rate: 2.1 Tb/s
- Both require real time processing and streaming
- SKA: (sq kilomer) array of antennas/telescopes
- Huge data: more data expected out of SKA dishes than the the current internet traffic...

Astronomy data: huge, structured, 99.999% noise

- "Correlator" in charge of doing the first level of filtering: intermediate data can be even huger
    - Must be done real-time
    - Output of correlation stored in current generation of telescopes but too big to be stored with SKA
- RFI Mitigation is the second level of "filtering"
    - Complex algorithms, computing intensive: currently done offline
    - SKA: will have to do it in real-time: only the output of this stage will be stored
    - Data too big to fit in current shared-memory systems: limited to 1 second of data, different from the current offline approach with access to stored raw data
    - New algorithm scalable (linear computational complexity), working for different scales (microsec to hours), quality as good as offline algorithm
- Complexity of algorithms to produce final products is increasing
- Also increasing complexity of data

SKA: construction first phase 2018-2023, second phase 2023-2030

- Distributed by design: Western Australia and Southern America
    - Still open: central or replicated archive?
- Image cubes to be distributed to SKA data centers
    - Rough estimate: 100 Gb/s
- Bring processing to data rather than the opposite?

This work demonstrated the efficiency of eScience with domain experts and computer scientists working together

- New HW architectures (including accelerators) change everything: optimisations, offline versus streaming, algorithms

Netherlands eScience Center trying to build an eScience network at European level

Performant Scientific SW: Theory vs. Reality - G. Raven

Former project leader of LHCb high-level trigger

LHCb challenge: trigger to reduce 40 Mhz collisions to 10k events saved

- Each collision is called an "event": each event is independent of the other
    - Thus the ability to use efficiently the "embarassingly parallel" approach
- Should be optimized both for find rare events (new physics) and fine studies of properties of frequent events (precision measures)
- Each event must be processed by a large chain of algorithm with non trivial dependency to produce reconstructed physics data from raw data
- Raw data is made of values from many sub detectors
- Decision graph for LHCb trigger: 15K nodes with a lot of repeated patterns hopefully. ~150 decisions: so O(100) nodes/decision
- Software validation very complex: tricky dependencies, potential for a lot of race conditions...
- A lot of physicists involved in writing the SW: experts in one particular area of the decision tree, various coding skills and performance awareness... difficulty of maintaining consistency. 2O people involved in writing 65% of LOC (core part) but 125 involved in 95% of LOC.

Challenge of evolving the SW to use efficiently new HW architectures

- Generation gap in people: young programmers tend to be used to powerful machines but have lost the knowledge on intrinsics and the impact on performance
- 3 dimensions to parallelism: inside a core, inside a box and LAN & WAN
    - HEP has been traditionally good mainly in the latter
    - HW threads required to fill up a core but difficult to test/benchmark sharing of cores and caches
- Analysis made on "cycle count" in HLT algorithms with 200 events: not hot spot, need to improve things in many places to get a benefit
    - 30% of cycle gained by touching tens of modules/algorithms
- Lessons learnt from this effort
    - Measure and benchmark: be sure you are improving things
    - don't do more work than strictly needed
    - Improve memory usage
    - Vectorization: use SIMD instructions. API must be vectorized too (be optimal for vectorization, hiding the details)

Vectorisation experience

- Lots of different instruction sets: how to write a code able to run optimally on all the archs
- Writing specific routine for each arch can lead to optimal perf but not sustainable, portable
    - Just work as POC for a few archs
- In progress/future: move towards higher level libraries: recompile same sources for several archs and automatize dispatching
    - Currently LHCb code doesn't speicify/mandate any arch and restrict to using SSE2, 10 years old...
- Need to reengineer the event data model (EDM)
    - Been design the wrong way: collection of arrays where it should have been an array of collections for efficient vectorization
- No possible big-bang change: possible to know where we want to go but it is trickier to find the way to go there...

LHCb upgrade challenge: with lumonisity 5x higher, the 1 Mhz readout is a bottleneck

- Upgrade readout to 40 Mhz: trigger less experiment
- Every pp interaction shipped to HLT (CPU farm): converge online/offline data processing
    - 32 Tb/s into the HLT farm
    - Requires increasing trigger efficiencies of hadronic modes by x2
    - 2-10 GB/s to storage
    - Role/types of accelerators in HLT farm not yet clear
- Will test the approach starting in Run2 (with the trigger already reducing event rate to 1 Mhz) to be able to improve/fix implementation by 2020: not expectation it will work from the first time

CERN White Paper from May 2014: "Data acquisition is where intruments meet IT systems"

- Costs and complexity must be reduced by replacing custom electronics with high-performance commodity processors and efficient SW

Dealing with HEP Data in 2020 and Beyond - M. Lammana

Main CERN responsibilities with LHC data

- Archiving (to tape), immediate reconstruction and exports to other WLCG sites (T1s): done simultaneously
- Data analysis: in conjunction with other WLCG sites
    - ♦ from disk-resident files

2 main storage technologies, both developed at CERN

- CASTOR: archiving, 15 years old
    - ♦ Currently 100 PB, 95% of files on tape
    - ♦ xrootd, gridftp and SRM
    - ♦ Krb and X509
- EOS: disk storage for analysis, 5 years old
    - ♦ Optimized for multiple concurrent read access
    - ♦ Currently ~25 PB, serving several 10s of GB/s
    - ♦ xrootd and httpd
    - ♦ 1500 disk servers... and still very resilient. Failures every day: handled transparently, intelligent replica placements to avoid replicating bottlenecks...
- Physical resources spread between Meryin and Budapest: still one global system, transparent to users
    - ♦ 2x100G links but have to deal with longer RTT (24 ms)

Technology: extrapolating from Run 1 experience

- Archiving: yearly increase 30%/year, no foreseen limit, market still interested by this kind of storage technology
- Disk: yearly 15% increase, not clear that this will last, no promising technology emerged yet

CERNBox: dropbox-like service started at CERN

- Data stays at CERN
- Based on Owncloud + EOS backend
    - ♦ Partnership with OwnCloud
- Great uptake: already 1600 users
- CERNBox visible from lxplus/lxbatch
- Sync possible with laptop/desktop

Ceph adopted as the "virtualized storage"

- Storage for services as RBD (virtual disks)
- Storage or metadata storage for services at the RADOS level (object storage): CASTOR, EOS diamond
    - ♦ CASTOR: diskservers no longer manage disk volumes: just a proxy to the storage cloud. No impact on clients.
    - ♦ Same approach prototypes for EOS
- Native S3 interface

Multiple federated data centers in the future

- Large farm operations understood/under control

- Federation brings a path for business continuity

Less and less static partitioning of data

- Ceph: dynamically allocate what you need in the global "pool of bytes"
- EOS: separated catalogs (endpoints) for each experiment but storage shared by each instance: no pool concept

-- MichelJouvin - 2015-03-11

This topic: LCG > GDBMeetingNotes20150210
Topic revision: r2 - 2015-03-12 - JeffTemplon