Caches All the Way Down: Infrastructure for Data Science

David Abramson
Director, Research Computing Centre
Professor of Computer Science
University of Queensland
david.abramson@uq.edu.au
Turtles all the way down

“a jocular expression of the infinite regress problem in cosmology posed by the "unmoved mover" paradox. The metaphor in the anecdote represents a popular notion of the theory that Earth is actually flat and is supported on the back of a World Turtle, which itself is propped up by a chain of larger and larger turtles. Questioning what the final turtle might be standing on, the anecdote humorously concludes that it is turtles all the way down”

https://en.m.wikipedia.org/wiki/Turtles_all_the_way_down
Last time I gave a version of this talk in Taiwan!
The Research Computing Centre
Core Technologies

- High Performance Computing
- Data Management
- Scientific Visualization
- Cloud Computing
- Scientific Workflows
What is Data Intensive Computing?
Data-Intensive Computing

- Very large data-sets or very large input-output requirements
- Two data-intensive application classes are important and growing

Data Mining & Data Analytics
Data-Intensive Computing

- Examples Applications:
  - Genome sequence assembly
  - Climate simulation analysis
  - Social network analysis
Data Intensive Pipelines

Capture & Pre-process

Interpret

Process

Store
Infrastructure Challenges of Big Data
Red Shift: Data keeps moving further away from the CPU with every turn of Moore’s Law

Slide courtesy Mike Norman, SDSC

data due to Dean Klein of Micron
It’s always been caches all the way down
Memory Hierarchy

1. Registers (1 cycle)
2. Cache (2-10 cycles)
3. Memory (100 cycles)
4. Remote Memory (10,000 cycles)
5. Flash Drives (100,000 cycles)
6. Spinning Disk (10,000,000 cycles)
7. Magnetic Tape

- Conventional Programming Languages
- Shared memory programming
- Message Passing
- Disk I/O
- Tape I/O
- Hierarchical File Systems
Infrastructure for Data Intensive Computing

- **Computation**
  - Large amounts of main memory
  - Parallel processors
  - Smooth out memory pyramid

- **Storage**
  - Significant long term storage
  - Smooth out the memory pyramid
  - Many views of same data
    - Parallel File System
    - Local access (POSIX)
    - Remote collaboration and sharing (Object store)
    - Sync-and-share
    - Web
    - Cloud
Data Intensive Computation Engine

- Parallel
  - High performance network
  - Good numeric performance
- Massive memory
  - Ability to hold whole data sets or data bases in memory
- High IO throughput
Parallel Supercomputers

- **Shared memory**
  - Non-uniform memory access
  - Cache coherence
  - Open MP

- **Distributed Memory**
  - Message passing
  - MPI

- **Programming methodology**
  - Domain decomposition
Massive Memory

- Put lots of memory on each node
  - What is the optimal size?
- Distributed Memory
  - Message passing?
- CC-NUMA architecture
  - Paying for cache coherence
- Distributed virtual memory
  - No free lunch - locality
FlashLite

• High throughput solid state disk
• Large amounts of main memory
• Software shared memory
• Inspired by SDSC Gordon
Why is flash SSD better than disk?

- Read latency for random IO is up to 100x faster than HDD (read head seek time)
- This speeds up database accesses enormously
What is FlashLite?

- FlashLite
  - ~70 compute nodes (~1600 cores)
    - Dual socket Intel E5-2680v3 2.5GHz (Haswell)
    - 512 GB DDR-2
    - 4.8 TB NVMe SSD
  - ScaleMP vSMP virtual shared memory
    - 4TB RAM aggregate(s)
FlashLite: Data Intensive Themes
ARC LIEF grant

- Directly manipulate large amounts of data
  - Large Memory Database Systems (Zhou, UQ)
  - Machine Learning and Classification (Zhang, Zhu, Tao and Chen, UTS)
- Integrate observational data and computation
  - Astrophysics (Drinkwater, UQ)
  - Healthy hearts (Burrage, Turner, QUT; Abramson, UQ).
  - Coastal Management (Tomlinson, Griffith)
  - Climate Change (Mackey, Griffith)
  - LiDAR processing (Olley, Griffith)
- Large main memories to operate efficiently
  - Genomics (Edwards, UWA/UQ; Coppel, Monash; Griffiths, Griffith)
- Significant temporary storage requirements.
  - Computational Chemistry (Bernhardt, UQ; Du, QUT)
Results to date
Significant Temporary Storage
Marlies Hankel, AIBN

- Gaussian 90
- Coupled cluster with single and double (substitutions from Hartree-Fock)
  - 24 cores, 30GB of ram for jobs, 200GB MaxDisk, about 143GB used
    • Walltime with SSD= 120751 s
    • Walltime with GPFS = 239289 s
    • 1.98 speedup
- Moeller-Plesset second order correlation energy correction
  - 24 cores, 250GB of ram for job, 100GB MaxDisk, about 1GB used
    • Walltime with SSD= 21191 s
    • Walltime with GPFS = 34653 s
    • 1.63 speedup
MPI with lots of memory
Christoph Rohmann, AIBN

- VASP
- Job running within one node on FlashLite used ~232GB of memory.
- So need 48 cores with 5GB per core on Tinaroo to be able to run this job.

<table>
<thead>
<tr>
<th>Cluster</th>
<th>cores</th>
<th>ram/core</th>
<th>flashdrive</th>
<th>walltime/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tinaroo</td>
<td>24</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FlashLite</td>
<td>24</td>
<td>6GB</td>
<td>no</td>
<td>Insufficient memory</td>
</tr>
<tr>
<td>FlashLite</td>
<td>24</td>
<td>10GB</td>
<td>no</td>
<td>10709</td>
</tr>
<tr>
<td>FlashLite</td>
<td>24</td>
<td>10GB</td>
<td>yes</td>
<td>8489</td>
</tr>
<tr>
<td>FlashLite</td>
<td>48</td>
<td>6GB</td>
<td>no</td>
<td>8705</td>
</tr>
<tr>
<td>Tinaroo</td>
<td>48</td>
<td>5GB</td>
<td>no</td>
<td>7799</td>
</tr>
</tbody>
</table>
The advent of the genomics era has increased exponentially the amount of data that needs to be analysed.

- Marker datasets now contain millions of markers instead of thousands.

- Cluster and order markers on a genetic linkage map.

- Efficient in memory management and “large” data sets with thousands of genetic markers.

- It uses an “all vs all” distance calculation that can be parallelised.

- OpenMP & C, vSMP

Hybrid SMP and DMM
Lutz Gross, Cihan Altinay, School of Earth Sciences, UQ

• eScript
• Solution of Partial Differential Equations (PDE) using Finite Elements (FEM)
• Timings @ 120 cores
  – MPI Only
    • Speedup: 54
  – MPI and OpenMP
    • Speedup: 52
  – OpenMP Only (vSMP)
    • Speedup of 41
Large Memory
Ondrej Hlinka, Stuart Stephen, CSIRO

• BioKanga – Genome Assembly
• Integrated toolkit of high performance bioinformatics subprocesses targeting the challenges of next generation sequencing analytics.
• Highly efficient short-read aligner which incorporates an empirically derived understanding of sequence uniqueness within a target genome
  – Hamming distances between putative alignments to the targeted genome assembly for any given read as the discriminative acceptance criteria
  – can process billions of reads against targeted genomes containing 100 million contigs and totaling up to 100Gbp of sequence.

• A large synthetic dataset (Similar CPUs):
  – Dell blade with 48 (2.1GHz) cores 3TB of RAM 32.25 hours
  – SGI UV 3K 48 (2.6GHz) cores and 3TB RAM 36.80 hours
  – FlashLite (MEX mode) – 24 (2.5 GHz) cores and 3TB RAM (6 nodes) 38.62 hours
But the caches continue …

MeDiCl
MeDiCi

- Centralising research data storage and computation
- Distributed data is further from both the instruments that generate it, some of the computers that process it, and the researchers that interpret it.
- Existing mechanisms manually move data
- MeDiCi solves this by
  - Augmenting the existing infrastructure,
  - Implementing on campus caching
  - Automatic data movement
- Current implementation based on IBM Spectrum Scale (GPFS)
FlashLite in the Data Centre

DDN SFA12KXE

FlashLite

Parallel file system
FlashLite in the Data Centre

- FlashLite
- Parallel file system
- Long term data collections

GPFS
NFS
DDN SFA12KXE
SGI DMF Disk/Tape
MeDiCI Wide Area Architecture

Machine Room Network

Parallel File System

Wide Area Network

Cache

Long term data collections

SGI DMF Disk/Tape

Automatic Recall

Second Tier
Lower cost & performance
capacity disk array

Third Tier
Tape or VTL
Active Archive
MeDiCI Wide Area Architecture

SGI DMF Disk/Tape

GPFS

NFS

Long term data collections

Wide Area Network
Identity!

- No single UID space across UQ/QCIF users
- Need to map UID space between UQ and Polaris
- GPFS 4.2
  - mmname2uid/mmuid2name
Object Storage

- S3 style objects becoming defacto standard for distributing data
- http put/get protocol
- Swift over GPFS
  - Unified Object/file interfaces
Data Data everywhere anytime

Managed Data
- ImageTrove
- myTardis
- OMERO

Unmanaged Data
- MeDiCI
- owncloud
- OpenClinica

Clinical Data
- S3, Swift
- Cloud Access

MeDiCI

QRIScloud Compute and Storage Fabric
Building on basic architecture

- A Declarative Machine Room
- Leveraging Cloud Storage
- Very Very Wide Area File Systems
- Supporting repository stacks
- Orchestrating Workflows
A Declarative Machine Room?

- Static allocation of disk and tape
- Policy driven allocation
  - RULE 'prefetch-list'
  - LIST 'toevict'

WHERE CURRENT_TIMESTAMP - ACCESS_TIME > INTERVAL '7' DAYS
  AND REGEX(misc_attributes,'[P]') /* only list AFM managed files */
MeDiCI Very Wide Area Architecture

Machine Room Network

Parallel File System

Wide Area Network

Cache

Object Store

GPFS

S3
MeDiCI Very wide area
Caches under OMERO

http: 60 seconds

GPFS: 5 seconds

3.66 GB
Caches under workflows

Capture & Pre-process

Interpret

Process

Store
Conclusions

- **FlashLite**
  - Parallel computer
  - Very large amounts of local memory and Flash disk
  - Still learning what works

- **MeDiCI**
  - Caches all the way down
  - Current PoC based on IBM GPFS (SS 4.2)
    - Three DDN appliances on campus
    - Two DDN GS12K in data centre.
    - UID mapping, object store under test
Acknowledgments

• **Australian Research Council**

• **Queensland Cyber Infrastructure Foundation (QCIF)**

• **CSIRO**
  – Ondrej Hlinka, Stuart Stephen

• **University of Queensland**
  – Jake Carroll, Michael Mallon, Kevin Smith, Marlies Hankel, Lutz Gross, Cihan Altinay, Christoph Rohmann

• **SDSC**
  – Mike Norman

• **AARnet**
  – Peter Elford