Optimizing Spark with Cray

STAC December 2015

Philip Filleul – FS Global Lead
pfilleul@cray.com

Cray: The Myth vs. Reality

Myths

- They are huge
- They are proprietary
- They are complex
- They are expensive

Vs. Reality!

- They can be – but they start less than a rack
- No: Intel, Linux, open standards, Hadoop
- Simpler and more productive than a grid
- No – cost competitive, lower TCO, higher value

Cray in FS: A Refresher

CRAY

Compute				Store	Analyze	
CVA/XVA + Strategy Back-Testing				IO bound workloads e.g. Strategy Back-Testing	Compliance, Surveillance, Risk, Cyber Security, Robo Advice	
Aries Interconnect	Density and Power	Intel Phi	Dense GPU	7.5TB to 1.7TB per sec Posix based PFS	Super Scale Spark	Super Scale Graph
Enables single memory space and low latency data sharing	Best density and power efficiency as shown in STAC A2	KNC and earliest access to KNL	8 GPU node that does not throttle back and is reliable	Appliance with extreme performance and reliability	Low latency fast shuffle	Production scale unpartitioned data

Workload Suitability of Spark in FS

- ✓ **Data parallel, Memory first, Fault tolerant, Productive**
- ✓ **Suitable to refactor e.g. risk workloads**
  - ✓ Not just nightly batch but enables model interaction intraday
  - ✓ Rapid prototyping
- ✓ **Full instrument trial/loss matrix can be Tb – easy for Spark to handle**
- ✓ **Productive – easy to describe parallel computations in a few lines of code with Scala**
- ✓ **Efficient – Optimized LA libraries can be accessed via JNI**

Spark Academic Benchmark on Cray

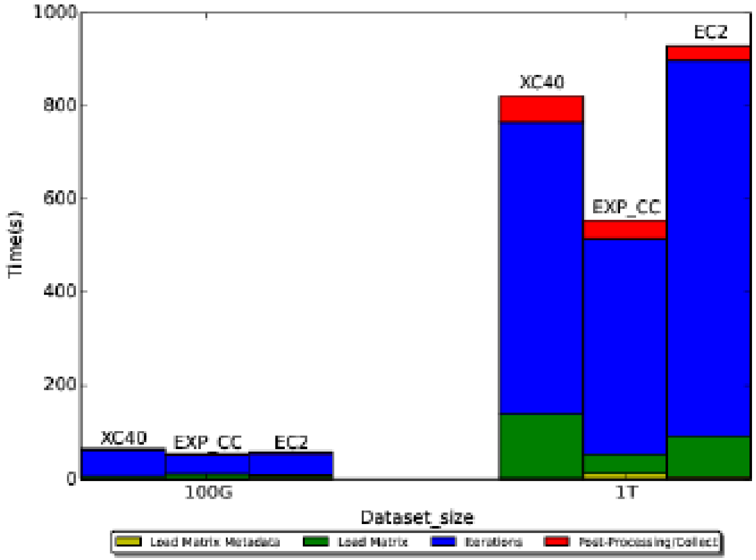
- **Scientific HPC Workload * (*not a STAC benchmark*)**
  - Mass spectrometry
  - Multi-rank matrix factorization
  - 1Tb dataset
  - Written in Scala with vectorized LA library

- **Targets**

Platform	Total Cores	Core Frequency	Interconnect	DRAM	SSDs
Amazon EC2 r3.8xlarge	960 (32 per-node)	2.5 GHz	10 Gigabit Ethernet	244 GiB	2 x 320 GB
Cray XC40	960 (32 per-node)	2.3 GHz	Cray Aries [1, 5]	252 GiB	None
Experimental Cray cluster	960 (24 per-node)	2.5 GHz	Cray Aries [1, 5]	126 GiB	1 x 800 GB

(* NERSC 2015)

Results



Platform	Rank	Total Runtime	Load Time	Time Per Iteration	Average Local Task	Average Aggregation Task	Average Network Wait
Amazon EC2 r3.8xlarge	16	24.0 min	1.53 min	2.69 min	4.4 sec	27.1 sec	21.7 sec
Cray XC40	16	23.1 min	2.32 min	2.09 min	3.5 sec	6.8 sec	1.1 sec
Experimental Cray cluster	16	15.2 min	0.88 min	1.54 min	2.8 sec	9.9 sec	2.7 sec

Further Spark Optimizations

- **Spark is not yet sophisticated in many areas e.g.**
  - Scratch storage does not account for a memory and storage hierarchy
    - Just round robin   vs….
    - Fill top tier (release when possible) then spill to next
  - Aggregations send data between nodes
    - Does not leverage shared storage
      - Node A->Node A storage ->Node B storage ->Node B   vs…
      - Node A ->shared storage->Node B

- **Cray and UC Berkeley AMPlab joint work**

COMPUTE | STORE | ANALYZE

Thank you!

pfilleul@cray.com