TEXAS ADVANCED COMPUTING CENTER
WWW.TACC.UTEXAS.EDU

TEXAS
The University of Texas at Austin

Pragmatic Performance: A Survey of Optimization Support at the Texas Advanced Computing Center

W. Cyrus Proctor

ISC Performance Engineering for HPC Workshop
tinyurl.com/tacc-pe-hpc

June 22, 2017

Pragmatic Performance: Where to Begin?

Snapshot:

- Billions of compute hours served
- Ten million job submissions
- Ten thousand active users
- Thousands of active research projects
- Hundreds of code bases run in production
- Tens of production resources

Pragmatic Performance: Where to Begin?

Snapshot:

- Billions of compute hours served
- Ten million job submissions
- Ten thousand active users
- Thousands of active research projects
- Hundreds of code bases run in production
- Tens of production resources

Challenge:

- How to address performance at this scale?

Pragmatic Performance: Where to Begin?

Broad spectrum of researchers:

- Novice users
- Domain scientists
- Industrial partners
- Performance specialists

Pragmatic Performance: Where to Begin?

Broad spectrum of researchers:

- Novice users
- Domain scientists
- Industrial partners
- Performance specialists

Challenge:

- How to empower our user base given their diverse needs and skills?

Pragmatic Performance: Where to Begin?

Diverse selection of hardware:

System	Highlights
Stampede2	KNL, SKL, Intel Omnipath
Stampede1	SNB/KNC, IB FDR, K20s
Lonestar5	HSW, Cray Aries, K40s
Maverick	IVB, IB FDR, K40s
Wrangler	HSW, IB FDR, Flash
Hikari	HSW, IB EDR, Encryption
Jetstream	HSW, GigE, Cloud
Chameleon	ARM, HSW, K80s, P100s, AMD, FPGAs

Pragmatic Performance: Where to Begin?

Diverse selection of hardware:

System	Highlights
Stampede2	KNL, SKL, Intel Omnipath
Stampede1	SNB/KNC, IB FDR, K20s
Lonestar5	HSW, Cray Aries, K40s
Maverick	IVB, IB FDR, K40s
Wrangler	HSW, IB FDR, Flash
Hikari	HSW, IB EDR, Encryption
Jetstream	HSW, GigE, Cloud
Chameleon	ARM, HSW, K80s, P100s, AMD, FPGAs

Challenge:

- How to characterize performance with this range of hardware?

Pragmatic Performance: Where to Begin?

- As part of a national cyberinfrastructure (NSF XSEDE), many resources are oversubscribed by up to 3x
- Performance and efficiency are key to minimize wasted cycles
- Identifying inefficient patterns can improve system utilization
- Reasons for inefficiencies are varied and require many techniques to mitigate
- Insufficient number of staff to manually address efficiency issues across the board

Pragmatic Performance: Where to Begin?

- To be effective, systems need to be characterized
- Knowing what is running is the first step to improving performance
- The continuous monitoring tool XALT by McLay and Fahey helps with that

XALT

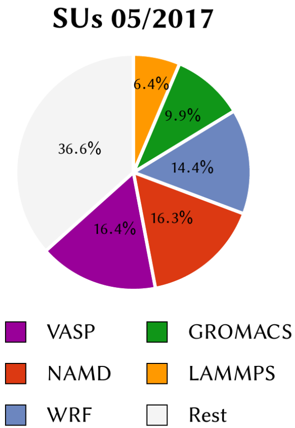
https://github.com/Fahey-McLay/xalt
http://doi.org/10.1109/HUST.2014.6

- Provides detailed job-level metadata
- XALT automatically tracks every
  - time a module is loaded/unloaded
  - executable run in a job and maps back to module
  - library linked to at compile time
  - library loaded at runtime
  - environmental variable set at runtime
- Generates weekly/monthly/yearly usage reports

XALT

- Lightweight
- Intercepts information at runtime via
  - linker (ld) wrapper
  - code launcher (e.g. mpirun)
  - ELF binary format hooks
- Collected into DB for analysis
- Powerful filtering mechanisms

XALT



- XALT provides information on what is being run
- Encompasses staff-provided and user generated codes
- This information helps drive
  - what software is staff-provided on the system
  - future system procurement decisions
  - training and outreach directives

XALT

Aside:

- Early 2015 Stampede1 demand increased
- Queue wait times: from 2-3 hours to 24 hours
- Mitigation efforts ensued
- Largemem queue wait time remained high
- With XALT, one application took 50% of cycles
- Moved to normal queue with fewer tasks per node

Staff-provided Software

https://github.com/TACC/hpc_spec
https://github.com/TACC/lifesci_spec

- Software is built and deployed via RPMs
- System-specific templates are generated and adapted for each application
- Staff software maintainers are generally advanced users if not developers
- Built for specific architecture/network/accelerators
- Dependency stacks are built for specific compiler and MPI library combinations
- Aggressive optimization/vectorization flags used with result verification
- Math libraries (generally MKL) are injected where possible

Staff-provided Software

Example highlights:

- R and Python with native MKL support
- Tensorflow optimized for GPUs and CPUs on CentOS 6 & 7

Staff-provided Software

Example highlights:

- R and Python with native MKL support
- Tensorflow optimized for GPUs and CPUs on CentOS 6 & 7

Aside:

OpenHPC

Design philosophy for OpenHPC grew out of TACC's software delivery model (Schulz)
https://openhpc.community

Pragmatic Performance: What's Next?

- Custom-built software driven by XALT data provides a transparent, performance-enhanced layer that the users don't have to worry about
- Majority of users are interested in getting science done, not performance
- Staff are charged with providing an intuitive, hardware-aware, software ecosystem
- The next step is to understand how well users' jobs are performing
- The continuous monitoring tool TACC Stats by Evans, Barth, and Hammond helps with this

TACC Stats

https://github.com/TACC/tacc_stats
https://doi.org/10.5281/zenodo.595073

- Collects job-level resource usage and performance data
- Interface with XALT and system aggregate logs via Splunk
- Curates and analyzes data

TACC Stats

- Runs on every node for every job (triggered by `inotify`)
- Collects hardware counters from Intel Processors (NHM/WTM/SNB/IVB/HSW/KNL)
- Collects Linux OS stats
- Network stats (Lustre/IB/GigE/Omnipath)
- 0.005% load on single core at 10min sampling (3% load at 1s)
- Computes job-level metrics and flags jobs for
  - inefficiencies
  - failures
  - "poor" performance
- Provides data via a web portal and SQL database

TACC Stats

For every job compute:

- Network (mean and max)
  - LFS IOPS
  - LFS Op wait times
  - LFS BW
  - IB BW
  - GigE BW
- OS
  - Memory HWM
  - CPU Usage
  - CPU Imbalance in time
  - CPU Imbalance in nodes
- Processor/Socket (mean)
  - Flops
  - Cycles per Instruction
  - Cycles per L1D replacement
  - Loads hits to L1, L2, LLC
  - All Load Operations
  - Memory Bandwidth
  - FP Vectorization %

TACC

TACC Stats: Job Dashboard

Job ID	UID	user	project	executable	start time	end time	run time (s)	requested time (s)
6311503	818567	rtevans	A-ccsc	wrf.exe	Jan. 7, 2016, 3:41 p.m.	Jan. 7, 2016, 4:11 p.m.	1805	180

File System	MB Read	MB Written
scratch-clilov	2.9e+03	1.6e+04
gsfs-clilov	0.0e+00	0.0e+00
home-st-clilov	8.9e-01	1.5e-02

Executable Path	/scratch/02561/rtevans/m10/wrf.exe
Working Directory	/scratch/02561/rtevans/m10
Links to Splunk Logs	Client Logs Server Logs

Processes Alive During Job

mpispawn sshd mpirun_rsh slurm_script ibrun sleep ssh wrf.exe bash

Click for For Tests Summary

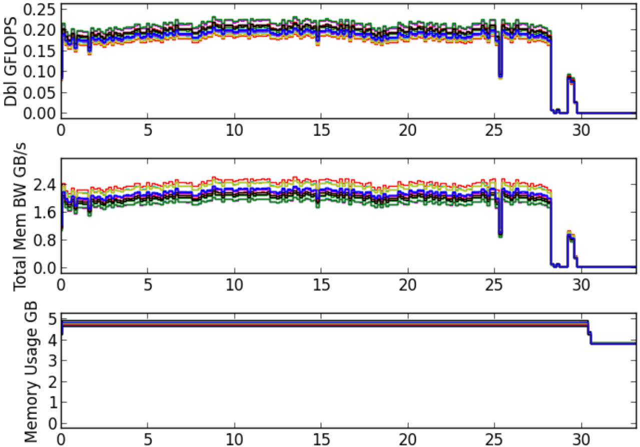
Test	Measured	Threshold	Result
MetaDataRate	6.3e+05	1.0e+04	Failed
Catastrophe	2.3e+00	1.0e-02	Passed
MDCWait	3.4e+03	1.0e+04	Passed

Click for Modules and Libraries

Module	Library
system	/lib64/ld-
system	/lib64/lib
system	/lib64/lib
system	/lib64/lib
system	/lib64/lib

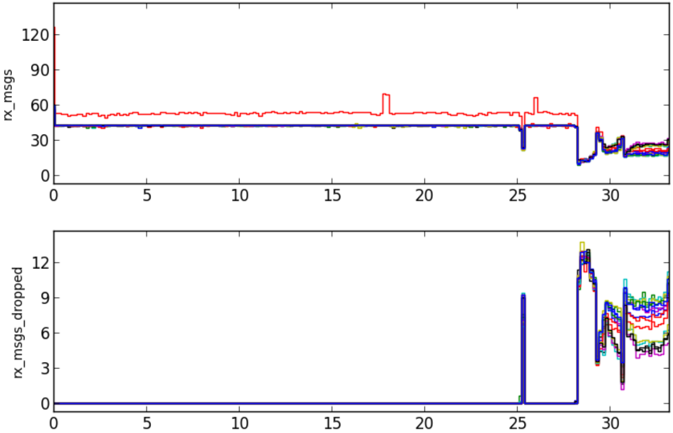
TACC Stats: Diagnosing Failures: Time Imbalance

OSS Hung



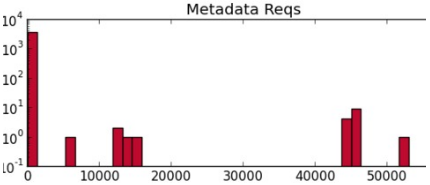
TACC Stats: Diagnosing Failures: Time Imbalance

OSS Hung



TACC Stats: Diagnosing Performance Issues

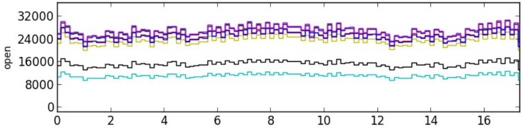
Hist WRF LFS IOPS Dec 2015



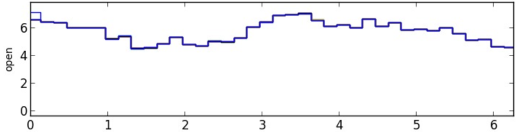
User Modified WRF App

- Before
  - Open/Close in inner loop!
  - CPU Usage = 63%
  - IOPS = 50k
- After
  - CPU Usage = 100%
  - IOPS = 12
  - runtime 1040m $\rightarrow$ 380m

Before



After



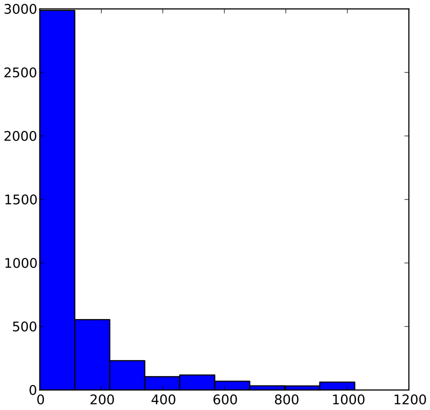
TACC Stats: Inform Hardware Procurement

Save money on underutilized hardware!

Largemem Nodes

- Stampede1 has 16 1TB nodes
- Average Mem/node 134 ± 184GB
- 94% Jobs Mem $<$ 512 GB
- New system procurement: Lonestar5 has 2 1TB nodes and 8 512GB nodes

Ave Memory: Largemem Jobs 2014



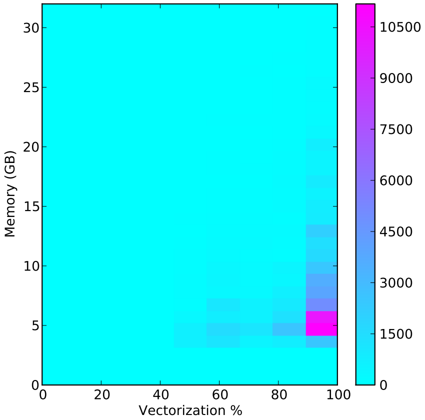
TACC Stats: Inform Application Migration

Stampede was upgraded with ~ 500 Intel KNLs.
What applications are likely to perform well immediately?

KNL Candidates from Jan-Apr 2016

- Fit in MCDRAM (16GB)
- Have high vectorization
- 40% jobs:
  - use $<$ than 16GB
  - have $>$ 50% vectorization
- Identify candidate users/apps: PlasComCM, Athena, Gromacs, Quantum Espresso, sextet.x (LQCD), GENE

Mem Use vs. Vec %



TACC Stats

- TACC Stats provides low-overhead snapshots in 10 minute intervals via a web interface that are invaluable to staff for historical/diagnostic purposes
- Staff address more than 8000 user inquiries (tickets) a year
- An appreciable amount benefit from the information provided by TACC Stats
- Soon, each user will be able reference their own historical data as well

Pragmatic Performance: Ad-hoc Monitoring

- The practices thus far have been implicit and part of a continuous monitoring scheme that requires no action from users
- When the need arises, either through explicit inquiry or from the suggestions of staff, a more detailed ad-hoc monitoring approach can be applied
- REsource MOnitoring for Remote Applications (REMORA) developed by Gómez and Rosales is designed with this mind

REMORA

https://github.com/TACC/remora
https://doi.org/10.1145/2834996.2834999

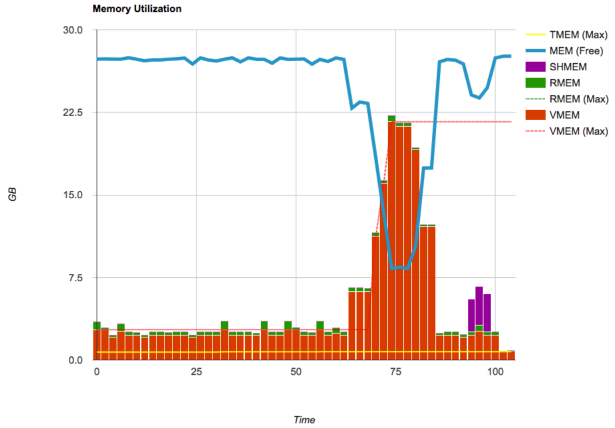
- Runs in user space
- Monitors all user activity for a given job
- Per-node and per-job resource utilization data
- Fine-grained temporal resolution (tunable)
- Simplified output for basic user
  - Highlights possible issues without overwhelming
- Raw data available for advance users
  - Deep analysis of each run possible
  - Post-processing tools provided

REMORA

Capabilities include:

- Detailed timing of the application
- CPU utilization
- Memory utilization
- NUMA information
- I/O information (FS load and Lustre/DVS traffic)
- Network information (topology, IB and Ethernet traffic)
- MPI Statistics
- Power and Thermal CPU info
- Accelerator support
  - KNC
  - NVIDIA GPUs

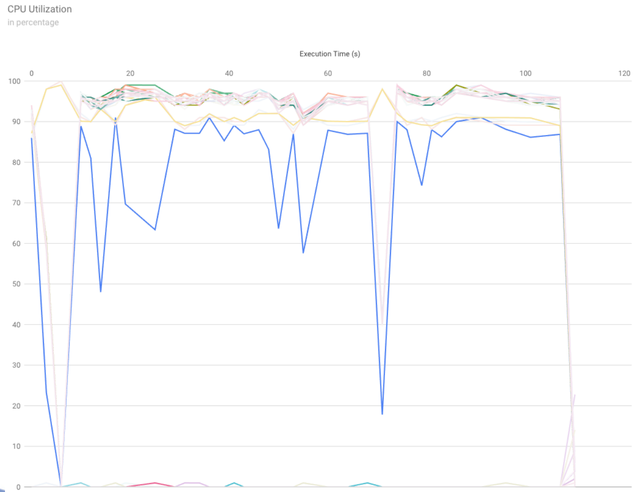
REMORA: Memory Utilization



For each node, at each time step:

- Free memory
- Aggregated
  - resident
  - virtual
  - shmem

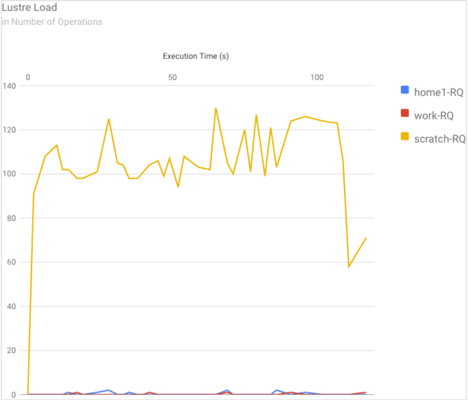
REMORA: CPU Utilization

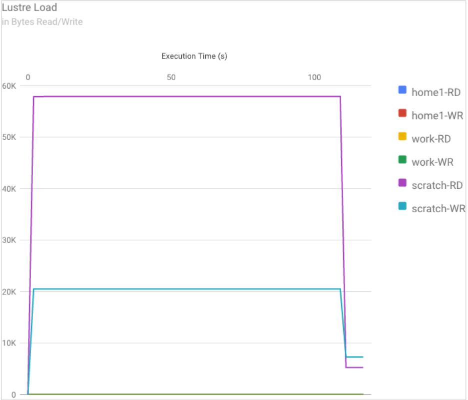


Interactive chart that shows CPU utilization at each time step

- 272 cores currently shown in the plot

REMORA: Lustre Support





- Automatic discovery of Lustre FS
- Data read/written and number of IOPS during each time step

REMORA

- REMORA serves as a powerful, concise, cluster and user friendly profiling tool
- Usage now eclipses other profiling/debugging tools in its 1.5 year production history
- Between TACC Stats and REMORA, the far majority of reasons for job failure or obvious poor performance can be quickly identified and mitigated
- Not only can staff provide insight but users can easily apply to other workflows

Pragmatic Performance: Education and Outreach

- One of the best ways of reaching researchers who are ready to take their codes and skills to the next level is through training
- TACC staff provide training opportunities to many different communities
- Most content can be found online via webpages or slides & webcasts for asynchronous consumption

Education and Outreach

https://portal.tacc.utexas.edu/training
https://www.youtube.com/channel/UCIyVQ1bICGCggZisXBSSRlw
https://www.tacc.utexas.edu/education/academic-courses

- University of Texas at Austin and Texas system training
- XSEDE Communities and webcasts
- University courses
- Training institutes
- STEM and industrial partner outreach
- Conference participation

Education and Outreach

https://portal.tacc.utexas.edu/training
https://www.youtube.com/channel/UCIyVQ1bICGCggZisXBSSR1w
https://www.tacc.utexas.edu/education/academic-courses

- University of Texas at Austin and Texas system training
- XSEDE Communities and webcasts
- University courses
- Training institutes
- STEM and industrial partner outreach
- Conference participation

Check out content from ISC17's tutorials on manycore programming and optimization

Pragmatic Performance: In-house Expertise

- TACC employs many domain and computer scientists who directly serve as consultants
- Performance-related inquiries are load balanced and answered by PhD researchers
- New research project proposals are reviewed by staff for consideration
- A key component of the review process is the XSEDE Extended Collaborative Support Services (ECSS)
- These services may be requested by either the requestors or the reviewers

XSEDE ECSS

https://www.xsede.org/ecss

Expertise is available over a wide range of areas:

- performance analysis
- petascale optimization
- efficient use of accelerators
- I/O optimization
- data analytics
- visualization
- use of XSEDE by science gateways
- workflows

XSEDE ECSS: Success Stories

National Flood Interoperability Experiment

- Collaboration between the National Weather Service, academia, and commercial partners designed to create
  - national flood forecasting system
  - real-time flood information services
- Routing Application for Parallel computation of Discharge (RAPID)
  - Key component of forecast system
  - Simulate river flow from inflow parameters
  - 10+ hours when analyzing large river basin
  - Too long to provide a real-time forecast
- Collaborative support project to improve RAPID runtime (Liu)
  - Implemented a hash table
  - Designed new data structures
  - Developed new algorithm

Runtime dropped from 1000s to less than 1 second!

XSEDE ECSS: Success Stories

Laser Interferometer Gravitational-Wave Observatory (LIGO)
Portability & Performance Tuning:

- LIGO team needed extra compute capacity for in-depth analysis of signals
- NSF suggested working with XSEDE program to determine if supplemental computing was essential
- CONDOR was in use for high throughput computing
- ECSS project created to port data analysis pipeline and optimize efficiency
- El Khamra helped transform workflow into portable VMs
- McCalpin improved the FFT performance component by a factor of 4

Conclusions

- To help improve performance, a large and complex HPC environment requires
  - well-built software
  - powerful tools
  - talented staff
  - informative training
- Let us know if these are of interest to you, or if there is a critical element we are not considering yet!

Thank you for your time and attention

cproctor@tacc.utexas.edu

License