EE-HPC-WG Workshop, SC13
Data Motion projects at ESSC

Presented by:

Stephen Poole
Chief Scientist – CSM
Director of Special Programs/ESSC

Currently deployed in Research at the DoD.
The ESSC Team

• ORNL
  – Stephen Poole
  – Joshua Lothian
  – Chung-Hsing Hsu
  – Jonathan Schrock
  – Brad Settlemyer
  – Greg Koenig
  – Pavel Shamis / Pasha
  – Manjunath Venkata / Manju
  – Oscar Hernandez
  – Matthew Baker
  – Sarah Powers
  – Nina Imam
  – Tiffany Mintz
  – OLCF/NCCS (Jim Rogers, Don Maxwell)

• Partners
  – UofH (many interns, Sidhartha)
  – DoD (SME’s) (CC)
  – Sonoma State University
  – Link Analytics
  – Colorado State Univ.
  – LANL / SNL / ANL
  – UTK
  – NMI – (Steve Hodson)
  – NCSU (Blair Sullivan)
  – QRI, Inc. (Jeff Kuehn)
  – SDSC
  – Natalie Bates

(U - Stephen Poole - ORNL)
Outline

• Where did we start and why? (2006)
• What did we want and why? (Goals/Ideas/Vision)
• What tools do we have now? (Benchmarks, Libraries…)
• New Benchmark
• How are we enabling these new tools/capabilities?
• How do we make data available and where?
A System of Systems focused on Data Analytics
(if data moves on/in it, instrument it - 2007)

Remote Partners (LANL/ANL...)

Dissemination

Collaboration

WAN Links

ORNl Campus

Advanced Interconnect

Cooperative Analysis / Viz

MPPs

Clusters Farms

Storage/DBS

Hybrid Computing

(U - Stephen Poole - ORNL)
Hierarchical Measurement Domains

ORNL has lots of power, *BUT* money=people/projects.

How to help the procurement and facilities folks get a better handle on real costs

UPS = Uninterruptible Power Supply
PDU = Power Distribution Unit
System Analytics
(overall system approach - 2007)

- External sensors
  - Networking (LAN/WAN)
  - Environment
- Internal sensors
- Collection SW
- Networking (LAN/WAN)
- Storage
- Benchmarks
- Math Tools
  - Statistical tools
  - Graph theory tools

What will fit on a chip?
What useful information can we extract?

Collect, Trace, Model, System Replay, Simulate…
Customizing Scheduling Algorithm via ESSC-DB

Design of a system to integrate several systems together with a customized scheduling algorithm inside Moab using SLURM as the resource manager, and the ESSC database as a repository of disparate sensor information for forensic analytics and all jobs. (Job, Machine type, Cost, Location, constraints...)
Some of what we have used

• HPL
  – High-Performance Linpack (HPL) is the de-facto standard for FP-Dense (~Z)
  – Great historical data base

• Graph500 / Green Graph500
  – Data Intensive HPC Benchmark

• SPEC
  – SSJ2008 (Java)

• XDD / IOR (Instrumented File I/O, LAN/WAN)

• GUPS/Guppie, Random Access (Other DoD kernels)

• DOE-SC Apps, DOE-NNSA public Kernels
Some of what we currently use (cont)

- **SystemBurn**
  - SystemBurn allows us to emulate different application behavior profiles within a single framework (LOADS/Hybrid LOADS)
  - Used in DOE and DoD procurement process and machine diagnostics
  - Integrated performance data derived from PAPI or est. op counts
  - Development of infrastructure for automatic maximization of power draw
  - Tight integration of SystemBurn with ESSC DB
  - Some existing loads (others can be written, roll your own)
    - Memory loads: LSTREAM, DSTREAM, DSTRIDE, LSTRIDE, GUPS
    - I/O Loads: WRITE, Scenarios (1-12), Networking (LAN/WAN)
    - “Power Virus”: PV1, PV2, PV3 streaming computation
    - Mixed Loads: CBA, ISORT, TILT
    - CUDA/OpenCL/OpenACC Loads: DGEMM, BLAS
    - SLEEP – a dummy do nothing load
    - PCI Bus Load
Proposed Capabilities of HIPATIA

- New Benchmark
  - HiGH Performance AdapTive Integrated Linear AlGebra Benchmark
    - HIPATIA (hy-pay-shə)
  - Scalable
  - Integer focused but will also evolve to use fixed point and others.
    - Not a lot of attention has been paid to non-FP problems (HW/SW)
  - User Configurable (with fixed/required runs, ala. HPL, Graph500)
    - Graphs (input)
    - Defined Matrix Types
      - Sparse, Dense, Structured, User defined (rules)
  - Fully Instrumented for Power and Performance (Data Motion costs)
  - Toolkits / Libraries available for HPL and others
    - Graph Generator(s)
    - Selected matrices
Proposed Capabilities (cont)

• Multiple implementations
  – C, OpenSHMEM, UPC, MPI, (Fortran, Cuda, OpenCL, ??)

• Additional areas
  – Will be incorporated into SystemBurn as a load module
    • So you can select between R, C Z, Fixed Point…
  – Will incorporate UCCS
    • With Power/Flow/Comms… tracing and cost models
  – Will incorporate signatures into DB (ESSC-DB)
  – Will be used by DoD/DOE (Applicable to: Oil, Informatics…)

• Hypatia
  – babelniche.wordpress.com for the image
UCCS
(Universal Common Communication Substrate)

(Runtime System)

Application Software

OpenSHMEM
SHMEM Tracer (SDSC)
TAU (Oregon)
(Open-SHMEM-Check)
(UPC-CHECK)
PPW

Other PGAS
ActiveMsg
MapReduce
I/O
MPI

UCCS (Instrumented)

Driver & Hardware Enhancements

EtherNet
Torrent PAMI
InfiniBand
Shared Memory
uGNI
Other Proprietary Networks

(U - Stephen Poole - ORNL)
# Graph Generators Progress

| Initial | • Identify limits of current generators (internal report)  
|         |   • Classical/Theoretical/Random, Internet, Real World Network, Geometric  
|         |   • Generate synthetic data set (we need useful sized ones)  
|         |       • Of great computational and learning value |
| Current | • Complete down selection process (implementations varied)  
|         |   • Implement final set of scalable generators in OpenSHMEM, UPC, MPI  
|         |   • Algorithm-optimized data structures for best performance |
| Next    | • Implement pluggable generator(s) for HIPATIA integration |
Power Studies for OpenSHMEM

• Assume you have a network with reasonable latency/BW (not MPI centric)
• Power is very sensitive on how we effectively use caches.
  – Small/medium message sizes tend to be more cache friendly.
  – Small fragmentation of messages is good for power
• Memory accesses are expensive
  – More for medium and large message
• Barriers (HW/SW) are expensive in terms of power
  – They raise the power states of CPUs if they spin
  – Alternative implementations are needed
• Polling for messages is expensive
  – RDMA hardware for PGAS may improve this.
• We need to explore event-based execution models to save power
OpenSHMEM Power Studies
Power v/s Cache Misses for \texttt{shmem\_putmem}()
(Mellanox SHMEM)

(I) Cores Power (Watts)

(II) DRAM Power (Watts)

(III) L3 (shared) Cache Misses

(IV) Normalized (Bandwidth / Watt) per message size
Aspect 4 (EE-HPC) Power Measurement Point:
Integrating measurements at A,B,C PLUS lower-rate measurements at D,E or F (to measure power supply losses) satisfy L1-L3 (entire machine)
We are working with NCCS/OLCF on D/E/F already. Some info is “difficult” and sensitive. We collect for power consumed, not peak. Already released some information.
What do we do with all of the data

- Repository for (sanitized) released data (LANL-Institutes?)
  - The IBM P7-IH (PERCS-DCIR) generates enormous data.

- Collecting a variety of system data is very important (Potential Predictors)
  - Application Signatures, Performance, Runtimes Traces
  - Power / Energy / Water (Cooling)
  - Resource Manager, Job Scheduler Information
  - Network (local / remote, HCA, Switch(s), Optics, Integrated NIC)
  - I/O (FileSystems)

- Helps guide system purchases and funding requirements
- Great feedback to the vendors and apps developers, compiler developers
- Helps determine power budgets
- A variety of machines/technologies (MPP, Clusters, …) All Vendors
Talks at SC13 on these topics

- https://github.com/jlothian/systemburn
- http://openshmem.org -> Announcements -> SC13 Schedule
- Power talks (Chung-Hsing)
- I/O talk Brad S.
Acknowledgements

This work was supported by the United States Department of Defense & used resources of the Extreme Scale Systems Center at Oak Ridge National Laboratory.

spoole@ornl.gov
swpoole@gmail.com