

HEC FSIO Session 6: Measurement and Understanding Talks & Roadmap

James Nunez, Los Alamos National Lab

August 2009

Problem Definition

- Research tools for measurement and understanding of parallel file system and end-to-end I/O performance are needed for advances in future file systems.
- There is a need for research into evolutionary ideas such as layered performance measurement, benchmarking, tracing, and visualization of I/O related performance data.
- More radical ideas to be explored include end to end modeling and simulation of I/O stacks and the use of virtual machines for large scale I/O simulation.

Current R&D Gaps

- Understanding system workload in HEC environments
- Standards and common practices for HEC I/O benchmarks and trace formats
- Simulation and Modeling
- Test beds for I/O Research
- Applying cutting edge analysis tools to large scale I/O

2006 HECURA/CPA Projects

- Performance Models and Systems Optimizations for Disk-Bound Applications; Mithuna S. Thottethodi, Purdue University
 - a unified and flexible disk-array access model that improves accuracy by accounting for (a) the contention on the interconnection network between disks and memory and (b) internal disk behavior
- File System Tracing, Replaying, Profiling, and Analysis on HEC Systems; Erez Zadok, SUNY at Stony Brook
 - exploring many aspects of HEC file systems such as tracing, replaying, profiling and analyzing
- Algorithms Design and Systems Implementation to Improve Buffer Management for Fast I/O Data Access; Xiaodong Zhang, Ohio State University Research Foundation and Song Jiang, Wayne State University
- Formal Failure Analysis for Storage Systems; Arpaci-Dusseau, Remzi H; Univ of Wisconsin-Madison
 - Wisconsin's Program Analysis of Storage Systems (PASS) program brings a more formal approach to the problem, utilizing programming language tools to build, analyze, test, and monitor these storage systems
- Toward Automated Problem Analysis of Large Scale Storage Systems; Narasimhan, Priya; Carnegie-Mellon University
 - Automating problem analysis
- SciDAC PDSI and SDM

2009 HECURA Projects and Presentations

- Automatic Extraction of Parallel I/O Benchmarks from HEC Applications; Xiaosong Ma, North Carolina State
- Visual Characterization of I/O System Behavior for High-End Computing; Kamil Iskra, University of Chicago
- RUI: Automatic Identification of I/O Bottleneck and Run-time Optimization for Cluster Virtualization; Xubin He, Tennessee Technological University
(**Spoke Monday**)

2008 Measurement and Understanding Gap Area

Area	Researcher	FY 07	FY 08	FY 09	FY 10	FY 11	FY 12	Rankings
Understand system workload in HEC environment	Amaci-Dusseau	■	■	■				 A comprehensive tool is nowhere in sight; problem is complex.
	Narasimhan	■	■	■				
	Reddy	■	■	■				
	Smirni	■	■	■	■			
	Zadok	■	■	■	■			
	SciDAC - PDSI	■	■	■	■			
	SciDAC - SDM	■	■	■	■			
Standards and common practices for HEC I/O benchmarks and trace formats	Zadok/Miller		■	■	■			 Danger of over simplifying problem and could drive vendors to incorrect solutions.
Testbeds for I/O Research	Ligon	■	■	■				 Simulators are being developed. No real testbeds being built. This problem will only get worse over time, i.e. as systems get bigger.
	Thottethodi	■	■	■				
Applying cutting edge analysis tools to large scale I/O	Reddy	■	■	■				 Data are becoming available from Labs including I/O traces. Many opportunities to evaluate this research.
	Zadok	■	■	■				
	LANL/CMU - Trace replay and Visualizer		■	■	■			

- | | | |
|--|---|---|
|  Very Important |  Greatly Needs Research |  Greatly Needs Commercialization |
|  Medium Importance |  Needs Research |  Ready and Needs Commercialization |
|  Low Importance |  Does Not Need Research |  Not Ready for Commercialization |
|  Full Calendar Year Funding |  Partial Calendar Year Funding |  On-Going Work |

Current R&D Gaps

- Understanding system workload in HEC environments
- Standards and common practices for HEC I/O benchmarks and trace formats
- Simulation and Modeling
- Test beds for I/O Research
- Applying cutting edge analysis tools to large scale I/O

Updated R&D Gaps

- Understanding system workload in HEC environments
- Standards and common practices for HEC I/O benchmarks
 - Removed “and trace formats” one reason is SNIA’s failure to standardize trace formats (IOTT)
- Simulation and Modeling
 - New gap added
- Test beds for I/O Research
- Applying cutting edge analysis tools to large scale I/O

Ranked R&D Gaps

- Understanding system workload in HEC environments
 - Score: 59
- Standards and common practices for HEC I/O benchmarks
 - Score: 23
- Simulation and Modeling
 - Score: 34
- Test beds for I/O Research
 - Score: 31
- Applying cutting edge analysis tools to large scale I/O
 - Score: 54

Discussion and Comments

- Standard Trace Formats
 - SNIA failed to standardize trace formats; this is probably hopeless or not worth it since what you want to collect depends on what you want to do with the data.
 - Maybe start with requirements on standard things to collect; complete/exhaustive list?
- Standards and common practices for HEC I/O benchmarks
 - “Locality incorporating benchmarks” - measure the memory hierarchy do not exist (SPECFS is the only useful one, but outdated)
- Simulation and Modeling
 - Work is being done already – UCSC and Clemson
- Test beds for I/O Research
 - Virtual machines and how do they help in our understanding
 - Need real/modern/relevant (many core) testbeds available to universities
- Applying cutting edge analysis tools to large scale I/O
 - Tracing moved here

General Comments

- Traces
 - Need more diverse workload traces, i.e. from data centers, data intensive (Google/Yahoo!) workloads
 - Want system level, not single application traces.
 - None of these gaps are specific to FSIO; encourage this community to look at tools from other areas (don't reinvent the wheel)

Current R&D Gaps

- Understanding system workload in HEC environments
 - Are there common/agreed upon points to trace or collect information at; we need to specify what these point are
 - Be careful how we do this; Pmpi is one example/success here, peruse is a non-success
 - Want “counters”
 - Should be called out under another bullet: Techniques to record and aggregate strings of unordered events(ex. Global clock) Way to order distributed events
- Standards and common practices for HEC I/O benchmarks and trace formats
 - Locality; benchmarks that measure the memory hierarchy do not
 - Standard for I/O traces is being worked – SNIA IOTTA; but this may be dead
 - First step is to define what information should be collected; maybe a family of formats
 - How to talk about traces, i.e. what is the information contained in the trace; metadata about traces; how do you understand that data you’re given
 - Collection at large scale becomes an issue
- Simulation and Modeling
 - Work on Normalization – How to compare vastly different file systems
 - UCSC work on simulator
- Test beds for I/O Research
 - No large testbeds to do research. Simulators are being built/used. Emphasis on simulators and models.
 - Virtual machines and how do they help in our understanding – probably fine for functionality testing
 - People are doing this but not enough resources; OpenSerrius – gives bare metal for testing
 - Need real many core (modern/relevant) testbeds available to universities
- Applying cutting edge analysis tools to large scale I/O
 - Scalability of collection of data and manipulation of amount of data; on-line techniques (Emphasis on Scalability)
 - Tracing discussion

More Discussion

- Integrate hooks into the system so others can collect traces later
- Testbeds should have built in tracing/ or at grant/funding time require/recommend tracing – encouragement
- Tracing – basic/high level format like IP packets;
- Encourage end-to-end approach – need to work with vendors