Cluster Scalability of Implicit and Implicit-Explicit LS-DYNA Simulations Using a Parallel File System

Mr. Stan Posey, Dr. Bill Loewe

Panasas Inc., Fremont CA, USA

Dr. Paul Calleja

University of Cambridge, Cambridge UK

Summary:

The parallel efficiency and simulation turn-around times of CAE software continue to be an important factor behind engineering and scientific decisions to develop models at higher fidelity. Most parallel LS-DYNA simulations use scalable Linux clusters for their demanding HPC requirements, but for certain classes of FEA models, data IO can severely degrade overall scalability and limit CAE effectiveness. As LS-DYNA model sizes grow and the number of processing cores are increased for a single simulation, it becomes critical for each thread on each core to perform IO operations in parallel, rather than rely on the master compute thread to collect each IO process in serial.

This paper examines the scalability characteristics of LS-DYNA for implicit and implicit-explicit models on up to 256 processing cores. This joint study conducted by the University of Cambridge and Panasas, used an HPC cluster environment that combines a 28 TFLOP Intel Xeon cluster with a Panasas shared parallel file system and storage. Motivation for the study was to quantify the performance benefits of parallel I/O in LS-DYNA for large-scale FEA simulations on a parallel file system vs. performance of a serial NFS file system.

The LS-DYNA models used for the study comprise cases that were relevant in size and physics features to current LSTC customer practice. The favourable results demonstrate that LS-DYNA with parallel I/O will show significant benefit for advanced implicit simulations that can be heavy in I/O relative to numerical operations. These performance benefits were shown to extend to a mix of concurrent LS-DYNA jobs that require concurrent data writes to a shared file system, which for an NFS-based file system would still bottleneck from its single data path for I/O. The paper also reviews CAE workflow benefits since, as an LS-DYNA simulation is completed, the same shared storage provides a platform for direct post-processing and visualization without the need for large file transfers.

Keywords:

High Performance Computing, HPC, Parallel I/O, Parallel File System, Linux Clusters, Storage System

1 Introduction

The combination of scalable CAE application software and high performance Linux clusters provides engineers and scientists with ongoing advancements towards a variety of simulations. The advantages for LS-DYNA with its efficient parallel scalability range from dramatic cost-performance improvements to high-fidelity solutions on clusters for simulations that were only recently judged as practical. For today's levels of advanced CAE simulation, I/O requirements have become a growing bottleneck that can often limit overall simulation scalability and workgroup collaboration. The use of parallel file systems are proven as an essential technology that enable commodity cluster environments to deliver their full potential in HPC scalability of both numerical and I/O opertations.

Automotive, aerospace, defense and manufacturing industries continue to face growing challenges to reduce design cycle times and costs; satisfy global regulations on safety and environmental concerns; advance military programs; and respond to customers who demand high-quality, well-designed products. Because of these drivers, the desire for production deployment of LS-DYNA and Linux clusters for high-fidelity multiphysics simulations, design optimization, and other complex requirements, continue to push LS-DYNA workload demands of rapid single job turnaround and multi-job throughput capability for users with diverse application requirements in a diverse HPC hardware and software infrastructure.

Additional HPC complexities arise for many LS-DYNA environments with the growth of multidiscipline CAE coupling of structural and CFD analyses, that all compete for the same HPC resources. Such requirements also drive I/O levels that prevent most system architecture's ability to scale. Yet for today's economics of HPC, the requirements of CPU cycles, large memory, system bandwidth and scalability, I/O, and file and data management – must be satisfied with high levels of productivity from conventional systems based on scalable, inexpensive clusters.

In order to manage the extreme I/O demands, entirely new storage system and software architectures have been introduced that combine key advantages of legacy shared storage, yet eliminate the drawbacks that have made them unsuitable for large distributed cluster deployments. Parallel NAS can achieve both the high-performance benefits of direct access to disk, as well as data-sharing benefits of files and metadata, that Linux clusters require for CAE scalability. That is, just as a cluster distributes computational work evenly across compute nodes, parallel NAS storage distributes data evenly across a shared file system for parallel data access directly between distributed cluster nodes and NAS disks.

As the number of compute cores are increased for single CAE simulations, in order to keep pace with fidelity and model growth, I/O operations should be performed in parallel to realize the essential benefits of overall simulation scalability. With a Panasas storage approach, each node on a cluster has direct access to read and write data on the shared storage and parallel file system, in order to maximize I/O performance during the computation phase of a CAE simulation. Once the simulation is complete, the same shared storage provides an end-user with direct access to the CAE results files for subsequent post-processing and visualization of the CAE simulation.

This paper examines HPC workload efficiencies for sample multidiscipline LS-DYNA applications on a conventional HPC Linux platform with proper balance for I/O treatment. Model parameters such as size, element types, schemes of implicit and explicit (and combined), and a variety of simulation conditions can produce a wide range of computational behavior and I/O management requirements. Consideration must be given to how HPC resources are configured and deployed, in order to satisfy growing LS-DYNA user requirements for increased fidelity from multidiscipline CAE.

2 LS-DYNA Applications in an HPC Environment

Finite element analysis (FEA) software LS-DYNA from Livermore Software Technology Corporation (www.lstc.com) is a multi-purpose structural and fluid analysis software for high-transient, short duration structural dynamics, and other multi-physics applications. Considered one the most advanced nonlinear finite element programs available today, LS-DYNA has proved an invaluable simulation tool for industry and research organizations who develop products for automotive, aerospace, power-generation, consumer products, and defense applications, among others.

Sample LS-DYNA simulations in the automotive industry include vehicle crash and rollover, airbag deployment and occupant response. For the aerospace industry, LS-DYNA provides simulations of bird impact on airframes and engines and turbine rotor burst containment, among others. Additional complexities arise from simulations of these classes since they often require predictions of surface contact and penetration, models of loading and material behavior, and accurate failure assessment.

From a hardware and software algorithm perspective, there are roughly three types of LS-DYNA simulation characteristics to consider: implicit and explicit FEA for structural mechanics, and computational fluid dynamics (CFD) for fluid mechanics. Each discipline and associated algorithms have their inherent complexities with regards to efficiency and parallel performance, and also regarding modeling parameters.

The range of behaviors for the three disciplines that are addressed with LS-DYNA simulations, highlights the importance of a balanced HPC system architecture. For example, implicit FEA using direct solvers for static load conditions, requires a fast processor and a high-bandwidth I/O subsystem for effective simulation turnaround times, and is in contrast to dynamic response, which requires very high rates of memory and I/O bandwidth with processor speed as a secondary concern. In addition, FEA modeling parameters such as the size, the type of elements, and the load condition of interest all affect the execution behavior of implicit and explicit FEA applications.

Explicit FEA benefits from a combination of fast processors for the required element force calculations, and memory bandwidth for efficient contact resolution that is required for nearly every structural impact simulation. CFD also requires a balance of memory bandwidth and fast processors, but benefits most from parallel scalability. Each discipline has inherent complexities with regard to efficient parallel scaling, depending upon the particular parallel scheme of choice. In addition, the I/O associated with result-file checkpoint writes for both disciplines, and increasing data-save-frequency by users, must also scale for overall simulation scalability.

Implementations of both shared memory parallel (SMP) and distributed memory parallel (DMP) have been developed for LS-DYNA. The SMP version exhibits moderate parallel efficiency and can be used with SMP computer systems only while the DMP version, exhibits very good parallel efficiency. This DMP approach is based on domain decomposition with a message passing interface (MPI) for communication between domain partitions, and is available for homogenous compute environments such as SMP systems or clusters.

Most parallel CAE software employ a similar DMP implementation based on domain decomposition with MPI. This method divides the solution domain into multiple partitions of roughly equal size in terms of required computational work. Each partition is solved on an independent processor core, with information transferred between partitions through explicit message passing in order to maintain the coherency of the global solution. LS-DYNA is carefully designed to avoid major sources of parallel inefficiencies, whereby communication overhead is minimized and proper load balance is achieved. In all cases the ability to scale I/O during the computation is critical to overall scalability in a simulation.

3 Parallel File Systems and Shared Storage

A new class of parallel file system and shared storage technology has developed that scales I/O in order to extend overall scalability of CAE simulations on clusters. For most implementations, entirely new storage architectures were introduced that combine key advantages of legacy shared storage systems, yet eliminate the drawbacks that have made them unsuitable for large distributed cluster deployments. Parallel NAS can achieve both the high-performance benefits of direct access to disk, as well as data-sharing benefits of files and metadata that clusters require for LS-DYNA scalability.

Panasas offers a parallel NAS technology with an object-based storage architecture that overcomes serial I/O bottlenecks. Object-based storage enables two primary technological breakthroughs vs. conventional block-based storage. First, since an object contains a combination of user data and metadata attributes, the object architecture is able to offload I/O directly to the storage device instead of going through a central file server to deliver parallel I/O capability. That is, just as a cluster spreads the work evenly across compute nodes, the object-based storage architecture allows data to be spread across objects for parallel access directly from disk. Secondly, since each object has metadata attributes in addition to user-data, the object can be managed intelligently within large shared volumes under a single namespace.

Object-based storage architectures provide virtually unlimited growth in capacity and bandwidth, making them well-suited for handling LS-DYNA run-time I/O operations and large files for post-processing and data management. With object-based storage, the cluster has parallel and direct access to all data spread across the shared storage, meaning a large volume of data can be accessed in one simple step by the cluster for computation and visualization to improve speed in the movement of data between storage and other tasks in an LS-DYNA workflow. Panasas provides this architecture by offering finely tuned hardware components that optimize the parallel file system software architecture capabilities.

4 LS-DYNA Performance Study

Performance and parallel efficiency for LS-DYNA is dependent upon many specifics of a system architecture and the implementation of MPI for that system, and the ability to scale I/O operations. Parallel computations require parallel I/O in some model cases, in order to scale the overall simulation. Structural FEA simulations in LS-DYNA often contain a mix of materials and finite elements that can exhibit substantial variations in computational expense, which may create load-balance complexities. The ability to efficiently scale to a large number of processors is highly sensitive to load balance quality of computations and data I/O.

For example, crashworthiness simulation of automotive vehicles exhibit such characteristics, of rapid gradient changes in the elements in an impact zone (and especially as models begin to approach multi-MM elements) whereas elements away from this zone observe small distortions. Similarly, an aerospace application for design of gas turbine engines for aircraft, has utilized the parallel scalability of LS-DYNA to reduce the time it requires to complete multi-MM-element models for blade-out simulation. There is a growing desire to combine both implicit and explicit time integration schemes in such blade-out simulations which requires run-time I/O to scale in order to scale the overall simulation.

As a demonstration on the benefits of a parallel file system to LS-DYNA cluster computing, resutls are presented for a variety of explicit and implicit LS-DYNA models on the large Linux cluster „Darwin“ at the University of Cambridge in the UK.

4.1 Cluster Environment

The University of Cambridge Darwin cluster was ranked 20th in the November 2006 Top 500 (www.top500.org) review of the world’s most powerful supercomputers, delivering 28 TFLOPS. At the time, Darwin was the largest academic supercomputer in the UK, providing 50% more peak performance than any other academic system.

Darwin is a cluster of 2340 cores, provided by 585 dual socket Dell PowerEdge 1950 IU rack mount server nodes. Each node consists of two 3.0 gigahertz dual core Intel Xeon (Woodcrest) processors, forming a single SMP unit with 8 gigabytes of RAM (two per core for four cores) running Scientific Linux CERN and interlinked by a QLogic InfiniBand. The cluster is organized into nine computational units (CUs) with each CU consisting of 64 nodes in two racks. The nodes within each CU are connected to a full bisectional bandwidth InfiniBand network which provides 900 MB/second bandwidth with an MPI latency of 1.9 microseconds. The CUs are connected to a half bisectional bandwidth Infiniband network to ensure that large parallel jobs can run over the entire cluster with good performance. In addition to the Infiniband network, each computational unit has a full bisectional bandwidth gigabit ethernet network for data and a 100 megabyte network for administration.

The cluster’s initial installation included 46 terabytes of local disk capacity and 60 terabytes of network file system storage provided by Dell PowerVault MD1000 disk arrays, with 15,000 rpm, 73GB SAS drives, connected to the cluster network over 10 gigabit Ethernet links. The storage pool was managed by the TerraGrid parallel file system. Additional details of the Darwin cluster can be found on the University of Cambridge HPCS Darwin overview web page: http://www.hpc.cam.ac.uk/darwin.html

Hardware

- 585 dual socket Dell PowerEdge 1950 server nodes
- 8 GB of RAM per node; 4.6 TB total memory

- InfiniPath QLE7140 SDR HCA interconnects and Silverstorm 9080 and 9240 switches1

Operating System

- Scientific Linux CERN SLC release 4.6

Software

- LS-DYNA 971

File System

- Panasas ActivStore AS5000, 4 shelves, 20 TB capacity.

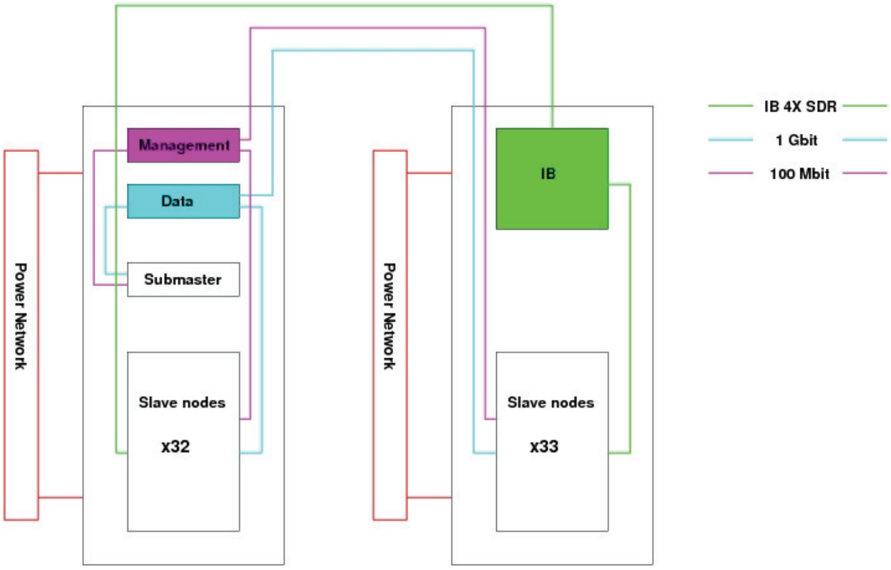
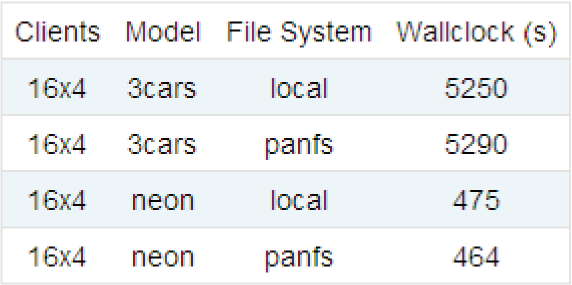


Figure 1: Schematic of 1 of 9 computational units with each CU consisting of 64 nodes in two racks

4.2 LS-DYNA Explicit Performance

The first tests conducted were the well known explicit models of 3car and refined-neon In order to set a base-line for expected LS-DYNA parallel efficiency. The choice was made to use 4 cores on each of 16 nodes for the test, for a total of 64 cores. Results compare the use of the local file system and the Panasas parallel file system PanFS:



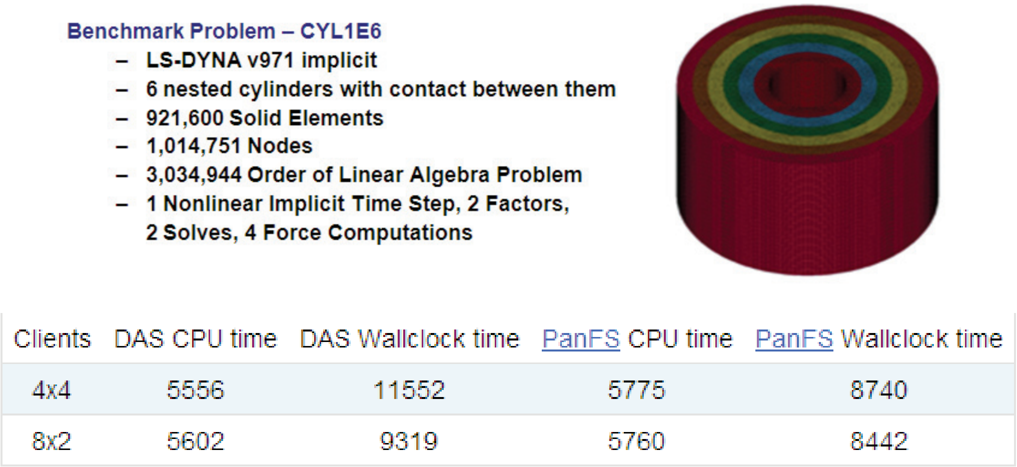
Clients	Model	File System	Wallclock (s)
16x4	3cars	local	5250
16x4	3cars	panfs	5290
16x4	neon	local	475
16x4	neon	panfs	464

Figure 2: Results of 3car and Refined-neon models for local file system vs. PanFS file system

The results provided in Figure 2. demonstrated that LS-DYNA was performing as expected, and that the Panasas shared parallel file system (network attached) was performing at the same speed as the local file system. This is an important result because the cluster was loaded with other jobs also with access to PanFS yet there was no degradation observed between PanFS and the local FS.

4.3 LS-DYNA Implicit Performance

Performance results of LS-DYNA implicit models should demonstrate the effect of data I/O much more than explicit owing to the use of sparse direct solvers that usually require an out-of-memory solution processing. This occurs because the stiffness matrix that must be factored is typically much larger than the allowable memory for a particular server or set of cluster nodes. The model in this case is comprised of 3M DOFs and a geometry of concentric cylinders that lends itself to efficient domain decomposition for the solver. Only 16 cores were used for this test but in configurations of 4 cores on 4 nodes and 8 cores on 2 nodes (in this case fully populating all cores on the node). A description of the model is provided in Figure 3.



Benchmark Problem – CYL1E6

- LS-DYNA v971 implicit
- 6 nested cylinders with contact between them
- 921,600 Solid Elements
- 1,014,751 Nodes
- 3,034,944 Order of Linear Algebra Problem
- 1 Nonlinear Implicit Time Step, 2 Factors, 2 Solves, 4 Force Computations

Clients	DAS CPU time	DAS Wallclock time	PanFS CPU time	PanFS Wallclock time
4x4	5556	11552	5775	8740
8x2	5602	9319	5760	8442

Figure 3: Description and results of implicit model CYL1E6 for the local file system vs. PanFS

The results provided in Figure 3. demonstrate that PanFS can perform substantially better than a local file system when comparing wallclock times. In the case of 4 cores on 4 nodes (4x4), PanFS was 32% faster than local (DAS) and for 8 cores on 2 nodes (8x2) PanFS was 10% faster. This advantage is due to I/O efficiencies because the CPU times for each (time spent in numerical operations) file system is roughly the same for both 4x4 and 8x2 as they should be, meaning numerical operations are independent of file system choice.

5 Conclusions

Joint studies conducted between research and industry organizations demonstrate that LS-DYNA with parallel IO on a parallel file system can show full parallel benefit for simulations that are heavy in IO relative to numerical operations. The favourable results were conclusive for a range of models on Linux clusters with a Panasas parallel file system. Benefits to industry include an expanded and more common use of implicit-explicit modeling.

A review was provided on the HPC resource requirements of various LS-DYNA applications, including characterizations of the performance behavior typical of LS-DYNA simulations on distributed memory clusters. Effective implementation of highly parallel LS-DYNA simulations must consider a number of features such as parallel algorithm design, system software performance issues, hardware communication architectures, and I/O design in the application software and file system.

Development of increased parallel capability will continue on both application software and hardware fronts to enable FEA modeling at increasingly higher resolutions. Examples of LS-DYNA simulations demonstrate the possibilities for highly efficient parallel scaling on HPC clusters in combination with the Panasas parallel file system and storage. LSTC and Panasas continue to develop software and hardware performance improvements, enhanced features and capabilities, and greater parallel scalability to accelerate the overall solution process and workflow of LS-DYNA simulations. This alliance will continue to improve FEA modeling practices in research and industry and provide advancements for a complete range of engineering applications.

6 Literature

[1] Gibson, G.A., R. Van Meter, "Network Attached Storage Architecture," Comm. of the ACM, Vol. 43, No 11, November, 2000.

[2] LS-DYNA User's Manual Version 971, Livermore Software Technology Corporation, Livermore, CA, 2007.

[3] M. DeBergalis, P. Corbett, S. Kleiman, A. Lent, D. Noveck, T. Talpey, and M. Wittle. The Direct Access File System. In *Proceedings of Second USENIX Conference.*

[4] S. Kodiyalam, M. Kremenetsky and S. Posey, “Balanced HPC Infrastructure for CFD and associated Multidiscipline Simulations of Engineering Systems,” Proceedings, 7th Asia CFD Conference 2007, Bangalore, India, November 26 – 30, 2007.