

Higher Returns on the Simulation Investment

Optimizing Linux clusters for ANSYS Mechanical software delivers fast turnaround on large problems.

By Joshua Bernstein, Software Engineer and Arend Dittmer, Director Product Management, Penguin Computing, California, U.S.A.

In many organizations, design engineers perform FEA (finite element analysis) simulations using desktop systems. With this approach, high-powered systems are required even though they are typically only fully utilized for short periods of time due to workflow interruption. Alternatively, solving more complex problems on shared back-end compute systems is more efficient. Offering excellent price/performance ratio, Linux®-based clusters have become a common platform for these back-end computations. However, the concept of achieving high performance through interconnected systems introduces performance and manageability challenges.

One of the biggest obstacles to a quick return on a cluster investment is the initial cluster setup. After deployment, the system configuration on compute nodes needs to remain consistent. Even minor discrepancies — a missed driver update on one system, for example — are hard to troubleshoot. In the case of a cluster upgrade or system failure on a compute node, the compute nodes need to be reprovisioned, leading to maintenance downtime. Moreover, it is difficult to identify and control processes in a cluster as compared to a single system.

To better understand optimal methods for installing and running ANSYS Mechanical software on a Linux cluster, a series of benchmark tests was performed by Penguin Computing and run on an Intel®, Cluster Ready certified cluster using Penguin Computing’s Relion® 1600 series servers, equipped with two dual-core Intel Xeon® 5160 CPUs. Scyld ClusterWare™ 4.20, a cluster management solution from Penguin Computing, was installed on the cluster [1]. Scyld ClusterWare is designed to make the management of a Linux cluster as easy as the management of a single desktop system and is fully compatible with Red Hat® Enterprise Linux®. With ClusterWare’s lightweight provisioning, compute nodes boot over the network from a master node into local memory. Avoiding a local operating system (OS) installation on every compute node guarantees configuration consistency, allows for easy node replacement and ensures cluster scalability. ClusterWare also provides a unified process space: All processes running in the cluster can be directly controlled from the master node.

Before an ANSYS Mechanical model can be solved in parallel, it must be decomposed so that the computations can be distributed. A sparse matrix is generated when performing the structural analysis and has a resulting size that depends upon the number of degrees of freedom in the model. Ideally, this matrix can be stored entirely in memory. In this case, the solver is run in the in-core (IC) mode. If the simulation run cannot be executed in-core, the ANSYS Mechanical software writes the sparse matrix to a file, passing through the disk input/output (I/O) subsystem. In this latter case, the simulation is run out-of-core (OOC), and the read/write I/O speed to the local scratch space greatly impacts solver performance.

Figure 1 shows performance results for ANSYS Mechanical benchmarks, tested using three memory configurations. As expected, memory configuration had the biggest impact on performance for the larger models — benchmarks 7 and 8. Significantly longer runtimes for these two models with the 8GB memory configuration occurred because they were solved out-of-core. Moving from an 8GB to a 16GB configuration resulted in a 32 percent

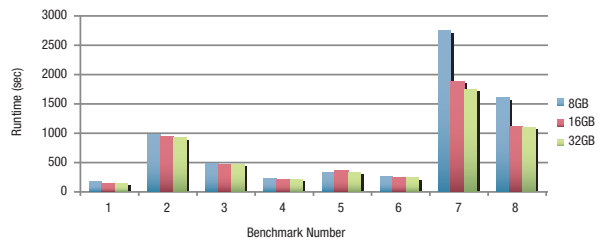


Figure 1. In-core vs. out-of-core performance comparison

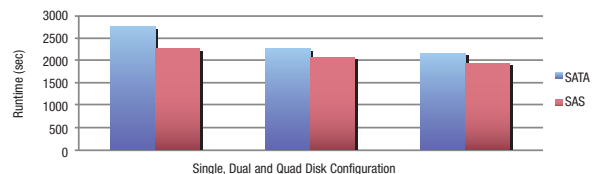


Figure 2. Relative performance of SATA vs. SAS disks

How to Obtain Optimal Performance?

Many users of ANSYS Mechanical software want to get the very best performance out of their software and hardware combination. Others wish to make the best hardware purchase decisions. Fortunately for both groups, ANSYS has put together a white paper on “Obtaining Optimal Performance in ANSYS 11.0.” This performance guide provides a comprehensive resource for ANSYS Mechanical users who wish to understand the factors that impact the solution performance of this software on current hardware systems. The guide provides information on ANSYS Mechanical computing demands, hardware considerations, memory usage, parallel processing and I/O considerations. The guide also includes general information on how to measure performance in the ANSYS Mechanical solvers and an example-driven section showing how to optimize performance for several ANSYS Mechanical analysis types and equation solvers. The guide provides summary information along with detailed explanations for users who wish to push the limits of performance on their hardware systems. Both Windows® and UNIX® or Linux® operating system issues are covered throughout the guide. This guide can be downloaded from the ANSYS Customer Portal.

— Ray Browell, Product Manager
ANSYS, Inc.

performance increase as the problem was now solved in-core. While the benefit from solving FEA solutions in-core may seem obvious, this characteristic is often overlooked when managing back-end compute systems.

To illustrate the influence of different local storage configurations on performance for an analysis model that runs out-of-core, benchmark 7 was run on a node configured with 8GB of RAM. Figure 2 presents observed runtimes for two different disk configurations. Serial attached SCSI (SAS) disk configurations performed up to 18 percent better than serial advanced technology attachment (SATA) configurations. SAS drives spin considerably faster than SATA drives — 15,500 RPM vs. 7,200 RPM — but their better I/O performance is partially offset by their higher cost and smaller capacities. Moving from a single disk configuration to a two-disk redundant array of independent drives (RAID0) configuration yielded performance gains of 10 percent for SAS disks and 18 percent for SATA disks, indicating a significant advantage for the parallel I/O provided by the RAID0 configuration.

Distributed ANSYS Mechanical technology spreads the computational workload of a single solver run across multiple systems. For benchmarking solver scalability, distributed benchmark BMD-4 was chosen. Each node in the cluster was configured with 8GB of RAM and used a RAID0 disk configuration consisting of four SAS drives.

Figure 3 illustrates the scalability of the BMD-4 benchmark. The job scales well on Ethernet and Infiniband®

fabrics. Marginal performance improvement is noted for the Infiniband fabric relative to Ethernet results. The cores used for this set of benchmark runs were allocated round-robin: Each process was launched on one core on a different system. After four cores on four systems had been allocated, the algorithm wrapped around and allocated the next core on the first node in the set, and so forth.

A second set of tests, shown in Figure 4, was performed using a node packing method for distributing processes onto cores. When scaling across an increasing number of cores, up to four processes were started on one node before moving to the next node. Better performance is achieved with the round-robin allocation method, as round-robin maximizes the amount of memory available for each process.

The presented benchmark results are useful for determining the optimal set of conditions for running an individual job. This does not consider the more realistic case in which many simultaneous ANSYS Mechanical runs need to be executed in parallel. Another option is to optimize for total throughput rather than for performance of individual jobs. When optimizing for throughput, the performance of individual jobs of the same priority has to be balanced against the number of simultaneous jobs running on the cluster. Best performance for high-priority jobs is achieved with round-robin allocation of cores on nodes that are dedicated exclusively to running one high-priority job at a time. ■

References

- [1] These benchmarks are described at <http://www.ansys.com/services/hardware-support-db.htm>.
- [2] To request the benchmark models, contact Shane Moeykens, shane.moeykens@ansys.com.

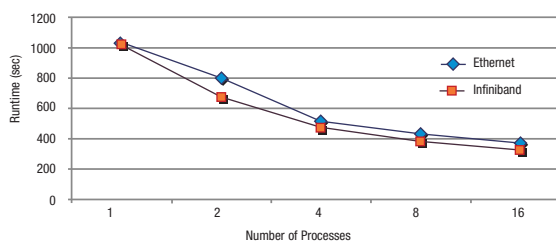


Figure 3. Distributed solver scalability

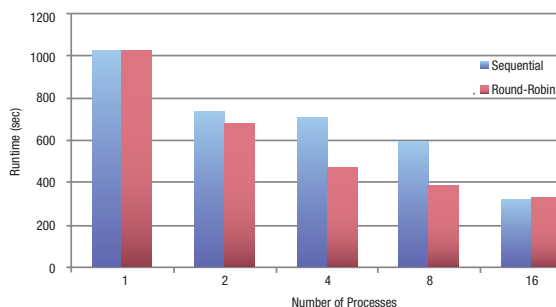


Figure 4. Performance comparison of round-robin vs. sequential core allocation