White Paper

SGI Altix Global Shared Memory

Performance and Productivity Breakthroughs for the SGI® Altix® 4000 Series and SGI Altix 450

Table of Contents

1.0 Introduction

Computer systems that implement a global shared memory (GSM) architecture while scaling to high processor counts offer a new dimension in the advancement of application performance. A GSM architecture can produce breakthroughs in achievable performance and user productivity, while offering entirely new ways of solving problems in science and engineering.

This paper describes the advantages of a GSM architecture with careful consideration of the data access hierarchy that is typical in high end computing. The speed with which a processor can access data drops by many orders of magnitude in the progression from on-chip cache, to off-chip cache, to system memory, and finally to disk. A well-implemented GSM architecture improves processor performance by maximizing the use of fast data access. Specific attributes of the SGI® Altix® NUMAflex™ GSM implementation are discussed along with real world examples of breakthrough computational results.

2.0 Background

2.1 Distributed Memory Architectures

Multiprocessor distributed memory systems (see Figure 1) consist of multiple “nodes” which communicate via a dedicated network of router elements and interconnect channels (links). Endpoint nodes generate and process the messages which travel across the network links whereas router elements simply forward messages toward their intended destinations. The computational, memory, and communications resources that give a system its computational capabilities reside within the individual endpoint nodes.

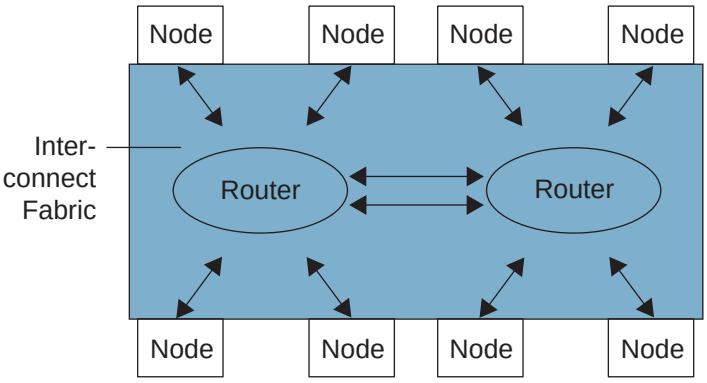

Figure 1. Typical system architecture with physically distributed memory.

In this discussion, we distinguish between two types of distributed memory architectures; those in which the local memory of a given node is implicitly visible and directly accessible from any processor or I/O agent in the system and those in which it is not. The former is a globally shared, distributed memory (GSM) architecture, while we refer to the latter as a segregated distributed memory (SDM) architecture (see Figure 2). Movement of data between endpoint nodes in a segregated architecture can only be accomplished via explicit I/O calls, whereas a globally shared architecture allows all processors in the system to use implicit direct access mechanisms regardless of memory location (local or remote).

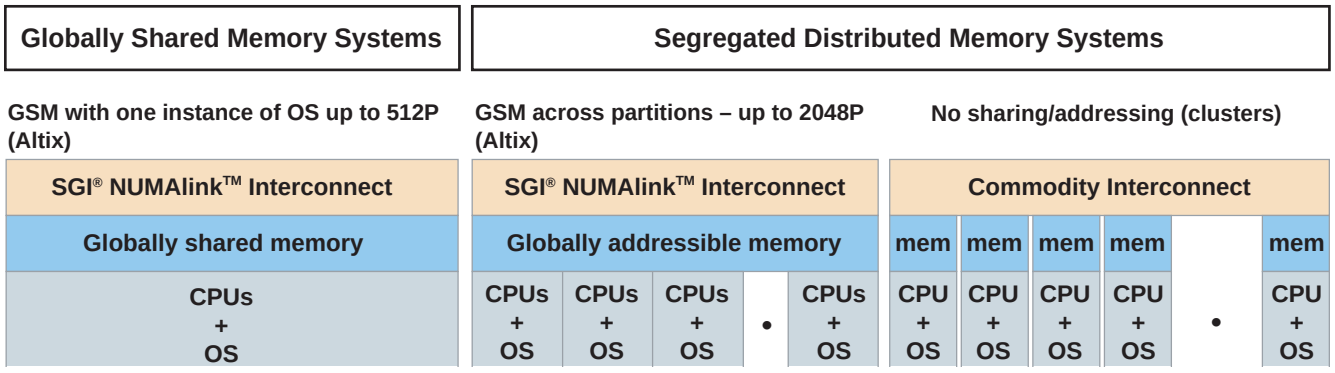

Figure 2. GSM versus SDM systems

2.2 Data Access Hierarchy

Recent years have seen rapid improvements in microprocessor cycle times. However, the performance of data storage outside the immediate processor domain (the process registers and on-chip caches) has not kept pace. In particular, main memory and I/O latencies have not improved nearly as rapidly. Today, on-chip processor caches provide access to data with latencies on the order of one or two nanoseconds. These latencies tend to scale directly with clock frequency improvements and thus track those trends closely. This contrasts strongly with typical local (on-node) SDRAM memory, delivering data to the processor core with an access latency of perhaps 100 nanoseconds—roughly 100X slower. Corresponding reductions in bandwidth occur as you progress through the data hierarchy as well.

Data Hierarchy Layer	Latency	Normalized Access Times
L1 Cache (1.6 GHz Itanium 9M)	0.625 nanoseconds	1
Local Memory	125 nanoseconds	2.0×10^2
Disk	3.6 nanoseconds	2.1×10^4

Table 1. Latency and resulting access times across the data hierarchy

These trends combine to form a tremendous gap between processor and memory performance. A generally accepted approach to mitigate the impact of this gap has been to rely upon larger processor cache subsystems in which successive levels of the cache trade off faster access times for larger capacity. Unfortunately, processor cache yields little benefit for applications that have random memory access patterns with very large memory footprints. The performance gap is especially severe in multiprocessor servers where sharing data may require traversing multiple cache hierarchies, spanning thousands of processor clock cycles.

While much has been written about this Memory Wall (a term coined by Wulf and McKee in their paper “Hitting the Memory Wall: Implications of the Obvious”), little has been written about the even more pronounced I/O Wall. As shown in Table 1, the Memory Wall is a result of the two orders-of-magnitude difference in cache and memory latencies. However, the latency difference between memory and disk—the I/O Wall—can exceed four orders of magnitude: Huge productivity gains are achieved when an application’s working data set fits entirely in main memory (as opposed to paging data in and out to disk). A familiar example of this is the common case of laptop or desktop PCs for which application performance (and therefore user productivity) is enhanced far more by installing additional memory than it is by upgrading to a faster processor.

Figure 3 summarizes the multiple order-of-magnitude differentials in latencies and bandwidth between levels within the hierarchy.

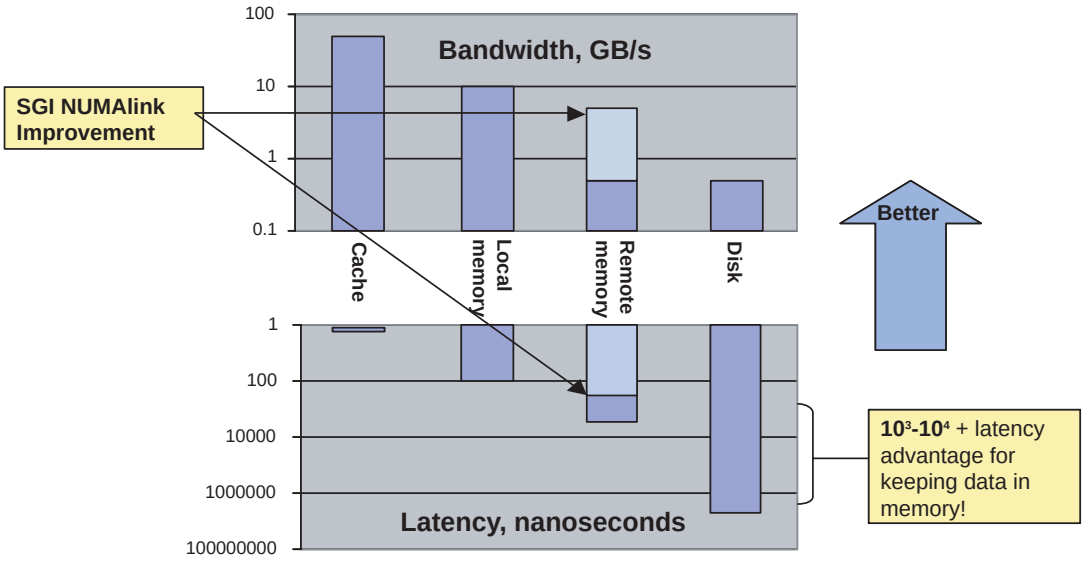

Figure 3. Bandwidth and latency worsen dramatically as data moves along the hierarchy from processor cache to disk storage.

3.0 The SGI NUMAflex Architecture—Massive Shared Memory Resources

In recent years, the capacity of global shared memory systems which can feasibly be constructed has increased dramatically. This advance has been led by SGI with successive improvements to the NUMAflex system architecture, introduced initially in the Origin product line. In addition, DRAM cost per bit has declined to the point where many terabytes of global shared memory is not only feasible but very affordable.

A key to making use of global shared memory is a system architecture that is able to provide efficient direct processor load-store access to the entire address space. With SGI's NUMAflex architecture, the SGI Altix system couples the large physical address space of the Intel® Itanium® processor, with a system interconnect capable of distributing that address space seamlessly across hundreds or even thousands of nodes. This means any load instruction that a processor issues can read any address in the entire global shared memory space of a system (on any node) exactly as it would read data in memory on the same local node.

In addition, the SGI NUMAflex system architecture as implemented on the SGI Altix® 450 and 4700 scalable blade servers allows independent scaling of processors, memory capacity, memory bandwidth, interconnect bandwidth, I/O connectivity, and I/O bandwidth.

3.1 Flexible Memory Scaling (Capacity and Bandwidth)

Memory capacity is expanded by adding more memory DIMMs into the system and can be accomplished in several ways:

- By increasing the number of DIMMs on existing system nodes
- By adding new processor-plus-memory nodes
- By adding memory-only nodes

The latter two options also increase the aggregate memory bank bandwidth in the system. The SGI NUMAflex architecture allows you to use any of these options in a flexible fashion. Notice also that the option to add memory-only nodes represents an efficiency not available in conventional clustered architectures since it eliminates the cost of a processor and related overhead. Unlike other systems that support large amounts of globally shared memory, Altix uses industry-standard DRAM DIMMs. Both factors work to make large capacity GSM efficient and affordable.

3.2 Advanced Features of Altix 450 and 4700 Servers

The SGI Altix 4700 platform is the culmination of years of focused effort. It is comprised of reliable, modular blades—interchangeable compute, memory, I/O and special purpose blades that deliver unprecedented configuration flexibility. The blade-to-NUMAlink™ architecture of the Altix 4700 enables users to mix and match eight blade options to create a scalable system with the exact capabilities they require—up to a current maximum of 1,024 processor cores and over 100 terabytes of globally addressable memory. Processing nodes in the Altix 4700 GSM design are provided by compute blades that provide either one or two processor sockets per blade (supporting dual core Intel Itanium 2 processors) and 12 DIMM slots supporting a maximum of 48GB of memory per blade when 4GB DIMMs are used. Memory-only blades with a similar DIMM configuration allow system memory to be scaled independent of processing capability for large memory problems.

The SGI Altix 450 Mid-range Server utilizes the same blade design to deliver system configurations scaling from 4 to 76 processor cores and up to 912 GB of shared memory.

3.2.1 Flexible I/O Scaling

System Category	Maximum Support Global Memory
x86 systems (e.g., Xeon™ or Opteron™)	32-64GB
RISC systems (IBM®, Sun®)	1-2TB
SGI Altix 4700	128TB

Table 2. Memory scaling comparisons for typical HPC platforms

The NUMAflex architecture of the SGI Altix 4700 and 450 also supports I/O Blades which can be flexibly configured in appropriate numbers according to the application requirements to deliver the desired amount of I/O to storage and networks. The blade design allows I/O to scale independently of processor and memory so you don't pay for resources you don't need to get the resources you do.

3.2.2 Integrated FPGA Processing

Both the SGI 4700 and the SGI 450 also support optional FPGA co-processing for optimal performance on specific algorithms using the SGI RASC™ (Reconfigurable Application Specific Computing) RC100 Blade. Traditional FPGA implementations are implemented on I/O cards and as a result have limited I/O performance, and are unable to take full advantage of system resources. By comparison, the SGI RC100 Blade acts just like any other node in the system, with direct access to Global Shared Memory and the full bandwidth of the NUMAlink interconnect. Each blade has two FPGAs and 10 DIMM slots. A complete software solution stack simplifies development. More than 50 algorithms have been ported to date with

substantial speed ups demonstrated. For instance, an Inverse Discreet Cosine Transform algorithm runs 35X faster on RASC than on a 1.5GHz Itanium II. The Altix 4700 supports up to 16 RASC RC100 blades per rack for accelerated parallel throughput while the Altix 450 supports up to 8 RC 100 blades.

3.2.3 Flexible Interconnect Scaling (Connectivity, Latency and Bandwidth)

The current SGI interconnect is called NUMAlink 4 and provides a raw single link aggregate transfer rate of 6.4GB/s (3.2 in each direction). Commodity clusters today use links with aggregate performance ranging from perhaps 100Mb/s to 2.5GB/s. NUMAlink 4 also includes the resources and protocols necessary to efficiently propagate coherent memory traffic across large system domains. Finally, NUMAlink interconnect technology delivers the industry's best hardware latencies: direct memory accesses to remotely located memory of a few hundreds of nanoseconds versus 2-30 microseconds for competing interconnects. Within the context of the NUMAflex architecture, NUMAlink interconnect technology provides the ability to independently scale both endpoint connectivity and system-wide bandwidth allowing individual systems to be optimized for particular applications which have differing requirements of bandwidth between nodes and mixes of various endpoint resources.

4.0 Benefits of Global Shared Memory

A system that supports global shared memory provides a number of significant performance, productivity, and system administration benefits over segregated architectures (clustered systems) which connect multiple nodes together via I/O-driven network interface cards (NICs).

4.1 Massive In-core Computation by Design

A GSM architecture allows much larger and more detailed models of physical systems to be entirely memory resident. Consider for example modeling not only the turbine blades of an aircraft engine, but the complete engine. Lacking adequately sized global shared memory, one must employ one or more compromise strategies to make the computation feasible. This may include:

- Reducing mesh resolution—the number of data points used to represent the physical entity being modeled
- Breaking the problem down—managing it as a set of smaller pieces, perhaps dealing with intermediate results stored in scratch files on disk
- Reducing numerical precision—dropping to 32-bit representations
- Approximated emulation of component subsystem behaviors—often using algorithmic models and artificially constrained properties

All such strategies represent potentially significant compromises in computational accuracy, performance, and productivity.

4.2 Massively Memory-Mapped I/O

For applications that are bound by random I/O accesses on large data sets, up to 10,000X performance increases can be gained by bringing the entire dataset into main memory. This becomes even more compelling when considering the price difference between memory versus disk. Current price trends for memory and disk technology predict only at most two orders of magnitude between these storage layers. This alone yields a price-performance advantage over disk I/O-based operations of 100X.

For example, globally shared memory can be exploited to dramatically improve database performance. Just as the move from storing databases on magnetic tape to magnetic disk allowed much higher performance through direct random access, global shared memory enables similar order of magnitude improvements in database performance. A large, global shared memory system can entirely eliminate the need to page data in from disk.

4.3 Highly Efficient Message Passing

A GSM system is obviously well suited to shared memory applications. However, it can also deliver superior performance for distributed memory applications (for example MPI message passing) where all communications and data are explicitly defined and allocated. Moreover, when the ability to directly load and store to remote memory is correctly exploited, most software overhead is eliminated yielding substantially reduced effective latencies. Altix systems utilizing the NUMAlink interconnect demonstrate superior efficiency for MPI.

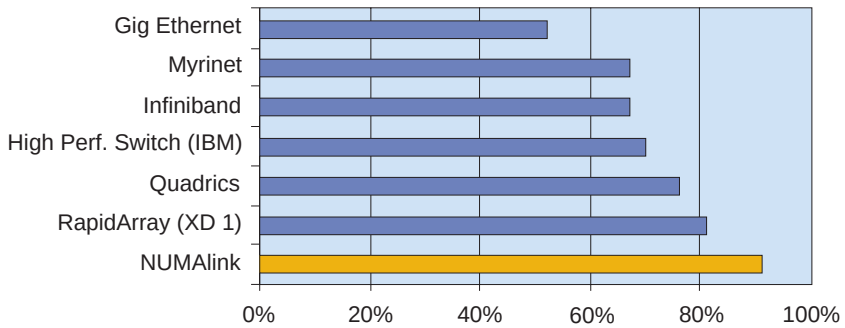

Figure 4. Interconnect efficiency for common MPI applications. Calculated as Linpack NxN Rmax/Rpeak. Source: Top500.org

4.4 Arbitrary Scaling of Problem Size

Applications ported to systems with a distributed memory organization (globally shared or segregated) are often optimized by decomposing the input datasets and carefully placing the pieces in physical memory local to the processors that will most often interact with them. Consider an input dataset growing such that the partitioned pieces exceed the capacity of the individual nodes. On a clustered system, a major repartitioning or possibly a complete change of algorithm may be required in order to maintain a reasonable level of performance. But with a well-designed GSM system, the latency and bandwidth penalties for fetching off-node data is comparatively minimal (whether addressing the memory directly or using message passing access). This means that simply by adding memory-only resources, one can run the same application without significant modification, and still obtain the needed performance and correct computational behavior.

4.5 Efficient Utilization of Memory Resources

Global shared memory unifies system memory resources that are (by definition) fragmented when using an SDM approach. Porting certain applications for execution on a cluster requires that some amount of the dataset be replicated—often onto each cluster node in the system. This wastes system memory capacity and effectively increases the cost per bit for memory resident data. Data replication results in multiple data copies that must be managed, increasing the burden on the system or application. A GSM implicitly avoids these complexities.

For instance, in a distributed ray tracing application running on a commodity cluster each node in the cluster must have the entire model replicated locally or the application will suffer unacceptable degradation in latency and bandwidth. Once the model size exceeds the size of a given node's memory, no further scaling of problem size can be obtained on that system (at least not without heroic efforts to implement complicated data caching schemes). On a GSM system, the effective cost-per-bit economy and the ease of scaling datasets to essentially arbitrary size yields a substantial advantage over other architectures.

4.6 Greatly Simplified Load Balancing

For many real-world parallel applications, processing resources are wasted due to load imbalances. With global shared memory, it's a simple matter to direct a processor that has finished one task to start on another since the data associated with all such tasks are accessed via a single common address space. In contrast, load balancing on a cluster requires data for a new task to be explicitly copied to the node in question before work can start, creating significant overhead in managing copies of data and results.

4.7 Efficient Development and Administration for Reduced TCO

When developing new codes, use of simpler programming models makes it easier to experiment with different algorithms. Shared memory programming provides a simple computational paradigm upon which to rapidly prototype and evaluate application codes. Well over half of all Altix users do some level of code development, benefiting from the ease of use afforded in up to 1,024 processor cores in a single, unified development and operating environment.

Further, managing a collection of compute resources within the context of a single instance of an operating system represents a fundamental reduction in costs associated with system administration. A recent study of total cost of ownership for high end computing systems (Muzio and Walsh, 2003) estimates that the equivalent of one full-time person is required to administer every 128 nodes (in this case, 'node' refers to a specific instance of an operating system). For example, a modern 256 processor cluster of 128 dual-processor nodes would yield a system having a peak compute power of perhaps 1.5 teraflops and would require one full time person just for node administration. In contrast, the SGI Altix system delivering the same level of performance in a single node of 256 processors can be configured as a single operating system image, requiring minimal administrative resources.

5.0 Real-Life Performance Gains and New Possibilities

5.1 Accelerating Applications

Industry-leading MPI results—SGI's NUMAlink interconnect technology offers industry-leadership in MPI latency and bandwidth (6.4GB/s bi-directional per link), delivering great performance on MPI codes. As shown in Figure 4 above, SGI Altix shines in Linpack efficiency as it does on larger-scale applications, especially for jobs scaling past a few processors.

Scaling to 512P and beyond at NASA—NASA has achieved extraordinary scaling of commonly used applications with minimal effort due in large part to what they describe as the 'trivial' nature of load-balancing on their 512 processor global shared memory Altix systems. They have done this through use of a programming model they have termed "multi-level parallelism" (Taft, 2000). This model has reduced programming time for scaling and optimization of very large, complex applications from many man-years down to a few weeks. The 512P results have rivaled and even exceeded those of the $500M Earth Simulator.

Bioinformatics—A Europe-based pharmaceutical research institute has developed a genomics search and matching algorithm that produces results up to 1000x faster than the

commonly used BLAST application. This algorithm relies on 192GB of memory available on each of two SGI Altix servers with 4 and 16 processors.

5.2 Scaling Problem Size

NWChem—The popular code NWChem is routinely run at one SGI customer site on only 4 processors of an SGI Altix system but using 80% of the 2TB of globally shared memory configured in that system. While execution may not scale efficiently across more than a modest handful of processors, big gains can be had simply by exploiting additional memory capacity.

Nastran—A similar example is Nastran, which normally does not scale beyond a few processors. This code sees considerable speedup when the job has access to larger shared memory.

5.3 Real-time Interaction with Massive Datasets

Large-scale seismic dataset interaction—Marathon Oil is continuously challenged to deliver larger and larger seismic datasets to interpreting geoscientists. For the first time, geoscientists can visualize and interact with more than 400GB of seismic data in real time. This is more than 4 times the previous record.

Real-time transaction and analytics processing–A leading casino management firm leverages SGI Altix in managing their operations. They summarize a gaming day into multiple Oracle tables so users can access Oracle Discover for marketing analysis and slot analysis. The result is a mixed environment with online trans-action processing and online analytical processing. Competitive systems need multiple servers to do this, but SGI's customer can do it all with the bandwidth of a single SGI Altix 350 server and avoid the hassle of trying to keep several systems in sync.

Memory resident databases such as Oracle Times Ten—In-memory databases, such as Oracle TimesTen, are increasing the options available to leverage Altix GSM and NUMAflex. In a customer benchmark, an Altix (with 1TB of memory) and Oracle Times Ten solution scaled to handle an in-memory database with more than 10 billion rows of data. Ingest rates were one order of magnitude faster than the performance target, and query person rates were one thousand times faster. Join person/order rates were a remarkable one million times faster than the performance target. The overall search performance of the Oracle TimesTen database on the SGI Altix system was over one thousand times faster than the customer target, far exceeding the capabilities of any other server on the market today.

5.4 Unprecedented Computational Options

Massive structural analysis—Recently, ANSYS announced that it became the first engineering simulation company to solve a structural analysis model with more than 100 million degrees of freedom (DOF[1]), making it possible for ANSYS customers to solve full-resolution models of aircraft engines, automobiles, construction equipment and other complete systems. In a joint effort with Silicon Graphics, Inc., the 111 million-DOF structural analysis problem was completed in only a few hours using only a handful of processors on a large memory SGI Altix computer. As a result of this work, ANSYS entered into a three-year partnership with SGI to advance the capabilities of ANSYS in parallel processing and large memory solutions.

Rapid response data analysis—A US Government Agency is using 32-processor SGI Altix systems equipped with 4TB of memory as the only solution capable of handling the massive data mining and search algorithms essential to their work.

6.0 Summary

The advantages of a global shared memory architecture are evident both in common high performance computing workloads and in grand challenge breakthroughs. Looking at the order-of-magnitude latency and bandwidth differences along data access pathway demonstrates the obvious benefit of keeping datasets and simulations fully resident in system memory. With the additional benefits of simplified load balancing, development, and system administration, the value of this versatile architecture delivers true return on investment. SGI Altix technology makes it possible to build systems with massive global memories that scale independent of processor count up to 128TB. The value of these systems has already been demonstrated across a wide range of problem types. Future achievements with very large global shared memory architectures are limited only by the imagination of application developers and users.

[1] DOF refers to the number of equations being solved in an analysis giving an indication of a model's size.

Corporate Office
1140 E. Arques Avenue
Sunnyvale, CA 94085
(650) 960-1980
www.sgi.com

North America +1 800.800.7441
Latin America +55 11.5185.2860
Europe +44 118.912.7500
Japan +81 3.5488.1811
Asia Pacific +1 650.933.3000

3785 [04.20.2007] **J14854**