The SGI® Roadmap – Taking HPC to the Next Level

Michael Woodacre, Chief Engineer,
Server Platform Division, SGI

Background

A number of technology trends over the past decade have shaped the direction of high performance computing. Commodity clusters, which now dominate HPC, have benefited from the continued performance increases seen with x86 processor cores. Consistent with the rise in cluster use, the MPI programming paradigm has become extremely popular over the last decade as it helped programmers build their codes in a highly portable fashion. These trends have allowed applications to ride the rapidly improving performance curves of mass market processors instead of being captive to a particular architecture or vendor.

But while Moore's Law growth of transistors promises at least a few more process generations ahead, the ability to translate this directly into processor core performance has become a major issue. Similarly, the expansion of cluster systems to larger scale has not yielded the performance gains hoped for by high end computing users. SGI has adapted its HPC strategy to these trends, taking a commodity-based approach, while developing important capabilities that are critical to moving beyond the limitations of Moore's Law and cluster-based parallelism

The SGI® Altix® System Development

SGI has been designing scalable shared memory systems for many years. By the late 1990's, proprietary components were increasingly hard to justify given the rising performance of lower cost, mass market parts. SGI chose to focus on providing value to its customers through its scalable system architecture.

The first system to be delivered to the market using this approach was the SGI Altix server. First introduced in 2003, the Altix moved away from the proprietary based designs of the earlier Challenge and Origin systems, to an approach that leverages more COTS (common off the shelf) components.

1. Intel® Itanium®2 processor (replacing MIPS).
2. ATI Graphics processor (replacing proprietary SGI designs).
3. Standard Linux® OS (replacing proprietary IRIX).
4. Commodity SDRAM DIMM (replacing proprietary DIMM designs).

This strategic direction has allowed SGI to continue delivering state of the art system performance to its customers, while driving down system costs to meet ever increasing market expectations. This performance is delivered through the powerful SGI NUMAlink™ interconnect and NUMAflex™ scalable shared memory architecture.

SGI NUMAlink Interconnect and NUMAflex Architecture

The NUMAlink interconnect is the fabric that binds all the elements together in SGI's NUMAflex system architecture. Multiple generations of NUMAlink have been designed sharing a number of common elements. Firstly, high bandwidth per pin. This is key to building scalable systems where physical packaging constraints come into play. For example, it helps the design of router ASICs with strong bandwidth and radix characteristics. Low latency interconnect is also key to building scalable systems. Improving the latency of critical operations within the interconnect provides the ability to increase the level of parallelism that applications will be able to effectively exploit. NUMAlink was designed to have an encoding of messages on the physical channel that is optimized for the transfer of small messages, while maintaining highly reliable data transport.

System packaging of scalable systems also plays significant importance as designers look to squeeze latency from interconnect paths. NUMAlink has been designed to deliver high performance over printed circuit boards as well as cables. This allows a uniform interconnect approach for scalable systems, both within a packaging enclosure, and for cabling packaging enclosures together. It also removes the need to add additional buffers etc. to drive signals between different components.

A fundamental aspect of the NUMAflex system architecture is the provision of globally addressable memory. This allows the processing elements to directly operate on data through load/store semantics at the hardware level, without the need to go through

a software layer as a cluster based approach requires. It's also important to note that a system with shared memory supported at the hardware level does not restrict you to only running applications designed to take advantage of shared memory. In fact, the SGI Altix system delivers industry leading MPI performance, even though MPI was designed to suite distributed memory or cluster style systems. In addition, globally addressable memory also makes operations such as dynamic load balancing more effective.

Scalable shared memory provides the foundation to build an operating system that can handle large processor counts. Altix currently supports a single operating system image size up to 512 processors. This can significantly reduce the administration costs of large HPC systems. Imagine the ease of managing just one operating system for 512 processors, as compared with the effort involved with 256 copies of the operating system in a cluster of 2P nodes. Beyond this, a large operating system image can also provide users with a more productive environment to work within. Users can easily manage their processes and datasets without a need to co-ordinate between systems. Users are also able to take advantage of a broader set of programming paradigms, such as OpenMP programming, up to much larger processor counts.

Multi-Paradigm Computing

With the barriers encountered to the advancement of processor core performance, general purpose processors have now turned to integrating multi-cores on a single chip to make use of the increasing number of transistors. This approach is attractive to the chip manufacturers as it is a relatively easy way to make use of the silicon area. However, not all applications will benefit, especially HPC applications that tend to require high bandwidth access to off chip memory. Further, the power and cooling problems associated with these designs present additional barriers to progress in this mode.

Given this scenario, industry observers and users alike have begun to recognize the need for a more effective approach to performance. SGI recognized this some time ago and therefore embarked on a development path to advance its shared memory architecture by supporting a variety of tightly coupled alternative processing elements. The first example of this approach can be seen with the visualization subsystem of SGI servers, where GPUs (graphics processing units) are deployed to accelerate graphics processing. These were initially connected into the system through standard IO interfaces (PCI-X/AGP).

More recently, SGI introduced the capability to *directly* couple novel processing elements into the NUMAflex system architecture. SGI's TIO ASIC provides a standard IO interface and in addition, it has a scalable system port (SSP), which opens up the NUMAlink interconnect to novel devices. Not only can these devices have full access to the high bandwidth NUMAlink interconnect, but the NUMAlink communication protocol is also opened up so these devices can interact directly with all shared memory in the system, as well as other mechanisms such as atomic memory operations.

The first product to exploit the SSP port couples FPGA (field programmable gate array) technology directly into NUMAlink. FPGAs, like general purpose processors have gained greatly from Moore's law. However, unlike scalar processors which have hit the wall with regard to increasing single thread performance, FPGAs can directly take advantage of the continued growth of available transistors. FPGA programming techniques directly expose the parallelism in an algorithm to the hardware so that it is possible to gain orders of magnitude performance improvements for some algorithms/ applications. SGI has introduced this technology to the marketplace as RASC™ – Reconfigurable Application Specific Computing.

Now that SGI has opened up the capability to attach novel devices to its scalable system architecture, it is working with a number of partners to exploit this ability and deliver exceptional performance gain for a variety of HPC applications. One example is the Clearspeed™ floating-point accelerator. Systems can be configured to deliver the best-performing processing elements for particular applications.

The move to multi-core processors also raises interesting questions about the direction of programming models. MPI programs have typically been written to run well on cluster based systems with multiple micro-second latencies. With multi-core and multi-threaded processors, it would be of great benefit if the programming model could take advantage of the evolving hierarchy of communications latency. Here is where the global addressable memory approach of the NUMAflex architecture allows seamless scaling of programming models. Programmers can and already do take advantage of the growing power of multiple processors and compute nodes, by reducing the need to re-code or re-optimize as operations that were once separated by node or chip boundaries move across those boundaries with ease.

Looking Ahead: SGI's Project Ultraviolet

Project Ultraviolet will build on the multi-paradigm capabilities delivered with Altix. The system architecture will be able to scale from a single node, all the way up to Petascale systems. Key advances will include:

- A new generation of interconnect, that will increase the global addressing reach, and implement communications protocols to increase the efficiency of packet and message level data transport.
- The incorporation of novel processing elements in addition to robust support for the next generation of Intel processors as the general purpose processing elements
- A second generation of the SSP to provide even greater control to these devices to increase data transport efficiency within the system architecture.
- A new data transport capability to deal with algorithms that traditionally mapped onto a vector paradigm. This will be used to supplement the microprocessors when dealing with data items that don't fit the cache-line orientated designs that mass market processors use.

Ultraviolet comprises a truly elegant combination of both flexibility and performance; one that can support everyday workload demands with a new level of productivity, while scaling up to power the next grand challenge problems which can't afford the limitations of today's clustered processor approach.

Summary

The HPC industry is facing new challenges as IC technology continues to deliver on Moore's law growth of transistors, but cores used at the heart of many cluster based systems stall in the delivery of better single thread performance. Novel computing elements, such as FPGAs and highly parallel floating point accelerators, are offering new potential to drive application performance forward. As we move towards Peta-scale computing, what will the programming models be to make effective use of such systems? SGI is taking a multi-paradigm approach, with its globally addressable memory architecture as the foundation, to build cost effective, scalable and versatile systems. SGI is already delivering on this vision, with the technology to build innovative solutions that directly address the problems of building high performance, high productivity systems.

HDF2AVS - A Simple to Use Parallel IO Library for Writing Data for AVS

Craig Lucas and Joanna Leng
Manchester Computing, University of Manchester

Motivation and Background

HDF2AVS is a library of routines to write out data, in parallel, in the HDF5 (Hierarchical Data Format) [3,4] data format, for input into the visualization system AVS/Express (Advanced Visual Systems) [1,2]. It is available as a Fortran 90 module and consists of various routines for writing out different types of array or coordinate data.

HDF uses a tree structure of groups and datasets. A group consists of zero or more datasets and other groups, together with supporting metadata. Datasets contain multidimensional arrays of data elements. The library is still in development at NCSA (National Center for Supercomputing Applications) [6]. We chose HDF for many reasons:

- it is a user defined format like XML
- it is a binary format that allows compression so drastically reducing the size of data files
- it is a format with longevity (NetCDF4 [7] is to be implemented on top of it)
- there is a dump facility that allows users to investigate the contents of binary files easily
- there is a reader for HDF already within AVS/Express
- parallel IO is supported.

There are many advantages to writing out data in parallel. On parallel machines there are two traditional approaches to writing data. One is to collect all data to one processor and then write this to disk. This obviously creates a communication overhead and can be slowed down further if there is not enough local memory on this master processor to hold the entire data set. These problems can be avoided by the other standard approach