



HPCTimes

April 2003

IN THIS ISSUE:

- ▶ **News from Microway®** [Pg. 1]
 - ▼ *Navion™ Clusters and Servers Launched in NYC*
 - ▼ *Customer News*
- ▶ **You Can Count on It** by Stephen Fried, President and CTO [Pg. 2]
 - ▼ *First Look: The AMD Opteron™ Processor*
- ▶ **Parallel Thoughts** by Ariel Cohen, Principal Scientist, Topspin Communications [Pg. 4]
 - ▼ *RDMA Offers Low Overhead, High Speed*
- ▶ **Microway Spotlight:** Nina Nitroy, Technical Support Manager [Pg. 4]

New From Microway

Launch of Navion™ Clusters and Servers

Microway's new Navion server/cluster nodes were introduced at the April 22 launch of the AMD Opteron™ processor in New York City. The Navion is offered with dual or quad AMD Opteron processors in 1U or 4U chassis, with up to 32 GB of memory and multiple PCI-X slots. Microway clusters include NodeWatch™/ MCMST™ proprietary remote cluster monitoring and management tools.

Cluster News

San Jose State University has purchased a 58-node Microway GigaCluster™ built with dual Athlon MP 2000+. The cluster includes Microway's NodeWatch™ Remote Cluster Monitoring and Management Tools and a RAID 5 storage subsystem. It will be used as both a research and educational platform. The major research applications involve numerical modeling of the atmospheres of Earth and Mars, with software and hardware optimizations under the direction of Dr. Scot Rafkin, Dr. Robert Chun, and Dr. Robert Bornstein. The cluster also provides an innovative platform for hands-on education in computational meteorology and computer science.

You Can Count On It by Stephen Fried, CTO

First Look: The AMD Opteron™ Processor

I had the great pleasure to attend the AMD Opteron™ launch on Tuesday, April 22, in New York City. Executives from AMD, IBM, Computer Associates, Microsoft and Oracle told the press how easy it was to port their software to Opteron. Of course, what they were telling us was obvious – since the Opteron supports the same 32-bit instruction set as the Intel® Xeon™, porting to it should be a snap. In fact, you don't even have to port to it, all you need to do is run your code. If it doesn't run, something is wrong. However, if you have code designed to run on both 32- and 64-bit machines and you have “if def'd” your C++ sources, inserting macros to handle 32- and 64-bit issues (something required to get code to run on both Alphas and Xeons), then even porting to the new architecture is still going to be a snap, as the only thing that will

change is the size of the integers. Applications which will benefit easily from 64-bits ought to be file servers and data base managers, which use large memories to cache data. While it will be fairly straightforward to get the memory and cache size based wins, its going to take a bit longer to see the total benefit which AMD64 technology can bring to the floating point world. To really hit full speed, you will need to take full advantage of the Intel SSE2 instruction set along with the extensions to SSE2 that AMD added to the Opteron. And, as of this writing, even though PGI is working this problem, the best compilers by far for generating high quality SSE2 code still come from Intel.

The Opteron brings a lot to the HPC table. It contains a number of features that have appeared in the past in parallel processing engines, including the Inmos Transputer and the Alpha EV79. All of these devices contain a memory controller built into the chip in addition to the CPU. They also contain additional circuits which make it possible for the CPUs to talk to their nearest neighbors. There are several major benefits to this approach. First, Opterons can communicate between themselves without having to send packets down a bus which they share with other processors and which terminates at a memory controller hub. In a two processor Xeon system, in which the Northbridge (memory controller) has the same bandwidth as a single Opteron, the Northbridge is shared between two CPUs and the total bandwidth available to a Xeon pair will be half that available to an Opteron pair – the Opterons in this situation have two memory controllers compared to the Xeon pair’s one. In addition, the time to fetch data out of memory for the Opterons will be less, simply because the only off-chip access will be the control signals required to drive the memory bus.

Next, in an Opteron pair, each processor can access the memory of the other. This is done by a low latency high speed bus which maintains internal cache coherency, and is known as the coherent HyperTransport™ bus. The efficient Inmos Transputer parallel architecture, in which each CPU had its own memory controller and four CPU interconnects called Links, is closely mimicked by the Opteron. Altogether there are three instead of four Links, which use a bus that takes the new name contributed by AMD – HyperTransport. Two of the three hubs are coherent are used to link Opterons together. The third is not, and is used by the processor to talk to lower speed I/O busses. The current generation of Opteron can link four CPUs together – in the future this will jump to eight.

HyperTransport is a “pass it along” style interconnect, i.e., a sequence of HyperTransport devices can be linked up in a row and talk to each other. This is not exactly what happens in a bus, and the reason this new approach had to be taken is the typical problem that we encounter with all electronics when it comes time to speed things up. The problem is called fan out. In a bus, everyone sits on the same group of parallel lines and listens in on the conversation. When the message is intended for them, they figure that out by decoding the bus’s addresses and turning on their latches, which hook the data. The problem with high speed busses is that all of those ears on the parallel lines end up loading down the lines, making it impossible to get really high speed performance. This is the reason that the number of PCI slots on a PCI hub (a gadget on a motherboard which distributes signals to the PCI bus) goes down as the frequency goes up. A typical hub can support four slots at 66 MHz, two slots at 100 MHz, but just one at 133 MHz. Although busses are an acceptable methodology when bus frequencies are on the order of 33 to 66 MHz – they don’t work with signals running above 100 MHz, which is where the HyperTransport runs. So, another technique was needed to make it possible to interface a number of devices, say four PCI slots (each can be thought of as a device when something is

plugged into it). The technique used is to eliminate the snoopers. In HyperTransport, the signal moves from point to point down a series of chips, which act as bridges. When a packet of information is intended for them, they pass it along to their client device; otherwise they pass it along to the next HT device in the chain. The typical time (I am using numbers here for typical LVDS devices) required to make a hop using the LVDS “pass it along” paradigm is on the order of 100 ns per hop. This means if a part is on one end of an HT chain and it wants to talk to one on the other end, it will take several hundred nanoseconds for the message to run the gauntlet. However, properly buffered, the added latency of the paradigm will not affect bandwidth, as each part in the chain contains input and output buffers which are set up as FIFOs. The case discussed above is dedicated to performing I/O.

In the case of an Opteron motherboard with four CPUs, the worst case latency is a two hop transfer across the square. This adds just 70 ns. For inter-processor communication the coherent HT bus is extremely efficient. Even when used to drive a long chain of peripheral devices, it is still very efficient, as the typical latencies of I/O peripherals run from 10’s of microseconds all the way out to milliseconds, which basically means that the half microsecond you waste going down the chain is not observable.

Finally, we get to the real benefit of coherent HyperTransport. In a ring of four processors, the busses between the CPUs can be simultaneously active. This is the same thing we saw with Transputers, where the most common parallel topology turned out to be the ring. In the case of a ring, the total bandwidth of the interconnect is the total number of active connections times the speed of a single interconnection. In the case of a four way ring, 66% of the communications will be with the nearest neighbors, and 33% with the diagonal member. And, if two or more processors are active at the same time on the HT interconnect, it becomes possible to effectively increase the inter-processor bandwidth. The only disadvantage of the HT paradigm is inter-processor latency. Using a single bus ala Intel to connect four CPUs together, only a single processor can be on the bus at a time, and the latency between that processor and the only thing it communicates with, the Northbridge, will be less than the latency of a cross corner transfer on the HT solution – communications can occur on a bus without hops, reducing inter-processor latency. However, as we mentioned above, the Opteron more than makes up for this by virtue of the fact that each processor has its own private on chip memory controller which eliminates the extra time a bus architecture needs to access its memory controller hub.

While Transputers talked to each other using links, Intel Xeons don't have any such feature. Rather, when two Xeons talk to each other, they actually do that by sharing memory. Therefore, if two Intel processors on the same bus want to share information, what they actually have to do is read and write memory through the Northbridge. On the other hand, Opterons can access each other's memory over the coherent HyperTransport bus, and it turns out that these off chip fetches only take between 100 and 140ns. This is roughly the same time that a single Intel Xeon can access its memory (AMD claims the latency of Athlon/Xeon motherboards with Northbridges are typically 170ns). The bottom line is, because the pair of Opterons communicating in cross court memory transactions each have their own private memories and controllers, each can fetch memory in about half the time of a Xeon or Athlon. In fact, it will take the Opteron system less time (140ns) to perform a cross court fetch than it takes a Xeon to make a simple fetch through the motherboard. In other words, the Opteron has a very low latency inter-processor connection. Where this really comes in handy is in running

SMP applications. True SMP problems that run faster in a low latency shared memory environment, like those on the heavy iron built by companies like Fujitsu, will run faster on an Opteron cluster than any other commodity cluster, excepting possibly the expensive HP Alpha EV79 based solution, which features a similar architecture. **To put things in perspective, the best latency between nodes in an MPI cluster is typically on the order of 10 microseconds. This is a factor of 100 greater than the inter-processor latency of an Opteron cluster. This means that an Opteron SMP cluster can execute parallel problems whose granularity is a factor of 100 “finer” than those executing on an MPI connected system.**

Summary: Emerging network technologies such as 10G Ethernet and InfiniBand make it possible to link servers at high speeds. However, these technologies can place a significant load on a server’s CPU and memory due to the improvement of connection speeds. Traditional architectures invoke multiple data copies at the kernel and application level when writing from one system to another. This tax is becoming so significant that in many applications the bottleneck has become the memory bandwidth and processing power of the system.

Remote Direct Memory Access (RDMA) is a network interface feature that lets one computer directly place information into the memory of another computer. The technology reduces latency by minimizing demands on bandwidth and processing overhead. RDMA is a component of InfiniBand and promises to provide quantum improvements in cluster performance especially for the high performance computing market.

RDMA on InfiniBand allows the application to bypass the kernel and issue commands to a network interface card without executing a kernel call. This eliminates steps in the write process and therefore provides an increase in performance. RDMA also includes safeguards to avoid data corruption in memory.

Clustered scientific computing applications that use MPI can see a dramatic performance improvement as a result of the low latency, low overhead and high throughput that interconnects supporting RDMA provide. Other early applications of RDMA are remote file server access via DAFS, and storage access for blade servers via SRP. RDMA is fast becoming an essential feature of high-speed clusters and server-area networks.

A full copy of this report by Ariel Copland, principal scientist at Topspin, appears at <http://www.nwfusion.com/news/tech/2003/0324tech.html>

Microway Spotlight

Microway Spotlight on Nina Nitroy, Technical Support Manager

Nina joined our technical support team in 1988 and was the senior support person for Microway’s line of x86 and i860 based FORTRAN, C++ and Fortran90 compilers in the late 1980s and 1990s. She holds a BS degree in Geology from Bloomsburg University. Before joining Microway she worked in Texas doing seismic data processing and mine modeling for oil exploration. Nina’s expertise ranges from scientific programming in FORTRAN, to evaluating engineering software packages, to managing data processing groups. Currently she integrates Linux O/S and application software on Microway’s Beowulf clusters, as well as provides benchmarking services to customers and oversees the Microway Technical Support Team.

What are today's challenges for cluster users?

The biggest challenge is the decision making that goes into building the cluster in the first place. Since Microway builds custom clusters, the software and hardware choices are highly varied. To get the best configuration to match the customer's needs requires some forward thinking. For instance, although configuring a diskless cluster may be the initial goal, it may actually be better to go with nodes that contain hard drives that can act stand alone if the cluster will be broken into smaller groupings or if some nodes will be used as stand alone workstations one day. Microway builds all sorts of clusters from high quantity 1U rack mounted clusters to nodes that are towers which will sit on shelves. It all depends on the immediate and eventual use of the cluster. Decisions on the hardware and software configuration can be even more difficult, if they involve the cooperation of several departments at the user's location.

What criteria need to be addressed when designing a Linux cluster?

Since Microway will configure the hardware and software to fit the customer's needs, it is a very good idea to think about exactly how the software will be used. Decisions such as how and if NFS will be utilized (which directories are to be exported), external network settings, whether or not network teaming or channel bonding is needed, and which compilers and message passing and/or batch scheduling software will be used should be explored during the quoting process. It is very important that the customer takes advantage of the expertise of Microway's technical support and integration departments which is available at the time of quoting and immediately thereafter to help with fine tuning the cluster. We ask in our cluster configuration questionnaire what the specific application of the cluster will be. We know that applications have certain requirements for specific versions of the kernel, operating system or libraries. We can head off dependency problems from the start if we have the bigger picture of the cluster usage. Microway can also quote and install the desired compilers so that the message passing software can be configured specifically for it.

Another consideration is how the operating system software will be supported on site and if consistency with existing computers is needed. If these questions are asked up front, Microway can configure the software such that it blends with the existing computers and the requirements of the organization that is responsible for maintaining the computers.

It is also a good idea to think about backing up a master node if there isn't a RAID involved. Sometimes the best solution is the simplest one: purchase an additional hard drive for the master computer that is an exact copy of the one that is installed.

What is your procedure for diagnosing a failed node or system?

The usual process of elimination is best. If a computer, out of the blue, will not produce video, but the power supply fans are on, note the following: are there error beeps on boot attempt? Open the unit and determine if all CPU fans are turning. Reseat all expansion slot boards (especially any riser cards) and push down on memory dimms. A reboot attempt could be made with one CPU at a time, or half the memory at a time. However, at any time, the unit may be shipped to Microway for diagnosis and repair.

If video is produced but the hard drive fails to boot, another known working hard drive can be swapped into the system. If a hard drive appears to have failed due to a hardware failure or a software corruption, Microway can replace the drive with the operating system installed, or

with a blank one for the purpose of cloning it (performing a simple bit by bit copy) from one of the other nodes.

An intermittently failing node requires more observations. First note whether the node tends to fail under heavy load or not. At this time it is a very good idea to remove the application from the equation and use memory, CPU and I/O intensive tests to pinpoint the problem. Copies of MEMTEST, and CPUBURN reside in the Technical Support area of the Microway website as well as on the boot loader and in /usr/local/sbin directories respectively (on a LINUX system). Other tests, “rundd” and “recomp” test I/O, CPU and memory simultaneously. Running these tests stresses the hardware and will determine once and for all its health without regard to application software behavior. It may be that the problem is, in fact, caused by a software or operating system issue instead.

Microway Technical Support is available to outline these basic procedures as well as to recommend specific and creative procedures for more obscure problems that may involve a customer’s application. Serial numbers of the computer tie the technical support person to the actual configuration so it is important to provide that information when contacting our team.

One final note, Microway installs notes on each specific configuration in a directory called Microway that resides in /root on a LINUX system. This directory contains all downloaded files, drivers, software, and readme files specific to the computer. Not only do we do custom work on the systems, but we document those things that are needed in order for customers to understand and maintain their systems. We consider it a priority to document those things that may deviate on a system or that could be considered ‘gotchas’ should a customer be interested in reproducing our work for any reason. This greatly reduces the chances of delays when diagnosing computers that are failing due to software changes or corruptions.

What do you like best about your job?

It is always satisfying not only to solve the immediate problem at hand, but also to provide creative solutions for the long term. I am in a position to get to know customers as opposed to just handle individual problem report tickets. Due to the nature of our products, there is a lot of opportunity to learn about technology as it unfolds and becomes available.

To receive HPC Times via email or to tell a friend about it visit: www.microway.com



ClusterWorld Conference & Expo – the first major event to focus entirely on clustered systems. If you work with clusters in any capacity, ClusterWorld Conference & Expo is the one event you cannot afford to miss this year. Learn more at <http://www.clusterworldexpo.com>.