Infiniband Based Cable Comparison

Makia Minich (minich@ornl.gov)
Oak Ridge National Laboratory
Oak Ridge, TN

June 19, 2007

Abstract

As Infiniband continues to be more broadly adopted in High Performance Computing (HPC) and datacenter applications, one major challenge still plagues implementation: cabling. With the transition to DDR (double data rate) from SDR (single datarate), currently available Infiniband implemntations such as standard CX4/IB4x style copper cables severely constrain system design (10m maximum length for DDR copper calbes, thermal management due to poor airflow, etc.). This paper will examine some of the options available and compare performance with the newly released Intel Connects Cables. In addition, we will take a glance at Intel's dual-core and quad-core systems to see if core counts have noticeable effect on expected IO patterns.

1 Overview and Goals

With the adoption of the Infiniband (IB) interconnect fabric, IB has increasingly proven to be a production worthy alternative to 10 Gigabit Ethernet (10GbE). However, because of power and signal integrity (data integrity) requirements there is a limitation on the maximum distance between end-points when using copper cabling. Making the problem even more challenging, as silicon technology improves, data rates increase (e.g SDR to DDR, and soon DDR to QDR) which causes the decrease in maximum cable distances. Table 1 shows the generally accepted lengths for SDR and DDR (as taken from the "Infiniband SDR, DDR, and QDR" white paper[1]). While short cable lengths might be acceptable in a small clusters, connecting larger cluster layouts, or even looking to utilize IB as a LAN solution for datacenter interconnectivity is far less feasible unless this length can be increased.

Table 1: Cable Lengths for SDR and DDR

Link Rate	Max Cable Length
1X - SDR	20m
4X - SDR	15m
12X - SDR	10m
4X - DDR	10m
12X - DDR	7m

For several years now, companies like Emcore have produced media converter modules that connect directly to the standard IB4x (CX4 style) IB connector and convert the electrical signals to and from optical signals over optical ribbon fiber terminated with MPO/MTP optical connectors. These modules significantly extend the maximum possible end-point separation but the economics become prohibitive for large node count systems due to module costs. Intel Connects Cables are a new entrant to the HPC and datacenter cabling market and offer the benefits of an active optical cable interconnect at a cost much closer to that of conventional copper cabling.

With this paper, we will evaluat Intel Connects Cables and compare it to the Emcore solution as well as traditional copper based cables. The comparison will focus primarily on the bandwidth, bidirectional bandwidth and latency performances of each cable looking while looking for any noticeable warning signs that might limit the use of any of these cables in HPC and datacenter applications.

1.1 System Layout

For the first part of this evaluation, four nodes from an x86_64 based linux cluster were utilized. Each node had a dual-socket single-core 3.4GHz Intel Xeon with an 8-lane PCI-Express based HCA card (the motherboard was an Intel SE7520JR23D). Two of the nodes will used Voltaire 4x SDR HCA's while the other two used Voltaire 4x DDR

[1] http://www.cisco.com/en/US/netsol/ns500/networking_solutions_white_paper0900aecd804c324e.shtml

HCA's. The nodes were connected back-to-back utilizing port 1 of each HCA as well as connection to a Flextronics F-X430046 DDR Infiniband switch. The connection options evaluated in this study are show in table 2.

Table 2: Connection Options Being Tested

Cable	**Length**
CX4 Copper	5m
Emcore QM3400, MPO Fiber	40m
Intel Connects Cables	1m 10m 100m

Once the baseline performance expected from these options was measured, two Intel S5000XAL0 Motherboards (with Woodcrest based logic) were tested. As a starting point, each node had dual-sockets which are fully populated with dual-core Intel CPUs, 4GB RAM, and a Voltaire PCI-express based SDR HCA. This test configuration was selected to allow the microprocessors to be replaced with new Intel quad-core processors.

1.2 Operating System Software

1.2.1 Operating System (OS)

All of the systems utilized in this evaluation ran Centos-4.0 in 64-bit mode with the `2.6.9-42.EL_lustre.1.4.7smp` kernel as supplied by Cluster File Systems[2]. This OS is provided to each of the nodes in a diskless fashion using Warewulf[3] to serve the filesystem images to each node in the cluster. This is a simple way to maintain all nodes at the same OS software levels while simplifying the overall system layout.

1.2.2 Infiniband Stack

The OpenFabrics Alliance[4] recently began distributing an enterprise version of their stack. Created through a collaboration between different Infiniband vendors and opensource community contributors, the OpenFabrics Enterprise Distribution (OFED) is touted as the stable and supported opensource Infiniband stack. While development is still ongoing for the main OpenFabrics software branch, the OFED stack takes snapshots in time, to create a supported product for the IB community. These releases supply an easy to build and install framework which allows users to start utilizing their Infiniband interconnect independent of vendor and OS stack loaded on the system. By unifying the Infiniband stack, it is easier to manage software revisions on multiple platforms as well as provide consistent API's for interconnect development on these platforms.

Testing is focusing on OFED 1.1 which contains support for the kernel involved in our testing. While normally we would build all of the tools associated with the Infiniband stack, the testing for this inquiry does not necessarily need things like MPI. More specifically, all we really need is remote-DMA access to pass large amounts of data across the interconnect. In addition, we will utilize the OpenSM available in the OFED 1.1 release to configure our fabric.

1.3 Test Suite

Provided as a default functionality test by the OpenFabrics Enterprise Distribution, `ib_send_bw` and `ib_send_lat` (low-level bandwidth and latency tests respectively) allow measurement of the total throughput that could be expected from the hardware (removing any constraints that the higher level IB protocols would impose).

When the utilities are run with the `-a` option, it performed with message sizes from 2 to 2^{23} bytes for bandwidth and 2 to 2^8 bytes for latency. An advantage of this tool is that it allows the user to specify the IB transport desired for the transfer: Reliable Connection (RC), Unreliable Connection (UC), and Unreliable Datagram (UD). While these three IBtransports should behave the same, in general, it is best to verify, as each transport has its own use; the Infiniband Trade Association website[5] can provide a good in-depth look at these transports.

1.4 Graph Naming Convention

All of the data in the graphs used in this paper attempt to follow the same naming schemes. For tests where there is no switch involved, the basic scheme is `<connection type>-<link speed>-<cable>`. For tests where a switch is involved, the scheme is `<connection type>-<cable on node A>-<cable on node B>-<link speed>`.

[2] http://www.lustre.org
[3] http://www.perceus.org/portal/project/warewulf
[4] http://www.openfabrics.org
[5] http://www.infinibandta.org/specs/

Any deviations from this system will be explained in that section. Table 3 describes the `<cable>` portion of the name.

Table 3: Cable Naming Description

Name	Description
cx4	8m copper cable with an IB4x (CX4-style) connector
emcore	40m MPO Fiber with EMCore Media Converter Module
100m	100m Intel Connects Cable
10m	10m Intel Connects Cable
1m	1m Intel Connects Cable

2 SDR Testing

The initial focus was on 4X SDR links. Currently, SDR is the most widely deployed IB technology as it has been around the longest. In the ORNL environment, SDR performance is very important because of system limitations (e.g. the Cray XT3 only has PCI-X slots leaving SDR as the only option). For this testing, the two nodes are connected back-to-back. OpenSM was setup to run on one of the two nodes to bring up the fabric.

The important feature to focus on in these graphs is how close the performance curves follow the copper (*-sdr-cx4) curve, as this is baseline performance expected from this connection.

2.1 Reliable Connection Testing

Figure 1 shows the results of the Reliable Connection (RC) testing. As can be easily seen, all of the configurations performed similarly, with a peak bandwidth of about 950MB/s.

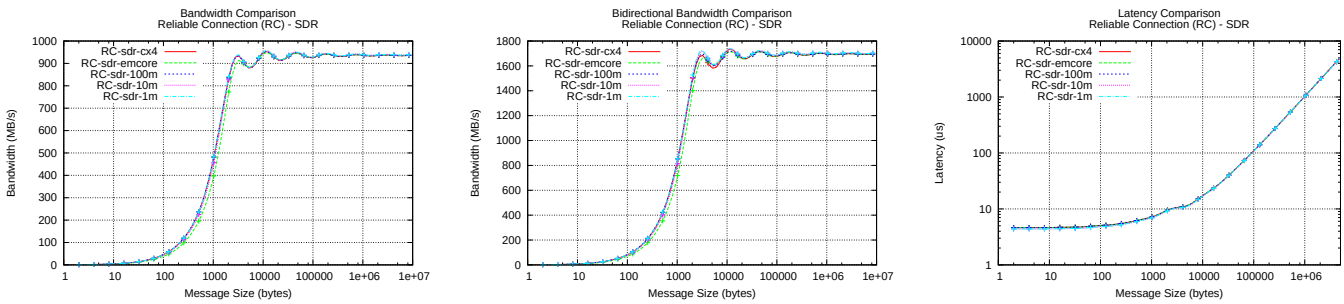

Figure 1: Centos-4.0 system with OFED-1.1, Voltaire SDR HCA connected back to back (no switch), dual-socket single-core 3.4GHz Xeon, 4GB DDR 400MHz Memory, Intel SE7520JR23D Motherboard.

2.2 Unreliable Connection Testing

Figure 2 shows the results for the Unreliable Connection (UC) testing. Again, there are no noticeable differences in performance and peak data rates reach around 950MB/s.

2.3 Unreliable Datagram Testing

Figure 3 shows the results for the Unreliable Datagram (UD) testing. Again, there is no noticeable deviations in performance among the interconnects.

3 DDR Testing

Since all of the SDR tests performed as expected, we wondered how the cables would perform at full DDR speeds. Cables were therefore tested using two nodes connected back-to-back. OpenSM was run on one of the two nodes to bring the link online.

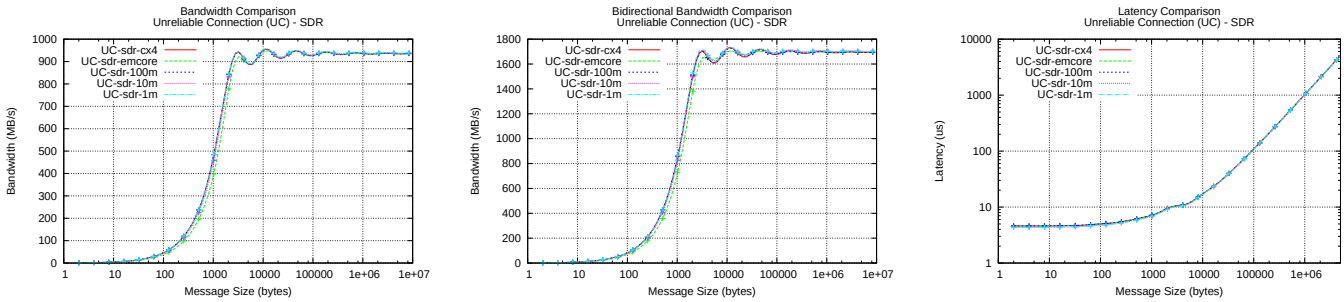

Figure 2: Centos-4.0 system with OFED-1.1, Voltaire SDR HCA connected back to back (no switch), dual-socket single-core 3.4GHz Xeon, 4GB DDR 400MHz Memory, Intel SE7520JR23D Motherboard.

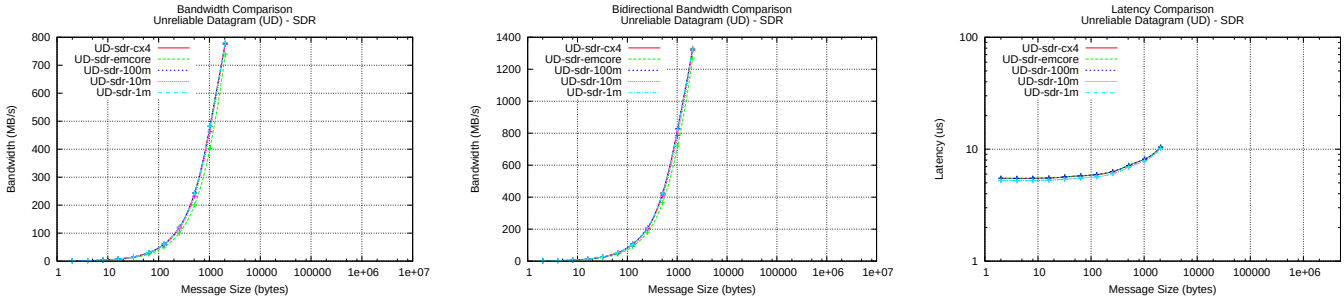

Figure 3: Centos-4.0 system with OFED-1.1, Voltaire SDR HCA connected back to back (no switch), dual-socket single-core 3.4GHz Xeon, 4GB DDR 400MHz Memory, Intel SE7520JR23D Motherboard.

3.1 Reliable Connection Testing

Figure 4 shows the results for Reliable Connectino (RC) on DDR. It is in this test that a deviance in performance is first observed. The Emcore module provided less bandwidth than the other options, but with a simple explanation. The Emcore media converter module is only rated for SDR speeds. (Emcore has now released a separate DDR module product which, at an additional cost, should bring performance back up to the rest of the group.) The Intel Connects Cables clearly deliver full DDR bandwidth as well as the low latency characteristic of the copper cabling.

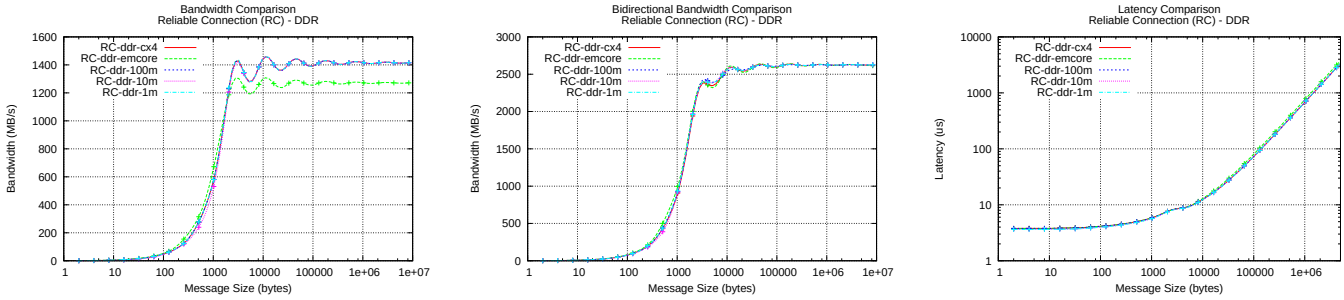

Figure 4: Centos-4.0 system with OFED-1.1, Voltaire DDR HCA connected back to back (no switch), dual-socket single-core 3.4GHz Xeon, 4GB DDR, Intel SE7520JR23D Motherboard.

3.2 Unreliable Connection Testing

Figure 5 clearly shows the performance drop for the Emcore module with the IB link operating in Unreliable Connection (UC) mode. Latency is roughly uniform across the group.

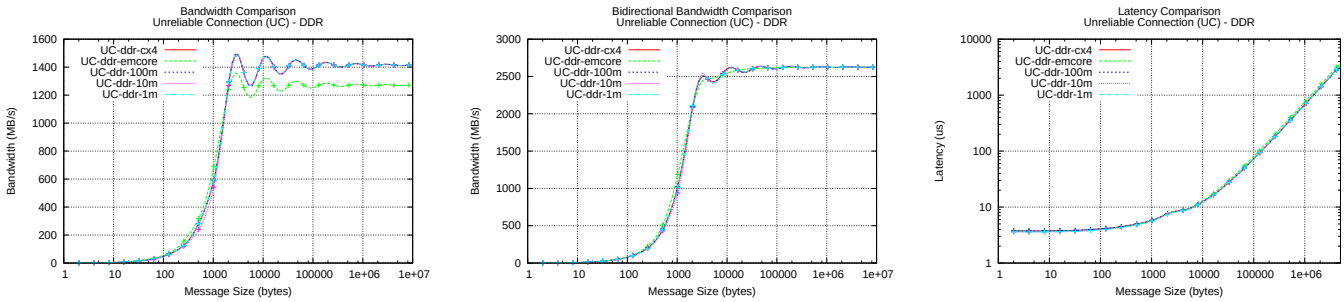

Figure 5: Centos-4.0 system with OFED-1.1, Voltaire DDR HCA connected back to back (no switch), dual-socket single-core 3.4GHz Xeon, 4GB DDR, Intel SE7520JR23D Motherboard.

3.3 Unreliable Datagram Testing

Figure 6 shows that because of packet size limitations with the Unreliable Datagram (UD) transport, the expected performance drop for the Emcore module is not observed.

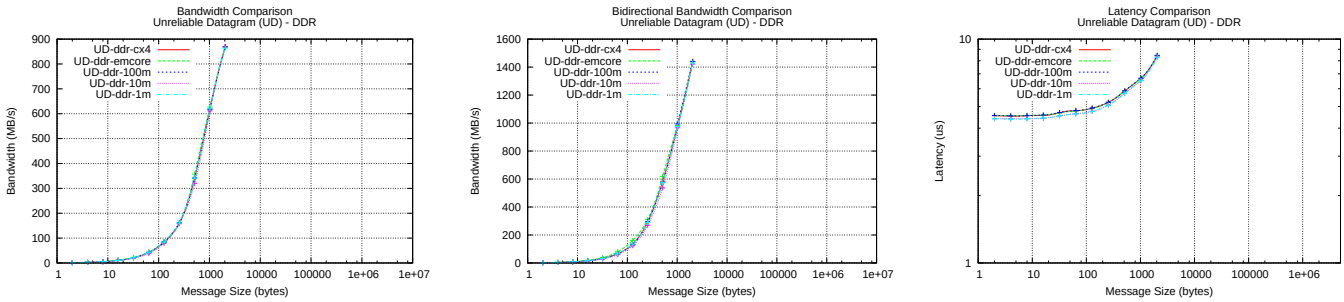

Figure 6: Centos-4.0 system with OFED-1.1, Voltaire DDR HCA connected back to back (no switch), dual-socket single-core 3.4GHz Xeon, 4GB DDR, Intel SE7520JR23D Motherboard.

4 DDR Switch Testing

Having completed the baseline DDR and SDR measurements (without an intermediate switch), performance testing was conducted utilizing a switch between the two endpoints. Since the Emcore media converter modules were limited to SDR speeds, were removed from the testing loop. Table 4 shows the options tested at DDR speed using the Flextronics F-X430046 Infiniband switch.

Table 4: Cable Layout

Legend	**Node A**	**Node B**
100m-100m	100m Intel Connects Cable	100m Intel Connects Cable
10m-10m	10m Intel Connects Cable	10m Intel Connects Cable
cx4-100m	CX4 (Copper)	100m Intel Connects Cable
cx4-10m	CX4 (Copper)	10m Intel Connects Cable
cx4-cx4	CX4 (Copper)	CX4 (Copper)

4.1 Reliable Connection Testing

As shown in figure 7 we immediately notice a potential issue. Utilizing the 100m cable (on either one or both HCA-to-switch connections) we see a significant bandwidth drop (of about 500MB/s). An explanation of this decrease will be presented in the next section of this paper.

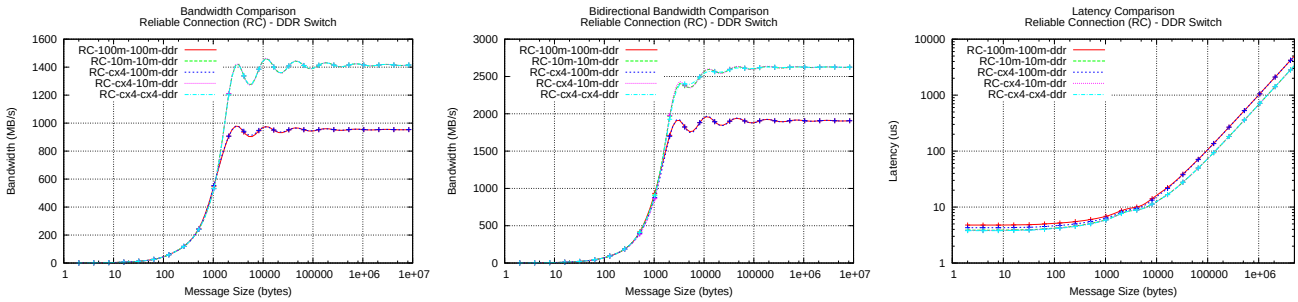

Figure 7: Centos-4.0 system with OFED-1.1, Voltaire DDR HCA connected to Fujitsu DDR Switch, dual-socket single-core 3.4GHz Xeon, 4GB DDR, Intel SE7520JR23D Motherboard.

4.2 Unreliable Connection Testing

Figure 8 shows the same results as the RC connection. The drop in performance seen previously is still observed.

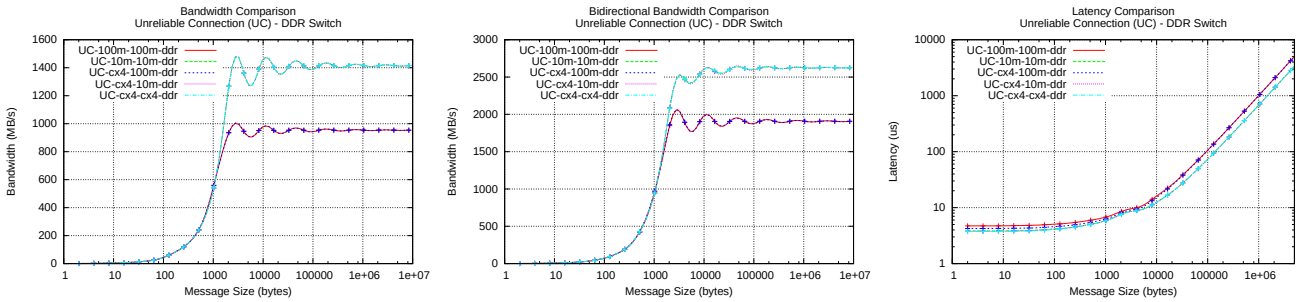

Figure 8: Centos-4.0 system with OFED-1.1, Voltaire DDR HCA connected to Fujitsu DDR Switch, dual-socket single-core 3.4GHz Xeon, 4GB DDR, Intel SE7520JR23D Motherboard.

4.3 Unreliable Datagram Testing

As with the DDR based Emcore testing for UD, figure 9 shows that the limitation of packet sizes in the UD protocol prohibits us from seeing the same performance drop as with RC and UC.

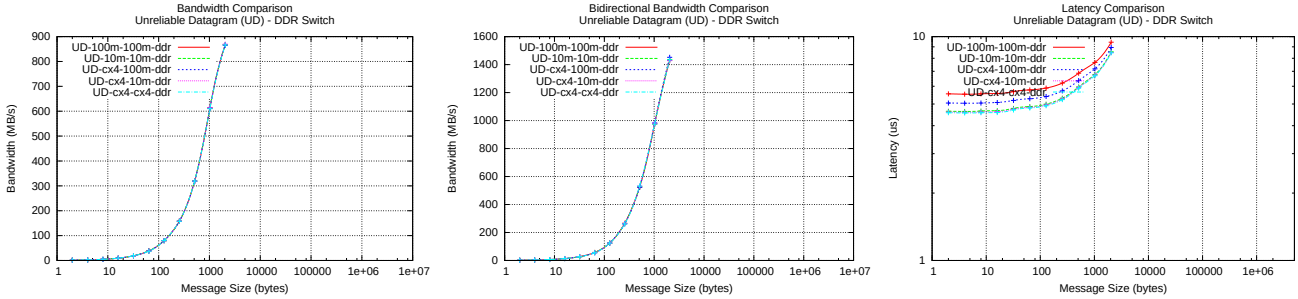

Figure 9: Centos-4.0 system with OFED-1.1, Voltaire DDR HCA connected to Fujitsu DDR Switch, dual-socket single-core 3.4GHz Xeon, 4GB DDR, Intel SE7520JR23D Motherboard.

5 Further 100m Investigation

The performance drop observed in the DDR Switch testing prompted further investigation as to origin of the bandwidth decrease in the 100m cable. After detailed discussion with Infiniband vendors, it was determined that the

problem lies in how much data can be kept in flight over such long links (due to buffering). Current HCA IB implementations utilize multiple virtual lanes, statically carving off resources (e.g. buffers) inside the HCA and reserving them for these separate virtual lanes. For example, if 8 virtual lanes are specified, each lane is allocated only 1/8th of the available resources. For the 100m length, the internal buffer was able to push all of its data onto the line and was forced to wait for the other end to receive the data before moving on to new data for transmission.

The solution is simply to intentionally reduce the number of virtual lanes. The Infiniband spec requires at least 2 virtual lanes (one virtual lane dedicated to managment data, with the second used for data). Using OpenSM, it was found that by default the link was set for 5 operational virtual lanes. To change this, we needed to do the following:

Listing 1: Changing Operational Virtual Lanes

```
opensm -c # dump the cache file
sed -i.bak s/max_op_vls 5/max_op_vls 1/ /var/cache/osm/opensm.conf
killall opensm
/etc/init.d/opensmd restart
```

This effectively reduces the virtual lanes to only one operational virtual lane for data.

In the following results, there is a slight deviation from the initial graphs. In this section, some of the data from the previous sections (100m-100m-ddr, 10m-10m-ddr, cx4-cx4-ddr) is shown and contrasted with new data resulting from the virtual lane change (cx4-100m-ddr-vl-change). This allows for graphical comparison of how the virtual lane change affected the performance of the 100m Intel Connects Cable.

5.1 Reliable Connection Testing

Figure 10 shows the result of our virtual lane change. As can be seen, this effectively moved our performance back up to the same speed as other cable options.

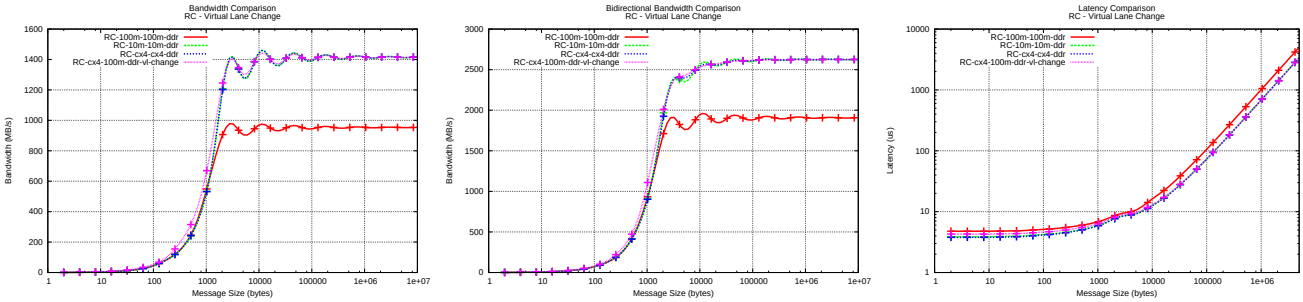

Figure 10: Centos-4.0 system with OFED-1.1, Voltaire DDR HCA connected to Fujitsu DDR Switch, dual-socket single-core 3.4GHz Xeon, 4GB DDR, Intel SE7520JR23D Motherboard.

5.2 Unreliable Connection Testing

Figure 11 shows that UC also shares the performance increase.

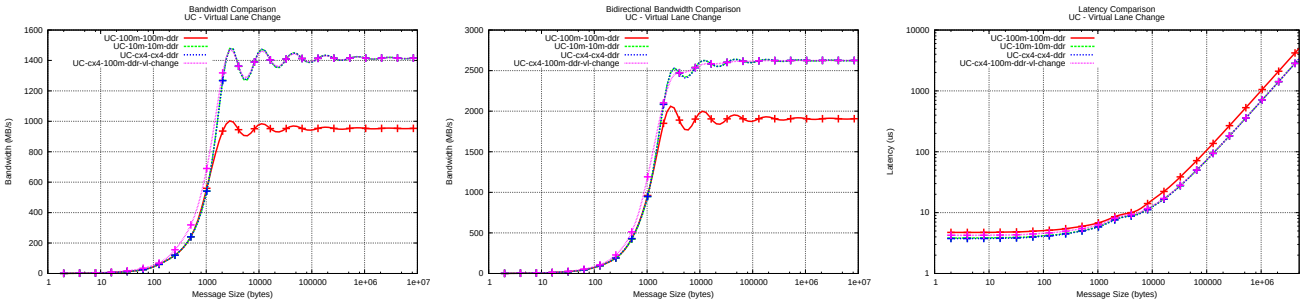

Figure 11: Centos-4.0 system with OFED-1.1, Voltaire DDR HCA connected to Fujitsu DDR Switch, dual-socket single-core 3.4GHz Xeon, 4GB DDR, Intel SE7520JR23D Motherboard.

5.3 Unreliable Datagram Testing

While UD wasn't affected (due to packet size limitations), figure 12 shows that this is still performing as expected.

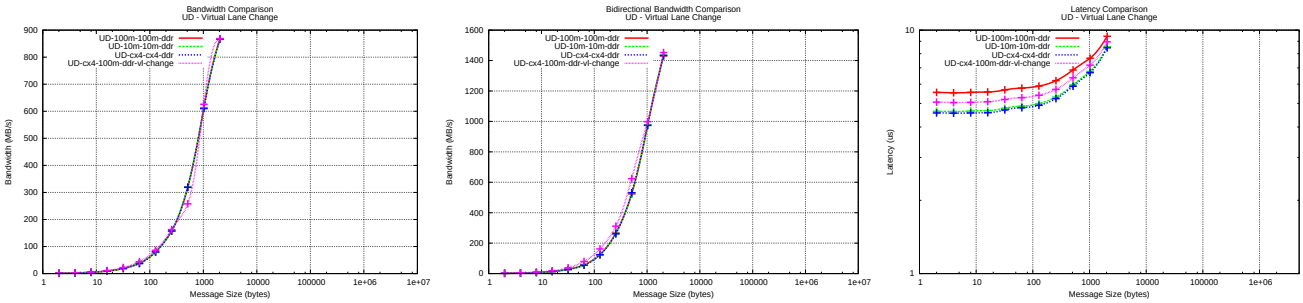

Figure 12: Centos-4.0 system with OFED-1.1, Voltaire DDR HCA connected to Fujitsu DDR Switch, dual-socket single-core 3.4GHz Xeon, 4GB DDR, Intel SE7520JR23D Motherboard.

6 Dual Core vs. Quad Core

To acheive higher and higher FLOP counts for our HPC applications, multi-core architechture has become increasingly mainstream in the commodity cluster space. Since it has been demonstrated Intel Connects Cables are suitable for use in widely available (and tested) hardware such as dual-socket, single-core Xeon based systems, we wondered if the addition of more CPU cores would adversely impact performance.

Utilizing the Woodcrest based nodes mentioned in section 1.1, we were able to evalutate dual and quad core configurations for their effect on the IO performance of the IB interconnect. In this section, we focused primarily on the 100m Intel Connects Cable (as this has proven to emphasize the most problems throughout all of the tests).

For the graphs in this section, the naming scheme is as follows: `<connection type>-<number of cores>-<cable>`. In this case the `number of cores` will either be `dual-core` or `quad-core`. Anything after the `cable` portion is a description of how the system was changed for that particular test.

6.1 Initial Testing

First attempts for dual and quad core testing revealed a number of problems that seemed to be related specifically to these Woodcrest nodes. Namely, even with a standard copper based IB cable, figure 13 shows that we measured only about 850MB/s (as compared to the expected 950MB/s). This was true for all of the cable options, but was even more pronounced when the 100m Intel Connects Cables were tested, dropping bandwidth even further to 760MB/s. Based on the previous testing using the 100m Intel Connects Cable as mentioned in section 5, this was easy to fix. The remaining question, however, was what happened to the other missing 100MB/s?

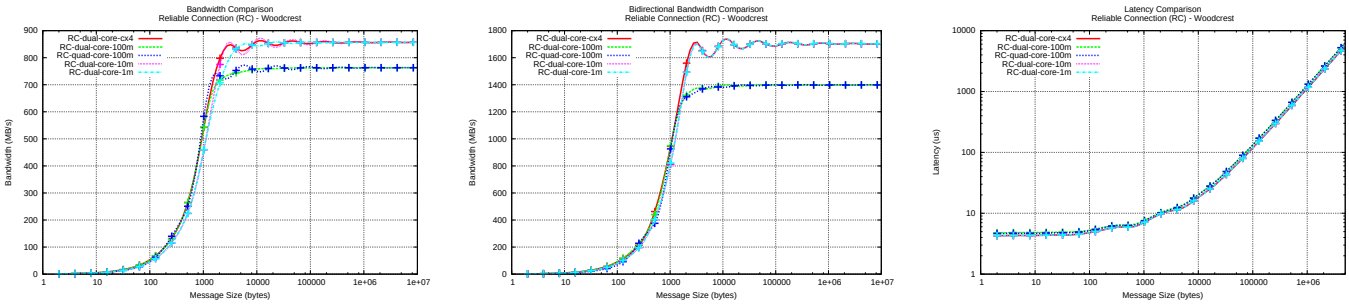

Figure 13: Centos-4.0 system with OFED-1.1, Voltaire SDR HCA connected back to back (no switch), dual-socket dual-core 2.66Hz Xeon or quad-core 2.00GHz Xeon, 4GB DDR 667MHz Memory, Intel S5000XAL0 Motherboard.

6.2 Woodcrest Fix

It was recalled that the `ib_mthca` kernel module has a few options available to it (on loading), and it was discovered that modifying the `tune_pci` option is essential for full bandwidth performance. By setting this option to one, the driver disregards the BIOS setting on the PCI bus, and increases the PCI burst. Figure 14 shows the before and after of test results (specifically with the CX4-style copper cables and 100m Intel Connects Cable).

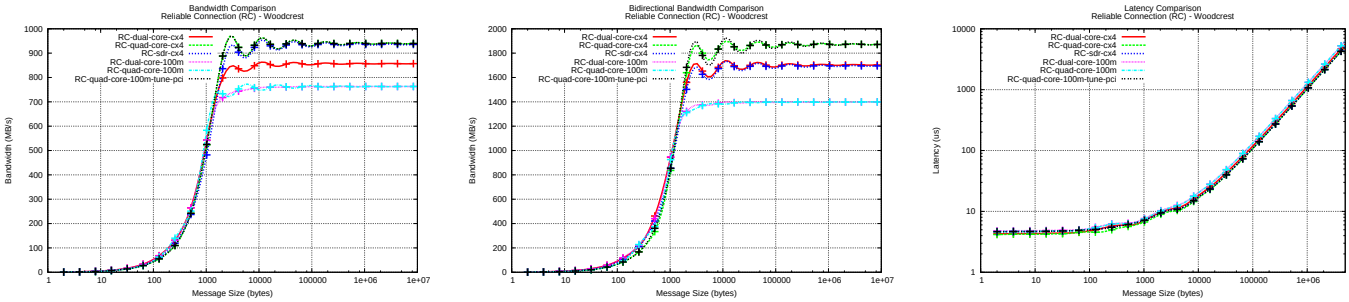

Figure 14: Centos-4.0 system with OFED-1.1, Voltaire SDR HCA connected back to back (no switch), dual-socket dual-core 2.66Hz Xeon or quad-core 2.00GHz Xeon, 4GB DDR 667MHz Memory, Intel S5000XAL0 Motherboard.

On these graphs, the result of the `tune_pci` tests are compared to the SDR results from the previous sections. Not only does this show that these Woodcrest nodes are able to drive the 100m Intel Connects Cables, but it also demonstrates that even with increased core counts, the IO subsystem does not suffer.

7 Conclusions

One of the biggest problems in deploying Infiniband into large scale HPC and data center environments is the short maximum distance and bulk/weight issues imposed by copper cabling. These limitations make the practical engineering of large scale clusters and data centers extremely challenging. The results from this evaluation show that the Intel Connects Cable is a viable option in significantly expanding past this constraint. With lengths up to 100m, we are able to span an Infiniband network across our data center and create useable, logically arranged machine layouts.

Another important piece to note is that the performance achieved by these cables was done *without* modification to any software; they were basically plug and play. The only change that was needed was in using the 100m length (there is perhaps a sweet spot in cable lengths between 10m and 100m, but for this testing there were no other options) and the decrease in virtual lanes. Due to the lack of current applications that actually take advantage of multiple virtual lanes, this change is not expected to affect the overall system capability. We hope that in future HCA's, the static resource binding will be changed into something more dynamic to allow us to fully exploit both compute and bandwidth resources.

Overall, Intel Connects Cables appear to allow Infiniband to cost effectively scale clusters and/or data centers beyond the 10m length restrictions of today's copper cabling. We no longer have to design our network around these boundaries, instead ORNL can finally lay out systems logically within the data center. As an example, the Intel Connects Cables allow our machine floor design at Oak Ridge National Laboratory (which spans two floors, one physically above the other) to logically place the different hardware (storage, vizualization clusters, supercomputers, backup systems) spanning both machine floors while still utilizing IB as the storage network interconnecting these separate "pieces". Instead of having to utilize more costly alternatives (e.g. 10GbE, or the Emcore Smartlink modules), we can use the Intel Connects Cables as simply as if we were just installing standard copper Infiniband cables.

8 Acknowledgments

The author would like to thank Gene Crossley and Gilad Shainer of Mellanox and Shikhar Parjan of Intel (and all of the others in the background) for help diagnosing the 100m performance drop.