MVAPICH with XRC: Early Experiences

Presentation at Open Fabrics Developers Conference (Nov. '07)

by

Dhabaleswar K. (DK) Panda
Department of Computer Science and Engg.
The Ohio State University
E-mail: panda@cse.ohio-state.edu
http://www.cse.ohio-state.edu/~panda

XRC Overview

- Without XRC support
  - Resources are required based on the number of communicating peers, regardless of location
- With XRC support
  - Resource requirements based on the number of hosts with communicating peers

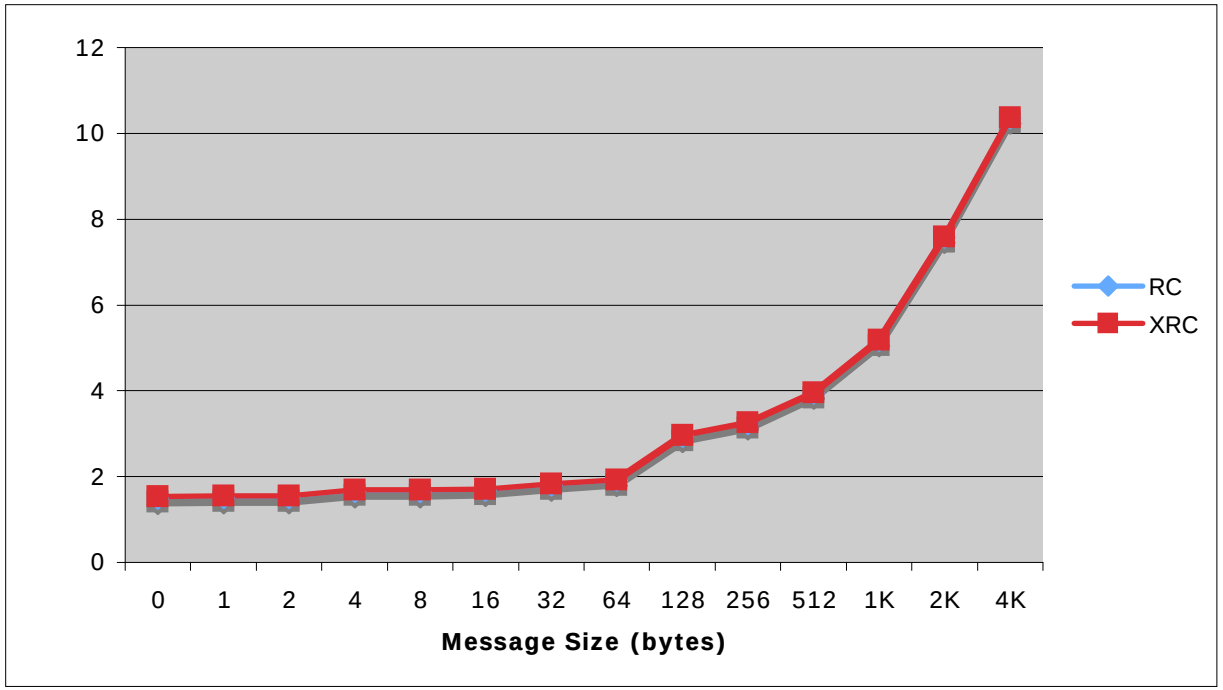
MVAPICH Changes

- A prototype implementation has been added to MVAPICH
- Basic Changes
  - Communication information is no longer strictly on a per peer-process basis
  - QPs as well as resource counts (and queues) are based on a node basis
  - Information on process->node location is already used by the shared memory

System Configuration

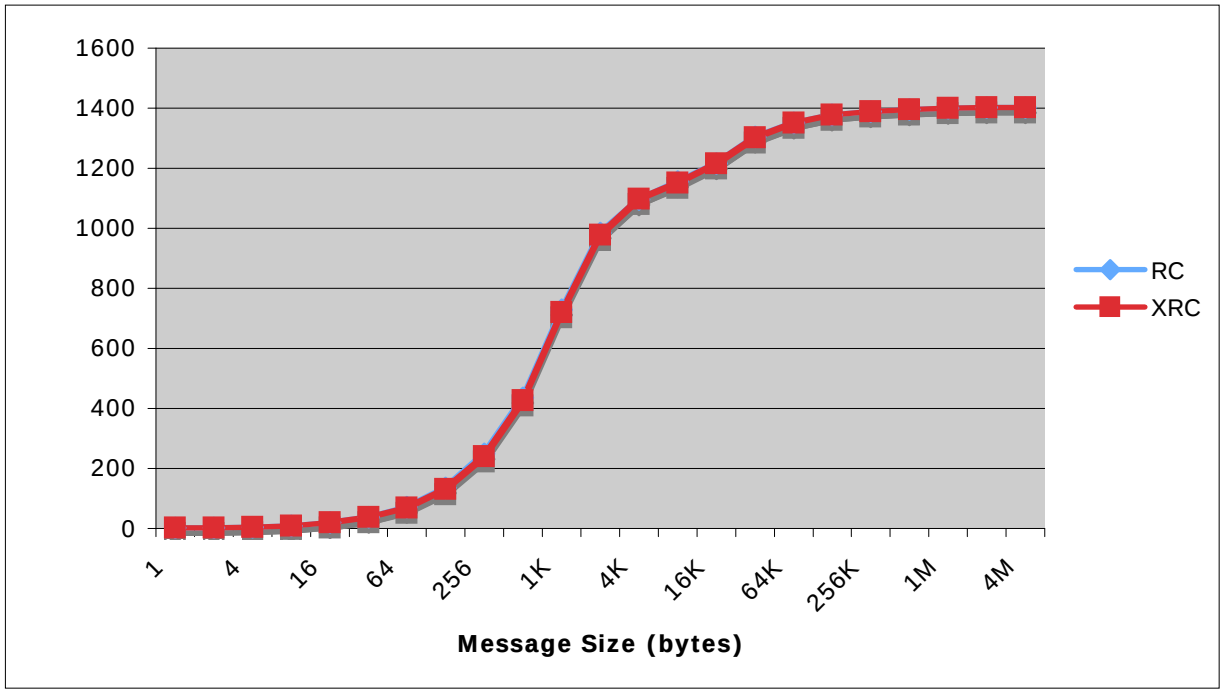
- Intel Clovertown systems
  - dual socket, 8 total cores
- OFED-1.3-20071109-0600
- ConnectX HCA
  - Firmware 2.2.258
- 24-port Mellanox MTS 24000 switch

Latency



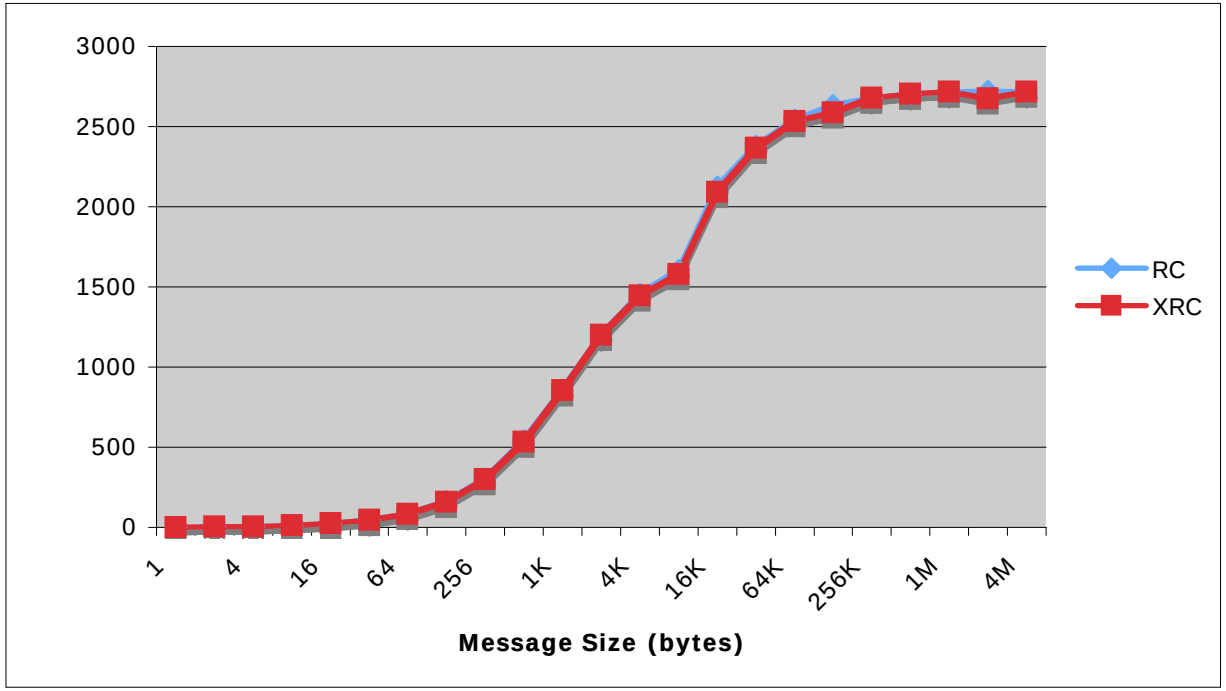
- Results for latency are nearly identical between the use of RC and XRC transports
- 1.49usec for RC, 1.54usec for XRC

Uni-Directional Bandwidth



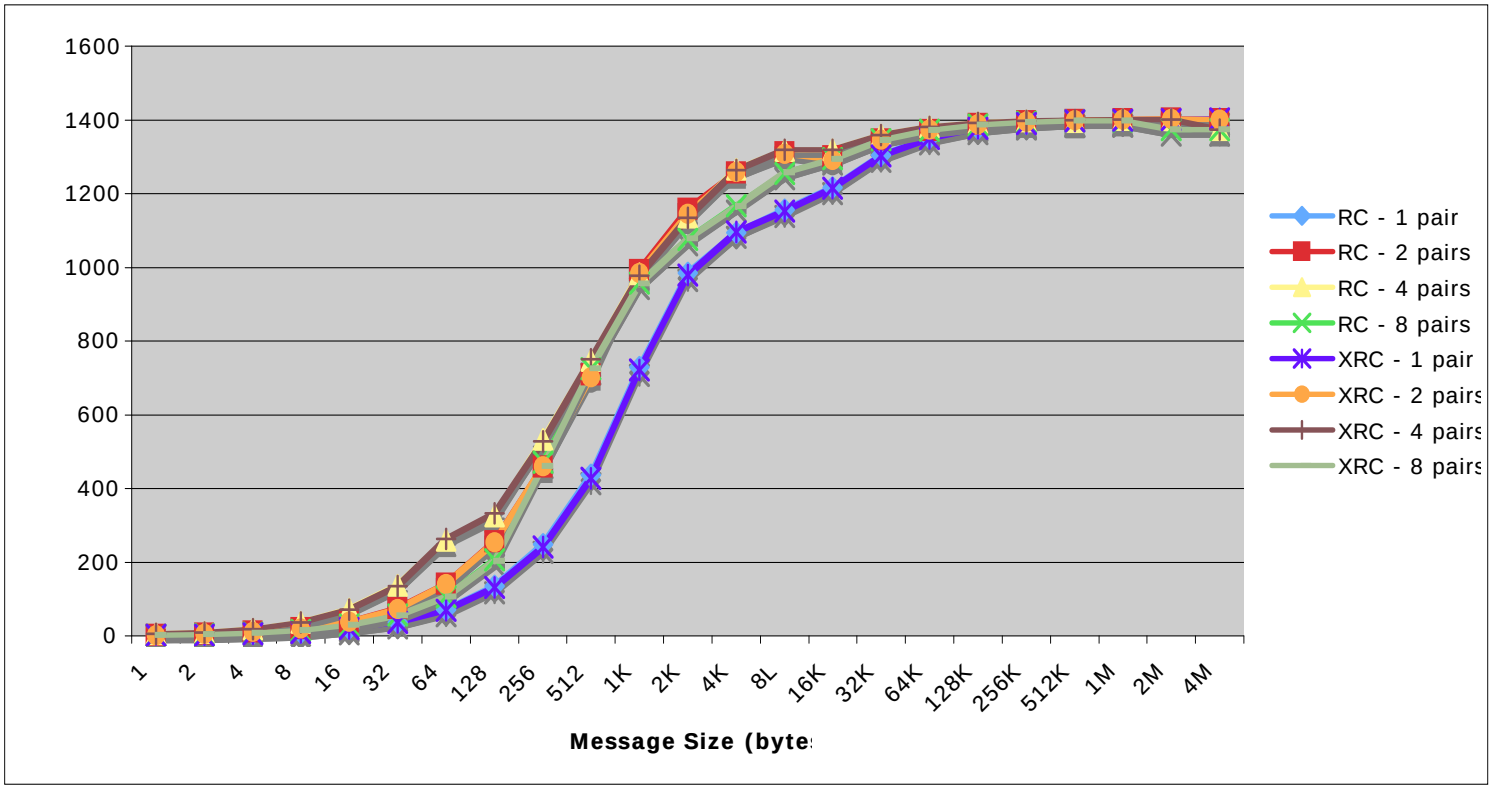
- Uni-Directional Bandwidth is identical in performance
- 1402 MB/sec maximum bandwidth on this platform

Bi-Directional Bandwidth



- Bi-Directional Bandwidth is also identical in performance
- 2722 MB/sec maximum bi-directional bandwidth on this platform

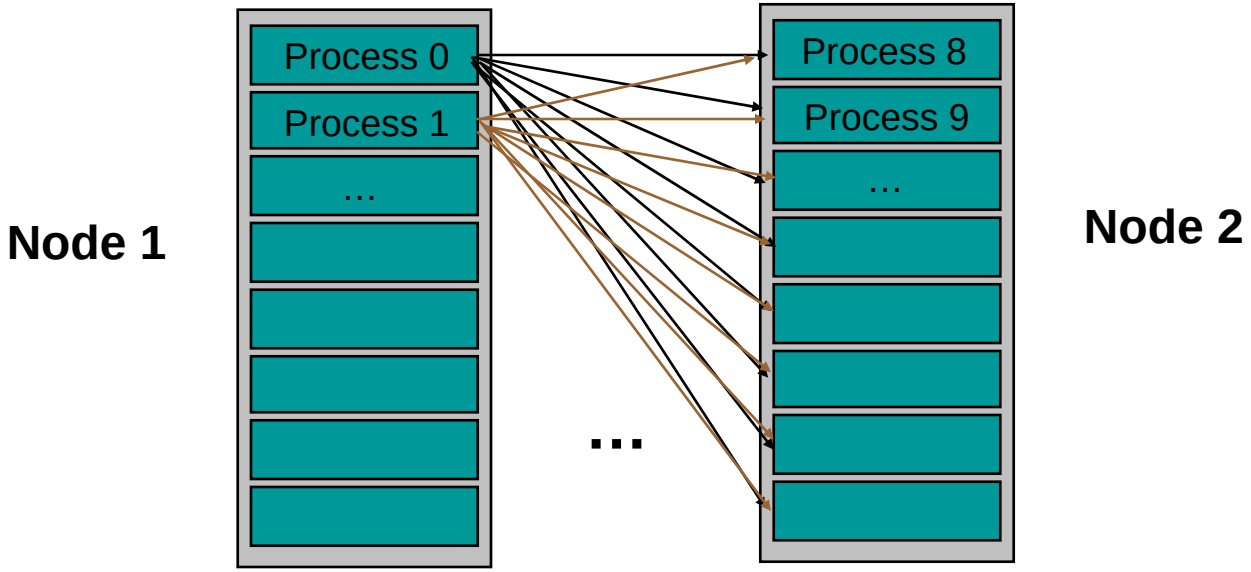
Multi-Pair Bandwidth



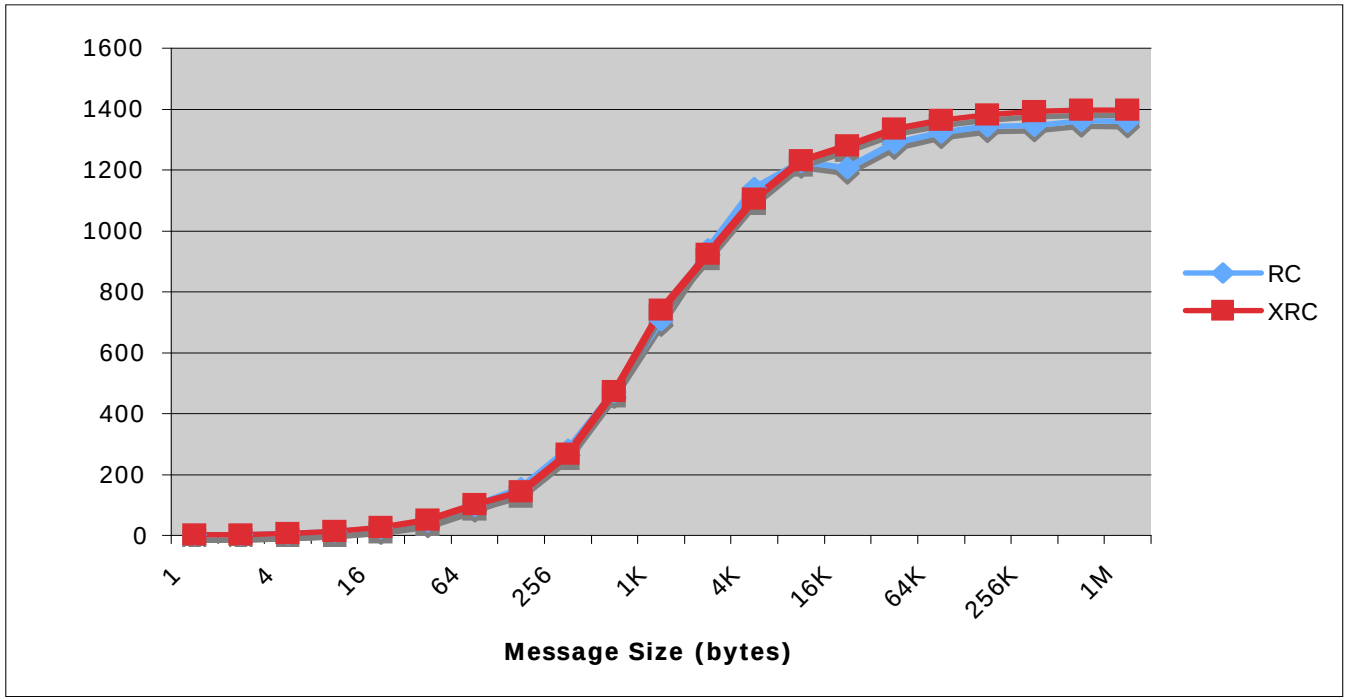
- Trends between RC and XRC remain the same

Many-to-Many Multi-Pair Bandwidth

- The earlier evaluations did not stress the XRC mechanism since each process is only communicating with a single other process on the other node
- Instead, each peer should communicate with every other process on the other node to stress the XRC mechanism



Many-to-Many Multi-Pair Bandwidth



- Here we see a slight difference between XRC and RC -- with XRC having a slightly higher bandwidth for large messages.
- A marginal (but very consistent) difference of 50 MB/sec

Conclusions

- XRC for MPI seems to perform equivalently to RC
  - Design will be incorporated into MVAPICH in the next release
- Larger-scale evaluation with application workloads is planned
- Plan to explore additional SRQs for different message sizes
  - Now that QPs are not required for different SRQs, this can be done efficiently
- Compare performance with UD-based design