Performance Evaluation of Open MPI on Cray XE/XK Systems

Samuel K. Gutierrez – LANL

Nathan T. Hjelm – LANL

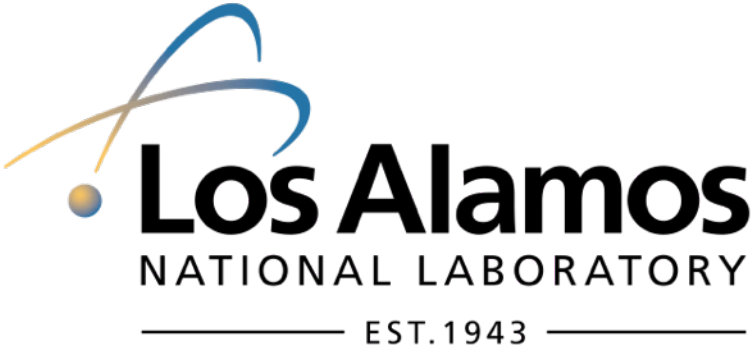
Manjunath Gorentla Venkata – ORNL

Richard L. Graham – ORNL

Hot Interconnects 2012

Aug 23, 2012

A Collaborative Effort



Outline

1. Open MPI Overview
2. Gemini Overview
3. Protocols Overview
4. Test Environment
5. Results
6. Conclusions/Future Work

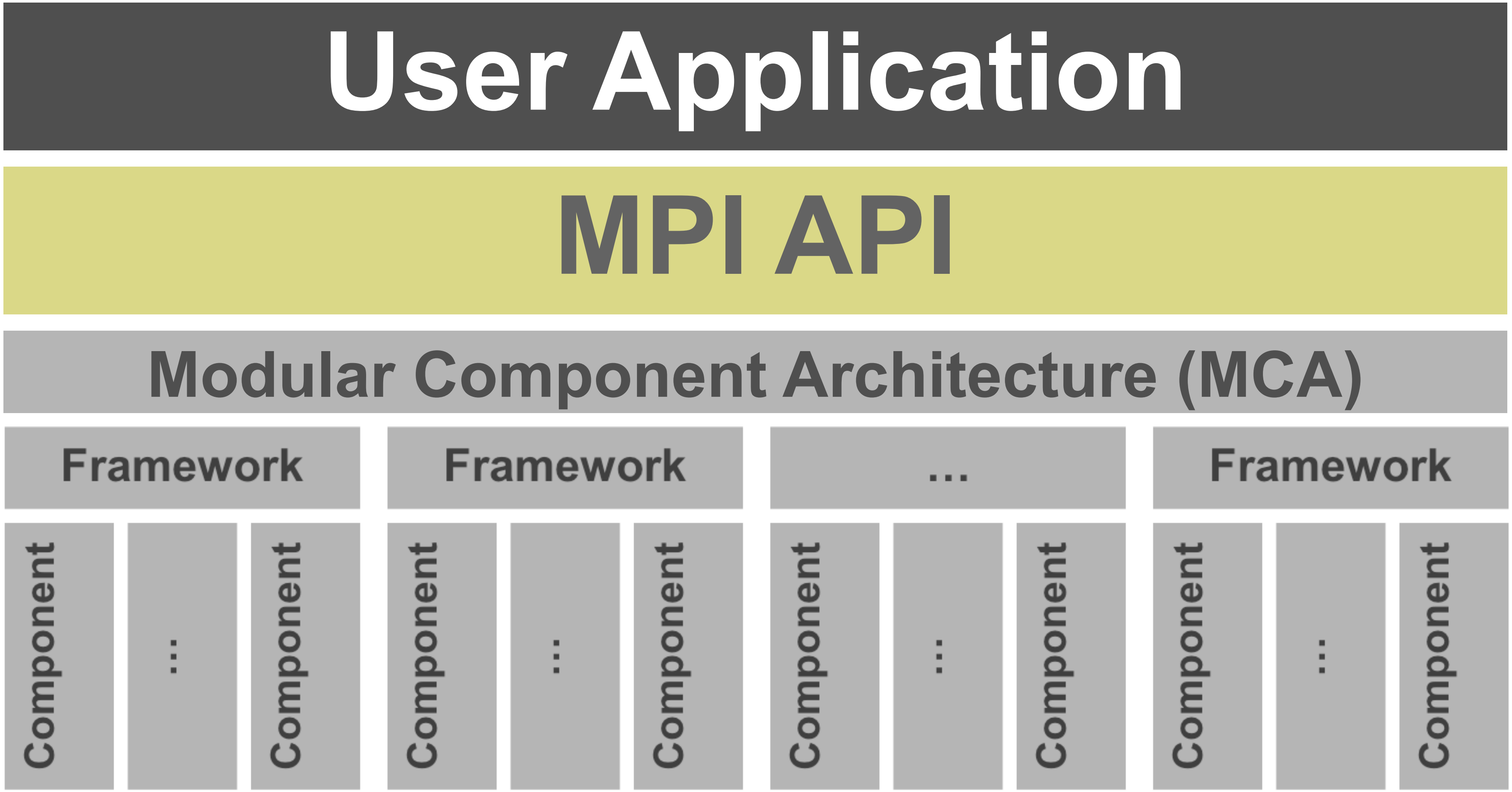
First Things First – Open MPI Overview

- **Open-Source Implementation of the MPI-2 Standard**

- **Developed and Maintained By**
  - Academia
  - Industry
  - National Laboratories

- **Supports a Range of High-Performance Network APIs**
  - Verbs (Infiniband, RoCE, iWarp)
  - PSM (QLogic/Intel HCAs)
  - MXM (Mellanox HCAs)
  - Portals (Cray SeaStar, Infiniband)
  - uGNI (Cray Gemini, Cray Ares)

Open MPI’s Plugin Architecture – A High-level Overview[1]



User Application

MPI API

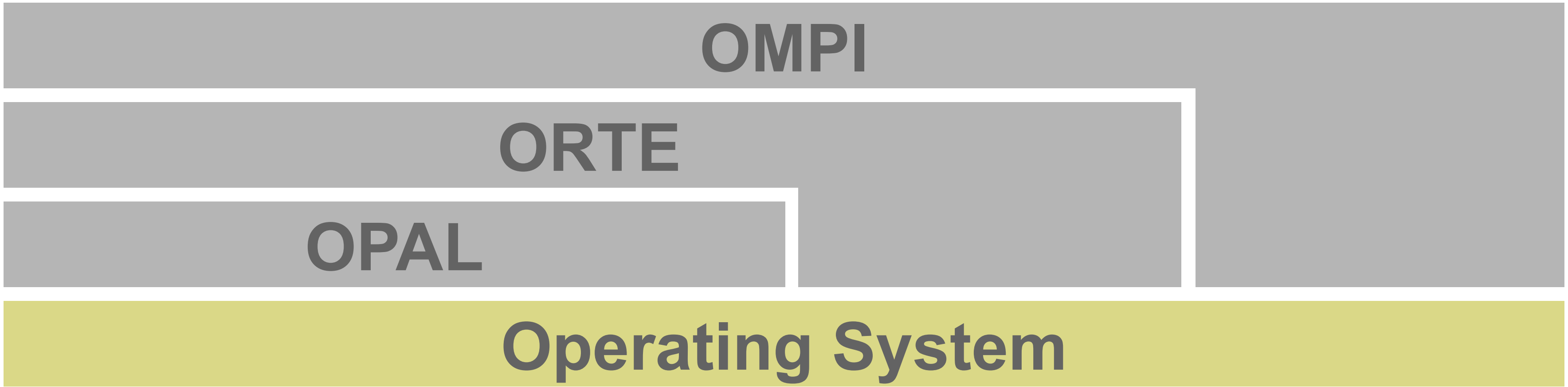
Modular Component Architecture (MCA)

Framework | Framework | … | Framework

Component … Component | Component … Component | Component … Component | Component … Component

Open MPI's Plugin Architecture – Main Code Sections[1]

- **Open MPI Layer (OMPI)**
  - MPI API and Support Logic
- **Open Run-Time Environment (ORTE)**
  - Run-Time System
- **Open Portability Access Layer (OPAL)**
  - OS-Specific/Utility Code



OMPI

ORTE

OPAL

Operating System

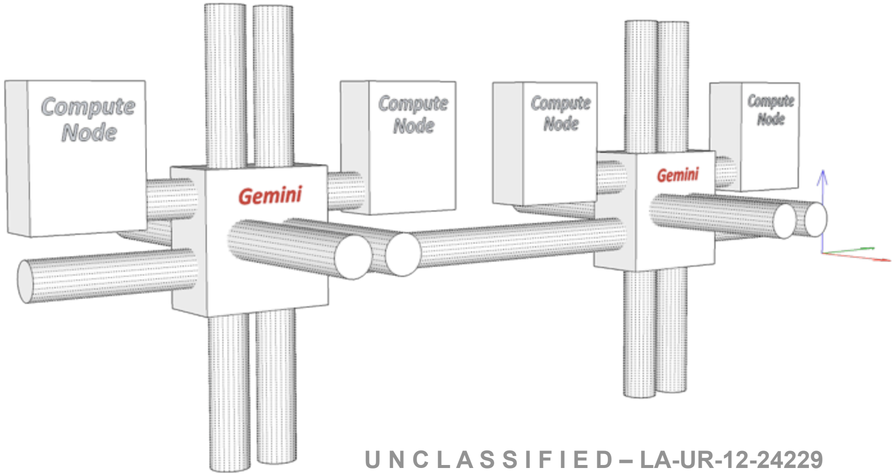
The Gemini System Interconnect[3] – An Overview

- **Network Used by the Cray XE and XK System Families**
  - Titan, Cielo, Hopper

- **Successor to the Cray SeaStar* Network Interconnect**

- **3D Torus Network Built of Gemini ASICs**

- **Gemini ASIC**
  - Provides 10 Torus Connections – 2 x (+X, -X, +Z, -Z) – 1 x (+Y, -Y)
  - Provides 2 NICs and a 48-port Router



OB1 PML High-Level Protocol Overview

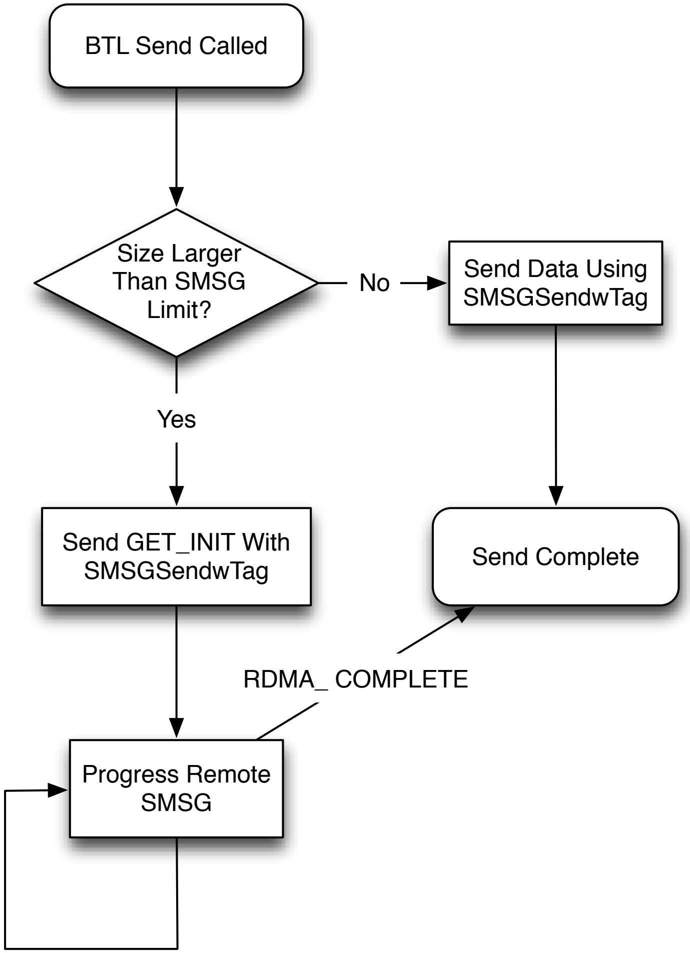
- **Eager Message Protocol**
  - Uses BTL buffered, inline, and in-place send protocols
- **Remote Get Protocol**
  - 2 Protocol Messages: RGET (ready to send + segment), FIN
  - Available When Registration Cache is Enabled and BTL Implements Get
- **RDMA Pipeline Protocol (Put)**
  - 3 Protocol Messages: RNDV + segment, RDMA, FIN
  - Used When Remote Get protocol is not Available
- **Remote Get Fallback (New)**
  - Essentially a Rendezvous
  - Fallback Initiated by the Receiver During Remote Get Protocol if BTL Get Protocol is not Available
- **Rendezvous (no RDMA)**

uGNI BTL Overview

- **Protocols**
  - Send
    - In-place Send for Small Messages Directly Using Small Message Protocol (SMSG)
    - Buffered Send Using Get for Larger Eager Messages (Eager Get)
  - Get
    - Uses FMA Or BTE
    - Available Only if Source And Destination Segments Are 4-Byte aligned and a Multiple of 4-Bytes in Size
  - Put
    - Uses Fast Memory Access (FMA) or Byte Transport Engine (BTE)
    - No Alignment Restrictions

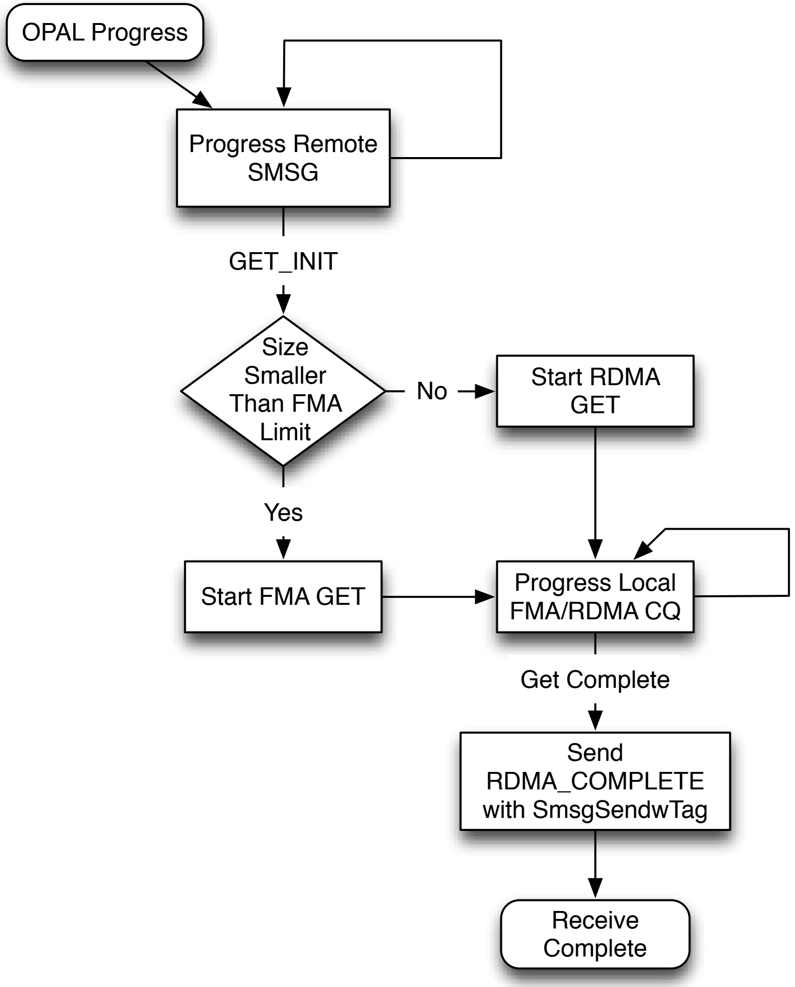
- **Lazy Connection Establishment**
  - Resource Utilization Directly Related to Application Communication Characteristics

uGNI BTL Eager Get Protocol Details (Send)


BTL Send Called
Size Larger Than SMSG Limit?
No
Send Data Using SMSGSendwTag
Yes
Send GET_INIT With SMSGSendwTag
Send Complete
RDMA_ COMPLETE
Progress Remote SMSG

Thursday, August 23, 12

uGNI BTL Eager Get Protocol Details (Receive)



UNCLASSIFIED – LA-UR-12-24229

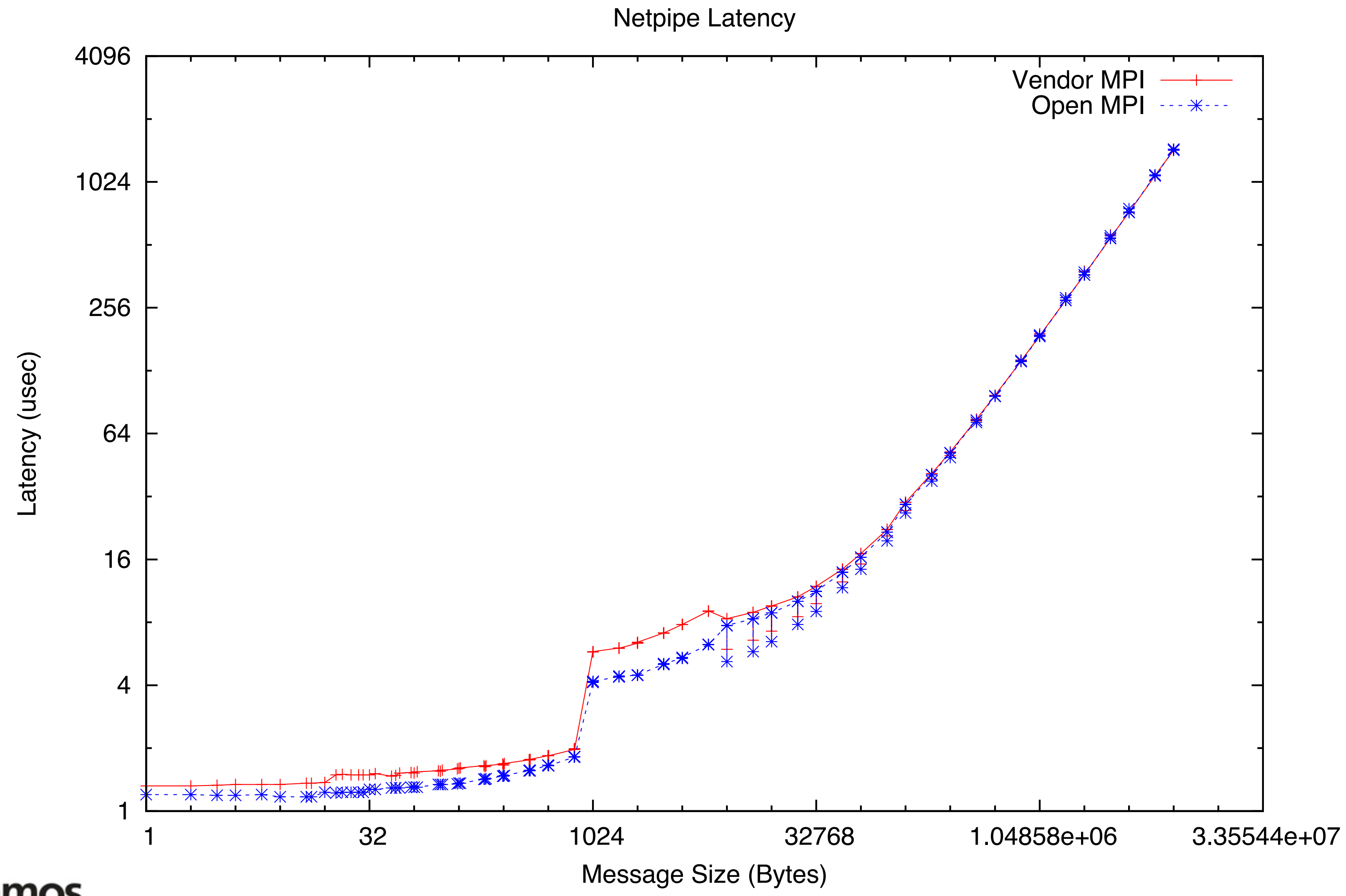
Vader BTL Overview

- **MPICH Nemesis-like Design**
  - Lock-Free Message Queues
  - "Fast Boxes" – I.e. Per-Peer Receive Queues for Short Messages
- **Copy Backend Changes Based on Message Size**
  - E.g. ***bcopy*** *[a,b)* - ***memcpy*** Otherwise
  - User Tunable with *Good* Defaults
- **Cross-Process Memory Mapping Allows for RDMA-Like Semantics**
  - Copy-In/Copy-Out (CICO) Avoided
  - No Backing Store Required
  - Heavy Use of Registration Cache to Amortize Attach Latency
  - Exposes Both Put and Get Interfaces to PML Layer
- **XPMEM Support Requires Kernel Patch and User-Level Library**
  - Already Available and Leveraged by Cray's Native MPI Implementation

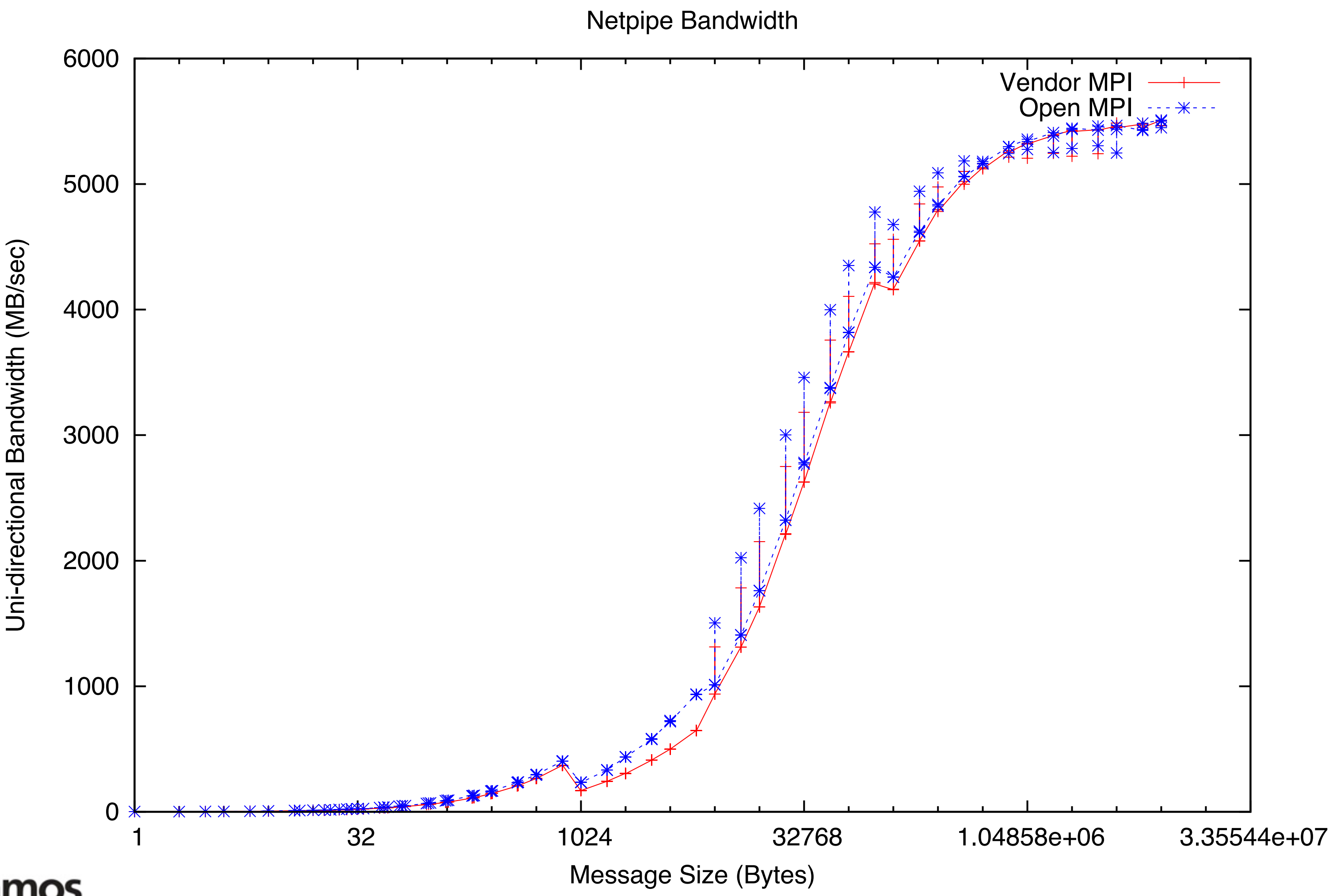
Test Environment

- **Testing Platforms**
  - **Cielito** - 1088 Core XE6
  - **Cielo** - 142,304 Core XE6
- **Microbenchmarks**
  - **NetPIPE –** Measure Lat/BW Benchmark
  - **AMG2006 –** Algebraic Multi-grid Solver
  - **LAMMPS –** Classical Molecular Dynamics Code
  - All Microbenchmarks Were Run on Live Production System
- **Launcher**
  - orterun

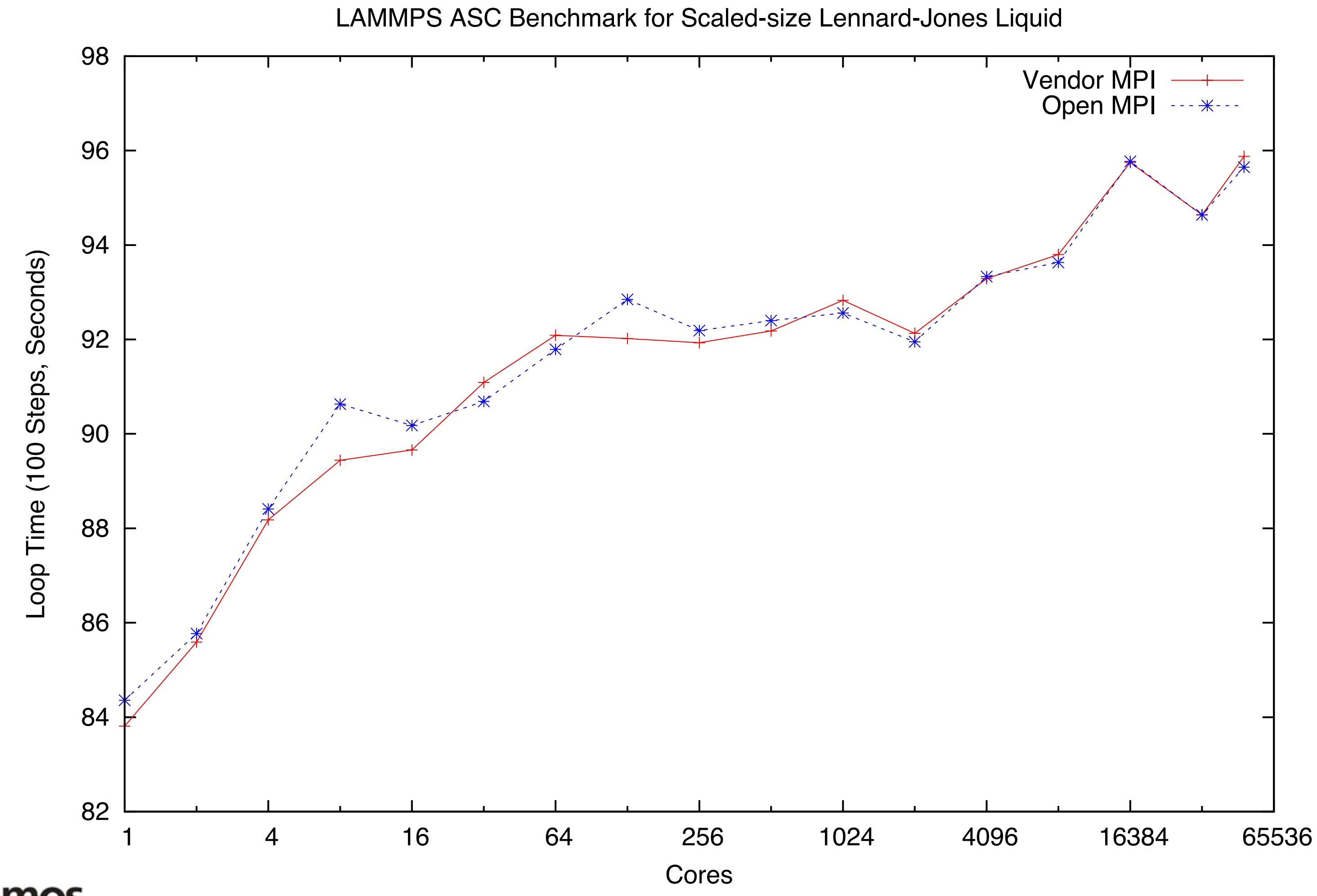
NetPIPE Latency on XE6 (on ASIC)



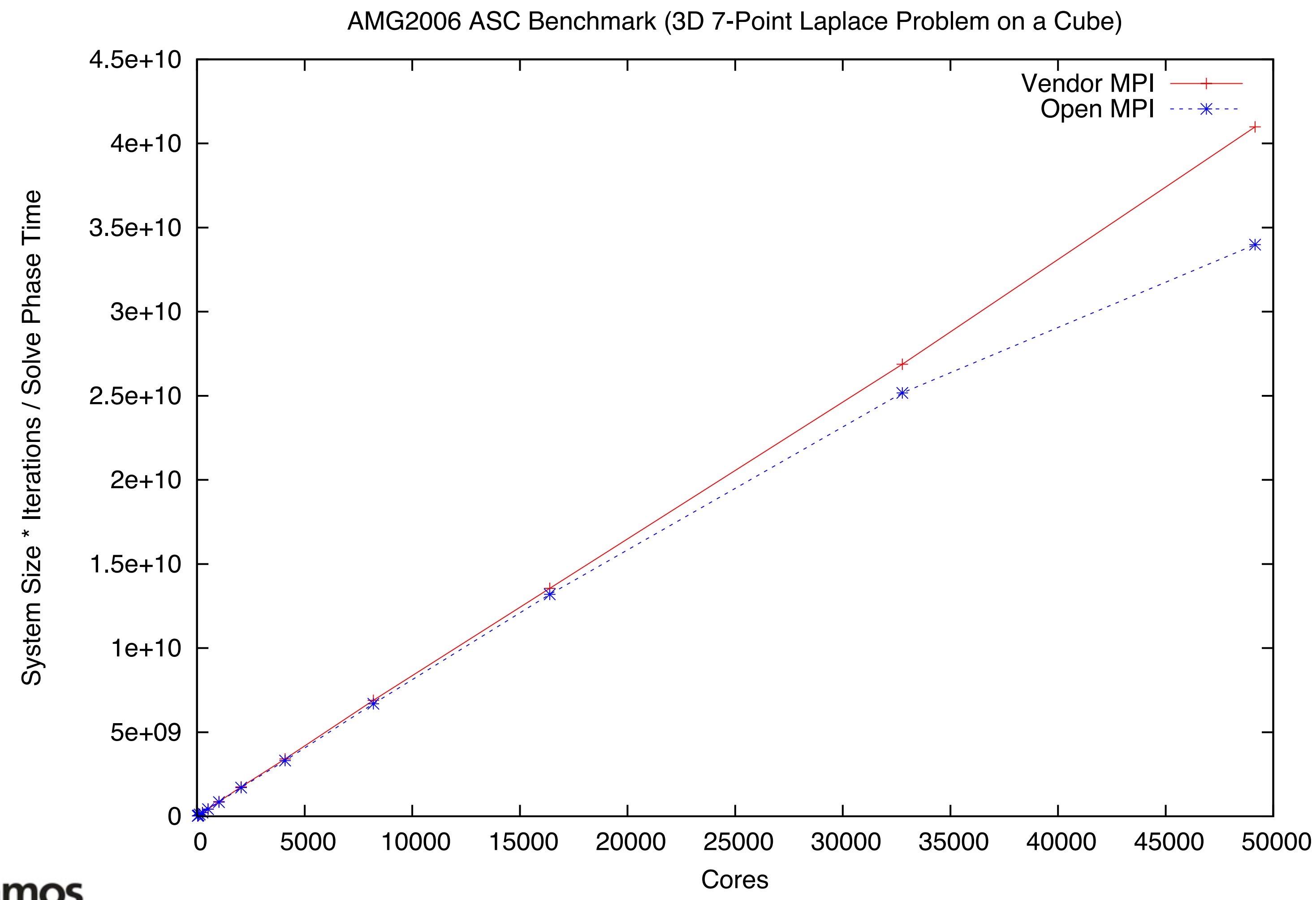
NetPIPE Bandwidth on XE6 (on ASIC)



Microbenchmark – LAMMPS



Microbenchmark – AMG2006



UNCLASSIFIED – LA-UR-12-24229

Conclusion and Ongoing/Future Work

- **Conclusion**
  - Bandwidth, Latency, and Scalability Similar to Vendor MPI Implementation
- **Stabilization/Optimization**
  - Improve Launch Scalability (Over a Minute to Launch 131072 MPI Tasks)
  - Investigating New Protocols (Shared Message Queue-- MSGQ)
  - Reduce Memory Requirements
- **Improved Collective Performance Using uGNI Atomics**
- **Work with Friendly Testers**
- **Prepare for General Release in Open MPI 1.7.0**

Thanks!

Questions?

- Questions?
- Comments?

References

- [1] Open MPI. 13 Feb. 2012 <open-mpi.org>.
- [2] R. Alverson, et al., “The Gemini System Interconnect,” in High Performance Interconnects (HOTI), 2010 IEEE 18th Annual Symposium on, Aug. 2010, pp. 83 –87.