



A Comparison of Three (or Four) MPI Implementations for Red Storm

Ron Brightwell

Sandia National Laboratories

Scalable Computing Systems Department

12th European PVM/MPI Conference

September 21, 2005



Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AC04-94AL85000.





Outline

- **Cray Red Storm / XT3**
- **MPI implementations**
- **Benchmarks**
- **Way too many graphs**
- **Conclusions**
- **Future work**





Red Storm

- True massively parallel processing machine
- Designed to be a single system
- Distributed memory MIMD parallel supercomputer
- Fully connected 3D mesh interconnect
- 10,368 compute nodes
- ~30 TB of DDR memory
- Red/Black switching: ~1/4, ~1/2, ~1/4
 - Only difference between Red Storm and XT3
- 8 Service and I/O cabinets on each end
 - 256 processors for each color

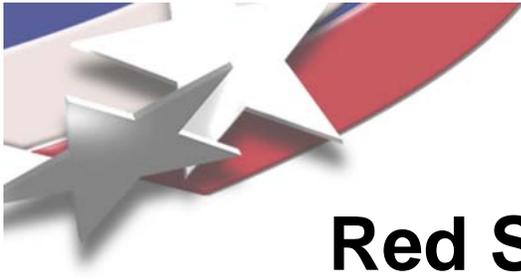




Red Storm System Software

- **Operating Systems**
 - LINUX on service and I/O nodes
 - Catamount lightweight kernel on compute nodes
 - LINUX on RAS nodes
- **Run-Time System**
 - Logarithmic loader
 - Intelligent node allocator
 - Batch system (PBS)
 - Libraries – MPI, I/O, Math
- **File Systems**
 - Lustre for both UFS and Parallel FS





Red Storm Processors and Memory

- **Processors**
 - AMD Opteron (Sledgehammer)
 - 2.0 GHz
 - 64 KB L1 instruction and data caches on chip
 - 1 MB L2 shared (Data and Instruction) cache on chip
 - Integrated dual DDR memory controllers @ 333 MHz
 - Integrated 3 Hyper Transport Interfaces @ 3.2 GB/s each direction
- **Node memory system**
 - Page miss latency to local processor memory is ~80 ns
 - Peak memory bandwidth of ~5.3 GB/s for each processor





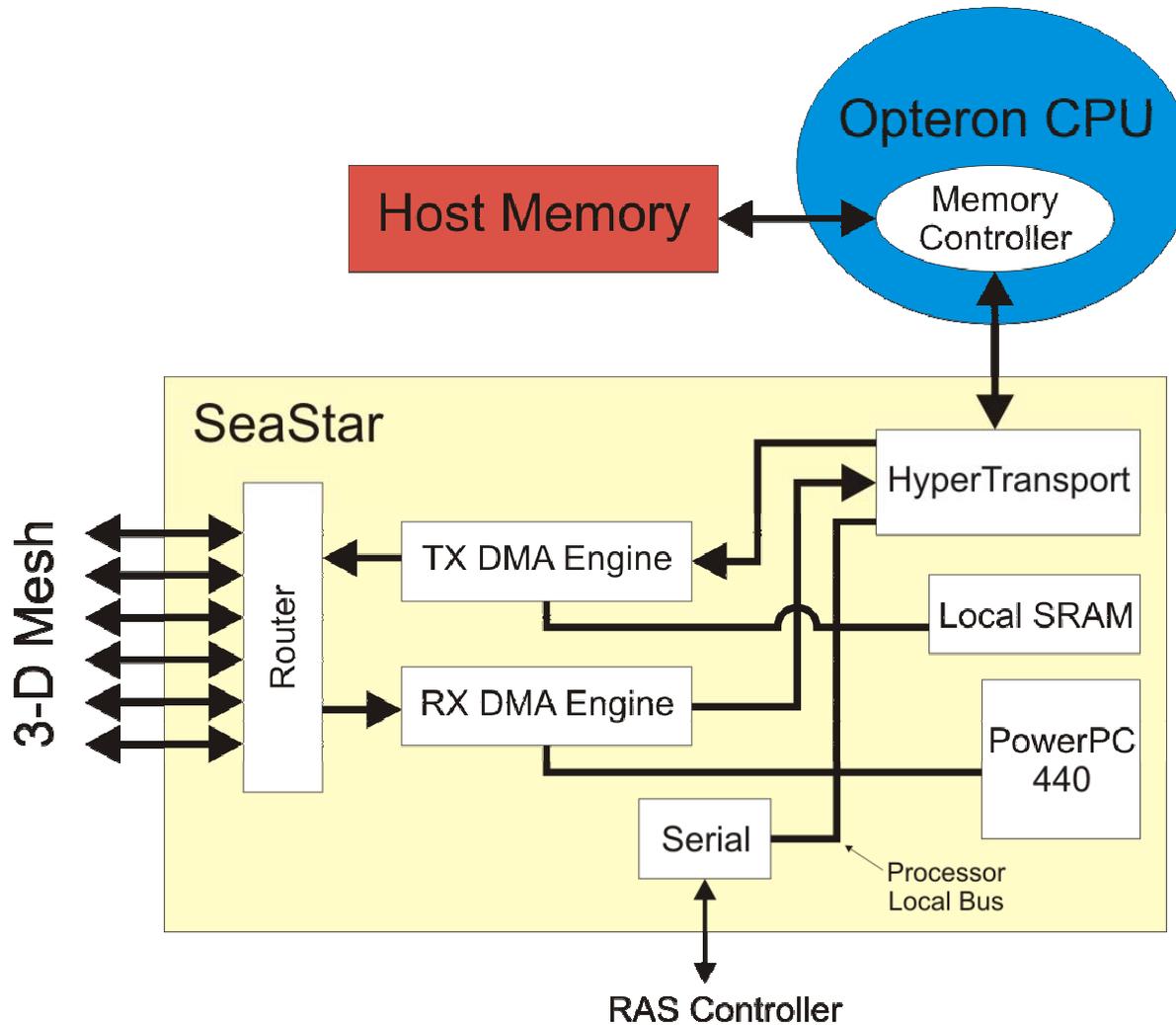
Cray SeaStar NIC/Router

- **16 1.6 Gb/s HyperTransport to Opteron**
- **500 MHz embedded PowerPC 440**
- **384 KB on-board scratch RAM**
- **Seven-port router**
- **Six 12-channel 3.2 Gb/s high-speed serial links**





SeaStar Block Diagram





SeaStar Embedded PowerPC

- **Message preparation**
- **Message demultiplexing**
 - MPI matching
 - Native IP packets
- **End-to-end reliability protocol**
 - Technically NIC-to-NIC protocol
- **System monitoring**





SeaStar Programming Interface

- **Cray chose Portals 3.3 API developed by Sandia and the University of New Mexico**
- **Portals was designed to support intelligent/programmable network interfaces**





Portals 3.3 for SeaStar

- **Cray started with Sandia reference implementation**
- **Needed single version of NIC firmware that supports all combinations of**
 - **User-level and kernel-level API**
 - **NIC-space and kernel-space library**
- **Cray added bridge layer to reference implementation to allow NAL to interface multiple API NALs and multiple library NALs**
 - **qkbridge for Catamount applications**
 - **ukbridge for Linux user-level applications**
 - **kbridge for Linux kernel-level applications**





SeaStar NAL

- **Portals processing in kernel-space**
 - Interrupt-driven
 - “generic” mode
- **Portals processing in NIC-space**
 - No interrupts
 - “accelerated” mode





MPI Implementations

- **MPICH 1.2.6 with Portals 3.3 device**
 - Originally developed for Cplant Linux clusters
- **MPICH 1.2.6 using Cray SHMEM-style device**
 - Presented last year at EuroPVM/MPI
- **MPICH2 0.97 with Portals 3.3 device**
 - Cray supported version for XT3
- **OpenMPI with OB1 PML and Portals 3.3 BTL**
 - Nightly snapshot from two weeks ago
 - Brand-spanking-new, first-attempt-at-getting-it-to-work, not-ready-for-prime-time, completely unoptimized, just-happy-to-have-it-working...





MPI Implementation Details

- **MPICH 1.2.6**
 - Uses a copy block for very short messages
 - Avoids waiting for wire-level acknowledgment
 - Avoid overhead of creating a memory descriptor
- **MPICH 1.2.6 and MPICH2**
 - Posted receive queue processing is done by Portals
- **MPICH 1.2.6/SHMEM and OMPI**
 - Posted receive queue processing is done by MPI

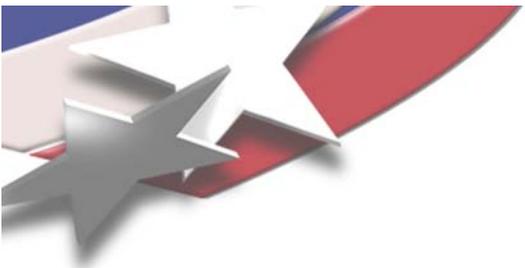




Micro-Benchmarks

- **ptlperf**
 - Ping-pong latency and bandwidth (uni- and bi-directional)
 - Single, persistent ME, MD, EQ
 - Best-case performance for Portals
- **mpil latency**
 - Standard best-case ping-pong latency and bandwidth
- **NetPIPE 3.6.2**
 - Ping-pong latency and bandwidth (uni- and bi-directional)
 - Streaming bandwidth
 - Implemented a Portals module





Disclaimer

- **Results are from a developer snapshot of a Sandia code base from last week**
- **This software may or may not make it to other Cray XT3 systems**
- **Accelerated implementation has undergone minimal tuning**



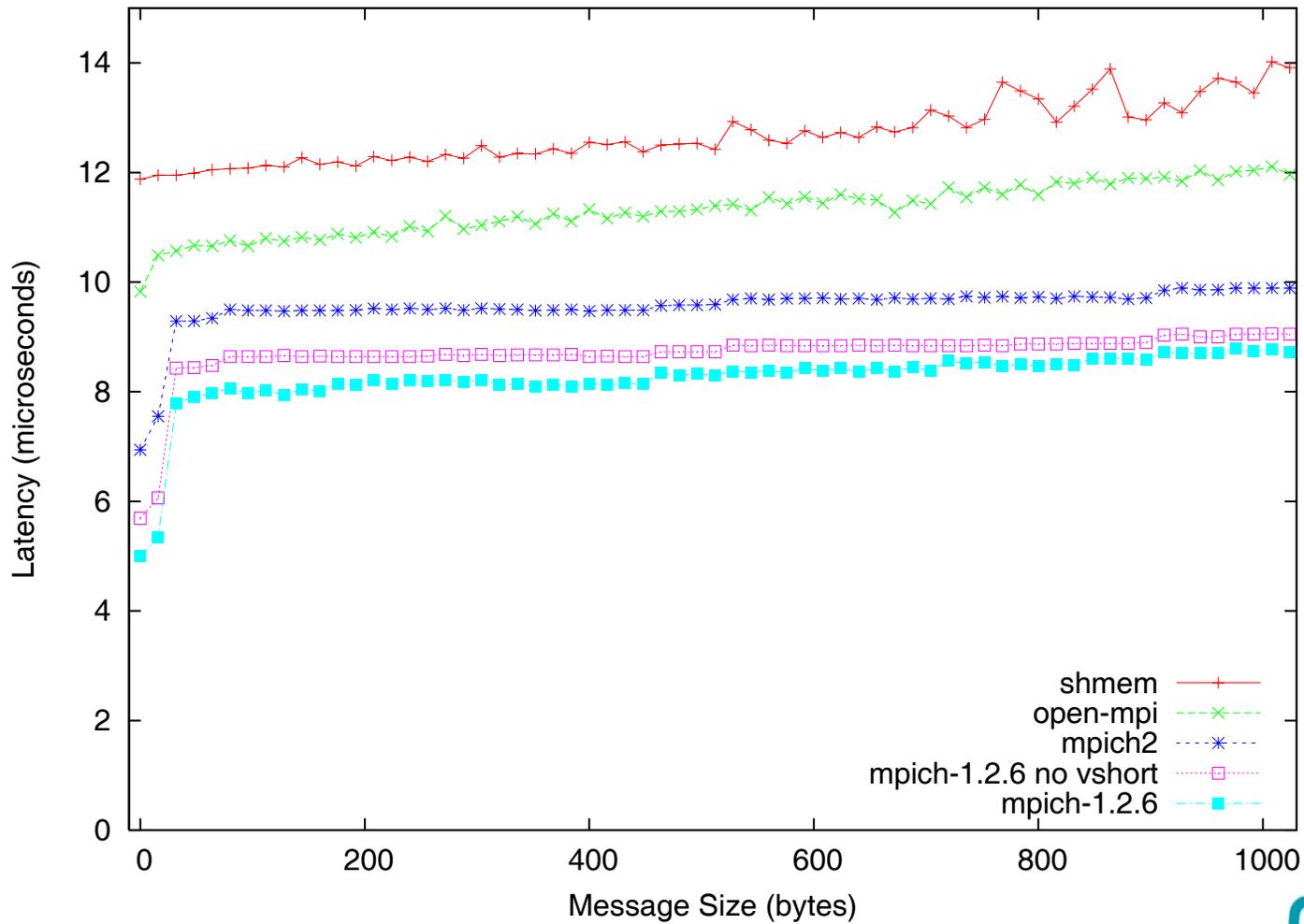


mpilatency Results



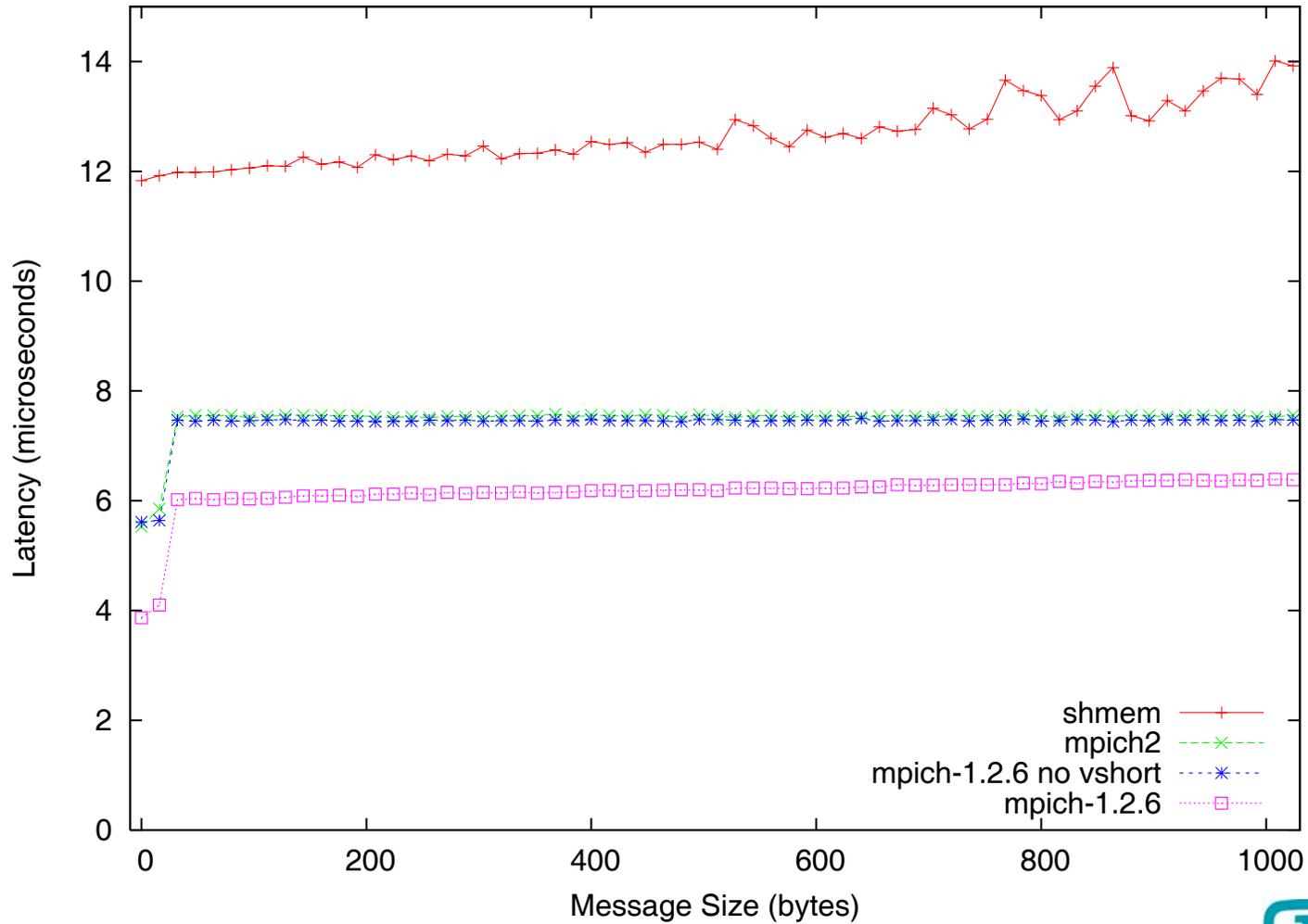


Generic



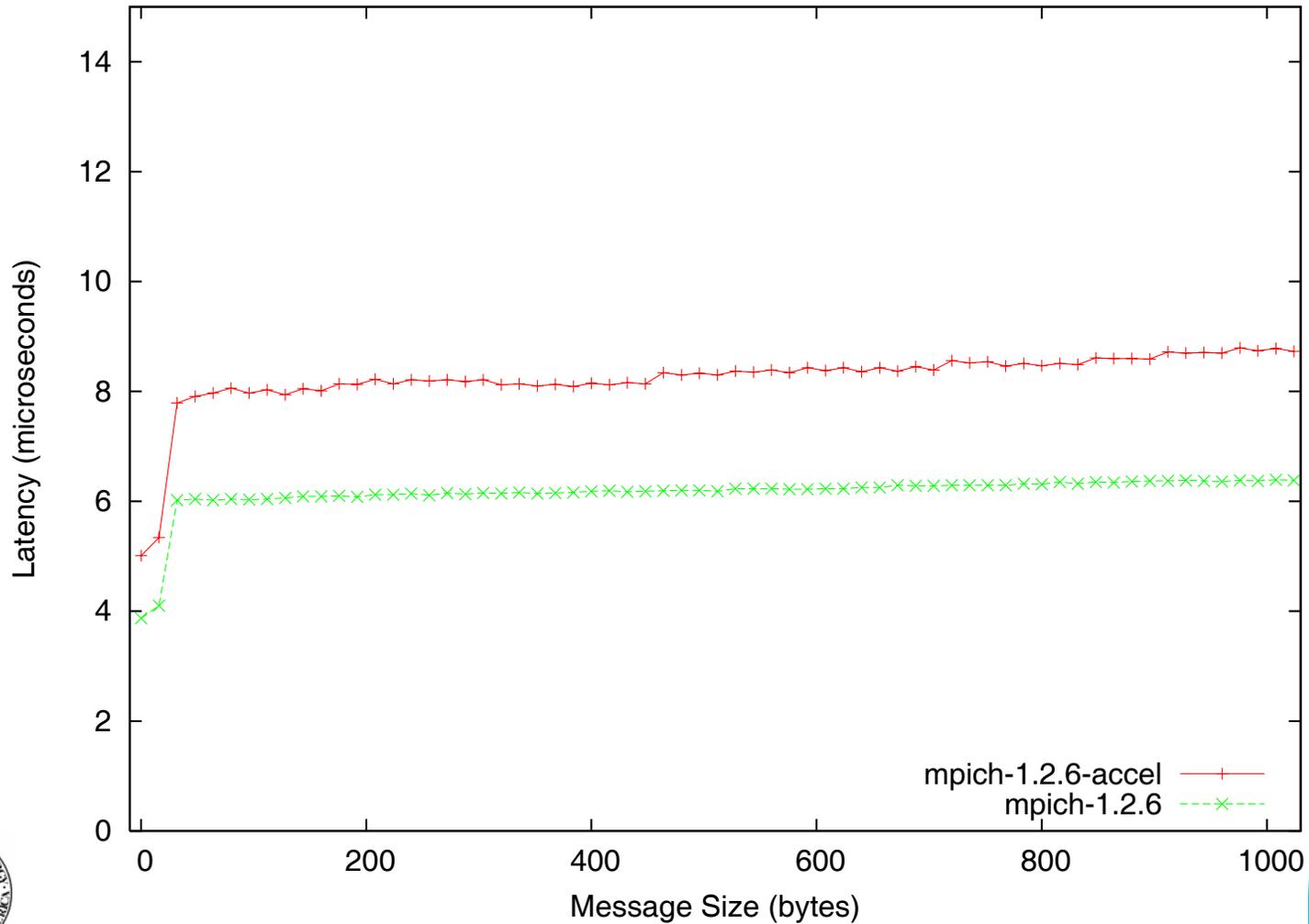


Accelerated





Accelerated vs. Generic



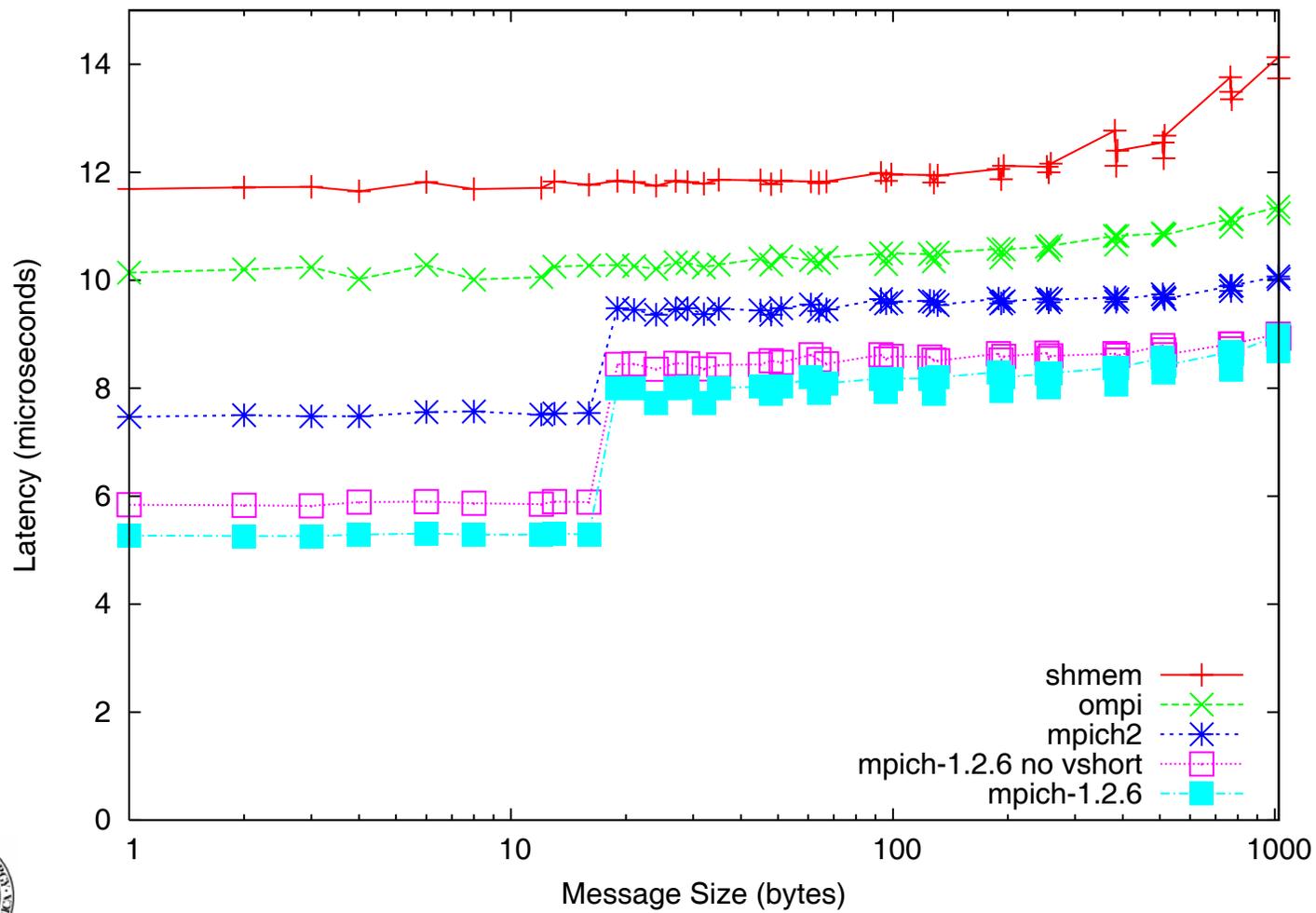


NetPIPE Results



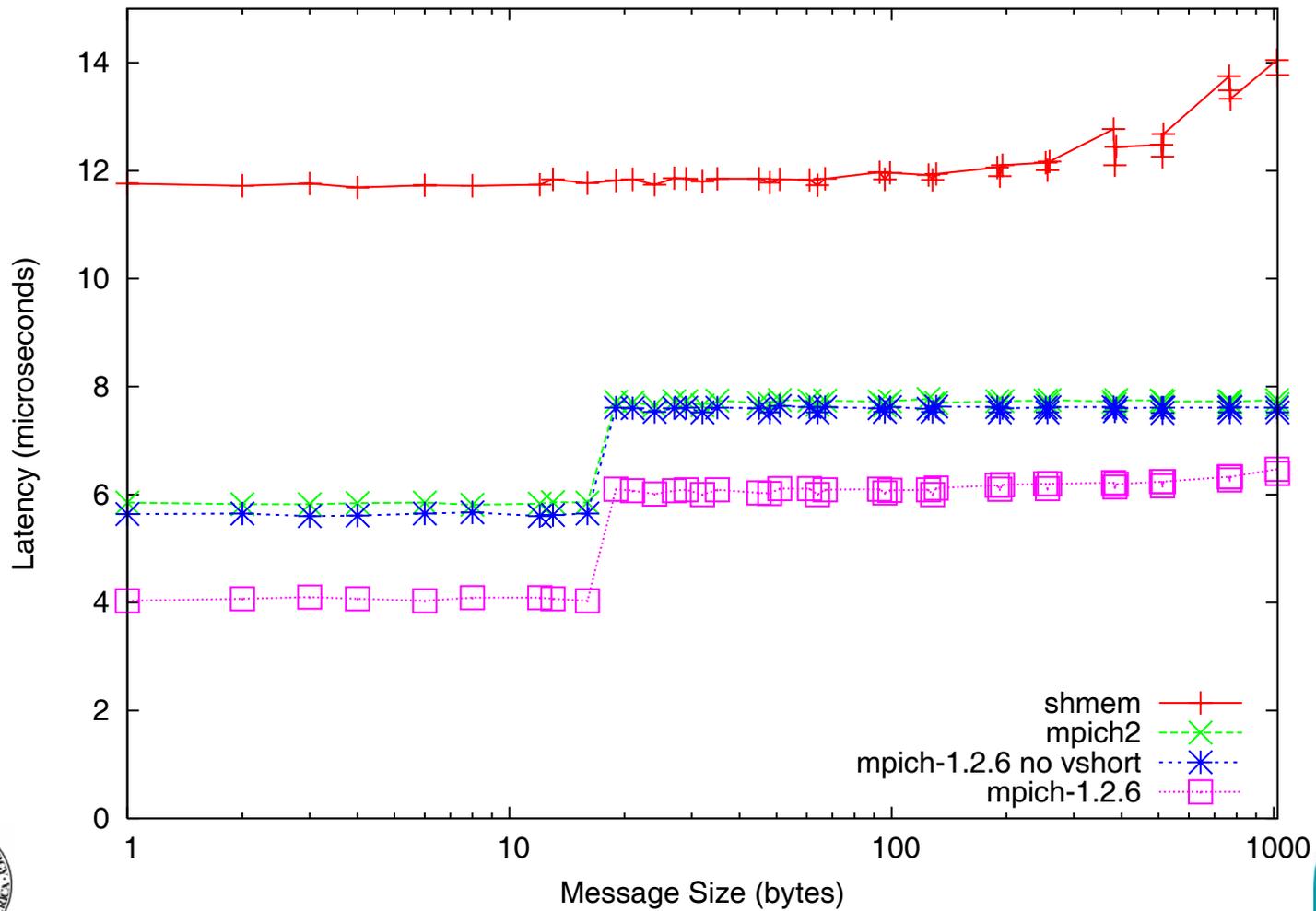


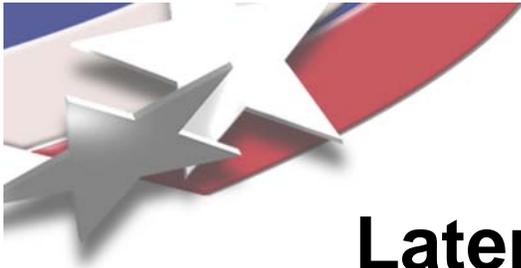
Latency - Generic



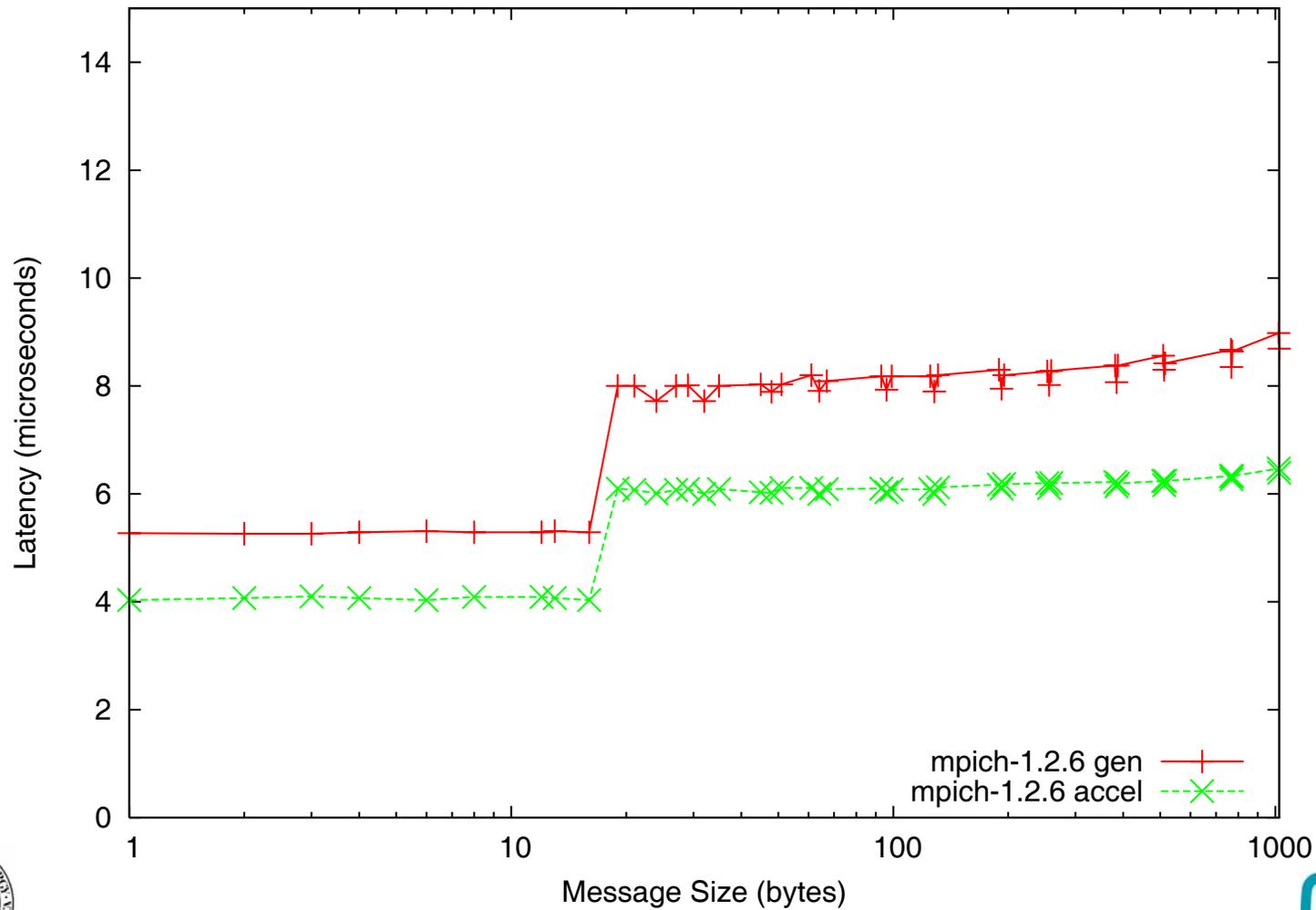


Latency - Accelerated



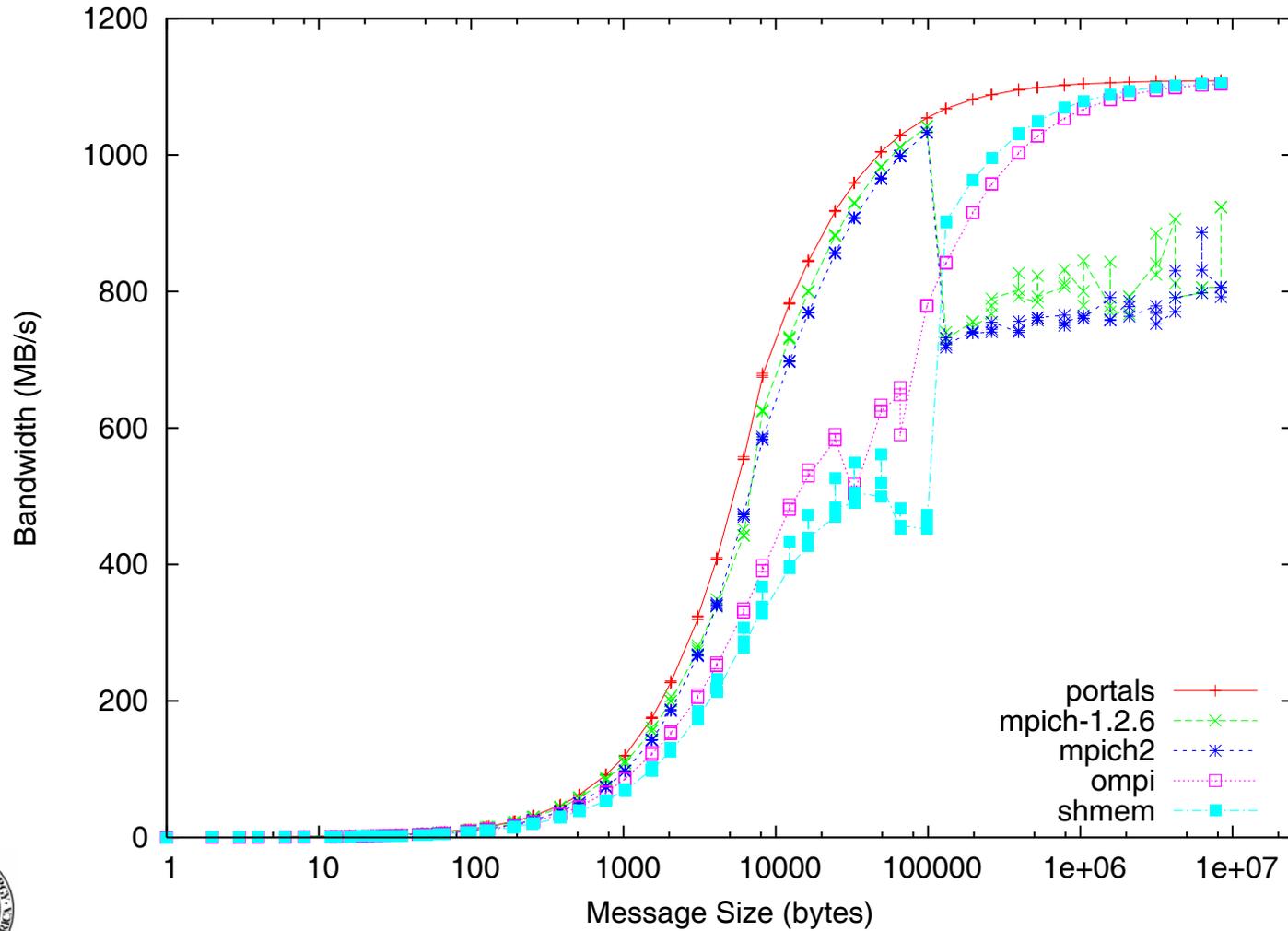


Latency – Generic vs. Accelerated



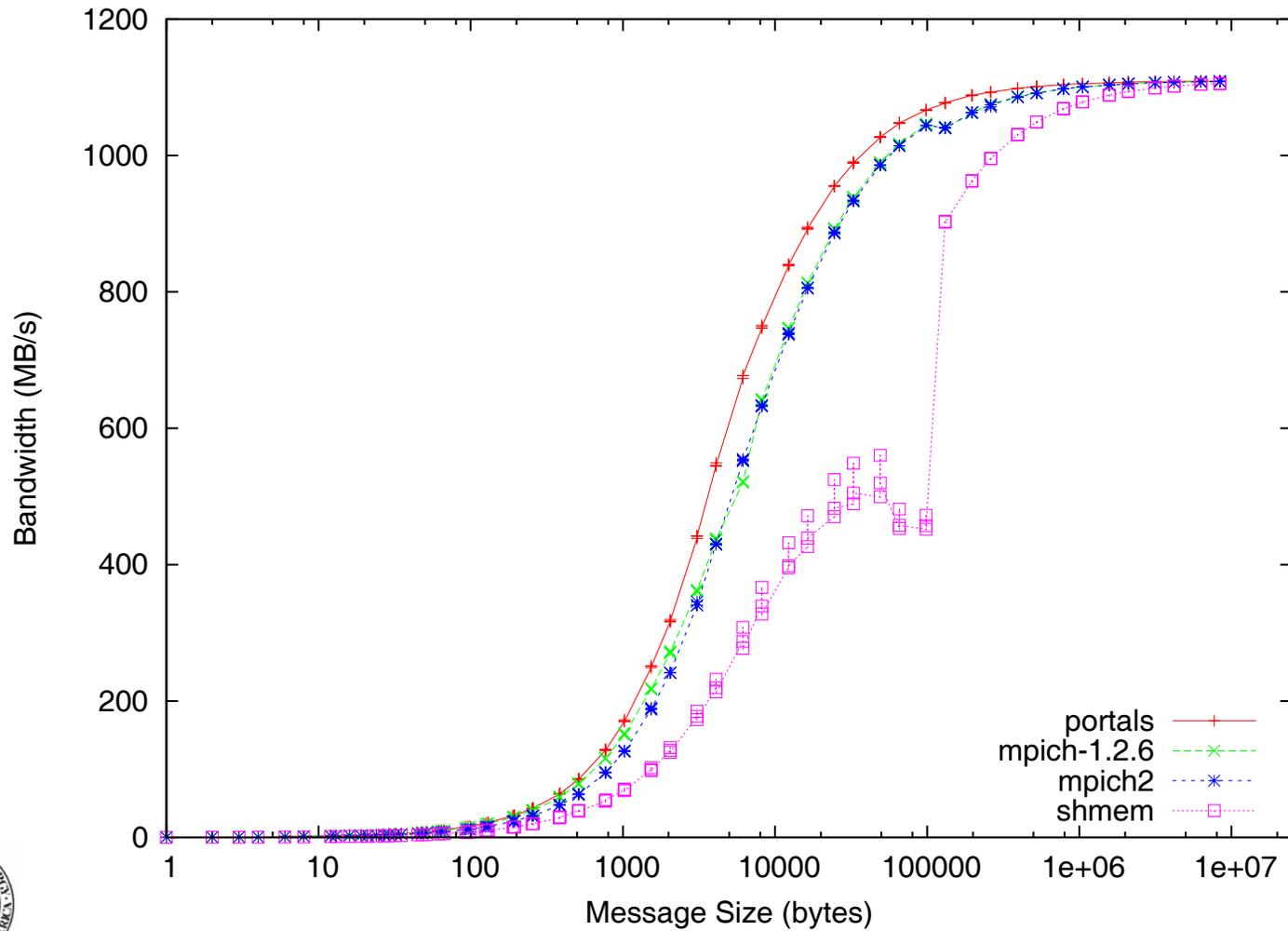


Bandwidth - Generic



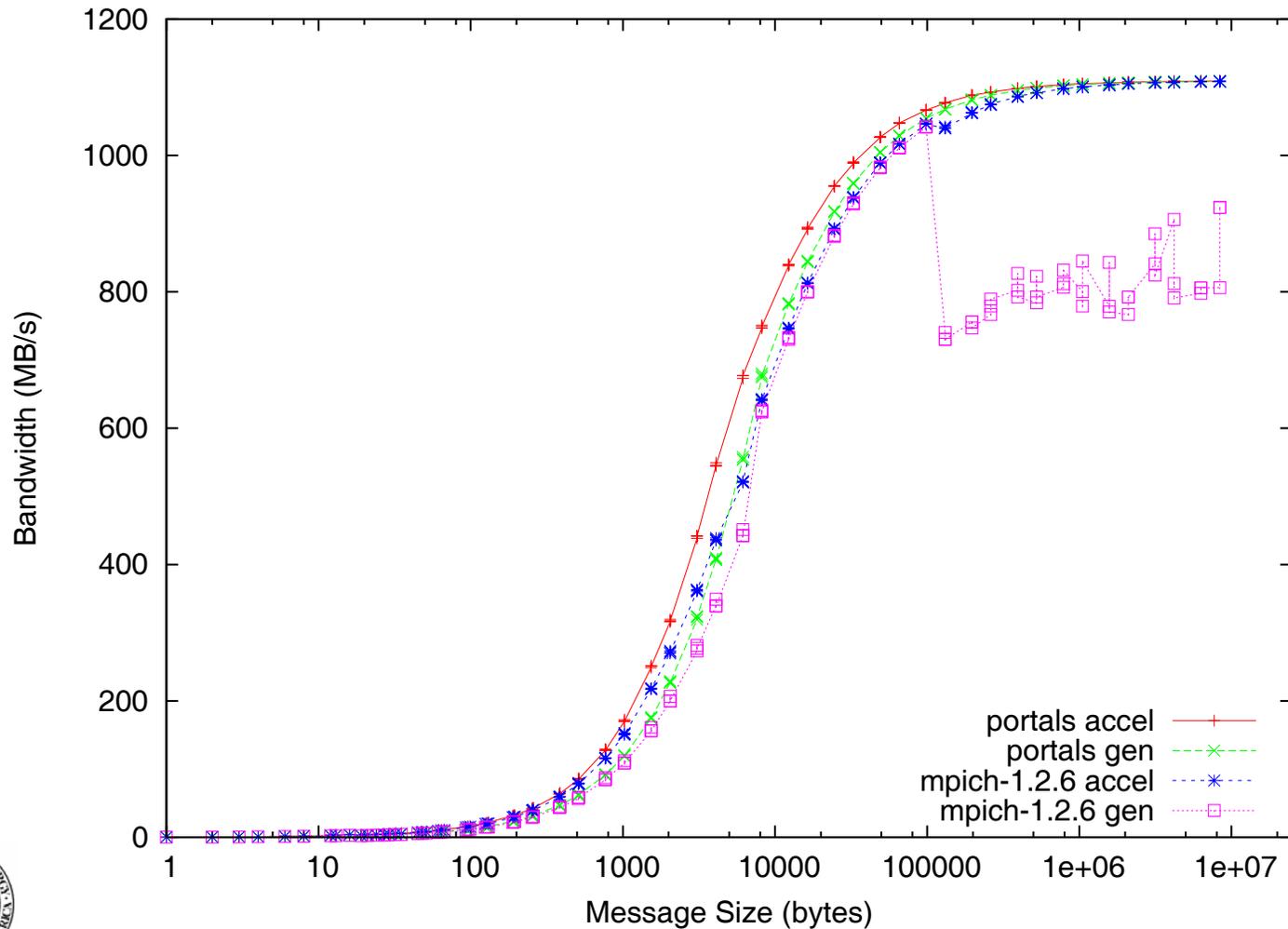


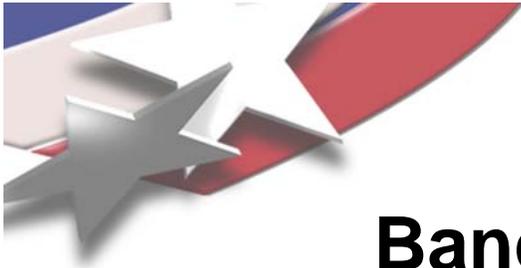
Bandwidth - Accelerated



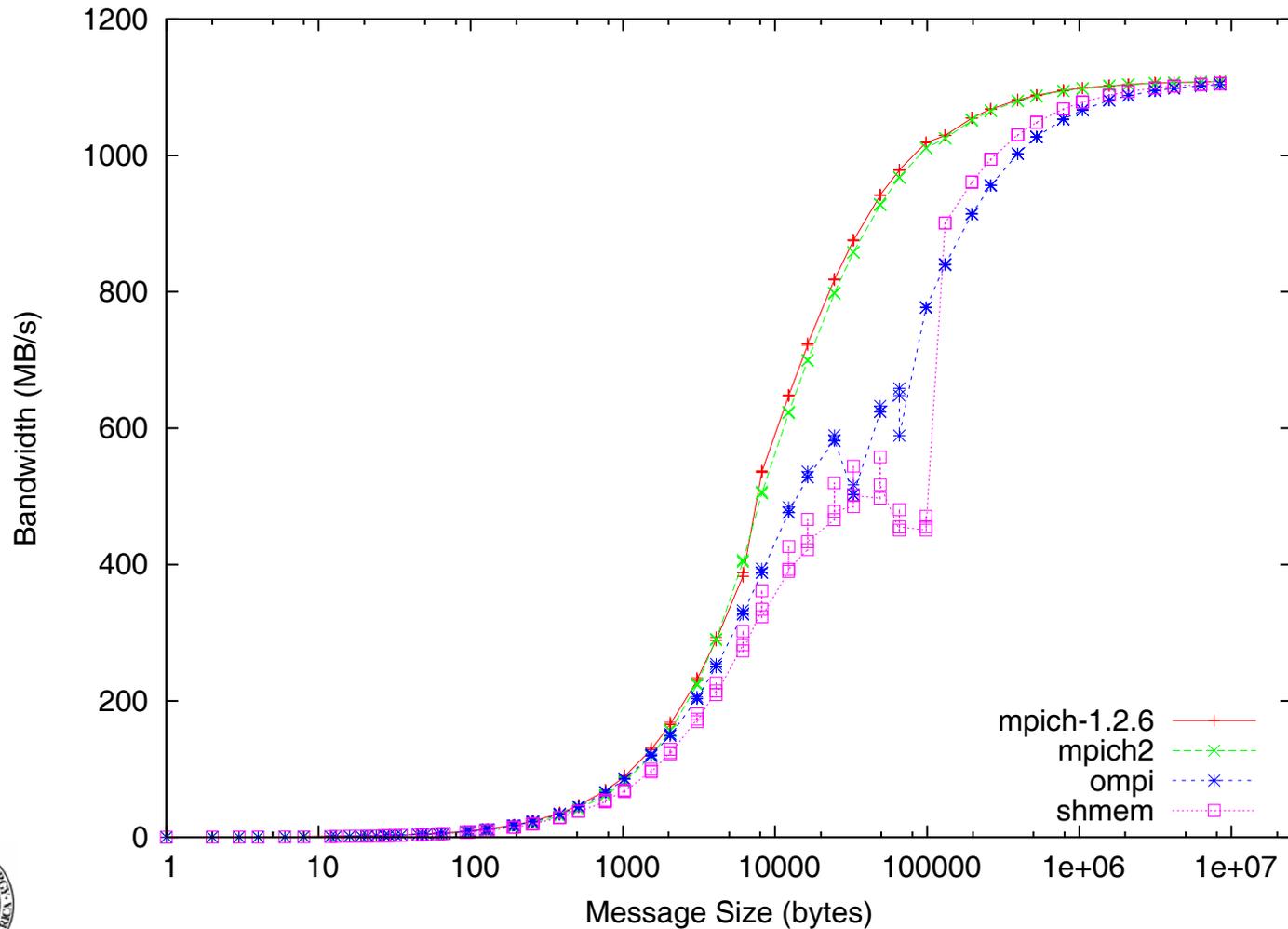


Bandwidth – Generic vs. Accelerated



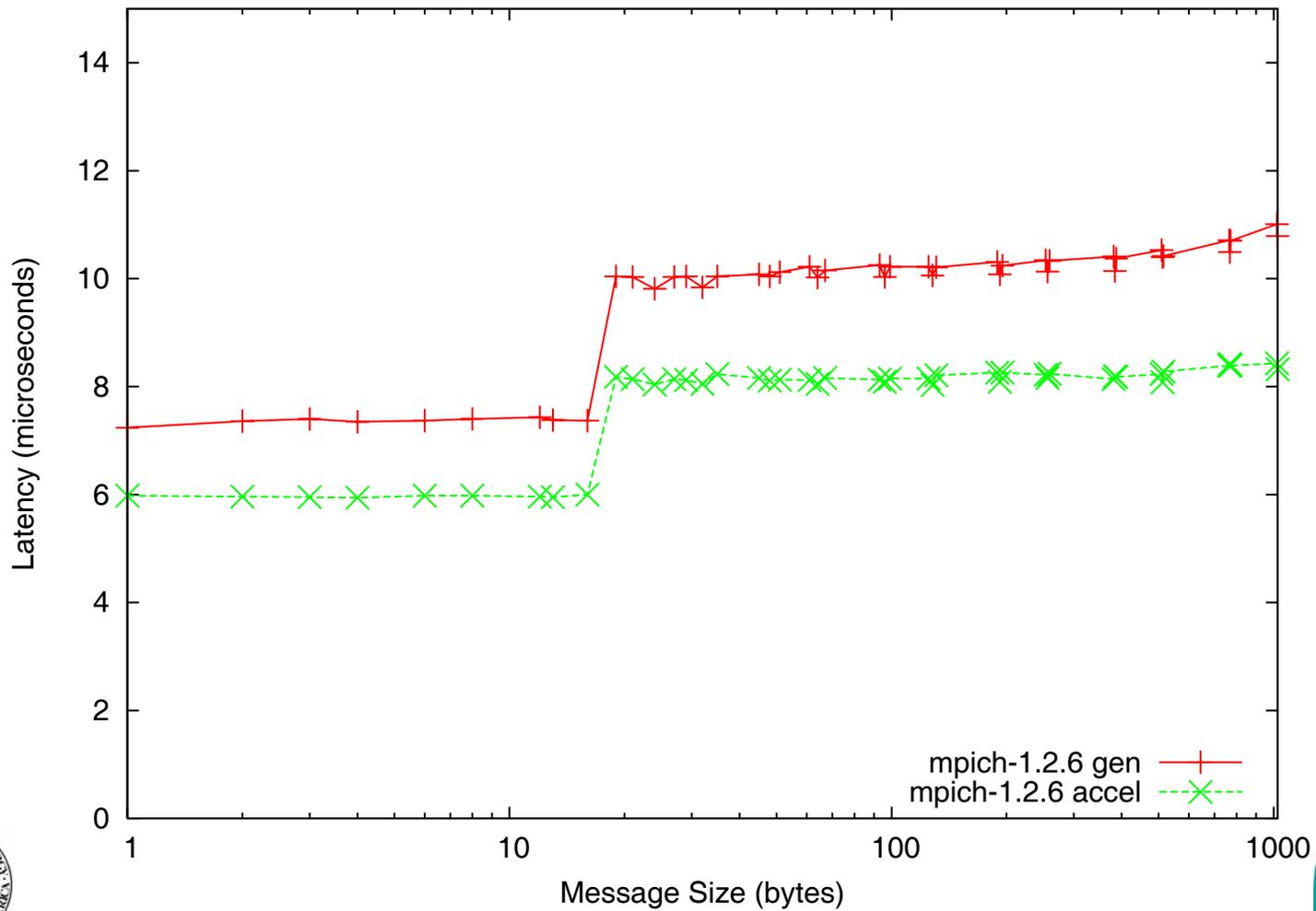


Bandwidth – Preposted - Generic



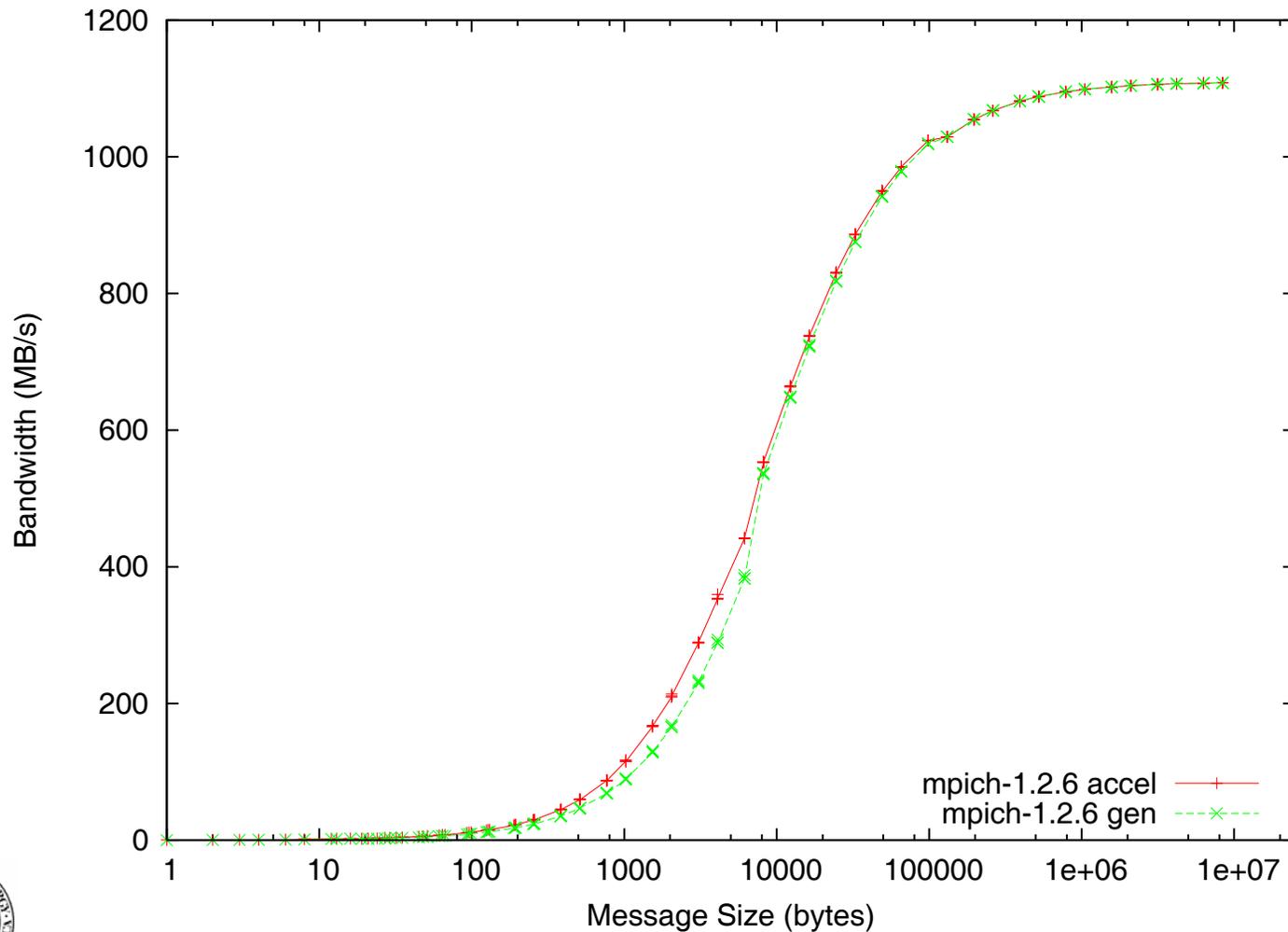


Latency – Preposted – Gen. vs. Accel.



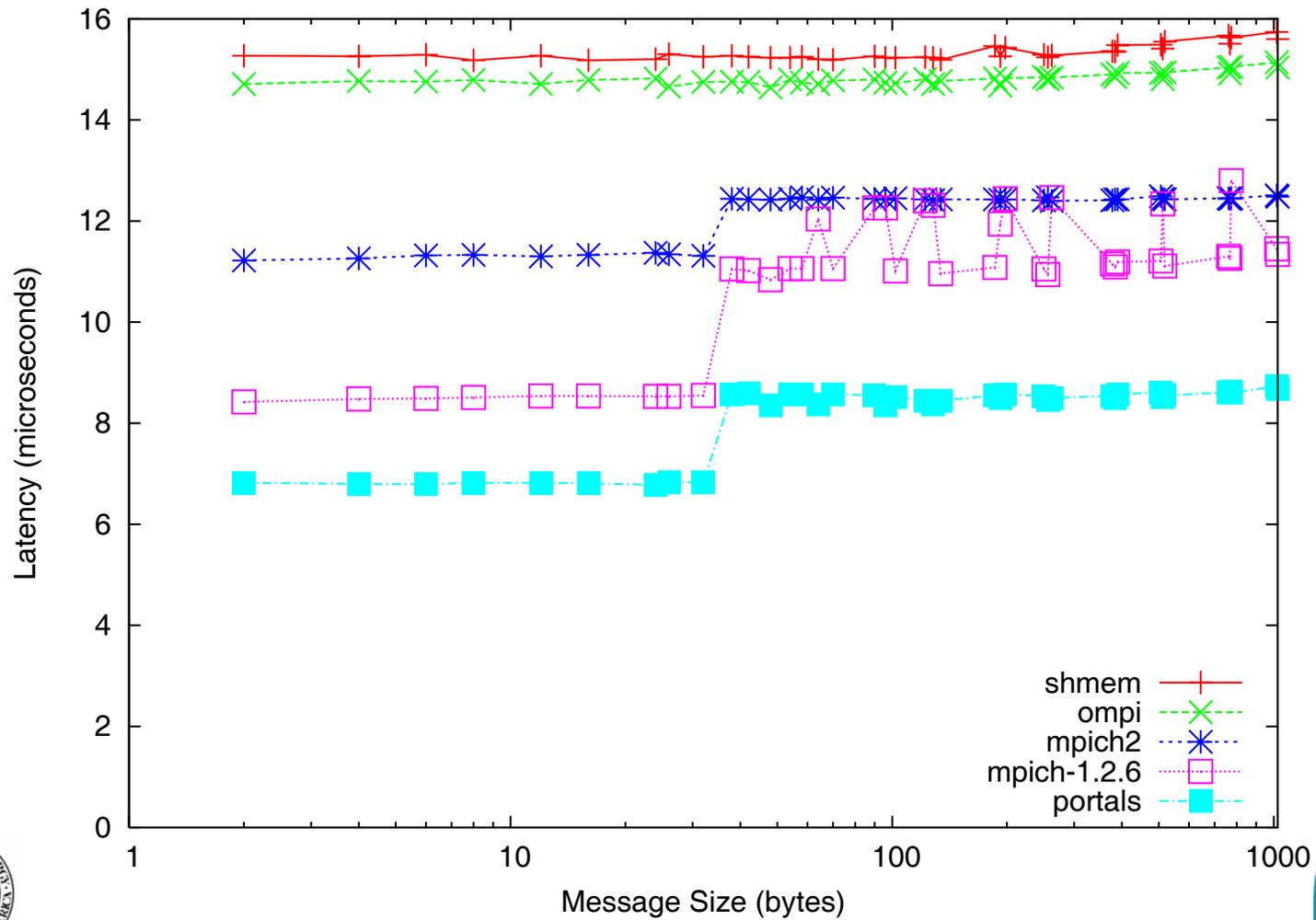


Bandwidth – Preposted – Gen. vs. Accel.



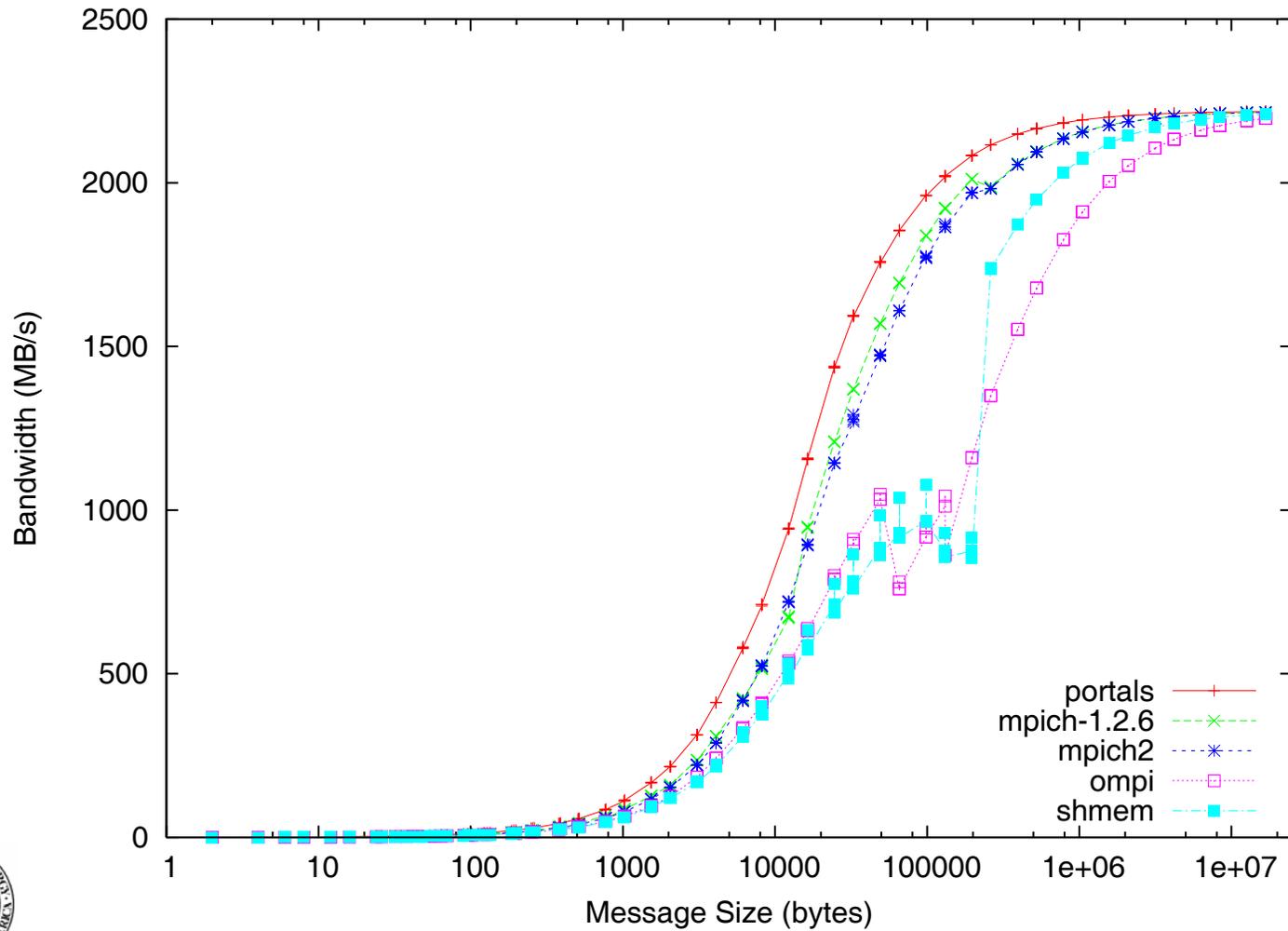


Latency - Bidirectional - Generic



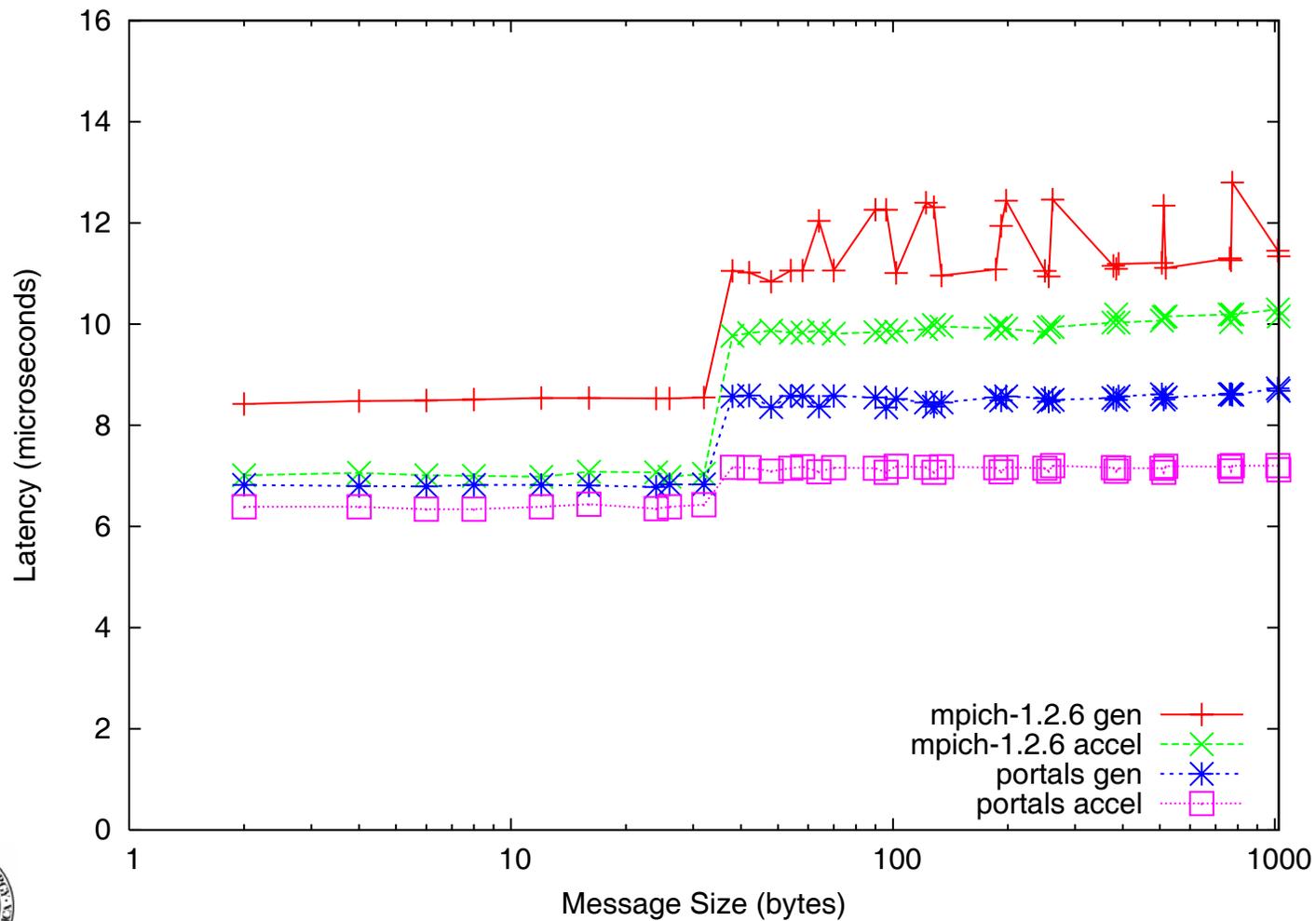


Bandwidth – Bidirectional - Generic



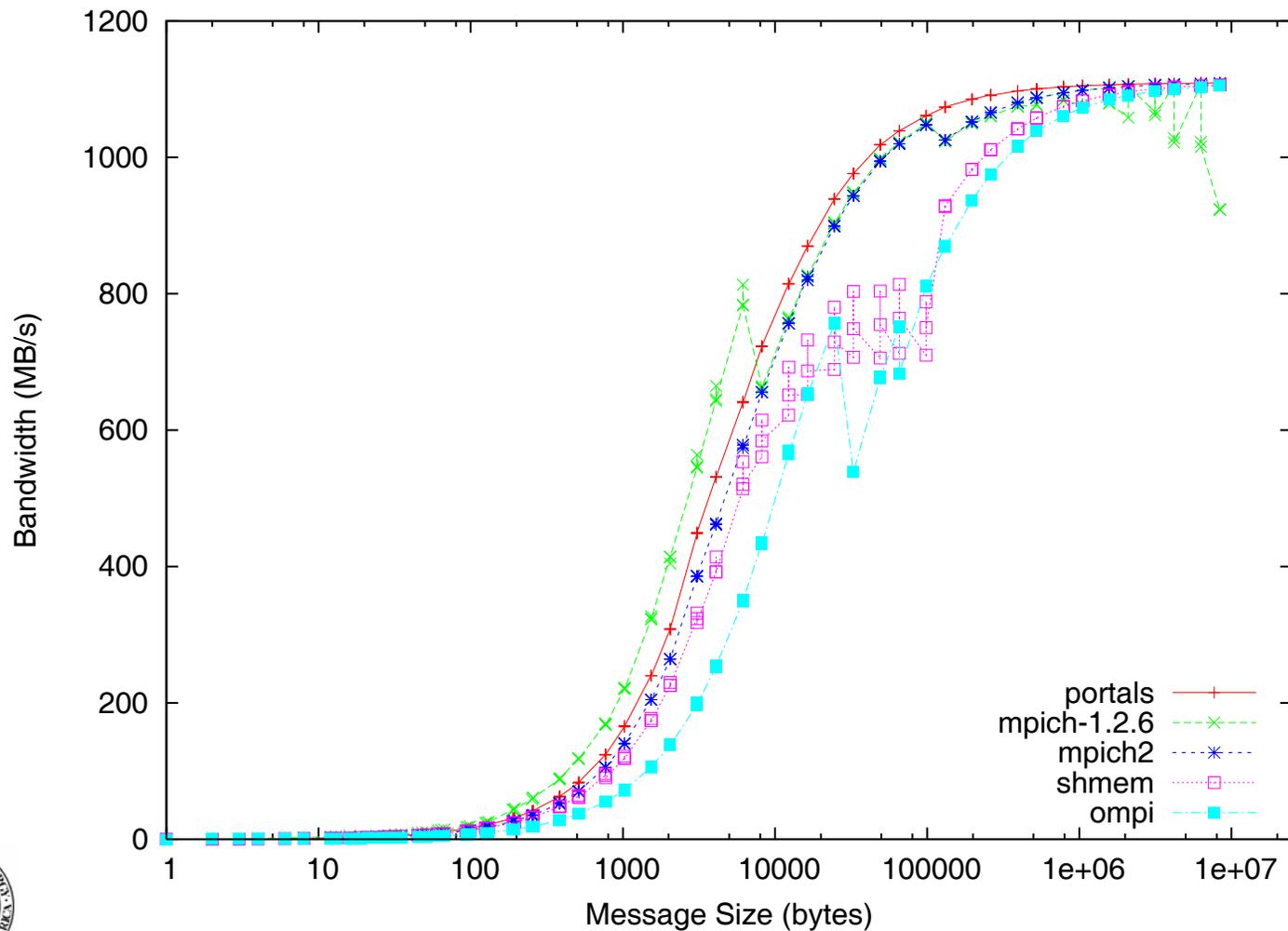


Latency - Bidirectional – Gen. vs. Accel.



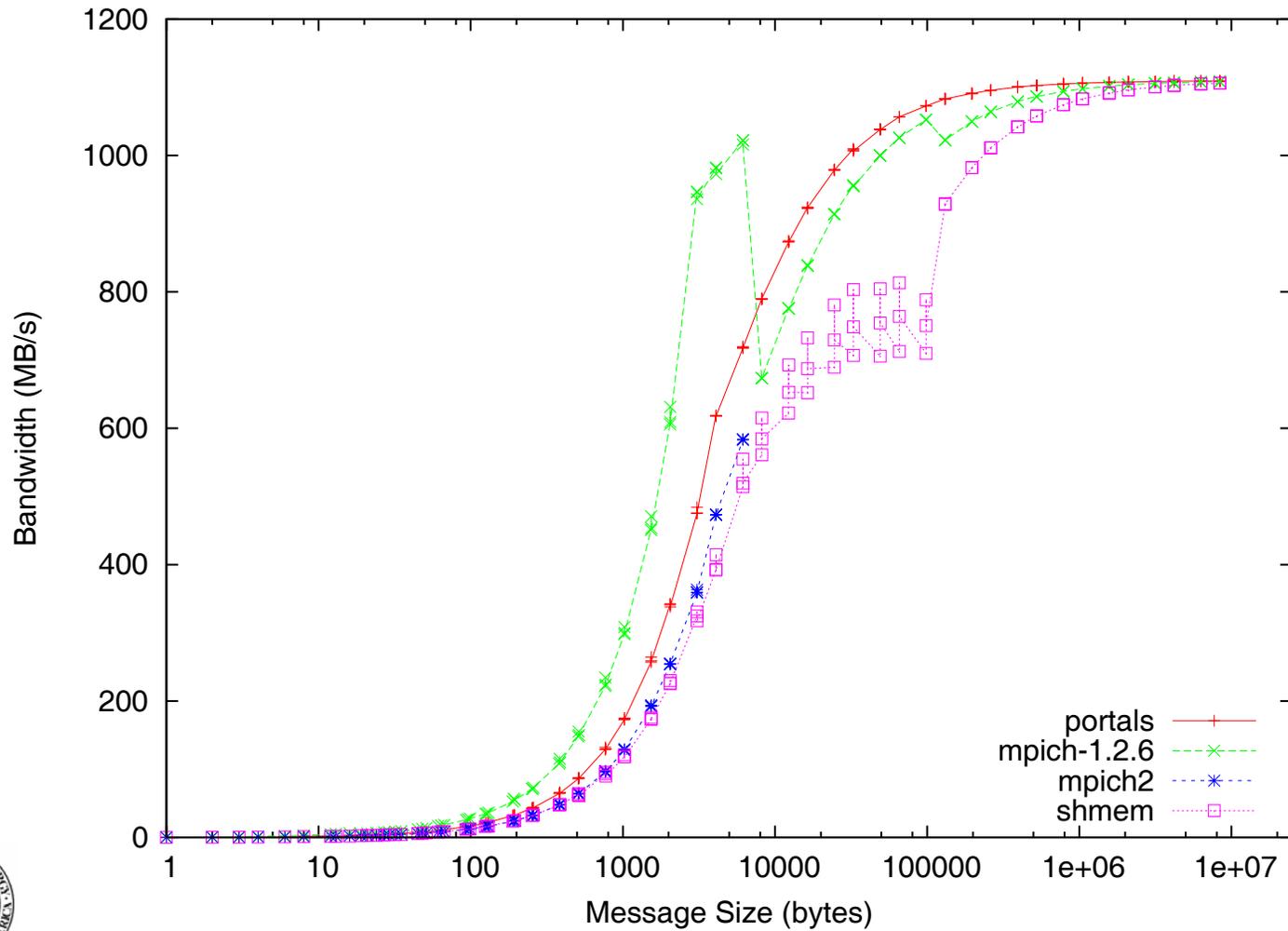


Streaming - Generic





Streaming - Accelerated





Conclusions

- **MPICH 1.2.6 short message optimization**
 - Decreases latency
 - Increases streaming bandwidth
- **MPICH2 is still slightly slower than MPICH 1.2.6 without short message optimization**
- **Portals-level matching provides a significant benefit**





Future Work

- **Tune accelerated implementation of Portals**
- **Work with Cray to get short message optimization into MPICH2**
- **More optimizations for MPI**
 - Use persistent memory descriptors for send side
 - Rendezvous protocol for benchmarking
- **More and better benchmarks**
 - CPU utilization/overhead
 - Collective operations
- **NIC-based collective operations**





Acknowledgments

- **Lots of people at Sandia**
 - **Especially Kevin Pedretti, Tramm Hudson, Keith Underwood, Jim Laros, and Sue Kelly**





Questions?

