OPEN IB ALLIANCE

Open MPI and OpenIB

Past, Present, and Future

http://www.open-mpi.org/

Overview

- The Open MPI project
  - High Performance Computing (HPC)
- Current OpenIB support
  - Deployment at Los Alamos
- Collaborative Efforts
  - Cisco's view of Open MPI and OpenIB

The Open MPI Project

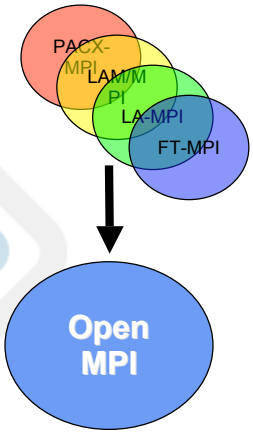
Jeff Squyres
Indiana University

Genesis

- Developers of FT-MPI, LA-MPI, LAM/MPI
  - Kept meeting at conferences in 2003
  - Culminated at SC 2003: “Let’s start over”
  - Open MPI was born
- Started serious design and coding work January 2004
  - Demonstrated at SC 2004

Open MPI

- Next generation MPI
- Merger of ideas from
  - FT-MPI
  - LA-MPI
  - LAM/MPI
  - PACX-MPI



Technical Contributors

- Indiana University
- The University of Tennessee
- The University of Houston
- Los Alamos National Laboratory
- High Performance Computing Center, Stuttgart
- Sandia National Laboratory - Livermore

Differences = Strength

- Each organization:
  - Shares some core values
  - Has non-overlapping / different goals
- …but that is ok!
  - In fact, this is what makes us strong

Open MPI Project Goals

- Open source philosophy
  - Vendor-friendly license (BSD)
  - Open development / access to source code
- Multi-purpose platform
  - Production quality
  - Vehicle for research
- Rapid deployment in new environments
- Shared development effort

Open MPI Project Goals

- Actively pursue community / 3rd party involvement
  - Vendors: ISV, network, system
  - 3rd party researchers and developers
- Prevent “forking” problem
  - Solicit feedback
  - Enable contributions without redistribution

Design Goals

- High performance
- Extend / enhance previous ideas
  - Component architecture
  - Message fragmentation / reassembly
  - Design for heterogeneous environments
    - Multiple networks (run-time selection and striping)
    - Node architecture (data type representation)
  - Automatic error detection / retransmission
  - Process fault tolerance

Implementation Goals

- All of MPI-1 and MPI-2
- Optimized performance
  - Low latency
  - High bandwidth
- Production quality
- Thread safety and concurrency (MPI_THREAD_MULTIPLE)
- Asynchronous progress

Implementation Goals

- Based on a component architecture
  - “Plug-ins” for different capabilities (e.g., different networks)
  - Same binary can utilize different capabilities
  - Allows independent distribution
  - Important to ISVs, system integrators
- Flexible run-time tuning
  - System-wide and per-user settings

Current Status

- v1.0 released November, 2005
- Supported networks
  - OpenIB, mVAPI, GM, MX, TCP, Portals, shared memory
- Supported run-time environments
  - SLURM, PBS / Torque, BProc, rsh / ssh, Yod, Xgrid, POE, BJS

Upcoming Work

- New collective algorithms
- General performance improvements
- Threading stability
  - MPI_THREAD_MULTIPLE
  - Asynchronous progress
- Data and process fault tolerance
- More run-time environments and networks

Open MPI and OpenIB: Current Status

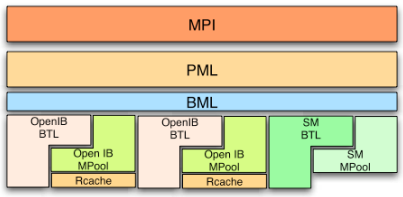
Tim Woodall

Los Alamos National Laboratory

LA-UR-06-0465

Point-to-Point Frameworks

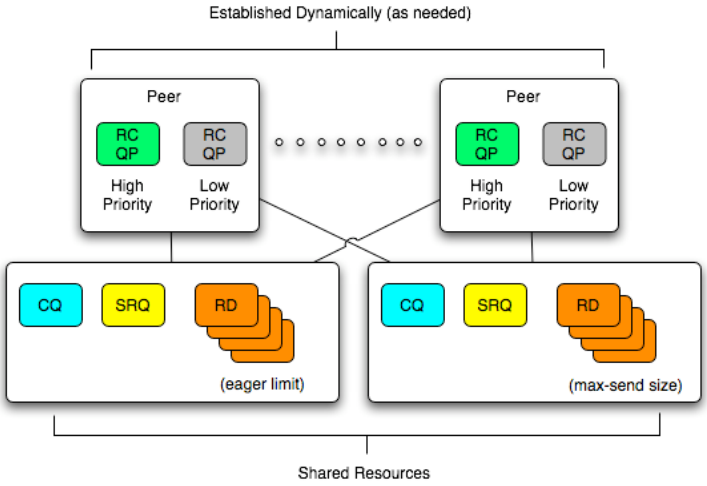
- **Point-to-point Messaging Layer (PML) implements MPI semantics**
- **BTL Management Layer (BML) multiplexes access to BTLs**
- Byte Transfer Layer (BTL) abstracts network interfaces
- Memory Pool (mpool) provides memory mgmt. / registration
- Rcache maintains optional MRU cache of memory registrations



OpenIB Support

- BTL module created for each active port
  - Upper layer supports message striping across available BTLs
- RC Based - Connections/Queue Pairs dynamically allocated as needed
  - Addressing communicated via seperate Out-Of-Band channel
- Resources can be allocated to each QP or a single Shared Receive Queue (SRQ)

Resource Allocation



OpenIB Planned Support ('06)

- Connection establishment via CM / SA Services
- Optional short message RDMA Write (small working set of peers)
- Multicast-based collectives
- Data Reliability / NIC failover
- UD based BTL (Sandia Livermore)
  - Explore scalability and performance
  - Leverage data reliability

Los Alamos: Coyote Cluster

- Production hardening on new 1400+ node system at LANL
  - Dual Opteron / Mellanox Mem-Free Cards
  - Open IB Stack (Mellanox IB gold 2.0.1)
- Current Testing (Limited to Individual SUs)
  - 270 nodes / 540 processes
  - HPL, Pallas, QCD, other apps
- No significant issues
  - However....

Shared Receive Queue Issues

- SRQ allows multiple QPs to use one pool of (fixed-size) buffers
  - Avoids the complexity of figuring out how to distribute buffers across QPs
  - Does not help determine
    - Appropriate number of buffers
    - Appropriate size of buffers
  - These are highly dependent on application behavior
    - Application message passing patterns are not fixed
    - Dependent on input data, number of processes, application algorithm, etc.
- Lack of flow control requires complexity in MPI to ensure that buffers will always be available

OpenIB Requests: Wish List

- Proposal: Shared Receive Block (Ron Brightwell - Sandia)
  - SRQ / RQ without individual message buffer boundaries
  - Use a large block of memory for incoming messages
  - Messages flow one right after the other into the block
  - Running offset allows the next message to start where the previous message ended (plus alignment)
  - Efficient use of memory dedicated to the network
    - Small messages do not waste memory
    - An 8 KB buffer is not consumed by an 8-byte message
  - Approach used by Portals for unexpected messages
- SRQ Flow Control?

OpenIB Requests: Wish List

- Caching of memory registrations is an Issue
  - Forced to integrate ptmalloc2 library to intercept sbrk() / munmap()
  - Not desirable: uses linker tricks that lead to compatibility issues
  - Glibc memory hooks are not thread safe
  - Need kernel support for notification of changes to registered pages
- Re-Register Verb
  - Provide the capability to extend an existing registration
  - Leverage existing page mappings
  - Should not invalidate pending operations

OpenIB Requests: Wish List

- Reliable multicast
- Scalable query of path / multi-path records / topology discovery
- User-level congestion notification

OPEN IB ALLIANCE

Collaborative Efforts

Shawn Hansen
Product Management,
Cisco Systems

Cisco's Position

- OpenIB is strategic
  - Cisco will move strategic focus to OpenIB in engineering and capital resources.
  - Cisco will shift from proprietary to OpenIB stack as quickly as possible.
- Consolidation of stacks
  - One stack, fully interoperable between vendors
  - InfiniBand provides core shipping RDMA foundation for future Cisco customers, including iWARP.

What's Required from Industry

- To harden OpenIB to commercial quality, we must provide:
  - Large-scale testing facilities
  - ISV application testing
  - Thriving partner ecosystem
    - Storage
    - File systems
    - Etc.

State of MPI

- MPI similarly requires consolidation
- Lack of coordination between vendors
  - Simultaneous duplicative engineering efforts
- Fracturing of commercial ISV testing slows InfiniBand adoption.
  - ISV: "Which one do I support?" "I don't have resources to test them all."
  - Customer: "I must have benchmarks and certifications."

The Promise of Open MPI

- Greater opportunities for collaboration
- Less fracturing in community
- Shared ISV certifications and testing
- Multi-OS support
- Multi-fabric support

Key Challenges

- Performance / latency tuning
- Consistent testing on large-scale clusters
- Access to current hardware (DDR, multi-switch networks)
- Need for commercial influence
- Drive ISV testing and benchmarking

Key Areas of Enhancement

- Tune MPI collective operations
- Add scalable RDMA (write) short message protocol.
- RDMA memory registration etc.
- Hardware multicast support
- Robust failover / data reliability
- MPI-IO
- iWARP support

Open MPI Workshop

- Hands-on tutorial for Open MPI developers
  - Overall architecture of Open MPI
  - Developing MPI support: point-to-point networks, collective algorithms, etc.
  - Developing RTE support: talking to schedulers, obtaining allocations, etc.
- Taught by Open MPI core developers
- April 17-21, 2006 (M-F)
  - Hosted by Cisco Systems, San Jose, CA, USA

Conclusions

- Open MPI is available now
  - http://www.open-mpi.org/
  - v1.0.2 coming shortly
- Don't like something? Need a new feature? Join the process!
  - Mailing lists
  - Subversion access
  - Vendor licensing
  - Developer access

OPEN IB ALLIANCE

Questions?