State of the Union

Jeff Squyres

Current Status

- Current stable release: v1.0.2
  - Small number of fixes for v1.0.3
  - Expect release at same time as v1.1
  - Separate branch in repository
- Head of development
  - Eventually to become v1.1 (mid-May?)
  - Performance and feature enhancements vs. 1.0.x (more later on this)

V1.0.x Status

MPI Conformance Status

- All of MPI-1, most of MPI-2
  - Does not include MPI-2 one-sided
  - Only as much MPI-2 I/O as ROMIO
  - +/- interface bugs
  - Fortran 90 bindings
  - MPI-2 dynamics “functional but klunky”

Top-Level Plugins

- Point-to-point networks
  - TCP, shared memory, GM, MX, mVAPI, OpenIB, Portals
  - True multi-device support
- Resource managers
  - rsh/ssh, BProc, SLURM, PBS/Torque, Xgrid, Yod
  - Must be inside an RM job

Performance

- “Reasonable”
  - Needs SM optimizations
  - Needs TCP optimizations
  - Does not have small message RDMA optimizations for InfiniBand
- Collective performance: bad
  - Standard linear / log algorithms
Threading
- Threading designed in from beginning
  - MPI_THREAD_MULTIPLE
  - Asynchronous progress
- Lightly tested
  - Serial applications running with MPI_THREAD_MULTIPLE

Esoteric Features
- Processor / memory affinity
  - NUMA-aware collectives: barrier, broadcast, reduce, allreduce
  - Affinity plugins for Linux, Solaris
- True MPI-2 I/O integration
  - MPI_Request, not MPIIO_Request
    (but MPIIO_Request works as well)
  - Can mix-n-match IO, point-to-point, and generalized requests in vector test / wait

Documentation
- Very little
  - FAQ keeps getting larger
  - But little in the form of “glossy PDF”
- Mailing lists are Googlable
  - Much traffic, questions

End of Life for v1.0
- Estimate releasing v1.0.3 around same time as v1.1
  - Contains any unreleased fixes for v1.0
  - Necessary: it’s the current stable series

Progress Since v1.0.x
- Overall performance enhancements
  - Decrease latency in Myrinet, IB, shmem
  - Better memory registration handling
  - Pipelined protocols (hides un/register latency)
  - Small message RDMA for IB
- [Far] Better collective performance
- Data reliability
  - Checksum, retransmit
Progress Since v1.0.x

- Basic MPI-2 one-sided implementation
- Various MPI interface fixes
- Heterogeneity
  - Mix 32 / 64 nodes in a single run
  - Mix endian machines in a single run
- Better Fortran 90 support
- PERUSE support
- More thread testing

Release Plan

- Just recently branched from trunk
- /branches/v1.1
- But then the key developers started making slides for this workshop 😞
- Needs testing and stabilization
- SWAG for release
- Mid-May 2006

Mostly Undefined

- SC timeframe
  - November 2006
- Only recently too a guess at the features
  - Much is undefined

Operating Systems

- Microsoft Windows
  - Compile Open MPI under Cygwin
    - Using native MS compilers
    - Distribute binaries
    - Requires Libtool 2.0 (unreleased)
  - TCP with rsh/ssh works
  - Targeted for November 2006 release (SC)
- Others (currently loosely tested)
  - Solaris, AIX, ...?
**Networks**
- Continue to optimize current networks
- New network possibilities
  - TCP/IPv6
  - SCTP
  - LAPI
  - Low latency Ethernet
  - uDAPL (in progress)

**Collectives**
- Active area of research (UTK)
  - Continue research work in this area
- More NUMA-aware collectives
- Topology-aware collectives
- Collective plugin framework version 2
  - Fine-grained algorithm selection
  - Non-blocking collectives (MPI extension)
- …more details TBD (still under design)

**Threading**
- Add asynchronous progress
  - IB, Myrinet, shared memory
- Heavy MPI_THREAD_MULTIPLE testing
  - Real multi-threaded MPI applications

**Fault Tolerance**
- Involuntary coordinated checkpointing
  - Application unaware that it was checkpointed
  - System- and user-level
  - By November 2006 (SC)
- FT-MPI technologies
  - Entirely new FT framework
  - In progress; possibly by SC
- NIC pause / failover
  - Possibly by SC

**Run-Time Enhancements**
- More resource managers
  - POE, SGE, XCPU, XGrid (improvements)
- Allow job submission
- Remote execution (launch from laptop)
- Attach / detach from running jobs
- Scalability improvements

**Esoteric Enhancements**
- Processor affinity tie-ins from resource manager
- “Better” MPI-2 IO support
- “Better” MPI-1 topology support
- Fortran 2003 MPI bindings
Interoperable MPI

- Being worked on by AIST (Japan)
- IMPI defines a wire protocol for MPI
  - Good for Grid-like scenarios
  - Allows hooking multiple MPI implementations into a single MPI_COMM_WORLD

“Someday”

- Move BTL’s down to OPAL
  - Provide more than just MPI
  - Co-Array Fortran, … etc.
- Greatly expand run-time system
  - Support more disconnected scenarios
  - Launch from laptop, disconnect
  - “Grid”-like scenarios (multi-cluster)