MPI Forum ABI WG Status and Discussion

Dublin meeting

September, 2008

Context

- “convenience” item
- Will allow dynamic run-time re-link (e.g. through LD_LIBRARY_PATH)
- Of most use to software providers that layer on top of MPI (ISVs, tools, parallel apps, etc.)
- Some push back from implementers

Status

- Focus on C bindings
- Hope that fortran WG will eliminate fortran specific bindings in the long run
- First step: agree on a common mpi.h spec
- After that: other items as needed
  - Environmental items (e.g. mpiexec)
  - Calling/Linkage conventions
  - …

Participating organizations

Initiating organizations:

- MathWorks
- Intel Corp.
- Microsoft Corp.
- QLogic
- HP

Members of working group

- alexander.supalov at intel.com
- davdav_2000 at yahoo.com
- david.solt at hp.com
- dries.kimpe at cs.kuleuven.be
- edric.ellis at mathworks.co.uk
- erezh at microsoft.com
- gil at dev.mellanox.co.il
- hayasaka at bx.jp.nec.com
- howardp at cray.com
- jeffb at lanl.gov
- jsquyres at cisco.com
- kannan.narasimhan at hp.com
- keller at hlrs.de
- kmm at cray.com
- lindahl at pbm.com
- mercierg at mcs.anl.gov
- ndebard at lanl.gov
- rabenseifner at hlrs.de
- sameer at cs.uoregon.edu
- schliephake at hlrs.de
- shirley at eecs.utk.edu
- terry.dontje at sun.com
- treumann at us.ibm.com
- yann.kalemkarian at bull.net
- jjolly at sandia.gov
- koziol at hdfgroup.org
- srimks11 at gmail.com

Status of mpi.h effort

- Spreadsheet in development that
  - Covers items is annex A
    constants, type definitions, info values and keys
  - Classifies items, (Organized by Bill Gropp's paper on GMPI)
    Compile-time values
    - Used in declarations (MPI_MAX_ERROR_STRING, etc.)
    - Other (MPI_ANY_SOURCE, MPI_ERR_TRUNCATE, etc.)

    Init-time constants (MPI_INT, etc.)
    Opaque objects (MPI_Comm, MPI_Datatype, etc.)
    Defined objects (MPI_Status)
    Defined pointers (MPI_BOTTOM, MPI_STATUS_NULL, etc.)
  - Exposes alternate implementations
    OpenMPI, MPICH 1 and 2, HP, LAMPI, NEC
  - Beginnings of an ABI column (focus of WG session)

Does the forum want to pursue this?

Path Forward

- MPI ABI standard, separate from MPI API standard, but under the umbrella of the MPI Forum
- Dynamic link is the main objective (binary compatibility)
- Startup (mpirun/mpiexec) is a secondary objective
- Include all major language bindings in the ABI standard (C, C++, Fortran) as we go
- Tied to MPI 2.1 and forward
- Implementers may choose to comply (or not)
- May be referenced in procurements
- May be implemented as a morph layer or native
- Initial reference implementation will be a morph layer (tbd who provides)
- May be different per platform (Linux, Windows, …)
- Different voting rules separate from MPI API forum (?)
- Need participation from major implementers in order to make progress

Focus groups

mpi.h: MPICH 2 (1.0.7) as base with mods as needed (e.g. max values for compile time values used in declarations)

Proceed with a reference morph layer implementation targeting (at least) MPICH 2.0, OpenMPI

Load a different module and it just works

Demonstrate with a ring program

Defer startup consideration

- Linux (Jeff, Alexander) on X86-64 cluster Red Hat Linux/ SUSE
- Windows (Alexander, Erez) – on X86-64 cluster
- Bring back to the December forum meeting

Technical tasks

- Common mpi.h (spreadsheet)