MPICH: 3.0 and Beyond

Pavan Balaji
Computer Scientist
Group Lead, Programming Models and Runtime systems
Argonne National Laboratory
MPICH: Goals and Philosophy

- MPICH continues to aim to be the preferred MPI implementations on the top machines in the world
- Our philosophy is to create an “MPICH Ecosystem”
MPICH on the Top Machines

1. **Titan (Cray XK7)**
2. **Sequoia (IBM BG/Q)**
3. **K Computer (Fujitsu)**
4. **Mira (IBM BG/Q)**
5. **JUQUEEN (IBM BG/Q)**
6. **SuperMUC (IBM InfiniBand)**
7. **Stampede (Dell InfiniBand)**
8. **Tianhe-1A (NUDT Proprietary)**
9. **Fermi (IBM BG/Q)**
10. **DARPA Trial Subset (IBM PERCS)**

- 7/10 systems use MPICH exclusively
- #6 One of the top 10 systems uses MPICH together with other MPI implementations
- #3 We are working with Fujitsu and U. Tokyo to help them support MPICH 3.0 on the K Computer (and its successor)
- #10 IBM has been working with us to get the PERCS platform to use MPICH (the system was just a little too early)
MPICH-3.0 (and MPI-3)

- MPICH-3.0 is the new MPICH2 :-)
  - Released mpich-3.0rc1 this morning!
  - Primary focus of this release is to support MPI-3
  - Other features are also included (such as support for native atomics with ARM-v7)

- A large number of MPI-3 features included
  - Non-blocking collectives
  - Improved MPI one-sided communication (RMA)
  - New Tools Interface
  - Shared memory communication support
  - (please see the MPI-3 BoF on Thursday for more details)
**MPICH 3.0 and Future Plans**

**3.0.x series**
- Full MPI-3 support
- Support for ARM-v7
- native atomics

- v3.0
  - v1.3.1
  - v1.3.2
  - v1.3.3

- v3.1 (preview)

**3.1.x series**
- Full Process FT
- Support for GPUs
- Active Messages
- Topology Functionality
- Support for Portals-4

**3.2.x series**
- Memory FT
- Dynamic Execution Environments
- MPI-3.1 support (?)
MPICH Fault Tolerance

- Fault Query Model
  - Errors propagated upstream
  - Global objects remain valid on faults
  - Ability to create new communicators/groups and continue execution

- Fault Propagation Model
  - Asynchronous Notification of faults to processes (global notification as well as “subscribe” model)

- Memory Fault Tolerance
  - Trapping memory errors and notifying application (e.g., global memory)
MPICH with GPUs

- Treating GPUs as first-class citizens
- MPI currently allows data to be moved from main memory to main memory
- We want to extend this to allow data movement from/to any memory
Dynamic Execution Environments with MPICH

- Ability to dynamically create and manage tasks
- Tasks as fine-grained threads
- Support within MPICH to efficiently support such models
- Support above MPICH for cleaner integration of dynamic execution environments (Charm++, FG-MPI)
Support for High-level Libraries/Languages

- A large effort to provide improved support for high-level libraries and languages using MPI-3 features

- We are currently focusing on three high-level libraries/languages
  - CoArray Fortran (CAF-2.0) in collaboration with John Mellor-Crummey
  - Chapel in collaboration with Brad Chamberlain
  - Charm++ in collaboration with Laxmikant Kale

- Other high-level libraries have expressed interest as well
  - X10
  - OpenSHMEM
Other Features Planned for MPICH

- Active Messages
- Topology Functionality
- Structural Changes
- Support for Portals 4
- Support for PMI-2
Thank You!

Web: http://www.mpich.org

More information on MPICH:

http://www.lmgtfy.com/?q=mpich