Fault Tolerance Working Group Update

Wesley Bland, Intel
MPI Forum BoF, Supercomputing ‘18
November 14, 2018
FTWG Overview

• Items already adopted for the next version of the MPI Standard (including the draft)
  • Improved error handler definitions
  • New predefined error handler
  • Non-catastrophic errors

• Items still being worked on
  • Splitting ULFM into smaller pieces to have better integration with other FT models
Improved Error Handling

• Clarified that MPI_ERRORS_ARE_FATAL should abort all processes in the job (connected processes)
  • This is what you’re currently experiencing anyway

• Changes where errors are raised when there’s nowhere else to do so
  • MPI_COMM_WORLD -> MPI_COMM_SELF
  • Backward incompatible change! (Though probably not significant)
New Error Raising Example

Old

MPI_alloc_mem(...);

Errhandler_fn(
    MPI_COMM_WORLD,
    MPI_ERR_MEM,
    ...
);

New

MPI_alloc_mem(...);

Errhandler_fn(
    MPI_COMM_SELF,
    MPI_ERR_MEM,
    ...
);

Only a problem when changing error handler of MPI_COMM_WORLD from default and not MPI_COMM_SELF. Important for localizing error recovery.
MPI_ERRORS_ABORT

- Added new predefined error handler which only causes processes in the affected communicator to abort

![Diagram]

**MPI_COMM_WORLD**

Error here

**NEW_COMM**

**MPI_ERRORS_ARE_FATAL**

 MPI_COMM_WORLD

Causes aborts everywhere

**NEW_COMM**
MPI_ERRORS_ABORT

• Added new predefined error handler which only causes processes in the affected communicator to abort
Non-Catastrophic Errors

- After an error is detected, the state of MPI is undefined.
- MPI should return as much information as possible about errors.
  - Gives users more control over how to handle errors.
  - If you really want to, you could construct a resilient point-to-point-only application on top of this change.
- This small change (along with the previous one), should actually provide enough error handling improvements to avoid application aborts during simple errors like resource exhaustion.
Still In Progress

• More General Fault Tolerance
  • Working to subdivide the ULFM proposal to allow other FT models (either in the MPI Standard or not) to live alongside it better.
  • Still hope to have a complete fault tolerance solution in MPI in the future.
  • May be more integrated with other long term topics things like “Sessions”.

Still	In	Progress
Call For Participation

• Always looking for more help (biggest bottleneck is participants’ time)
• Join the (semi) weekly conference calls
  • Wednesdays at noon Eastern US
  • Dial-in info sent to mailing list
• Contact the WG
  • mpiwg-ft@lists.mpi-forum.org
• Contact me directly
  • wesley.bland@intel.com
• See our progress (and all other WGs)
  • https://www.mpi-forum.org/mpi-40/
  • https://www.github.com/mpiwg-ft/ft-issues