ExaMPI

Derek Schafer

SimCenter: Center of Excellence for
Applied Computational Science and Engineering
The University of Tennessee at Chattanooga

ExaMPI – What is it?

- Fully progressive, modular C++ MPI implementation
- Implements 3.1 (with support for key MPI 4.0 features)
- Research vehicle, not a replacement for mature MPI implementations

- ExaMPI Paper:
- Skjellum, A., Rüfenacht, M., Sultana, N., Schafer, D., Laguna, I., & Mohror, K. (2019, September). ExaMPI: A Modern Design and Implementation to Accelerate Message Passing Interface Innovation. In Latin American High Performance Computing Conference (pp. 153-169). Springer, Cham.

ExaMPI – What is it not?

- Not a full-featured product; not trying to “boil the ocean”
- Not all MPI features -- yet
  - Some by choice (e.g., support for MPI_ANY_ SOURCE in point to point)
  - Relevance over coverage (e.g., MPI one-sided)
- Where we put our effort:
  - Persistent first strategy
  - Agile prototyping
  - Focus on baseline + MPI 4.0 extensions
  - Learning and implementing best C++ practices

ExaMPI – Why use it?

- Supports quick prototyping of new ideas
- Modern C++ source base allows tractable experimentation
- Not a cumbersome as tweaking big MPI implementations
  - Sandbox for students/researchers
  - Quicker feedback/help
- Use as vehicle to demonstrate new performance, new abstractions, that could:
  - Be proposed as MPI-5+ additions
  - Feed back into production MPI implementations

Noteworthy ExaMPI Features

- Strong progress engine
  - Weak progress engine coded, but off by default
  - Working on means to toggle between, or have multiple simultaneously
- User-Level Schedules
- Algorithms
  - Designed to be persistent
  - “Schedules lite”
  - Simple way to describe collective pattern

Summary of Prospective Apps (I)

- CLAMR (26 of 38)
  - Mostly MPI File functions
  - ‘v’ collectives
- COMB (23 of 31)
  - Packing functions
  - Advanced MPI datatype functions
  - Cartesian topology functions
- SNAP (6 of 7)
  - MPI Cart sub

CLAMR	68.42%
Comb	74.19%
ExaMPM	100.00%
Fiesta	45.00%
Quicksilver	85.71%
SNAP	85.71%
miniAero	100.00%

Summary of Prospective Apps (II)

- FIESTA (36 of 80)
  - MPI File and MPI Info functions take 2/3rds
  - MPI Types and ‘v’ collectives take most of remainder
- Quicksilver (24 of 28)
  - MPI Cancel related functions, MPI Scan
- ExaMPM (5 of 5) & miniAreo (9 of 9)
  - Should be fully supported
  - May require additional features/constants

CLAMR	68.42%
Comb	74.19%
ExaMPM	100.00%
Fiesta	45.00%
Quicksilver	85.71%
SNAP	85.71%
miniAero	100.00%

Application Support – Next Steps

- In general:
  - More Datatype support
    - Add support for packing datatypes
    - Test new optimizations
  - ‘v’ Collectives
  - More cartesian topology support
- Partitioned communication support
- Extra MPI features may be needed based on selected apps
- Test the proxy applications identified

Other Research Areas

- Collectives:
  - Smarter algorithms
  - Neighborhood collectives
  - Partitioned collectives (in the future)
  - New collectives that aren't well described in MPI
- GPU support
- Full C++ MPI interface
- And other optimizations!

Questions