Scalable Out-of-core OpenSHMEM Library for HPC

Antonio Gómez-Iglesias, Jérôme Vienne, Khaled Hamidouche,
Christopher S. Simmons, William L. Barth, Dhabaleswar K. Panda

Texas Advanced Computing Center
Ohio State University
The University of Texas at Austin

OpenSHMEM 2015
July, 2015

Out-of-core Methods

- Applications with large memory requirements → normal nodes are not enough
- Offload data onto files
  - I/O is slow
  - Need for efficiently storing/retrieving data from disk
- Popular method in many applications

Problems of Out-of-core Methods

- I/O becomes a bottleneck at large scale
- Model not well suited for distributed file systems
  - Very high load in the servers
  - Possible crashes on the file system

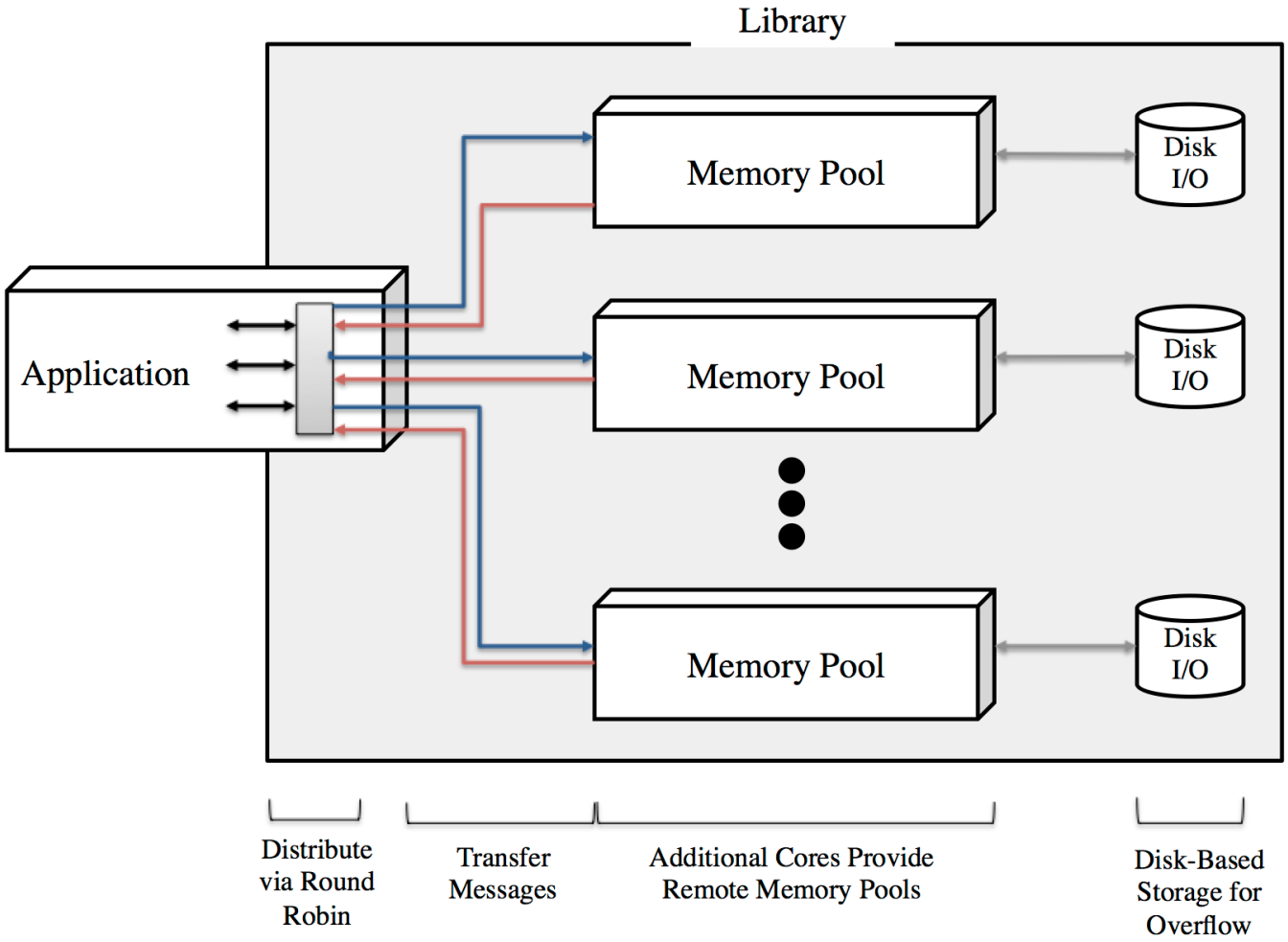
A Distributed Out-of-core Method

- Large clusters, with many nodes
- Offload data to nodes, not files
- Each node has a local disk $\rightarrow$ use it
- Each node has memory, even better than local disk

Distributed Out-of-core

- Yes: only memory is used in the nodes
- Nodes do not perform calculations
- Is this a waste of resources?

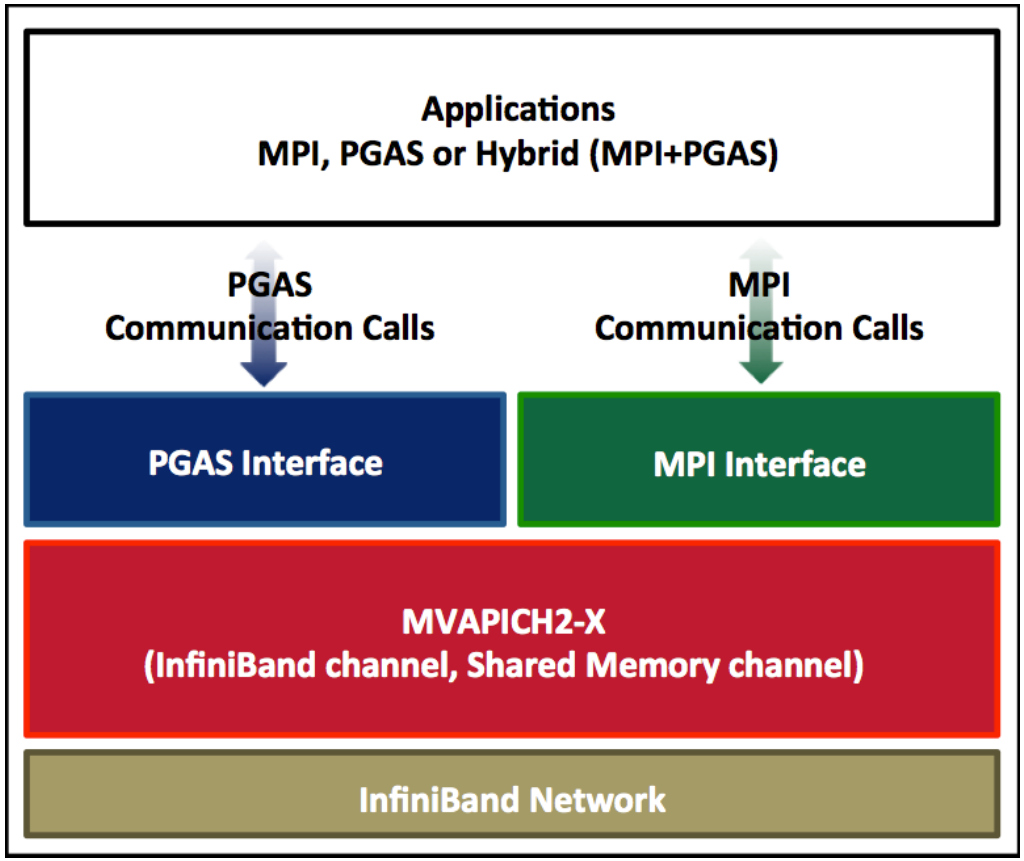
GRVY Model


Library
Memory Pool
Disk I/O
Application
Memory Pool
Disk I/O
Memory Pool
Disk I/O
Distribute via Round Robin
Transfer Messages
Additional Cores Provide Remote Memory Pools
Disk-Based Storage for Overflow

MVAPICH2 Software

- High Performance open-source MPI Library for InfiniBand, 10Gig/iWARP, and RDMA over Converged Enhanced Ethernet (RoCE)
  - MVAPICH (MPI-1), MVAPICH2 (MPI-2.2 and MPI-3.0), Available since 2002
  - MVAPICH2-X (MPI + PGAS), Available since 2012
  - Support for GPGPUs (MVAPICH2-GDR) and MIC (MVAPICH2-MIC), Available since 2014
  - Support for Virtualization (MVAPICH2-Virt), Available since 2015
  - Used by more than 2,425 organizations in 75 countries
  - Empowering many TOP500 clusters
  - Available with software stacks of many IB, HSE, and server vendors
  - http://mvapich.cse.ohio-state.edu
- System-X from Virginia Tech (3rd in Nov 2003) $\rightarrow$ Stampede at TACC (8th in Jun15)

MVAPICH2-X


Applications
MPI, PGAS or Hybrid (MPI+PGAS)
PGAS
Communication Calls
MPI
Communication Calls
PGAS Interface
MPI Interface
MVAPICH2-X
(InfiniBand channel, Shared Memory channel)
InfiniBand Network

MPI Implementation

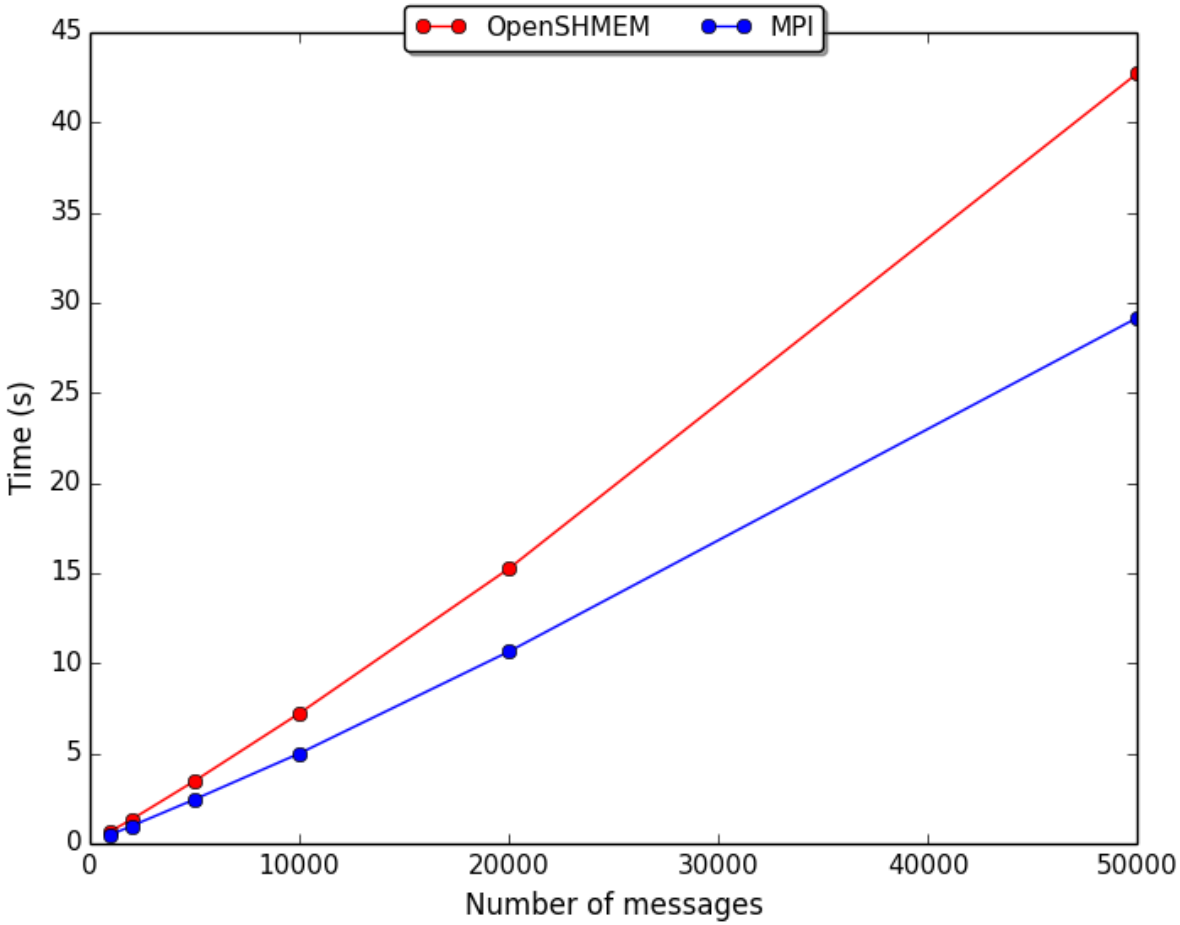
- Master-slave model
- Point-to-point communication
- High level of synchronization

OpenSHMEM Synchronization

Locks

- Easy implementation with *shmem_set_lock* & *shmem_clear_lock*
- Set a lock when writing shared data, clear it once the data has been written
- Only *shmem_put* used

Locks. Results

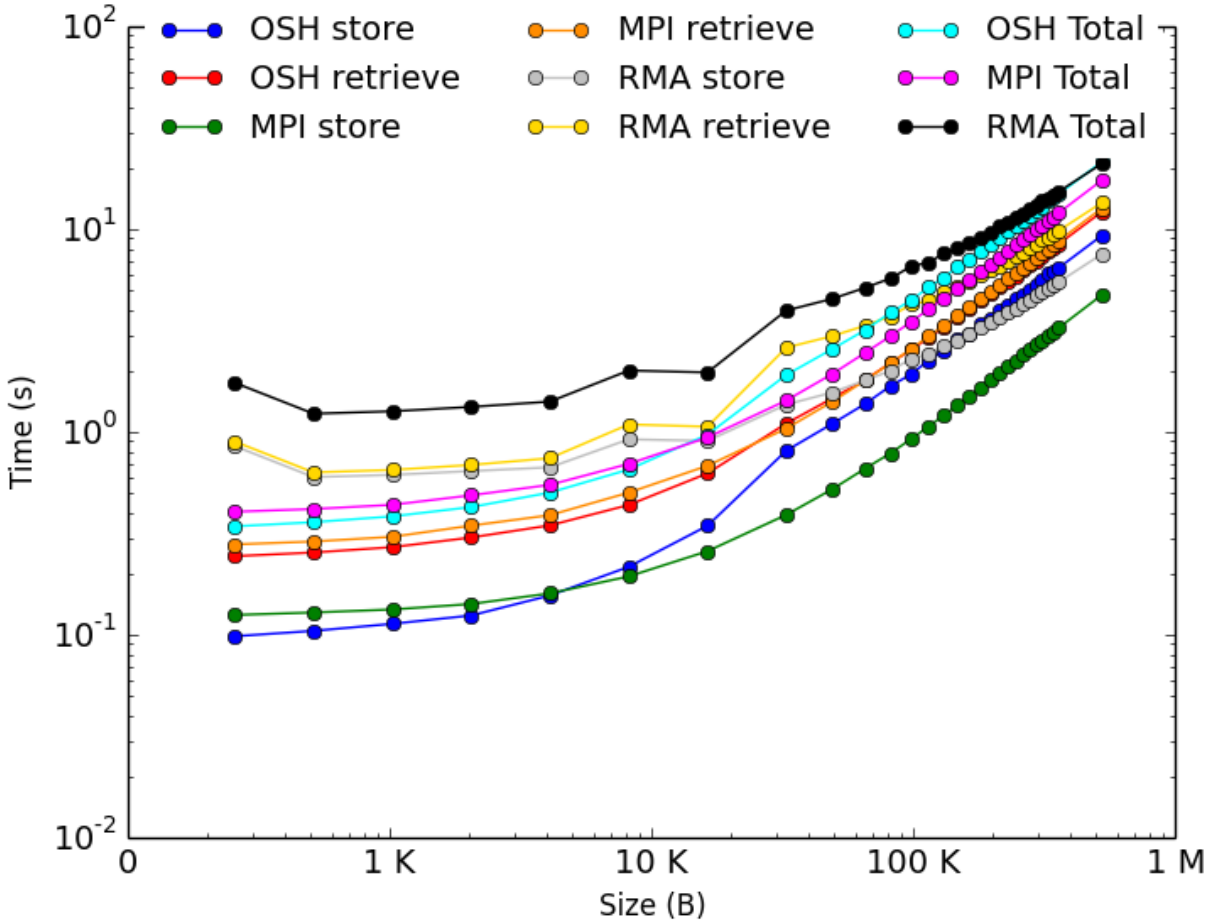


OpenSHMEM Synchronization

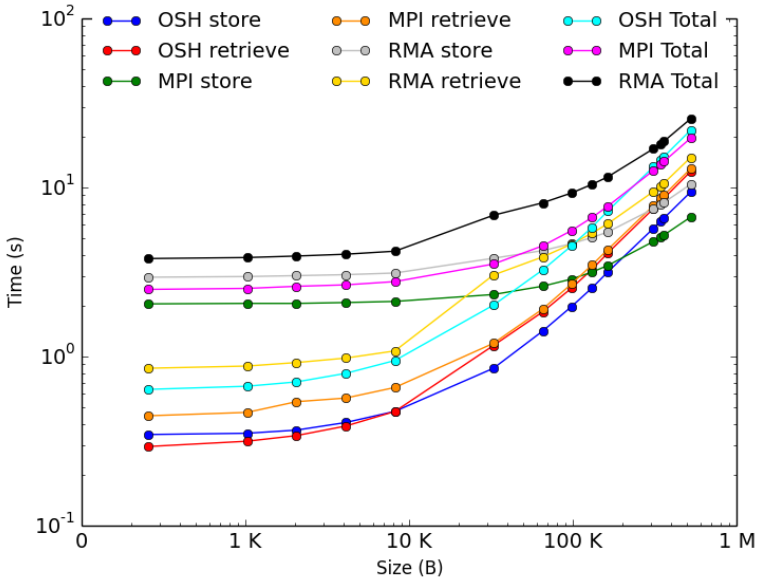
Active Polling

- Instead of using locks, the processes synchronize using *shmem_wait*
- Faster implementation than locks
- Larger change in the code

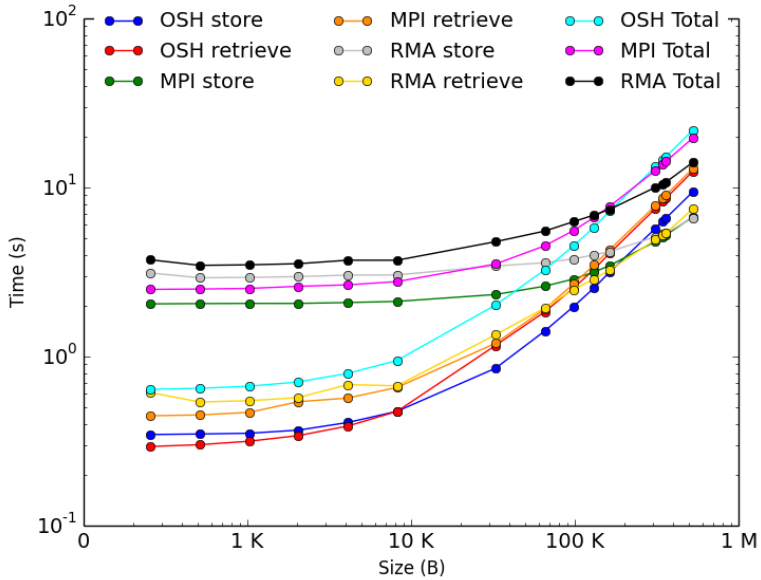
Active Polling. 128 processes



Locks. 2048-4096 processes



2048 processes



4096 processes

Conclusions

- OpenSHMEM is a good option for implementing an out-of-core library
- *Easy* porting
- Important to choose the best synchronization model
- More work being done

Questions?

agomez@tacc.utexas.edu