



# Open MPI State of the Union XII Community Meeting SC19

Jeff  
Squyres



George  
Bosilca



Brian  
Barrett



Josh  
Hursey



# BOF Feedback Form

<https://www.open-mpi.org/sc19/>



# EuroMPI/USA'20

September 21-24, 2020  
Austin, TX



## Important Dates

**Full paper Submission:** April 22, 2020. (AOE)

**Notification of acceptance (Paper):** July 1, 2020.

**Workshops and Tutorials Submission:** February 24, 2020. (AOE)

<https://eurompi.github.io/>



# Open MPI versioning

Quick review

# Open MPI versioning

- Open MPI uses “**A.B.C**” version number triple
- Each number has a specific meaning:
  - A** This number changes when backwards compatibility breaks
  - B** This number changes when new features are added
  - C** This number changes for all other releases

# Definition

- Open MPI vY is backwards compatible with Open MPI vX (where Y>X) if:
  - Users can compile a correct MPI / OSHMEM program with vX
  - Run it with the same CLI options and MCA parameters using vX or vY
  - The job executes correctly

# What does that encompass?

- “Backwards compatibility” covers several areas:
  - Binary compatibility, specifically the MPI / OSHMEM API ABI
  - MPI / OSHMEM run time system
  - `mpirun` / `oshrun` CLI options
  - MCA parameter names / values / meanings



## Version Roadmaps

# v3.0.x (Prior stable)

- Release managers
  - Brian Barrett, AWS
  - Jeff Squyres, Cisco
- Immanent release: v3.0.5
  - November 2019
  - *Maybe one more release...?*
- Maintenance mode
  - No new features for life of series
- Major features
  - `MPI_THREAD_MULTIPLE` support by default



# v3.1.x (Prior stable)

- Release managers
  - Brian Barrett, AWS
  - Jeff Squyres, Cisco
- Immanent release: v3.1.5
  - November 2019
  - *Maybe one more release...?*
- Maintenance mode
  - No new features for life of series
- Many usability features over 3.0.x



# v4.0.x (Current stable)

- Release managers
  - Howard Pritchard,  
Los Alamos National Lab
  - Geoff Paulsen, IBM



- Current release: v4.0.2
  - November 2019
- Lots of bug fixes and performance improvements
- **Big changes:**
  1. Removed MPI-1 APIs not prototyped in mpi.h by default
  2. IB support now via UCX
  3. ABI compatible with 3.x
  4. MPIR usage deprecated

**REMINDER**

# Deprecation notice: MPIR

- MPIR interface is used internally to launch / attach tools and debuggers
- The maintainer for Open MPI's MPIR is retiring!
- Initially announced at SC'17 BOF:
  - Unless someone else takes over, this is the plan:
    - Deprecation notice in NEWS in early CY2018
    - User runtime warnings in mid/late CY2019 (v4.0.0)
    - Removal in CY2020 (replaced by PMIx-based tool support)

# v5.0.x (Future)

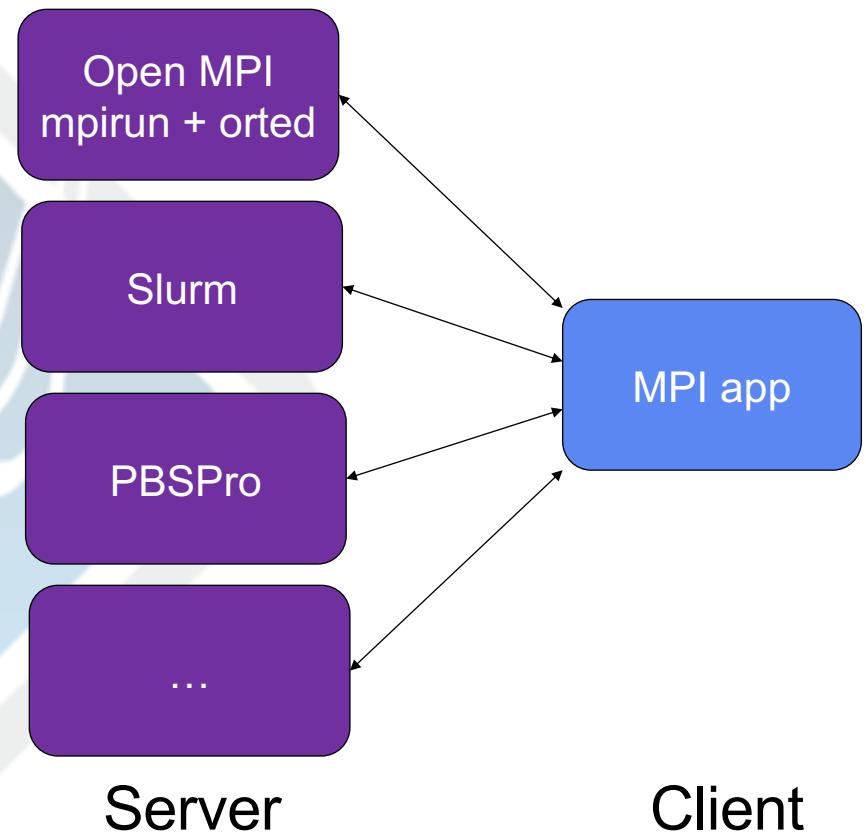
- Still under active discussion
  - <https://github.com/open-mpi/ompi/wiki/5.0.x-FeatureList>
- Some potentially contentious issues:
  - Remove the openib BTL
  - Remove C++ bindings
  - Remove support for ancient gfortran
  - Remove MPI-1 / MPI-2 deleted interfaces
  - Remove MPIR
  - Break ABI compatibility with v3.x / v4.x
  - Make PMIx a first-class internal API
  - Remove support for PMI-1, PMI-2

# The role of PMIx

- PMIx has effectively replaced much of Open MPI's runtime system (ORTE)
  - Apps directly depend on PMIx
- ORTE still exists
  - Only used with mpirun / mpiexec (in any environment)
  - Typical for ssh-based launching for unmanaged environments
- Open MPI still contains an embedded copy of PMIX
  - But also supports compiling against an external PMIx

# How to debug PMIx issues?

- PMIx has two sides:
  - Server (infrastructure)
  - Client (MPI apps)



# PMIx logging variables

- DIR is “client” or “server”
- This one gives general information:
  - PMIX\_MCA\_pmix\_DIR\_base\_verbose
- These give specific types of information:
  - PMIX\_MCA\_pmix\_DIR\_get\_verbose
  - PMIX\_MCA\_pmix\_DIR\_connect\_verbose
  - PMIX\_MCA\_pmix\_DIR\_fence\_verbose
  - PMIX\_MCA\_pmix\_DIR\_pub\_verbose
  - PMIX\_MCA\_pmix\_DIR\_spawn\_verbose
  - PMIX\_MCA\_pmix\_DIR\_event\_verbose
  - PMIX\_MCA\_pmix\_DIR\_iof\_verbose
- Set via environment variable

```
mpirun -x \
PMIX_MCA_pmix_client_base_verbose=100 ...
```
- Or PMIx site-wide config  
`$prefix/etc/pmix-mca-params.conf`



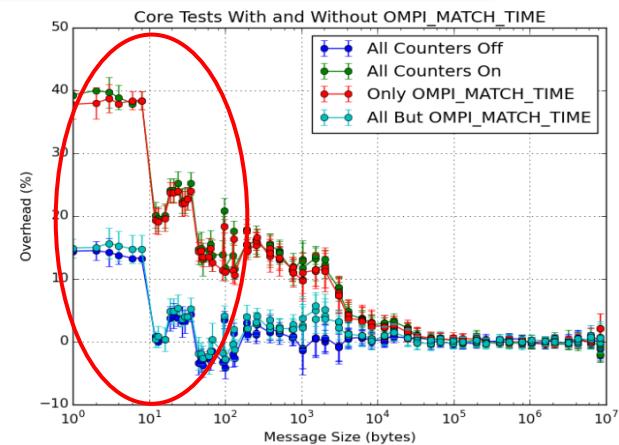
# UTK Open MPI Activities

George Bosilca  
University of Tennessee



# SPC: MPI\_T Software Performance Counters

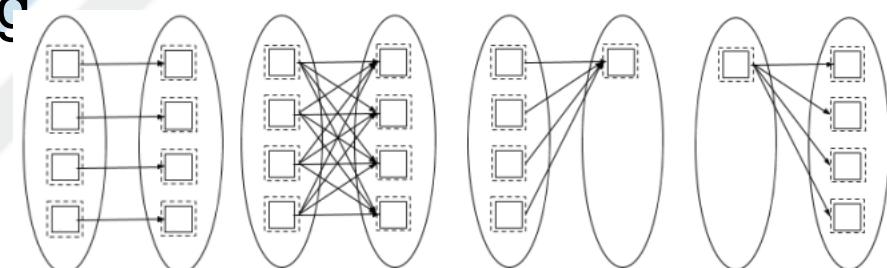
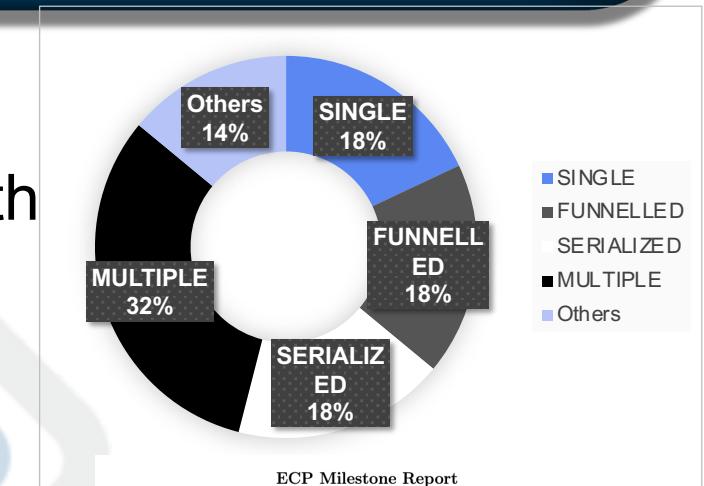
- Similar to PAPI counters but exposing internal information not available through other means
  - Out-of-sequence messages, time to match, number of unexpected, instant bandwidth, collective bins
- Can be configured to expose counters into a jobid shared file (XML + binary)
  - PMIx plugins to gather or monitor online the state of the job
  - Can detect deadlocks or pinpoint slowdowns in different metrics (such as message rate or bandwidth)



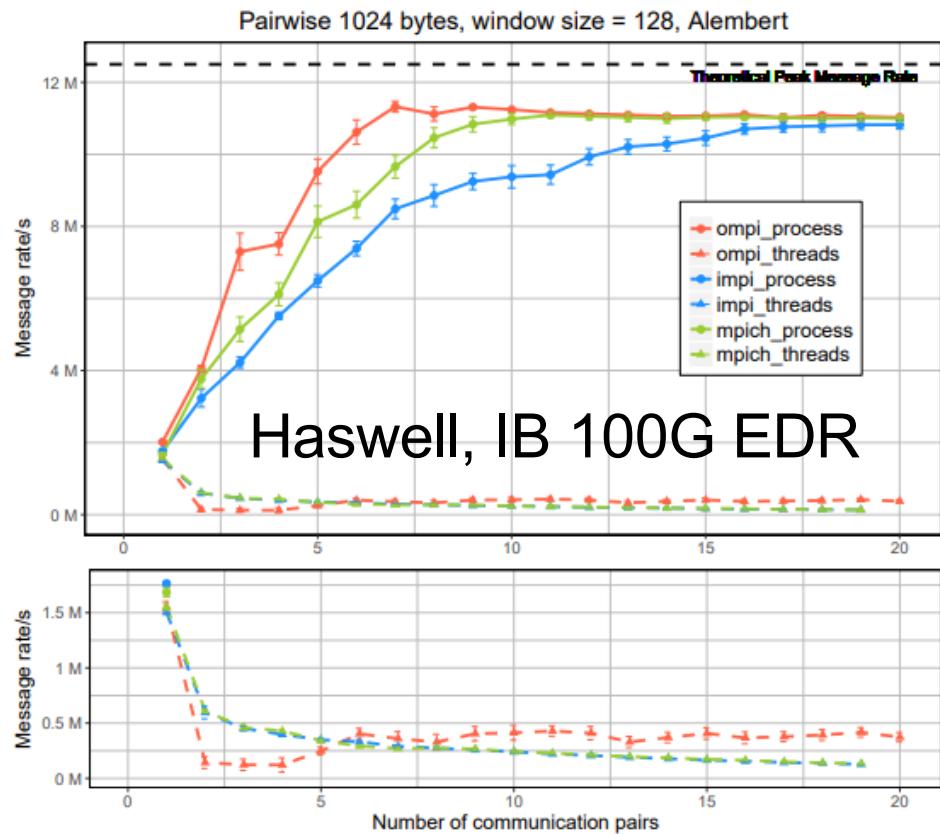
# Threading



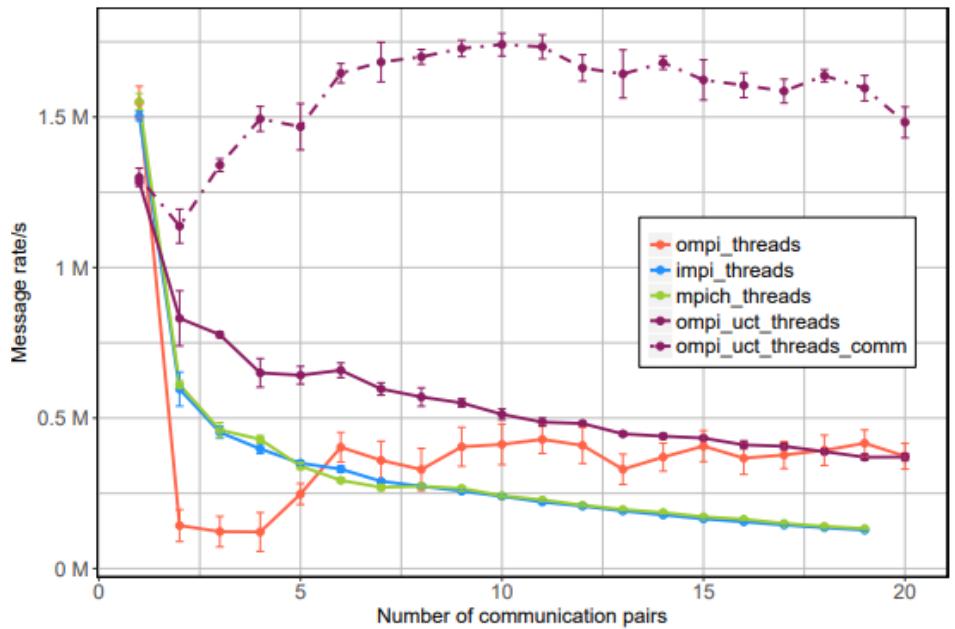
- Released Multirate benchmark
  - Different communications patterns with different workloads
  - Between different entities (threads or processes)
  - Multiple communicators
  - Enforced thread/process binding



# Threading

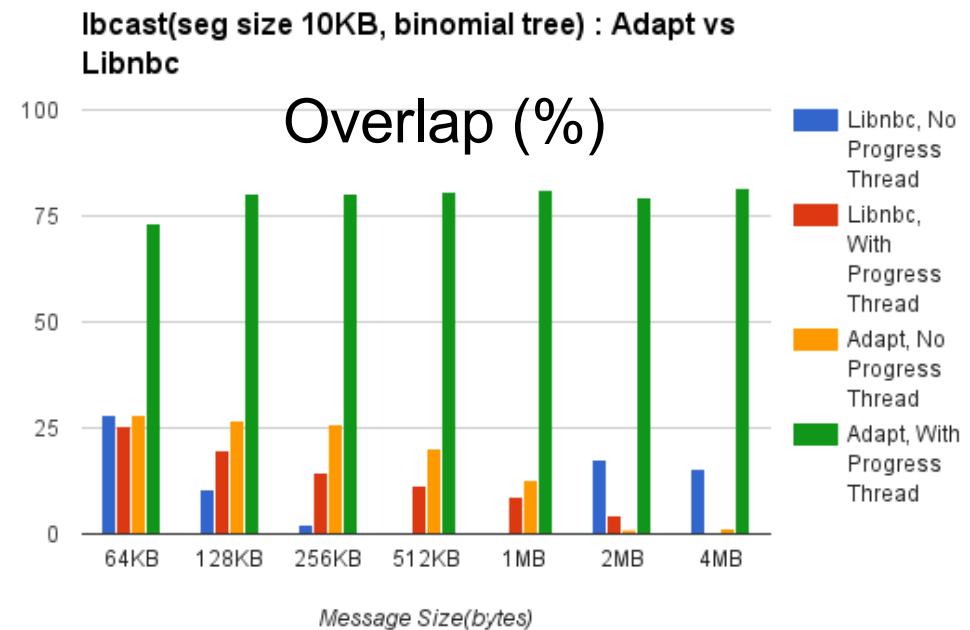


- Houston we have a problem !



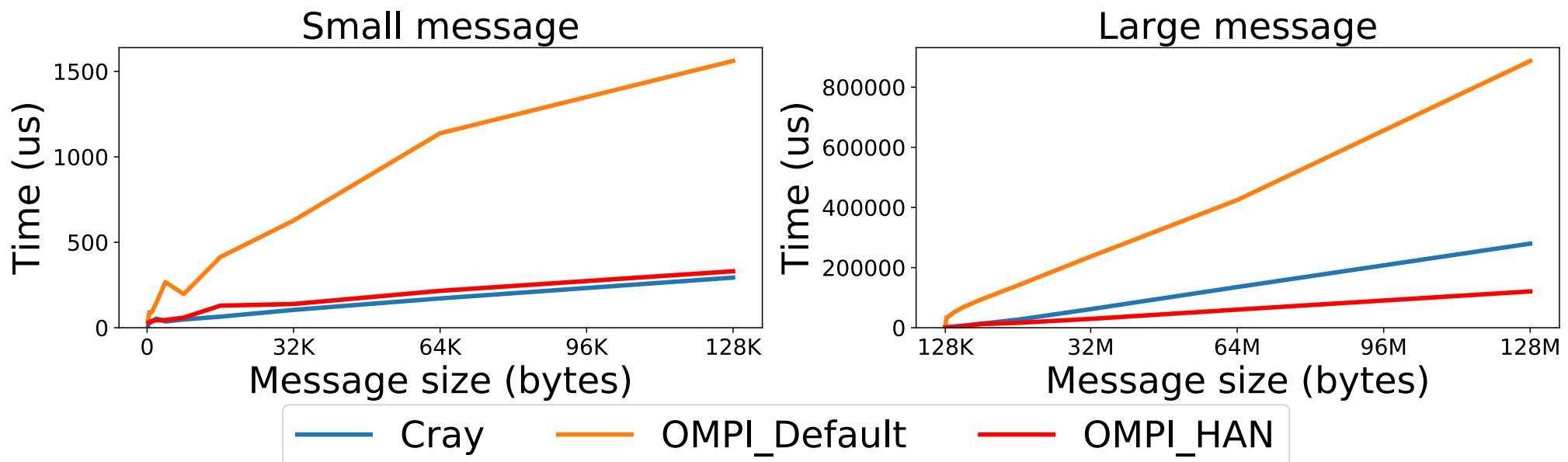
# HAN: Adaptive Collective communications framework

- More complex architectures demand more complex collective frameworks
  - Time for a refresh of the tuned collective framework
- Architecture aware (ADAPT)
  - Hybrid Architecture
- Shared Memory (SM<sup>2</sup>)
  - One-sided communications
- Noise Reduction (FUTURE)
- Overlap needed for non-blocking collectives



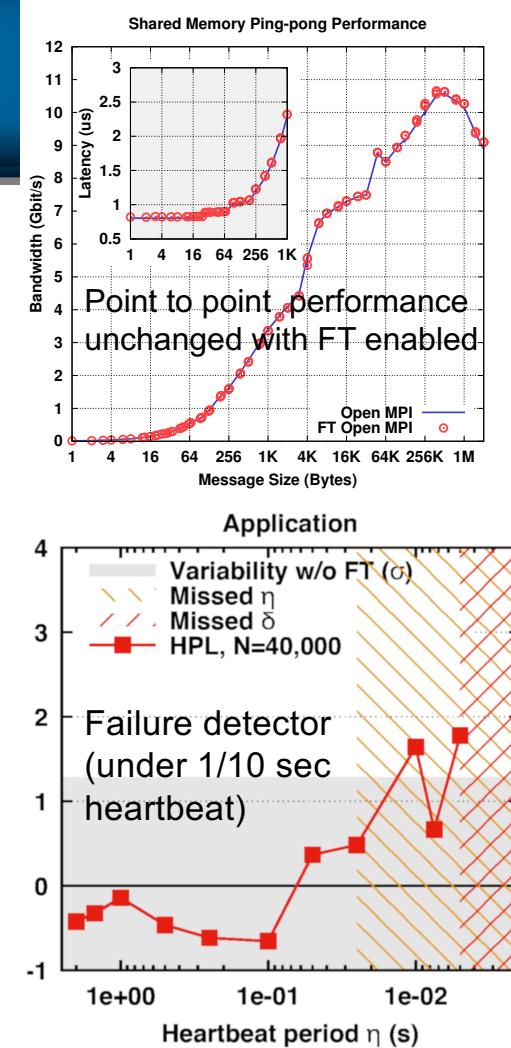
# HAN: Adaptive Collective communications framework

MPI\_Bcast on 4K processes on Shaheen



# Resilience - User Level Failure Mitigation (ULFM)

- Move the underlying resilient mechanisms outside ULFM/OMPI
  - Failure detector and reliable broadcast in PPRTE
  - Used in OMPI ULFM and SUNY OpenSHMEM
- ULFM 4.0.2u1 released
  - Based on OMPI 4.0.2 (will remain in sync)
- ULFM master follows OMPI master
  - Long transition to integrate ULFM in OMPI master
- Scalable fault tolerant algorithms demonstrated in practice for revoke, agreement, and failure detection (SC'14, EuroMPI'15, SC'15, SC'16)





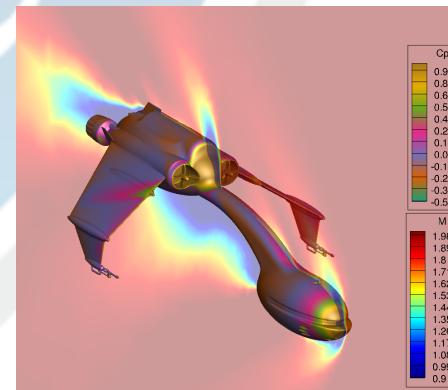
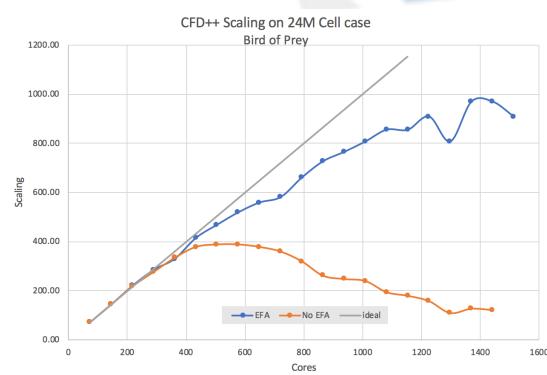
# Open MPI and AWS

Brian Barrett



# Elastic Fabric Adapter (EFA)

- EFA is AWS's HPC/ML-oriented NIC
  - First NIC requiring Libfabric with Open MPI
  - Lots of Libfabric development
  - Lots of Open MPI OFI MTL testing



# General Improvements

- TCP BTL
  - Patches outstanding to fix multi-NIC matching
  - Should make containers and non-routable NICs more predictable
- AWS Batch Integration
  - Run Open MPI applications in cloud batch scheduler

# Next Year Plans

- Continue performance work
  - OFI MTL
  - Collectives
- Lots and lots of application testing
- You tell us!
  - AWS Forums – HPC section



Ralph Castain

# Fifteen<sup>+</sup> Years of Open MPI ...and Retirement



THANK  
YOU!



# Los Alamos Open MPI Activities



# Los Alamos - Current Work

- MPI Sessions prototype based on Open MPI:
  - <https://github.com/hpc/ompi/hpc/sessions>
  - [https://github.com/hppritchha/mpi\\_sessions\\_tests](https://github.com/hppritchha/mpi_sessions_tests)
  - Investigating use in DASK
- Adding support for Argobots and Qthreads:
  - <https://github.com/open-mpi/ompi/pull/6578>
- Release manager role for 4.0.x

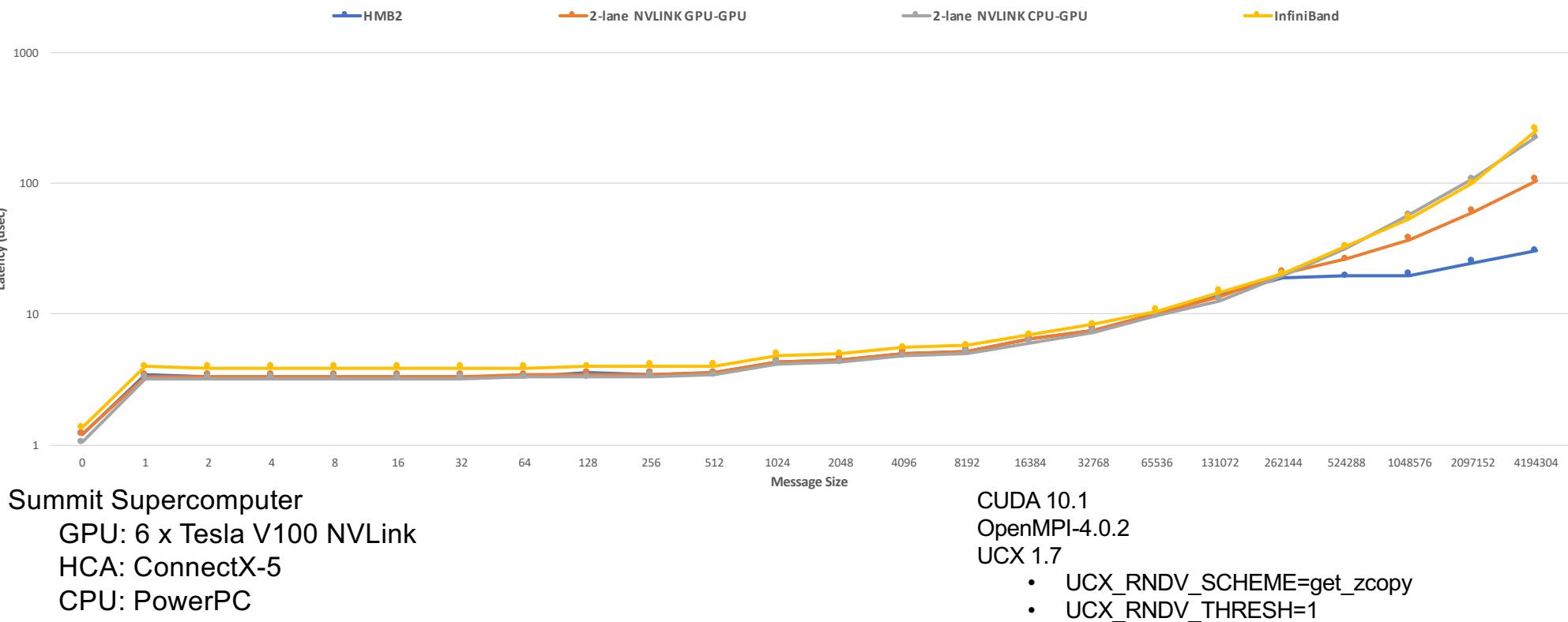


# Open MPI / UCX: A Performance Evaluation of CUDA support on Summit Supercomputer

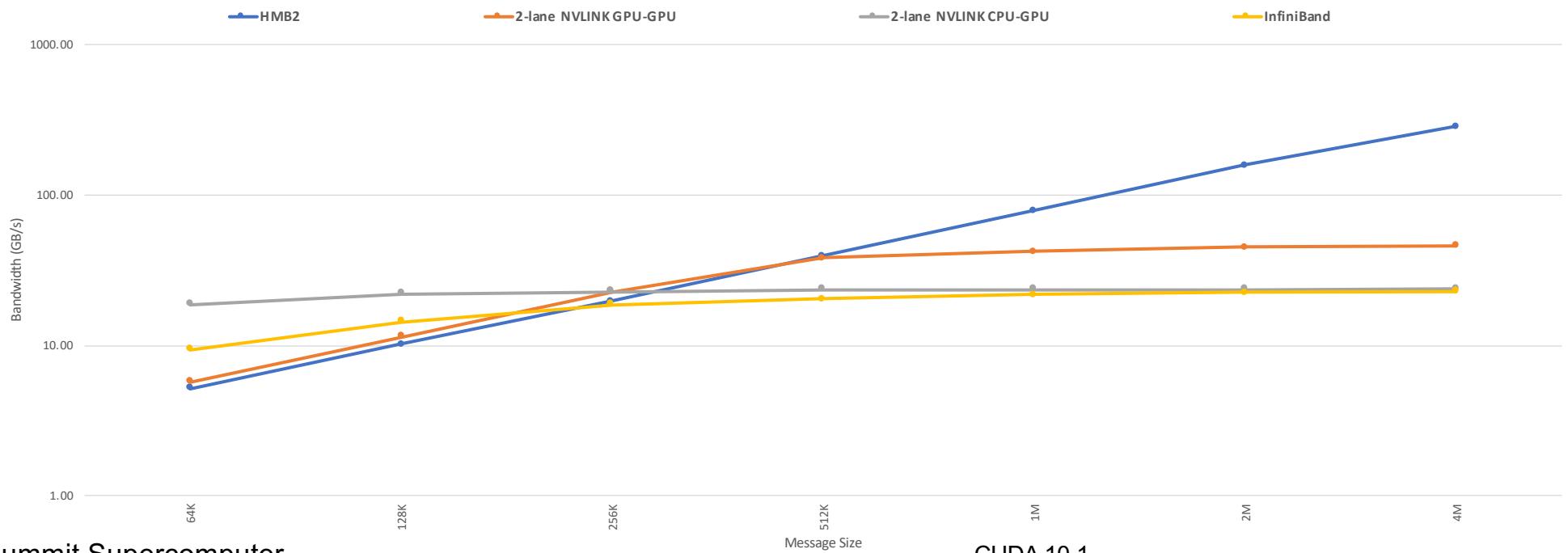
Mellanox Technologies



# Summit OSU MPI Latency Benchmark



# Summit OSU MPI Bandwidth Benchmark



Summit Supercomputer

GPU: 6 x Tesla V100 NVLink

HCA: ConnectX-5

CPU: PowerPC

CUDA 10.1

OpenMPI-4.0.2

UCX 1.7

- UCX\_RNDV\_SCHEME=get\_zcopy
- UCX\_RNDV\_THRESH=1

# Performance Summary

	GPU HBM2	2 Lane NVLINK GPU-GPU	2-Lane NVLINK CPU-GPU	IB EDR x2
Theoretical Peak BW	900 GB/s	50 GB/s	50 GB/s	25 GB/s
Available Peak BW	723.97 GB/s	46.88 GB/s	46 GB/s	23.84 GB/s
UCX Peak BW	349.6 GB/s	45.7 GB/s	23.7 GB/s	22.7 GB/s
% Peak	48.3	97.5	51.5	95.2



# IBM Spectrum MPI

Joshua Hursey  
IBM





# Delivering Robust & Sustained High Performance for Scalable MPI Applications

## Accelerated & Enhanced MPI Point-to-Point

- Driving maximum performance from POWER9, InfiniBand, and GPU hardware.
- Supports direct transfer of GPU buffers between GPUs and across the InfiniBand network.

## Dynamic & Optimized MPI Collectives

- Best algorithm selected per call at runtime.
- Includes Power optimized and hardware accelerated (e.g., SHARP) algorithms.

## Usability Features Targeting Installation, Startup, Debugging, and Profiling

- Scalable to thousands of nodes and nearly a million processes!

## Integration with IBM solutions such as LSF, ESSL, and Spectrum Scale

Built on the open source Open MPI project with **IBM value add** and **IBM service and support**

### IBM Messaging Software based on Open MPI

#### IBM added functionality

Collective Library, PAMI Network Driver, Power Architecture Tuning, Cluster Test Tools, Packaging for ISV/OEM models, GPU optimizations, Integrated Performance Analysis Tools and more...

### Open MPI

IBM Mellanox Nvidia ...

**Spectrum MPI 10.3 is based on Open MPI 4.0.x with PMIx 3.1.x**

Spectrum MPI  
Community Edition  
Available In Dec.!



Nov 2019  
Top500

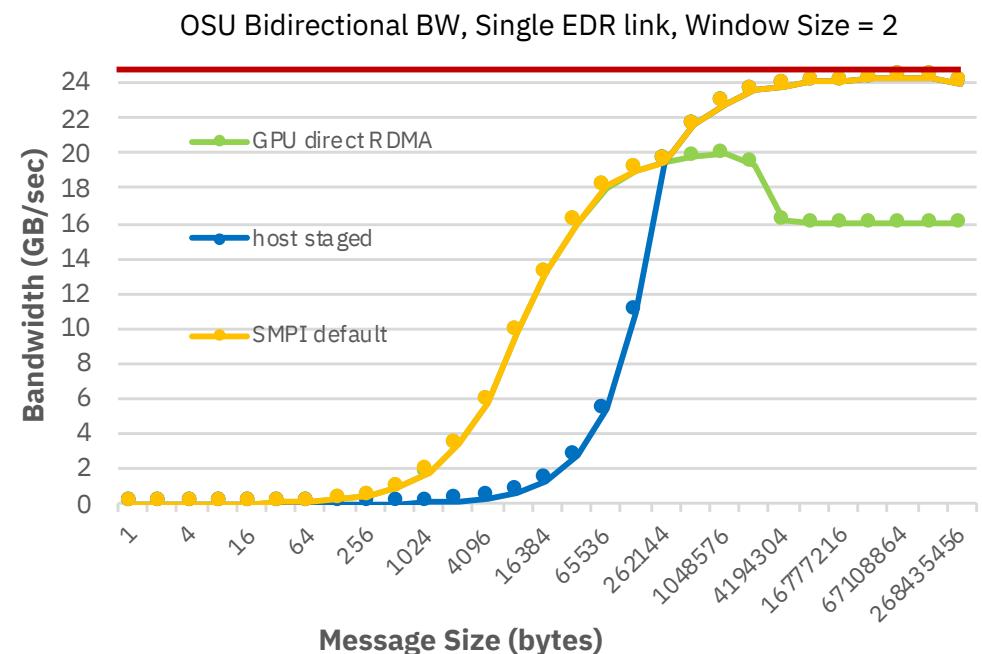
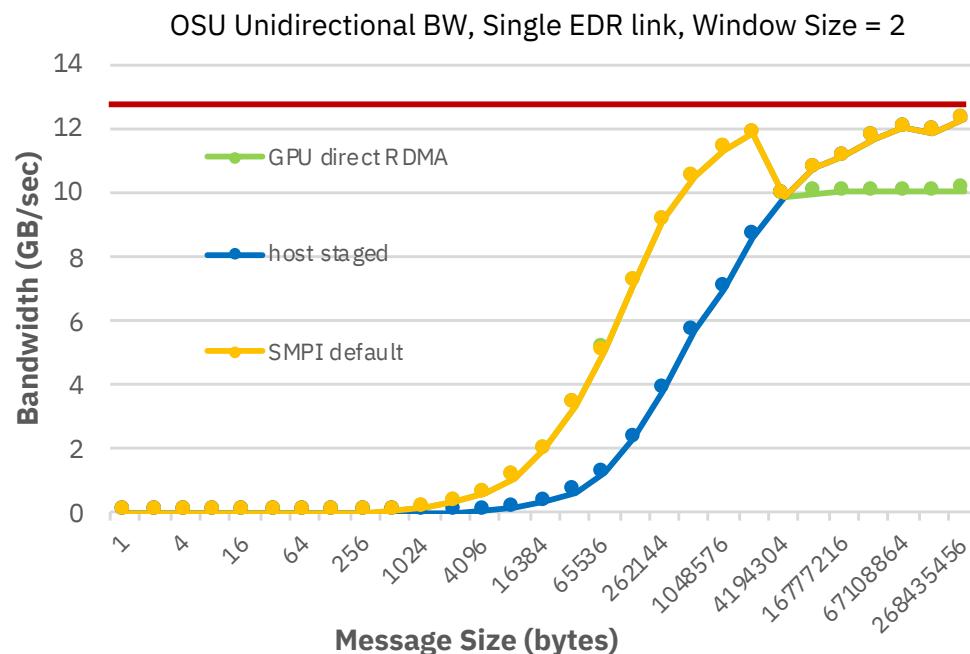
# Delivering Robust & Sustained High Performance for Scalable MPI Applications

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	DOE/SC/Oak Ridge National Laboratory United States	Summit - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband IBM	2,414,592	148,600.0	200,794.9	10,096
2	DOE/NNSA/LLNL United States	Sierra - IBM Power System S922LC, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband IBM / NVIDIA / Mellanox	1,572,480	94,640.0	125,712.0	7,438
10	DOE/NNSA/LLNL United States	Lassen - IBM Power System S922LC, IBM POWER9 22C 3.1GHz, Dual-rail Mellanox EDR Infiniband, NVIDIA Tesla V100 IBM / NVIDIA / Mellanox	288,288	18,200.0	23,047.2	
11	Total Exploration Production France	PANGEA III - IBM Power System AC922, IBM POWER9 18C 3.45GHz, Dual-rail Mellanox EDR Infiniband, NVIDIA Volta GV100 IBM	291,024	17,860.0	25,025.8	1,367

**CORAL**  
COLLABORATION  
OAK RIDGE • ARGONNE • LIVERMORE



# MPI Point-to-Point Enhancements in PAMI



When transferring GPU buffers across servers, **Spectrum MPI automatically switches** between GPUDirect RDMA and host staged protocols to **deliver the best bandwidth performance for all message sizes**.

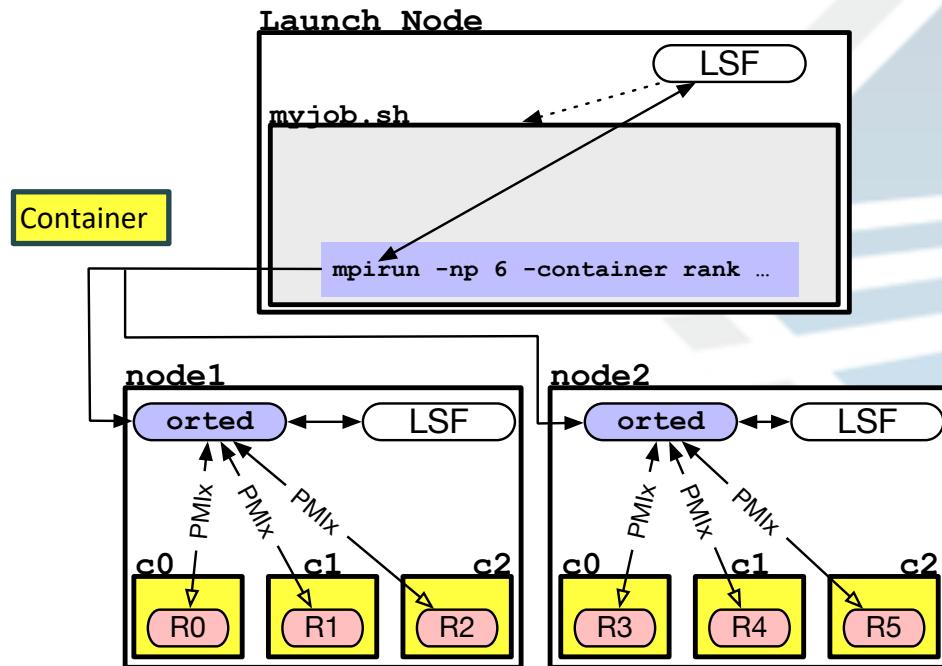


# Container Ready Supporting Applications on Bare Metal & Private/Public Cloud

**Two different container modes commonly used in HPC environments**

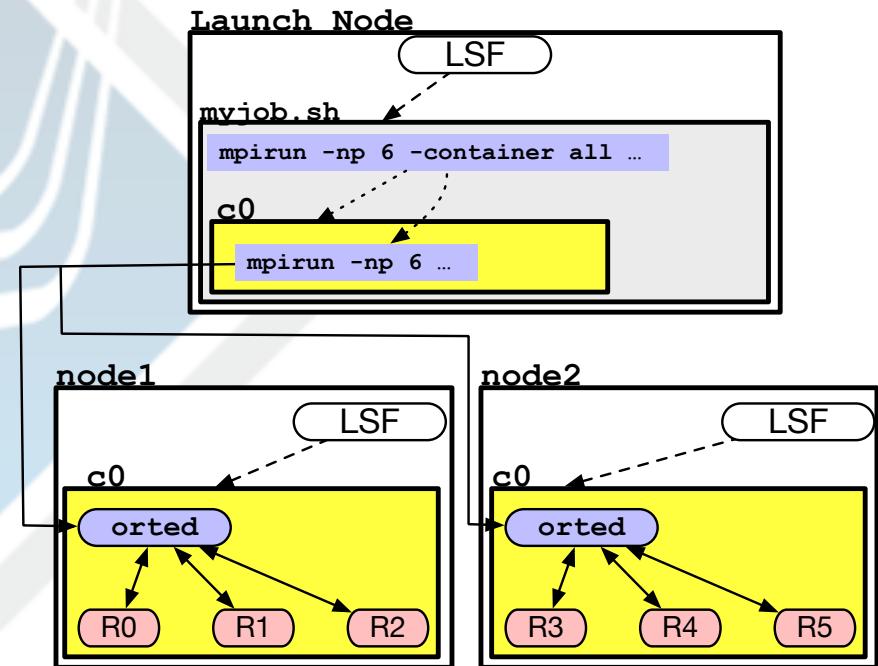
## Rank Contained mode

One container per application process



## Fully Contained mode

One container per node



# Where do we need help?

- Code
  - Any bug that bothers you
  - Any feature that you can add
- ***User documentation***
- Testing (CI, nightly)
- Usability
- Release engineering

We ❤️ ↗



Come join us!