



Open MPI State of the Union XIII.5 ECP Community BOFs 2021

Jeff
Squyres



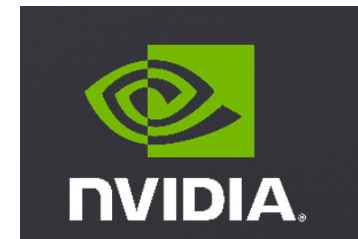
George
Bosilca



Howard
Pritchard



Josh
Ladd



Also see...

Deep dive into Open MPI, PMIx, PRRTE (video recording and slides):

The ABCs of Open MPI

Decoding the Alphabet Soup of the Modern HPC Ecosystem

Presented in conjunction with the EasyBuild community

<https://www.open-mpi.org/video/?category=general>



Version Roadmaps

v4.0.x (Stable, still supported)

- Release managers

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



- Current release v4.0.5 (Aug 2020)
- v4.0.6 will release soon
- Recent bug fixes
 - Fix problem when building Open MPI under macOS Big Sur
 - Fix issue using Flux PMI and UCX
 - Update internal PMIx to 3.2.3 to address some MPI_Comm_spawn issues
 - Fix an issue with MPIR_proctable visibility for debuggers

v4.1.x (Stable, supported)

- Release managers
 - Raghu Raja, AWS
 - Jeff Squyres, Cisco



- Current release v4.1.0 (Dec 2020)
- v4.1.1 will release soon
- Recent bug fixes
 - Apple M1 silicon support
 - Restore OB1 default functionality
 - Improved AVX detection
 - Fixed MPIR breakpoint detection
 - Fixed several build/link issues
 - Fixed multiple SLURM issues
 - Fixed MPI IO progression issues
 - Improved HAN collectives

How do I enable ADAPT and/or HAN?

Public service
reminder for v4.1
from the EasyBuild
presentation

- Either of two different ways:
 - a. Set the MCA priority of adapt and/or han to 100. For example:
`$ mpirun --mca coll_adapt_priority 100 --mca coll_han_priority 100 ...`
 - b. Include adapt and/or han in the coll MCA parameter. For example:
`$ mpirun --mca coll han,adapt,tuned,sm,basic ...`
- Do I have to enable *both* ADAPT and HAN?
 - a. No.
 - b. Specifically: you can use them independently or together.

PLEASE TEST WITH REAL APPS!

V5.0.x (soon)

- Release managers

- Austen Lauria, IBM
- Geoffrey Paulsen, IBM
- Joshua Ladd, NVIDIA

- Branched in March 2021
- Initial release planned for summer 2021
- Nightly snapshots:
 - <https://www.open-mpi.org/nightly/v5.0.x>



v5.0.0 (soon)

- (Some of the) Major changes:
 - Now using PRRTE
 - Upgraded to PMIx v4.x
 - “vader” has been renamed “sm” (shared memory)
 - Deleted the OpenIB BTL
 - IB/RoCE now supported via UCX
 - Removed MPI C++ bindings
 - Removed MPIR interface
 - Revamped documentation

1. Quick start
2. Getting Help
3. General notes
4. Building and installing
5. Version numbers and binary compatibility
6. Validating your installation
7. Open MPI API extensions
8. Open MPI Java bindings
9. Examples
10. Frequently Asked Questions (FAQ)
11. Developer's guide
12. Internal frameworks
13. License

Note: this TOC subject to change before release!

Open MPI

The Open MPI Project is an open source implementation of the [Message Passing Interface \(MPI\) specification](#) that is developed and maintained by a consortium of academic, research, and industry partners. Open MPI is therefore able to combine the expertise, technologies, and resources from all across the High Performance Computing community in order to build the best MPI library available. Open MPI offers advantages for system and software vendors, application developers and computer science researchers.

Search Page

- [1. Quick start](#)
- [2. Getting Help](#)
 - [2.1. Where to send?](#)
 - [2.2. For run-time problems](#)
 - [2.3. For compile problems](#)
- [3. General notes](#)
 - [3.1. Platform Notes](#)
 - [3.2. Compiler Notes](#)
 - [3.3. General Run-Time Support Notes](#)

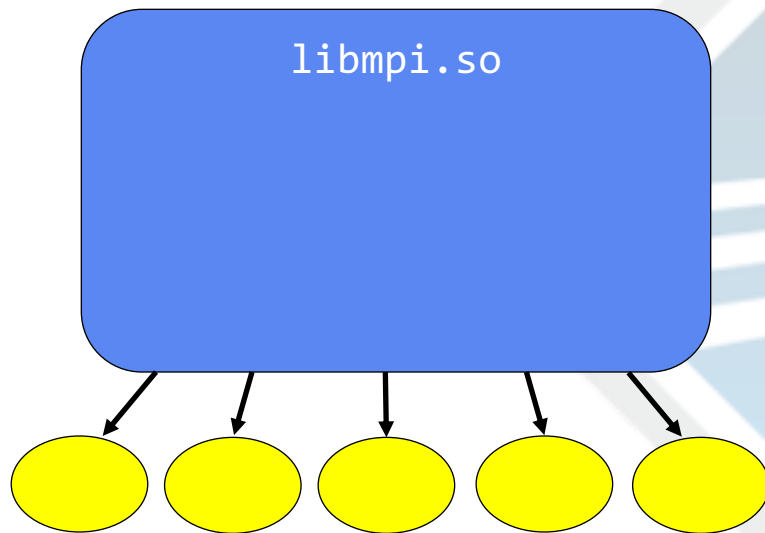
Open MPI v5.0.0 documentation is moving to readthedocs.io!

Content is being revamped and fully refreshed for v5.0.0

Will hopefully be ready in time for the v5.0.0 launch 🙌

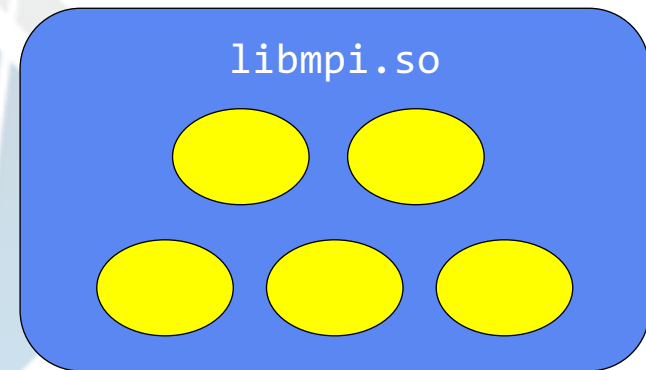
v5.0.x defaults

Open MPI \leq v4.x



Plugins loaded dynamically at run time

Open MPI 5.0.x



Plugins located in the library

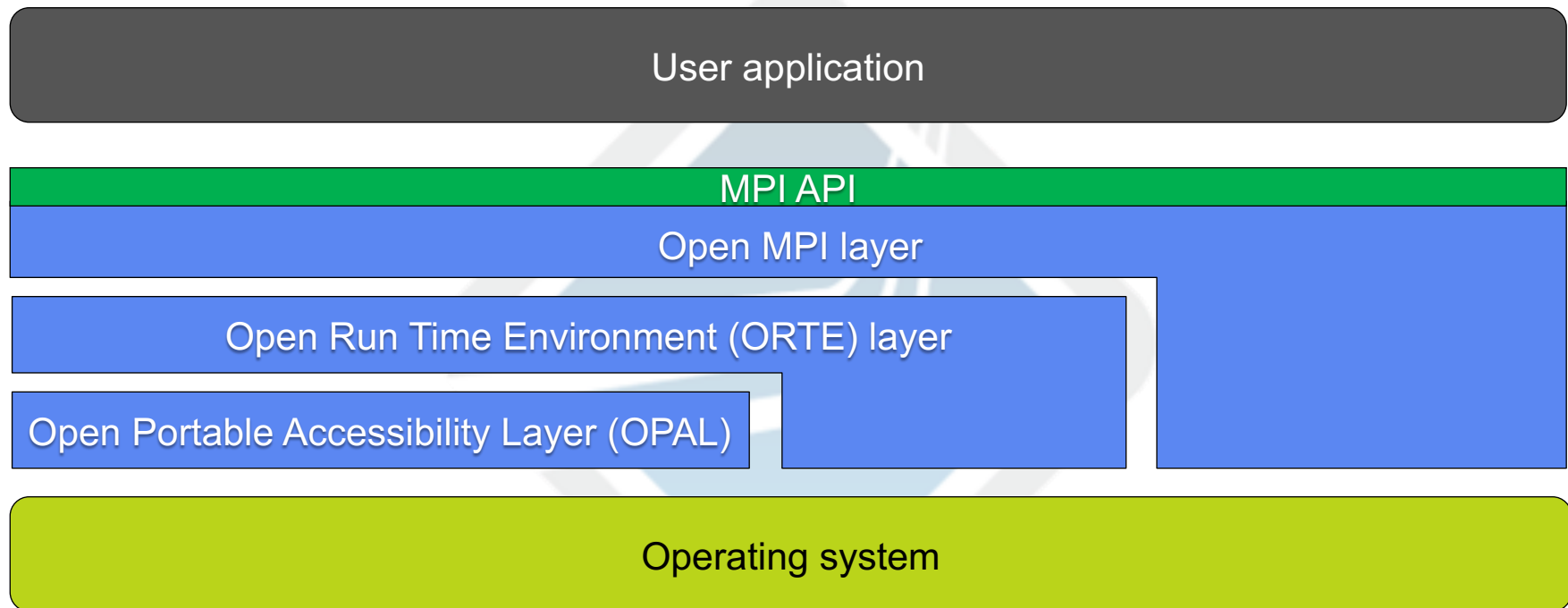
v5.0.x packaging

Package	v4.x	v5.0.x
hwloc	Prefer external	Prefer external
libevent	Prefer external	Prefer external
Open PMIx	Prefer external	Prefer external
ROMIO	Internal	Internal
Treematch	Internal	Internal
PRRTE		Prefer external

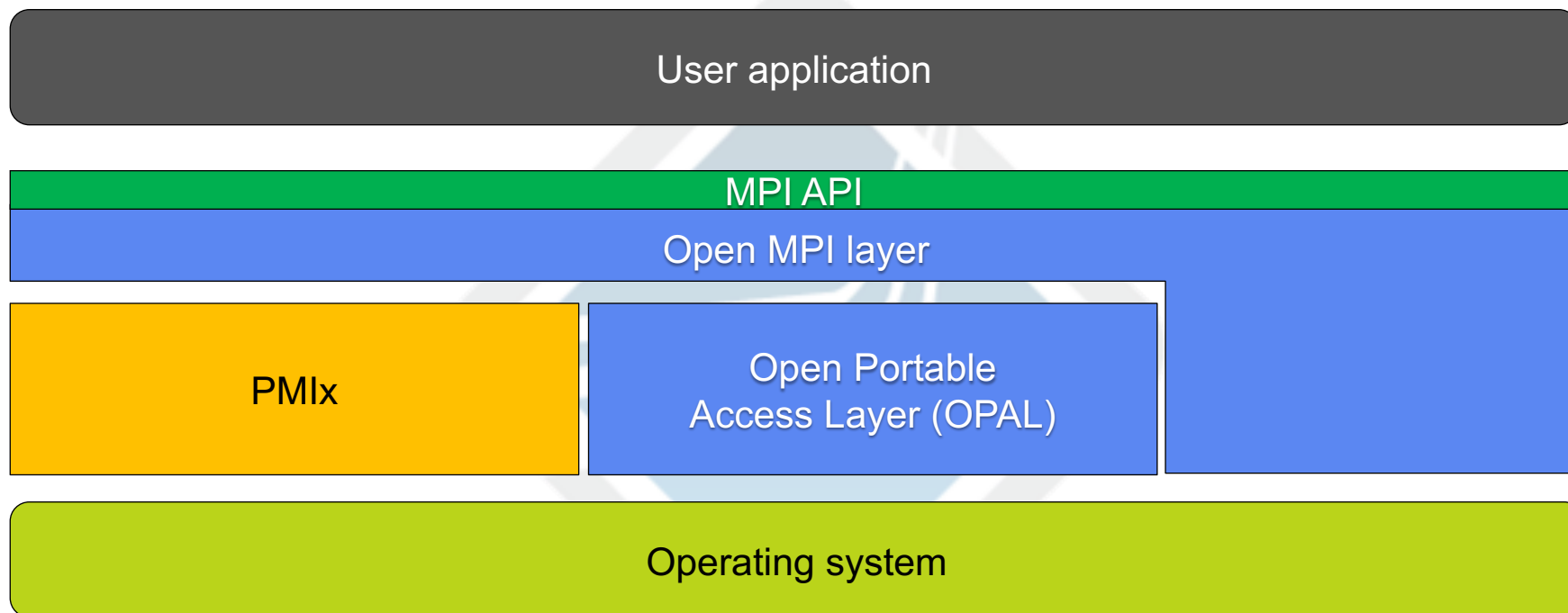
“Prefer external” = Package is included in the Open MPI source code distribution, but configure will look for a system-installed copy and try to use that rather than the included copy

“Internal” = Package is included in the Open MPI source code distribution, and will only use the included copy

Open MPI software layering \leq v4.x



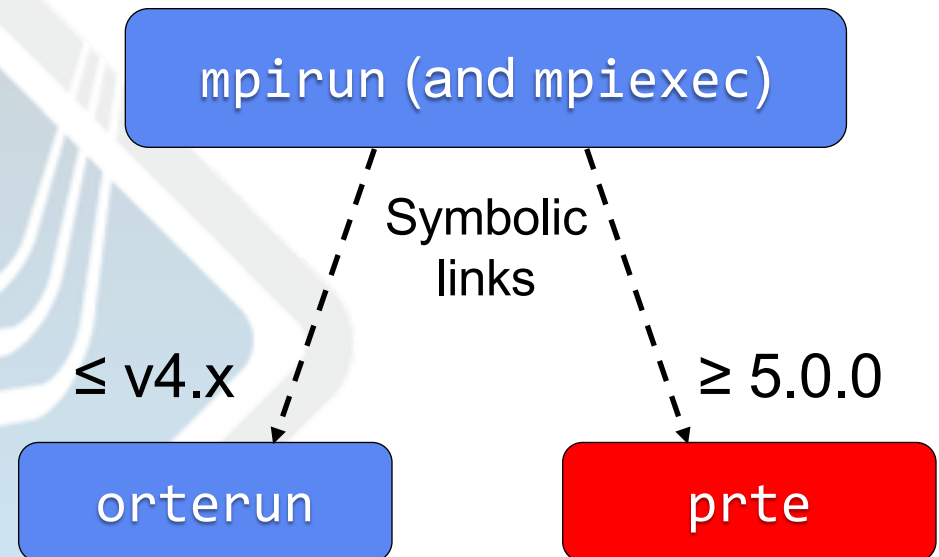
Open MPI software layering $\geq 5.0.0$



ORTE → PRRTE

(more on this in a few slides)

- ORTE effectively forked
 - Evolved into the PMIx Reference Runtime Environment (PRRTE)
- Run time no longer specifically tied to Open MPI
 - Also suitable for other environments



Other v5.0.0 changes / features

- Backwards compatibility break from v4.x
 - ABI and command line options
- PRRTE v2.0.x
 - Disabled “single dash” command line options (e.g., --mca, not -mca)
 - MPIR replaced by PMIx debugging interface
 - “Instant on” (for supported networks)
- New features:
 - User Level Fault Mitigation (ULFM) support
 - FP16 support (MPIX extension)
 - GPFS support
 - AVX support for MPI reduction operations
- Improved performance:
 - MPI collectives (HAN, ADAPT)
 - UCX multi-threaded support
 - Atomic operations (ARM64, PPC, C11)
 - MPI datatypes



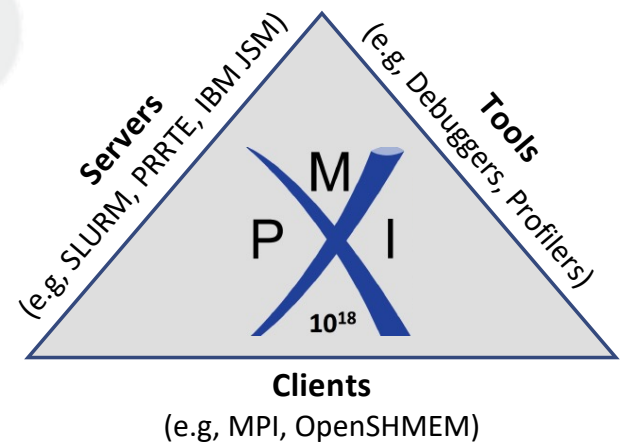
PMIx, OpenPMIx, & PRRTE

ECP Community BOF Days
Updates and Roadmap for the PMIx Community
March 31, 2021 from 11-12:30 US Eastern time

What is PMIx?

PMIx is a standard API providing libraries and programming models with portable and well-defined access to commonly available system services.

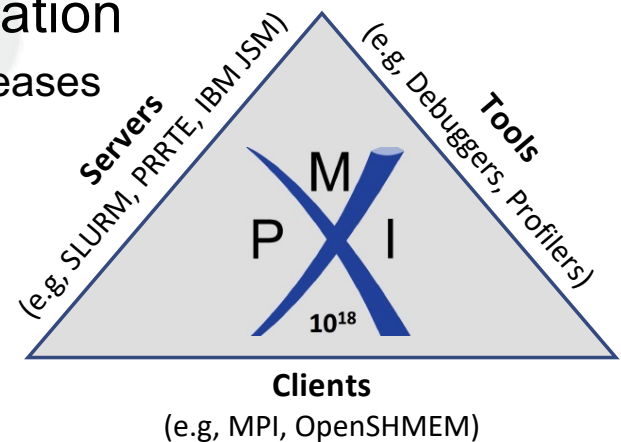
- PMIx is a messenger between these pieces of software, not a doer.
 - Facilitates the interaction between applications, tools, runtime environments.
- Open, community driven standard.
- Use cases: (summarized list)
 - **Process wire-up** via either business card exchange or “instant on” (where supported)
 - **Tool connections** including debugger support
 - **Event notification** used by fault tolerant libraries
 - Application/Job/Node **environment discovery**
 - Job scheduler interaction <https://github.com/pmix/pmix-standard>
<https://github.com/pmix/governance>



What is OpenPMIx?

OpenPMIx is a feature complete implementation of the PMIx standard.

- OpenPMIx provides the implementation to connect **PMIx-enabled clients** (like Open MPI) with **PMIx-enabled Tools** (like debuggers) and **PMIx-enabled Servers** (like PRRTE, SLURM, IBM JSM)
 - Works primarily as a messenger between these pieces of software, not a doer.
- Open, community supported, scalable implementation
 - OpenPMIx releases tied to corresponding PMIx Standard releases
 - Proving ground for new PMIx standard additions
 - Currently used on many large scale HPC systems including all of the top 3 systems in the Top500 Nov. 2020 list.
 - Cross-version compatibility allowing clients to use a different version of OpenPMIx than the server or tool.



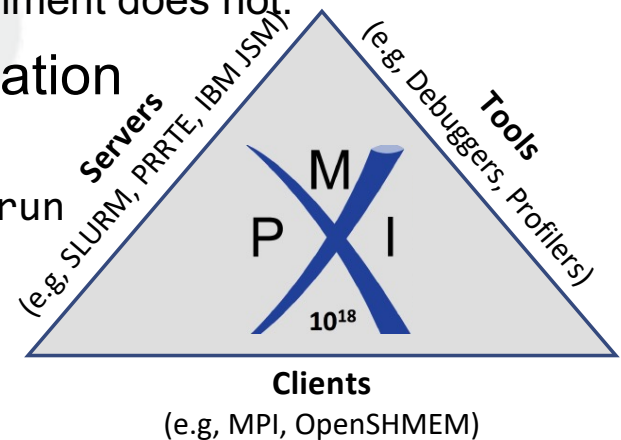
<https://github.com/openpmix/openpmix>

What is PRRTE?

The PMIx Reference RunTime Environment (PRRTE) is a featureful, scalable, PMIx-enabled runtime environment.

- PRRTE supports the PMIx standard interfaces needed for **PMIx-enabled clients** (like Open MPI) and **PMIx-enabled Tools** (like debuggers) to interact across HPC systems through the PMIx interface.
 - PRRTE provides a PMIx environment even if the host environment does not.
- Open, community supported, scalable implementation
 - Proving ground for new PMIx standard additions
 - Supports one-off jobs via prterun & multiple jobs via prte/prun
 - PMIx tool interface support (replacement for MPIR)
- Evolution of the ORTE runtime from Open MPI into a standalone project.

<https://github.com/openmix/prrte>



PRRTE and Open MPI v5.0.x

- PRRTE is the default runtime for Open MPI v5.0.x
 - Requires OpenPMIx \geq v4.0.1 and PRRTE \geq v2.0.0
 - Provides much of the same feature set as ORTE plus some!
 - Open MPI interacts with the runtime **only via PMIx** standard interfaces
 - PMI-1 and PMI-2 interfaces are **no longer supported**.
 - Some `mpirun` **command line options have changed** in an effort to simplify the interface and to better support what is provided.
 - The MPIR process acquisition interface has been **removed**.
 - Replaced with the more generic **PMIx Tools interface**
 - A MPIR-Shim is available along with a porting guide for tool projects
 - <https://github.com/openpmix/pmi-shim>
 - <https://github.com/openpmix/mpir-to-pmix-guide>



LANL Update

Howard Pritchard

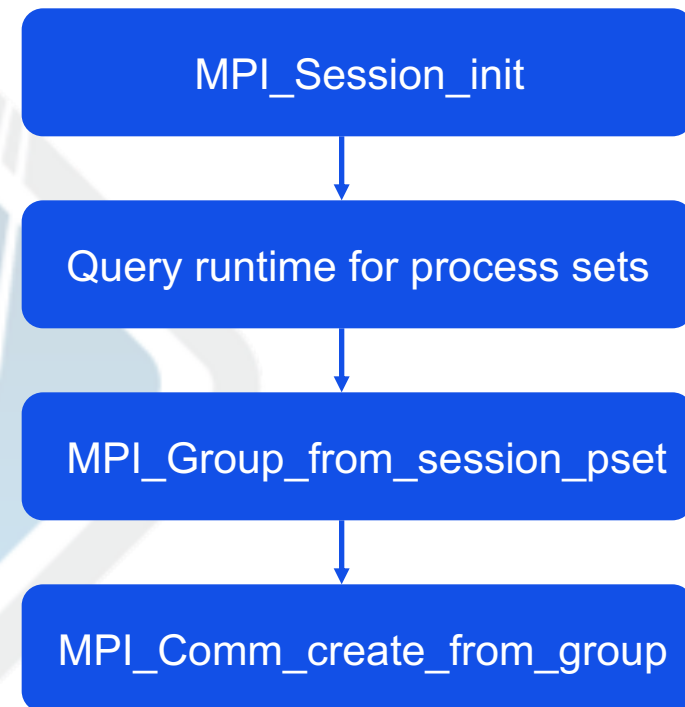
Los Alamos National Laboratory



LA-UR-21-22943

MPI Sessions Update

- Part of MPI 4 standard
- Prototype based on Open MPI available at <https://github.com/hpc/ompi.git> *sessions_new* – branch
- Basic tests available at <https://github.com/open-mpi/ompi-tests-public.git>
- Sessions functionality available when using OB1 PML or OFI MTL (exCID support)
- Plan to merge into master post-v5.0.0 release



OpenPMIx Testing

- Collaborating with Nvidia to develop a PMIx unit test suite using an alternative resource manager(RM)
- Ensures OpenPMIx is functioning as expected irrespective of RM
- Incorporates internal correctness checks
- Keeping individual tests simple – to better help isolate failures
- Available as part of OpenPMIx - https://github.com/openpmix/openpmix/tree/master/test/test_v2

Odds and Ends

- Testing Open MPI with Intel oneAPI compilers
 - Preparation for ANL Aurora system
 - Some turbulence with ifx
- MPI Testing Tool (MTT)
- One of the maintainers of Spack Open MPI package
 - Package going to need revamping for 5.0.x releases



Mellanox/Nvidia Update

Joshua Ladd
Sr. Director, Nvidia

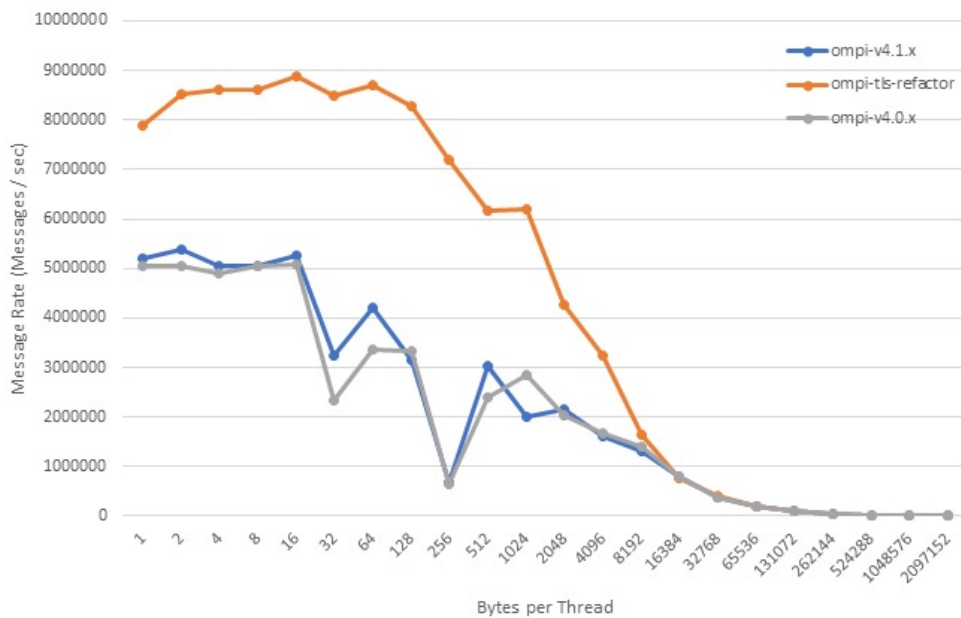


Scaling Open MPI to Exascale

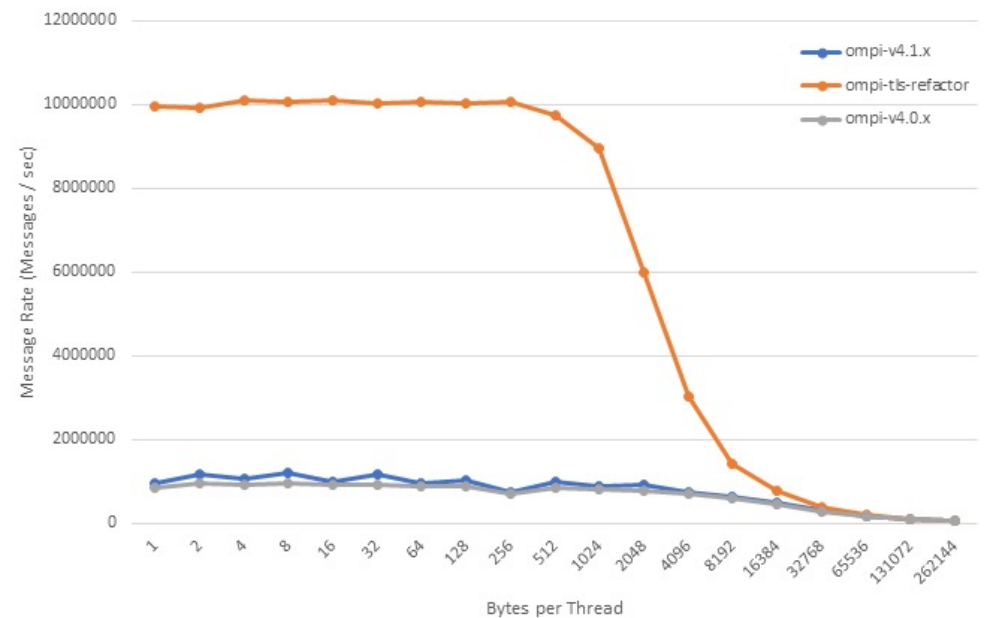
- UCX OSC thread-multiple performance optimized.
 - Available in v5.0 out-of-the-box 🎉
- OSHMEM thread-multiple performance optimized.
 - Available in preview starting in v4.0.3
 - `--mca smpi_ucx_nb_progress_thresh_global 2048`
 - `--mca smpi_ucx_nb_ucp_worker_progress 64`
 - `--mca smpi_ucx_default_ctx_ucp_workers 2`
 - Out-of-the-box in v5.0.0.

UCX OSC Thread Multiple

RMA-MT Message Rate, Threads = 2, Jazz HDR200

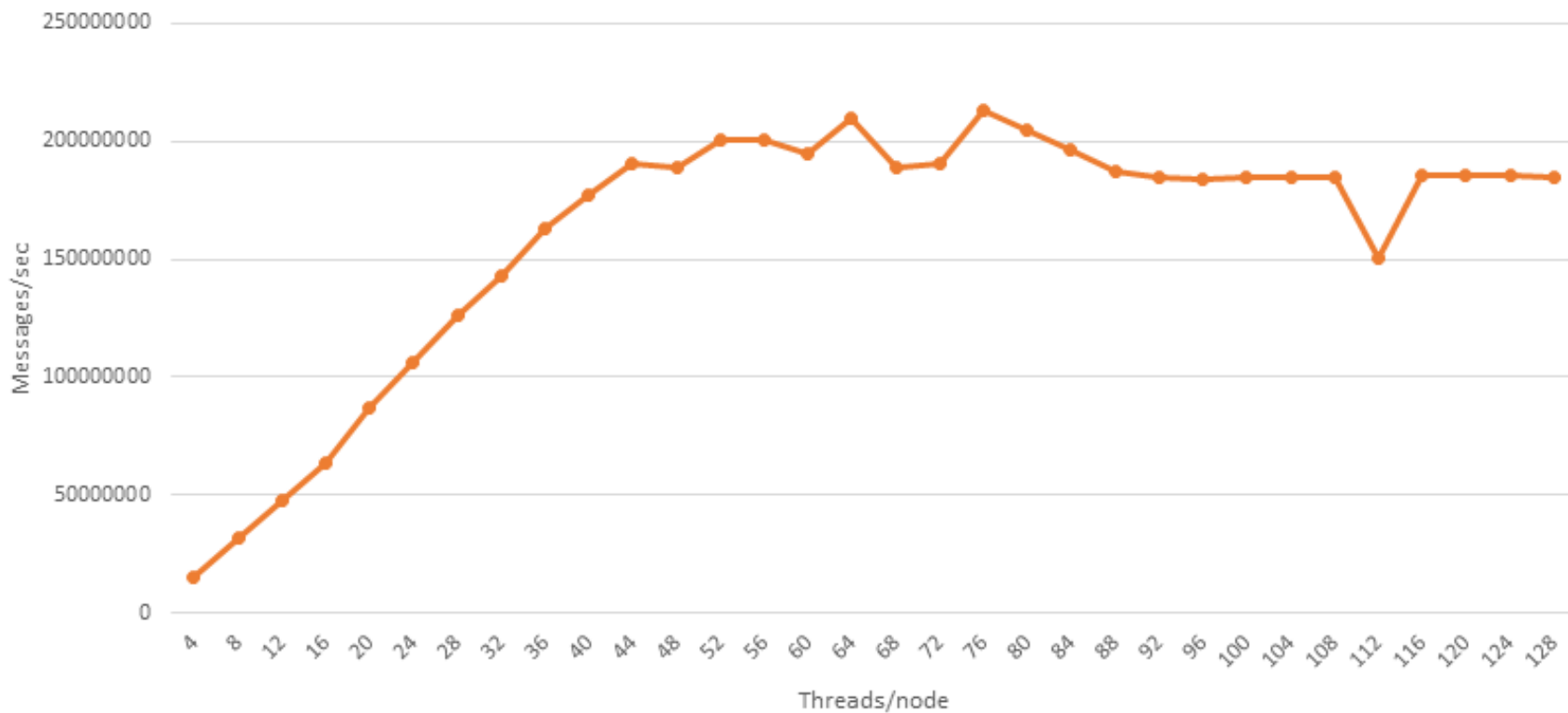


RMA-MT Message Rate, Threads = 16, Jazz HDR200



OSHMEM Thread Multiple

AMD ROME, HDR200, shmem_ctx_put64(), uni-directional





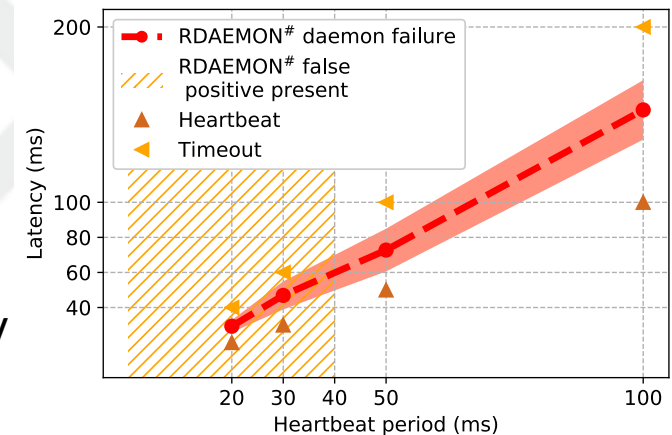
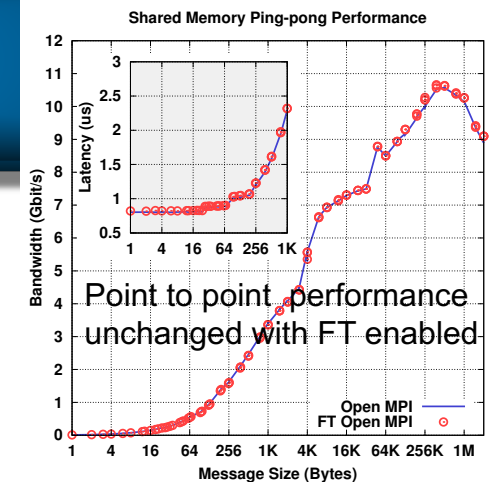
UTK Open MPI activities

George Bosilca



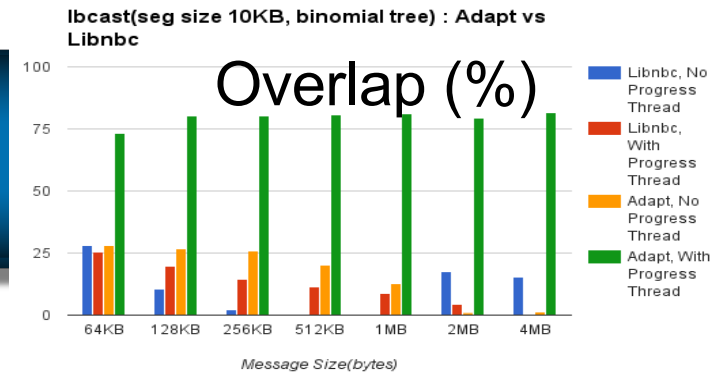
Resilience - User Level Failure Mitigation (ULFM)

- ULFM now integrated in master
 - Based on OMPI 4.0.2 (will remain in sync)
 - `--with-ft=mpi`
- Move the underlying resilient mechanisms outside ULFM/OMPI
 - Failure detector and reliable broadcast in PPRTE
 - Used in OMPI ULFM and SUNY OpenSHMEM
- Scalable fault tolerant algorithms demonstrated in practice for revoke, agreement, and failure detection (SC'14, EuroMPI'15, SC'15, SC'16)
- Extensions are in development for additional asynchrony (async spawn, shrink) and info keys for more flexible behaviors (automatic revocation, consistent collectives, consistent communicator creation)

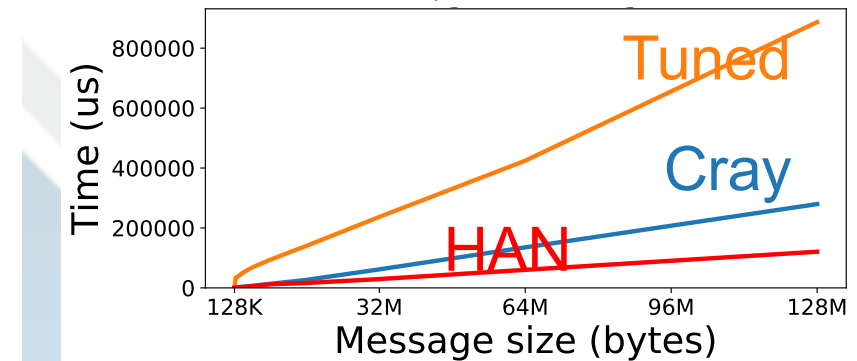


Adaptive Collective communications framework (HAN and ADAPT)

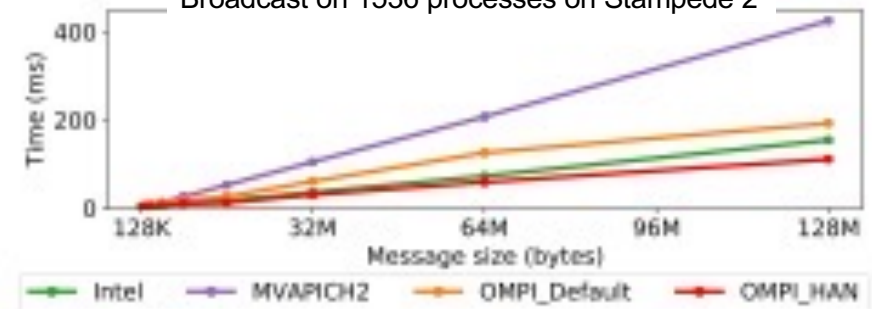
- More complex architectures demand more complex collective frameworks
 - Tuned was not built to support hierarchical architectures nor cope with system noise nor provide overlap
- Event-driven collective based on a dataflow state machine
 - Architecture aware
 - Reactive to network noise
 - Provide support for overlap for non-blocking collectives
- Summit tuning in underway



Broadcast on 4k processes on Shaheen II

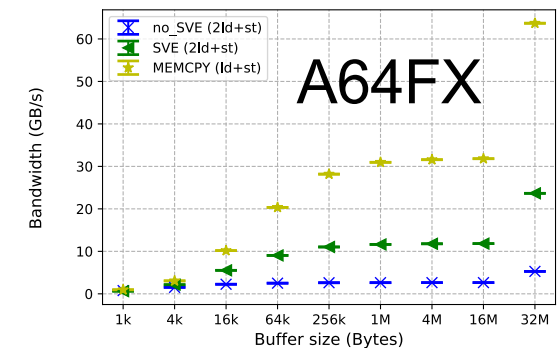
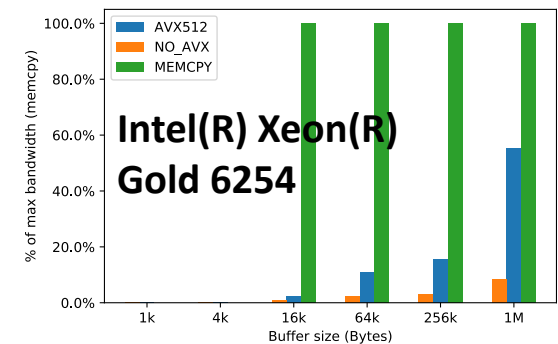


Broadcast on 1536 processes on Stampede 2



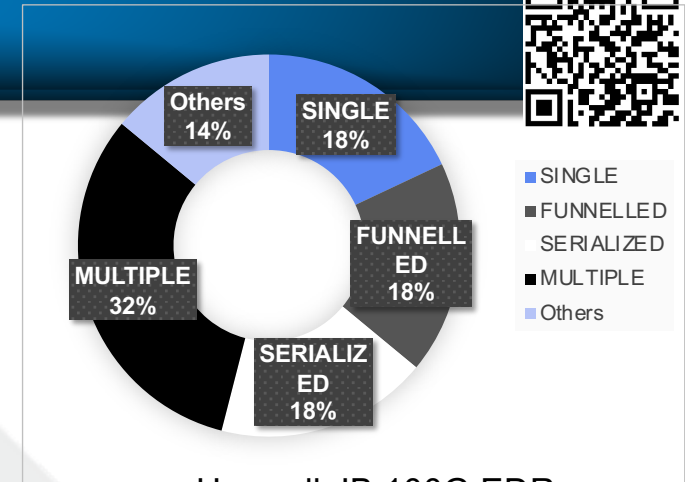
Explicit support for vectorial ISA

- First step of adding support for vectorial instructions to MPI_Op
 - Intel AVX* and ARM SVE
 - Distinction between build capabilities and execution capabilities
- Lessons can now spread to other parts of the code, especially the datatype engine
 - Prefetching, gather/scatter for non-contiguous datatypes

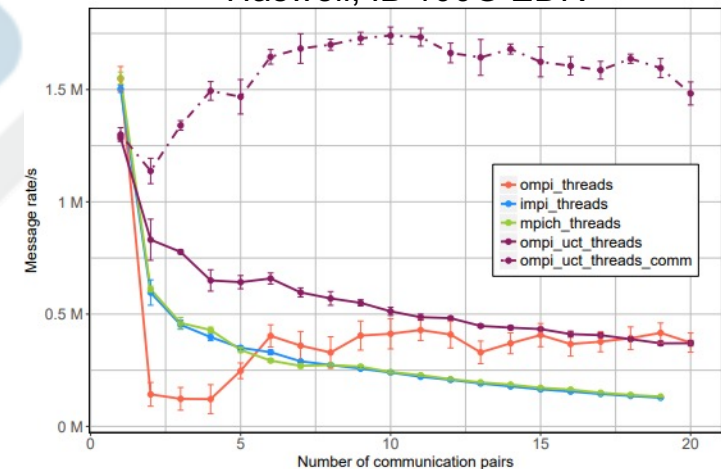


Threading

- Growing interest from the community (inside and outside ECP)
 - The MPI Forum is also looking at some aspects (partitioned communications)
 - Injection rate is important, and OMPI has addressed this in the past reasonably well
 - Extraction rate remains an issue, at the MPI level and at the user level
 - Redesigned how requests are waited upon continuations and synchronizations
 - Work in progress (code available in github) and proposal coming to the MPI Forum soon

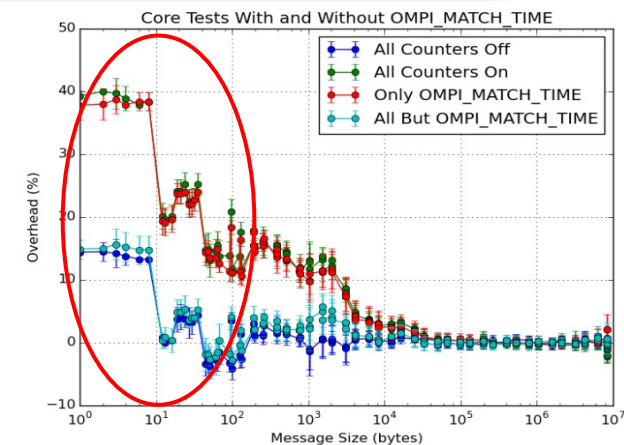


Haswell, IB 100G EDR



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)





IBM Spectrum MPI

Joshua Hursey, Austen Lauria, Geoff Paulsen
IBM





IBM
Spectrum
MPI

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, JSM, ESSL, and Spectrum Scale

Supports 5 of the top 20 systems in the Top500 as of Nov. 2020

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

Nvidia
(Mellanox)

ICL UT

Cisco

Fujitsu

...

Spectrum MPI 10.4 is based on Open MPI 4.0.x with PMIx 3.2.x

Spectrum MPI Community Edition Available!



IBM
Spectrum
MPI

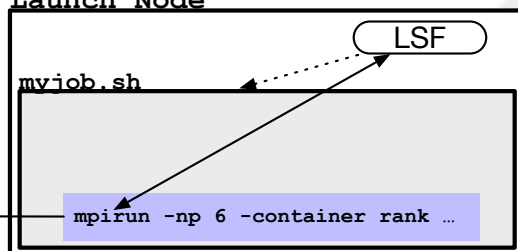
Container Ready Supporting Applications on Bare Metal & Private/Public Cloud

Spectrum MPI options to support launching applications in the two different container modes commonly used in HPC environments

Rank Contained mode

One container per application process

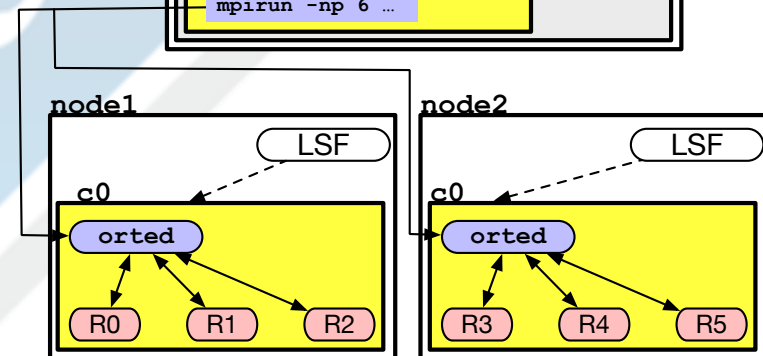
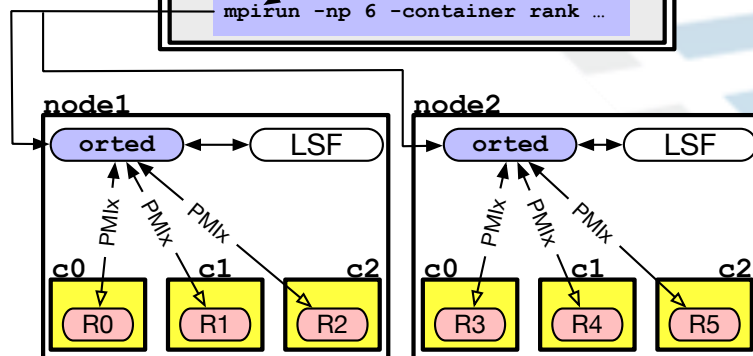
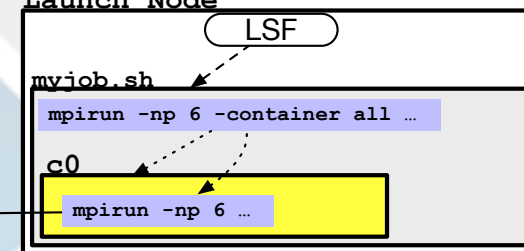
Launch Node



Fully Contained mode

One container per node

Launch Node



https://www.ibm.com/support/knowledgecenter/SSZTET_10.4/smpi02_containerized_apps.html



Open MPI at Amazon

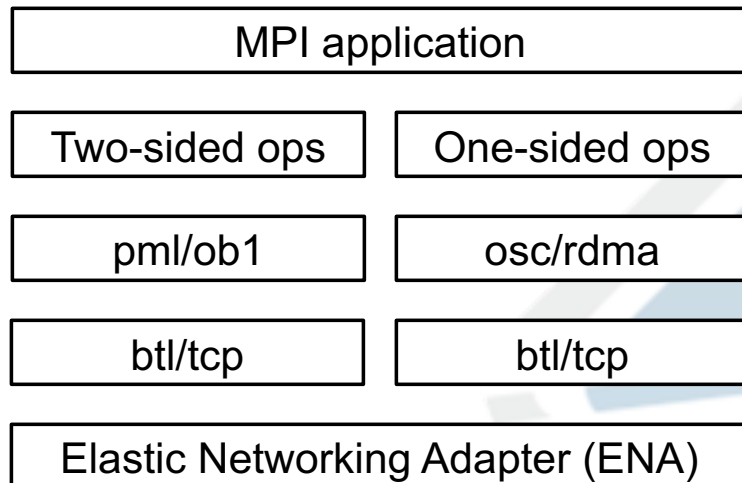
Raghu Raja and Brian Barrett



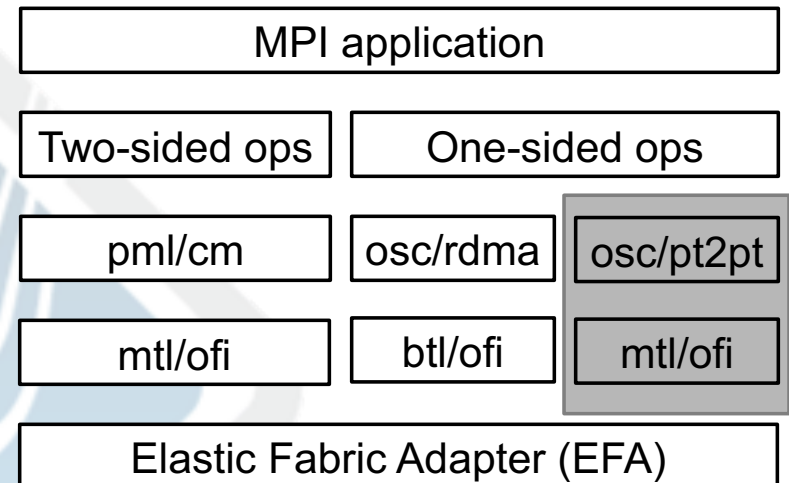
Community Involvement

- Feature development and investments in long-term maintenance
 - TCP BTL
 - Reachable and graph solver enhancements, multi-nic fixes and enhancements, etc.
 - Libfabric components
 - Support for multi-rail systems, AV improvements, revamped provider query flow, etc.
 - Providing GPUDirect RDMA support for the OFI MTL in 5.0 (full GPUDirect RDMA support with Libfabric in the works)
 - Performance enhancements
 - Collective algorithm tuning framework, improved default thresholds, automated monitoring internally for regressions.
 - PMIx Runtime
 - More recently, schizo framework improvements in preparation for Open MPI 5.0.
 - Lots more to come.
- Resources to maintain and improve the quality of Open MPI codebase
 - Continued funding for the project's web and community CI infrastructure
 - Nightly testing w/MTT – Active release branches (currently, v4.0.x, v4.1.x, v5.0.x) tested with multiple test suites
- Release management and project advocacy

Frameworks and Components



- ENA uses SR-IOV to deliver up to 100Gbps
- Supported on all current generation instance types, except T2.
- More details available [here](#).



- EFA provides lower and consistent latencies than TCP
- Supports OS-Bypass communication and leverages the [Scalable Reliable Datagrams protocol](#).
- More details available [here](#).

With OMPI versions 5.0 and newer, one-sided operations use the OFI BTL in place of the osc/pt2pt component used with prior releases



Omni-Path and PSM2

Michael Heinz



CORNELISTM
NETWORKS


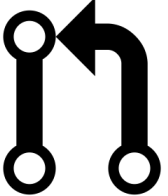
Omni-Path acquired by Cornelis Networks

- Support for Open MPI on Omni-Path continues!
- Cornelis Networks launched in the Fall of 2020
 - Acquired Omni-Path business from Intel
 - Product Line, IP, inventory, manufacturing
 - Resumed product development
 - Open MPI support will be a priority
- **NOTE**: PSM3 is not a PSM2 replacement



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!