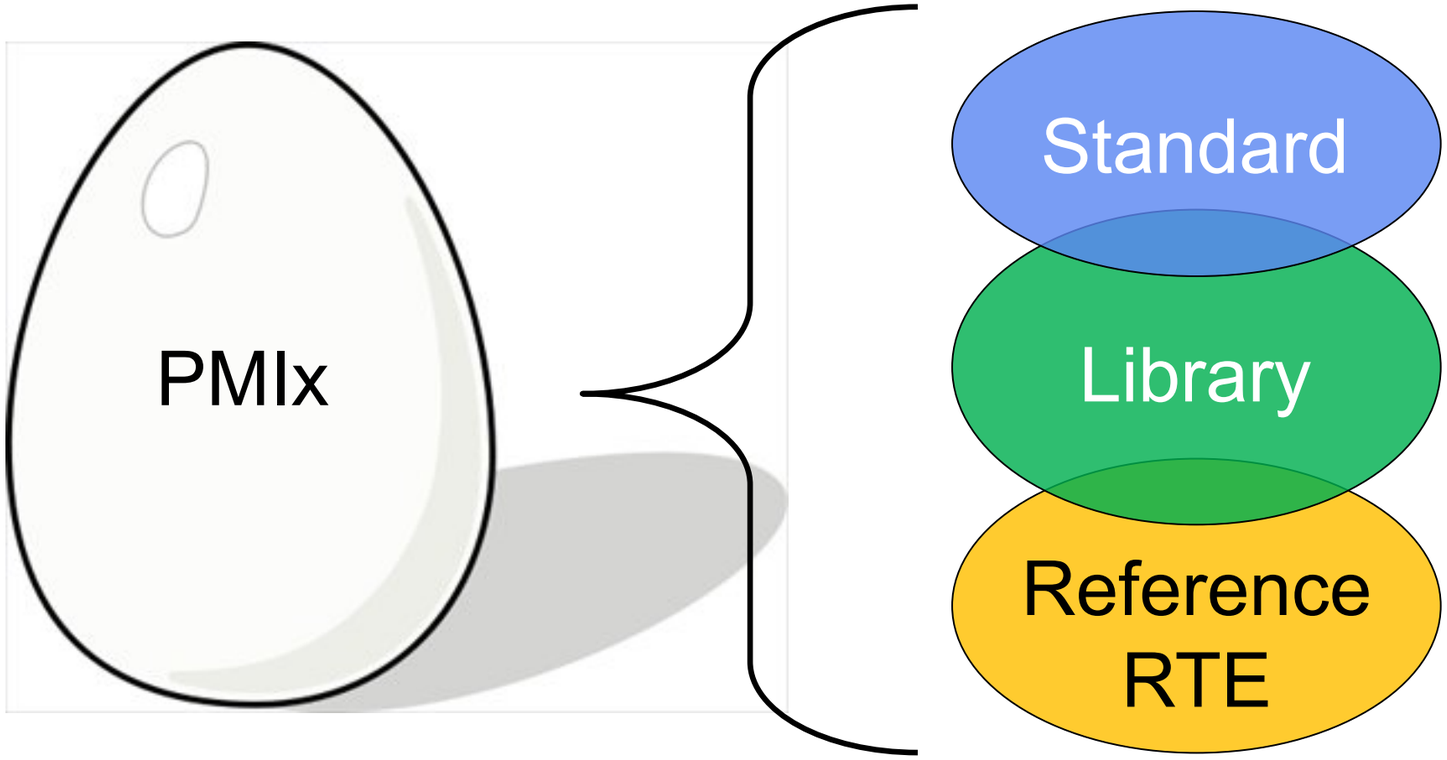
PMIx: Process Management for Exascale Environments

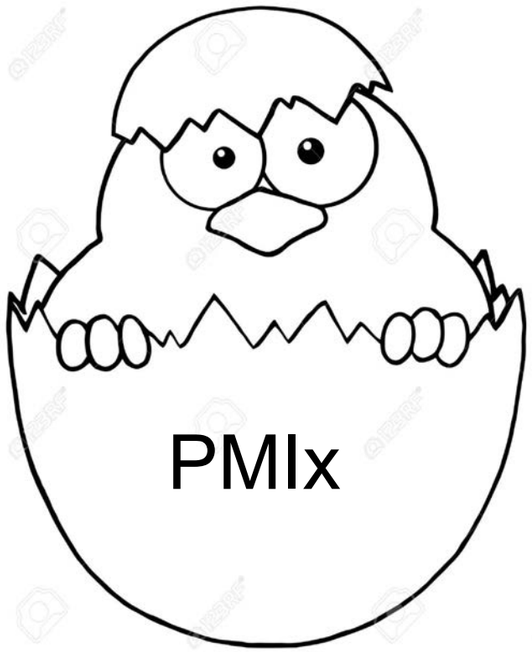
Agenda

- State of the Community
  - Ralph H. Castain (Intel)
- PMIx Standard
  - Josh Hursey (IBM)
  - Kathryn Mohror (LLNL)
- Q&A

With Success…



With Success…Maturity



PMIx Standard

Implementation independent

Formal Governing body

https://github.com/pmix/

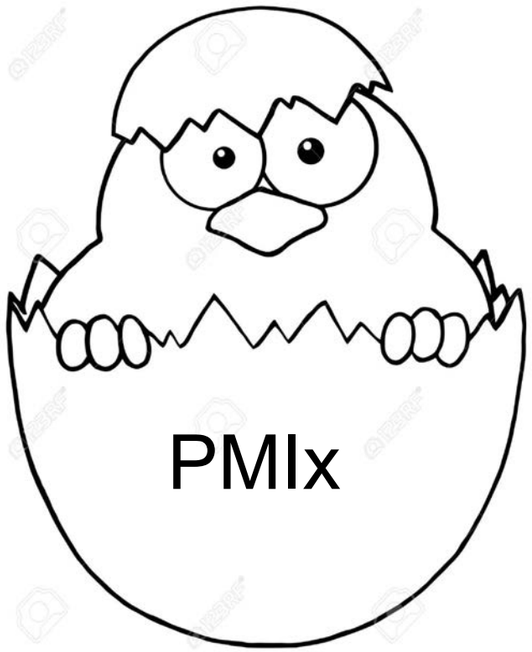
OpenPMIx Library

Still distributed as libpmix

Same consortium of contributors

https://github.com/openpmix/

With Success…Maturity



Split after PMIx v4 (1Q2020)

PMIx Standard

Implementation independent

Formal Governing body

https://github.com/pmix/

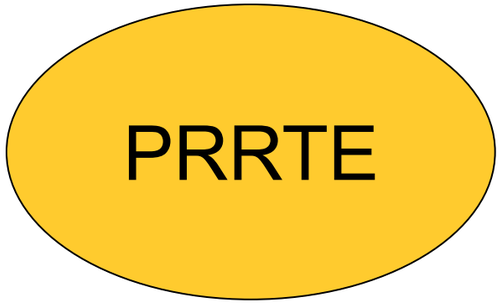
OpenPMIx Library

Still distributed as libpmix

Same consortium of contributors

https://github.com/openpmix/

With Success…Maturity



Shim for environments missing some PMIx functionality

Affiliated with OpenPMIx

https://github.com/openpmix/prrte

Over The Next Year

- Website refactoring (https://pmix.org)
- GitHub reorganization…done!
- PMIx v4 releases
  - Standard – under the “old” system, reorganize doc a bit, complete the original v4 plan
  - Library – track the v4 Standard, retain “libpmix” name
- Growing effort on v5 of the Standard

State of OpenPMIx

- Release of v4.0 series
  - Attribute queries
  - Process groups/sets
  - Network coordinates/topology
  - Extended tool/debugger support
  - Python bindings
  - Scheduler, query access to fabric info

1Q2020

State of OpenPMIx

- Release of v4.0 series
  - Attribute queries
  - Process groups/sets
  - Network coordinates/topology
  - Extended tool/debugger support
  - Python bindings
  - Scheduler, query access to fabric info

2020 Focus:

Dynamic Workflows

(Spark, TensorFlow)

Storage Integration

(Lustre, …?)

Adoption

- RMs
  - SLURM, IBM's Job Step Manager (JSM)
  - PBSPro, Fujitsu, Flux, ParaStation Management
- Libraries
  - MPI: Open MPI, Spectrum MPI, Fujitsu MPI, MPICH, Intel MPI, HPE-MPI
  - OpenSHMEM: Stonybrook, OSHMEM, SOS, OREO
- Tools
  - Debugger integration under development

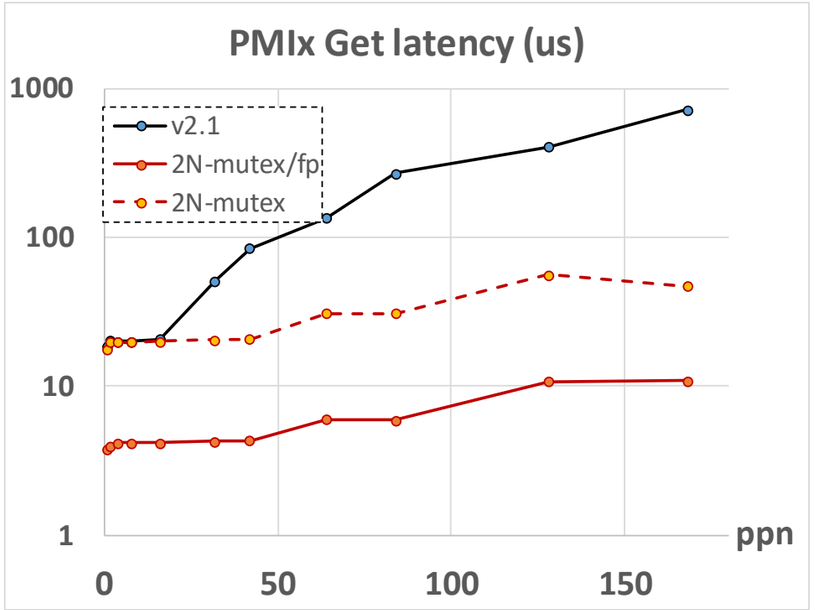
OpenPMIx Performance Analysis

Artem Y. Polyakov, Boris I. Karasev, Joshua Hursey, Joshua Ladd, Mikhail Brinskii and Elena Shipunova

A performance analysis and optimization of PMIx-based HPC software stacks.

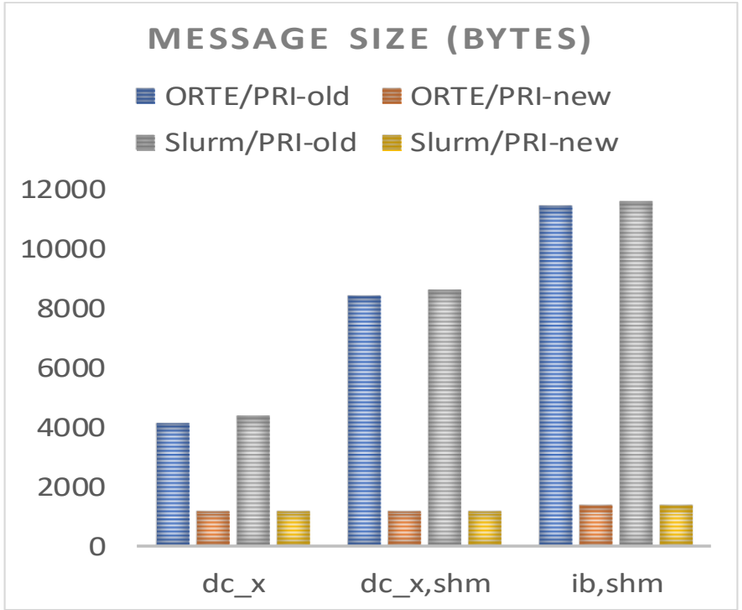
EuroMPI 2019 (https://doi.org/10.1145/3343211.3343220)

PMIx_Get latency optimizations



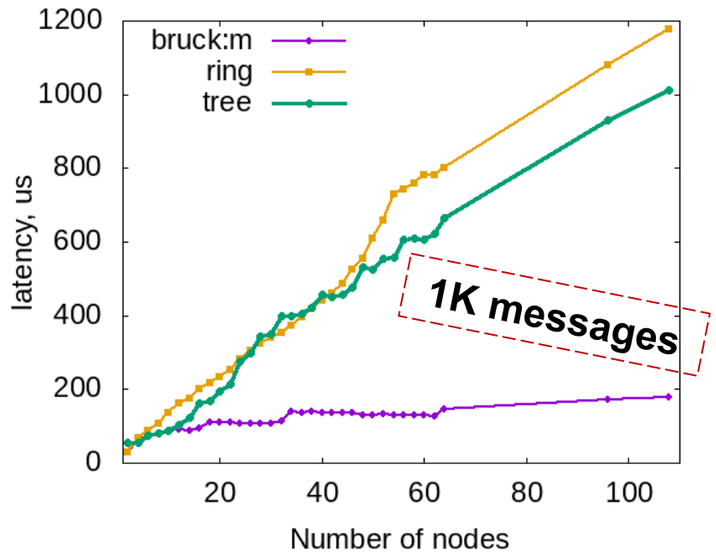
ORNL Summit system (256 nodes)
SW stack: IBM JSM / Open MPI / IBM PAMI / PMIx
Node: 2 IBM POWER CPUs, 600 GB RAM
CPU: 22-core IBM POWER 9, 8 HW threads / core

PMIx_Fence message size optimization



Intel 64 system (64 nodes)
SW stack: Slurm / Open MPI / UCX / PMIx (varied)
Node: 2 Intel CPUs, 128 GB RAM
CPU: 16-core Intel Broadwell (2.6 GHz), 1 HWT/core

PMIx_Fence exchange optimization
(Adaptation of the Burck Allgatherv)



Intel 64 system (108 nodes)
SW stack: Slurm (**+Bruck impl.**), UCX-based OOB
Node: 2 Intel CPUs, 128 GB RAM
CPU: 16-core Intel Broadwell (2.6 GHz), 1 HWT / core

Nov. 2019 Top500

OpenPMIx at Scale

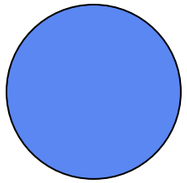
Rank	System	Resource Manager	Rmax (Pflops/s)
#1	ORNL Summit	IBM JSM	148.600
#2	LLNL Sierra	IBM JSM	94.640
#5	TACC Frontera	Slurm	23.516
#8	AIST ABCI	Fujitsu	19.880
#9	SuperMUC-NG	Slurm	19.477
#10	LLNL Lassen	IBM JSM	18.200
#11	Total Pangea III	IBM Spectrum MPI	17.860

Range of Coverage

MPI Wireup

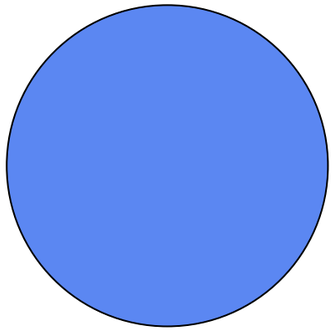
Slurm
Fujitsu
Flux
PSM
Cray (shim)

+Events



JSM

+InstantOn



PBSPro

+Dynamic Workflow
(sets/groups, dyn alloc)
+Storage Directives

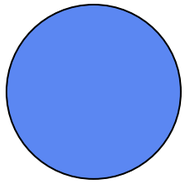
PRRTE
ORTE (OMPI)

Range of Coverage

MPI Wireup

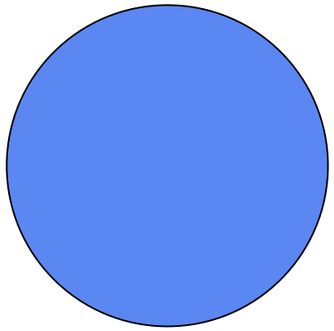
Slurm
Fujitsu
Flux
PSM
Cray (shim)

+Events



JSM

+InstantOn



PBSPro

+Dynamic Workflow
(sets/groups, dyn alloc)
+Storage Directives

PRRTE
ORTE (OMPI)

PRRTE => shim to even coverage

SC19 Paper

Similar results on Cray

Characterizing the Performance of Executing Many-Tasks on Summit

Session: 3rd International Workshop on Emerging Parallel and Distributed Runtime Systems and Middleware (IPDRM'2019)

Author/Presenters: Matteo Turilli, Andre Merzky, Thomas Naughton, Wael R. Elwasif, Shantenu Jha

Event Type: Workshop

Registration Categories:
W

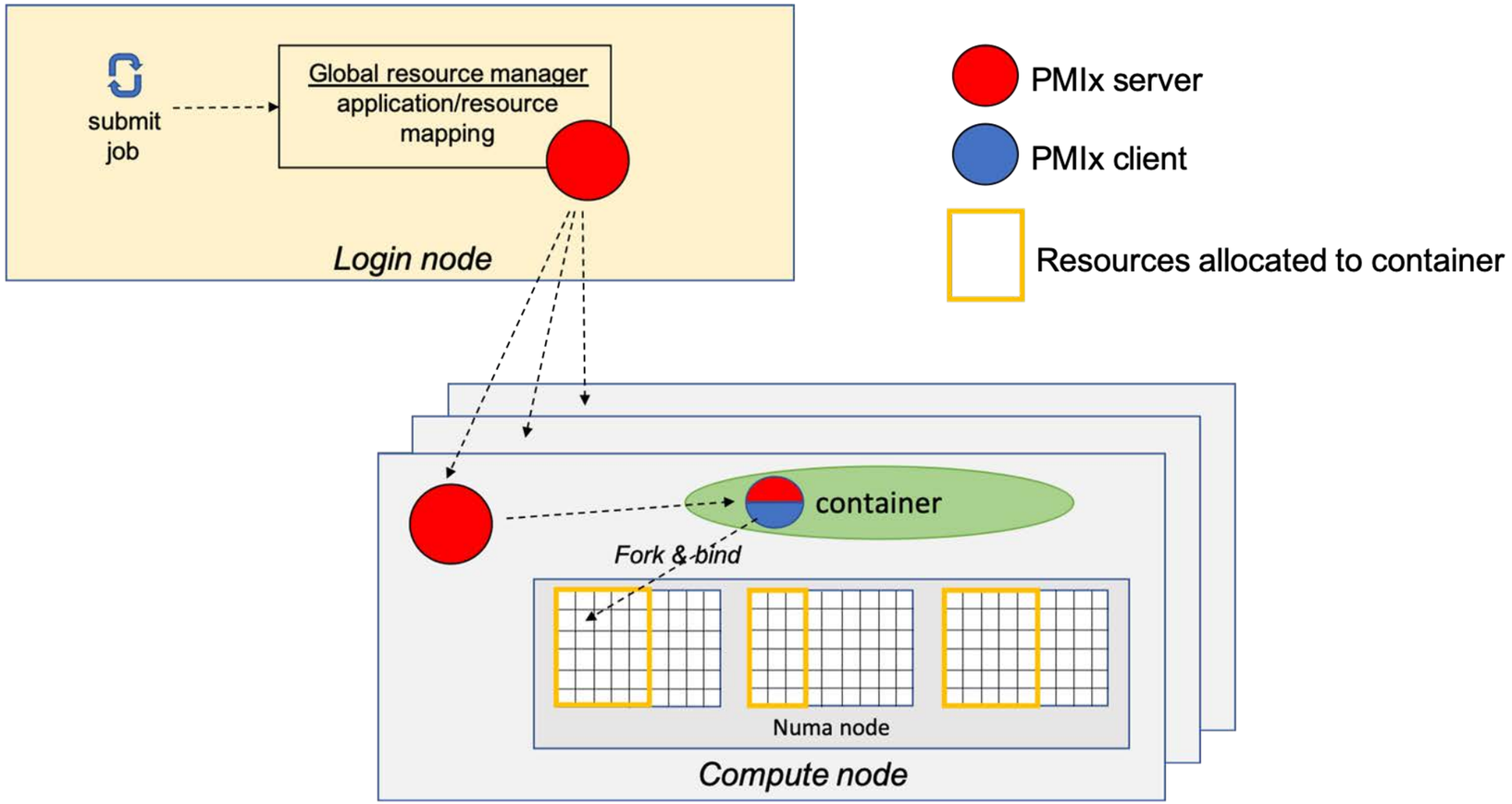
Tags:
Compiler Analysis and Optimization | Middleware | Parallel Programming Languages, Libraries, and Models | Runtime Systems

Time: Friday, 22 November 2019, 9:07am - 9:25am

…for workloads comprised of homogeneous single-core, 15 minutes-long tasks we find that: PRRTE scales better than JSM for > O(1000) tasks; PRRTE overheads are negligible; and PRRTE supports optimizations that lower the impact of overheads and enable resource utilization of 63% when executing O(16K), 1-core tasks over 404 compute nodes.

On-node resource management w/ Singularity

Proposed Architecture



Geoffroy Vallee, Carlos Eduardo Arango Gutierrez, and Cedric Clerget
On-node resource manager for containerized HPC workloads.
CANOPIE-HPC 2019 (https://conferences.computer.org/sc19w/2019/#!/toc/0)
https://www.canopie-hpc.org/program/

- Add a new PMIx thread to the Singularity runtime
- Interface Singularity with the global resource manager
- Create a scalable hierarchy of PMIx servers
- Ease the mapping of resources
- Enable resource management for everything running in containers

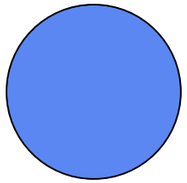
Sylabs.io

Range of Coverage

MPI Wireup

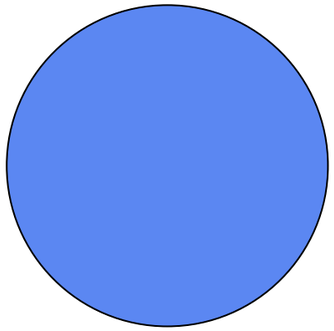
Slurm
Fujitsu
Flux
PSM
Cray (shim)

+Events



JSM

+InstantOn



PBSPro

+Dynamic Workflow
(sets/groups, dyn alloc)
+Storage Directives

PRRTE
ORTE (OMPI)

Remember to specify coverage in RFPs!
(doc reorg to help)

With Success…Maturity
…and Retirement

Agenda

- State of the Community
  - Ralph H. Castain (Intel)
- PMIx Standard
  - Josh Hursey (IBM)
  - Kathryn Mohror (LLNL)
- Q&A

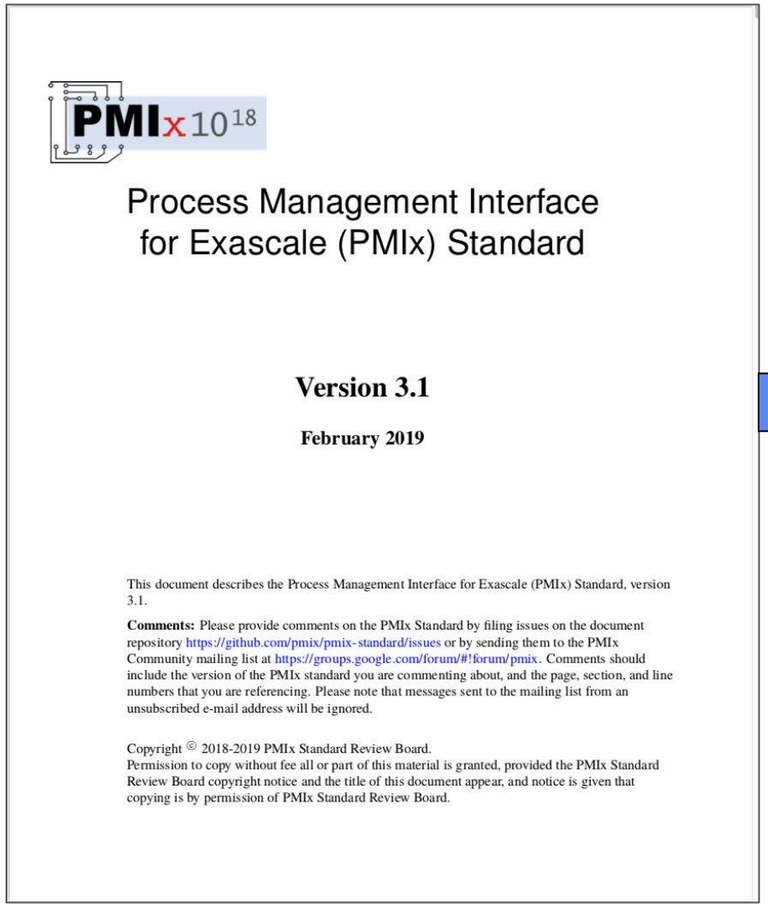
PMIx Standard

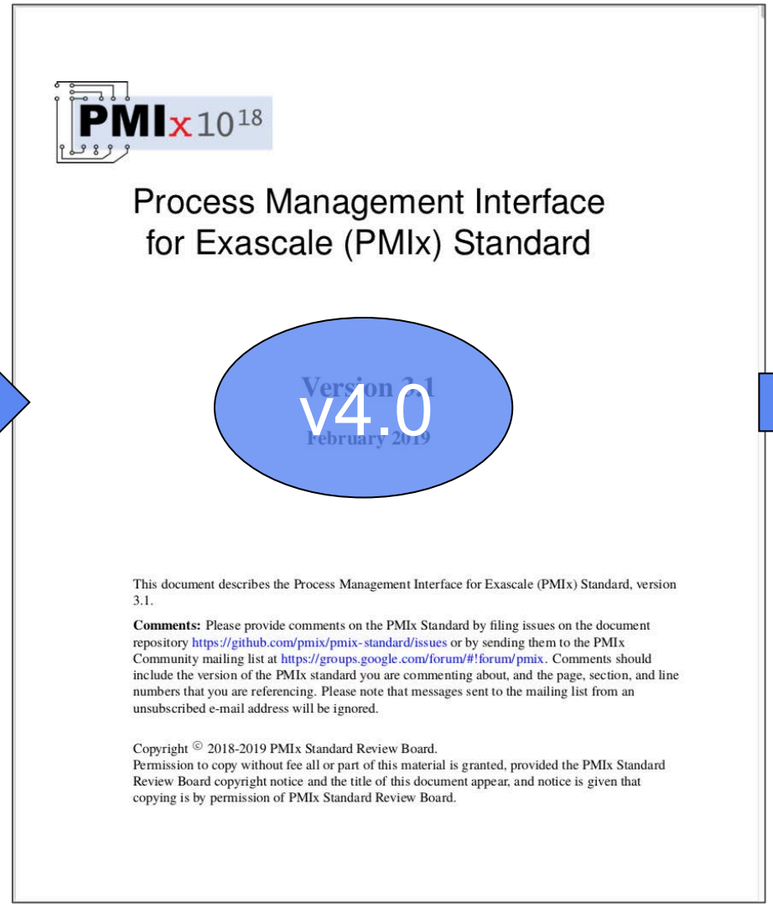
PMIx Standard document specifies the syntax and semantics of the PMIx API in addition to describing API rationale, providing advice to implementors and users, and protocol specifications for complex API interactions for, for example, debuggers.

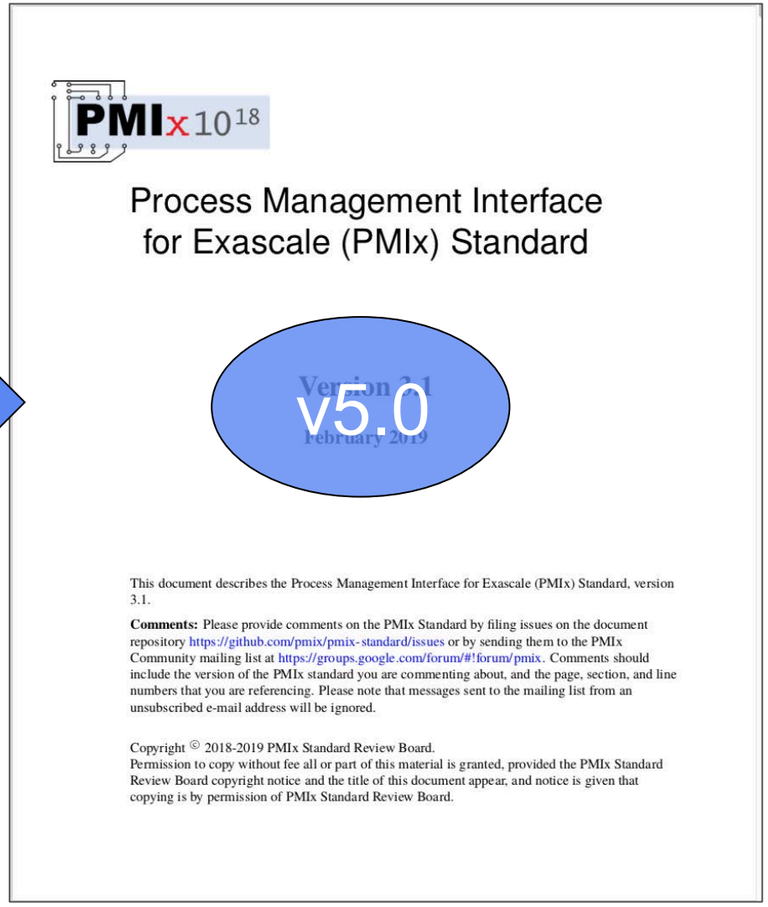
- Governed by the PMIx Administrative Steering Committee (ASC) with representation from a wide range of research, academic, and industrial organizations.
  - Meets quarterly to vote on proposals to move the PMIx standard forward.
  - Currently working on PMIx Standard v5

https://github.com/pmix/pmix-standard
https://github.com/pmix/governance

PMIx Standard







Feb. 2019 **1H 2020** **2H2020**

PMIx Standard v4.0

- Feature set:
  - Attribute queries
  - Process groups/sets
  - Network coordinates/topology
  - Extended tool/debugger support
  - Python bindings
  - Scheduler, query access to fabric info.

PMIx Standard v5.0

- Feature set: *(under development)*
  - Re-organization of the document
    - Use case slicing
    - Clearly distinguish client / server / tool sets of interfaces
    - A more implementation agnostic document
  - Any interfaces ready from the working groups

Implementation Agnostic Working Group

Working Group Champion: David Solt (IBM)

Implementation Agnostic Working Group

- **Goal**: Review the PMIx standard and rework text which assumes or requires a specific implementation of the standard
  - Multiple implementations of PMIx should be encouraged
  - Text should not assume a particular implementation architecture or design
  - Make a clear division between client/tool interfaces and system management software interfaces
  - Broaden scope of PMIx from HPC only to distributed computing

Why?

- Current doc was written for both client developers and System Software developers implementing the “back end” of OpenPMIx
- Example of current text:
  - The PMIx Standard defines and describes the interface developed by the PMIx Reference Implemenation (PRI). Much of this document is specific to the PRI's design and implementation. Specifically the standard describes the functionality provided by the PRI, and what the PRI requires of the clients and Resource Managers that use it's interface.
- Want to transition the document to its role as a standard

About this effort

- To follow the Chapter 1 Pull Request (the main focus of our work to date)

  https://github.com/pmix/pmix-standard/pull/192

- To join the WG, subscribe to and look for meeting notices

  https://groups.google.com/forum/#!forum/pmix-forum-wg-impl-agnostic

Current Status

- Finishing a substantial rework to Chapter 1 (Introduction)
- Began reviewing Chapter 2 (PMIx Terms and Conventions)
- Planning for how to best organize remaining chapters to be more usable by distinct readers: client developers and back end developers.

Welcome Participation

- Very interested in a few more people to join us.
- Lots of work to do.
- The less you know about PMIx the better!
- Covet the perspective of someone reading text for the first time with no prior knowledge

https://groups.google.com/forum/#!forum/pmix-forum-wg-impl-agnostic

Functionality Working Group

Working Group Champion: Stephen Herbein (LLNL)

Executive Summary

- **Problem**: Given the size and structure of the PMIx standard document, it can be difficult to find the PMIx components necessary for a given use-case
- **Goal:** Provide a mechanism for focusing in on the aspects of the standard that are of interest to a particular user/use-case
- **Proposed Solution**: compile a list of high-level use-cases and the PMIx capabilities required to support them
  - **Answer the question:** "To do _____ with PMIx, what interfaces should I use and my resource manager support?"

Goals

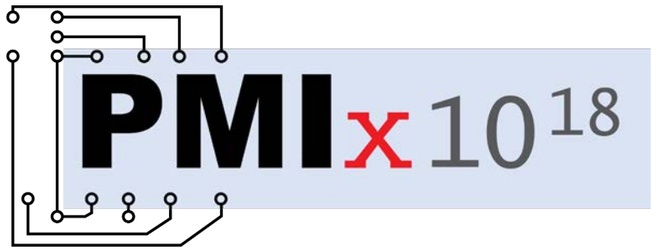
- **Short Term:**
  - Use-Case Milestone 1 (Jan 2020):
    - bootstrap (business card exchange)
    - debuggers / tools
  - Use-Case Milestone 2 (Q1/Q2 2020):
    - hybrid programming models
    - instant-on
    - workflow/application management

Goals

- **Long Term:**
  - Add interfaces for querying "use-case support"
  - Identify and document emerging use-cases
    - cross-version compatibility
    - fault tolerance
    - groups/flexible allocations
    - power control
    - tiered storage

Call for Participation

- Help Wanted
  - Identifying use-cases that we've missed
  - Reviewing documented use-cases
  - Documenting identified use-cases
- How to Join
  - Subscribe to our Google Group mailing list
    - pmix-forum-wg-func-slices
  - Weekly call
    - Wednesdays @ 9am PT / 12pm ET
    - Webex info sent via mailing list
    - Email me for a calendar invite (herbein1@llnl.gov)



Storage Working Group

Shane Snyder

Argonne National Lab

WG Mission

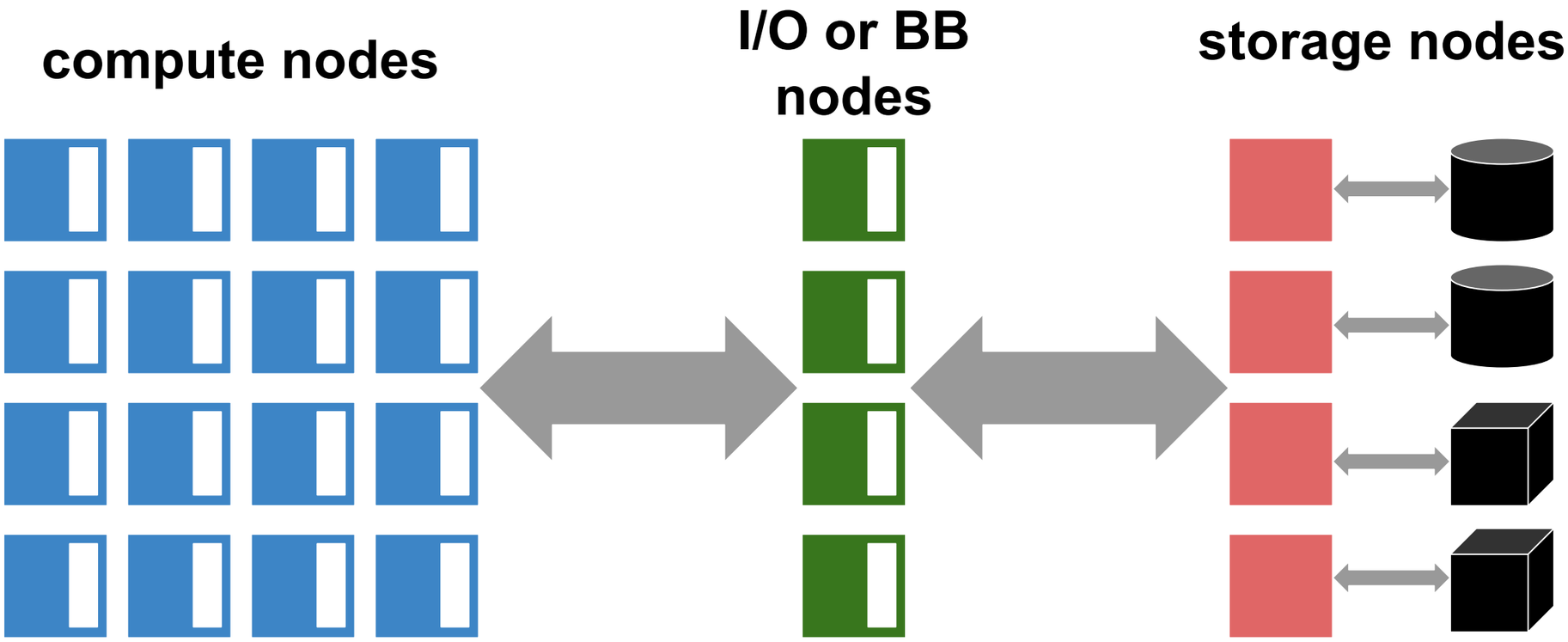
- Extend PMIx standard APIs to support application, I/O middleware, and WLM/RM interaction with HPC storage hierarchies
  - Embrace PMIx philosophy of flexibility by abstracting vendor-specific approaches, ensuring their independence
- Possible use cases:
  - data staging/unstaging orchestrated by WLMs (like Cray DataWarp)
  - general asynchronous data movement across storage layers
  - scalable shared library loading
  - interrogation of storage hierarchy resources and their characteristics
  - ...

WG Mission

- Extend PMIx standard APIs to support application, I/O middleware, and WLM/RM interaction with HPC storage hierarchies
  - Embrace PMIx philosophy of flexibility by abstracting vendor-specific approaches, ensuring their independence
- Possible use cases:
  - data staging/unstaging orchestr
  - general asynchronous data m
  - scalable shared library loading
  - interrogation of storage hierarc
  - ...

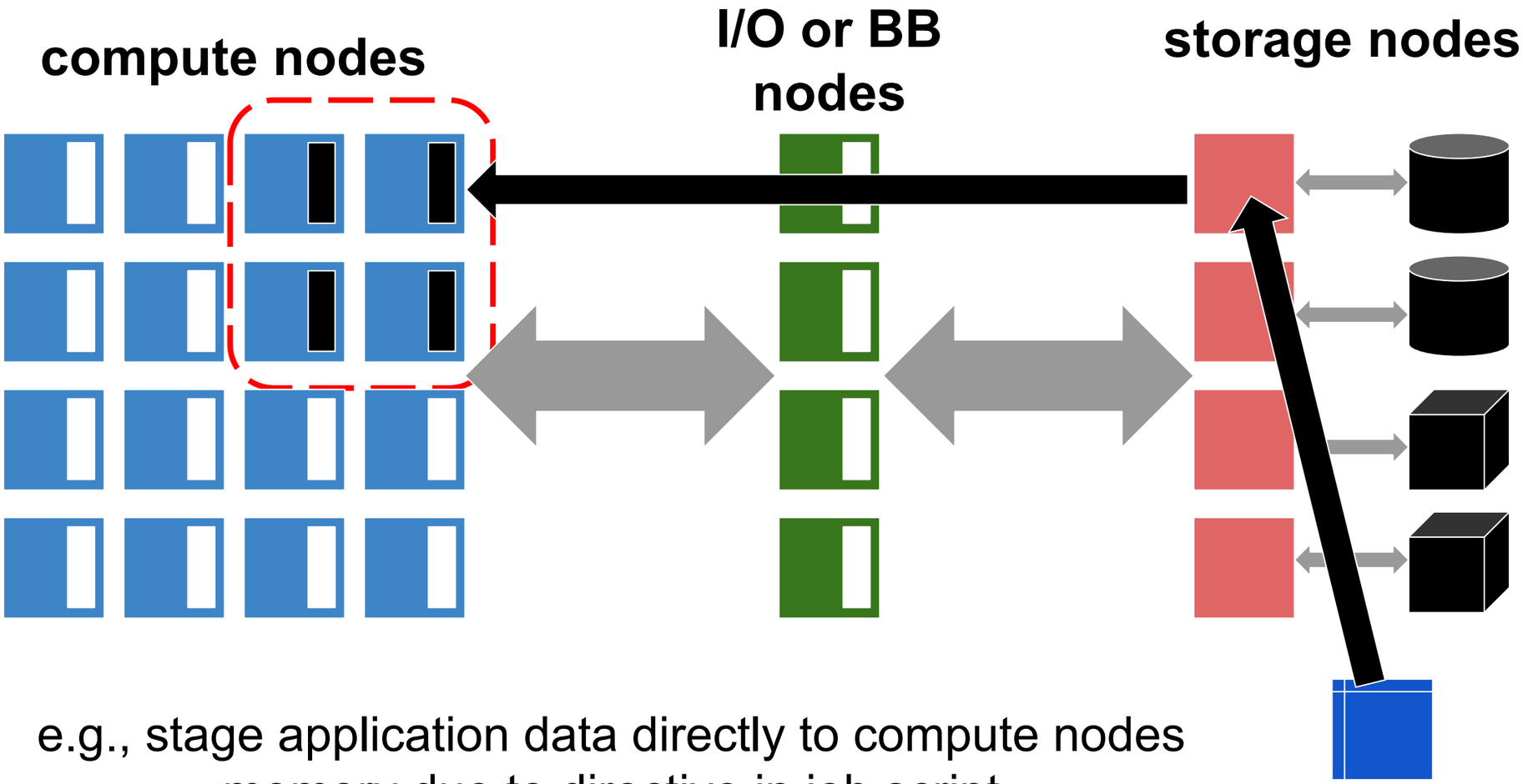
Part of the challenge will be prioritizing which core storage mechanism we want to standardize

Use case: staging/unstaging



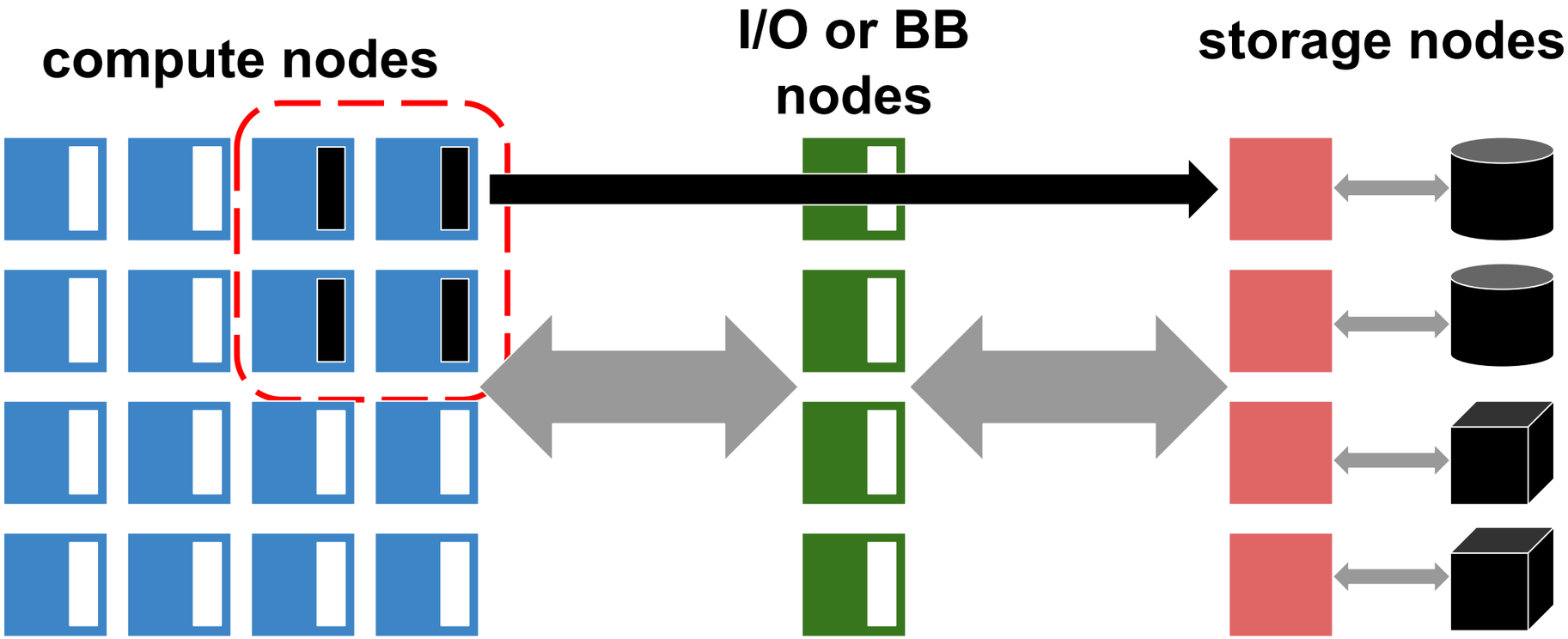
Growing HPC storage hierarchy creates flexible opportunities for staging/unstaging application data

Use case: staging/unstaging



e.g., stage application data directly to compute nodes memory due to directive in job script

Use case: staging/unstaging



e.g., unstaging application data due to job termination or due to runtime request from job

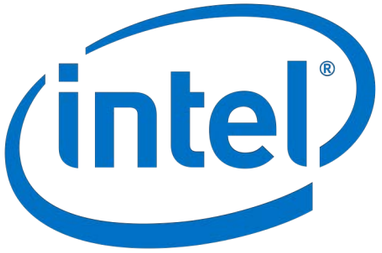
Current status

- This WG is just starting, so there's plenty of opportunity to get involved and contribute
  - Aiming for initial meeting sometime in December
- Seeking a diversity of participants to shape WG findings
  - I/O library developers
  - storage system developers
  - workflow system developers
  - vendors
- Please talk to me if you are interested in participating!
  - my email: ssnyder@mcs.anl.gov
  - WG mailing list: pmix-forum-wg-storage@googlegroups.com

Dynamic Workflows Working Group

Jai Dayal

jai.dayal@intel.com



Dynamic Workflows: Mission

- Many applications have dynamic resource requirements
  - Requirements can change for many reasons
    - Performance, interesting data, faults
  - Programming paradigm can drive resource needs
    - Compute optimal resource balance based on discovery of capabilities vs problem space
- WG Goal
  - Define abstracted, portable methods for WLM/RM-App coordinated workflow operations

Use Case: Cloud/HPC Mgmt

- Cloud scheduling built around streaming apps
  - Can shift around as needed by simple kill/replace
  - Location is arbitrary – no “huddling” around network required
- HPC workloads
  - Care about relative location
  - Employ various relocation strategies
    - Checkpoint/restart, data coalesce and respawn, …
    - Must preserve state across relocation/restart
  - Increasingly dynamic resource utilization
    - Creates scheduling “bubbles”

Use Case: Cloud/HPC Mgmt

- Cloud scheduling built around streaming apps
  - Can shift around as needed by simple kill/replace
  - Location is arbitrary – no "huddl… …uired
- HPC workloads
  - Care about relative locati…
  - Employ various relocation…
    - Checkpoint/restart, data…
    - Must preserve state across reloc…
  - Increasingly dynamic resource utilization
    - Creates scheduling "bubbles"

RM-agnostic methods for…
Declaring ability/mechanism for relocation
Handshake relocation coordination
Allocation requests (expansion, return)

Use-Case: Spark-like

- Spark on HPC systems
  - Allocate dedicated “island”
    - Max size that might be needed
  - Start one process, compute needs “as you go”
    - Low utilization score
- Need RM-app coordination
  - Request allocations as needed
  - Return resources
    - No longer needed
    - Not needed for a while

Current Status

- Call for participation (now)
- Some relevant APIs/attributes in Standard
  - Allocate, job control
  - Need to evaluate, adjust, augment
- Promote adoption
  - Example paradigms
    - Spark, Tensorflow, ADIOS, Dataspaces?
  - Example environments
    - Kubernetes, others?
- Please talk to me if you are interested in participating!
  - My email: jai.dayal@intel.com
  - WG mailing list: pmix-forum-wg-workflows@googlegroups.com

Agenda

- State of the Community
  - Ralph H. Castain (Intel)
- PMIx Standard
  - Josh Hursey (IBM)
  - Kathryn Mohror (LLNL)
- Q&A