openHPC
http://openhpc.community

Meeting of the Technical Steering Committee (TSC) Board

Tuesday, October 10th
11:00am ET

Meeting Logistics

- https://www.uberconference.com/jeff_ef

- United States : +1 (510) 224-9559 (No PIN needed).

Antitrust Policy Notice

- Linux Foundation meetings involve participation by industry competitors, and it is the intention of the Linux Foundation to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of, and not participate in, any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.

- Examples of types of actions that are prohibited at Linux Foundation meetings and in connection with Linux Foundation activities are described in the Linux Foundation Antitrust Policy available at http://www.linuxfoundation.org/antitrust-policy. If you have questions about these matters, please contact your company counsel, or if you are a member of the Linux Foundation, feel free to contact Andrew Updegrove of the firm of Gesmer Updegrove LLP, which provides legal counsel to the Linux Foundation.

THE LINUX FOUNDATION

Agenda

- SC'17 BoF Update
  - Schedule update: Wednesday, Nov 15: 12:15-13:15
- Linaro Connect
  - ARM Data Center Day (Sep 28)
  - multiple references to OpenHPC in talks (Riken, Sandia, Renato/Linaro)
- Review Cycle #4
- Recipe default update
- PMIx

Review cycle #4

- Reviews for Cycle #4 are now complete

Component Name	# of Reviewers	# of accepts	# of rejects	Avg. Priority
mpi4py	7	7	0	7.4
LIKWID	7	6(3*)	1	5.3

* w/ changes

- Propose to target mpi4py for v1.3.3 release
  - baseline start from PR submitted by requester
- Propose to target LIKWID for future release (e.g. v1.3.4)
  - majority of discussion around LIKWID was on different methods to access H/W counter info
    - direct access from user
    - access via setuid daemon
    - perf_event backend in kernel
  - I believe general consensus for those voting to accept was to highlight use of the daemon
    - explicitly call out that this is setuid

Recipe default update for ethernet

- An item we've had on the back burner from earlier this year (April) is to update the recipe to be more friendly to ethernet-only environment
  - think we've done a reasonable job calling out which MPI stacks are applicable to a given interconnect, but some users have been confused about what other items are optional in ethernet-only environment
- Current recipes assume IB environment and call out separate mods if using OPA
- Updated approach targeted for v1.3.3
  - target ethernet environment and call out IB or OPA via optional stanzas

```
if [[ ${enable_ib} -eq 1 ]];then
     yum -y groupinstall "InfiniBand Support"
     yum -y install infinipath-psm
     systemctl start rdma
fi
...
if [[ ${enable_opa} -eq 1 ]];then
     yum -y install opa-basic-tools
     systemctl start rdma
fi
```

Recipe default update for ethernet

- This approach does necessitate a change in suggesting a default MPI stack
- Recall that we provide a variety of lmod-default variants so an admin can choose what stack is default in their environment
- From 1.3.2 docs:

4.5 Setup default development environment

System users often find it convenient to have a default development environment in place so that compilation can be performed directly for parallel programs requiring MPI. This setup can be conveniently enabled via modules and the OpenHPC modules environment is pre-configured to load an `ohpc` module on login (if present). The following package install provides a default environment that enables autotools, the GNU compiler toolchain, and the MVAPICH2 MPI stack.

```
[sms]# yum -y install lmod-defaults-gnu7-mvapich2-ohpc
```

Tip

If you want to change the default environment from the suggestion above, OpenHPC also provides the GNU compiler toolchain with the OpenMPI and MPICH stacks:

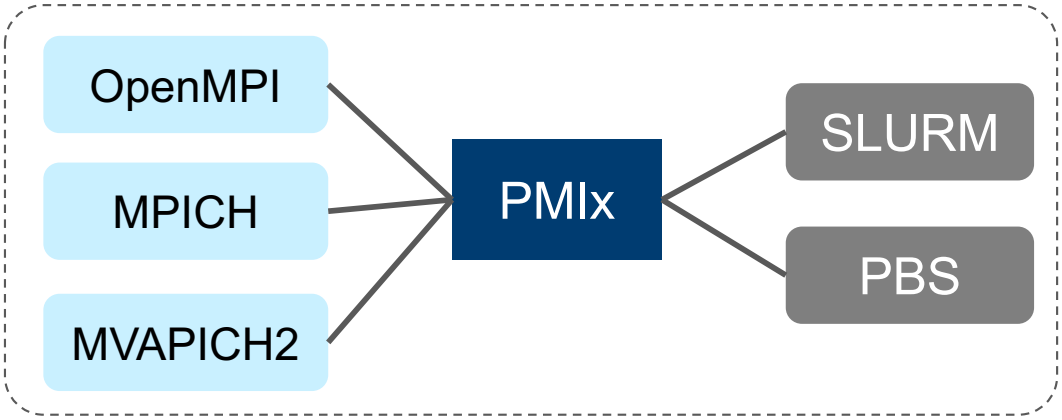
- lmod-defaults-gnu7-openmpi-ohpc
- lmod-defaults-gnu7-mpich-ohpc

- MVAPICH2 targets IB/OPA so not appropriate for default in ethernet recipe. Today, avail options are OpenMPI or MPICH
- Go with OpenMPI?

PMIx

- Another item on the backburner from Review Cycle #3 is the inclusion and usage of PMIx
- Reminder on long term intent: resource managers adopting PMIx over time so we can leverage as unified job launch support library
- SLURM has started to include PMIx in it's distribution, but prefer external build for use by multiple resource managers in the future

Desired End Game



PMIx

- After working on this for current release, have some issues and subtleties to discuss on how we would like to proceed
- Current state:
  - latest SLURM supports PMIx 1.x (support for linkage against external build or use distribution included with SLURM)
  - latest OpenMPI (3.x) supports native use of PMIx
  - PBS committed to supporting PMIx at some point, but not available yet. We continue to use latest PBS Pro open source release (June 2016)
  - MPICH and MVAPICH2 do not have direct PMIx support, but have capability to use PMI1 or PMI2 (typically the variants provided by SLURM)
    - PMIx provides backwards compatibility for PMI1/2 via conversion to PMIx equivalents
    - intent has been to leverage these to gain support for MPICH variants

PMIx

- The good news:
  - OpenMPI (3.x) works
    - can have single build of OpenMPI that uses PMIx under SLURM, no PMIx under PBS Pro
    - prun launcher updated to launch via `srun --mpi=pmix`
  - thanks to Adrian, can also have single OpenMPI build support ethernet, IB, and OPA

- The ok news:
  - can get MPICH builds to use PMI interfaces provided by PMIx to run under SLURM
  - initial testing showed failures with singleton (ie ./a.out); reported to PMIx developer and have patch in place to resolve that
  - enabling PMI-based job launch with MPICH disables standalone job launch (ie. no mpiexec.hydra). Even if you build/install that by hand, unable to use (PMI provides a standard API for client/server, but does not specify protocol)
    - hence, cannot have single PMIx-enabled MPICH build for both SLURM/PBS today
    - we presumably could once we have a PBS release that supports PMIx
    - added environment variable to resulting MPICH module to indicate pmix support (so prun can act accordingly):
      - setenv("OHPC_MPI_LAUNCHERS","pmix")

PMIx

- The bad news:
  - at present, cannot get MVAPICH2 to work using PMI interfaces provided by PMIX
    - iterating with PMIx developer, not clear this will work, but maybe....
    - if it does, will have similar issue of no longer having mpiexec.hydra available for use with PBS
    - likely require separate build for PBS

- Other items to consider
  - to date, our MPI stacks have no formal requirements on the resource manager
  - if we introduce PMIx in an update, we have to be careful to account for existing systems that have SLURM/PBS installed
  - in particular, there is an implied version requirement for OpenMPI and MPICH to use PMIx (ie. the new build of SLURM with PMIx support enabled is needed)
  - having a single build of the MPI stack then becomes problematic as it would pull in resource manager as well

PMIx - potential packaging options

- Option #1
  - leave current MPI build configurations mostly as is (without PMIx), but enable additional variant(s) for MPIs where it works, e.g. openmpi3-pmix-slurm-gnu7-ohpc, mpich-pmix-slurm-gnu7-ohpc
- Option #2:
  - move to convention where MPI stacks are RMS specific and add formal version requirements (w/ PMIx enabled)
  - e.g. mpich-slurm-gnu7-ohpc, mpich-pbs-gnu7-ohpc
  - potential gotcha: folks would be unable to use some of the MPI stacks outside of resource manager (ie. no mpiexec.hydra for use with hostfile/ssh)
- Option #3
  - hold off for now: wait till PMIx supported by both resource managers
- Option #4
  - don't add formal requirements but call out the need to upgrade SLURM in release notes if desiring to use latest MPI builds
- Thoughts/discussion?