Simplify System Software Stack Development and Maintenance

Jeff Adams
Intel HPC Platform Software Product Manager
Data Center Solutions Group

Courtesy of openHPC
Session Agenda and Objective

• Why a community system software stack?
• The HPC system software problems we all share
• The OpenHPC community
• The OpenHPC software stack
• How to get involved
Current State of System Software Efforts

Fragmented efforts across the ecosystem – “Everyone building their own solution.”

A desire to get exascale performance & speed up software adoption of hardware innovation

New complex workloads (ML, Big Data, etc.) drive more complexity into the software stack

THE REALITY: We will not be able to get where we want to go without a major change in system software development
Community Effort to Realize Future State

A Shared Repository

- Security
- Power Management
- System Performance Monitoring
- Application Libraries
- Parallel File Systems
- Dev Tools

Stable HPC Platform Software that:

- Fuels a vibrant and efficient HPC software ecosystem
- Takes advantage of hardware innovation & drives revolutionary technologies
- Eases traditional HPC application development and testing at scale
- Extends to new workloads (ML, analytics, big data)
- Accommodates new environments (i.e., cloud)
OpenHPC History

**ISC ’15**
‘Birds of a Feather’ (BoF) discussion about the merits of and interest in a community supported HPC repository and management framework

**SC ’15**
Follow-on BoF for a comprehensive open community HPC software stack

**Linux* Foundation**
Working group collaborating to define participation agreement, initial governance structure and solicit volunteers

**Linux Foundation**
Announces technical, leadership and member investment milestones with founding members and formal governance structure
OpenHPC Community Mission and Vision

**Mission**
To provide a reference collection of open-source HPC software components and best practices, lowering barriers to deployment, advancement, and use of modern HPC methods and tools.

**Vision**
OpenHPC components and best practices will enable and accelerate innovation and discoveries by broadening access to state-of-the-art, open-source HPC methods and tools in a consistent environment, supported by a collaborative, worldwide community of HPC users, developers, researchers, administrators, and vendors.
A Linux Foundation Collaborative Project

WWW.OpenHPC.Community
OpenHPC Stack Overview
## OpenHPC Components—v1.3.4

<table>
<thead>
<tr>
<th>Functional Areas</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base OS</td>
<td>CentOS 7.4, SLES12 SP3</td>
</tr>
<tr>
<td>Architecture</td>
<td>x86_64, aarch64</td>
</tr>
<tr>
<td>Administrative Tools</td>
<td>Conman, Ganglia, Lmod, LosF, Nagios, pdsh, pdsh-mod-slurm, prun, EasyBuild, ClusterShell, mrsh, Genders, Shine, Spack, test-suite</td>
</tr>
<tr>
<td>Provisioning</td>
<td>Warewulf, xCAT</td>
</tr>
<tr>
<td>Resource Mgmt.</td>
<td>SLURM, Munge, PBS Professional, PMIx</td>
</tr>
<tr>
<td>Runtimes</td>
<td>OpenMP, OCR, Singularity</td>
</tr>
<tr>
<td>I/O Services</td>
<td>Lustre client, BeeGFS client</td>
</tr>
<tr>
<td>Numerical/Scientific Libraries</td>
<td>Boost, GSL, FFTW, Hypre, Metis, Mumps, OpenBLAS, PETSc, PLASMA, Scalapack, Scotch, SLEPc, SuperLU, SuperLU_Dist, Trilinos</td>
</tr>
<tr>
<td>I/O Libraries</td>
<td>HDF5 (pHDF5), NetCDF (including C++ and Fortran interfaces), Adios</td>
</tr>
<tr>
<td>Compiler Families</td>
<td>GNU (gcc, g++, gfortran), Clang/LLVM</td>
</tr>
<tr>
<td>MPI Families</td>
<td>MVAPICH2, OpenMPI, MPICH</td>
</tr>
<tr>
<td>Development Tools</td>
<td>Autotools, cmake, hwloc, mpi4py, R, SciPy/NumPy, Valgrind</td>
</tr>
<tr>
<td>Performance Tools</td>
<td>PAPI, IMB, mpiP, pdtoolkit TAU, Scalasca, ScoreP, SIONLib</td>
</tr>
</tbody>
</table>
Basic Cluster Install Example

- Starting install guide/recipe targeted for flat hierarchy
- Leverages image-based provisioner: Warewulf or xCAT
  - PXE boot (stateful or stateless)
  - Optionally connect external Lustre file system
- Need hardware-specific information to support (remote) bare-metal provisioning
OpenHPC Development Infrastructure

• The ‘usual’ software engineering stuff:
  • GitHub (SCM and issue tracking/planning)
  • Continuous Integration (CI) Testing (Jenkins)
  • Documentation (Latex)

• Capable build/packaging system
  • At present: we target a common delivery/access mechanism that adopts Linux sysadmin familiarity
  • Require Flexible System to manage builds
  • A system using Open Build Service (OBS) supported by back-end git

OpenHPC Build System: OBS

- Manage Build Process
- Drive Builds for multiple repositories
- Repeatable builds
- Generate binary and src rpms
- Publish corresponding package repositories
- Client/server architecture supports distributed build slaves and multiple architectures
OpenHPC Integration/Testing/Validation

- Install recipes
- Cross-package interaction
- Development environment
- Mimic use cases common in HPC deployments
- Upgrade mechanism
OpenHPC Integration/Test/Validation

- Standalone integration test infrastructure
- Families of tests that can be used during:
  - Initial install process
  - Post-install process
  - Tests that touch all of the major components installed
- Expectation is that each new component included will need corresponding integration test collateral
- Integration tests are included in the GitHub repo
- Global testing harness includes a number of embedded subcomponents
OpenHPC New Software Requests & Additions

• Submission site: https://github.com/openhpc/submissions
• Quarterly reviews
• Next deadline: December 8, 2017
• 18 submissions accepted since launch
OpenHPC Community BoF Event

Wednesday, November 15\textsuperscript{th}
12:15 – 1:15
Location: 507
OpenHPC Booth: 395

[Northwest corner]
Thank you!

Courtesy of openHPC

Jeff Adams
jeff.adams@intel.com
971.470.6701
Backup
OpenHPC Community Governance Overview

- Governing Board
- Board Representatives
  - Board Representatives for other Members, Indirect Representation (X:1)
- TSC Director
- TSC
  - Technical Project Leadership
  - Architecture, Component selection, releases, day-to-day tech work
- HPC Community
OpenHPC TSC Role Overview

OpenHPC
Technical Steering Committee (TSC)

- Project Leader
- Integration Testing Coordinator(s)
- Upstream Component Development Representative(s)
- End-User / Site Representative(s)
- Maintainers