OpenHPC: Project Overview and Updates

Karl W. Schulz, Ph.D.
Software and Services Group, Intel
Technical Project Lead, OpenHPC

5th Annual MVAPICH User Group (MUG) Meeting
August 16, 2017 • Columbus, Ohio
Outline

• Brief project overview
• New items/updates from last year
• Highlights from latest release
OpenHPC: 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.
OpenHPC: a brief History…

ISC’15
BoF on the merits/interest in a community effort

ISC’16
v1.1.1 release
Linux Foundation announces technical leadership, founding members, and formal governance structure

SC’15
seed initial 1.0 release, gather interested parties to work with Linux Foundation

SC’16
v1.2 release BoF

MUG’16
OpenHPC Intro

ISC’17
v1.3.1 release BoF

June 2015
Nov 2015
June 2016
Nov 2016
June 2017
OpenHPC Project Members

Mixture of Academics, Labs, OEMs, and ISVs/OSVs

Project member participation interest? Please contact Jeff ErnstFriedman jernstfriedman@linuxfoundation.org
Governance: Technical Steering Committee (TSC)

Role Overview

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

Note: We just completed election of TSC members for the 2017-2018 term.
- terms are for 1-year
OpenHPC TSC – Individual Members

- Reese Baird, Intel (Maintainer)
- David Brayford, LRZ (Maintainer)
- Eric Coulter, Indiana University (End-User/Site Representative)
- Leonordo Fialho, ATOS (Maintainer)
- Todd Gamblin, LLNL (Component Development Representative)
- Craig Gardner, SUSE (Maintainer)
- Renato Golin, Linaro (Testing Coordinator)
- Jennifer Green, Los Alamos National Laboratory (Maintainer)
- Douglas Jacobsen, NERSC (End-User/Site Representative)
- Chulho Kim, Lenovo (Maintainer)
- Janet Lebens, Cray (Maintainer)
- Thomas Moschny, ParTec (Maintainer)
- Nam Pho, New York Langone Medical Center (Maintainer)
- Cyrus Proctor, Texas Advanced Computing Center (Maintainer)
- Adrian Reber, Red Hat (Maintainer)
- Joseph Stanfield, Dell (Maintainer)
- Karl W. Schulz, Intel (Project Lead, Testing Coordinator)
- Jeff Schutkoske, Cray (Component Development Representative)
- Derek Simmel, Pittsburgh Supercomputing Center (End-User/Site Representative)
- Scott Suchyta, Altair (Maintainer)
- Nirmala Sundararajan, Dell (Maintainer)

New for 2017-2018

https://github.com/openhpc/ohpc/wiki/Governance-Overview
Target system architecture

- Basic cluster architecture: head node (SMS) + computes
- Ethernet fabric for mgmt. network
- Shared or separate out-of-band (BMC) network
- High speed fabric (InfiniBand, Omni-Path)
OpenHPC: a building block repository

[ Key takeaway ]

- OpenHPC provides a collection of pre-built ingredients common in HPC environments; fundamentally it is a package repository.

- The repository is published for use with Linux distro package managers:
  - yum (CentOS/RHEL)
  - zypper (SLES)

- You can pick relevant bits of interest for your site:
  - if you prefer a resource manager that is not included, you can build that locally and still leverage the scientific libraries and development environment.
  - similarly, you might prefer to utilize a different provisioning system.
Newish Items/Updates

changes and new items since we were last together at MUG’16
Switched from ISOs -> distribution tarballs

- For those who prefer to mirror a repo locally, we have historically provided an ISO that contained all the packages/repodata
- Beginning with v1.2 release, switched to tarball based distribution
- Distribution tarballs available at:
  
  http://build.openhpc.community/dist

- A “make_repo.sh” script is provided that will setup a locally hosted OpenHPC repository using the contents from downloaded tarball

```
# tar xf OpenHPC-1.2.1.CentOS_7.2_x86_64.tar
# ./make_repo.sh
--> Creating OpenHPC.local.repo file in /etc/yum.repos.d
--> Local repodata stored in /root/repo
```

```
# yum repolist | grep OpenHPC
OpenHPC-local          OpenHPC-1.2 - Base
OpenHPC-local-updates  OpenHPC-1.2.1 - Updates
```
More Generic Repo Paths

• Starting with the v1.3 release, we adopted more generic paths for underlying distros

[OpenHPC]
name=OpenHPC-1.3 - Base
baseurl=http://build.openhpc.community/OpenHPC:/1.3/CentOS_7
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-OpenHPC-1

[OpenHPC-updates]
name=OpenHPC-1.3 - Updates
baseurl=http://build.openhpc.community/OpenHPC:/1.3/updates/CentOS_7
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-OpenHPC-1

Similar approach for SLES12 repo config -> SLE_12
Release Roadmap Published

<table>
<thead>
<tr>
<th>Release</th>
<th>Target Release Date</th>
<th>Expectations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3.2</td>
<td>August 2017</td>
<td>New component additions and version upgrades.</td>
</tr>
<tr>
<td>1.3.3</td>
<td>November 2017</td>
<td>New component additions and version upgrades.</td>
</tr>
</tbody>
</table>

Previous Releases

A history of previous OpenHPC releases is highlighted below. Clicking on the version string will take you to the Release Notes for more detailed information on the changes in a particular release.

<table>
<thead>
<tr>
<th>Release</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3.1</td>
<td>June 16, 2017</td>
</tr>
<tr>
<td>1.3</td>
<td>March 31, 2017</td>
</tr>
<tr>
<td>1.2.1</td>
<td>January 24, 2017</td>
</tr>
<tr>
<td>1.2</td>
<td>November 12, 2016</td>
</tr>
<tr>
<td>1.1.1</td>
<td>June 21, 2016</td>
</tr>
<tr>
<td>1.1</td>
<td>April 18, 2016</td>
</tr>
<tr>
<td>1.0.1</td>
<td>February 05, 2016</td>
</tr>
<tr>
<td>1.0</td>
<td>November 12, 2015</td>
</tr>
</tbody>
</table>

• Have had some requests for a roadmap for future releases
• High-level roadmap now maintained on GitHub wiki: https://github.com/openhpc/ohpc/wiki/Release-History-and-Roadmap
Component Submission Site

• A common question posed to the project has been how to request new software components?

• We now have a simple submission site for new requests:
  – https://github.com/openhpc/submissions
  – requests reviewed on rolling basis at roughly a quarterly cadence

• Example software/recipes that have been added via this process:
  – xCAT
  – BeeGFS
  – PLASMA, SLEPc
  – clang/LLVM
  – MPICH
  – PBS Professional
  – Singularity
  – Scalasca
Opt-in System Registry Now Available

- Interested users can now register their usage on a public system registry
- Helpful for us to have an idea as to who is potentially benefitting from this community effort
- Accessible from top-level GitHub page

OpenHPC System Registry

This opt-in form can be used to register your system to let us (and the community) know that you are using elements of OpenHPC.

* Required

Name of Site/Organization *
Your answer

What OS distribution are you using? *
- CentOS/RHEL
- SLES
- Other: 

Site or System URL
Your answer
Multiple Architecture Builds

• Starting with v1.2 release, we also include builds for **aarch64**
  - both SUSE and RHEL/CentOS now have aarch64 variants available for latest versions (SLES 12 SP2, CentOS 7.3)

• Recipes/packages being made available as a **Tech Preview**
  - some additional work required for provisioning
  - significant majority of development packages testing OK, but there are a few known caveats
  - please see [https://github.com/openhpc/ohpc/wiki/ARM-Tech-Preview](https://github.com/openhpc/ohpc/wiki/ARM-Tech-Preview) for latest info

<table>
<thead>
<tr>
<th>Base OS</th>
<th>x86_64</th>
<th>aarch64</th>
<th>noarch</th>
</tr>
</thead>
<tbody>
<tr>
<td>CentOS 7.3</td>
<td>424</td>
<td>265</td>
<td>32</td>
</tr>
<tr>
<td>SLES 12 SP2</td>
<td>429</td>
<td>274</td>
<td>32</td>
</tr>
</tbody>
</table>

**v1.3.1 RPM counts**
Dev Environment Consistency

OpenHPC providing consistent development environment to the end user across multiple architectures.

```
karl@sms001:--> module avail

------- /opt/ohpc/pub/moduledeps/gnu-mpich -------

ados/1.10.0  mpiP/3.4.1  petsc/3.7.0  scorep/3.0
boost/1.61.0  mumps/5.0.2  phdf5/1.8.17  sionlib/1.7.0
fftw/3.3.4   netcdf/4.4.1  scalapack/2.0.2  superlu_dist/4.2
hypre/2.10.1  netcdf-cxx/4.2.1  scalasca/2.3.1  tau/2.25.2
imb/4.1      netcdf-fortran/4.4.4  scipy/0.18.0  trilinos/12.6.4

------- /opt/ohpc/pub/modulefiles/gnu -------

R_base/3.3.1  metis/5.1.0  numpy/1.11.1  openmpi/1.10.4
 GSL/2.2.1    mpich/3.2  (L)  ocr/1.0.1  pdtoolkit/3.22
 HDF5/1.8.17  mvapich2/2.2  openblas/0.2.19  superlu/5.2.1

------- /opt/ohpc/pub/moduledeps/gnu -------

ados/1.10.0  mpiP/3.4.1  petsc/3.7.0  scorep/3.0
boost/1.61.0  mumps/5.0.2  phdf5/1.8.17  sionlib/1.7.0
fftw/3.3.4   netcdf/4.4.1  scalapack/2.0.2  superlu_dist/4.2
hypre/2.10.1  netcdf-cxx/4.2.1  scalasca/2.3.1  tau/2.25.2
imb/4.1      netcdf-fortran/4.4.4  scipy/0.18.0  trilinos/12.6.4

------- /opt/ohpc/pub/modulefiles/gnu -------

R_base/3.3.1  metis/5.1.0  numpy/1.11.1  openmpi/1.10.4
 GSL/2.2.1    mpich/3.2  (L)  ocr/1.0.1  pdtoolkit/3.22
 HDF5/1.8.17  mvapich2/2.2  openblas/0.2.19  superlu/5.2.1

------- /opt/ohpc/pub/modulefiles/gnu -------

EasyBuild/2.9.0  clustershell/1.7.2  ohpc  (L)  prun/1.1  (L)
automtools  (L)  gnu/5.4.0  (L)  papi/5.4.3  valgrind/3.11.0
```

x86_64

aarch64
End-user software addition tools

OpenHPC repositories include two additional tools that can be used to further extend a user’s development environment

- EasyBuild and Spack
- leverages other community efforts for build reproducibility and best practices for configuration
- modules available for both after install

```bash
# module load gcc7
# module load spack
# spack compiler find
==> Added 1 new compiler to /root/.spack/linux/compilers.yaml gcc@7.1.0
# spack install darshan-runtime
... # . /opt/ohpc/admin/spack/0.10.0/share/spack/setup-env.sh 
# module avail
----- /opt/ohpc/admin/spack/0.10.0/share/spack/modules/linux-sles12-x86_64 ----- 
  darshan-runtime-3.1.0-gcc-7.1.0-vhd5hhg  m4-1.4.17-gcc-7.1.0-7jd575i
  hwloc-1.11.4-gcc-7.1.0-u3k6dok       ncurses-6.0-gcc-7.1.0-13mdumo
  libelf-0.8.13-gcc-7.1.0-tsgwr7j       openmpi-2.0.1-gcc-7.1.0-5imqlfb
  libpciaccess-0.13.4-gcc-7.1.0-33gbduz pkg-config-0.29.1-gcc-7.1.0-dhbp2a2i
  libsigsegv-2.10-gcc-7.1.0-1j5rntg    util-macros-1.19.0-gcc-7.1.0-vkdpa3t
  libtool-2.4.6-gcc-7.1.0-ulicbkz      zlib-1.2.10-gcc-7.1.0-gy4dtna
```
Variety of recipes now available
Choose your own adventure…

Initially, we started off with a single recipe with the intent to expand.

Latest v1.3.1 release continues to expand with multiple resource managers, OSes, provisioners, and architectures:
• Install_guide-CentOS7-Warewulf-PBSPro-1.3.1-x86_64.pdf
• Install_guide-CentOS7-Warewulf-SLURM-1.3.1-aarch64.pdf
• fInstall_guide-CentOS7-Warewulf-SLURM-1.3.1-x86_64.pdf
• Install_guide-CentOS7-xCAT-SLURM-1.3.1-x86_64.pdf
• Install_guide-SLE_12-Warewulf-PBSPro-1.3.1-x86_64.pdf
• Install_guide-SLE_12-Warewulf-SLURM-1.3.1-aarch64.pdf
• fInstall_guide-SLE_12-Warewulf-SLURM-1.3.1-x86_64.pdf

• Additional resource manager (PBS Professional) with v1.2
• Additional provisioner (xCAT) with v1.3.1
Template scripts

Template recipe scripts are proved that encapsulate commands presented in the guides:

```bash
# yum/zypper install docs-ohpc

# ls /opt/ohpc/pub/doc/recipes/*//*//*//*/*/*.recipe.sh
/opt/ohpc/pub/doc/recipes/centos7/aarch64/warewulf/slurm/recipe.sh
/opt/ohpc/pub/doc/recipes/centos7/x86_64/warewulf/pbspro/recipe.sh
/opt/ohpc/pub/doc/recipes/centos7/x86_64/warewulf/slurm/recipe.sh
/opt/ohpc/pub/doc/recipes/centos7/x86_64/xcat/slurm/recipe.sh
/opt/ohpc/pub/doc/recipes/sles12/aarch64/warewulf/slurm/recipe.sh
/opt/ohpc/pub/doc/recipes/sles12/x86_64/warewulf/pbspro/recipe.sh
/opt/ohpc/pub/doc/recipes/sles12/x86_64/warewulf/slurm/recipe.sh

# ls /opt/ohpc/pub/doc/recipes/*/input.local
/opt/ohpc/pub/doc/recipes/centos7/input.local
/opt/ohpc/pub/doc/recipes/sles12/input.local

# compute hostnames
c_name[0]=c1
c_name[1]=c2
...
# compute node MAC addresses
c_mac[0]=00:1a:2b:3c:4f:56
c_mac[1]=00:1a:2b:3c:4f:56
...
```

input.local + recipe.sh == installed system
Test Suite

• Initiated from discussion/requests at SC’16 BoF, the OpenHPC test suite is now available as an installable RPM (introduced with v1.3 release)

• \# yum/zypper install test-suite-ohpc
  - creates/relies on “ohpc-test” user to perform user testing (with accessibility to run jobs through resource manager)
  - related discussion added to recipes in Appendix C

```
[sms]$ su - ohpc-test
[test@sms ~]$ cd tests
[test@sms ~]$ ./configure --disable-all --enable-fftw
checking for a BSD-compatible install... /bin/install -c
checking whether build environment is sane... yes
...
---------------------------------------------- SUMMARY ---------------------------------------------

Package version................ : test-suite-1.3.0

Build user....................... : ohpc-test
Build host....................... : sms001
Configure date................... : 2017-03-24 15:41
Build architecture.............. : x86_64
Compiler Families............... : gnu
MPI Families.................... : mpich mvapich2 openmpi
```
Project CI infrastructure

• TACC is kindly hosting some CI infrastructure for the project (Austin, TX)
• Using for build servers and continuous integration (CI) testbed.

http://test.openhpc.community:8080

Many thanks to TACC and vendors for hardware donations!!: Intel, Cavium, Dell
Community Test System for CI in use

http://test.openhpc.community:8080

OpenHPC CI Infrastructure
Thanks to the Texas Advanced Computing Center (TACC) for hosting support and to Intel, Cavium, and Dell for hardware donations.

<table>
<thead>
<tr>
<th>S</th>
<th>Name</th>
<th>Last Success</th>
<th>Last Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3</td>
<td>(1.3 to 1.3.1) - (centos7.3,x86_64) - (warewulf+slurm)</td>
<td>2 days 12 hr - #41</td>
<td>1 hr 12 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (centos7.3,x86_64) - (warewulf+pbspro) - UEFI</td>
<td>2 days 7 hr - #390</td>
<td>56 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (centos7.3,x86_64) - (warewulf+slurm) - long cycle</td>
<td>2 days 3 hr - #892</td>
<td>1 hr 2 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (centos7.3,x86_64) - (warewulf+slurm) - tarball REPO</td>
<td>2 days 5 hr - #48</td>
<td>1 hr 2 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (centos7.3,x86_64) - (warewulf+slurm+PSXE)</td>
<td>2 days 4 hr - #726</td>
<td>2 hr 14 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (centos7.3,x86_64) - (warewulf+slurm+PSXE+OPA)</td>
<td>2 days 7 hr - #80</td>
<td>1 hr 51 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (centos7.3,x86_64) - (xcat+slurm)</td>
<td>2 days 13 hr - #271</td>
<td>1 hr 1 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (sles12sp2,x86_64) - (warewulf+pbspro) - tarball REPO</td>
<td>2 days 8 hr - #45</td>
<td>44 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (sles12sp2,x86_64) - (warewulf+pbspro) - UEFI</td>
<td>2 days 4 hr - #70</td>
<td>45 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (sles12sp2,x86_64) - (warewulf+slurm)</td>
<td>2 days 8 hr - #780</td>
<td>53 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (sles12sp2,x86_64) - (warewulf+slurm+PSXE) - long cycle</td>
<td>2 days 3 hr - #114</td>
<td>1 hr 53 min</td>
</tr>
<tr>
<td>1.3</td>
<td>(1.3.1) - (sles12sp2,x86_64) - (warewulf+slurm+PSXE+OPA)</td>
<td>2 days 8 hr - #36</td>
<td>1 hr 33 min</td>
</tr>
</tbody>
</table>

All recipes exercised in CI system (start w/ bare-metal installs + integration test suite)
OpenHPC v1.3.1 Release
June 16, 2017
OpenHPC v1.3.1 - Current S/W components

<table>
<thead>
<tr>
<th>Functional Areas</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base OS</td>
<td>CentOS 7.3, SLES12 SP2</td>
</tr>
<tr>
<td>Architecture</td>
<td>x86_64, aarch64 (Tech Preview)</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</td>
</tr>
<tr>
<td>Runtimes</td>
<td>OpenMP, OCR, Singularity</td>
</tr>
<tr>
<td>I/O Services</td>
<td>Lustre client (community version), BeeGFS client</td>
</tr>
<tr>
<td>Numerical/Scientific</td>
<td>Boost, GSL, FFTW, Metis, PETSc, Trilinos, Hypre, SuperLU, SuperLU_Dist, Mumps, OpenBLAS, Scalapack</td>
</tr>
<tr>
<td>Libraries</td>
<td>I/O Libraries</td>
</tr>
<tr>
<td></td>
<td>HDF5 (pHDF5), NetCDF (including C++ and Fortran interfaces), Adios</td>
</tr>
<tr>
<td>Compiler Families</td>
<td>GNU (gcc, g++, gfortran)</td>
</tr>
<tr>
<td>MPI Families</td>
<td>MVAPICH2, OpenMPI, MPICH</td>
</tr>
<tr>
<td>Development Tools</td>
<td>Autotools (autoconf, automake, libtool), Valgrind,R, SciPy/NumPy, hwloc</td>
</tr>
<tr>
<td>Performance Tools</td>
<td>PAPI, IMB, mpiP, pdtoolkit TAU, Scalasca, ScoreP, SIONLib</td>
</tr>
</tbody>
</table>

Notes:
- Additional dependencies that are not provided by the BaseOS or community repos (e.g. EPEL) are also included
- 3rd Party libraries are built for each compiler/MPI family
- Resulting repositories currently comprised of ~450 RPMs

Future additions approved for inclusion in v1.3.2 release:
- PLASMA
- SLEPc
-pNetCDF
- Scotch
- Clang/LLVM

New with v1.3.1

OpenHPC Component Updates

- Part of motivation for community effort like OpenHPC is the rapidity of S/W updates in our space
- Rolling history of updates/additions:
Other new items for v1.3.1 Release

Meta RPM packages introduced and adopted in recipes:

- these replace previous use of groups/patterns
- general convention remains
  - names that begin with "ohpc-*" are typically metapackages
  - intended to group related collections of RPMs by functionality
- some names have been updated for consistency during the switch over
- updated list available in Appendix E

<table>
<thead>
<tr>
<th>Group Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ohpc-autotools</td>
<td>Collection of GNU autotools packages.</td>
</tr>
<tr>
<td>ohpc-base</td>
<td>Collection of base packages.</td>
</tr>
<tr>
<td>ohpc-base-compute</td>
<td>Collection of compute node base packages.</td>
</tr>
<tr>
<td>ohpc-ganglia</td>
<td>Collection of Ganglia monitoring and metrics packages.</td>
</tr>
<tr>
<td>ohpc-gnu7-iodibs</td>
<td>Collection of IO library builds for use with GNU compiler toolchain.</td>
</tr>
<tr>
<td>ohpc-gnu7-mpich-parallel-libs</td>
<td>Collection of parallel library builds for use with GNU compiler toolchain and the MPICH runtime.</td>
</tr>
<tr>
<td>ohpc-gnu7-openmpi-parallel-libs</td>
<td>Collection of parallel library builds for use with GNU compiler toolchain and the OpenMPI runtime.</td>
</tr>
<tr>
<td>ohpc-gnu7-perf-tools</td>
<td>Collection of performance tool builds for use with GNU compiler toolchain.</td>
</tr>
<tr>
<td>ohpc-gnu7-python-libs</td>
<td>Collection of python related library builds for use with GNU compiler toolchain.</td>
</tr>
<tr>
<td>ohpc-gnu7-runtimes</td>
<td>Collection of runtimes for use with GNU compiler toolchain.</td>
</tr>
<tr>
<td>ohpc-gnu7-serial-libs</td>
<td>Collection of serial library builds for use with GNU compiler toolchain.</td>
</tr>
<tr>
<td>ohpc-intel-impiparallel-libs</td>
<td>Collection of parallel library builds for use with Intel(R) Parallel Studio XE toolchain and the Intel(R) MPI Library.</td>
</tr>
<tr>
<td>ohpc-intel-iodibs</td>
<td>Collection of IO library builds for use with Intel(R) Parallel Studio XE software suite.</td>
</tr>
<tr>
<td>ohpc-intel-mpich-parallel-libs</td>
<td>Collection of parallel library builds for use with Intel(R) Parallel Studio XE toolchain and the MPICH runtime.</td>
</tr>
<tr>
<td>ohpc-intel-openmpi-parallel-libs</td>
<td>Collection of parallel library builds for use with Intel(R) Parallel Studio XE toolchain and the OpenMPI runtime.</td>
</tr>
<tr>
<td>ohpc-intel-python-libs</td>
<td>Collection of python related library builds for use with Intel(R) Parallel Studio XE toolchain.</td>
</tr>
<tr>
<td>ohpc-intel-runtimes</td>
<td>Collection of runtimes for use with Intel(R) Parallel Studio XE toolchain.</td>
</tr>
<tr>
<td>ohpc-intel-serial-libs</td>
<td>Collection of serial library builds for use with Intel(R) Parallel Studio XE toolchain.</td>
</tr>
<tr>
<td>ohpc-nagios</td>
<td>Collection of Nagios monitoring and metrics packages.</td>
</tr>
<tr>
<td>ohpc-slurm-client</td>
<td>Collection of client packages for SLURM.</td>
</tr>
<tr>
<td>ohpc-slurm-server</td>
<td>Collection of server packages for SLURM.</td>
</tr>
<tr>
<td>ohpc-warewulf</td>
<td>Collection of base packages for Warewulf provisioning.</td>
</tr>
</tbody>
</table>
Other new items for v1.3.1 Release

• A new compiler variant (gnu7) was introduced
  – in the case of a fresh install, recipes default to installing the new variant along with matching runtimes and libraries
  – if upgrading a previously installed system, administrators can opt-in to enable the gnu7 variant

• The meta-packages for “gnu7” provide a convenient mechanism to add on:
  – upgrade discussion in recipes (Appendix B) amended to highlight this workflow

# Update default environment
[sms]# yum -y remove lmod-defaults-gnu-mvapich2-ohpc
[sms]# yum -y install lmod-defaults-gnu7-mvapich2-ohpc

# Install GCC 7.x-compiled meta-packages with dependencies
[sms]# yum -y install ohpc-gnu7-perf-tools \  
  ohpc-gnu7-serial-front-end \  
  ohpc-gnu7-serial-libs \  
  ohpc-gnu7-io-front-end \  
  ohpc-gnu7-io-libs \  
  ohpc-gnu7-python-front-end \  
  ohpc-gnu7-python-libs \  
  ohpc-gnu7-runtimes \  
  ohpc-gnu7-mpich-parallel-libs \  
  ohpc-gnu7-openmpi-parallel-libs \  
  ohpc-gnu7-mvapich2-parallel-libs

note: could skip this to leave previous gnu toolchain as default
parallel libs for gnu7/mpich
Coexistence of multiple variants

Consider an example of system originally installed from 1.3 base release and then added gnu7 variant using commands from last slide

$ module list

Currently Loaded Modules:
  1) autotools  2) prun/1.1  3) gnu7/7.1.0  4) mvapich2/2.2  5) ohpc

$ module avail

----------------------------------- /opt/ohpc/pub/moduledeps/gnu7-mvapich2 -----------------------------------
   adios/1.11.0  imb/4.1    netcdf-cxx/4.3.0    scalapack/2.0.2    sionlib/1.7.1
   boost/1.63.0  mpiP/3.4.1   netcdf-fortran/4.4.4    scalasca/2.3.1    superlu_dist/4.2
   fftw/3.3.6  mumps/5.1.1    petsc/3.7.6      scipy/0.19.0      tau/2.26.1
   hypre/2.11.1  netcdf/4.4.1.1 phdf5/1.10.0    scorep/3.0      trilinos/12.10.1

----------------------------------- /opt/ohpc/pub/moduledeps/gnu7 -----------------------------------
  R_base/3.3.3 metis/5.1.0    numpy/1.12.1    openmpi/1.10.7
   gsl/2.3         mpiich/3.2   ocr/1.0.1      pdtoolkit/3.23
   hdf5/1.10.0               mvapich2/2.2 (L)  openblas/0.2.19 superlu/5.2.1

----------------------------------- /opt/ohpc/pub/modulefiles -----------------------------------
  EasyBuild/3.2.1                gnu/5.4.0       ohpc (L)    singularity/2.3
     autotools (L)               gnu7/7.1.0  (L)    papi/5.5.1    valgrind/3.12.0
  clustershell/1.7.3             hwloc/1.11.6       prun/1.1 (L)

previous installed from 1.3 release

everything else from 1.3.1 updates (add-on)
Summary

• Technical Steering Committee just completed it’s first year of operation; new membership selection for 2017-2018 in place

• Provided a highlight of changes/evolutions that have occurred since MUG’16
  - 4 releases since last MUG
  - architecture addition (ARM)
  - multiple provisioner/resource manager recipes

• “Getting Started” Tutorial held at PEARC’17 last month
  - more in-depth overview: https://goo.gl/NyiDmr

Community Resources

http://openhpc.community (general info)
https://github.com/openhpc/ohpc (main GitHub site)
https://github.com/openhpc/submissions (new submissions)
https://build.openhpc.community (build system/repos)
http://www.openhpc.community/support/mail-lists/ (mailing lists)