National Aeronautics and Space Administration



Modules Under SLES12
&
Using Software in pkgsrc

NAS Webinar

Apr 12, 2017

NASA Advanced Supercomputing Division

www.nasa.gov

Why Migrating to SLES12 Now?

- SLES11 used on NAS systems is no longer supported by vendors and the Linux community.
- There are security concerns for continuing to use unsupported SLES11.
- Upgrade to SLES12 should have happened many months ago. We can no longer postpone it.

Question? Use the Webex chat facility to ask the Host

SLES12 Testing by Users

- Systems with SLES12: **Pleiades/Electra**; not yet ready for Merope, Endeavour, Lou.
- **pfe27**% qsub **–q alphatst** –lselect=..:model={san,ivy,has,bro,bro_ele} (PBS Provisioning)
  - 2 hours limit
  - Each user limited to 2 running jobs at a time (may change depending on demand)
  - No charge against your SBU allocation
- Test your existing executables
  - Check if your code runs and if the results are correct
- Recompile under SLES12 if necessary
  - Check if any software package is missing for compilation
  - SLES12 executable won't run under SLES11
- Test other software (Matlab, IDL, Tecplot, Totalview, …)
- Report problems or missing software that you need to support@nas.nasa.gov
- Length of testing period currently is 1 month.

Question? Use the Webex chat facility to ask the Host

Software Modules under SLES12

- Goal: better management of public modules
  - /nasa (where public modules reside) is very full
  - Many modules built under SLES 11 are very old or broken
  - Many software packages have multiple versions under /nasa

    12 versions of Tecplot 360; 17 hdf5; 19 pgi compiler

- Plan to provide at most 2-3 versions of each under SLES12

```
-------------------- /nasa/modulefiles/sles12 --------------------
comp-intel/2015.3.187    idl/8.1                  netcdf/4.4.1.1_mpt
comp-intel/2016.2.181    mathematica/11.0.0       netcdf/4.4.1.1_serial
comp-pgi/16.10           matlab/2016b             pagecache-management/0.5
comp-pgi/17.1            mpi-sgi/mpt              szip/2.1.1
hdf4/4.2.12              mpi-sgi/mpt.2.12r26      tecplot/2017r1
hdf5/1.8.18_mpt          mpi-sgi/mpt.2.14r19      totalview/2017.0.12
hdf5/1.8.18_serial       mpi-sgi/mpt.2.15r20      vtune/2016_u3
------------------- /nasa/modulefiles/pkgsrc/sles12 -------------
boost/1.62     pkgsrc/2016Q4 python/2.7.12 python3/3.5.2
```

both 360 and chorus

has memoryscape

Question? Use the Webex chat facility to ask the Host

Software Modules under SLES12

- Rely heavily on **pkgsrc** for open source software
- NAS staff will add more public modules depending on demand
  - If the need cannot be satisfied by using pkgsrc/2016Q4
  - Multiple groups need it built the same way. If just 1 user or 1 group only, build it under 1 user of the group and allow other group members to access it
- If you load modules in your .cshrc, .bashrc, .profile, etc. or PBS scripts, they should be updated with new versions built under SLES12.

Question? Use the Webex chat facility to ask the Host

SLES11 Software Modules

- Old SLES11 modules can still be accessed.

```
% module use –a /nasa/modulefiles-sles11
% module load name_of_old_module
```

- KB 541 contains a partial list of SLES11 modules that won't work under SLES12.
- **When the full migration to SLES 12 takes place:**
  - **These old modules built under SLES 11 are no longer supported. If you have issues with them, you have to use modules built under SLES 12.**
  - **Old modules that are broken or pose security concerns will be removed without notice.**
  - **Old modules that are not used (according to our logs) will be removed without notice.**

Question? Use the Webex chat facility to ask the Host

What is pkgsrc?

https://www.pkgsrc.org/

"pkgsrc is a framework for building third-party software on NetBSD and other UNIX-like systems, currently containing over 17,000 packages. It is used to **enable freely available software to be configured and built easily on supported platforms**."

At NAS, we built ~100 packages from pkgsrc.

What packages are in pkgsrc?

http://pkgsrc.se

"pkgsrc by the web, pkgsrc.se, is intended to be **an easy navigated frontend to the NetBSD packages collection**. …. Only CURRENT and the latest stable branch is updated."

Why go with pkgsrc?

- Building individual software packages is tedious and time consuming
- Using pkgsrc provides a systematic and automatic build process
- pkgsrc provides updates to software on a regular basis
- If the same software is also available under system directories (such as /usr/bin, etc), pkgsrc's version is often more up to date

Question? Use the Webex chat facility to ask the Host

NAS pkgsrc

- Starting with 2014Q1, a SLES11 build was done ~ every quarter
- Selected (not all) software packages from pkgsrc are built
  - selection criteria:
    - ✓ replacement for old existing modules; for example, boost/1.48.0
    - ✓ requests from users
  - current state (2016Q4 built under SLES12):
    - ✓ Total of ~100 selected in the NAS list
    - ✓ Actual packages built ~850 due to software dependency on others
      For example, bioperl depends on > 100 packages
    - ✓ Some (5) software failed to build, thus they are not in the NAS pkgsrc:
      devel/doxygen, editors/emacs, geography/qgis, math/octave, net/tigervnc

Question? Use the Webex chat facility to ask the Host

What are available in NAS pkgsrc?



% ***module help pkgsrc/2016Q4*** (show some packages included)

```
----------- Module Specific Help for 'pkgsrc/2016Q4' --------------
This module is a collection of software built from NetBSD's pkgsrc
system. Some of the key packages included are:

        armadillo-4.45.2        bioperl-1.007001        bison-3.0.4nb3
        bmake-20150505          boost-1.62.0            bzr-2.6.0
        clang-3.9.0             cscope-15.8anb2         cwrappers-20161125
        elinks-0.12rc6nb11      fetch-1.8nb3            gdal-lib-1.11.3nb7
        git-2.11.0              GMT-4.1.4nb8            gnuplot-5.0.4nb1
        GraphicsMagick-1.3.25   h5utils-1.12.1nb8       hdf5-c++-1.8.18
        htop-2.0.2              ImageMagick-7.0.3.10    lrzip-0.621
        lz4-1.7.4.2             lzop-1.03               mercurial-4.0.1
        metis-4.0.3             mg-20110905nb2          mono-4.0.4.1nb5
        nano-2.7.2              ncview-1.92dnb8         nedit-5.5nb9
        netcdf-fortran-4.2nb4   nvi-1.81.6nb10          pbzip2-1.1.13
        pkgfind-20111022        qgis-2.14.3nb3          R-3.3.2
        R-ncdf-1.6.6nb4         rhash-1.3.4             ruby-2.2.6
        screen-4.4.0            tmux-2.3nb2             valgrind-3.12.0
        vtk-4.2.6nb6            wget-1.18nb3

For a complete list of the packages included with this module, please
run "/nasa/pkgsrc/sles12/2016Q4/sbin/pkg_info."
```


Question? Use the Webex chat facility to ask the Host

What are available in NAS pkgsrc?

- To show a complete list

```
% module load pkgsrc/2016Q4
% pkg_info
```

- To check if a package (for example, cairo) is in pkgsrc/2016Q4

```
% pkg_info | grep -i cairo
cairo-1.14.8            Vector graphics library with cross-device output support
cairo-gobject-1.14.8 Vector graphics library with cross-device output support
py27-cairo-1.10.0nb4 Python bindings for cairo (python-2.x version)
```

or use –e option which tests for existence of a given package

```
% pkg_info –e cairo
cairo-1.14.8

% pkg_info –e cairo-gobject
cairo-gobject-1.14.8
```


Question? Use the Webex chat facility to ask the Host

What a package requires/was built with?



Built with

%*pkg_info –N cairo*

Requires

%*pkg_info –n cairo*

```
Information for cairo-1.14.8:

Requires:
MesaLib>=10.5.3
libX11>=1.1
libxcb>=1.9.3
libXext>=0.99.0
libXrender>=0.9.2
lzo>=2.01
zlib>=1.2.3
fontconfig>=2.10.93nb2
freetype2>=2.4.11
png>=1.6.0nb1
pixman>=0.30.0
```

```
MesaLib-11.2.2
libX11-1.6.4
libxcb-1.12
libXext-1.3.3
libXrender-0.9.10
lzo-2.09
zlib-1.2.8nb3
fontconfig-2.12.1
freetype2-2.7nb1
png-1.6.27
pixman-0.34.0nb1
digest-20160304
gcc49-4.9.4
dri2proto-2.8
dri3proto-1.0
xproto-7.0.31
presentproto-1.0
glproto-1.4.17
damageproto-1.2.1
fixesproto-5.0
xextproto-7.3.0
inputproto-2.3.2
kbproto-1.0.7
xcb-proto-1.12
xf86vidmodeproto-2.3.1
xf86driproto-2.1.1nb1
renderproto-0.11.1
libXt-1.1.5
cwrappers-20161125
```


Question? Use the Webex chat facility to ask the Host

How was it built?



%*pkg_info –B cairo* <- Show some of the important definitions used when building the binary package

```
Information for cairo-1.14.8:
Build information:
ABI=64
BUILD_DATE=2017-03-22 12:27:31 -0700
BUILD_HOST=Linux pfe27 4.4.38-93.1.20170217-nasa #1 SMP Wed Dec 14 12:59:43 UTC 2016 (2d3e9d4)
x86_64 x86_64 x86_64 GNU/Linux
CATEGORIES=graphics
CC_VERSION=gcc-4.9.4
CFLAGS=-O2 -pipe  -I/nasa/pkgsrc/sles12/2016Q4/include -I/usr/include -I/nasa/pkgsrc/sles12/2016Q4/
include/libdrm -I/nasa/pkgsrc/sles12/2016Q4/include/freetype2
CMAKE_ARGS=-DCMAKE_COLOR_MAKEFILE=OFF
CONFIGURE_ARGS=--disable-gl --enable-fc --enable-pdf --enable-ps --enable-tee --enable-xlib --
enable-xlib-xrender --enable-xcb --disable-quartz --disable-quartz-font --disable-quartz-image --
prefix=/nasa/pkgsrc/sles12/2016Q4 --build=x86_64-unknown-linux --host=x86_64-unknown-linux --
mandir=/nasa/pkgsrc/sles12/2016Q4/man
CONFIGURE_ENV=DL_CFLAGS= DL_LDFLAGS= DL_LIBS= PTHREAD_CFLAGS=\ -pthread\  PTHREAD_LDFLAGS=\ -pthrea
PTHREAD_LIBS=-lpthread\ -lrt PTHREADBASE=/usr FREETYPE_CONFIG=/nasa/pkgsrc/sles12/2016Q4/bin/
freetype-config LIBTOOL="/tmp/hsp/pkgsrc/2016Q4/graphics/cairo/work/.cwrapper/bin/libtool "
…
CPPFLAGS=   -I/nasa/pkgsrc/sles12/2016Q4/include -I/usr/include -I/nasa/pkgsrc/sles12/2016Q4/
include/libdrm -I/nasa/pkgsrc/sles12/2016Q4/include/freetype2
FFLAGS=-O
GLIBC_VERSION=2.22 (git bbab82c25da9)
HOMEPAGE=http://cairographics.org/
LDFLAGS=  -L/nasa/pkgsrc/sles12/2016Q4/gcc49/lib/gcc/x86_64-unknown-linux/4.9.4 -Wl,-R/nasa/pkgsrc/
sles12/2016Q4/gcc49/lib/gcc/x86_64-unknown-linux/4.9.4 -L/nasa/pkgsrc/sles12/2016Q4/gcc49/lib64 -
Wl,-R/nasa/pkgsrc/sles12/2016Q4/gcc49/lib64  -L/nasa/pkgsrc/sles12/2016Q4/lib -L/usr/lib64 -Wl,-R/
usr/lib64 -Wl,-R/nasa/pkgsrc/sles12/2016Q4/lib
```


Question? Use the Webex chat facility to ask the Host

How was it built (cont'd)?




```
PROVIDES=/nasa/pkgsrc/sles12/2016Q4/lib/cairo/libcairo-trace.so
PROVIDES=/nasa/pkgsrc/sles12/2016Q4/lib/cairo/libcairo-trace.so.0
PROVIDES=/nasa/pkgsrc/sles12/2016Q4/lib/cairo/libcairo-trace.so.0.0.0
PROVIDES=/nasa/pkgsrc/sles12/2016Q4/lib/libcairo-script-interpreter.so
PROVIDES=/nasa/pkgsrc/sles12/2016Q4/lib/libcairo-script-interpreter.so.2
PROVIDES=/nasa/pkgsrc/sles12/2016Q4/lib/libcairo-script-interpreter.so.2.11400.8
PROVIDES=/nasa/pkgsrc/sles12/2016Q4/lib/libcairo.so
PROVIDES=/nasa/pkgsrc/sles12/2016Q4/lib/libcairo.so.2
PROVIDES=/nasa/pkgsrc/sles12/2016Q4/lib/libcairo.so.2.11400.8
REQUIRES=/lib64/libc.so.6
REQUIRES=/lib64/libdl.so.2
REQUIRES=/lib64/libm.so.6
REQUIRES=/lib64/libpthread.so.0
REQUIRES=/lib64/librt.so.1
REQUIRES=/nasa/pkgsrc/sles12/2016Q4/lib/libX11.so.6
REQUIRES=/nasa/pkgsrc/sles12/2016Q4/lib/libXau.so.6
REQUIRES=/nasa/pkgsrc/sles12/2016Q4/lib/libXdmcp.so.6
REQUIRES=/nasa/pkgsrc/sles12/2016Q4/lib/libXext.so.6
REQUIRES=/nasa/pkgsrc/sles12/2016Q4/lib/libXrender.so.1
REQUIRES=/nasa/pkgsrc/sles12/2016Q4/lib/libbz2.so.0
REQUIRES=/nasa/pkgsrc/sles12/2016Q4/lib/libexpat.so.1
…
REQUIRES=/usr/lib64/libEGL.so.1
REQUIRES=/usr/lib64/libGL.so.1
REQUIRES=/usr/lib64/libX11-xcb.so.1
REQUIRES=/usr/lib64/libXdamage.so.1
REQUIRES=/usr/lib64/libXfixes.so.3
REQUIRES=/usr/lib64/libXxf86vm.so.1
REQUIRES=/usr/lib64/libcairo.so.2
REQUIRES=/usr/lib64/libdrm.so.2
…
REQUIRES=/usr/lib64/libxcb-present.so.0
REQUIRES=/usr/lib64/libxcb-sync.so.1
REQUIRES=/usr/lib64/libxcb-xfixes.so.0
REQUIRES=/usr/lib64/libxshmfence.so.1
```


Question? Use the Webex chat facility to ask the Host

Limitations of NAS pkgsrc

- It may not include a package or version you want
- A package may not have been built with certain things enabled/disabled the way you want
- Its automatic build process does not build packages that rely on vendor packages (such as mpi4py with mpi-sgi/mpt)

Question? Use the Webex chat facility to ask the Host

How to use NAS pkgsrc?

- Use executables (and libraries) self-contained in pkgsrc
  - Be aware of the order in your search path $PATH

    The first instance found under $PATH will be used
- Build additional Python packages/versions for your own use
  - Use virtualenv + pip (KB 540)
- Build software on top of libraries under pkgsrc
  - Be aware of the order of shared library search path

    The first instance found under $LD_LIBRARY_PATH will be used

Question? Use the Webex chat facility to ask the Host



Use executables (and libraries) that are self-contained in pkgsrc

Question? Use the Webex chat facility to ask the Host

Order of your PATH (2016Q2 and older)



The first instance found under $PATH is used.

```
% module purge
% echo $PATH   (this will show the list of default system paths)
/usr/local/bin:/usr/bin:/bin:/usr/X11R6/bin:/PBS/bin:/usr/sbin:/sbin:/opt/c3/bin:
/opt/sgi/sbin:/opt/sgi/bin

% module show pkgsrc/2016Q2
-------------------------------------------------------------------
/nasa/modulefiles-sles11/pkgsrc/2016Q2:
system          /bin/logger -p local2.info -t envmodule ychang1 display pkgsrc/2016Q2
module-whatis   A collection of software built via NetBSD's pkgsrc
append-path     INFOPATH /nasa/pkgsrc/2016Q2/info
append-path     MANPATH /nasa/pkgsrc/2016Q2/man
append-path     PATH /nasa/pkgsrc/2016Q2/bin:/nasa/pkgsrc/2016Q2/sbin

-------------------------------------------------------------------

% module load pkgsrc/2016Q2
% echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/X11R6/bin:/PBS/bin:/usr/sbin:/sbin:/opt/c3/
bin:/opt/sgi/sbin:/opt/sgi/bin:/nasa/pkgsrc/2016Q2/bin:/nasa/pkgsrc/2016Q2/sbin
```

Set this way to avoid unexpected behavior

Question? Use the Webex chat facility to ask the Host

Examples (2016Q2 and older under SLES11)

- Executable (such as magick) without a version under system paths

```
% module load pkgsrc/2016Q2
% magick
```

- Executable (such as cmake, makeinfo, python, perl, ruby, tclsh, wget, etc) with a version under system paths

  Even after loading the pkgsrc module, the system copy (/usr/bin/cmake, version 2.8.12) is found first. Workaround is:

```
% /nasa/pkgsrc/2016Q2/bin/cmake     (version 3.5.2)
```

- For commonly used software, like gcc, python, separate modules were created **with its path prepended**.

```
% module load python/2.7.11
% python
```

The pkgsrc executables have RPATH encoded (via `-Wl,-R/nasa/pkgsrc/2016Q2/lib`) so the libraries they depend on can be found without having to set LD_LIBRARY_PATH

Question? Use the Webex chat facility to ask the Host

Order of your PATH (2016Q4/SLES12)




```
%  module show pkgsrc/2016Q4
-------------------------------------------------------------------
/nasa/modulefiles/pkgsrc/sles12/pkgsrc/2016Q4:
system           /bin/logger -p local2.info -t envmodule ychang1 display pkgsrc/2016Q4
module-whatis    A collection of software built via NetBSD's pkgsrc
prepend-path     INFOPATH /nasa/pkgsrc/sles12/2016Q4/info
prepend-path     MANPATH /nasa/pkgsrc/sles12/2016Q4/man
prepend-path     PATH /nasa/pkgsrc/sles12/2016Q4/bin:/nasa/pkgsrc/sles12/2016Q4/sbin
-------------------------------------------------------------------
```

```
% module load pkgsrc/2016Q4
% echo $PATH
/nasa/pkgsrc/sles12/2016Q4/bin:/nasa/pkgsrc/sles12/2016Q4/sbin:/usr/local/bin:/usr/
bin:/bin:/usr/X11R6/bin:/PBS/bin:/usr/sbin:/sbin:/opt/c3/bin:/opt/sgi/sbin:/opt/
sgi/bin
```

Most users expect using the copy under pkgsrc when they load the pkgsrc module. Prepending will be used for future releases if no major issues with pkgsrc/2016Q4.

Examples
(2016Q4 under sles12)

- For all executables with or without a version under system paths, loading the 2016Q4 module will cause the one under 2016Q4 to be found first

```
% module load pkgsrc/2016Q4
% magick
% cmake                    (from pkgsrc, not /usr/bin/cmake)
```


Question? Use the Webex chat facility to ask the Host

Build additional Python packages or different versions for your own use

Question? Use the Webex chat facility to ask the Host

The pip and virtualenv tools

- pip (python2) or pip3 (python3) tools to install packages

```
% module load python/2.7.12    (from pkgsrc/2016Q4)
% pip install scipy==0.18.1
```

-> won't install because you are not allowed to install under /nasa/pkgsrc/sles12/2016Q4/lib/python2.7/site-packages/scipy/

```
% pip install --user scipy==0.18.1
```
(use this if you want a specific version)

-> install under your **~/.local/lib/python2.7/site-packages**

- virtualenv tool to manage your isolated python environment

```
% module load python/2.7.12
% virtualenv --system-site-packages ~/myenv
```
(create new or overwrite)

```
% source ~/myenv/bin/activate
```
(or activate.csh for csh)

```
[myenv]% pip install --upgrade scipy
```
(if a package was already installed)

-> install latest version under ~/myenv/lib/python2.7/site-packages

Question? Use the Webex chat facility to ask the Host

Example
Build mpi4py with sgi-mpt for your own use

- Install mpi4py in your private python environment

```
pfe27% module load mpi-sgi/mpt
pfe27% module load python/2.7.12
pfe27% virtualenv --system-site-packages ~/myenv
pfe27% source ~/myenv/bin/activate.csh
[myenv] pfe27% pip install mpi4py (installed under ~/myenv/lib/python2.7/site-packages)
[myenv] pfe27% deactivate
pfe27%
```

- Run python code with mpi4py

```
pfe27% qsub -I -lselect=2:ncpus=4:model=bro -q alphatst
PBS> module load mpi-sgi/mpt
PBS> module load python/2.7.12   (no need for this)
PBS> source ~/myenv/bin/activate.csh
[myenv] PBS> mpiexec –np 8 python hello.py
Hello, world! This is rank 0 of 8 running on r607i5n16
Hello, world! This is rank 2 of 8 running on r607i5n16
Hello, world! This is rank 3 of 8 running on r607i5n16
Hello, world! This is rank 1 of 8 running on r607i5n16
Hello, world! This is rank 4 of 8 running on r607i5n17
Hello, world! This is rank 5 of 8 running on r607i5n17
Hello, world! This is rank 7 of 8 running on r607i5n17
Hello, world! This is rank 6 of 8 running on r607i5n17
```


Question? Use the Webex chat facility to ask the Host

Build Your Own Software on Top of pkgsrc Libraries

straight-forward if pkgsrc library needed is in /nasa/pkgsrc/sles12/2016Q4/lib/xxx

more attention required if pkgsrc library needed is in /nasa/pkgsrc/sles12/2016Q4/lib

Question? Use the Webex chat facility to ask the Host

pkgsrc library needed is in a subdirectory of /nasa/pkgsrc/sles12/2016Q4/lib




```
PROVIDES=/nasa/pkgsrc/sles12/2016Q4/lib/cairo/libcairo-trace.so
```

- Nothing other than cairo related libraries are in this directory

```
pfe27% ls /nasa/pkgsrc/sles12/2016Q4/lib/cairo
cairo-fdr.a      cairo-fdr.so.0.0.0  libcairo-trace.so.0
cairo-fdr.la     libcairo-trace.a    libcairo-trace.so.0.0.0
cairo-fdr.so     libcairo-trace.la
cairo-fdr.so.0   libcairo-trace.so
```

- Compile Time

```
% module load comp-intel/2016.2.181
% icpc -o a.out cairo.cpp
  -I/nasa/pkgsrc/sles12/2016Q4/include/cairo
  –L/nasa/pkgsrc/sles12/2016Q4/lib/cairo –lcairo-trace
```

- Run Time

```
% setenv LD_LIBRARY_PATH /nasa/pkgsrc/sles12/2016Q4/lib/cairo
% a.out
```


Question? Use the Webex chat facility to ask the Host

pkgsrc library needed is in /nasa/pkgsrc/sles12/2016Q4/lib

- What libraries are in /nasa/pkgsrc/sles12/2016Q4/lib?

  There are >2000 library files there.

  They belong to various packages.

  For example,

  libblas, libboost, libhdf5, libnetcdf, libz

- To use such libraries, one need to pay attention to the order of shared library search paths

Question? Use the Webex chat facility to ask the Host

Order of shared library search paths for all executables on Pleiades (the first matching library is used)

1) **LD_LIBRARY_PATH** (if set during runtime)

   NAS pkgsrc modules do not set LD_LIBRARY_PATH on purpose so that software that does not use pkgsrc will not be affected.

2) **rpath** (if run-time search path is encoded in the executable)

3) **system default library paths** (entries in /etc/ld.so.conf, /lib, /usr/lib)

On pfe's, /etc/ld.so.conf includes:

```
/usr/x86_64-suse-linux/lib
/lib64
/usr/lib
/opt/kde3/lib64
/opt/gnome/lib
include ld.so.conf.d/*.conf
/usr/local/lib64
/lib
/usr/X11R6/lib64
/opt/kde3/lib
/usr/lib64/gtk-2.0/modules
/usr/local/lib
/usr/lib64
/usr/X11R6/lib
/opt/gnome/lib64
```


Question? Use the Webex chat facility to ask the Host

Build Your Own Software on Top of pkgsrc

- Getting rpath encoded in executable, avoid using LD_LIBRARY_PATH
  - link time (no need to load a pkgsrc module)

```
setenv LD_RUN_PATH /nasa/pkgsrc/sles12/2016Q4/lib
-I/nasa/pkgsrc/sles12/2016Q4/include
-L/nasa/pkgsrc/sles12/2016Q4/lib –lxxx
```

Or

```
-I/nasa/pkgsrc/sles12/2016Q4/include
-L/nasa/pkgsrc/sles12/2016Q4/lib –lxxx
-Wl,-R/nasa/pkgsrc/sles12/2016Q4/lib
```

(overrides LD_RUN_PATH)

Caveat: When 2016Q4 is deleted in the future, your code won't work, unless the library can be found somewhere else.

  - run time (no need to load a pkgsrc module)

    No need to set LD_LIBRARY_PATH or LD_RUN_PATH

- Otherwise
  - link time

```
-I/nasa/pkgsrc/sles12/2016Q4/include
-L/nasa/pkgsrc/sles12/2016Q4/lib –lxxx
```

Caveat: LD_LIBRARY_PATH affects all executables including those you want to use a library NOT from pkgsrc

  - run time (no need to load a pkgsrc module)

```
setenv LD_LIBRARY_PATH "/nasa/pkgsrc/sles12/2016Q4/lib"
```

(if LD_LIBRARY_PATH has not been set)

or `setenv LD_LIBRARY_PATH "/nasa/pkgsrc/sles12/2016Q4/lib:$LD_LIBRARY_PATH"`

Example
h5_cmprss.c with hdf5 in pkgsrc




```
% module load comp-intel/2016.2.181
% setenv LD_RUN_PATH /nasa/pkgsrc/sles12/2016Q4/lib
% icc -o a.out1 h5_cmprss.c –I/nasa/pkgsrc/sles12/2016Q4/include
      -L/nasa/pkgsrc/sles12/2016Q4/lib –lhdf5
% ldd a.out1  (no need to have LD_RUN_PATH set when run)
libhdf5.so.10 => /nasa/pkgsrc/sles12/2016Q4/lib/libhdf5.so.10


% module load comp-intel/2016.2.181
% icc -o a.out2  h5_cmprss.c  –I/nasa/pkgsrc/sles12/2016Q4/include
      -L/nasa/pkgsrc/sles12/2016Q4/lib –lhdf5 -Wl,-R/nasa/pkgsrc/sles12/2016Q4/lib
% ldd a.out2
libhdf5.so.10 => /nasa/pkgsrc/sles12/2016Q4/lib/libhdf5.so.10


% module load comp-intel/2016.2.181
% icc -o a.out3 h5_cmprss.c –I/nasa/pkgsrc/sles12/2016Q4/include
      -L/nasa/pkgsrc/sles12/2016Q4/lib –lhdf5
% ldd a.out3
libhdf5.so.10 => not found
% setenv LD_LIBRARY_PATH "/nasa/pkgsrc/sles12/2016Q4/lib:$LD_LIBRARY_PATH"
% ldd a.out3
libhdf5.so.10 => /nasa/pkgsrc/sles12/2016Q4/lib/libhdf5.so.10
```

Warning:
pkgsrc's hdf5 – NO parallel enabled
hdf5/1.8.18_mpt - parallel enabled

Question? Use the Webex chat facility to ask the Host

Example

```
% module show hdf5/1.8.18_mpt
-------------------------------------------------------------------
…
prepend-path    LD_LIBRARY_PATH /nasa/hdf5/1.8.18_mpt/lib
…
prepend-path    INCLUDE /nasa/hdf5/1.8.18_mpt/include
```

```
% module purge
% echo $LD_LIBRARY_PATH
LD_LIBRARY_PATH: Undefined variable.

% setenv LD_LIBRARY_PATH /nasa/pkgsrc/sles12/2016Q4/lib
% module load hdf5/1.8.18_mpt

% echo $LD_LIBRARY_PATH
/nasa/hdf5/1.8.18_mpt/lib:/nasa/pkgsrc/sles12/2016Q4/lib

% ldd a.out1
libhdf5.so.10 => /nasa/hdf5/1.8.18_mpt/lib/libhdf5.so.10

% ldd a.out2
libhdf5.so.10 => /nasa/hdf5/1.8.18_mpt/lib/libhdf5.so.10

% ldd a.out3
libhdf5.so.10 => /nasa/hdf5/1.8.18_mpt/lib/libhdf5.so.10
```

Additional Info

https://www.nas.nasa.gov/hecc/support/kb/entry/xxx

- xxx= 541 for sles12 migration testing
- xxx= 493 for using pkgsrc (need update about prepend)
- xxx= 540 for using virtualenv to build Python packages

Send questions or issues to support@nas.nasa.gov

Start testing NOW. Postponing the migration is not an option.