



Spack-Based Production Programming Environments on Cray Shasta

Paul Ferrell, Timothy Goetsch, Francine Lapid
HPC-ENV

Scalable Tools Workshop 2024

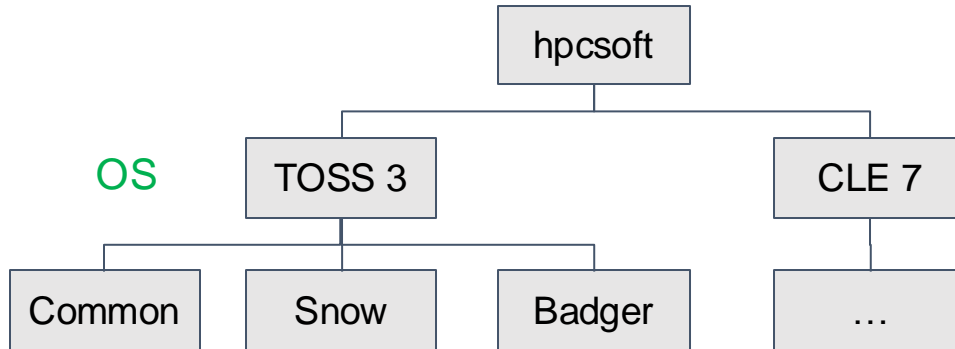
LA-UR-24-28619

The Programming and Runtime Environment Team

“To enable productive science by providing an intuitive, robust, reliable and timely user environment via third party software, tools, utilities, maintenance and support services.”

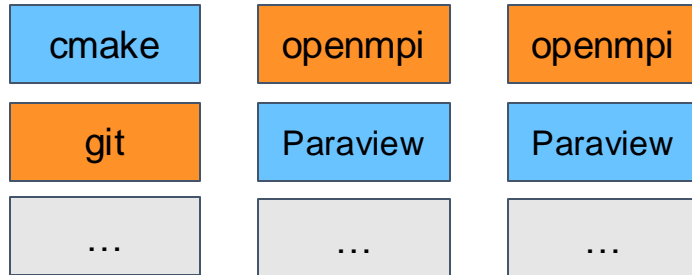
Old Deployment Method

Legacy Software Stack



OS

Machine



Software

Products managed by Paul

Products managed by Shivam

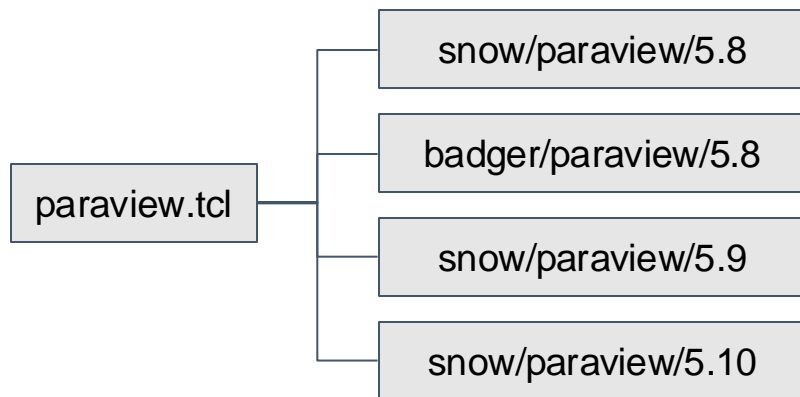
example install prefix: /usr/projects/hpcsoft/toss3/fog/scorep/7.1-gcc-10.3.0-openmpi-4.1.1/

Legacy Software Management

Modulefiles

- one modulefile for each product
 - modulefiles resolve the correct machine, architecture, OS to lead to the correct installation directory

tcl: /install-prefix/\${os}/\${machine}/scorep/\${version}-\${comp_name}-\${comp_ver}-\${mpi_name}-\${mpi_ver}



Necessary improvements

- Build from source against system dependencies
 - One common install ‘language’
 - Long Term Support
 - Shared installs where possible
- Installs from inventory
- Reproducibility
- Combinatorial Installs
- Auto-generate module files

Spack

What is Spack?

- Non-privileged package management tool developed at LLNL
- Large HPC-focused package library
- Uses YAML for configuration files
- Widespread adoption

and more!

TCE2 (Tri-Lab Computing Environment)

```
bin/process.sh
```

```
-f lanl
```

```
-e cos2-x86_64-cc80
```

```
-i /usr/projects/hpcsoft/tce/23-03
```

```
-m
```

TCE2

```
bin/process.sh
```

```
-f lanl
```

```
-e cos2-x86_64-cc80
```

```
-i /usr/projects/hpcsoft/tce/23-03
```

```
-m
```

Facility

Environment

Install
Path

Module
Generation

```
tce-configs/
```

```
  README.md
```

```
  RELEASES.md
```

```
  bin
```

```
  ci-tests
```

```
  environments
```

```
  facilities
```

```
  lanl-build-notes
```

```
  modulefiles
```

```
  package_repos
```

Install Layout

tce2/cray-sles-15-x86_64-slingshot-none

Spack

packages.yaml

spack.yaml

etc.

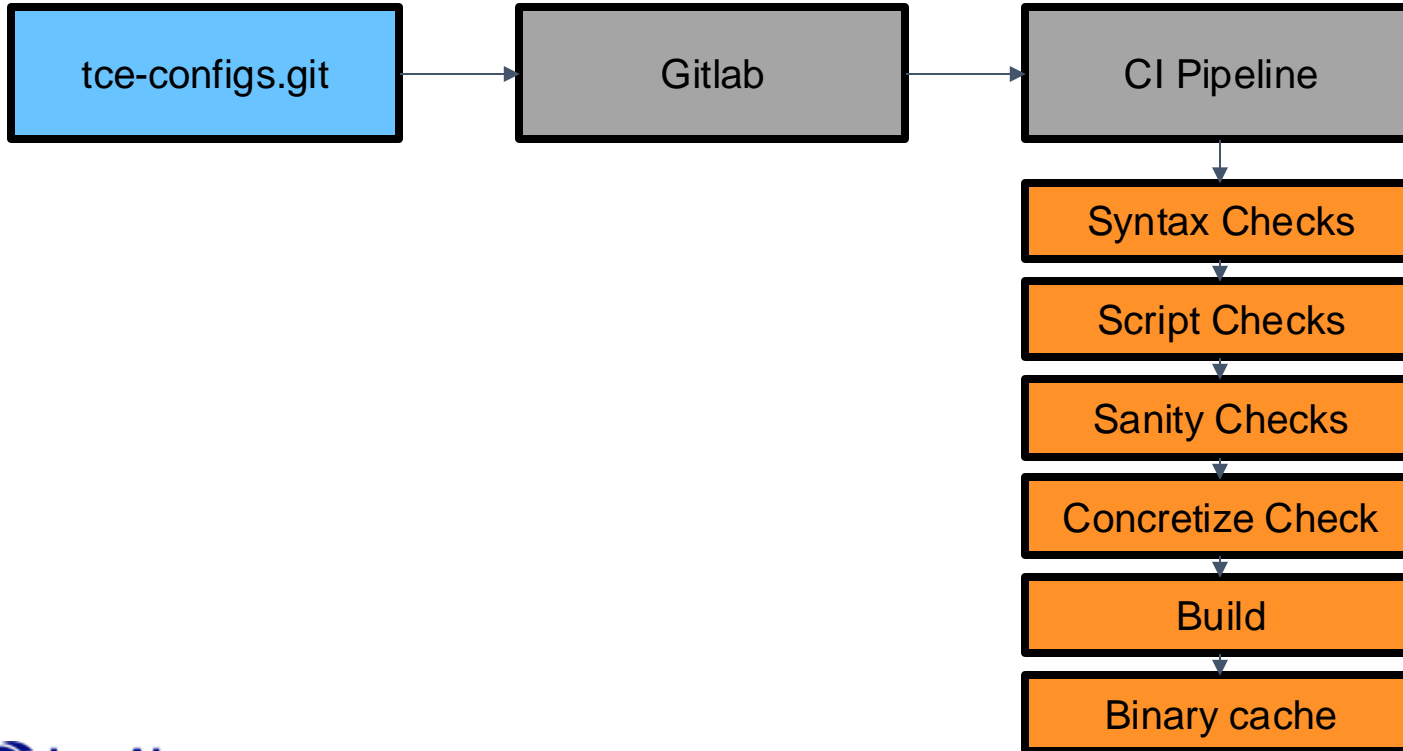
package
repos

logs

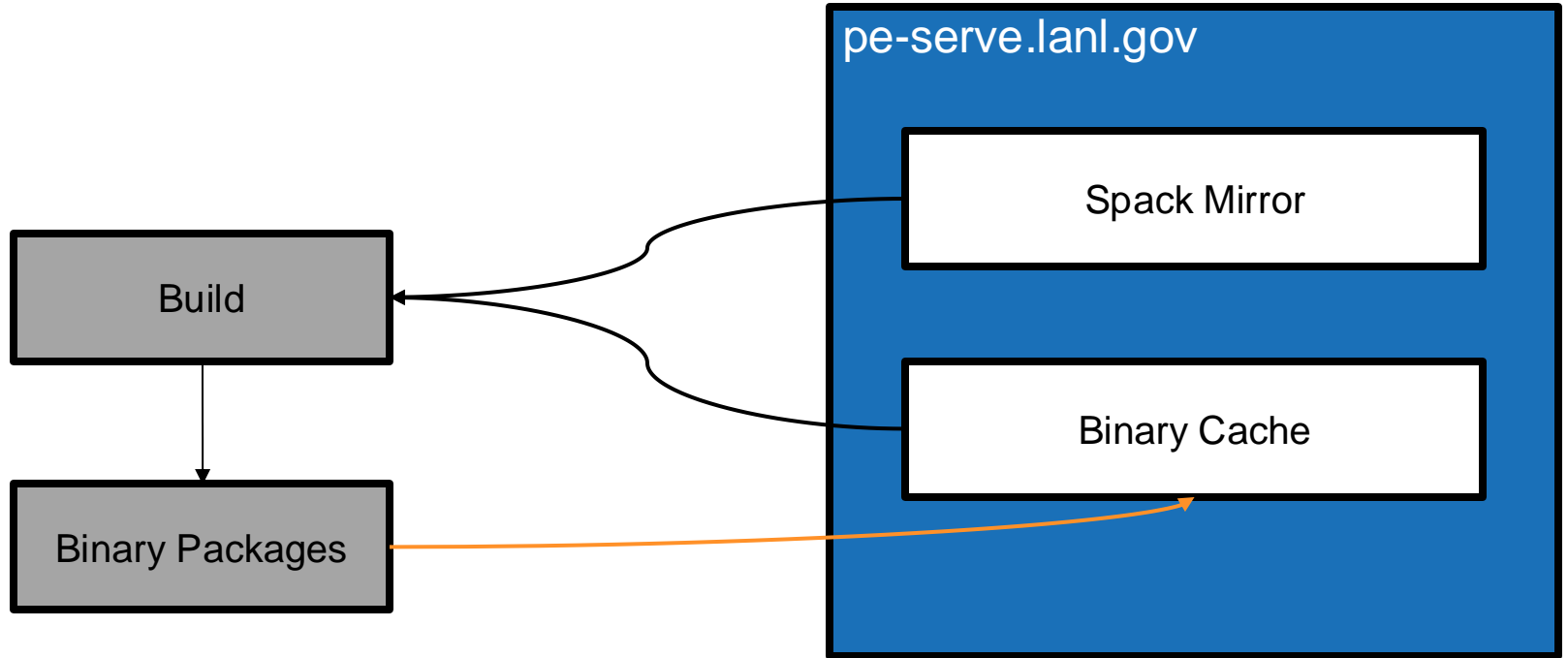
module files

packages

Continuous Integration



Caching



Debugging

The Spack Way

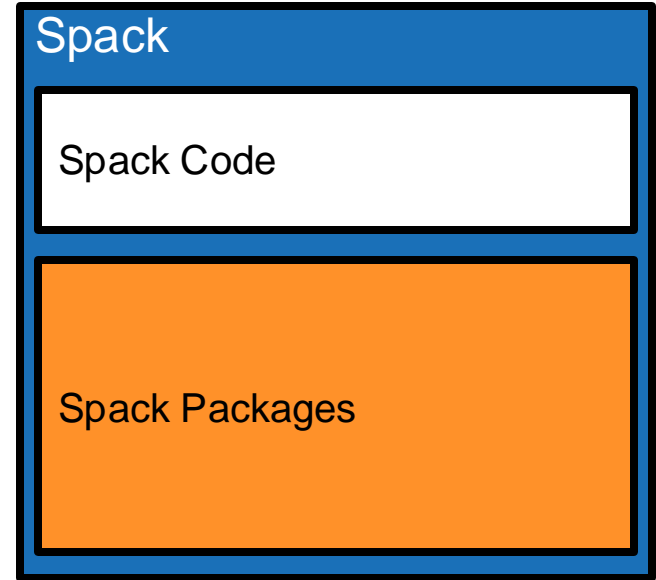
```
spack install mypackage  
  
spack build-env mypackage bash  
# Play around in the build environment
```

Our Way/The TCE Way

```
pushd /tmp/lapid/spack-stage/spack-stage-mypackage-<hash>  
  
source spack-build-env.txt spack-build-env-mods.txt
```

Spack Version Upgrades

- Code upgrades
 - Usually retains backwards compatibility
 - Fixes issues
- Package upgrades
 - No compatibility guarantees
 - Variant changes
 - Paradigm changes
 - (possible) new bugs
 - New dependencies

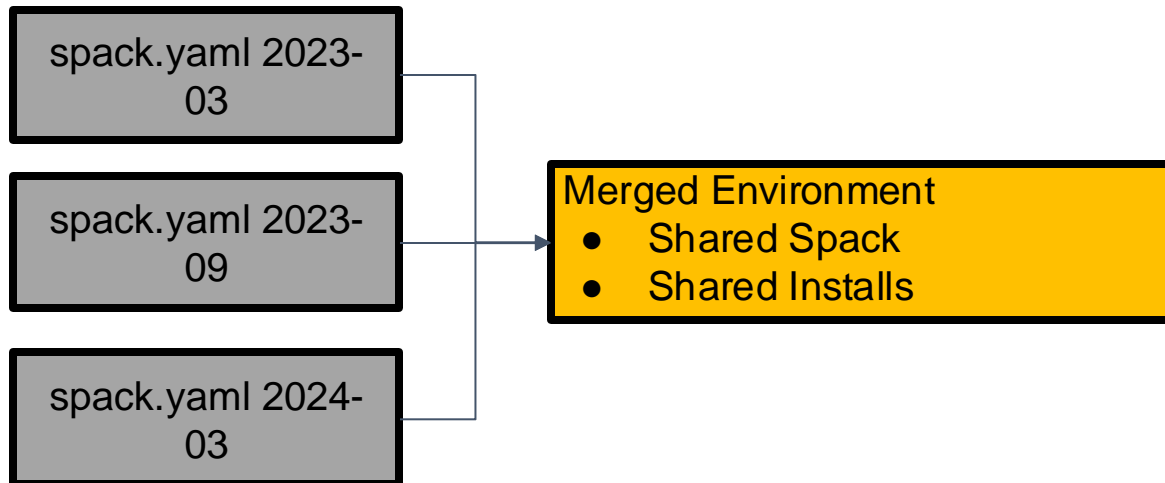


Release Model Options

Singular spack.yaml

- all new software added to one continuous version
- Spack concretization times increase
- issues when switching Spack versions

- Periodic, small releases
- Concretize separately
- Separate environment



The Cray Programming Environment

HPE Cray OS

- HPE Cray Programming Environment (CPE)
 - Communication, scientific, and IO libraries
 - Compiling and programming environments
 - Performance analysis tools
 - Debuggers

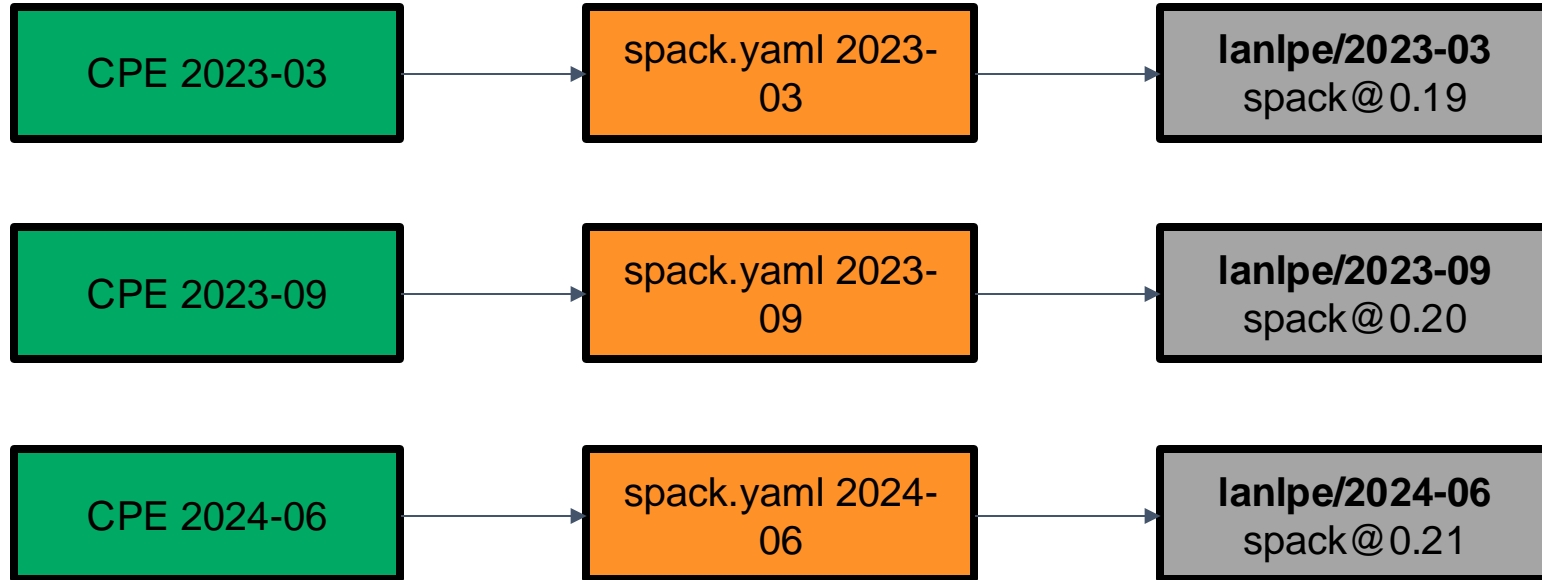
https://www.hpe.com/psnow/doc/a50002722enw?jumpid=in_lit-psnow-red

Limitations

- Spack* with hierarchical Lmod
 - Externals based on modulefiles
- Fixes to Cray Issues may take time
 - Take longer to figure out (without cray-mpich source)
 - Ticket turnaround time
 - Fixes then wait on next CPE or CSM release

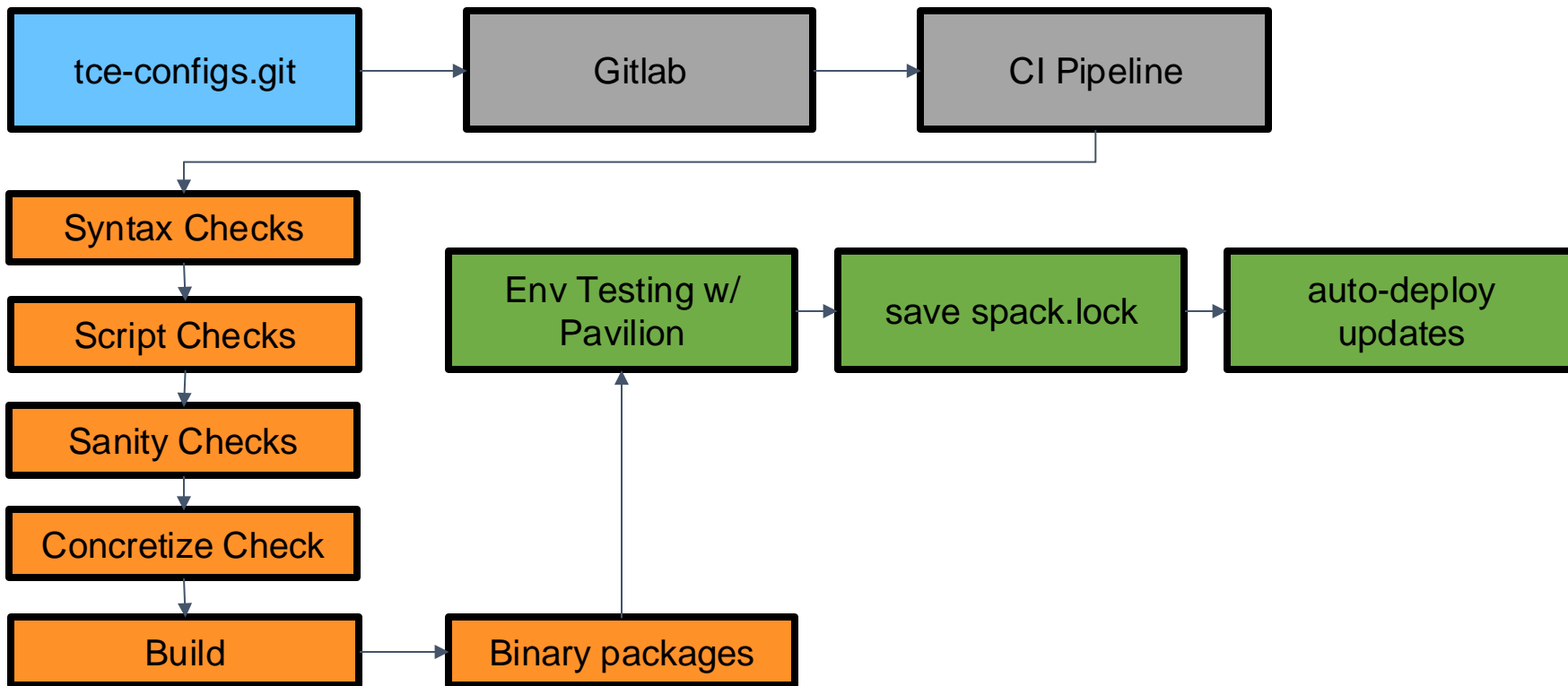
*older versions of Spack. This issue may have been fixed in newer versions that we have not started to use yet

TCE2 with Cray Release Model



Conclusion

Next steps



Conclusion

- Transitioning to using Spack has been costly
- Upgrades are worth it
 - Team structure – anyone can support and add new packages
 - Modernized Infrastructure
 - Can support more for users
 - New Cray clusters require more software support
 - Consistency

Thanks for listening!

My email: lapid@lanl.gov

Team's email: preteam@lanl.gov