

---

---

# Path forward on code repository management and packaging

Jacob King\* (Tech-X)

NIMROD Team Meeting  
May 24th 2023

---

---

Work supported by US DOE

\*Also affiliated with Fiat Lux

# Reminder: Tech-X public facing web infrastructure is being retired

- Impacted: SVN server, nimrodteam.org website, mail lists
- All gitlab infrastructure is independent of Tech-X
- Tech-X is moving infrastructure behind the internal firewall and outsourcing & retiring external SVN and website hosting
- Mail lists are not impacted (already outsourced as Google groups) – but Google group access is restricted to Tech-X employees

Code repository is focus of discussion today

# What is hosted where?

- Tech-X svn server
  - Code versions: nimdevel, nimuw
  - Package repository: nimall
  - Package management system: bilder
  - Externals: fciowrappers, scimake, nimtests
  - Document/input storage: nimruns, nimpapers
- Gitlab.com
  - Code versions: nimrod-aa1, nimrod-multispecies
  - Package repository: nimpack

# Moving from nimall (Tech-X svn) to nimpack (gitlab.com)

## nimall (Tech-X svn)

- Contains svn externals for bilder, nimdevel, fciowrappers, superlu\_dist, nimtests

## nimpack (gitlab.com)

- Contains git submodules for spack, nimrod-aa1 and/or nimrod-multispecies

### Proposal:

- Switch from nimall to nimpack
- Switch from bilder to spack
- Incorporate fciowrappers (two source files) into nimdevel externals
- Move nimdevel/nimuw to gitlab.com after APS-DPP

# Bilder vs spack for package management

## Bilder

- Open-source, Tech-X hosted
- Tech-X developed
- Tech-X supported
- Focus on Tech-X products
- Written in shell

## Spack

- Open-source, open-access
- Large developer community
- DOE supported
- Focus on HPC
- Written in Python
- Extensive documentation

Learning spack can be a transferable skill-set as it is in use by many HPC centers

# Spack tutorial based on nimpack documentation

<https://gitlab.com/NIMRODteam/nimpack>

(need to be logged into gitlab.com to view, send me a username if you haven't been added)

Spack features:

- Flexible dependency chains (e.g. MPI, OpenMP, OpenACC)
- Large built-in package support (e.g. SuperLU\_DIST)
- Ability to use system packages (e.g. system cmake & MPI)
- Developer workflows for many build configurations
- Package export as modules (e.g. module load nimrod)

# Moving the repository after DPP minimizes disruption

- Plan to retain commit history, keep select branches
  - If plan is finalized, will solicit input on which branch to keep/discard
- Make svn repository read-only after move to prevent (more) fragmentation

# Final thoughts

Any thoughts on alternatives?

If we continue with nimall/bilder, who will maintain it?

Do we want to do code review with merge requests?