



TEXAS ADVANCED COMPUTING CENTER

WWW.TACC.UTEXAS.EDU



TEXAS

The University of Texas at Austin

Lmod: The Lua-Based Environment Module System

Matthew Cawood

SC25 – St. Louis, Missouri

What is Lmod?

- ▶ Lmod is a **Lua-based environment module system** for managing software stacks on HPC systems.
- ▶ Supports both **Lua and Tcl modulefiles**.
- ▶ Enables **hierarchical MODULEPATHs, module families, and site policy enforcement**.
- ▶ Used at major HPC centers worldwide including TACC, NERSC, CSCS, and BSC.



Recap: Where We Left Off (SC24)

- ▶ Focused on improving modulefile scripting, hooks, and hierarchy handling.
- ▶ Introduced the next-generation tracking database for usage collection.
- ▶ Simplified configuration and policy definition for large sites.
- ▶ Feedback from SC24 deployments drove the Lmod 9 feature roadmap.

From 8.x to 9.x – What Changed Under the Hood

- ▶ Consolidated over 60 feature and performance patches into the new 9.x branch.
- ▶ New dependency engine reduces redundant file reads and improves load order resolution.
- ▶ Tracking database re-engineered – up to **100× smaller footprint** at TACC.
- ▶ Expanded site-policy features: irreversible unloads, hide/forbid logic.
- ▶ Maintains full backward compatibility with existing 8.x stacks.

New Features and Enhancements

- ▶ **Irreversible mode:** unloading can now set environment variables for cleanup.
- ▶ **depends_on_any():** allows dependency on any member of a module set.
- ▶ **Enhanced hide{/forbid{}**: expression-based control by user, group, or date.
- ▶ **Optional tracking v2:** database shrinks 100× while preserving analytics.
- ▶ **New hooks (e.g., decorate_module)** for module tagging and logging.

Performance and Reliability

- ▶ Spider cache and dependency engine now reduce ‘module avail’ from 4s → <0.5s on 40k-module trees.
- ▶ Refactored collection handling avoids NFS metadata bottlenecks.
- ▶ Improved `family()` logic speeds hierarchical resolution.
- ▶ Expanded compatibility: bash, zsh, fish, and nushell all supported.

Documentation and Usability

- ▶ Major expansion of `lmod.readthedocs.io` with in-depth technical docs.
- ▶ Added sections on advanced hooks, tracking v2, and policy examples.
- ▶ Clear migration guide from 8.x → 9.x.
- ▶ Real-world stats: TACC reduced usage DB size from 1 GB/day → 10 MB/day.

Site Policy and Governance

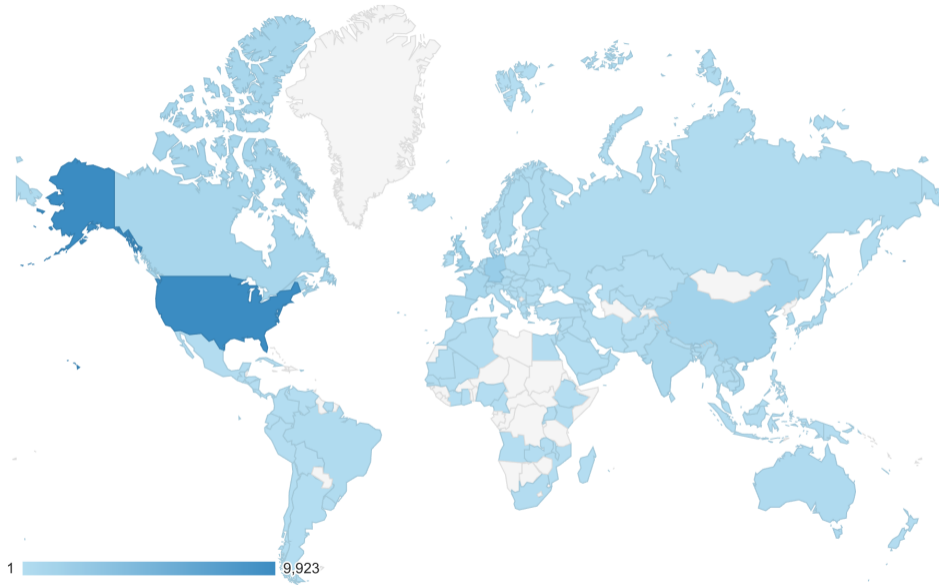
- ▶ Conditional **hide()/forbid()** rules by user, group, or time window.
- ▶ Deprecation warnings for outdated compilers or toolchains.
- ▶ Irreversible unloads enable post-cleanup or variable resets.
- ▶ Simplifies compliance and software lifecycle management.

Operational Impact at TACC and Partner Sites

- ▶ On Frontera and Stampede3: module load latency **reduced by 40%** in deep hierarchies.
- ▶ Tracking database ingest reduced **100×**.
- ▶ GPU/CPU module tagging integrated via site hooks for usage analytics.
- ▶ Single `lmod_config.lua` replaced multiple legacy init scripts.

Looking Ahead

- ▶ Ongoing performance profiling for massive module trees (>100k modules).
- ▶ Additional shell and scheduler integration (Slurm, PBSPro).
- ▶ Exploring distributed tracking and federated analytics.
- ▶ Continued collaboration via GitHub and EUM workshops.



Get Involved

- ▶ Source: github.com/TACC/Lmod
- ▶ Docs: lmod.readthedocs.io
- ▶ Mailing lists: [lmod-announce](#), [lmod-users](#).
- ▶ Contribute: report issues, propose hooks, share use cases.

Thank You

Questions? Feedback?

Contact: `mcawood@tacc.utexas.edu`

`lmod.readthedocs.io`