



TEXAS ADVANCED COMPUTING CENTER

WWW.TACC.UTEXAS.EDU



TEXAS

The University of Texas at Austin

Lmod at EB user Meeting

Robert McLay

January 27, 2021

Introduction



- ▶ Features and History
- ▶ Advanced Topics
- ▶ Future work?

Features

- ▶ Reads for TCL and Lua modulefiles
- ▶ One name rule.
- ▶ Support Software Hierarchy (but not required!)
- ▶ Spider Cache: fast `$ module avail`
- ▶ Properties (gpu, mic)
- ▶ Semantic Versioning: 5.6 is older than 5.10
- ▶ family(“compiler”) family(“mpi”) support
- ▶ Optional Tracking: What modules are loaded?
- ▶ Many other features: ml, collections, hooks, extended default, nag ...

depends_on()

- ▶ Modules X and Y depends on Module A
- ▶ ml purge; ml X; ml unload X; \Rightarrow unload A
- ▶ ml purge; ml X Y; ml unload X; \Rightarrow keep A
- ▶ ml purge; ml X Y; ml unload X Y ; \Rightarrow unload A
- ▶ ml purge; ml A X Y; ml unload X Y ; \Rightarrow keep A

Dynamic Cache files for Large Module Trees

- ▶ Groups that have a large number of specialize modules.
- ▶ Want Opt-in for these modules

Dynamic Cache files for Large Module Trees (II)

► bioPkgs.lua

```
prepend_PATH("LMOD_RC", "/path/to/cache_descript/descript.lua")  
if ( mode() ~= "spider") then  
    prepend_path("MODULEPATH", "/path/to/bioPkgs")  
end
```

► descript.lua

```
scDescript = {  
    {  
        dir = "/path/to/bioPkg/cacheDir",  
        timestamp = "/path/to/bioPkg/timestamp.txt",  
    },  
}
```

Lmod 8+ new features

- ▶ Extended Default
- ▶ The TCL interpreter is now (optionally) embedded with Lmod.
- ▶ New Function:
`extensions("numpy/1.16.4", "scipy/1.4")`
- ▶ A better way to handle special modules

Extended Default

- ▶ Long version number are a pain. (e.g. intel/18.0.4)
- ▶ With Extended Default: module load intel/18 will load the “highest” or marked default.
- ▶ Useful: Want to load intel/17 but don't remember which is the latest 17.0.* and intel/19.0.5 is the default.

Embedded TCL interpreter

- ▶ Lmod now embeds the TCL interpreter.
- ▶ Speeds up avail and load when there are many “.version” or “.modulerc” files.
- ▶ It is still faster to use “.modulerc.lua” files over TCL version files.

extensions() function

- ▶ `extensions()`: Tells users that a module has extensions
- ▶ E.G: python has `numpy` and `scipy`
- ▶ `extensions("numpy/1.16.4, scipy/1.4")`

extensions() function (II)

- ▶ Users can use spider to find extensions.
- ▶ Users can use avail to list extensions base name
- ▶ See examples

Checking your module tree 8.4.3+

- ▶ New command added:
`$LMOD_DIR/check_module_tree_syntax`
- ▶ Reports syntax errors across the entire `$MODULEPATH`
- ▶ Report which modules have multiple marked defaults sets
- ▶ Precedent order: default symlink, `.modulerc.lua`, `.modulerc`, `.version`
- ▶ Does not check `SYSTEM MODULERCFILE` for defaults.

Knowing what kind of default your module is (I)

- ▶ A user does “`module load Foo`”
- ▶ With Version `Foo/foo` and `Foo/3.12`
- ▶ Lmod must pick something.
- ▶ Suppose version 3.12 is the default.

Knowing what kind of default your module is (II)

- ▶ Lmod uses “loose version” idea from python.
- ▶ Version 3.12 \Rightarrow “00003.00012.*zf”
- ▶ Version foo \Rightarrow “*foo.*zf”
- ▶ Marked default \Rightarrow “^0003.00012.*zf”
- ▶ System Modulerc \Rightarrow “s0003.00012.*zf”
- ▶ User Modulerc \Rightarrow “u0003.00012.*zf”
- ▶ Ascii order: *, 0,1,2, ..., 9, ^, s, u

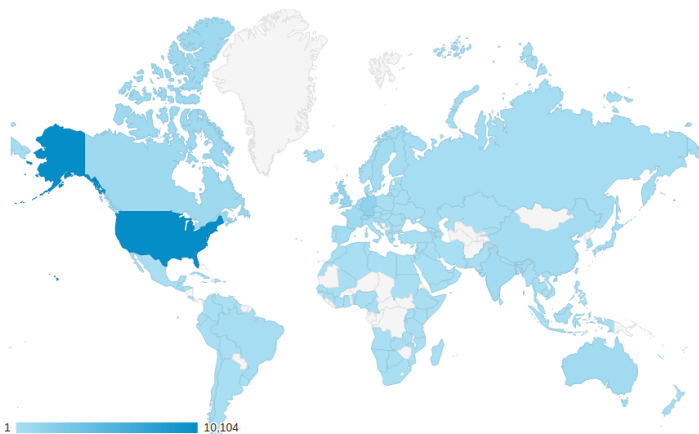
Knowing what kind of default your module is (III)

- ▶ Lmod sorts loose version strings to find the highest.
- ▶ Lmod now copies the loose version string to module table in the user env.
- ▶ Show example of “ml -mt” from Lmod 8.4.20
- ▶ New function `Mname:defaultKind()` available in `SitePackage.lua`:
- ▶ Returns: none, marked, system, user

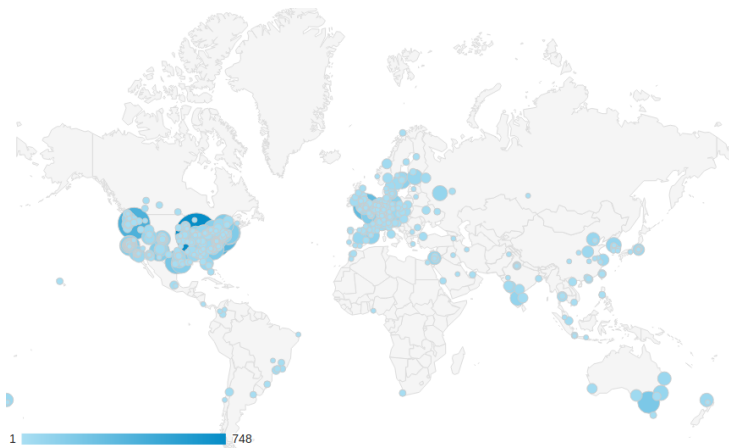
Future Work

- ▶ Lmod can optionally track usage.
- ▶ Future: Make it easier to not remember loads after 1 year.
- ▶ Get Lmod to support the break function.
- ▶ Support for Tmod4's advanced version specifiers `module load foo@2.4:2.8`
- ▶ A monthly discussion group?

Lmod Doc usage by Country



Lmod Doc usage by City



Conclusions: Lmod 8+



- ▶ Latest version: <https://github.com:TACC/lmod.git>
- ▶ Stable version: <http://lmod.sf.net>
- ▶ Documentation: <http://lmod.readthedocs.org>