

# Demo

# Demo

...but let's first talk about setups

# State of Lean 3 Setup

- lean+leanpkg: via GitHub Releases, or elan
- elan: Lean version manager, via curl | sh
- leanproject: store & restore .olean tars, via PyPI
- your favorite editor + Lean extension

**IT HAS BEEN**



**DAYS**

**SINCE THE LAST BROKEN LEAN SETUP**

customize

# State of Lean 3 Setup

- lean+leanpkg: via GitHub Releases, or elan
- elan: Lean version manager, via curl | sh
- leanproject: store & restore .olean tars, via PyPI
- your favorite editor + Lean extension

Look forward to further dependency fun in Lean 4:

- C compiler chain
- external FFI libraries
- Lean dependencies compiled into plugins

...and I don't want to write any of this

Can't someone else do that for us?

# Nix

Nix is a purely functional build system & package manager

- hermetic, reproducible builds against pinned dependencies
- built-in support for remote binary caches & distributed builds
- the world's largest package repository

...works on Linux, macOS and WSL

# Sebastian's Experiment: Nix-Based Lean Setup

Perfect .olean/.c/.o local & remote caching

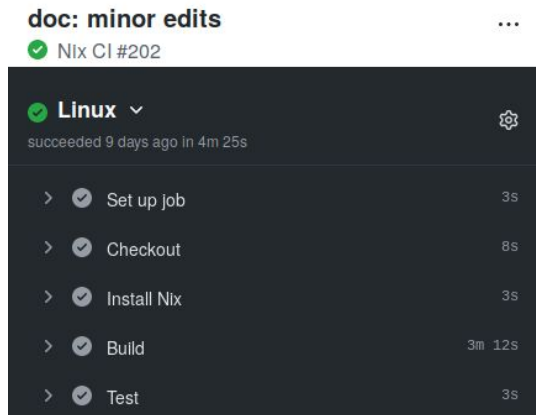
Perfectly reproducible Lean development setup

- change branches without losing progress
- build while changing branches, build commits without checkout



**10.14 GIB**  
STORAGE

<https://lean4.cachix.org/>





# Nix Setup: Unresolved Questions

- Does it scale?
- Is it actually simpler/more robust?
- Relationship with `leanpkg.toml`?

# Demo

for real this time

```
~/lean/deriv  master  ls
Deriv.lean  flake.lock  flake.nix
~/lean/deriv  master  nix run .#vscode-dev Deriv.lean
```

```
Built '/nix/store/rblka5kkknlds92jjai9s19xf4hmry-Init.Data.Nat.Basic.drv' in 5.3 s.
Built '/nix/store/4q8cznxsq2q6rr1ix8289lp3ncrliws-Init.Control.Option.drv' in 1.6 s.
Built '/nix/store/s7pv68lwwd79gld8qm6a8z9gppadvy27-Init.Control.Reader.drv' in 1.2 s.
```

Type 'h' for help.

✓ Lock flake inputs

✓ Query missing paths

✓ Fetch 137 / 137 store paths, 458.0 / 458.0 MiB

• Build 43 / 1111 derivations, 12 running

- leancpp (20 s) (buildPhase): [ 61%] Building CXX object util/CMakeFiles/util.dir/name\_generator.cpp.o
- Init.Notation-cc (11 s):
- vscode-1.52.1 (9 s) (fixupPhase): shrinking /nix/store/5myfrbjf1gshpjqqjihfsab3c9hvc8zhj-vscode-1.52.1/lib/vscode/code
- Init.Data.Option.Basic-cc (2 s):
- Init.Control.Except-cc (2 s):
- Init.Control.Option (1 s):
- Init.Data.Option.Instances (1 s):
- Init.Control.Reader (1 s):
- Init.Control.State (1 s):
- Init.Data.List.Basic-depRoot (0 s):
- Init.Data.Nat.Basic-cc (0 s):
- Init.Data.Nat.Control-depRoot (0 s):
- Init.System.Platform-depRoot (0 s):
- Init.WF-depRoot (0 s):

Deriv.lean > {} Deriv > {} Expr > add

```
14 partial def pown (a : Int) : Int → Int
15   | 0 => 1
16   | 1 => a
17   | n =>
18     let b := pown a (n / 2)
19     b * b * (if n % 2 = 0 then 1 else a)
20
21 partial def add : Expr → Expr → Expr
22   | Val n, Val m => Val (n + m)
23   | Val 0, f => f
24   | f, Val 0 => f
25   | f, Val n => add (Val n) fa
26   | Val n, Add (Val m) f => add (Val (n+m)) f
```

PROBLEMS 4 ...

Filter (e.g. text, `**/*.ts`, `!*/node_modules/**`)



Deriv.lean 4

unknown identifier 'fa' Lean 4 server [25, 47]