

Tips and Tricks to Fix your Nix

FOSDEM 2026

Yvan Sraka

[~yvan](#)

I am currently working for Numtide (a consulting company), contributing to Rust, Nix, and Haskell upstream tooling.

You can find me on Mastodon: <https://functional.cafe/@yvan>

On the menu

1. Debugging **Expressions**
2. Debugging **Builds**

Debugging Expressions

What's the Problem?

Nix is an **interpreted** (dynamically and implicitly typed) **functional** (with function currying by default) programming language.

What's the Problem?

Error reporting is **bad**; the line reported in the stack trace is often quite far away from the actual mistake you've made.

Quick Advertisement!

Look, **<https://sraka.xyz/posts/contracts.html>** is cool :)

Setup Your \$EDITOR

Use an LSP! Like `nil` and `nixd`. Configure checks on save or a pre-commit hook that can, e.g., run `deadnix` and `treefmt`.

Use the REPL

I use it all the time; it's also a great way to learn the Nix language.

--trace and break

Yay, debug prints... but **NEW:** a --debugger option!

Demo

```
let
    zA = 30;
    zB = builtins.trace 13 12;
in
{
    zC = builtins.break zA + zB;
}
```

Debugging Builds

What's the Problem?

Nix isn't incremental. When a command fails, you have to rebuild (*and almost always re-evaluate*) from scratch the whole thing, and there's no built-in way to interactively debug your issue...

Visualize Incremental Builds

`nix-output-monitor` gives you a nice interface to visualize what takes time in your build, and even better, `nix-fast-build` reuses the same TUI but builds your dependencies in parallel.

When It Doesn't Work

Use `nix derivation show` to inspect .drv files, and `nix-tree` or `nix why-depends` to trace dependency chains.

Demo

```
> nix why-depends /nix/store/*-pandoc-cli-3.6 \
    /nix/store/*-ncurses-6.5

/nix/store/fy9cyhdyb32jgvbjs7kwadfnx644dp97-pandoc-cli-3.6
/nix/store/wp8fa5j4qj1x4mysnkh196dkrhjyr3pg-lua-5.4.7
/nix/store/yw7vb4hamv9mqgbgf7598zvis7k2spyx-ncurses-6.5
```

To Go Further

- Debug a Failed Derivation with `breakpointHook` and `cntr`
- Unit Test Your Nix Code

Q/A