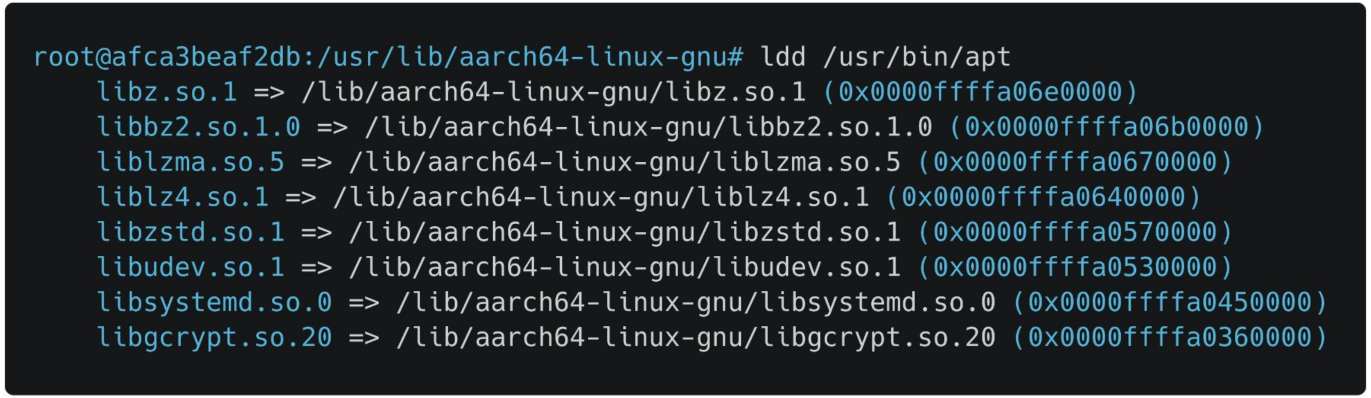
Was ist NIX?

Motivation

- Unter UNIX Systemen liegen Binaries und shared Objects im File-System
- Struktur Konventionen werden durch den **F**ile **H**ierarchy **S**tandard (FHS) beschrieben
- Binaries können z.B. gegen shared-libraries linken



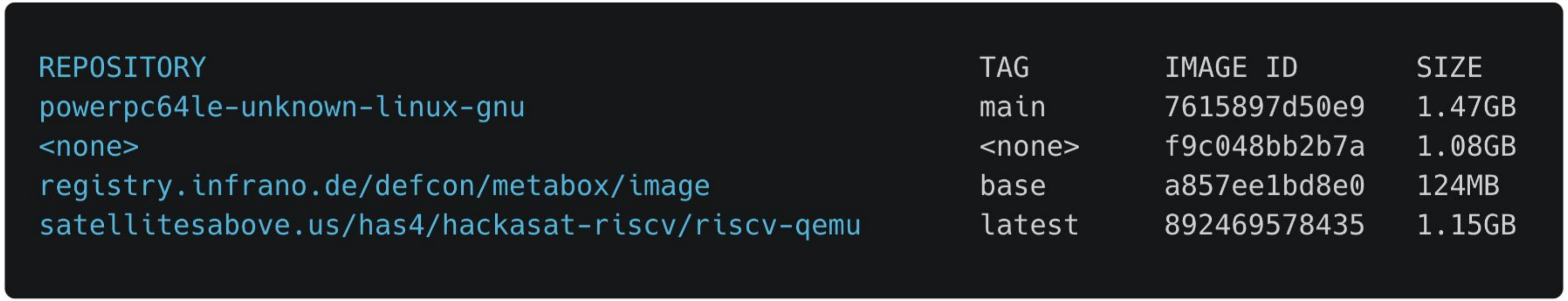
```
root@afca3beaf2db:/usr/lib/aarch64-linux-gnu# ldd /usr/bin/apt
    libz.so.1 => /lib/aarch64-linux-gnu/libz.so.1 (0x0000ffffa06e0000)
    libbz2.so.1.0 => /lib/aarch64-linux-gnu/libbz2.so.1.0 (0x0000ffffa06b0000)
    liblzma.so.5 => /lib/aarch64-linux-gnu/liblzma.so.5 (0x0000ffffa0670000)
    liblz4.so.1 => /lib/aarch64-linux-gnu/liblz4.so.1 (0x0000ffffa0640000)
    libzstd.so.1 => /lib/aarch64-linux-gnu/libzstd.so.1 (0x0000ffffa0570000)
    libudev.so.1 => /lib/aarch64-linux-gnu/libudev.so.1 (0x0000ffffa0530000)
    libsystemd.so.0 => /lib/aarch64-linux-gnu/libsystemd.so.0 (0x0000ffffa0450000)
    libgcrypt.so.20 => /lib/aarch64-linux-gnu/libgcrypt.so.20 (0x0000ffffa0360000)
```

Motivation

- Dependencies können dadurch nicht scharf beschrieben werden
  - Welche FFMpeg Version
  - Welcher C Compiler
  - Welche Flags
- Teilweise können bspw. Versionen im Filename codiert werden, z.B. /bin/python3.7
- Applikationen können konfligierende Anforderungen haben

Docker

- Jede Applikation kriegt einen eigenen Namespace
- + keine Konflikte mehr
- - jede Applikation shipped jetzt ihre eigene FHS



```
REPOSITORY                                        TAG        IMAGE ID       SIZE
powerpc64le-unknown-linux-gnu                     main       7615897d50e9   1.47GB
<none>                                            <none>     f9c048bb2b7a   1.08GB
registry.infrano.de/defcon/metabox/image          base       a857ee1bd8e0   124MB
satellitesabove.us/has4/hackasat-riscv/riscv-qemu latest     892469578435   1.15GB
```

💡 NIXs Idee

- Wenn der Build isoliert wäre, könnten wir sagen: Paket = Inputs + Receipe
- PacketHash = InputHash1 + InputHash2 + … + Hash(Receipe)
- Mit diesem PacketHash können wir auch jedem Packet einen wohldefinierten Ort im File System zuordnen: /nix/store/packetHast-packetName/
- Artefakte *fast* immer gleich

Wie baut NIX Pakete?

- NIX evaluiert Receipe in DSL => ✨ Package Derivation ✨
- Alle Dependencies (durch Hash!), Dateien, Env, Builder
- Outputs inkl. Pfad

```
{
  "/nix/store/sg3sw1zdddfkl3hk639asml56xsxw8pf-hello-2.10.drv": {
    "outputs": {
      "out": {
        "path": "/nix/store/dvv4irwgdm8lpbhdkqghvmjmjknrikh4-hello-2.10"
      }
    },
    "inputSrcs": [
      "/nix/store/9krlzvny65gdc8s7kpb6lkx8cd02c25b-default-builder.sh"
    ],
    "inputDrvs": {
      "/nix/store/8pq31sp946581sbh2m18pb8iwp0bwxj6-stdenv-linux.drv": [
        "out"
      ],
      "/nix/store/cni8m2cjshnc8fbanwrxagan6f8lxjf6-hello-2.10.tar.gz.drv": [
        "out"
      ],
      "/nix/store/md39vwk6mmi64f6z6z9cnnjksvv6xkf3-bash-4.4-p23.drv": [
        "out"
      ]
    },
    "platform": "x86_64-linux",
    "builder": "/nix/store/kgp3vq8l9yb8mzghbw83kyr3f26yqvsz-bash-4.4-p23/bin/bash",
    "args": [
      "-e",
      "/nix/store/9krlzvny65gdc8s7kpb6lkx8cd02c25b-default-builder.sh"
    ],
    "env": {
      ...
    }
  }
}
```

Vorteile

- Reproduzierbarkeit
- Binary Caching
- Keine Kollisionen => Beliebig viele Versionen vom selben Paket können existieren
- Trivial Parallelisierbar

```
$ readelf /nix/store/zlllyiaig35pg5va0z5sh4yndv37v6vx-networkmanager-1.42.2/bin/nmtui -d
Dynamic section at offset 0x96660 contains 34 entries:
  Tag        Type                         Name/Value
 0x0000000000000001 (NEEDED)             Shared library: [libnm.so.0]
 0x0000000000000001 (NEEDED)             Shared library: [libgio-2.0.so.0]
 0x0000000000000001 (NEEDED)             Shared library: [libgobject-2.0.so.0]
 0x0000000000000001 (NEEDED)             Shared library: [libglib-2.0.so.0]
 0x0000000000000001 (NEEDED)             Shared library: [libnewt.so.0.52]
 0x0000000000000001 (NEEDED)             Shared library: [libc.so.6]
 0x0000000000000001 (NEEDED)             Shared library: [ld-linux-x86-64.so.2]
 0x000000000000001d (RUNPATH)            Library runpath: [/nix/store/zlllyiaig35pg5va0z5sh4yndv37v6vx-
networkmanager-1.42.2/lib:/nix/store/5fk74drrnrhgmcwxvsnmv2lx1srgdfkp-glib-
2.74.5/lib:/nix/store/7gxa4fsdyqv1cqjf3d65kn7yz3bmbghm-newt-
0.52.23/lib:/nix/store/8xk4yl1r3n6kbyn05qhan7nbag7npymx-glibc-2.35-224/lib]
 0x000000000000000c (INIT)               0x415000
```

NixOS

.DRV
Wait, It's just another Package?
Always has been
imgflip.com

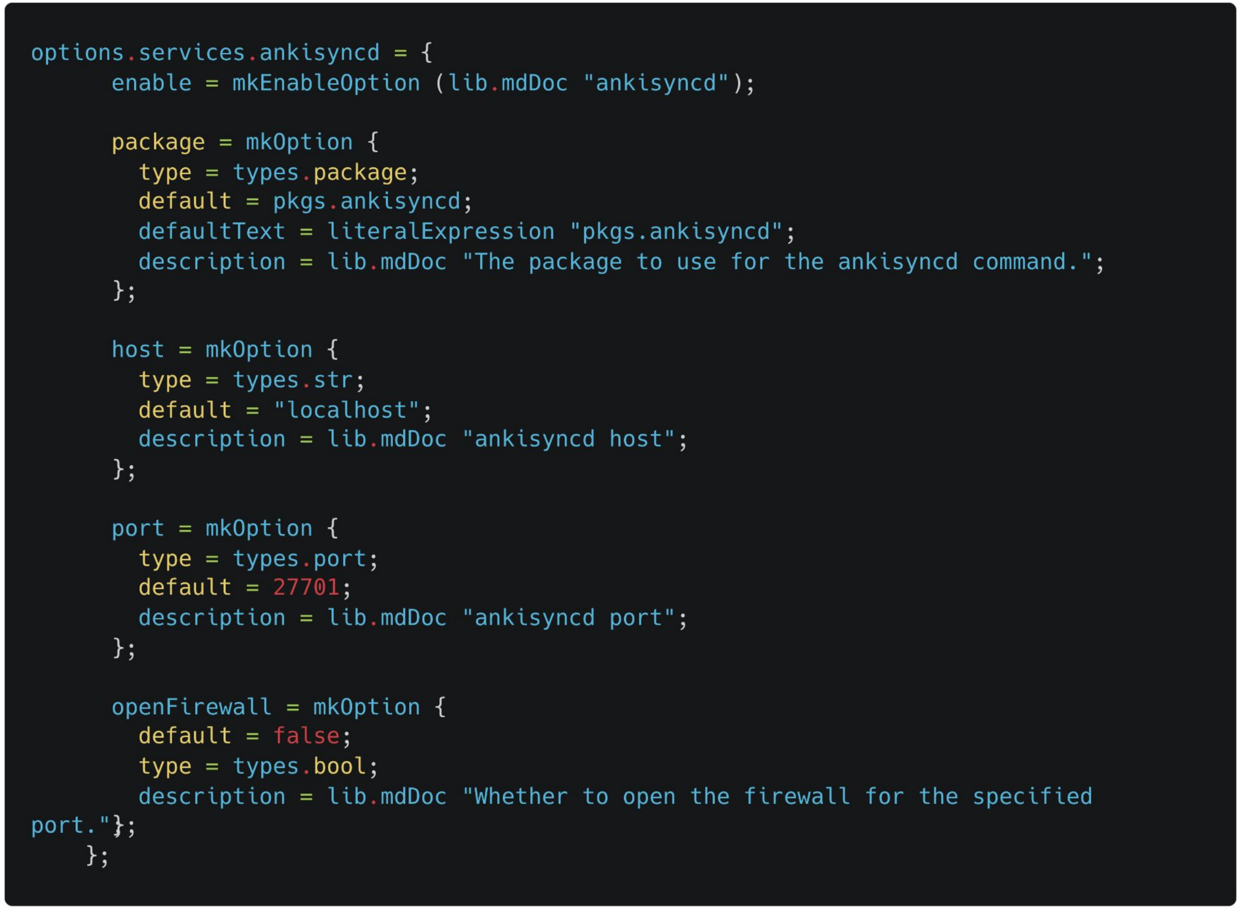
NixOS

- Dein System ist ein Package mit allen Applikationen und Config Files als Input und
  - SystemD Config
  - EFI Partition
  - PATH
  - etc.
- als Output
- Bei Systemstart kann beliebige Systemconfig gewählt werden
- Rollbacks for free

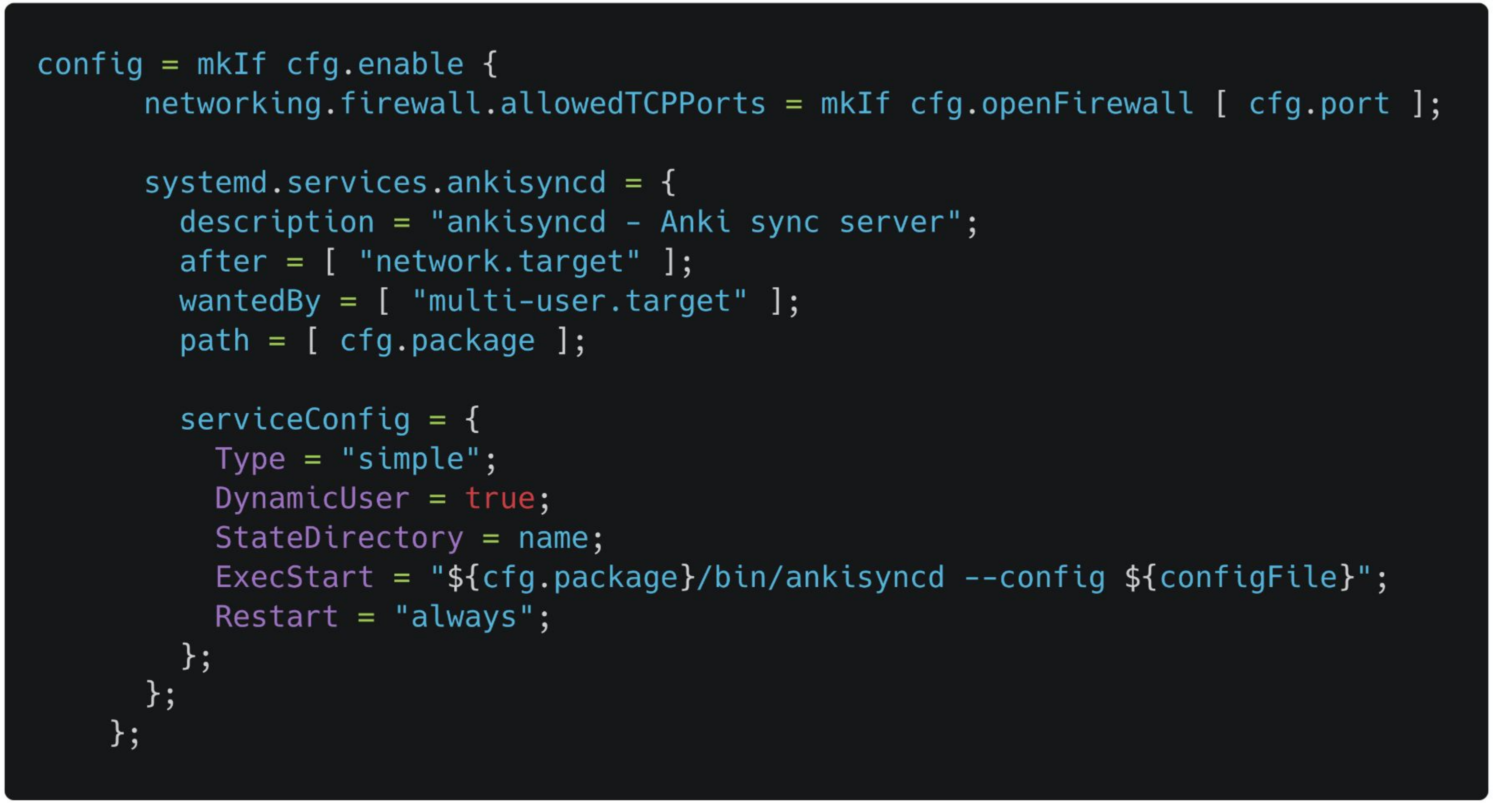
```
{ config, lib, pkgs, ... }:

...

environment.shells = [ "/run/current-system/sw/bin/zsh" ];

users = {
  mutableUsers = false;
  extraGroups.docker.gid = lib.mkForce config.ids.gids.docker;
  extraUsers = [
    {
      uid             = 2000;
      name            = "cstrahan";
      group           = "users";
      extraGroups     = [ "wheel" "networkmanager" "docker" "fuse" "vboxusers" ];
      isNormalUser    = true;
      passwordFile    = "/etc/nixos/passwords/bela";
      useDefaultShell = false;
      shell           = "/run/current-system/sw/bin/zsh";
    }
  ];
};

nix = {
  package = pkgs.nixUnstable;
  useSandbox = true;
  binaryCaches = [ "https://cache.nixos.org" ];
  trustedBinaryCaches = [ "https://cache.nixos.org" ];
  requireSignedBinaryCaches = true;
  distributedBuilds = true;
};

...
```

NixOS Modul

- quasi Terraform Provider
- übersetzt deklarative Config in System Config (Services anlegen, Scripts / Config Files generieren, etc.)



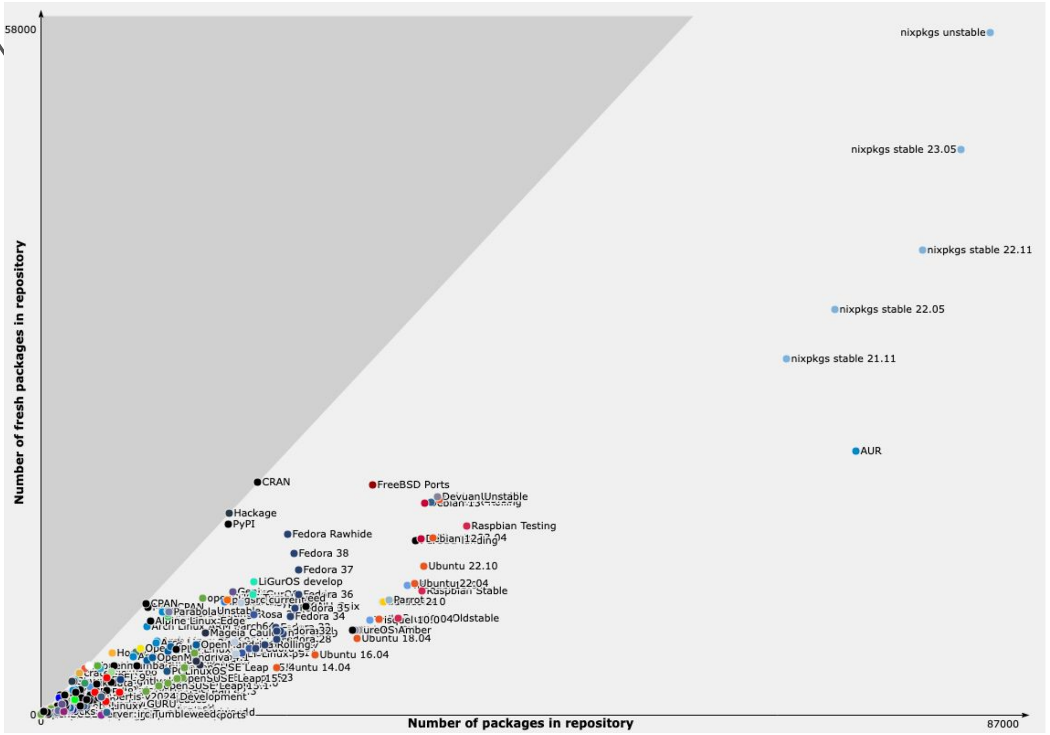
```nix
options.services.ankisyncd = {
      enable = mkEnableOption (lib.mdDoc "ankisyncd");

      package = mkOption {
        type = types.package;
        default = pkgs.ankisyncd;
        defaultText = literalExpression "pkgs.ankisyncd";
        description = lib.mdDoc "The package to use for the ankisyncd command.";
      };

      host = mkOption {
        type = types.str;
        default = "localhost";
        description = lib.mdDoc "ankisyncd host";
      };

      port = mkOption {
        type = types.port;
        default = 27701;
        description = lib.mdDoc "ankisyncd port";
      };

      openFirewall = mkOption {
        default = false;
        type = types.bool;
        description = lib.mdDoc "Whether to open the firewall for the specified
port."};
    };
```



```
config = mkIf cfg.enable {
      networking.firewall.allowedTCPPorts = mkIf cfg.openFirewall [ cfg.port ];

      systemd.services.ankisyncd = {
        description = "ankisyncd - Anki sync server";
        after = [ "network.target" ];
        wantedBy = [ "multi-user.target" ];
        path = [ cfg.package ];

        serviceConfig = {
          Type = "simple";
          DynamicUser = true;
          StateDirectory = name;
          ExecStart = "${cfg.package}/bin/ankisyncd --config ${configFile}";
          Restart = "always";
        };
      };
    };
```

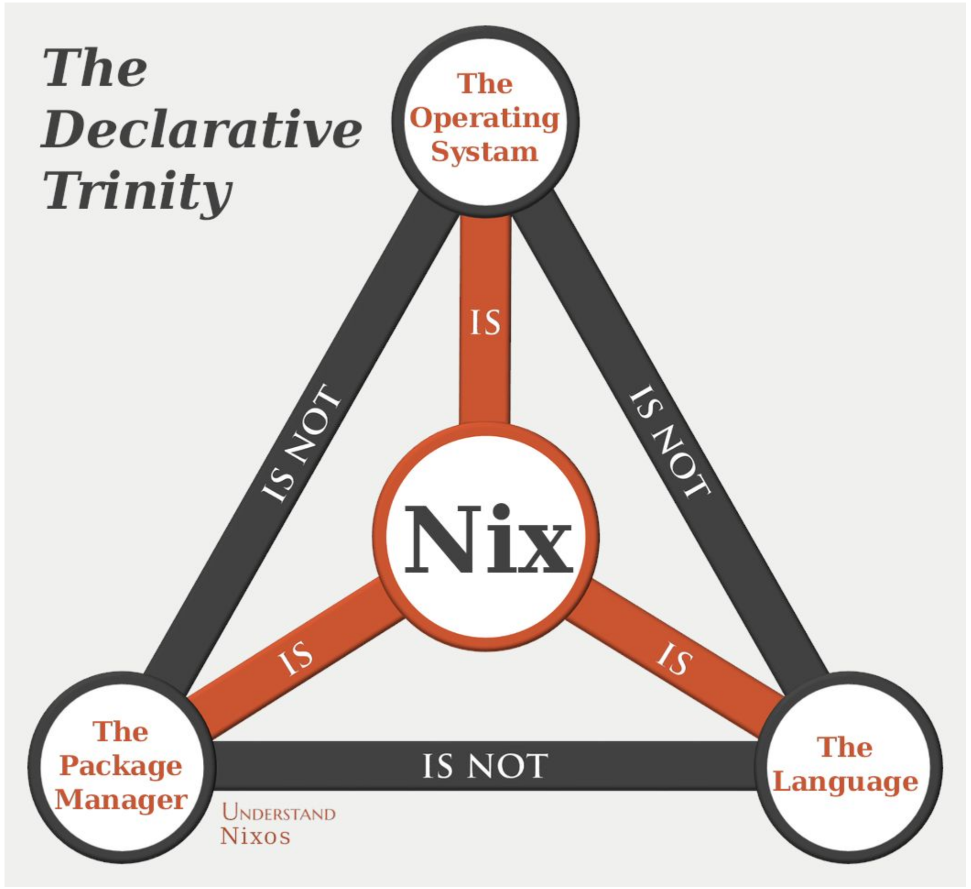
Nixpkgs

- Package Repository / Binary Cache für N
  Pakete und Module
- Definitionen in *Nix*, der Domain Specific
  Packaging Language von Nix



Fragen?

Folien basierend auf
https://serokell.io/blog/what-is-nix


The Declarative Trinity
The Operating System
Nix
The Package Manager
The Language
IS
IS
IS
IS NOT
IS NOT
IS NOT
Understand Nixos