



# Flatcar Linux: An Immutable OS for Containers and Kubernetes

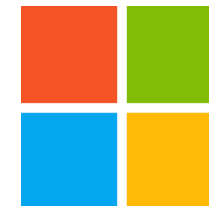
Jan Bronicki

# whoami



@John15321

## Jan Bronicki



Graduate of PWr:

- Automatyka I Robotyka @ W4
- Zarządzanie I Inżyniera Produkcji @ W10

Flatcar Maintainer &  
Software Engineer @ Microsoft

# What is Flatcar Linux

## A minimal container optimized Linux distribution

- Community stewarded
- Made for running containers – focused on doing this and only this well
- Think of the OS itself as being treated like a container image
- Orchestrated & Unorchestrated: k8s, Docker, Podman etc
- Immutable OS
- Automatic Handsfree updates



# Community Stewarded

What does it mean? Who are we?

# Flatcar is a CNCF Project

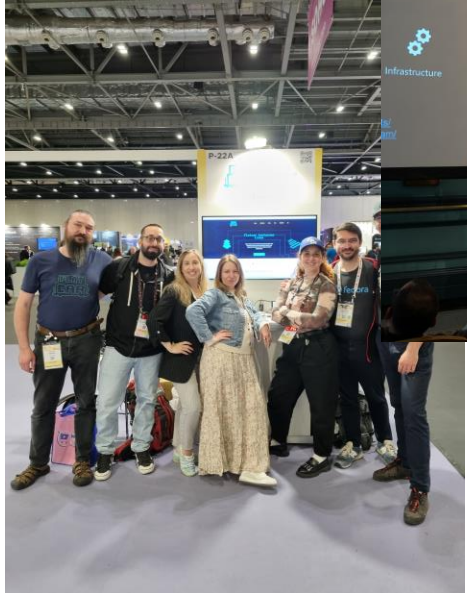


**CLOUD NATIVE  
COMPUTING FOUNDATION**



Flatcar was accepted to CNCF on  
2nd of August 2024

# Community stewarded



# Community stewarded

## Active contributors

Active contributor is an individual who performed activities such as commits, issues, or pull requests during the selected time period. [Learn more](#)

☒ Include collaborations ⓘ

287 ⬆️ 17.6% (+43)  
vs. 244 last period

Weekly

Monthly

Quarterly



## Organization



Microsoft



Cloudbase Solutions S.R.L.



Google Summer of Code



mailbox.org



Equinix Services, Inc.



STACKIT



dpark.io



Gentoo Linux



UCL



Red Hat, Inc.



Qube Cinema



Wekk



University of Rochester



Pleaco



Wiz



IONOS SE



AloTech-Full-Stack-Bootcamp



Netic A/S



Hedgehog

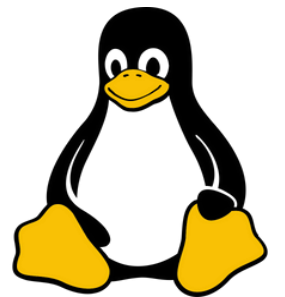


Rackspace US, Inc.

# Flatcar Container Linux FOSS Collaborations

## Who do we collaborate with:

- Gentoo, Fedora CoreOS
- Grub, Drakut, Systemd, Afterburn/Ignition + other Cloud Native Projects & Tools
- Member of Openwall's oss-security non-disclosure list
- Co-Founder of UAPI Group (cross-distro SIG for immutable, image based Linux)



# Flatcar Runs Everywhere



## Environment

Cloud

Virtualization

Bare metal

## Options

Azure, AWS, GCP,  
and many more

VMware, VirtualBox,  
libvirt, QEMU

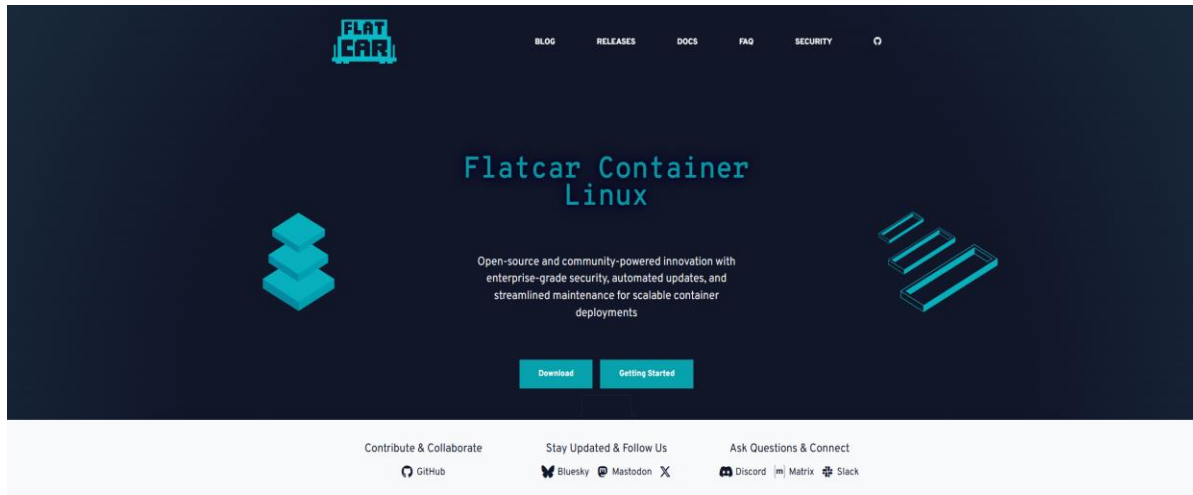
PXE / iPXE

**Also:** Terraform, Go bindings, Cluster API

# Flatcar in Action

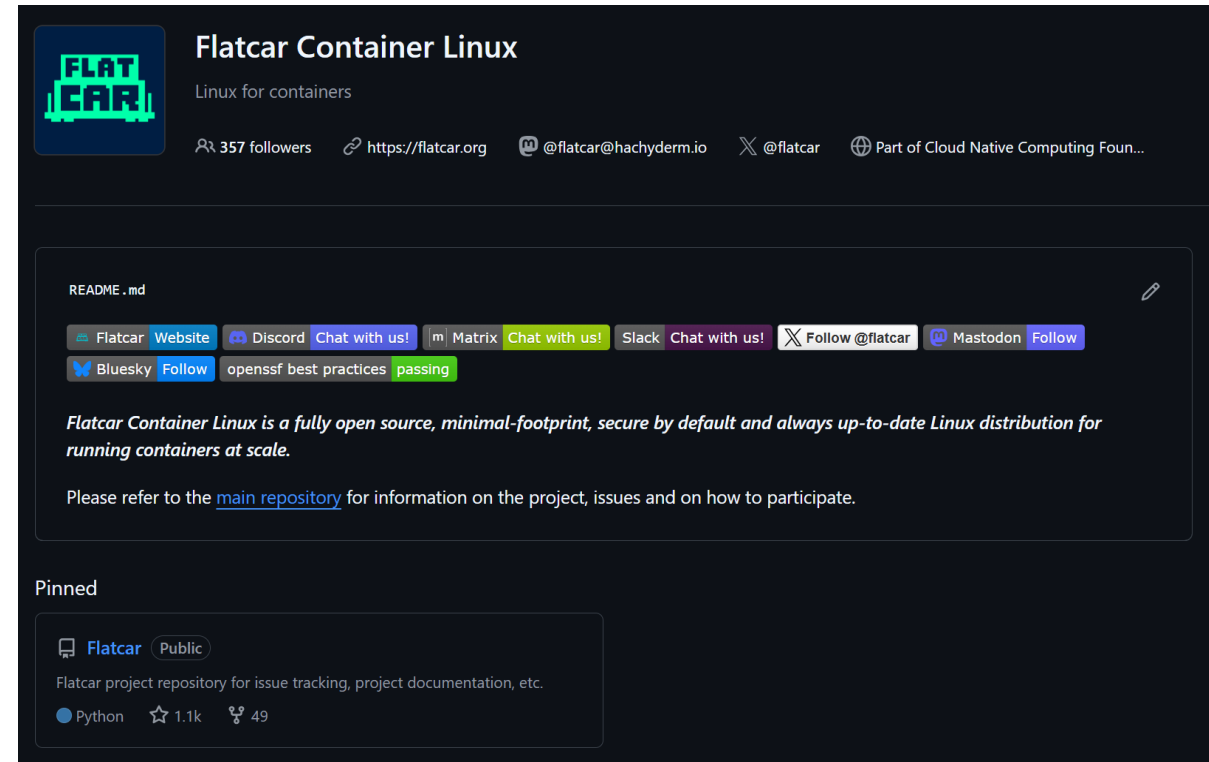


# Community stewarded



## Why Flatcar Container Linux?

Imagine an OS that works as hard as you do—streamlining your container infrastructure while cutting costs, reducing headaches, and safeguarding your business. Flatcar is engineered for those who demand efficiency, reliability, and peace of mind.



# Community stewarded

Discord, Matrix, Slack - Our day-to-day comms

Office hours - Every 2nd Tuesday, 2:30pm UTC

Dev Sync - Every 4th Tuesday, 2:30pm UTC

Bug Smash - Every last Friday of a month





# Flatcar Container Linux

What is it and how do I use it?

# Flatcar?

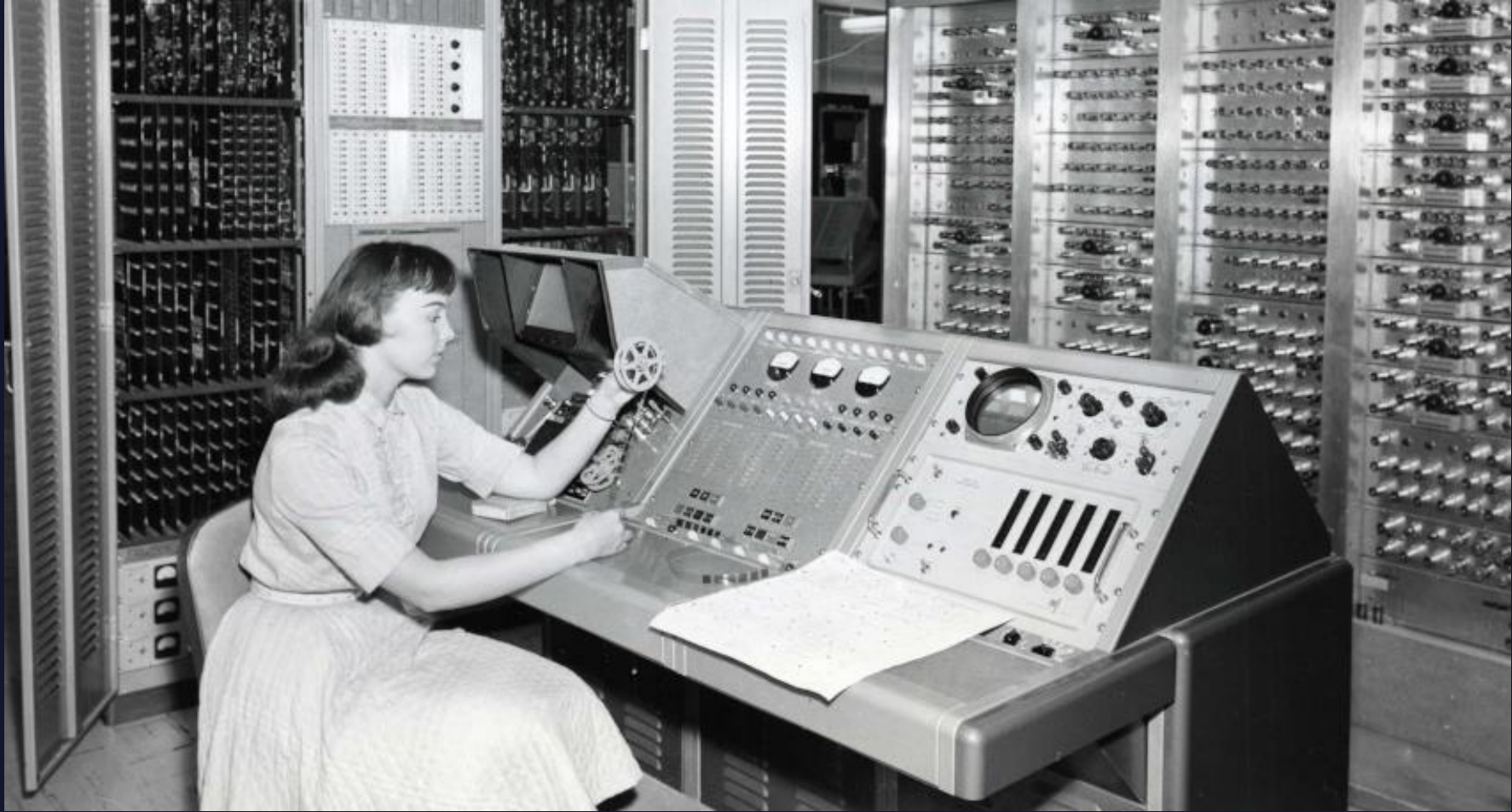


# Flatcar!

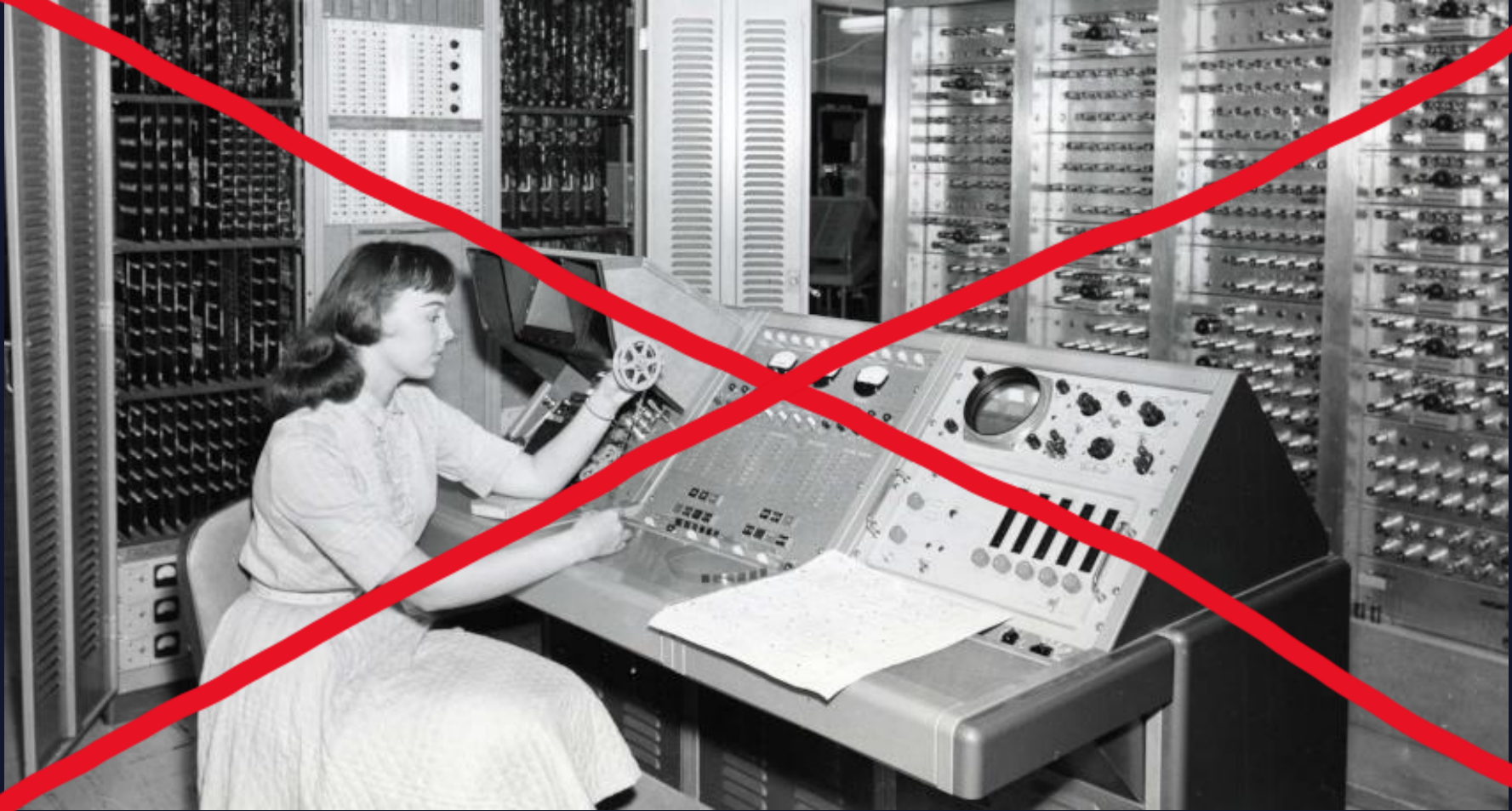
FLAT  
CAR



# General Purpose Linux UX Philosophy



# Not Our Goal



# Flatcar Container Linux UX Philosophy



# Functionality, not Features

## General-purpose Linux

Choose your shell,  
desktop, language stack

Manage updates:  
backports vs. new repos  
vs. waiting

Think about features: this  
version, this package

You build the system you  
want

Components may shift  
under your feet

## Flatcar

We choose for you  
(no pkg manager)

Whole OS updates  
automatically

Think about functionality:  
does it work?

We deliver a light switch

Contract: runtime stays  
the same

# Provisioned, not Installed

## Install

Make choices during setup

Interactive process

Each machine slightly different

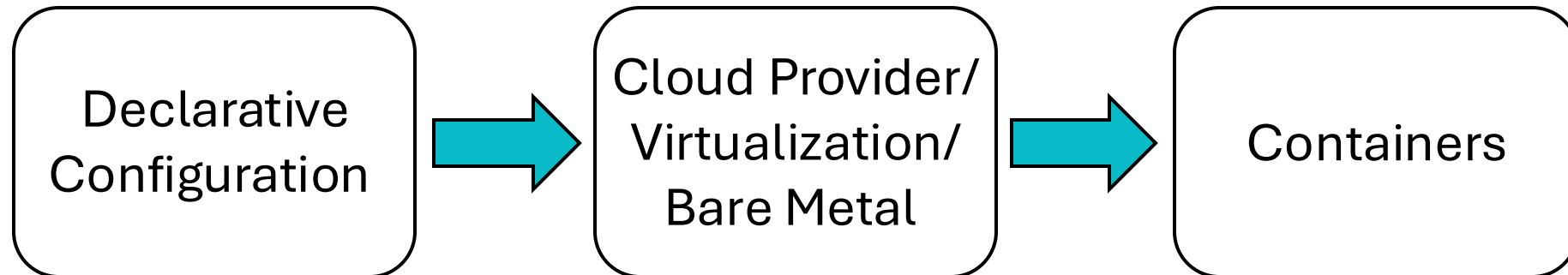
## Provision

Declare what you need up front

One declarative configuration

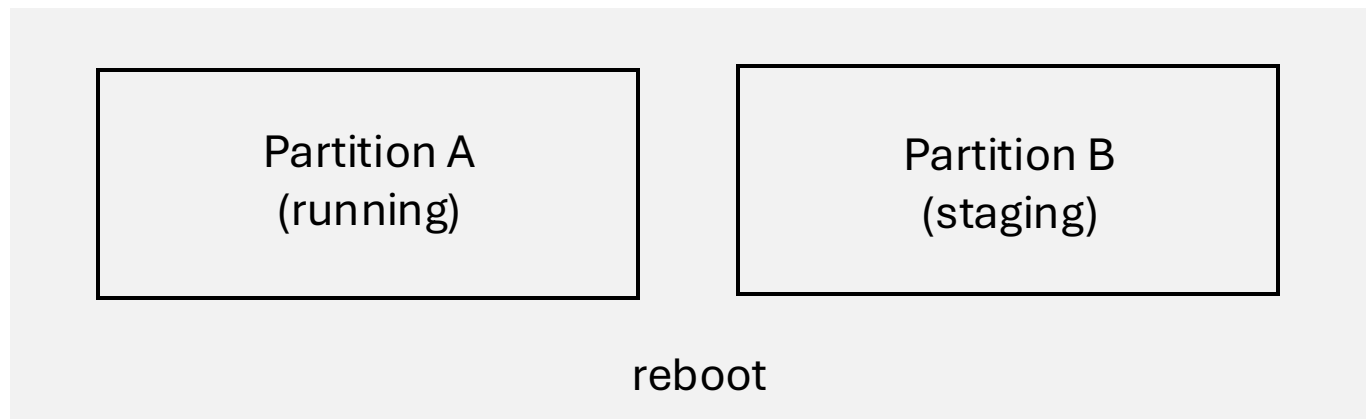
Every machine identical

**Same idea as containers:** you don't bake config into the image - you write config and apply it at launch.



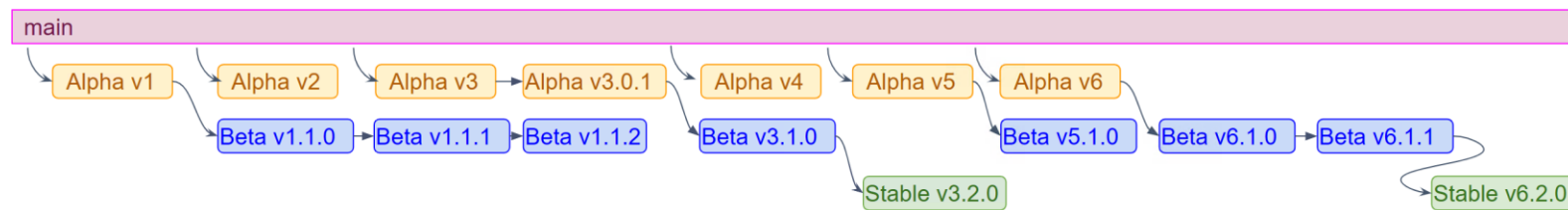
# Immutable by Design

- First boot: provisioned from config. After that: the base OS doesn't change
- OS is read-only and dm-verity protected
- No individual package updates - the entire OS updates as one unit
- Same config + same base OS = identical machine every time
- System extensions (sysexts)



- Verified image started to inactive partition
- Reboot activates the new OS
- Rollback = reboot to old partition
- No intermediate states - it works or it rolls back

# Channels



## Alpha → Beta → Stable (+ LTS)

- **Alpha:** Fully tested, may have incomplete features. For developers.
- **Beta:** Production-ready. Run as canaries alongside stable.
- **Stable:** Widespread production. Promoted from beta.
- **LTS:** Long-term support track for environments that need slower change



# Demos!

 [github.com/John15321/SesjaLinuksowa](https://github.com/John15321/SesjaLinuksowa)



 flatcar.org



**Thank you!**