Ambient CI engine

2026-11 at minidebconf in Cambridge

Lars Wirzenius, the Radicle project

Slides and other info



liw.fi/2026-minidebconf

Demo: small

- create a projects file as below
- `ambient run hello`
- `ambient log hello--format html --output /tmp/log.html`
- `firefox /tmp/log.html`

```
projects:
  hello:
    image: /scratch/ambient-images/ambient-boot.qcow2
    source: /home/liw/radicle/pathdedup
    plan:
      - action: shell
        shell: |
          echo hello, world
```

Demo: real

```yaml
projects:

  dummy:
    image: /scratch/ambient-images/ambient-boot.qcow2
    source: /home/liw/radicle/pathdedup
    pre_plan:
      - action: cargo_fetch
      - action: deb_get
        suite: trixie
        packages:
          - cargo-outdated
          - libc6-dev
          - libc-dev-bin
          - libc-bin
    plan:
      - action: deb_install
      - action: cargo_clippy
      - action: cargo_test
      - action: shell
        shell: |
          cargo outdated --offline
      - action: deb
    post_plan:
      - action: dput
```

What's wrong with CI?

Debugging

- push a change, wait for CI to run, try to figure out why it failed
- repeat, repeat, repeat, ...

Capacity

- competition for limited shared resource
- waiting is anti-fun

Generality

- usually only containers
- what about other operating systems?

Security

- run untrusted code, especially dependencies
- vetting and review do not find many problems
- better to prevent bad things from happening

Ambient CI: what?

Make it safe and secure to run CI on untrusted code.

A CI system I actually want to use.

The road to Ambient

- Jenkins
- Ick
  - control Jenkins, which used virtual machines
  - also, `libvirt` + `vmdebootstrap`
- Ick
  - `systemd-nspawn`
- Other CI systems at various jobs
- A fateful breakfast with Daniel and Rob at Lazy Fox
- Dream and vision
- Name, the difficult part

Later:

- Development of alpha version of Ambient

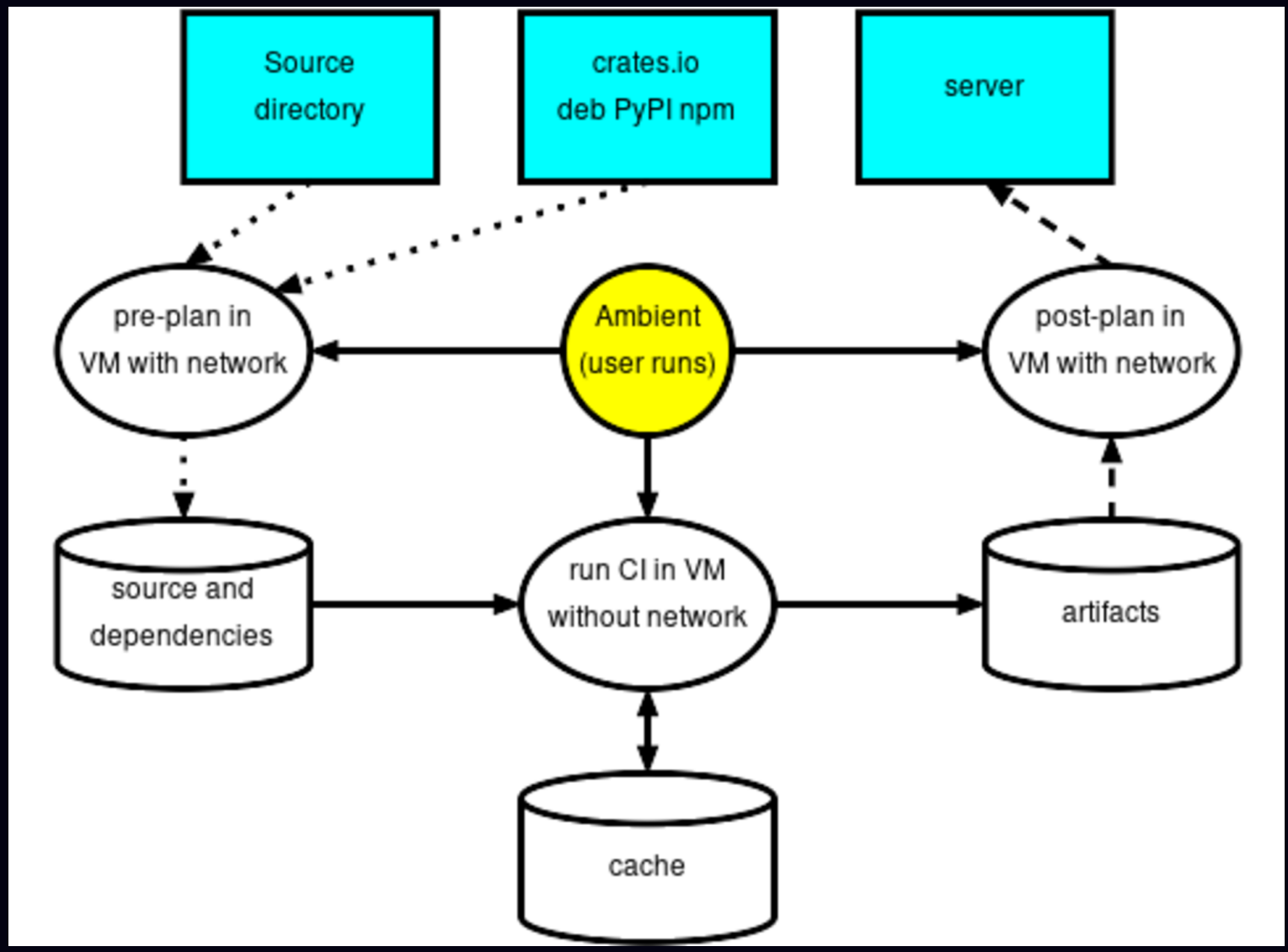
Even later:

- Paid to work on Radicle CI
- Paid to work on Ambient, too.

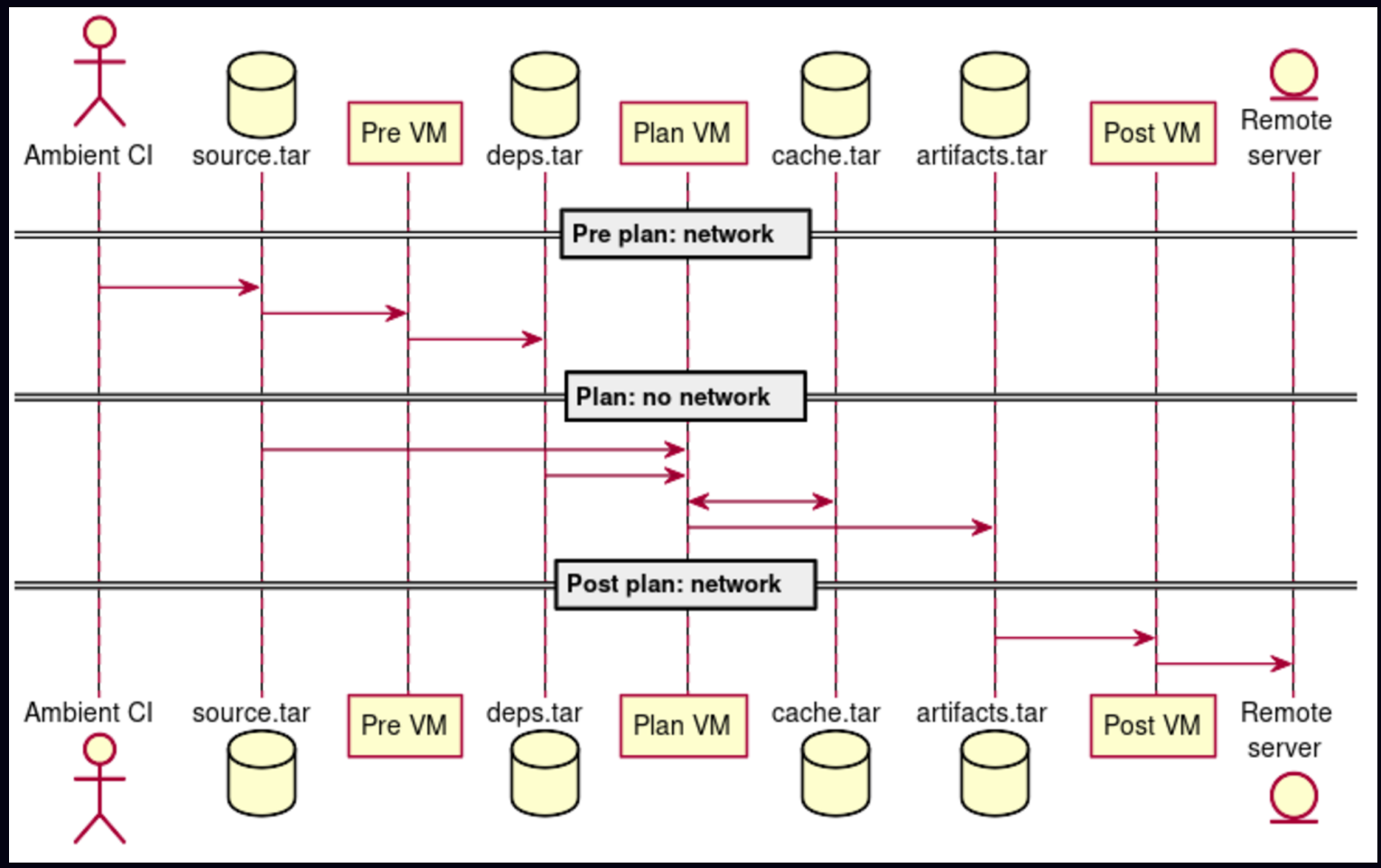
Shape of Ambient

- Local first: you run `ambient` to run CI.
  - "CI as a service" via Radicle
- QEMU + `kvm`
- You provide a VM image
  - I have one with Debian 13
  - cloud images work, ish

Ambient CI: how?



Ambient CI: how?



Ambient security

What MUST be safe and secure:

- Pre and post plan actions in Ambient
  - do NOT execute any code from software under test
  - must ONLY execute trusted, vetted code
  - host is isolated, but guest must not attack via network

- QEMU and KVM
  - must NOT allow code to escape containment
  - there may be a problem with this
  - all clouds will suffer if containment is breached

- System `tar` to create, extract
  - must NOT overwrite or include files not specified

Ambient with Radicle

- No server component in Ambient that reacts to changes in a repository.
  - or even changes to source directory
  - user must always invoke `ambient` by hand

- I use Radicle for this.
  - Radicle node always runs on my laptop
  - Radicle CI runs on a server
  - changes synced from my laptop to CI server trigger a run

- A self-standing server component for Ambient would be welcome.
  - could support any other Git forge
  - I don't have time or energy for this myself

Poll



Have you ever flown in a Zeppelin or other airship? Or want to?

I eat dogfood.



- Ohiopetwatch, CC BY-SA 3.0
- https://creativecommons.org/licenses/by-sa/3.0
- via Wikimedia Commons

Demo site

`callisto.liw.fi`

- Radicle CI with Ambient
- open source projects
- no payment required to use
- I learn what it's like to host this

Possible improvements

- Shell access to VM on failure.
- Better VM base image management.
- Build VM base image in Ambient.
- Better support for cloud images.
  - any VM image with `cloud-init` support
  - idealy, any operating system, too
- Non-linear plans with concurrency.
- Attestation of artifacts.
- More continous deployment.
- Base images that aren't Debian/amd64.

Takeaway

I do not say Ambient is the best CI system that will ever exist. That would be hubris and I am old and wise.

- Ambient works. At least for small projects.
- Try it, maybe?
- Tell me what you want and need.

Questions?



lazyfox.fi in Helsinki

Recommended, but be warned about starting projects.