



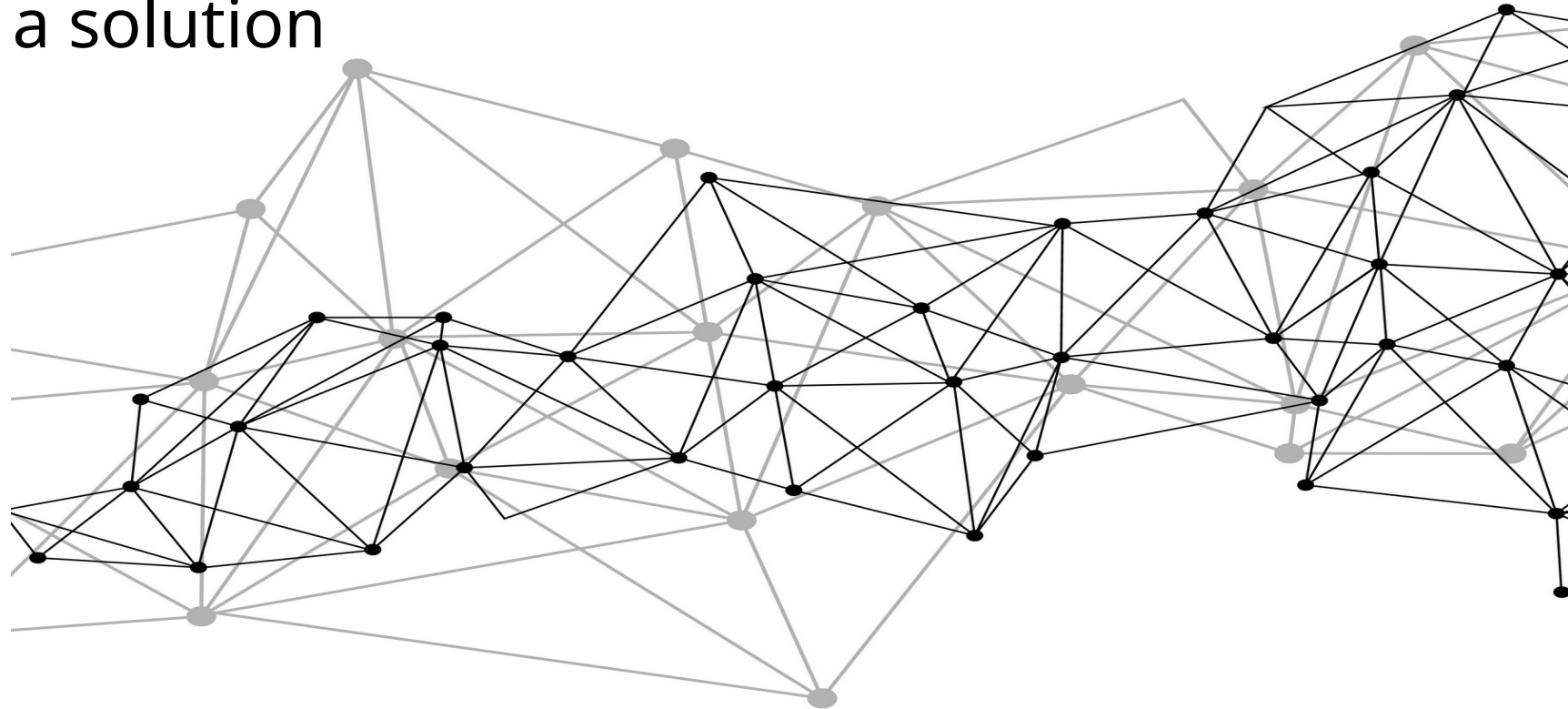
Trusted - Automated - Private Platform as a (self hosted) Service

- Ready to serve you, your family, your community, your business
- Connected to the Internet but not dependent on the Internet
- Resilient, Backed up, and ready for off-grid when desired
- Implemented with a solid security architecture and automated patching strategy
- Fully opensource
- Running on your own hardware

Lars Rossen, August 2025

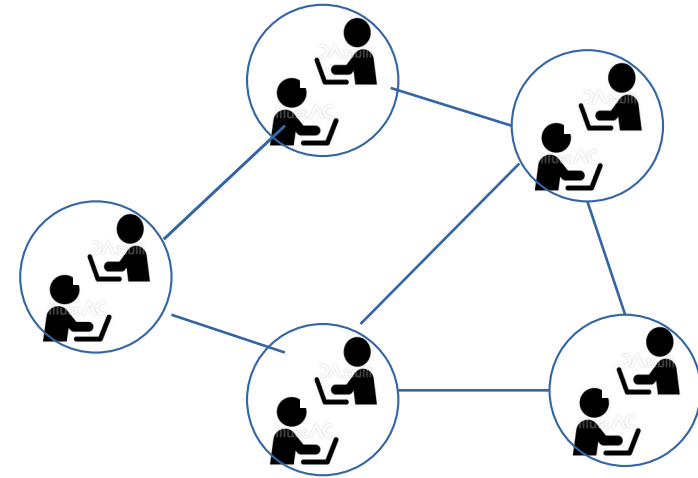
TAPPaaS: It is really about “connecting the dots”

- The Problem
- The Vision & Design Principles
- Intermezzo: clearly this must already be solved?
- The Structure of a solution
- The Roadmap
- Call for action

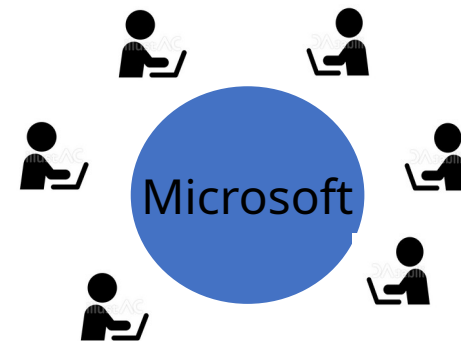


Problem Part One: The Internet is Broken

- The original “**intent**”:
 - Internet was an interconnection of networks.
 - Each network was self contained but benefited from collaboration and federation with other networks
- We lost our way and today the interconnected way of the Internet imply there is no “local” network that works



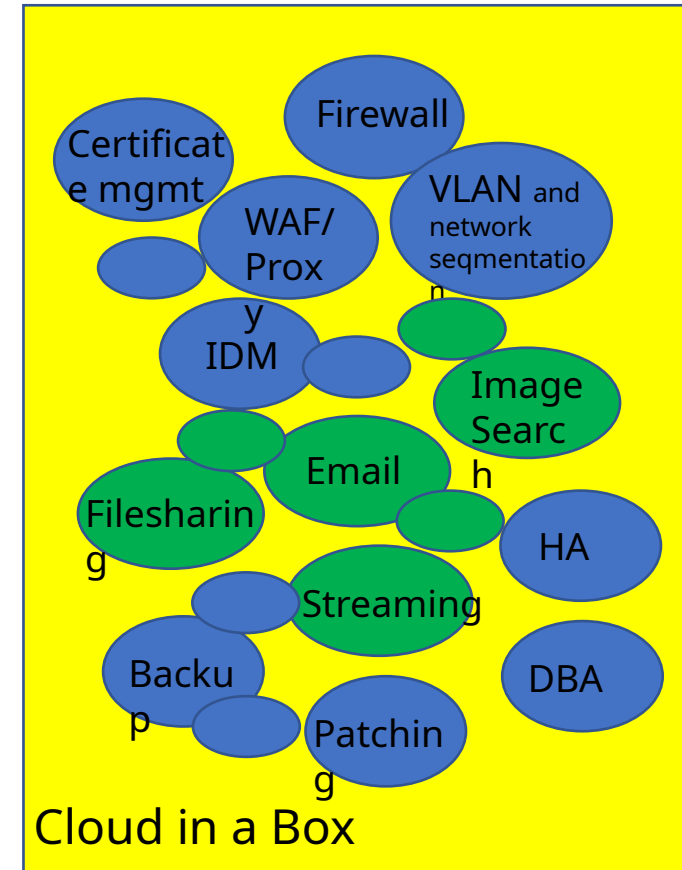
Bazar
Model



Cathedral
Model

Problem Part Two: Today's IT is complex

- Expectation on what IT can do have increased exponentially the last 30 years
- The big 5 have done a fantastic job of packaging up complexity
 - Single identity
 - Everything is interlinked and shareable within the ecosystem
- You can get all the ingredients for Free (FOSS)
 - But you need to make the (tapas) dish yourself



Why Should I care:

- Cost

Simple cost: Netflix+Spotify+1Tb cloud storage: 50\$/md per family: 600\$/year
TAPPaaS Server: 2K\$: should last 5 years

- Privacy

Convenience wins over Privacy: Let us revert that

- Resilience

What if Google decide you are a spammer?
What if Russia disrupt the Internet

Vision

- Store all pictures locally, without losing phone integration and search.
- Have your own Netflix and Spotify, no fees, you own your films and music
- Have a micro internet that works even if power and internet is down
- Own your data
- Get rid of the ever increasing cloud prices
- Ensure you small business, is not being held ransom by hyperscalers
- Ensure you can still “work” if there is a global outage.

Springle on Local AI

Goal / Vision

Create a small but scalable deployment platform with the following characteristics:

- OpenSource based: ensure no commercial entity can take control
- GitOps based: ease of deployment and upgrade/patch
- Data encrypted in transit and at rest
- Implement Zero Trust architecture
- Provide common platform service for cloud deployment of Applications/Services
- Must function with limited internet access
- Delivery model: Onprem Platform as a Service.
 - With no lockin: A consumer of the platform can at any time take over the management, or export the data in a free and open format

Nail all deployment decisions

Clearly this is already solved!!

- Think Home Lab.
 - Proxmox + OPNsense + Nextcloud
 - HexOS, ZimaOS, CasaOS
- Think Home Lab YouTubers
 - Techno Tim, Jim's Garage, ...
 - Home Network Guy, Lawrence Systems, ...

Deployment decisions are confusing

So who is this for

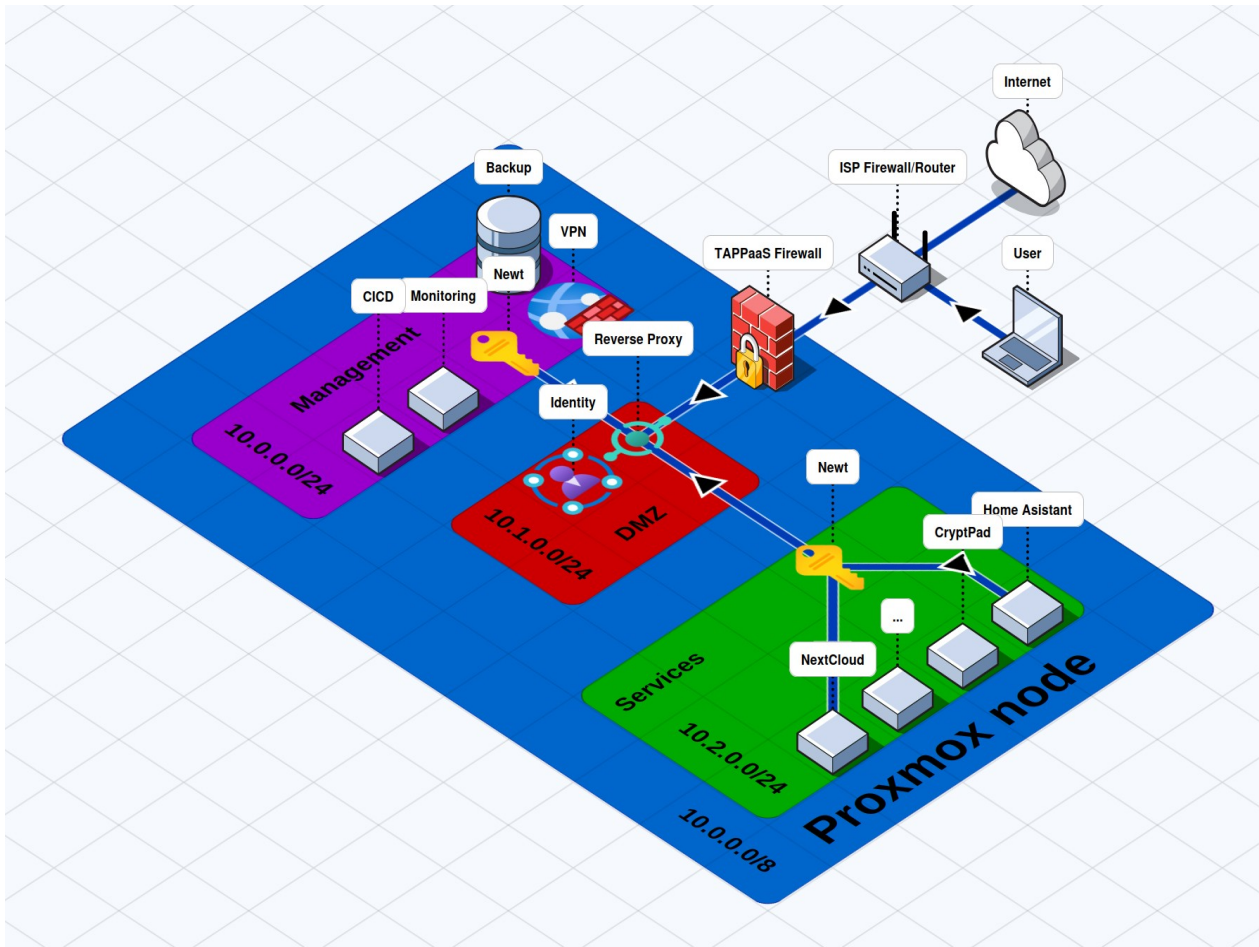
Prosumer:

- Family 'IT' person, who needs to manage the 'IT' needs at home to enable sharing personal data (photos, videos, documents) between other family members, relatives and friends
- Community 'IT' leader, who wants to manage the 'IT' needs for the community to enable collaboration and sharing of community data (photos, videos, documents) between community members
- (SMB) Business owner, who needs to (organize) the 'IT' needs for the business to enable collaboration on business data between employees, customers, and business partners \

Fellow Tweaker:

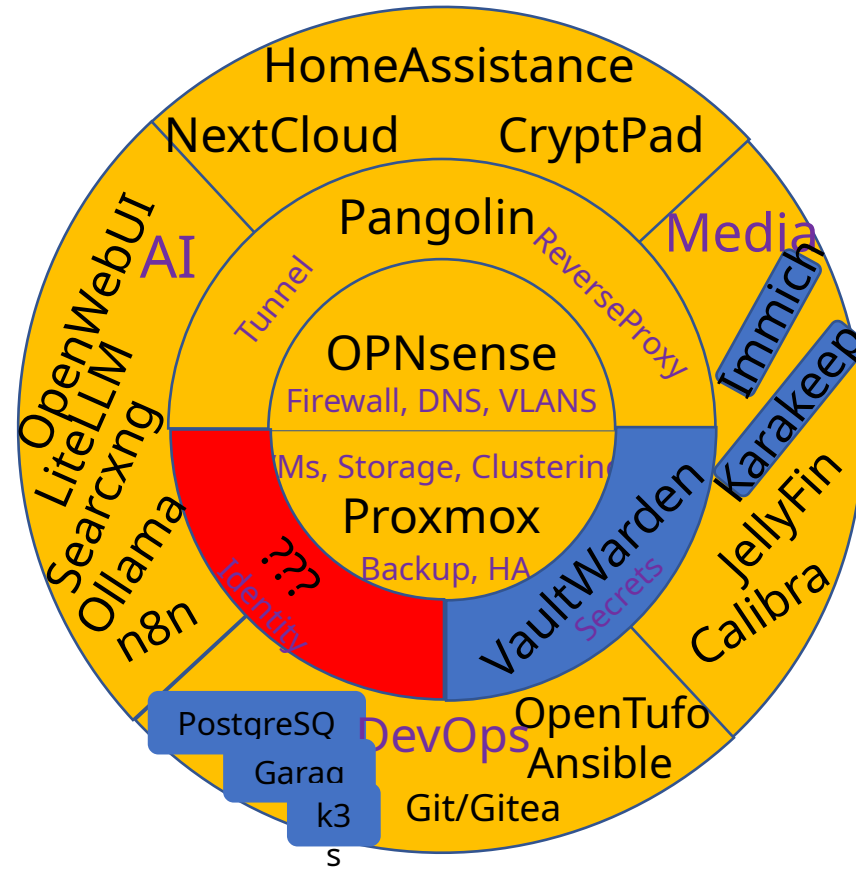
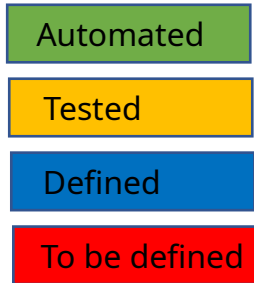
- who have the technical expertise needed to contribute
- Homelabbers who run their own homelab)
- Sysadmins / devops engineers at (SM)businesses - who turn work into hobby projects
- who recognize the im-balance between public- and private local cloud
- Who want to help restore this / provide an alternative
- who share the understanding that for this to work - at scale - ease of use may be the biggest inhibitor to adoption.

Structure of a solution



- Curate list of Software
 - Define capabilities
 - find matching SW.
 - Make selection
- Define configuration and integrations
- Test installation and integration of software
- Automate installation and maintenance

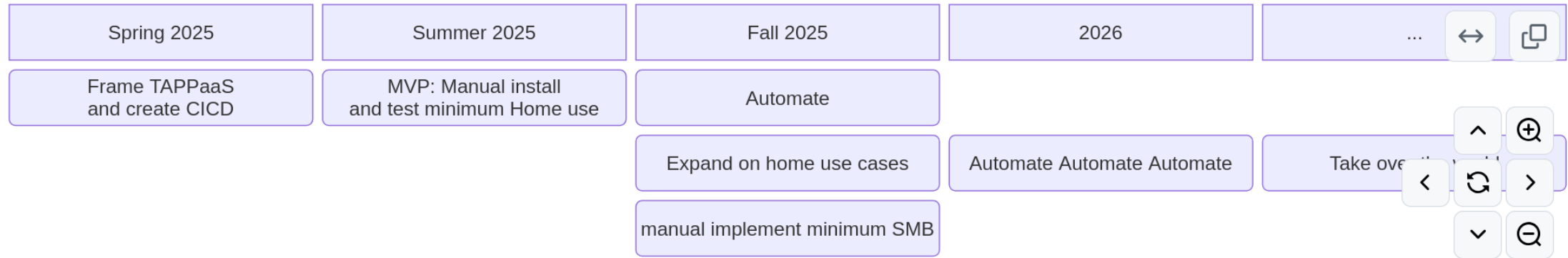
TAPPaaS: Whats on the dish



Eating the Elephant one bite at a time

(disclaimer: no elephants were harmed creating TAPPaaS)

Even though there is enough open source software out there to cater for most of the TAPPaaS needs then there is an almost endless amount of software modules and details to work out and integrate. For that reason we are following the following conceptual roadmap:



We are done with Framing, working hard on an MVP

Call to action / Plea for help

- GOTO:
 - tappaas.org
 - [github/tappass/tappaas](https://github.com/tappass/tappaas)

- Sanity check the approach
- Review selection of software
- Test system
- Write automation

Concrete Help needed:

- Issues with Pangolin locally
- Identity management
- Getting TAPPaaS.org to look cool
- Secrets management for CICD

Any Questions

