ProxWire

Connecting Proxmox VE nodes across the globe!Creating large-scaled Proxmox clusters for automated workloads.

A non-commercial project by gyptazy

Who's gyptazy?

Florian Paul Azim Hoberg

- Known as **gyptazy** in the tech community
- Working at **credativ GmbH** in Mönchengladbach, Germany
- Technical Lead / Sr. Consultant
- FOSS Contributor
- Loves FreeBSD

Garden Linux

Developer & Maintainer at the Garden Linux project. Mostly working on ARM64, SELinux and PyTest.

Ansible Modules

Developing & contributing powerful Ansible modules for infrastructure management.

ProxLB

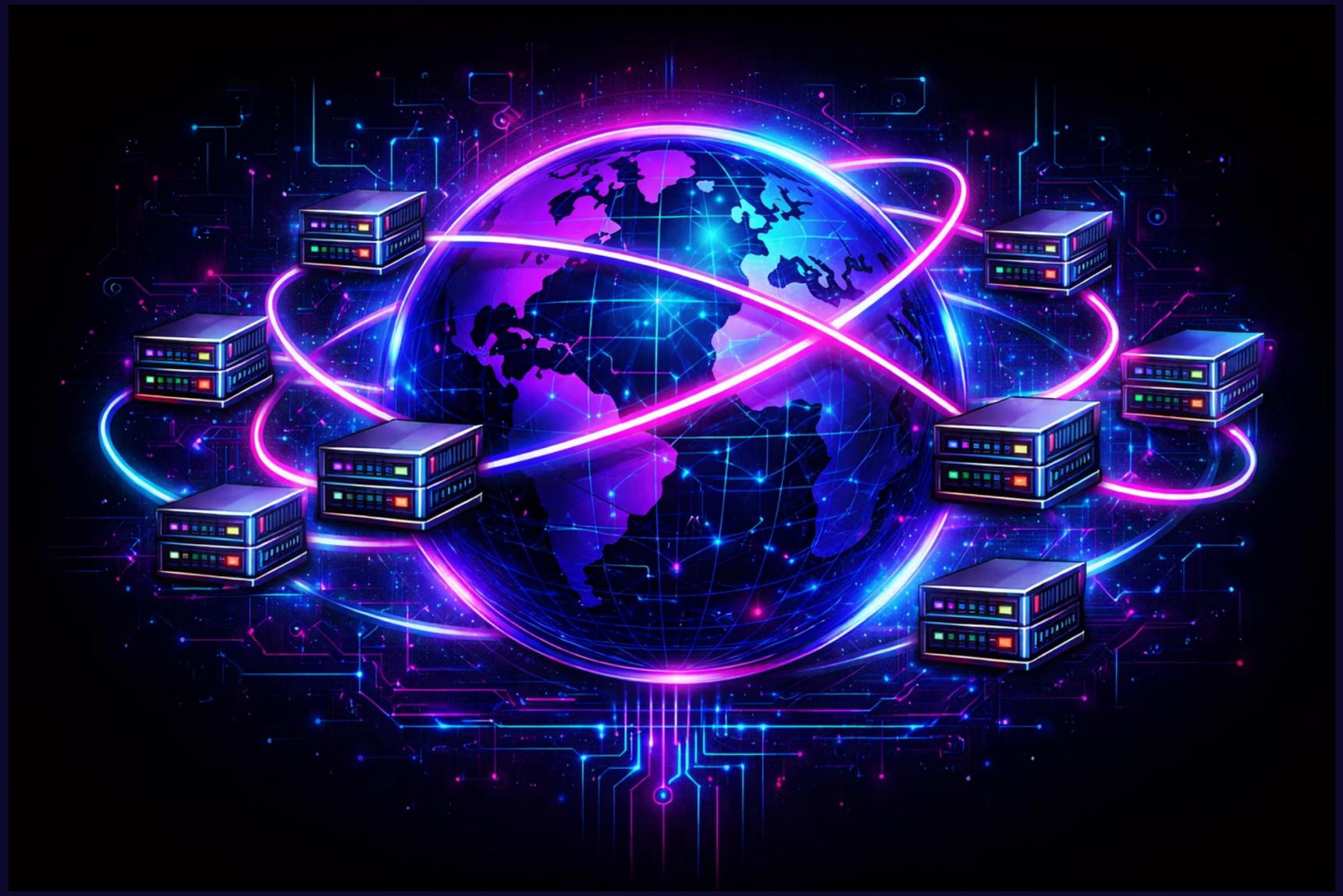
Founder & lead developer of this innovative load balancing solution for Proxmox environments.

BoxyBSD

Founder & lead developer of the BoxyBSD project. Providing free VPS instances to newcomers & beginners to learn and practice BSD based system.

What is ProxWire?

ProxWire is an underlying tool to create Proxmox VE based clusters across larger distances - even over continents!



But Why?

ProxWire addresses key challenges in traditional Proxmox deployments, offering a more flexible and robust solution for distributed infrastructure.

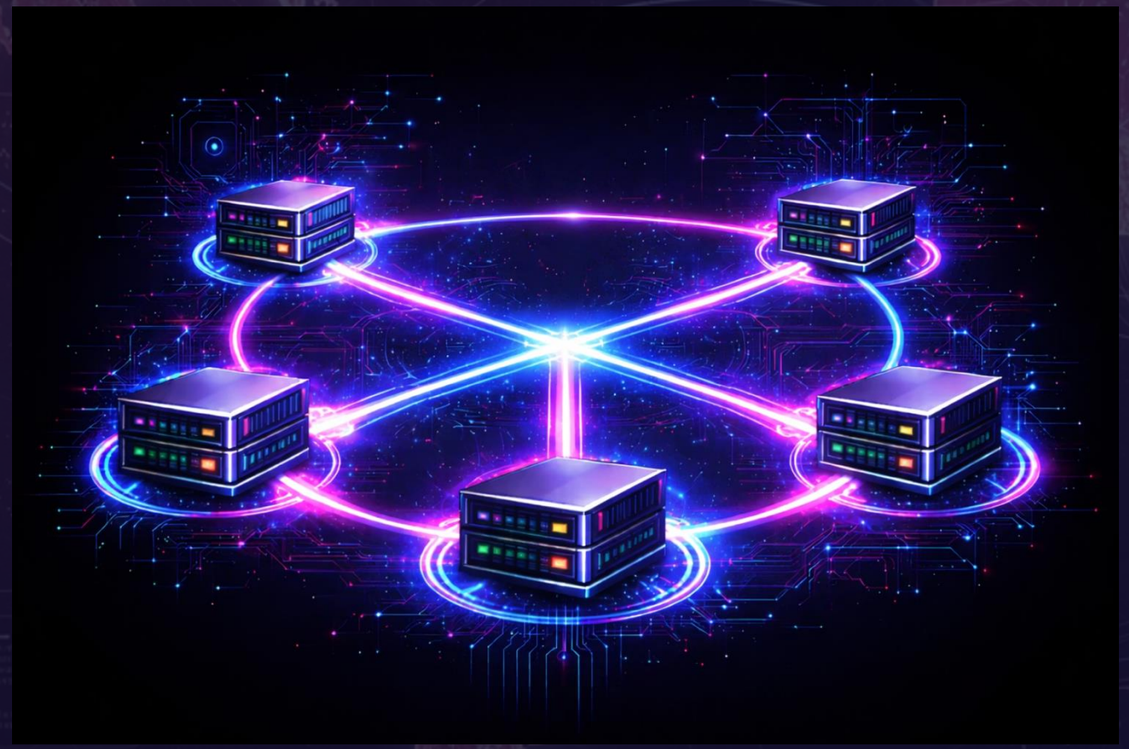
Breaking the Concept of On-site Clusters

- A Node is a Node
  - Just place fitting workloads
- No technical region awareness is required
  - You might still want to have an imaginary one for distribution of workloads

Skip Dealing with CoroSync

- Breaking the Multi-Master Setup
  - Using standalone Orchestrators (like vSphere, Orchestra)
- No bandwidth issues for "real time" sync
- Removing the latency barrier
  - Usually ~60 seconds until a migration starts, we can lower it!

What Changes Now?



What does ProxWire consist of ?

ProxWire is an underlying tool to create Proxmox VE based clusters across larger distances - even over continents!

Agent

A lightweight service deployed on each Proxmox VE node, responsible for secure communication and data collection within the ProxWire ecosystem.

Server Instance

The central component acting as an API endpoint, managing data aggregation and housing the core logic for cluster orchestration and operations.

Webinterface

A user-friendly management portal providing an intuitive graphical interface for monitoring, configuring, and controlling your entire distributed Proxmox environment.

Sounds good - but?

Storage!

Tackling Distributed Storage

Ceph

- RBD Mirroring is the solution
  - Journal or snapshot based
- Asynchronous (latency, quorum)
- High bandwidth over WAN / DF

ZFS

- Based on ZFS send/rec functions
- Asynchronous (latency)
- High bandwidth over WAN / DF

Asynchronous replication methods can lead to data divergence or loss in failure scenarios. Careful planning for consistency and recovery is crucial.

Distributed Storage: Conclusion

Not a solution for all scenarios

This approach isn't universally applicable; carefully assess your specific use case requirements and constraints.

Fits best for smaller instances

Optimal performance and resource management are typically achieved with smaller, more focused workloads and VMs.

Requires high performance networking

A robust, low-latency network infrastructure is absolutely critical to avoid bottlenecks and maintain data consistency.

Can still result in potential data loss

Asynchronous replication methods carry inherent risks; robust backup and disaster recovery strategies are essential.

Questions?

Thanks!

Author: Florian Paul Azim Hoberg @gyptazy

Web:	https://gyptazy.com
Email:	gyptazy@gyptazy.com (0xB1B88CBB)
Matrix:	@gyptazy:gyptazy.com
X:	@gyptazy
Fediverse:	@gyptazy@gyptazy.com