

# Managing and Running Multiple Kubernetes Clusters in Hybrid Setups



Simon Pearce – Systems Architect – SysEleven

Sebastian Scheele – CEO – Loodse

# Introduction



**Sebastian Scheele**  
CEO & Co-Founder

E-Mail: [sebastian@loodse.com](mailto:sebastian@loodse.com)  
Twitter: [@hrscheele](https://twitter.com/hrscheele)

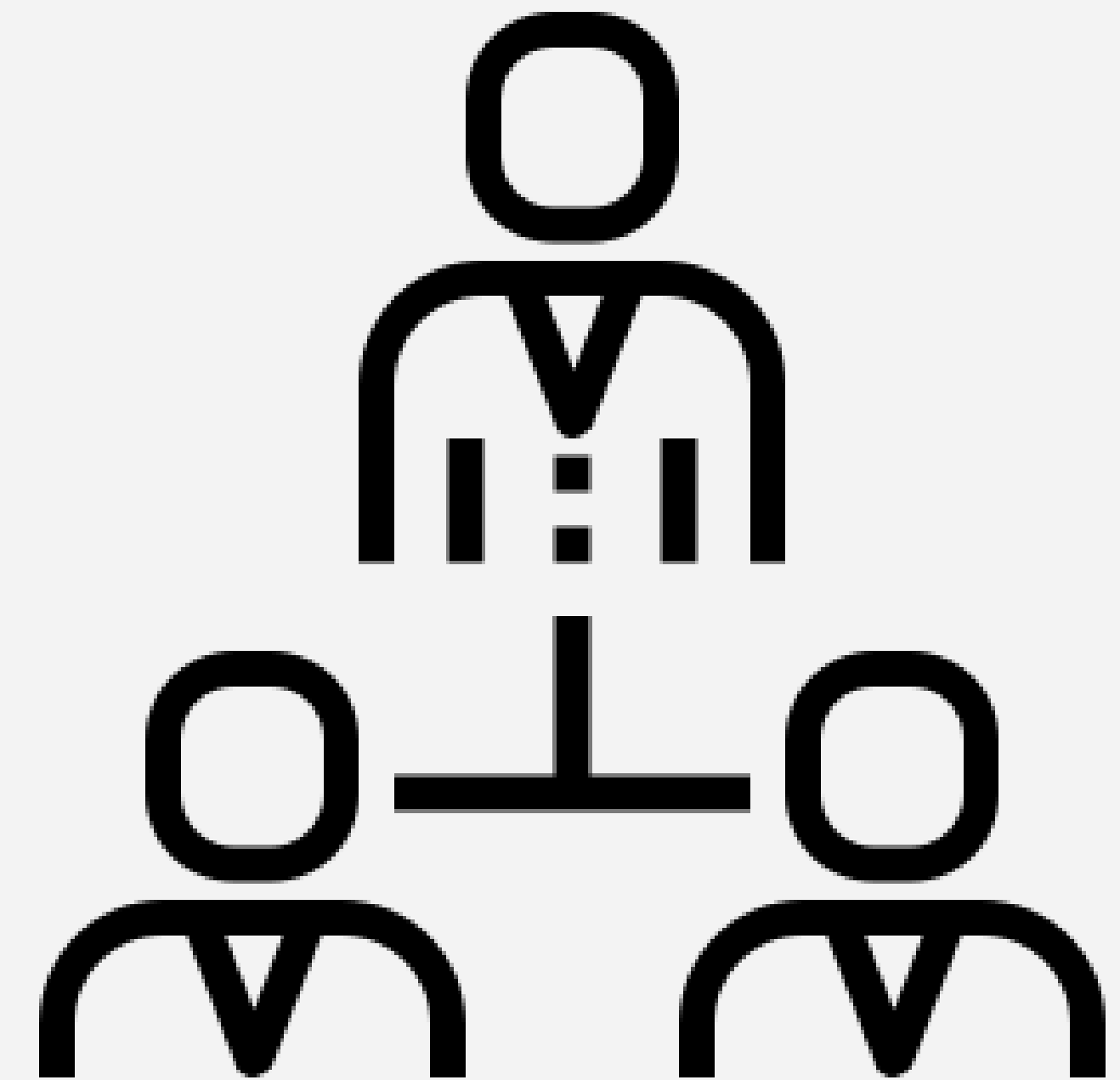


**Simon Pearce**  
Senior Systems Architect

E-Mail: [s.pearce@syseleven.de](mailto:s.pearce@syseleven.de)

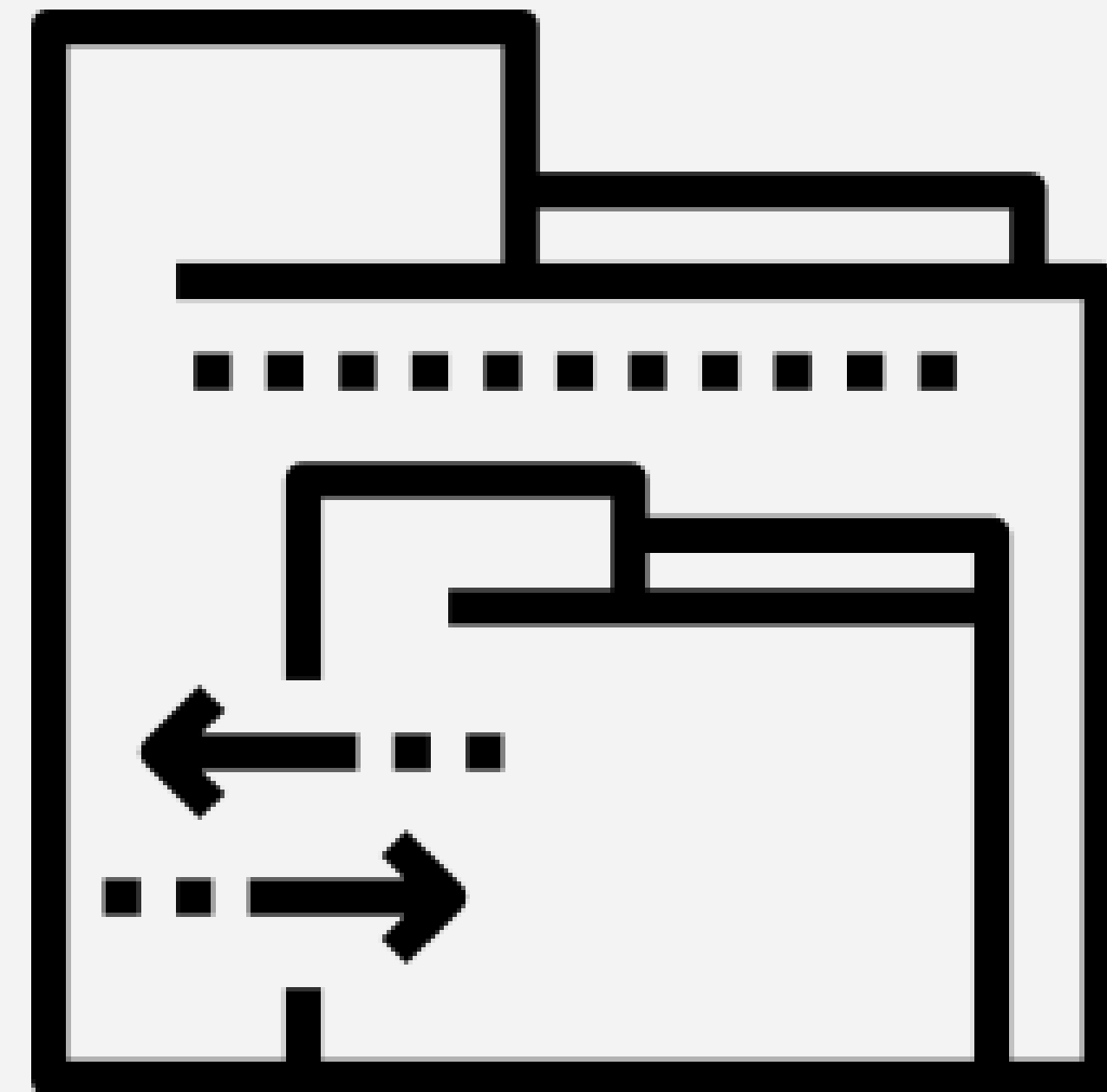
# The Challenge

- Management interface for multiple clusters
- Multi-tenancy
- Master components as managed service
- Administration interface
- Choice of add-ons/CNI



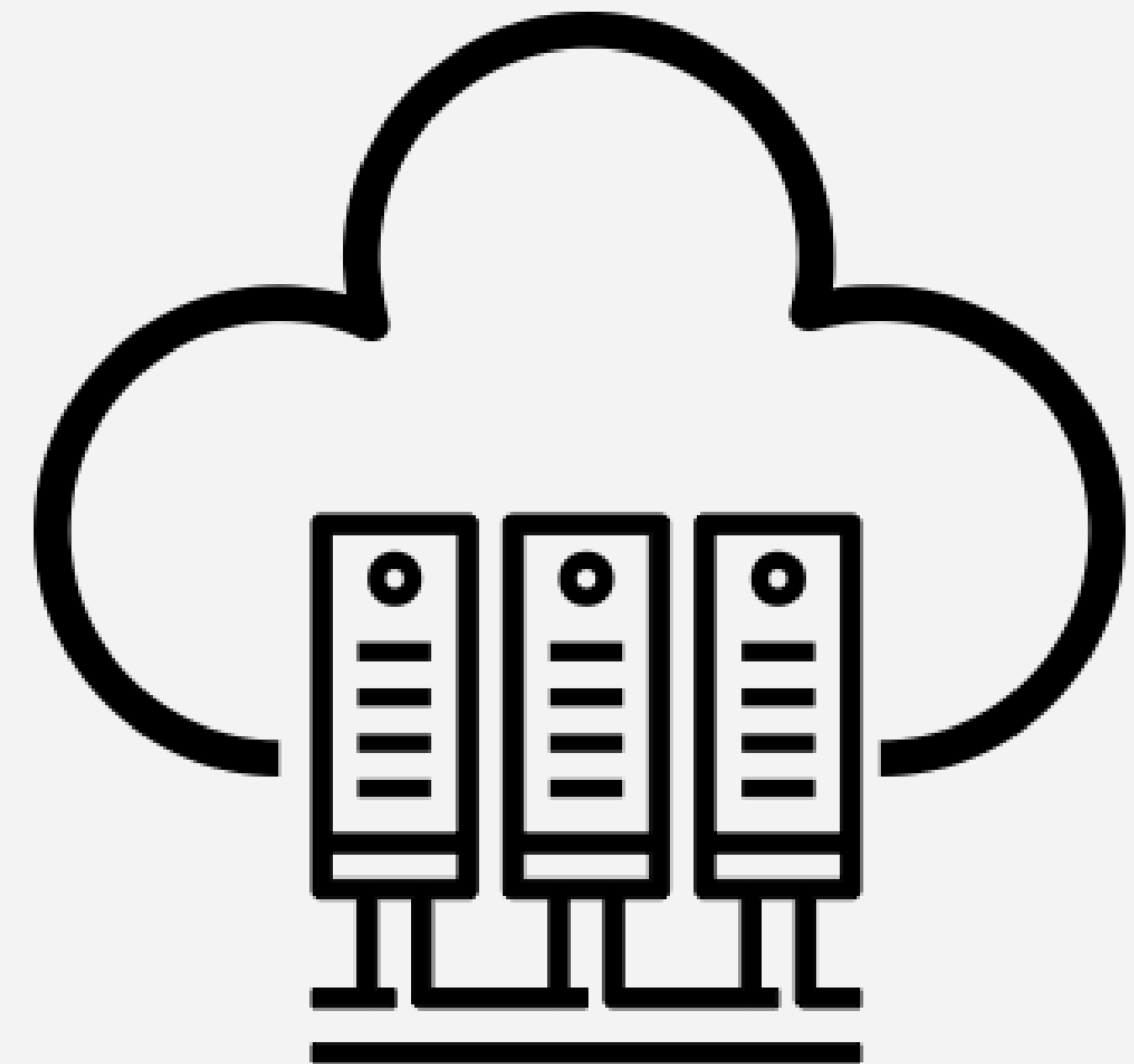
# The Challenge

- k8s master as container
- Single service endpoint for multiple clusters
- Upgrade multiple clusters
- User/role management
- Installation of helm charts/monocular



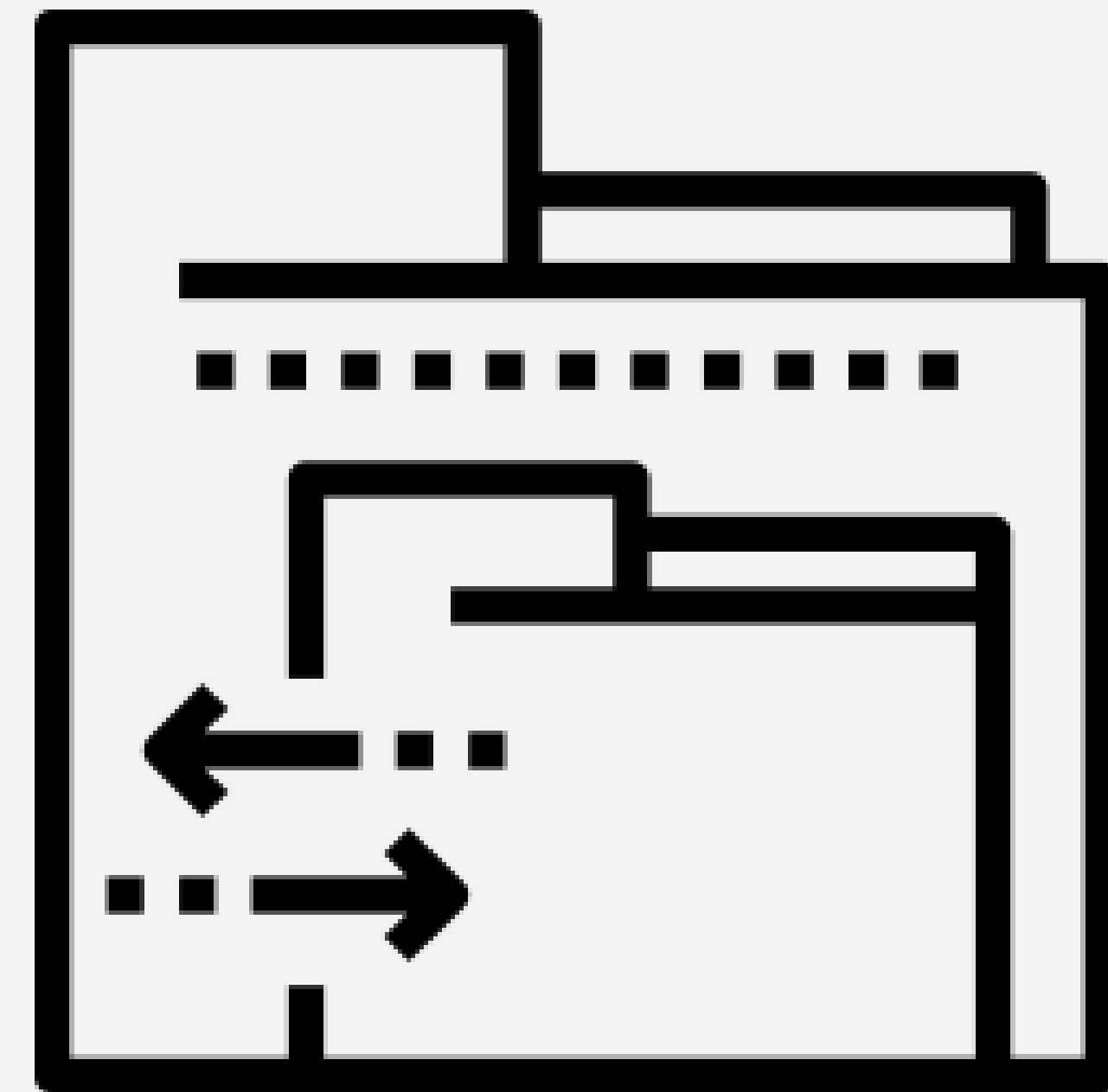
# The Challenge

- Existing tools focus on deploying a single cluster
- Access to k8s master
- HA requires a minimum of 3 VM's per cluster
- Additional tooling is required

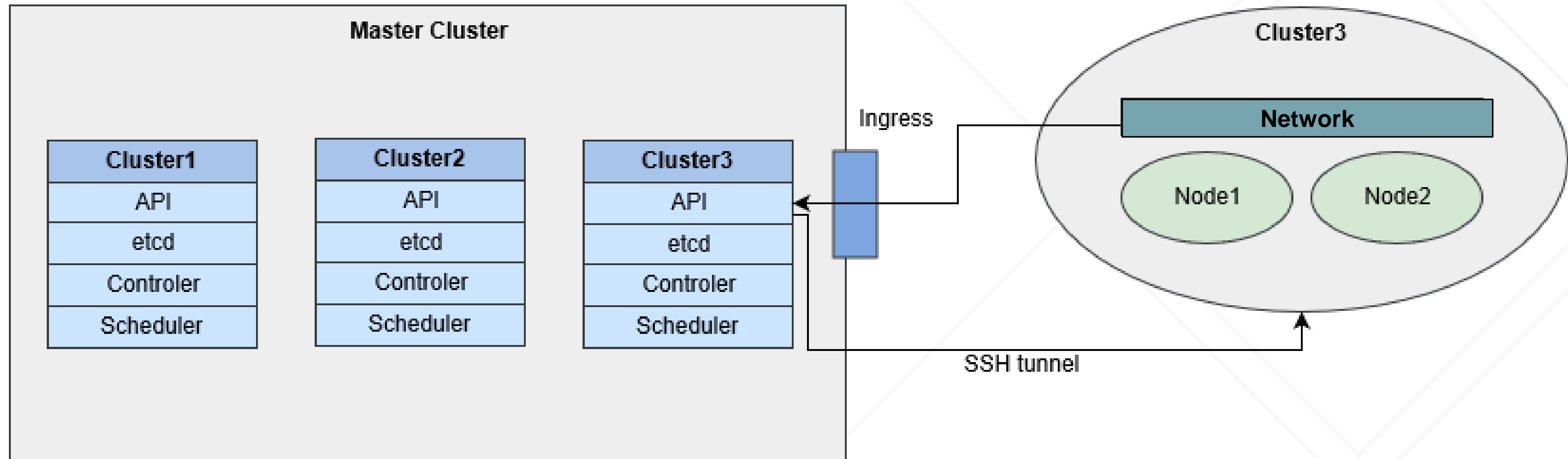


# Google-Like Container Engine

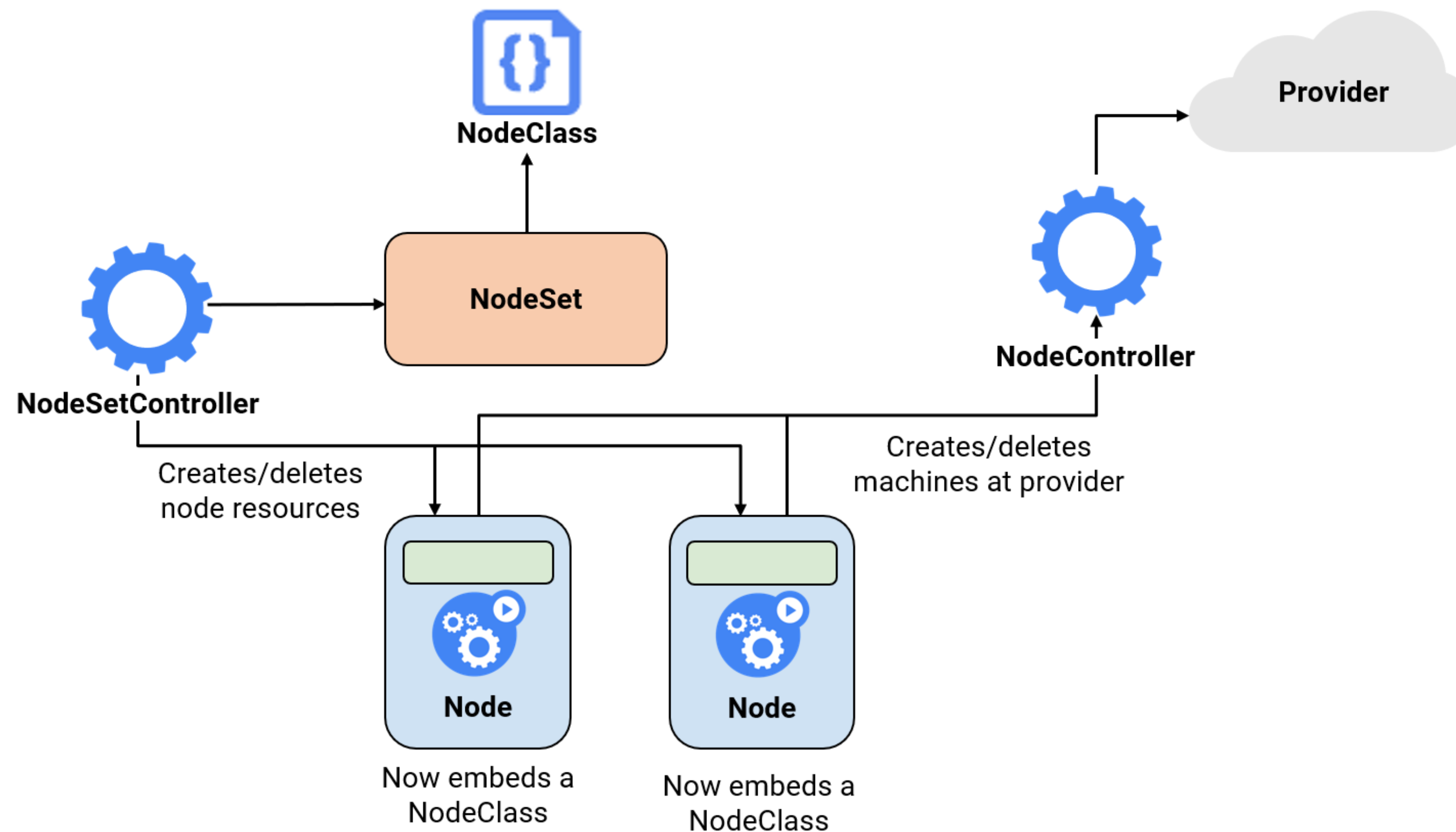
- Self-service for developers
- Update of clusters
- Install cluster add-ons cni, helm charts, dashboard
- Add and remove worker nodes
- Autoscale worker nodes
- Configure external load balancer via ingress rules
- Automatic backup and recovery for k8s master



# Google-Like Container Engine



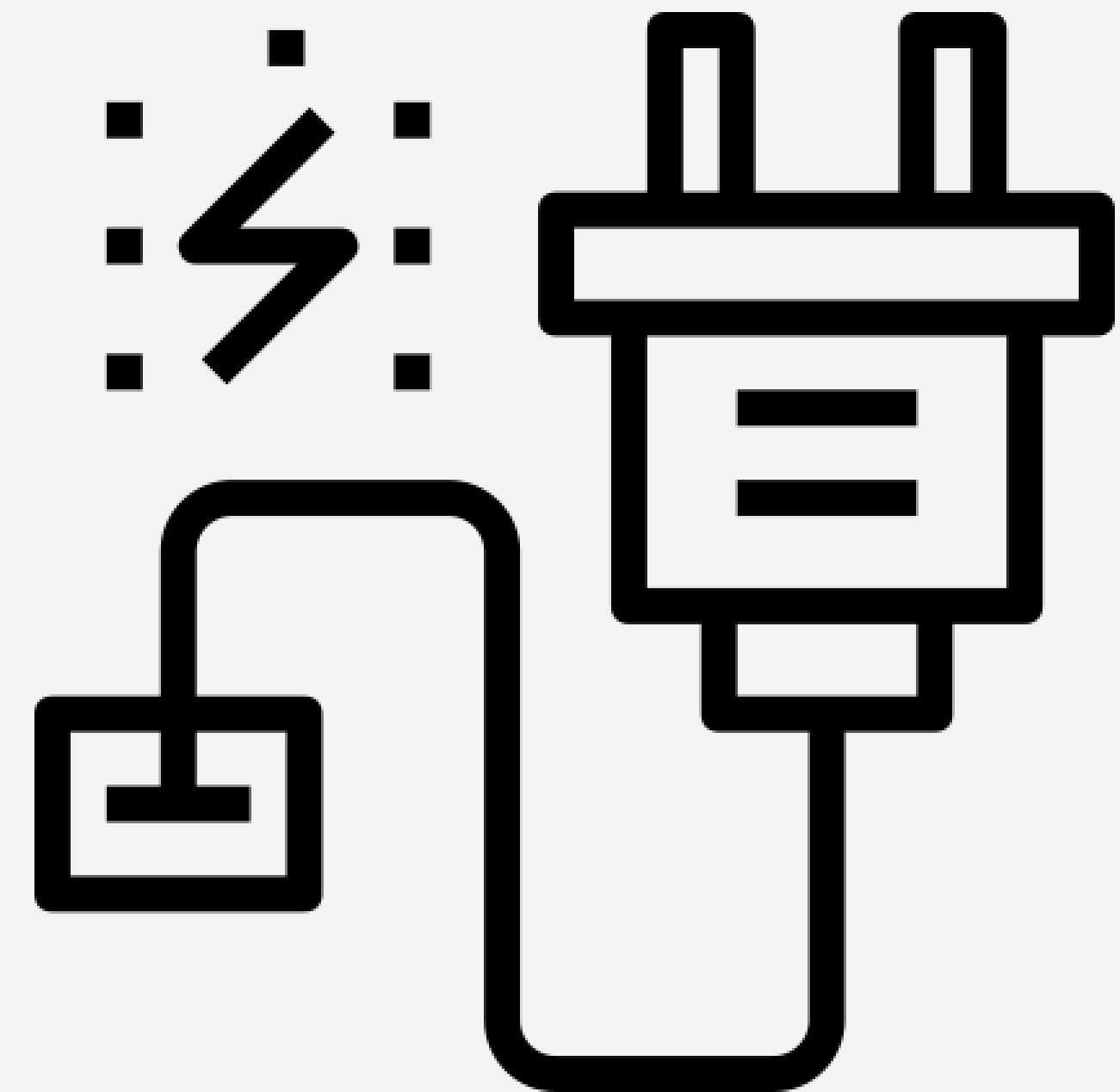
# Google-Like Container Engine





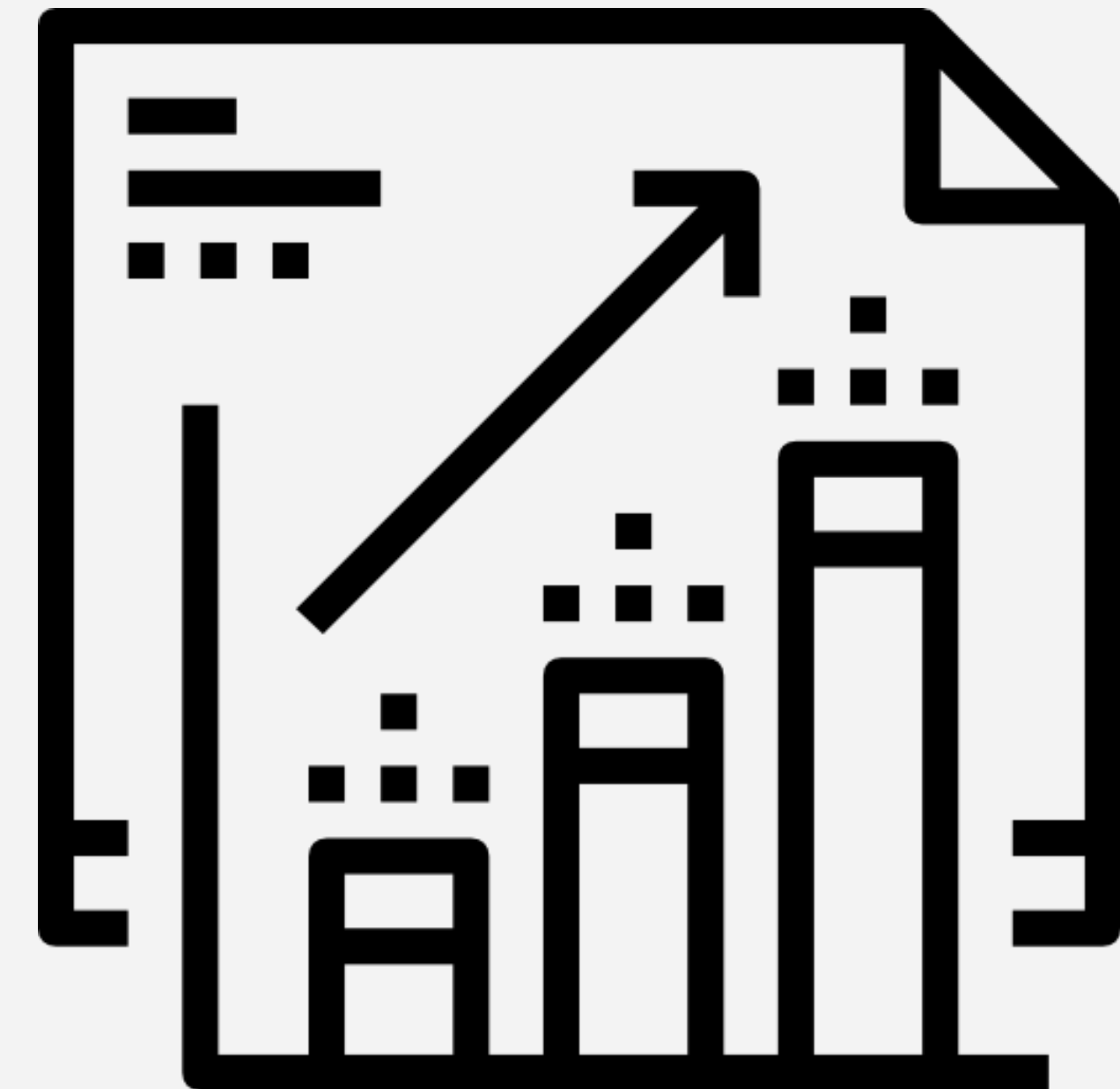
# Authorization

- Support different identity provider
- Seamless management- & cluster-login
- RBAC & network policy



# Multi Provider Support

- Same setup for k8s master
- Only deploy VM, Docker and kubelet
- Same team can operate all setups



# Hybrid Setup

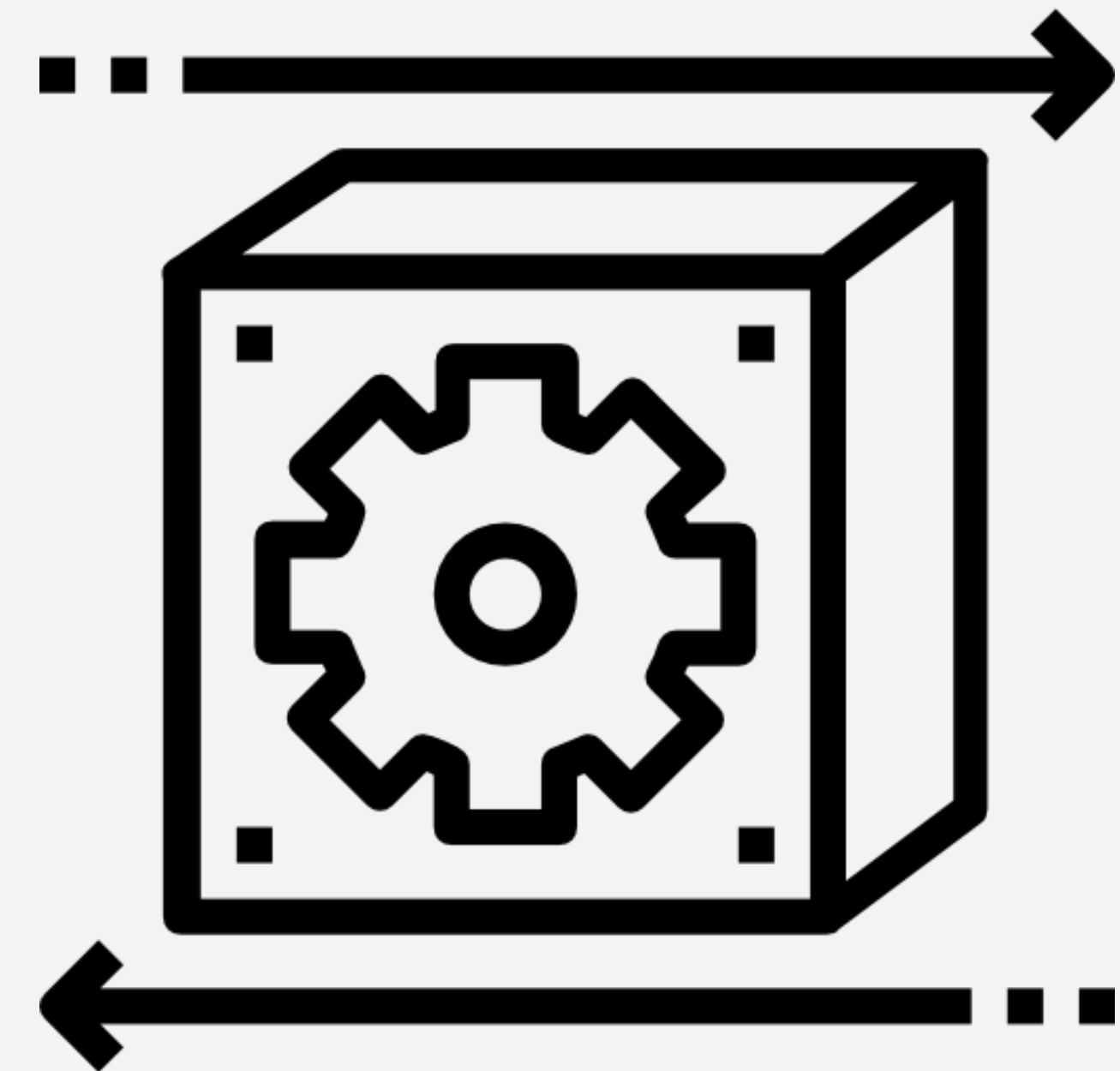
- OpenStack cloud
- Two distributed cloud regions
- Bare metal - Isolated environment
- Two storage zones
- Integration of additional data centers or cloud providers possible



# Live Demo

# Lessons Learned

- K8s-clusters are not enough
- Use a different API port per cluster
- Do not limit etcd memory resources
- Take regular etcd snapshots on persistent storage
- Restrict access to master components



# Roadmap



- Improve cluster authentication / user management
- Worker node autoscaling
- Support for different linux distributions
- Configure external load balancer (cloudflare) via ingress rules
- Standard way for CI/CD
- Automate cluster upgrade process



Any questions?



Thank you for listening!