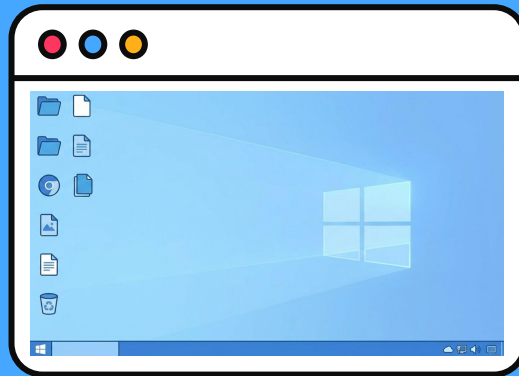
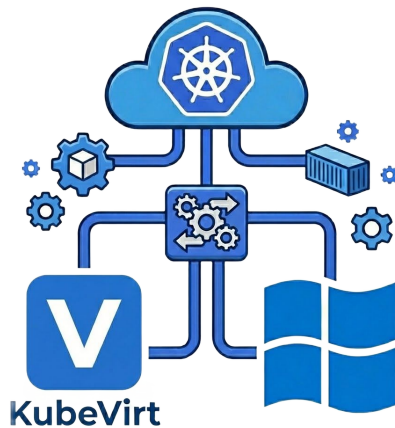


Ctrl + Alt + Deploy



KUBERNATIC



ODIT.Services

Windows Workloads as a Service



KubeCon



CloudNativeCon

Europe 2026

KUBERNATIC ODIT.Services



Nicolai Ort
Platform Engineer
ODIT.Services

@nicolaiort



Tobias Schneck
Principal Architect
Kubermatic

@toschneck



Windows?!?

Didn't we agree on all things cloud native?





Specialized Software



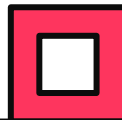
Microsoft
Active Directory



SketchUp

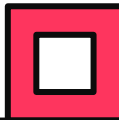
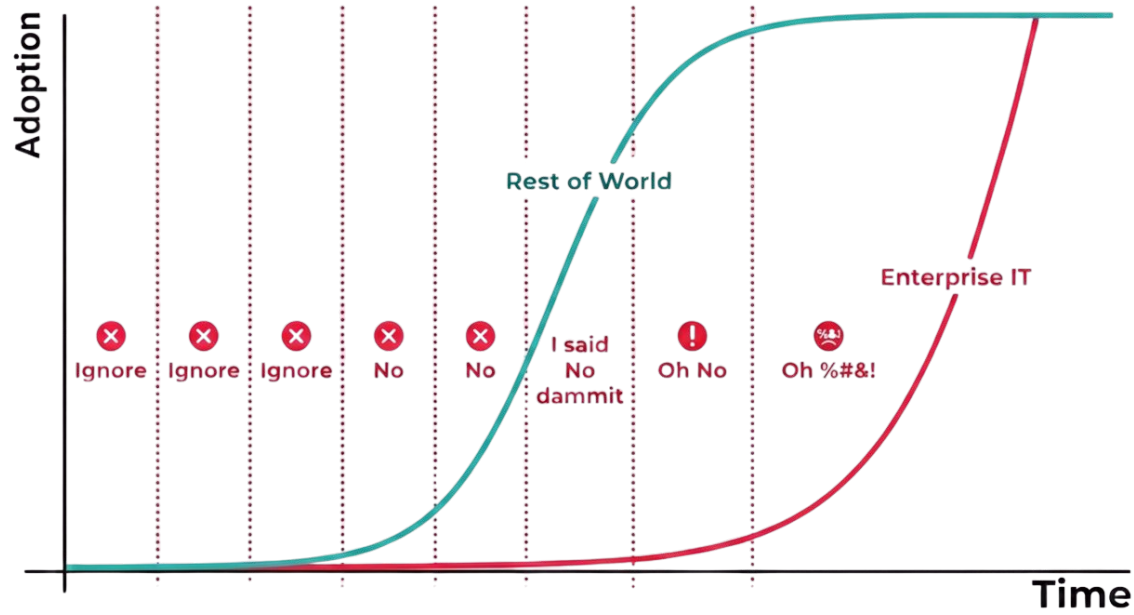


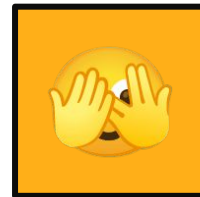
Microsoft®
SQL Server™*





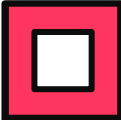
A Look at Cloud Adoption



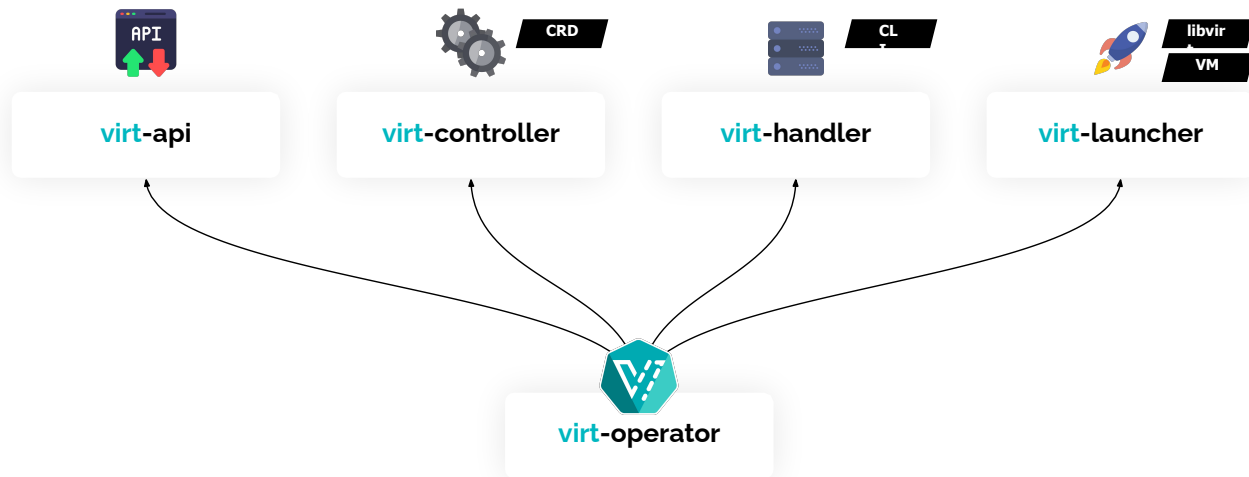


VMs on Kubernetes?

Sounds like heresy!

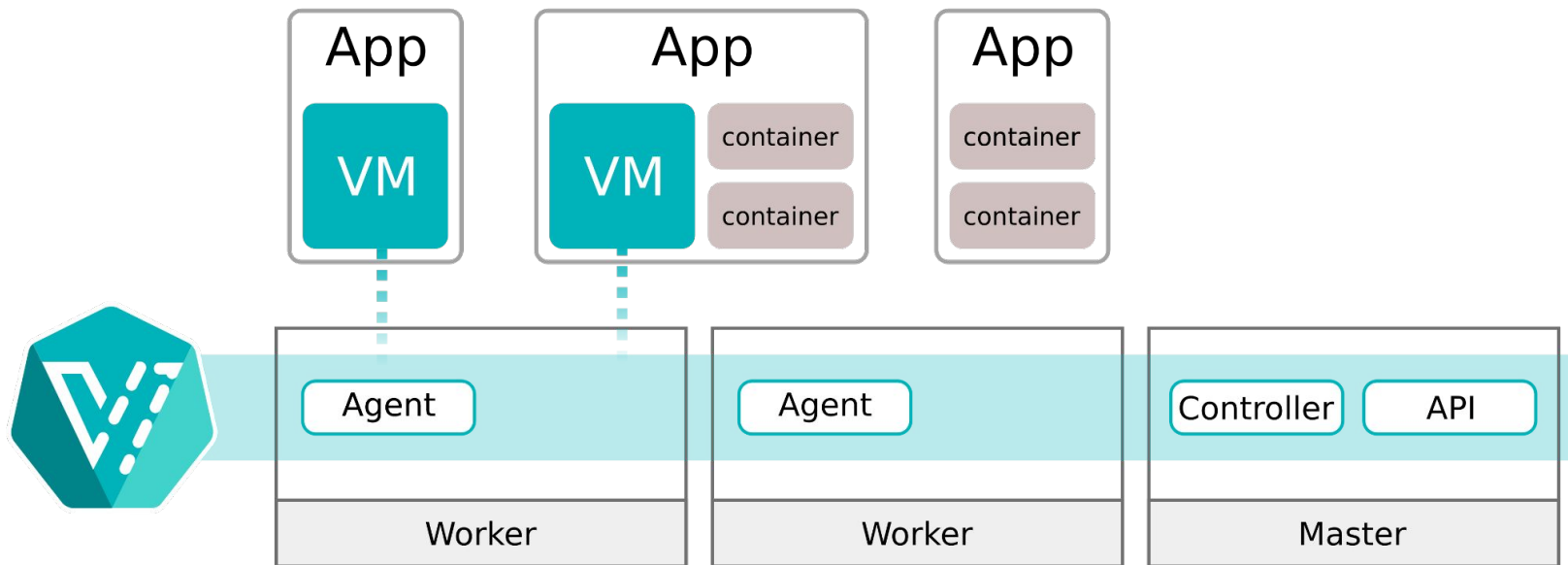


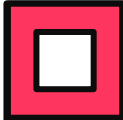
What is KubeVirt?



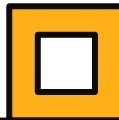
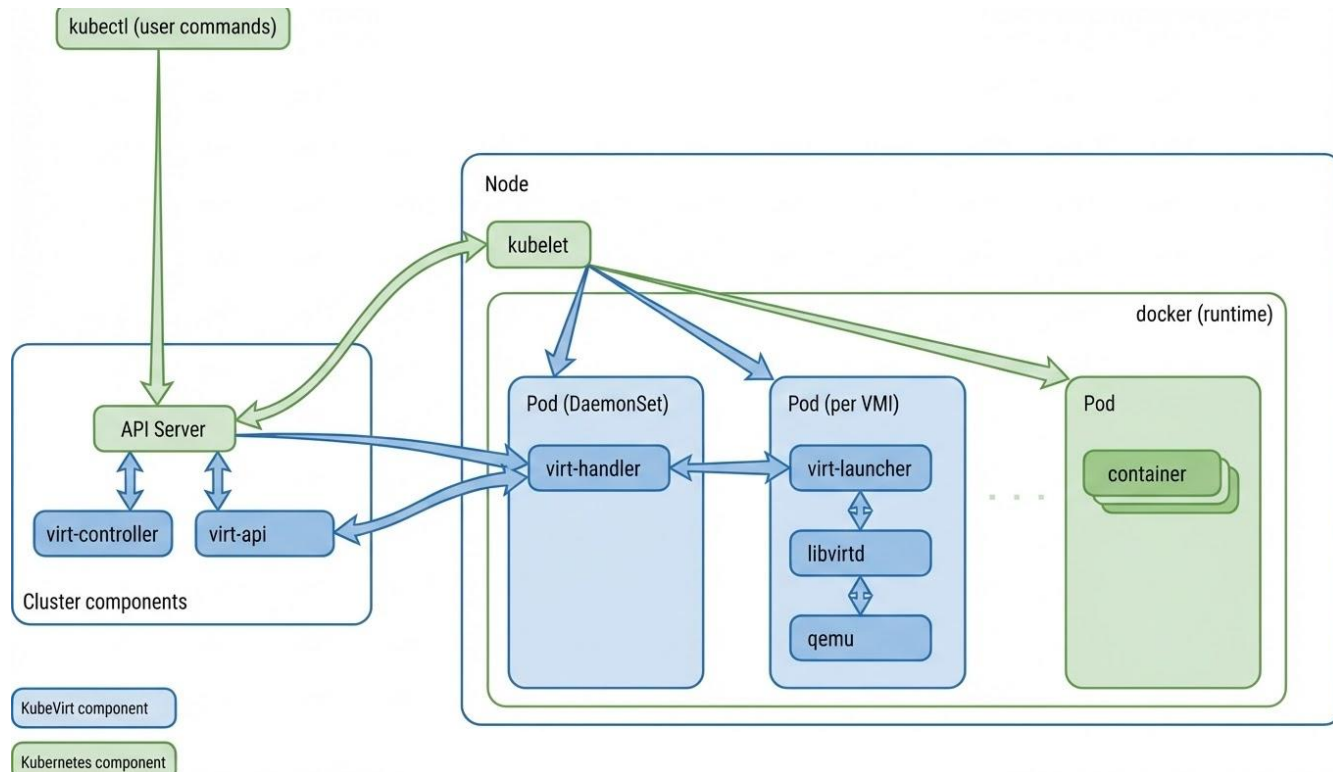


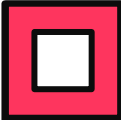
Architecture





Architecture

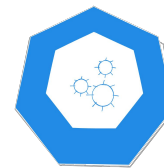
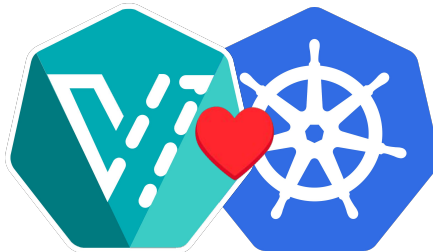




Profit from Kubernetes-Native Features



Ingress & TLS



Operators &
GitOps (Declarative VM)



Seamless
Load Balancing



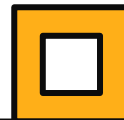
Flexible Scheduling
& (Anti-) Affinity



Native & CNI
Network Policies



Mount Secrets
& Configmaps





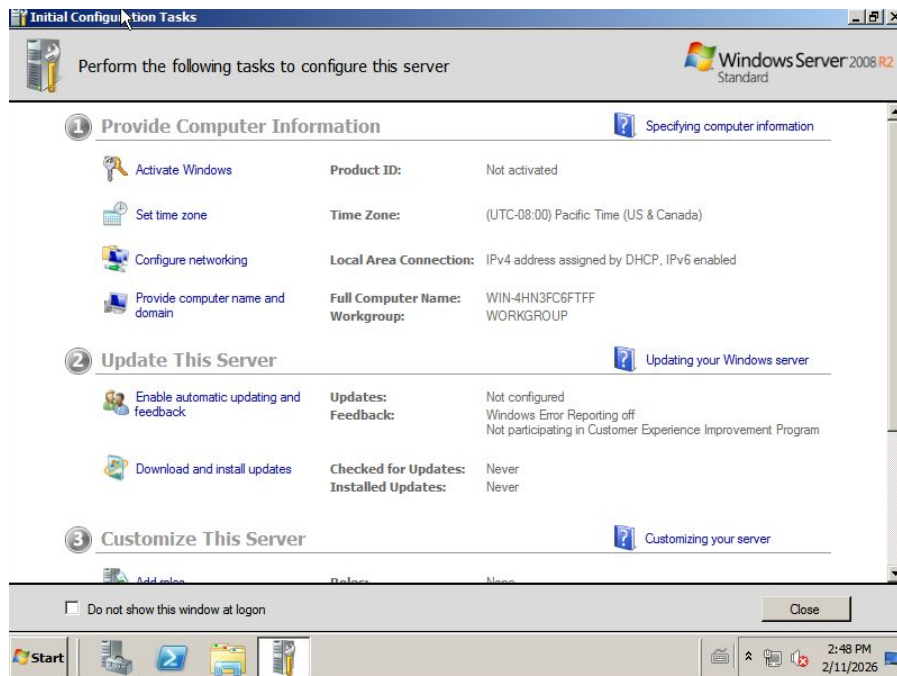
Scenarios

What did we test for you?

Just running Windows (Server)

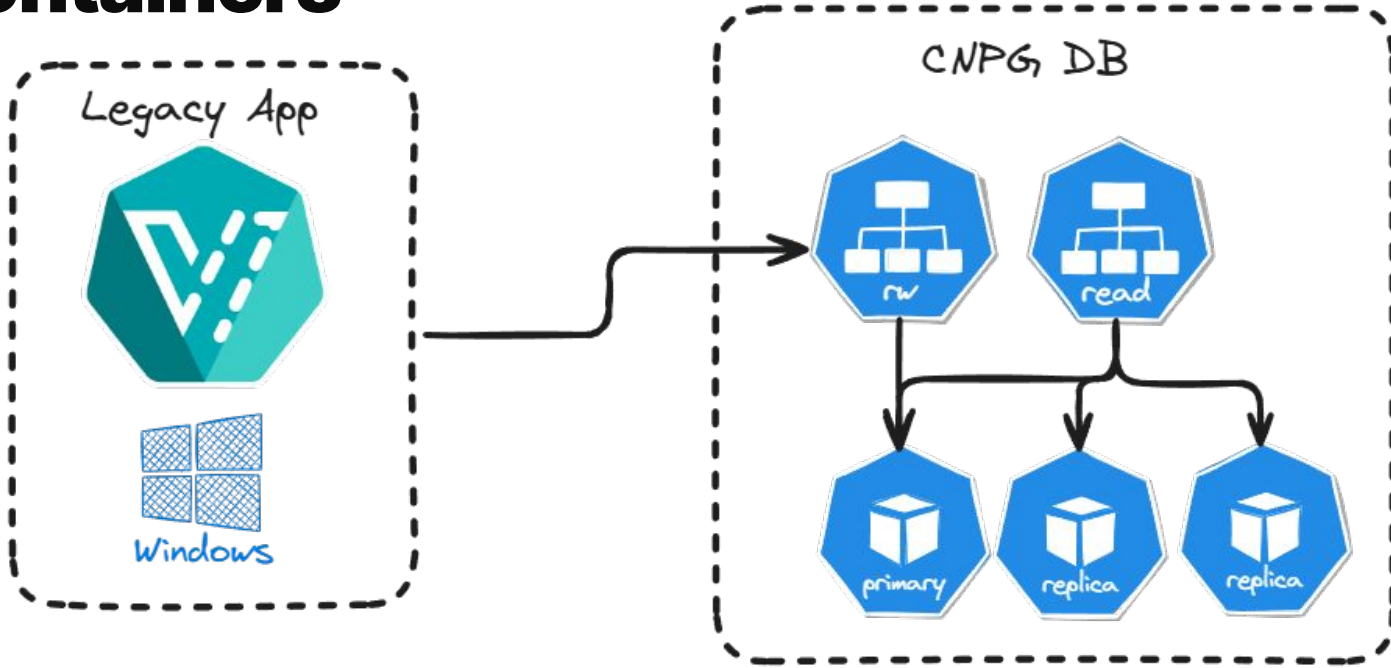
Including but not limited to:

- Windows XP 
- Windows 10 
- Windows 11 
- Windows Server 2008 
- Windows Server 2016  
- Windows Server 2019  
- Windows Server 2022  
- Windows Server 2025 



(Based on testing an default VirtualMachineClusterPreference)

Legacy on VMs, dependencies in Containers



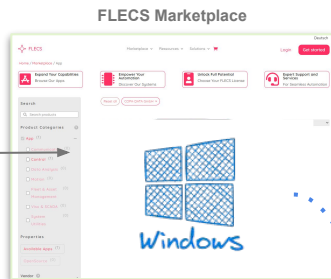
Legacy Windows VM as a Service for Industrial Edge



Software
Defined
Automation

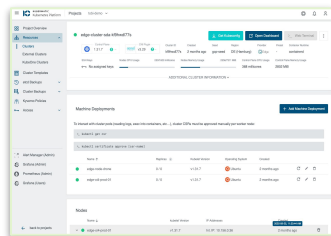


Automation
Engineer
selects App to
deploy from
Marketplace



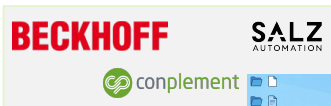
Marketplace calls
Kubernetes API
to deploy App

Kubermatic Kubernetes Platform

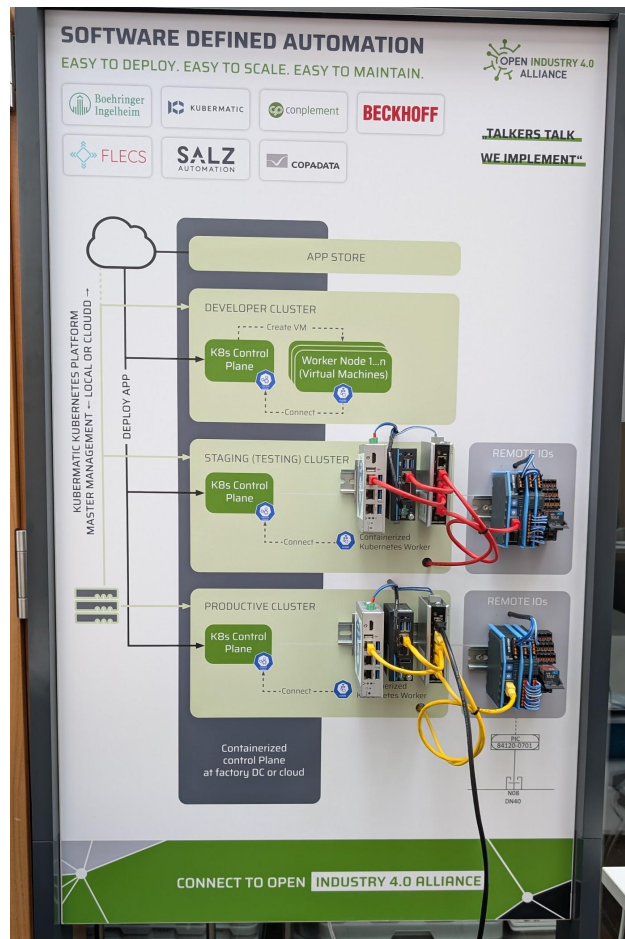


K8s Admin
manages
K8s
Clusters

Edge Device or VM



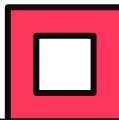
Device Admin
connect to K8s
Clusters





Why Windows as a Service

- No Tickets -> Order predefined Service via Service Store
- Pre-Defined Services
- Testable Provisioning
- Alternative to “hey Joe give me an VM”
- Multitenancy Abstraction
- Cloud Services Style On-Prem
 - Legacy side-by-side by modern Services
 - Lift and Shift Migration Service



MAY THE DEMO GODS



BE WITH US

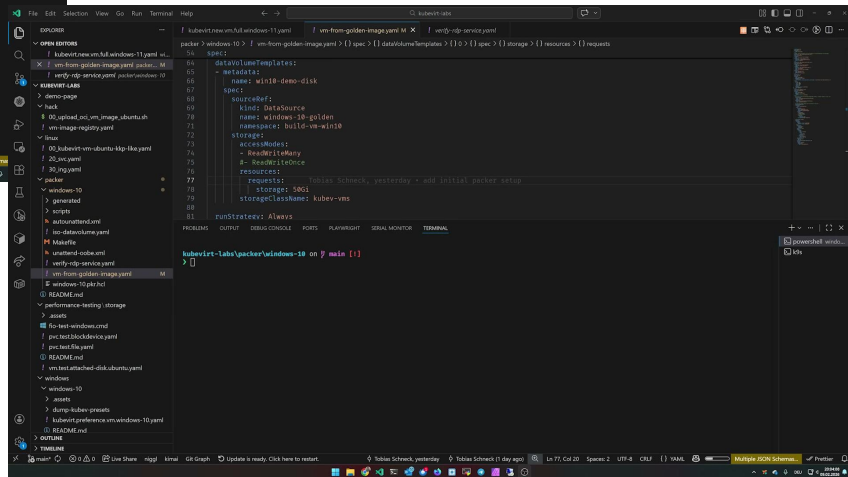


Demo time

Please pray for us



Automation and Presets



Demo: Install - Our resources

The manual way

```
apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    kubevirt.io/vm: vm-windows-10
  name: vm-windows-10
  namespace: vm-windows-10
spec:
  dataVolumeTemplates: <1 item: yaml:anything;
  #####

  runStrategy: Always
  template:
    metadata: yaml:anything
    spec:
      architecture: amd64
      dnsConfig:
        nameservers:
          - 8.8.8.8
      dnsPolicy: None
      domain:
        # --- CPU / Memory ---
        cpu:
          cores: 2 # Example value, adjust as needed
        resources:
          requests:
            memory: "4Gi" # Example value
          limits:
            memory: "4Gi" # Example value

# --- Features (Crucial for Windows) ---
features:
  acpi: {} # Standard Power Management
  apic: {} # Interrupt Controller
  smm: {}
  # Hyper-V enlightenments significantly improve performance
  hyperv:
    relaxed: {}
    vpic: {}
    vpxindex: {}
    spinlocks:
      spinlocks: 8191
    synic: {}
    synictimer:
      direct: {}
      tlbflush: {}
    frequencies: {}
    reenlightenment: {}
    ipi: {}
    runtime: {}
    reset: {}
    # vendorid: # Sometimes needed, e.g., vendorid: 'kubevirt'

# --- Clock (align with Hyper-V) ---
clock:
  utc: {}
```

Automation and Presets

```
apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  name: win10-golden
  namespace: win10-golden
  labels:
    kubevirt.io/vm: win10-golden
spec:
  # Instance type and preference manage CPU, memory,
 instancetype:
    kind: VirtualMachineClusterInstancetype
    name: u1.large
  preference:
    kind: VirtualMachineClusterPreference
    name: windows.10.virtio
```



Demo: Install - Our resources

The manual way

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  # Empty Disk
  name: vm-windows-10-disk
  namespace: vm-windows-10
spec: PersistentVolumeClaimSpec
---
apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  labels:
    kubevirt.io/vm: vm-windows-10
  name: vm-windows-10
  namespace: vm-windows-10
spec:
  dataVolumeTemplates:
  - metadata:
      creationTimestamp: null
      name: vm-windows-10-iso
    spec:
      source:
        http:
          url: http://vm-image-registry.kube-system.svc/vms/win10-2026-02.iso
      storage:
        accessModes:
          - ReadWriteOnce
        resources:
          requests:
            storage: 15Gi
        storageClassName: kubev-vms
```

Automation and Presets

```
apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  name: win10-golden
  namespace: win10-golden
  labels:
    kubevirt.io/vm: win10-golden
spec:
  # Instance type and preference manage CPU, memory, firmware, Hyper-V, clock, e
 instancetype: yaml:anything
  preference: yaml:anything

  # Clone the golden image into a new disk for this VM
  dataVolumeTemplates:
  - metadata:
      name: win10-golden-disk
    spec:
      sourceRef:
        kind: DataSource
        name: windows-10-golden
        namespace: build-vm-win10
      storage:
        accessModes:
          - ReadWriteMany
        resources:
          requests:
            storage: 50Gi
        storageClassName: kubev-vms
```





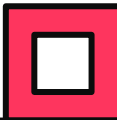
Demo: Install

The manual way

Automation and Presets

```
18 spec:
19   - name: windows-11
20     kind: VirtualMachine
21     metadata:
22       name: windows-11-demo
23     template:
24       spec:
25         domain:
26           clock: (align with Hyper-V) ---
27           timer:
28             type:
29               preset: false
```

```
55 spec:
56   - name: windows-11
57     kind: VirtualMachine
58     metadata:
59       name: windows-11-demo
60     template:
61       spec:
62         domain:
63           clock: (align with Hyper-V) ---
64           timer:
65             type:
66               preset: false
```





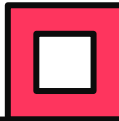
Demo: Install

The manual way

Automation and Presets

```
18 spec:
19   platform: Black
20   ...
21   osVersion: kubermatic-11/v1
22   kind: VirtualMachine
23   metadata:
24     labels:
25       special: vm-windows-11-demo
26   runStrategy: Always
27   template:
28     spec:
29       domain:
30         clock: (align with Hyper-V) ---
31         timer:
32           type:
33             preset: false
```

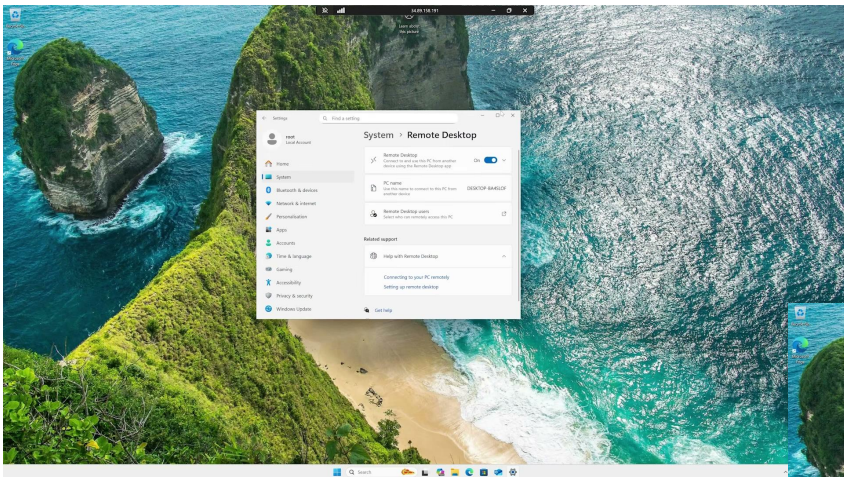
```
55 spec:
56   dataVolumeTemplates:
57     - metadata:
58       name: windows-11-golden
59       namespace: kubermatic-vm
60     spec:
61       sourceRef:
62         kind: DataSource
63         name: windows-11-golden
64         namespace: kubermatic-vm
65       storage:
66         requests:
67           storage: 50Gi
68           storageClassName: kubermatic-vm
```





Demo: Install

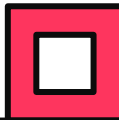
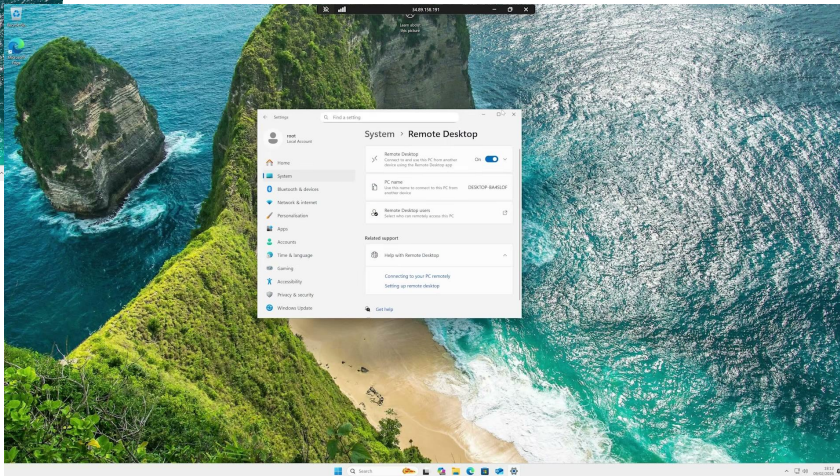
The manual way



Realtime: 27min (attended)

Automation and Presets

Realtime: 7min (unattended)

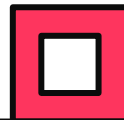




FIO

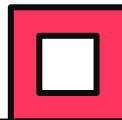
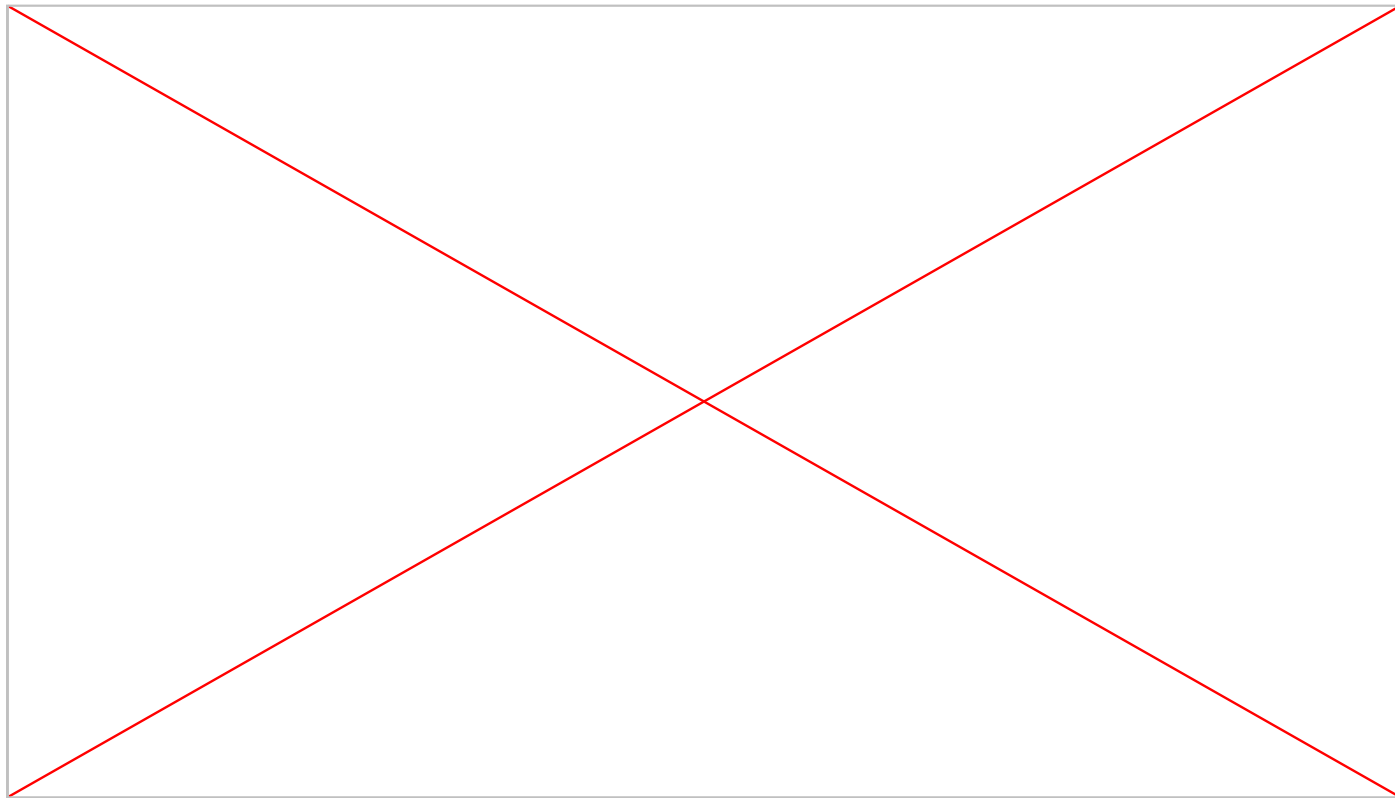
(nested Virtualization Test Environment with Rook Ceph on GCP)

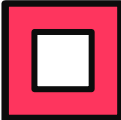
OS	Drivers	Bus	IOPS	Speed
Ubuntu (Pod)	CSI	CSI	9629	37.6MiB/s
Ubuntu 24.04 (VM)	✓	SCSI	9749	38.1MiB/s
Windows 10 (VM)	✗	SATA	3567	13.9MiB/s
Windows 10 (VM)	✓	SCSI	9964	38.9MiB/s



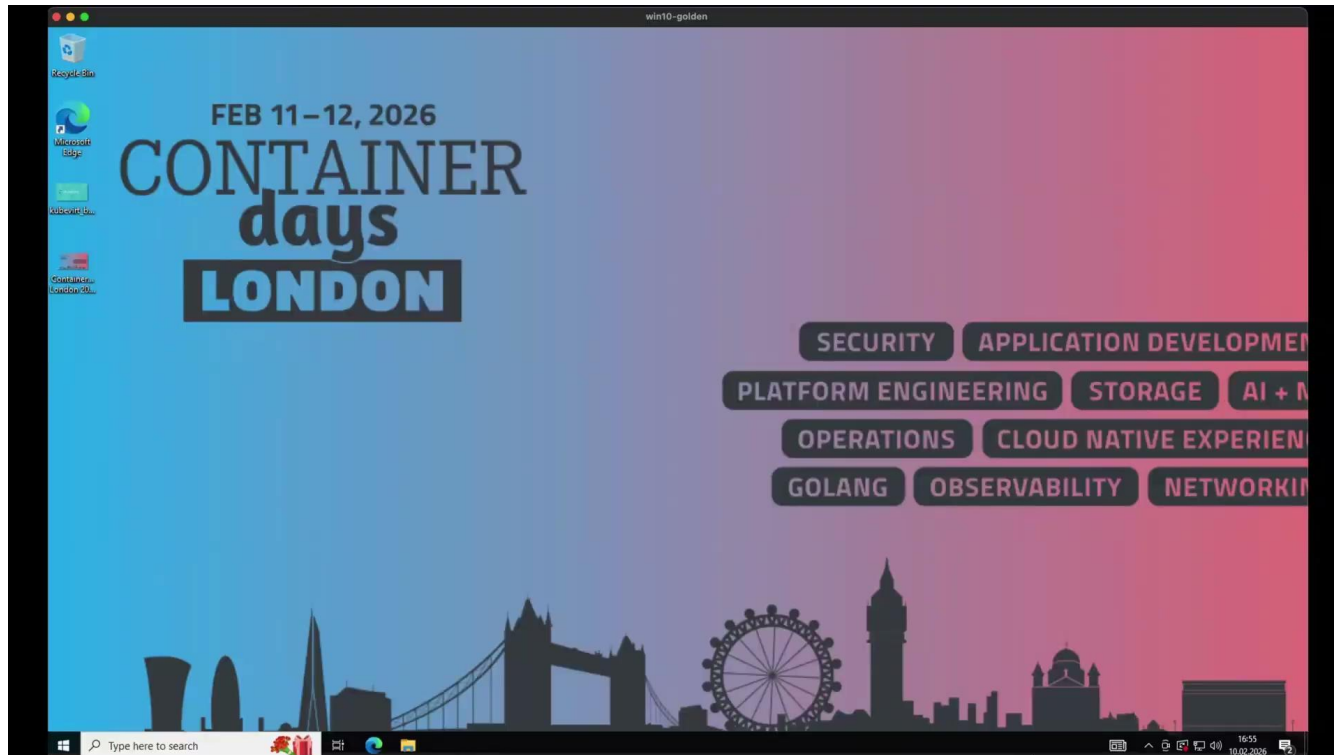


Demo: Live Migration

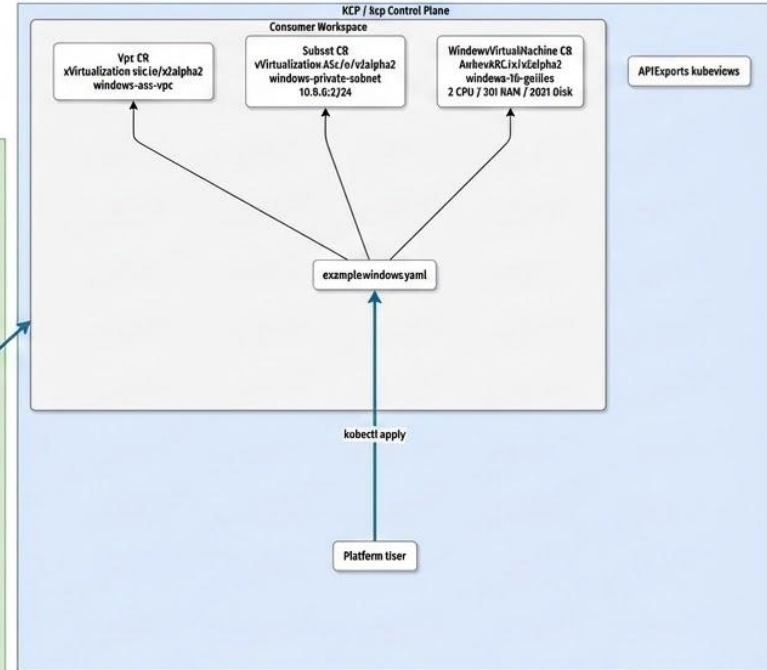
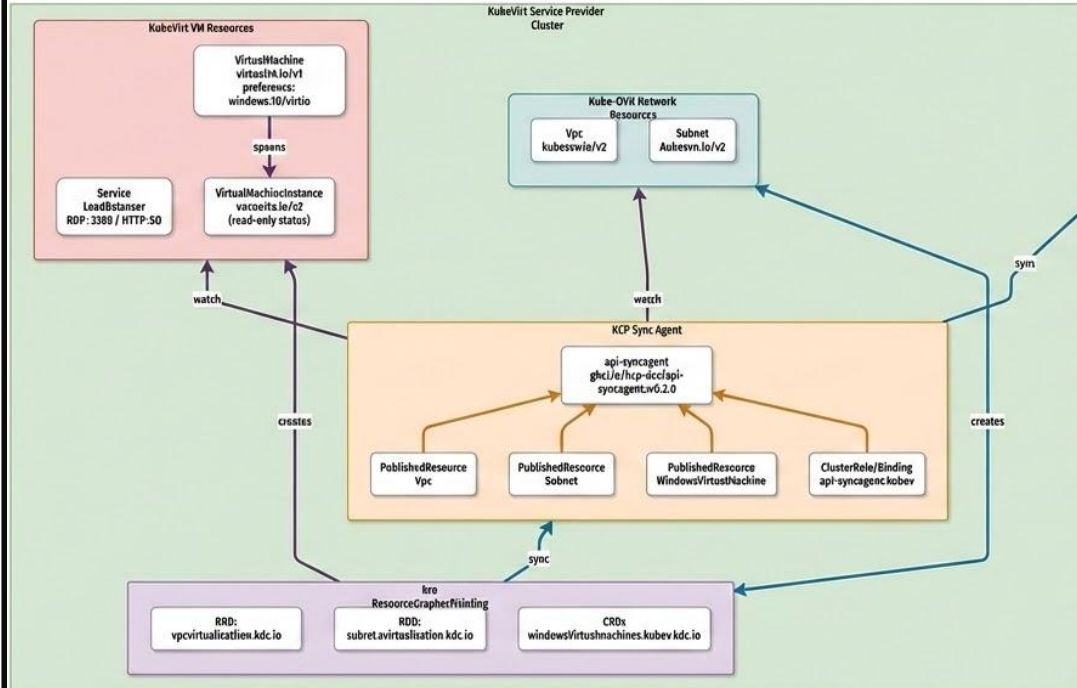




Demo: Snapshot/Restore



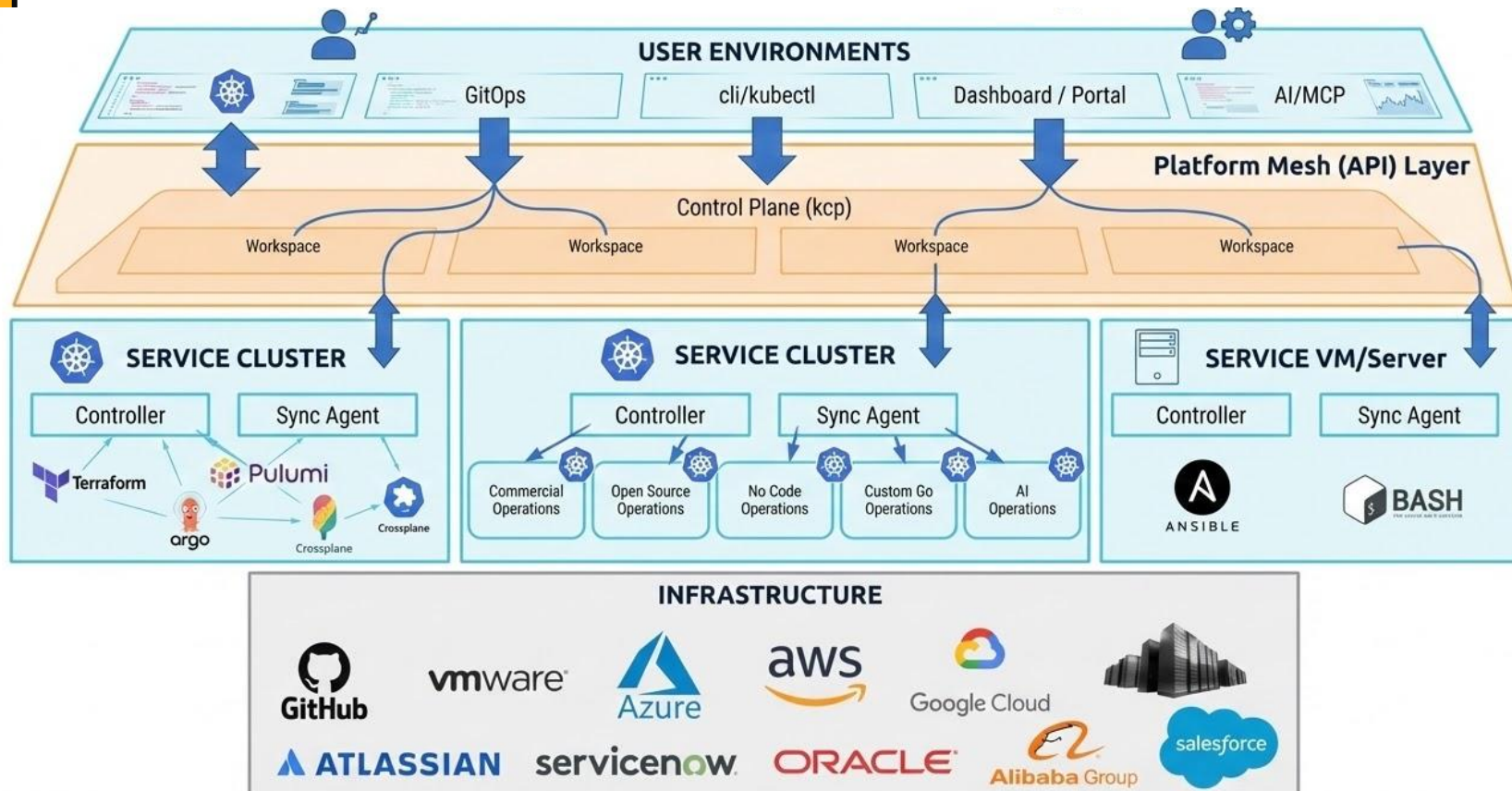
Architecture : Windows as a Service



Windows as a Service

by Kubermatic KDP

KDP - Developer Platform and Service Marketplace



KDP - Service Consumer / Marketplace

 **KUBERMATIC**
Developer Platform

Service Catalogdemo

Services

redis

Certificate Management

Database Management Syst...

Source Code Management

Web Applications

Manage

Services

Members

Roles

Namespaces

Config Maps

Secrets

Service Catalog

Browse and discover services available in the catalog.

☐ Hide added services

Sort by: Name (A-Z)



Database Management Systems

Provision and manage databases

Data

Provided by Kubermatic



Certificate Management

Manage certificates and keys

Security

Provided by Kubermatic



redis

A cloud-native service for modern applications.

General

Provided by Kubermatic



Source Code Management

Manage Git repositories

Development

Provided by Kubermatic



Web Applications

Deploy web applications

Web

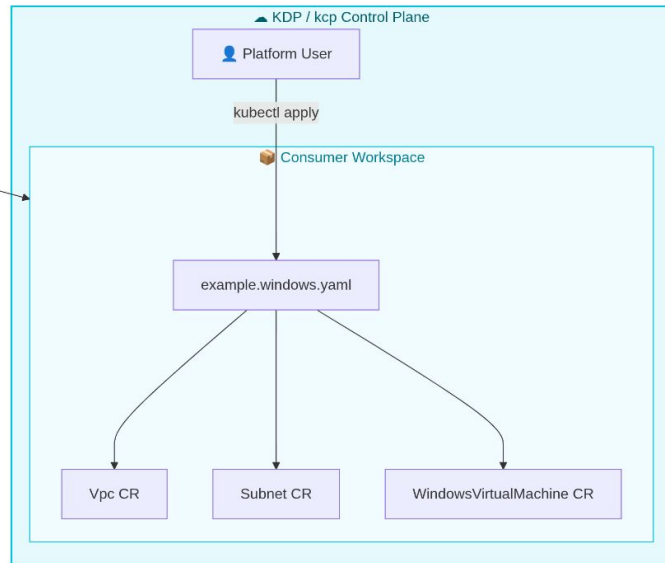
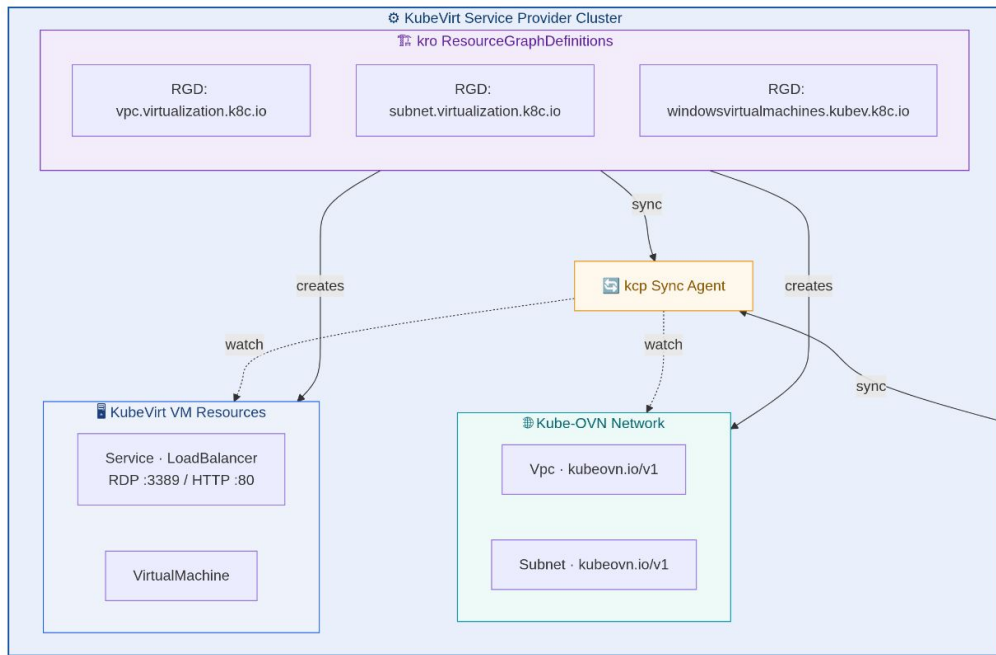
Provided by Kubermatic

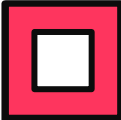
Powered by Kubermatic - Docs



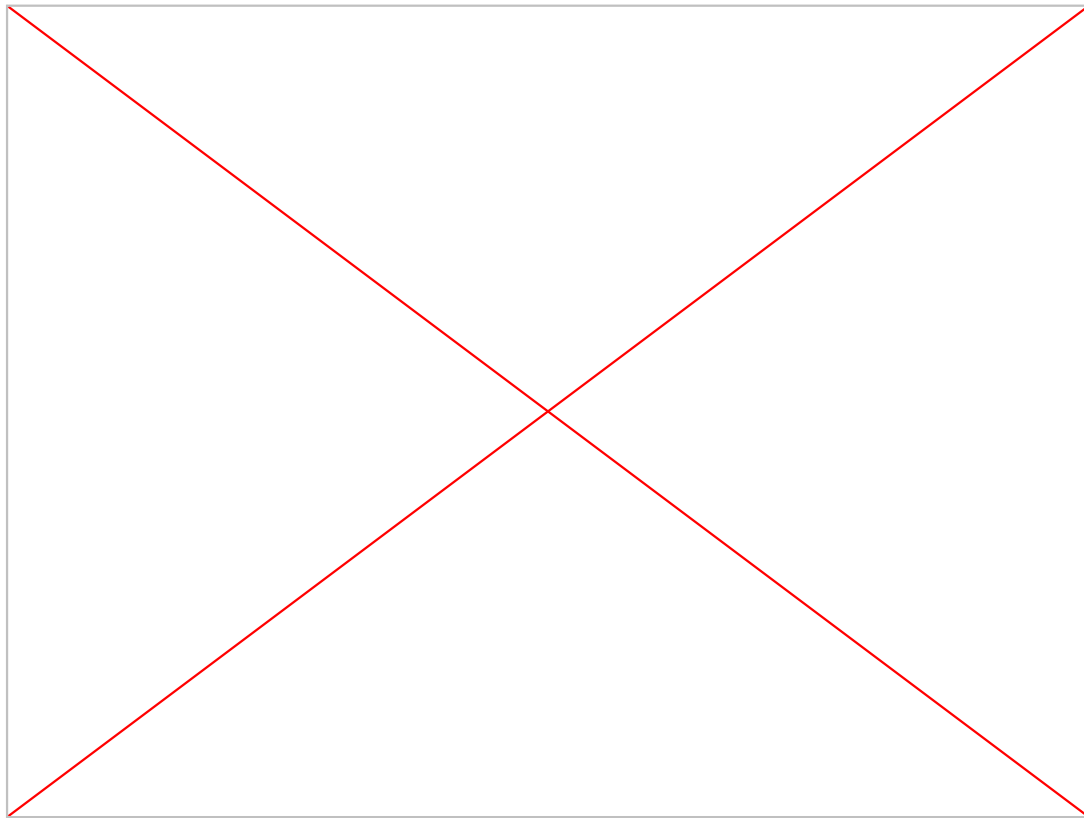
 KUBERMATIC  ODIT.Services

Architecture : WaaS via KDP

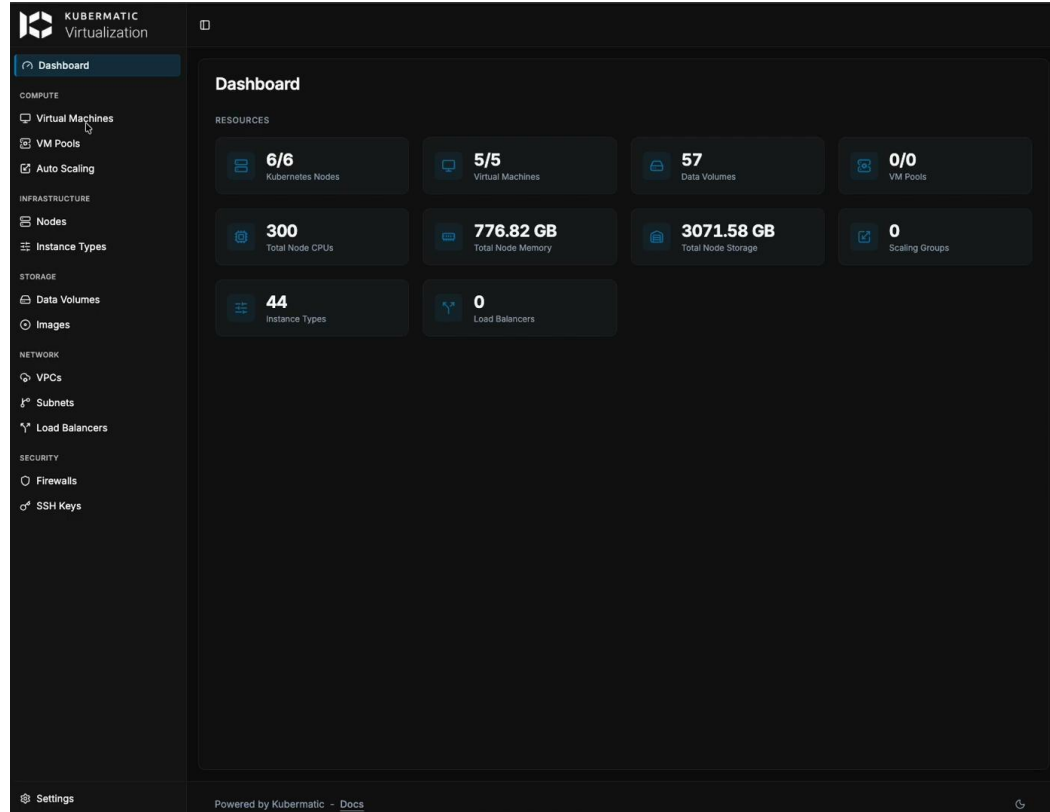




Demo: Windows as a Service via KDP



Demo: Windows as a Service via KDP





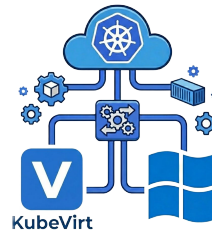
90%

Lessons learned

What went (not so) well?

The Good

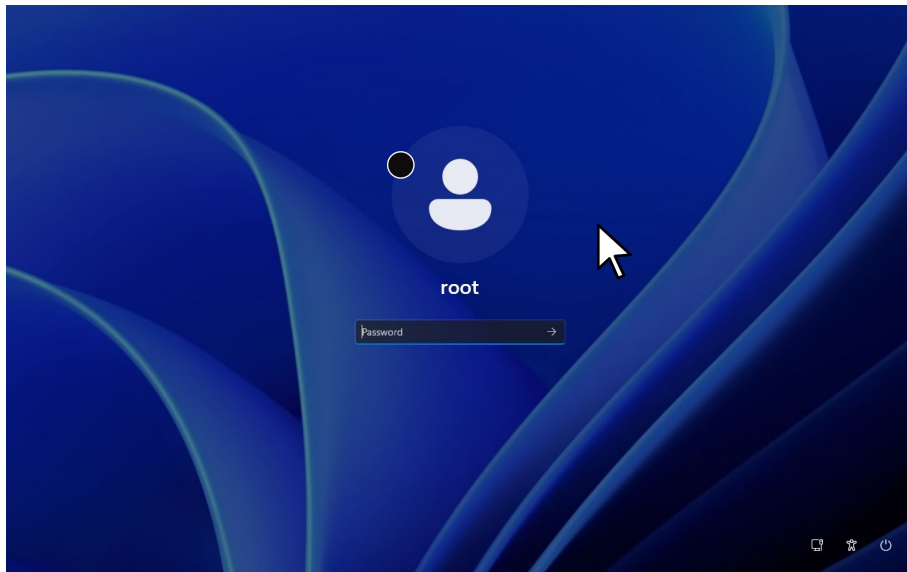
- Live migrations ✓
- Dynamic disk management ✓
- NICs in multiple networks (multus) ✓
- Snapshots/Clones ✓
- KVM console ✓
- RDP ✓
- VM Ports as Services ✓
- Import/Export VMs ✓
- Flavor- & VM-Presets ✓
- Standard and/or individual customer API ✓



IT JUST WORKS



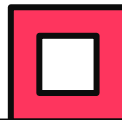
The bad(ish)



The solutions



virtio-win





The bad



Windows "CloudInit"

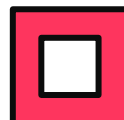
SYSPREP



The solution

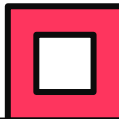
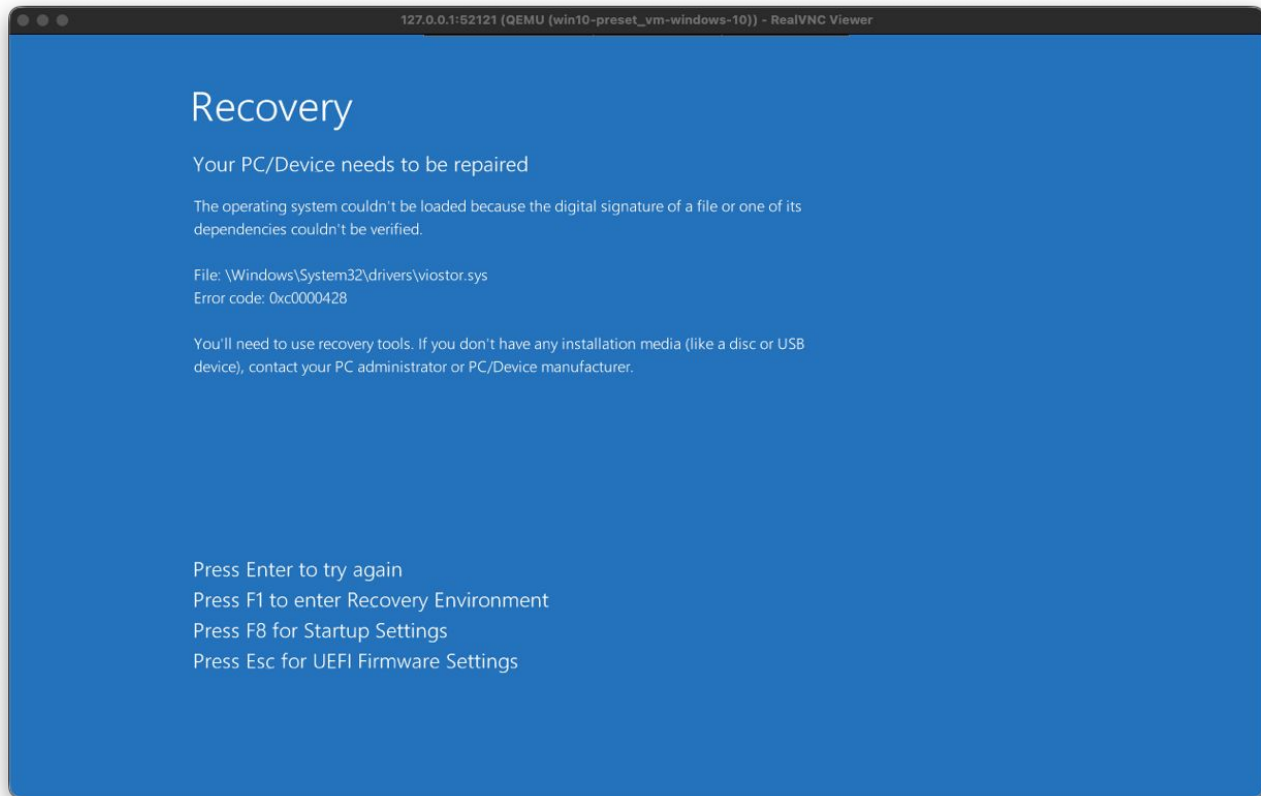
HashiCorp

Packer





The ugly





When to use



Mixed environments with some legacy applications



Licensing-cost focused orgs



Standardized stack (cattle)



Incremental migration

and when not



100% Windows-only virtualization setups (deeper tooling integration)



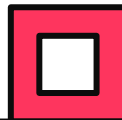
Limited Kubernetes/KVM experience



Existing robust VM management with no reason to switch



Non-standard environments (pets)



IT DEPENDS

Try it, test it, validate it, run it

Final Thoughts about KubeVirt



Growing use of Kubernetes & KubeVirt as sovereign Open Source alternative



It pretty much supports Windows the same as other popular hypervisors



Tech Stack simplification: Manage Containers and VMs through the same Kubernetes



Scaling Horizontally and Vertically is known pattern in Kubernetes



Multiple Vendor race, will grow the ecosystem



Thank you!



<https://github.com/kubermatic-labs/kubevirt-labs>



Credits

- Andrei Kvapil (CozeStack)
- Felix Matouschek (RedHat)
- Mario Fahlandt (Kubermatic)

