



KARL

KUBE ACCELERATED REAL TIME LAB

Architecture Insight



Introduction to KARL solutions

Karl is an innovative technology that enables to redefine the concept of Desktop as a Service (DaaS). All KARL software solutions are based on a patented technology (Patent No. WO2023/012553).

Unlike other traditional DaaS platforms, such as Citrix, Azure Virtual Desktop and Amazon Workspaces use only virtual machines (VMs) to provide virtualized workspaces, KARL leverages the efficiency and lightness of containers to ensure a more scalable, secure and highperformance experience.

Classic DaaS solutions rely on centralized application and desktop management, offering large-scale virtualization across VMs, which tend to be more resource intensive and less flexible when it comes to dynamic scalability. KARL, however, relies on a Kubernetes infrastructure to orchestrate lightweight and highly isolated containers that can be deployed to users as KARL Instances, virtual workstations.

This approach not only reduces operating costs, but also improves reliability and safety as containers are designed to be self-repairing with immediate recovery in the event of errors. Thanks to containerization, KARL offers a higher level of flexibility and can be deployed on public, private or hybrid clouds, easily integrating with leading cloud service providers such as Google Cloud Platform (GCP), Azure, AWS and OCI.

An emblematic example is Citrix, which has dominated the DaaS market for years, but its VM-based structure has shown limits in terms of scalability, speed of deployment and operating costs compared to what KARL offers. Our solution makes resource management more dynamic: when a KARL Instance is not in use, resources are automatically redistributed between active instances, thus optimizing overall efficiency. This is one of the main strengths of Karl. compared to other traditional solutions.



KARL DaaS infrastructure Architecture VS Traditional DaaS architecture

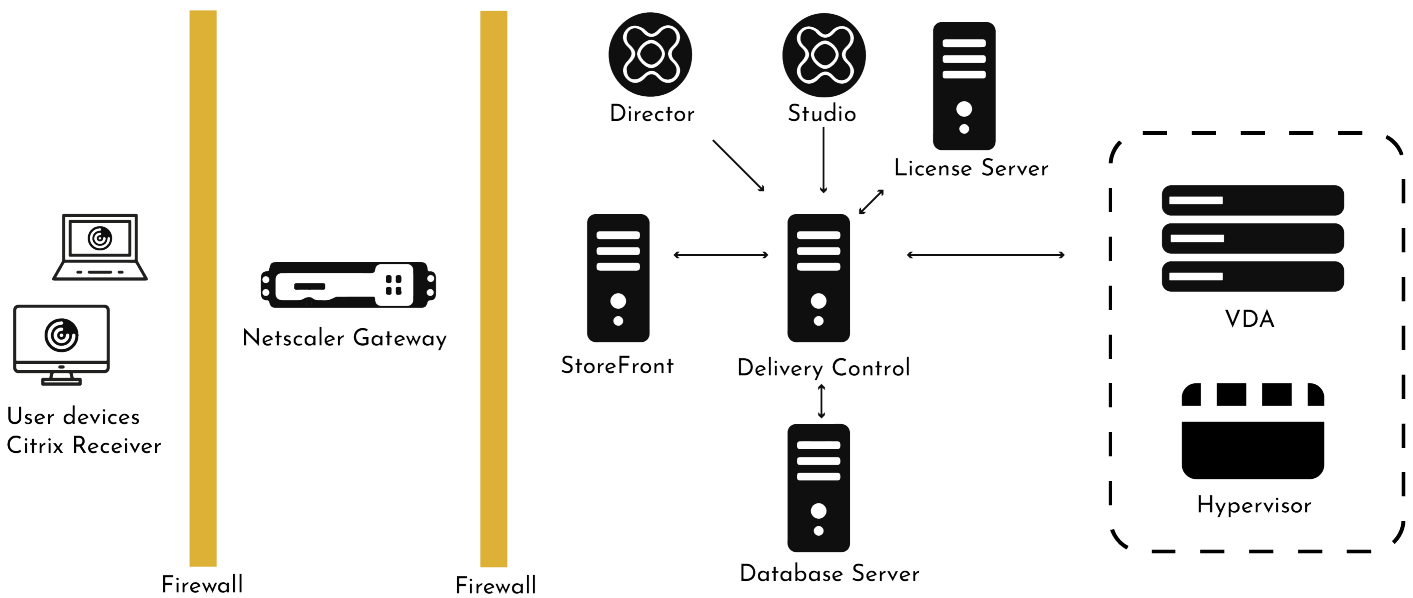


Figure 1 Traditional Architecture Citrix DaaS

The above diagram in figure 1 shows the traditional architecture of a Desktop-as-a-Service (DaaS) infrastructure, in this case Citrix. Users connect through Citrix Receiver to their devices, passing through NetScaler Gateway (for security and remote access). After passing the firewall, the workflow involves key components such as StoreFront, which manages access to applications and virtual desktops, the Delivery Controller, which distributes sessions to users, and the Database Server, which keeps track of operational information. The actual resources (virtual desktops) are run on virtual machines managed by a hypervisor (Xen Server), and the VDA (Virtual Delivery Agent) allows the distribution of the resources.

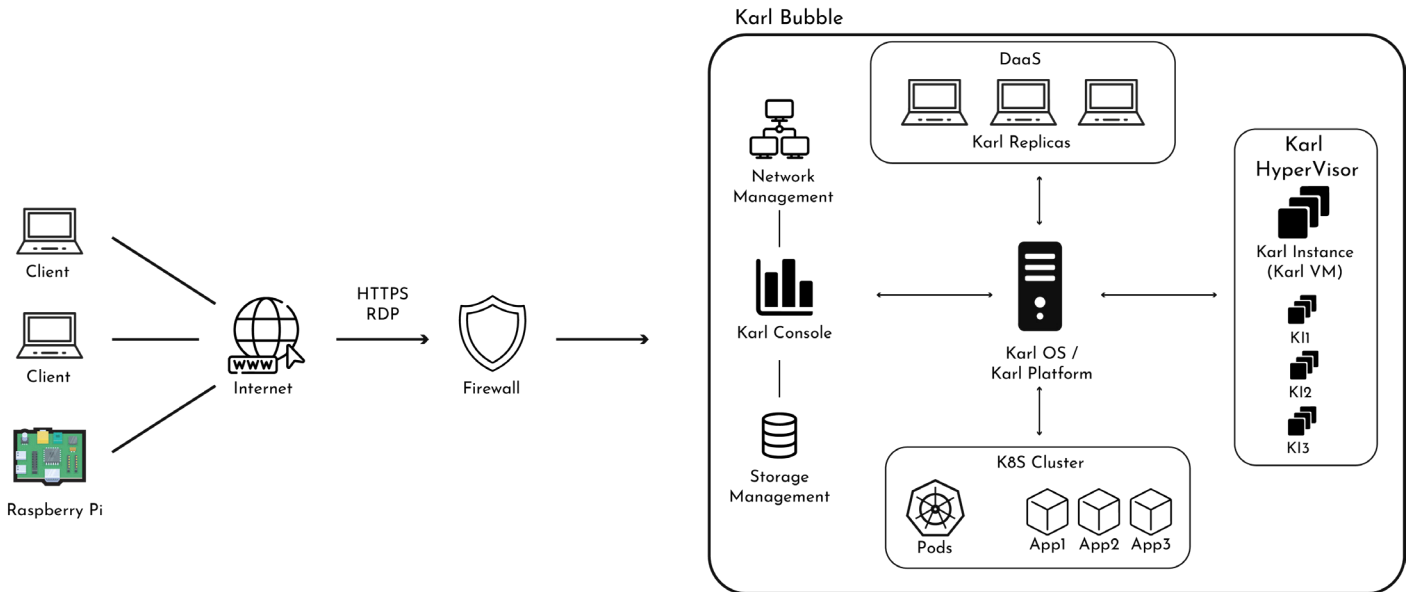


Figure 2 KARL DaaS Architecture

The KARL solution differs significantly for using containers instead of virtual machines. While Citrix relies on a complex and heavy VM infrastructure (requires a hypervisor and resource management through complete machines), Karl uses special containers (Karl Instances), which are lighter and can be scaled and deployed faster, using less resources.

Advantages of Karl:

- **Resource efficiency:** Karl Instances are lighter than VMs, which allows you to run more instances with less overhead.
- **Scalability:** Karl allows you to dynamically scale instances based on demand faster than you need to start or stop VMs.
- **Security and isolation:** Each container is isolated from the others, providing greater security without the complexity of a hypervisor. The same policy applies to a complex IT infrastructure based on customer requests.
- **Flexibility:** Karl’s multi-cloud approach allows for easier integration with different cloud providers and the possibility of building infrastructures in private datacenters or hybrid clouds, while Citrix requires a more rigid configuration.

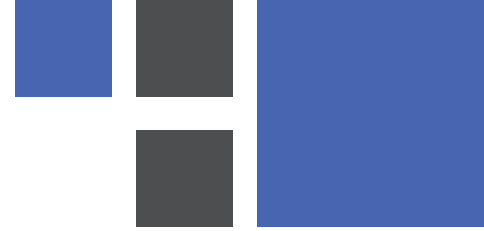
Karl Platform - Presentation

Karl Platform is the heart of the Karl ecosystem and the first product of the patented technology-based line. It is a multi-cloud software platform, compatible with the main cloud service providers (GCP, Azure, AWS, OCI), able to be deployed on public, private and hybrid clouds.

The Karl Platform is designed to enable the centralized management of a complex IT infrastructure, through the distribution of Karl Instances to end users. Karl Instances are virtual workstations that can be used on any device (PC, tablet, smartphone or even IoT devices), provided an internet connection of at least 20 Mbits is available.

A key feature of Karl Platform is its management dashboard, which allows administrators to manage the entire IT infrastructure and underlying Kubernetes cluster. Through the dashboard, administrators can:

- Manage the Kubernetes cluster, orchestrating the cluster's workloads (workloads), network, storage and other key resources.
- Manage the Karl System, the top level of the cluster that contains all the Karl Instances. Administrators can monitor the resources assigned to each instance (CPU, RAM, storage, network adapters), view event logs and connect to instances via a built-in console.
- Create and manage Karl Instances pools. This allows you to manage groups of virtual machines with collective actions, such as increasing or decreasing the resources (RAM, CPU) of all instances in a pool, improving operational efficiency. In addition, using Ephemeral Persistent Volume Claims (PVC), a single disk image can power multiple virtual machines simultaneously, reducing storage costs.



Requirements and Installation of Karl Platform

Technical requirements for Karl Platform:

- **Cloud-ready:** GCP, Azure, AWS, OCI, or a private cloud infrastructure.
- **Kubernetes cluster:** Must be provisioned with at least 3 nodes to ensure high availability and scalability of the infrastructure.
- **IAM (Identity and Access Management):** The system must support integration with authentication systems such as LDAP, SAML and OpenID, to securely manage administrative accesses.

Steps for installing Karl Platform:

- **Creating the Kubernetes Cluster:** Start a Kubernetes cluster on physical or virtual nodes, configuring the necessary resources (CPU, RAM, storage) based on the expected load.
- **Installing the Karl Platform:** Use the installation script provided by Karl Platform to install the platform on the nodes of your Kubernetes cluster.
- **Configuration of the management dashboard:** Access the administrative dashboard and configure the system, integrating IAM with enterprise authentication services.
- **Cluster resource configuration:** Define the workloads, network and storage to be allocated to Karl Instances. Use the dashboard to monitor and optimize resources deployed in your cluster.
- **Creating Karl Instances:** Through the dashboard wizard, create Karl Instance templates and distribute them to end users, ensuring secure access via RDP or browser.
- **Management and maintenance:** Continuously monitor resources through the dashboard, fixing any instance malfunction issues via automated error detection system.

Karl OS - Presentation

Karl OS is an operating system designed specifically to make the most of Karl Platform's containerized infrastructure. This operating system is optimized to work in bare metal environments, and is based on a containerization philosophy that allows an efficient and dynamic use of resources.

Unlike traditional operating systems, which require complex management of hardware resources and services, Karl OS runs as a bare metal operating system (such as Hyper-V, VMWare ESXi, Xen Server, etc.) Enabling almost instant usability of the services provided (DaaS, Karl Hypervisor, K8S). When a Karl Instance is activated, the system automatically allocates the necessary resources (CPU, RAM, storage) and optimizes them in real time, ensuring that each user can access a virtual workstation always performing, Regardless the underlying physical hardware.

Another key feature of Karl OS is the complete isolation of user resources: all user configurations and files are stored on network disks within the Kubernetes cluster, ensuring that no data is lost in case of errors or malfunctions of the instance.

Requirements and Installation of Karl OS

Technical requirements for Karl OS:

- One or more Bare Metal Servers without O.S.
- Stable Internet connection with a minimum speed of 20 Mbits to ensure the use of instances.

Steps for installing Karl OS:

- **Installing the operating system through removable media:** we'll provide installation support and wizard.
- **Installation Wizard:** the installation media provides a guided wizard for installation and initial configuration.
- **Karl Instance Deployment:** Use the pool feature to deploy Karl OS-based Karl Instances to end users. Each user will have a dedicated virtual workstation, accessible via RDP or browser.
- **Single VM Deployment:** Use the Single Karl Instance Build feature to create new VMs from ISO media. You can also import VMs from other hypervisors by converting and importing virtual machine disks.
- **Deploy workloads:** Use the K8S cluster feature to deploy applications in the form of microservices.
- **Monitoring and optimization:** Thanks to the Karl OS console, administrators can constantly monitor the performance of Karl Instances, dynamically adapting resources to ensure an optimal experience.
- **Error management:** In case of problems or malfunctions of the Karl Instances, Karl OS has an automatic recovery system that dissolves the problem Karl Instances and creates new ones without interrupting the workflow of users.

For more on our products take contact with one of our valuable partners.
We'll help you to find the right provider if you mail us at
info@karl-technology.com