Edge Cloud
Reference Architecture

Version 2.2 – May 2021

Abstract

To support digital transformation initiatives, IT departments need the right blend of on-premises, public, and edge cloud environments to support a variety of existing and emerging use cases while avoiding vendor lock-in and enabling cost optimization. They also need to combine containers with Virtual Machine workloads in a shared environment in order to get the most out of both—mature virtualization technologies plus secure container orchestration. This document presents a powerful distributed Edge Cloud Architecture for OpenNebula that is composed of Edge Clusters that can run **any workload**—both Virtual Machines and application containers— **on any resource**—bare-metal or virtualized— **anywhere**—on-premises and on a cloud provider. Our Edge Cloud Architecture enables true hybrid and multi-cloud computing by combining public and private cloud operations with workload portability and unified management of IT infrastructure and applications.

We have defined this architecture to be much simpler than traditional cloud computing architectures, which are usually composed of complex, proprietary general-purpose software systems for storage and networking. This architecture has been created from the collective information and experiences of hundreds of users and client engagements over the last ten years. It builds on storage and networking technologies that already exist in the Linux operating system and on modern storage hardware that is available from existing cloud and edge providers, leading to a greatly simplified design. Our Edge Cloud Architecture implements enterprise-grade cloud features for performance, availability, and scalability with a very simple design that avoids vendor lock-in and reduces complexity, resource consumption, and operational costs.

Contents

Glossary

DFS	Distributed File System
HA	High Availability
I/O	Input/Output
IaC	Infrastructure as Code
QCOW	QEMU Copy On Write
VM	Virtual Machine

1. What is OpenNebula?

OpenNebula is a **powerful, but easy-to-use, open source solution to build and manage Enterprise Clouds**. It combines virtualization and container technologies with multi-tenancy, automatic provision, and elasticity to offer on-demand applications and services. OpenNebula provides a single, feature-rich and flexible platform with **unified management of IT infrastructure and applications** that **avoids vendor lock-in** and **reduces complexity, resource consumption, and operational costs**. OpenNebula manages:

- **Any Application**: Combine containers with Virtual Machine workloads in a common shared environment to offer the best of both worlds: mature virtualization technology and orchestration of application containers.
- **Any Infrastructure**: Unlock the power of a true hybrid and multi-cloud platform by combining edge, public, hosted, and private cloud operations.
- **Any Virtualization**: Integrate multiple types of virtualization technologies to meet your workload needs, from a fully virtualized environment to system containers and serverless deployments.

OpenNebula provides the necessary tools for running containerized applications from Kubernetes and Docker Hub, while ensuring enterprise requirements for your DevOps practices. It helps organizations to easily embrace Multi-cloud, Hybrid, and Edge Computing, allowing them to grow their Enterprise Cloud on demand with infrastructure resources from third-party Public Cloud and bare-metal providers such as AWS and Equinix Metal. OpenNebula supports a number of virtualization technologies, including VMware and KVM Virtual Machines for fully virtualized clouds, LXC system containers for container clouds, and Firecracker microVMs for serverless deployments.

This white paper describes OpeNebula's Edge Cloud Architecture, which is composed of Edge Clusters that can run **any workload**—both Virtual Machines and application containers—**on any resource**—bare-metal or virtualized—**anywhere**—on-premises or on a cloud/edge provider. If you want to find out more about our innovative approach towards edge computing, please refer to our white paper on how to Accelerate Edge Cloud Computing with OpenNebula.[1] If you are interested in designing and deploying an OpenNebula cloud on-premises on top of VMware vCenter, please refer to our VMWare Cloud Reference Architecture.[2] If you are interested in building an OpenNebula cloud using your choice of open source platforms and technologies, please refer to our Open Cloud Reference Architecture.[3]

The development of OpenNebula follows a bottom-up approach driven by the real need of sysadmins, DevOps, and corporate users. OpenNebula is an **open source product** with a healthy and active community, and is commercially supported by OpenNebula Systems through its **OpenNebula Subscription**. Releases are produced on a regular basis and delivered as a single package with a smooth migration path. More information on the benefits of running an OpenNebula cloud can be found on the key features page.[4]

2. Design Principles for Distributed Edge Cloud

The pressure on IT Departments and Operation Teams is increasing at a high rate. They are expected to:

- **Innovate** to bring new applications and services to market faster.
- **Be agile** to meet the increasing needs of developers for new development tools and frameworks, like containers or infrastructure as code (IaC), while continuing to offer infrastructure on demand to application administrators and ensuring enterprise requirements for DevOps practices.

[1] https://support.opennebula.pro/hc/en-us/articles/360050159871-Accelerate-Edge-Cloud-Computing-with-OpenNebula-White-Paper
[2] https://support.opennebula.pro/hc/en-us/articles/206652953-VMware-Cloud-Reference-Architecture-White-Paper
[3] https://support.opennebula.pro/hc/en-us/articles/204210319-Open-Cloud-Reference-Architecture-White-Paper
[4] https://opennebula.io/discover/

- **Adapt to stay relevant** in a new reality where hybrid cloud and multi-cloud are quickly growing, and edge computing is the next IT transformation.

From the experience of working with hundreds of users and client engagements, we have defined an Edge Cloud Architecture that builds a single distributed cloud platform to run any workload—from virtualized to containerized—across multiple clusters that can run anywhere—on-premises or at the edge—and on any resource—from bare-metal to virtual—with unparalleled availability, performance, and simplicity.

- The architecture defines a **complete end-to-end solution** based on proven open source storage and networking technologies that already exist in the Linux operating system to avoid vendor lock-in; minimize IT complexity, resource consumption, and operational costs; maximize performance, availability, and reliability; and simplify the automation of its deployment and management.
- The architecture implements **a distributed approach** with a single cloud Front-end that controls one or several interconnected Edge Clusters that can run in multiple geographically distributed data center locations, and cloud and edge resource providers.
- **Clusters can be added and removed dynamically** in order to meet peaks in demand, or to implement fault tolerance strategies or latency requirements.
- The **Edge Clusters are based on a common reference architecture** that has been designed to be deployed on any resource, from bare metal to virtual, enabling **workload portability**.
- The **Edge Cluster's internal design follows a hyper-converged approach** that uses the optimal infrastructure configurations in each environment to ensure isolation and performance, to easily scale-in and scale-out nodes to match compute and storage needs, and to migrate workload.
- The edge architecture **enables true hybrid and multi-cloud computing,** and leverages the growing ecosystem of cloud, edge, and access network cloud providers.
- The **Edge Clusters are capable of operating autonomously** in case of no network connectivity, to allow applications running in local data centers to continue their operation without any management service.
- The system supports modern applications, **combines application containers and Virtual Machines on a single platform**, and integrates with existing Virtual Machine and container images hubs and marketplaces. While new workloads based on containers, microservices, and functions (serverless) should generally be stateless and ephemeral, almost all business applications require data persistence in some form. This is why the Edge Cloud Architecture provides support for both non-persistent and persistent VMs and containerized applications.

Lastly, it produces a **single vendor experience** because the complete cloud stack is fully supported and optionally managed by OpenNebula Systems. This means a simplified experience for procurement, consulting, and support. You can accelerate and streamline roadmap development, migrations, and upgrade paths with a single vendor.

3. High Level Reference Architecture

Modern cloud environments have evolved to highly complex backends that limit their reliability and hinder their operability. They are composed of proprietary, costly general-purpose software systems for storage and networking that are unnecessarily complex because they were designed to solve too many problems at once. The OpenNebula Edge Cloud Architecture departs from this trend by using lightweight systems and software components with modest hardware requirements that are easily healed and operated while also providing maximum application performance, and by implementing an innovative design specifically designed to manage virtualized applications in distributed multi-site environments.

OpenNebula's Edge Cloud Architecture is articulated around the concept of Edge Cluster. An Edge Cluster is a hyperconverged functional set of managed objects that include storage, network, and host resources. An

Edge Cluster provides all the resources needed to run virtualized or containerized applications. OpenNebula's management services, including scheduling, monitoring and life-cycle management, run in the cloud Front-end and orchestrate the Edge Clusters. The Front-end also provides access to the administration tools, user interfaces, and API. Although the requirements may vary depending on the number and size of the clusters and API load, the Front-end node only requires 8 GB of main memory and 4 cores. The basic building blocks of the architecture, including the Edge Clusters, are depicted in Figure 1.

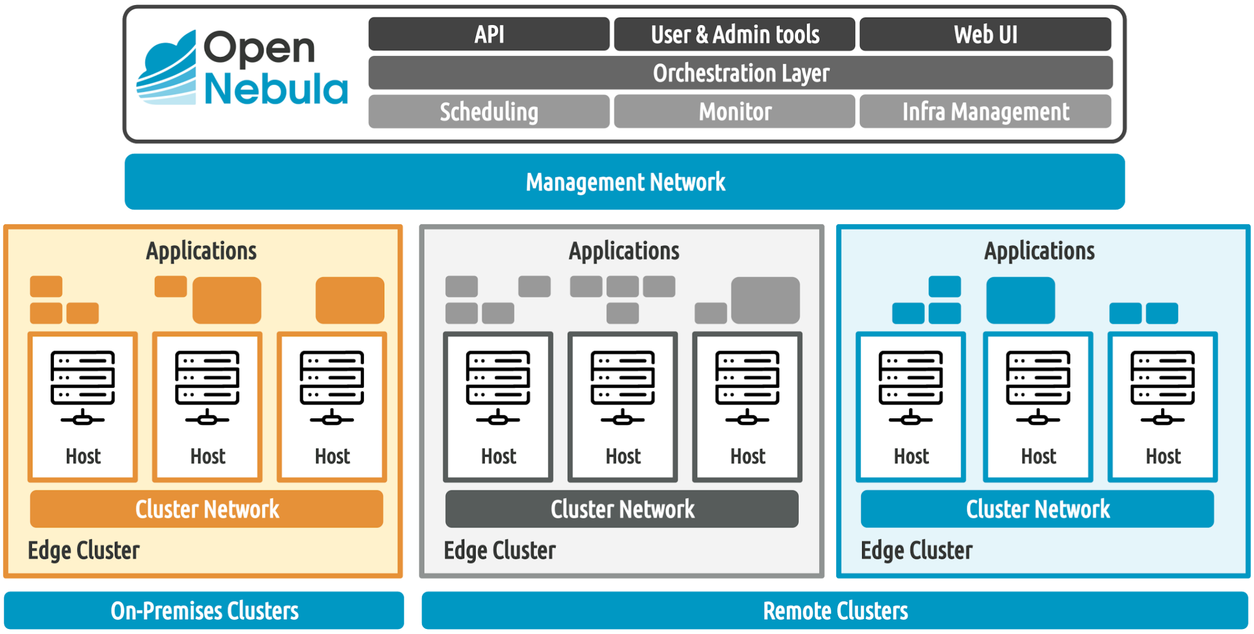


Figure 1. Main components of our Edge Cloud Architecture.

The Edge Clusters are built on virtualization hosts and an interconnection network. The virtualization hosts are responsible for providing applications with execution resources (e.g. CPU, memory, or network access) through a suitable virtualization form, e.g. Virtual Machines (KVM), system containers (LXC), or microVMs (Firecracker/KVM). The actual virtualization technology depends on how the cluster hosts are provisioned (see Section 4) and the profile of the workload. Additionally, hosts supply the cluster with the storage space needed to run the applications.

The configuration of the hosts of a cluster is homogeneous in terms of the installed software components, OpenNebula administration user, and accessible storage. The Front-end nodes need to be connected to the cluster hosts to receive status and monitoring updates as well as to initiate management operations. Cluster nodes are interconnected through one or more private networks, normally used for storage transfers as well as for application communication across virtualization hosts. Finally, access to a public link is required to provide applications with internet connectivity.

The Edge Cloud Architecture is able to provide a lightweight and easy-to-use storage platform for medium-sized clusters consisting of tens of nodes that can run on-premises and on-cloud, and on physical and on virtualized resources. Overall, OpenNebula's Edge Cloud Architecture is able to manage hundreds of these clusters, as they operate autonomously in terms of networking and storage, and handle thousands of virtualized hosts and tens of thousands of virtualized applications.[5]

[5] https://docs.opennebula.io/stable/deployment/references/one_scalability.html#front-end-oned-scalability

4. Edge Cluster Deployment Models

The main advantage of the OpenNebula Edge Cloud Architecture is the ability to deploy clusters anywhere. This provides application mobility and true multi-cloud computing in the most literal sense. Clusters can be deployed on on-premises infrastructure, on public bare-metal providers, and on virtualized cloud environments to enable powerful hybrid and multi-cloud computing. Infrastructure teams have the flexibility to choose their preferred hardware platform and cloud provider, and deliver an exceptional OpenNebula experience.

In particular clusters can be provisioned in three different forms:

- **On-premises** assumes full management capabilities of the resources and has no restrictions in terms of networking. Usually, this type of provision uses in-house data center resources.
- **Metal Remote** presents some restrictions on IP addressing and connectivity of resources. A metal provision usually requires interacting with a cloud/edge provider API and uses bare-metal instances.
- **Virtual Remote** presents limitations on the capabilities of the cluster hosts as well as network connectivity. A virtual remote cluster deployment is usually based on Virtual Machine instances from a cloud or edge provider.

Note that some cluster deployments or some cloud providers may impose some restrictions on the functionality available in the cluster (e.g. supported hypervisors or host connectivity). As shown in Figure 1, a single Front-end can manage clusters from multiple locations at the same time. While the cluster architecture has been tested on the main major cloud providers, the automatic provision feature only supports Equinix Metal and AWS. OpenNebula is developing drivers for other widely used cloud and edge providers.

5. Storage Architecture

Our Edge Cloud Architecture is based on **OneStor**, a specialized storage solution—fully supported by OpenNebula Systems—that has been developed for the efficient management of disk images in highly distributed environments. OneStor has been designed to meet the following requirements:

- **Access to external (public) marketplaces** that act as global image repositories. Examples include OpenNebula Marketplace or Docker Hub, but also private HTTP repositories or container registries.
- **Minimize image transferring and maximize application I/O performance**. Clusters can use a cloud/edge deployment model over public internet links. Moreover, on-premises provisions can scale to a high number of hosts. Storage should not be a bottleneck for any of these situations.
- **Simple deployment.** Reduce the complexity and technology footprint of the solution by using technologies that already exist in the Linux operating system to accommodate any deployment model, both on physical and virtual resources, as well as increasing the reliability of the backend.

The Edge Cloud Architecture combines a 3-tier global architecture for image distribution with an enhanced Filesystem Datastore with replica caching, snapshotting, and backups within each Edge Cluster. The 3-tier storage architecture (see Figure 2) consists of:

- **Tier 1 - Marketplace**. This tier consists of the remote servers and storage implementing the global application image repositories. It includes third-party sites like Docker Hub or Linux Containers, as well as OpenNebula Public Marketplace.

- **Tier 2 - Image Datastores**. This tier consists of the zone image datastores and provides the primary image storage location for the OpenNebula zone. This storage area is provided for one or more dedicated nodes. Its contents are cached and replicated on demand to each cluster. This supports the Edge Cluster deployment on any location as well as to scale the on-premises infrastructures.

- **Tier 3 - Cluster Replicas.** Applications images are cached within a cluster in dedicated replica hosts to minimize image transferring. The replica hosts make use of a specialized distribution system to make the images available to all cluster hosts and support snapshotting for host failure recovery.

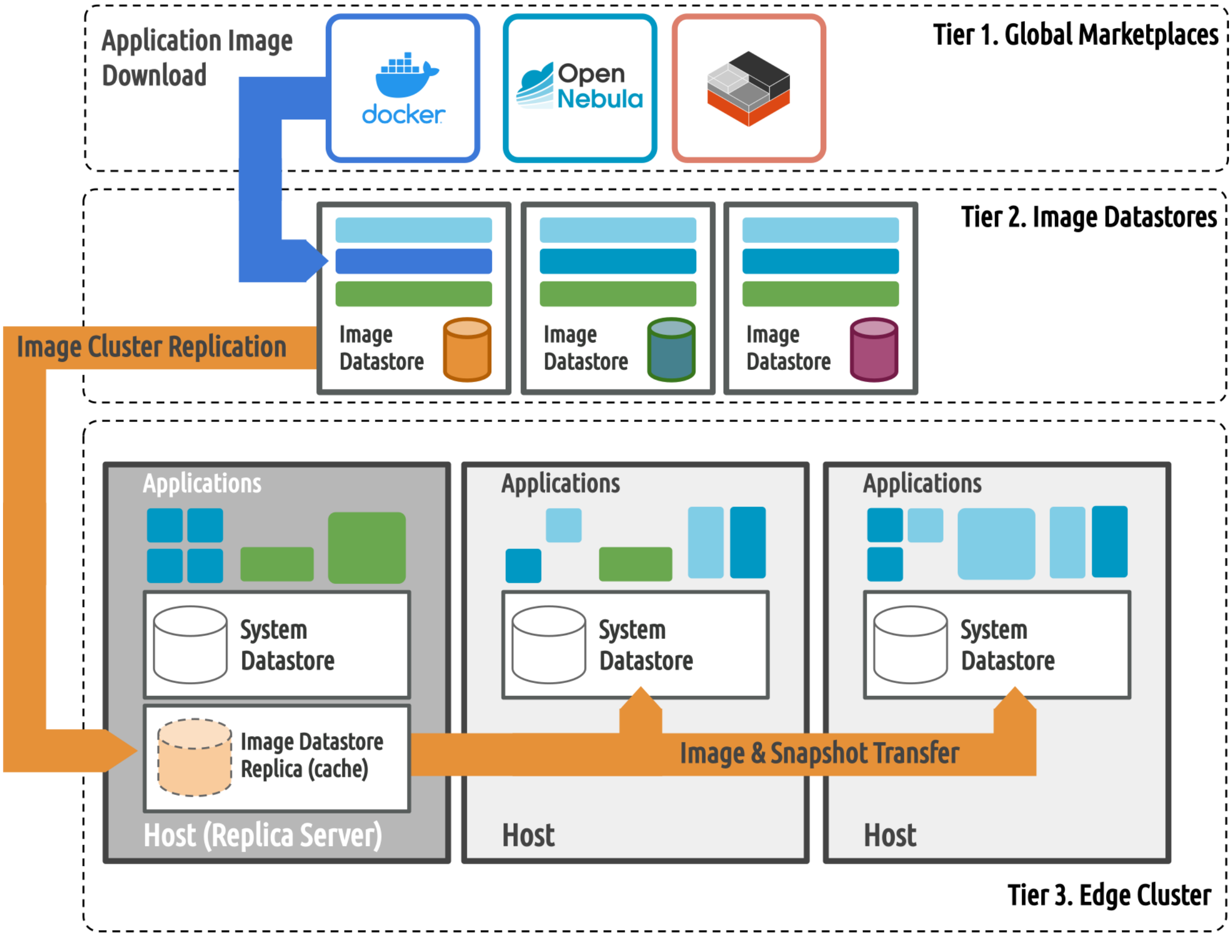


Figure 2. The 3-tier storage system of the Edge Cloud Architecture.

Within each tier-3 cluster, disk images are transferred between the Image and System datastores by using an enhanced SSH transfer mode with replica caching and snapshotting that greatly improves its scalability, performance, and reliability. The new replica mode caches the images in each cluster so they are available close to the hypervisors to reduce the bandwidth requirements to the tier-2 Image datastore servers and considerably reduce deployment times. This is especially important in highly distributed edge deployments where copying images from the tier-2 Front-end to the tier-3 cluster hypervisors could be very slow.

The edge storage architecture has been implemented with lightweight technology components that translate into modest hardware requirements, like SATA SSDs and 10 gigabit networks. Moreover, its deployment follows a hyperconverged approach that does not require dedicated servers to implement a distributed storage system. This reduces the complexity of the solution, enabling the use of the local storage area of the cloud cluster hosts.

Application Deployment Performance

Application images are based on files with qcow2 format to reduce file transfer and instantiation times with minimal overhead. The use of qcow2 files to back disk images also eases backup solutions, reduces image transfer times, and implements advanced features like snapshotting in an efficient way. The expected deployment times depend on the interconnection links between tiers and disk sizes. As a reference, for a bare-metal remote cluster (tier 3) in Frankfurt connected using public internet links to a tier-2 Image datastore located in Amsterdam, we have obtained average deployment times of 5s (hot caches) and 35s (first-time, empty cache) for applications using qcow2 disks of 0.5 GiB. Average deployment times for the aforementioned clusters, with cached and without cached images, are described in Figure 3. When the clusters are deployed on premises using 1Gb/s links, the deployment time is always 5s.

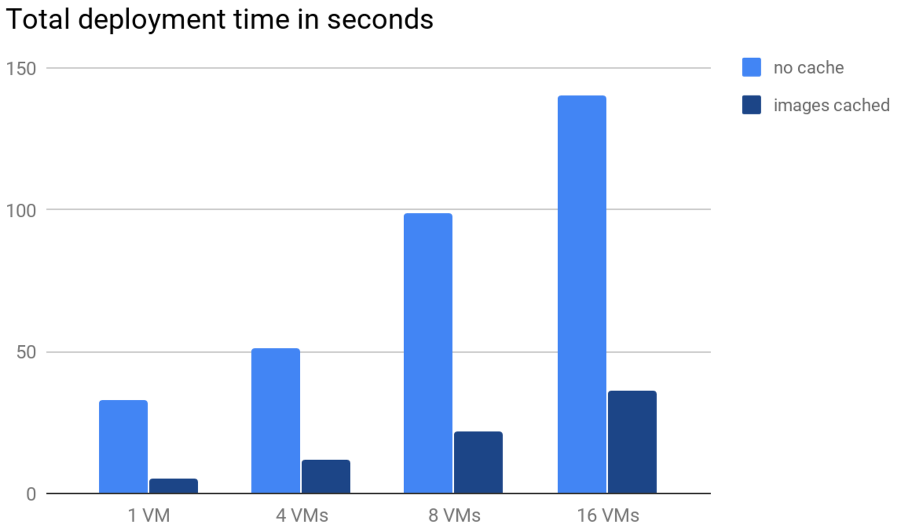


Figure 3. Comparison of deployment times with/without cached images.

Application I/O Performance

The applications run from the direct attached storage of the hosts to maximize the available I/O performance delivered to the applications. The I/O performance is close to that of the native host and it is only impacted by the virtualization layer. In order to provide an estimation of this overhead, we have run the Flexible I/O tester on both the host and a VM. The measures have been taken on a t1.small metal instance on Equinix Metal and a VM running on the very same server. Table 1 shows a summary of the characteristics of both machines.

Physical Server	Virtual Machine
1 x Intel Atom C2550 @ 2.4Ghz 8GB RAM 1 x 80GB SSD DISK NIC 2 x 1Gbps Bonded Ports	Virtual Machine KVM kernel 4.18.0-147.8.1.el8_1.x86_64 Libvirt-6.0.0 Qemu (cache=none, io=native)

Table 1. Main characteristics of the servers used to run the benchmarks.

Table 2 shows the average bandwidth obtained while running the benchmark for read and write operations, both sequential and random. The performance is similar between the physical host and VM, with differences of less than 5%. Note that some latency is hidden by the virtual page cache.

		Host	Virtual Machine
Sequential	Read	179 MiB/s	171 MiB/s
	White	168 MiB/s	181 MiB/s
Random	Read	84.8 MiB/s	90.3 MiB/s
	White	90.2 MiB/s	94.8 MiB/s

Table 2. Average IO bandwidth for read/write operations on the physical host and Virtual Machine.

Application Snapshot Performance

In order to improve the availability of the Edge Cluster blocks, live migration is supported within the same cluster. Application snapshots are also kept within the Edge Cluster (tier 3) to enable fast recovery from the last application checkpoint.

The recovery operation can potentially impact on two different areas:

- I/O noise generated by the snapshot operation that could reduce the I/O of neighbor applications. In our case this cost is negligible as it is based on the QEMU Redirect-on-Write feature.
- Network bandwidth to move the recovery snapshots to the cluster replica server. In this case we use a delta-transfer algorithm to reduce the information transferred to the server. However, this time will increase as the contents of the disk diverges from the original.

Another important aspect to consider is the VM recovery time. Compared to recovering a VM without any snapshot the time is similar, as the base images are already located on the cluster replica (tier 3) and the only additional overhead is the transfer of the disk snapshot, already available in the Edge Cluster as well.

Alternative Storage Back-ends

OpenNebula provides support for most popular enterprise-class SDS (Software Defined Storage) backends, such as Ceph or Gluster, designed to offer high scalability and availability[6]. Although high-performance and feature rich, the operation of these platforms require extensive experience, higher financial investment, dedicated hardware (whether or not they are hyperconverged), and a significant amount of resources. In general, their cost, complexity, and resource requirements prevent their practical use in Edge Clusters.

6. Networking Architecture

Clusters use four types of networks:

- **Storage network** devoted to the distribution system of application images.
- **Private network** to implement application interconnection networks.
- **Management network** to interconnect the hosts to the Front-end services.
- **Public network** to interconnect applications to the Internet. Usually requires a predefined set of

[6] https://support.opennebula.pro/hc/en-us/articles/360019581717-Choosing-the-Right-Storage-for-Your-Cloud-Report

public IPs available in the Edge Cluster.

Note that the Management traffic could be routed through the public network when the clusters are deployed remotely.

The characteristics of each network strongly depend on the deployment model used for a cluster. The on-premises model provides full networking capabilities and is based on standard linux bridging. Private application networking is implemented through VLAN tagging (802.1Q).

On the other hand, cloud or Edge Clusters may present some limitations to the topology available for each network type depending on the selected provider. In general, the OpenNebula network stack uses specific provider drivers to register elastic public IPs to applications as well as private networks.

In both cases, applications benefit from the features implemented by OpenNebula's network kernel, including security groups, automatic network scheduling, and user self-provisioning models.

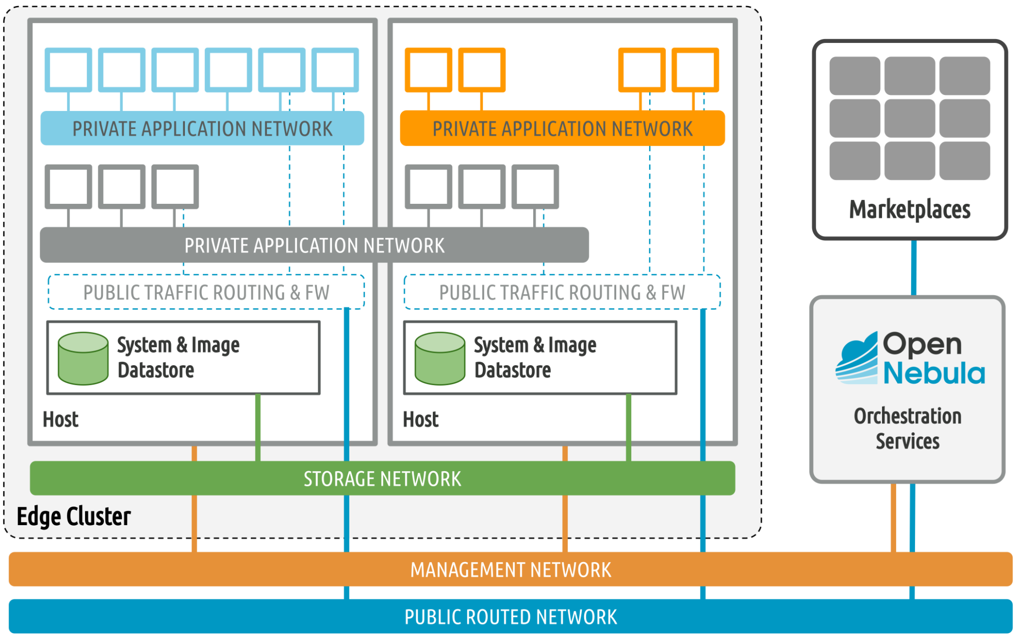


Figure 3. Overview of the networks used in the Edge Cloud Architecture.

7. High Availability

Availability is a critical aspect of cloud architecture design, mostly when it comes to application data recovery and integrity. The Edge Cloud Architecture provides multiple mechanisms and built-in capabilities to achieve high availability and guarantee that your data and services are always available:

- **Front-end**: OpenNebula uses a distributed consensus protocol, based on Raft, to provide fault tolerance and state consistency across OpenNebula Front-end services. A minimum of three Front-ends needs to be deployed in order to support one node failure. When running the Front-end on cloud resources, the Front-end HA clusters can be deployed across countries or even continents.
- **Multi-cluster**: The Edge Cloud Architecture helps you to automatically deploy geo-distributed cloud and edge environments that enable multi-cloud availability patterns by allowing applications to span across multiple clusters. Thanks to this approach, you can make your applications more highly

available, scalable, and fault tolerant than traditional single data center infrastructures.

- **Cluster**: OpenNebula implements autonomous operation of remote clusters in case clusters running on cloud and edge locations might lose their connection with the centralized centers. Applications running in a cluster can continue their operation without any management service.

- **Host**: The 3-Tier Replica Storage architecture in the Edge Clusters implements an availability system based on periodic snapshots that are used to recover from VM and host failures by fencing the node to prevent split-brain conditions (soft, because of local I/O) and automatically restarting the application in another node. Application snapshots are kept within the cluster (tier 3) to enable fast recovery from the last application checkpoint. Availability can be improved within each node by HW replication (HD Raid and NIC) and network paths, and within the application by implementing application-level HA, when data and application state integrity is required, across multiple clusters.

- **Application**: The networking architecture of the Edge Cluster and the OpenNebula orchestration capabilities allow developers to use application-specific replication mechanisms where data and application state integrity is required. Moreover, maintenance work can be performed while the system is operating and host failures can be mitigated thanks to the support for application live-migration within cluster hosts.

8. Automatic Provision of Edge Clusters

OpenNebula brings the provisioning tools and methods needed to dynamically grow on demand a private cloud infrastructure with resources running on remote cloud and edge providers. This disaggregated cloud approach allows a seamless transition from centralized private clouds to distributed edge-like cloud environments. Companies are able to grow their private cloud with resources at cloud and edge data center locations to meet peaks in demand or the latency and bandwidth needs of their workload. This approach involves a single management layer where organizations can continue using the existing OpenNebula images and templates, keep complete control over the infrastructure, and avoid vendor lock-in.

The OneProvision[7] tool allows the deployment of a fully operational Edge Cluster in a remote provider and the management of its full life-cycle, starting with its provisioning and maintenance, until the unprovision. Each cloud or edge location (the “**provision**”) is defined as a group of physical hosts allocated from the remote bare-metal or virtual provider. They are fully configured with the user-selected hypervisor and enabled in the cloud stack for the end-users.

9. Run Any Application, Anywhere

One of the critical parts of an OpenNebula cloud is its ability to support modern applications, combine application containers and Virtual Machines on a single platform, and integrate with existing Virtual Machine and container image hubs and marketplaces.

OpenNebula has inherent access to its own **Marketplace**, which enables users to import images from a public repository (containing images of common use that have been tested and certified by OpenNebula Systems) or from private repositories. These images can be added to a datastore and used at any time by existing VM templates or instances.

[7] http://docs.opennebula.io/stable/advanced_components/ddc/index.html

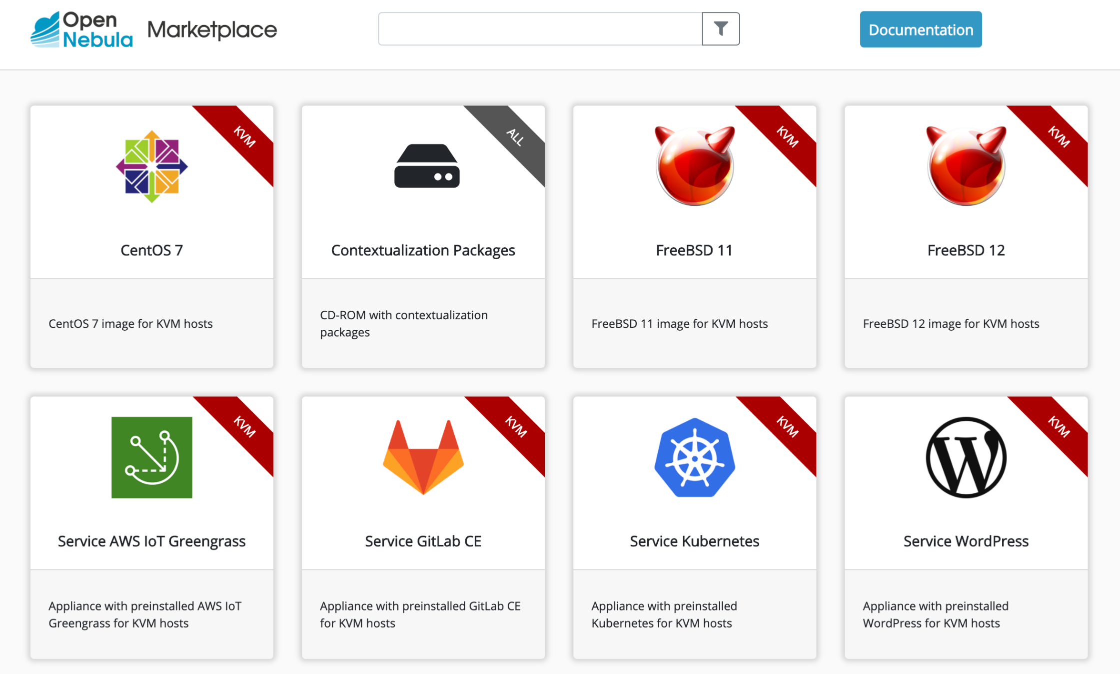
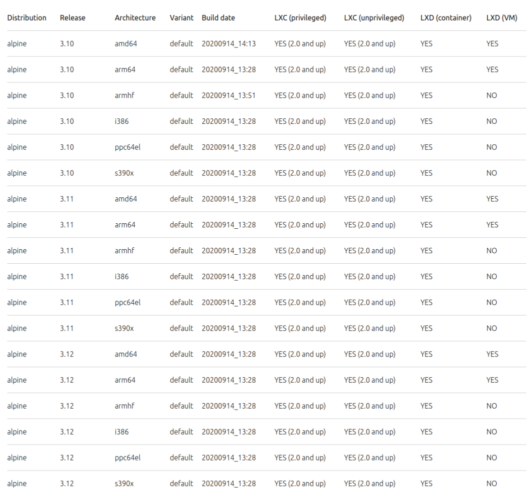


Figure 5. Some of the virtual appliances available in the OpenNebula Public Marketplace.

OpenNebula users can easily download, contextualize and add virtual appliances from other public marketplaces, including **Linux Containers**[8] and **TurnKey Linux**.[9]



Distribution	Release	Architecture	Variant	Build date	LXC (privileged)	LXC (unprivileged)	LXD (container)	LXD (VM)
alpine	3.10	amd64	default	20200914_14:13	YES (2.0 and up)	YES (2.0 and up)	YES	YES
alpine	3.10	arm64	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	YES
alpine	3.10	armhf	default	20200914_13:51	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.10	i386	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.10	ppc64el	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.10	s390x	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.11	amd64	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	YES
alpine	3.11	arm64	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	YES
alpine	3.11	armhf	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.11	i386	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.11	ppc64el	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.11	s390x	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.12	amd64	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	YES
alpine	3.12	arm64	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	YES
alpine	3.12	armhf	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.12	i386	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.12	ppc64el	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO
alpine	3.12	s390x	default	20200914_13:28	YES (2.0 and up)	YES (2.0 and up)	YES	NO

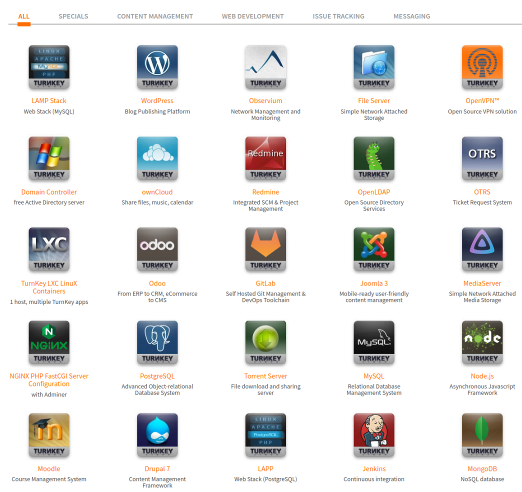


Figure 6. Some of the images provided by Linux Containers (left) and TurnKey Linux (right).

Application development is now increasingly relying on microservices architectures, which avoid breaking the whole stack when a particular feature is modified or added to the application. This trend in development is tightly coupled with the way these applications are deployed, usually through **application containers.** From version 5.12 "Firework" onwards, OpenNebula also comes with seamless integration with **Docker Hub**,[10] permitting direct execution of Docker Hub images on any hypervisor in your OpenNebula

[8] https://images.linuxcontainers.org
[9] https://www.turnkeylinux.org
[10] https://hub.docker.com

open cloud. This is a development and distribution model for applications and is especially suited for edge environments, since only a few legacy applications will ever be deployed on the edge, and new applications will most likely be developed using these modern models.

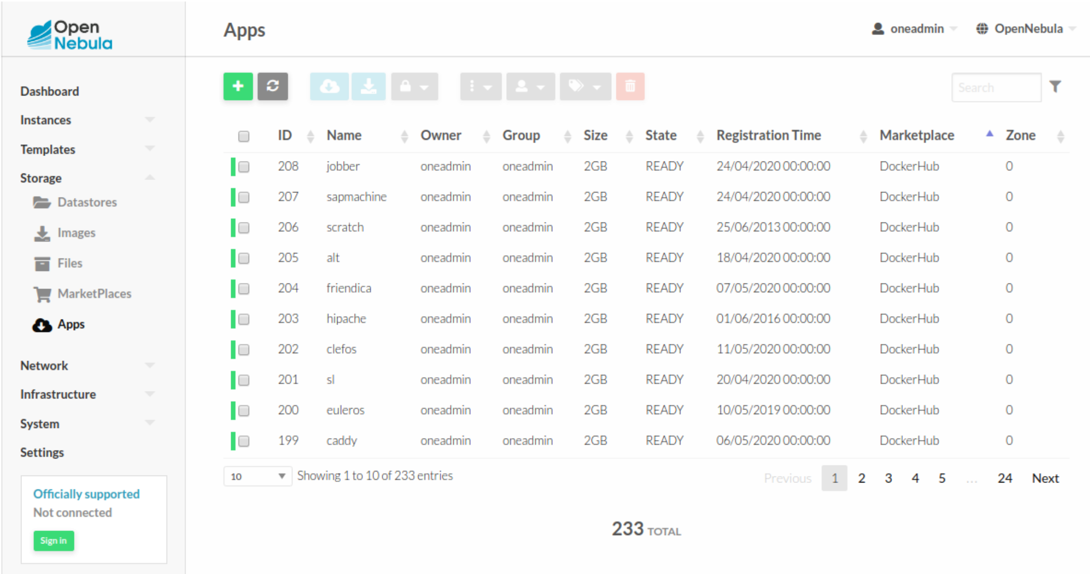


Figure 7. OpenNebula's native integration with the Docker Hub marketplace.

10. Ready for a Test Drive?

You can evaluate OpenNebula and build a cloud in just a few minutes by using **miniONE**,[11] our deployment tool for quickly installing an OpenNebula Front-end inside a Virtual Machine or a physical host, which you can then use to easily add remote Edge Clusters based on KVM, LXC or Firecracker.

11. Conclusions

It is clear that the evolution of the modern cloud has led to the creation of highly complex systems, often based on proprietary technologies. This document outlines OpenNebula's choice for simplicity and open source alternative, and presents a powerful Edge Cloud Architecture that is composed of Edge Clusters. They are built on demand using storage and networking technologies that already exist in the Linux operating system, and can run **any workload**—both virtual machines and application containers—**on any resource**—bare-metal or virtual—**anywhere**—on premise, on the cloud, or at the edge.

Our Edge Cloud Architecture enables true hybrid and multi-cloud computing by combining public and private cloud operations with workload portability and unified management of IT infrastructure and applications. Now you can enjoy a single vendor experience: we offer Enterprise support for the complete software stack through our OpenNebula **Software Subscription** and managed cloud services through a new OpenNebula **Managed Subscription**, so your team can forget about infrastructure and focus on business workloads. Contact us —we look forward to helping you at any stage of your cloud computing journey.

[11] https://minione.opennebula.io

LET US HELP YOU DESIGN, BUILD, AND OPERATE YOUR CLOUD

CONSULTING & ENGINEERING

Our experts will help you design, integrate, build, and operate an OpenNebula cloud infrastructure

OPENNEBULA SUBSCRIPTION

Get access to our Enterprise Edition and to our support and exclusive services for Corporate Users

MANAGED SERVICES

Our team of experts can fully manage and administer your OpenNebula cloud for you

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement ONEedge 880412

ONEedge is an OpenNebula project developing innovative features to bring private cloud computing to the Edge (ONEedge.io)

Sign up for updates at OpenNebula.io/getupdated

LINUX FOUNDATION SILVER MEMBER

CLOUD NATIVE COMPUTING FOUNDATION SILVER MEMBER

LF EDGE GENERAL MEMBER

GAIA-X data-infrastructure.eu

Rev2.2_20210507