Architecting price-performance private cloud

October 2021

Introduction

The success of general purpose private cloud infrastructure hinges primarily on economics. Well architected private cloud can be a cost-effective extension to the public cloud infrastructure, ensuring maximum multi-cloud cost optimisation. Leveraging the multi-cloud architecture enables organisations to always host their workloads where it makes most sense from an economical standpoint. This in turn ensures long-term total cost of ownership (TCO) reduction, leaving more budget for innovation and growth.

At the same time, making wrong decisions when architecting a private cloud may have a negative impact on the business. A poorly architected private cloud can lead to performance degradation, making it not suitable for the purpose. In turn, the efforts around performance optimisation, focused on adding more expensive components, can quickly lead to a lack of control over the budget. As a result, it may turn out that the private cloud being built is not cost-efficient at all and may not meet the desirable return on investment (ROI) goals for the organisation.

This is why architecting a price-performance private cloud is so important. Although private cloud implementation entails significant CapEx costs that businesses have to pay upfront, it can still be configured for the best price-performance, while ensuring the required capacity. Combined with lower license and subscription costs, and full automation of infrastructure deployment and post-deployment operations, the optimal private cloud architecture leads to maximum CapEx and OpEx efficiency. As a result, the surplus budget allows organisations to offer their services at a better price or to invest in research and innovation.

This whitepaper demonstrates Canonical's best practices for architecting price-performance private cloud. We show that making careful hardware choices results in highly performant infrastructure while ensuring minimum CapEx and OpEx spendings. The whitepaper follows with a case study from Canonical's internal cloud, including performance results, deployed according to those best practices. Finally, based on the hardware bill of materials (BOM) from this cloud, we perform price-performance analysis and demonstrate the cost per virtual machine (VM).

Ensuring price-performance when architecting private cloud

Architecting a price-performance private cloud starts at the hardware level. Although other costs (licenses, subscriptions, operations, etc.) have to be taken into account as well, they are usually vendor and customer dependent, while the cloud architecture is universal. Therefore, for our reference architecture we mainly focus on hardware selection and configuration guidelines to maximise private cloud performance while minimising hardware costs.

Overview

A typical infrastructure budget, shown in Fig. 1, consists of the four pillars: hardware and facilities, software licenses and services, internal operations and maintenance, and research and development (R&D). While the first three represent raw costs, the latter one allows businesses to explore new technologies, adopt them, transform and as a result, become more competitive in the market. Therefore, every organisation is trying to maximise its R&D and business transformation spendings.

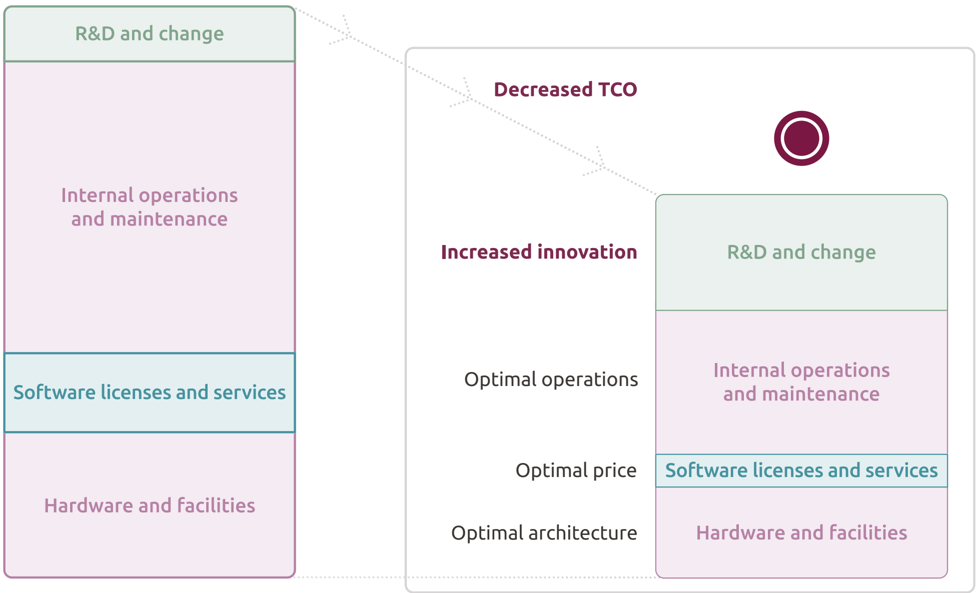

Fig. 1. Typical infrastructure budget.

On the other hand, it is essential to keep the budget under control. Both hardware, software and operations costs put pressure on the internal information technology (IT) teams. So in order to expand their R&D and change budget, they have to reduce other spendings. It is very important not to lower the quality of the platform, however, as poor infrastructure quality has a direct impact on the quality of the workloads running on top of it. This leads to poor quality of services offered by businesses, unhappy customers, claims, etc.

So, in order to lower the infrastructure TCO, while expanding the R&D and change budget, organisations have to design their infrastructure for an optimal architecture, optimal operations and optimal service cost. Optimal, i.e. the cheapest possible, not affecting the quality. The optimal service cost is straightforward. It results from choosing a vendor that applies the most economical and transparent pricing structure on the market.

Optimal operations result from choosing a platform which requires less interaction post-deployment (i.e. provides full operations automation). In turn, the optimal architecture results from applying proper methodology for hardware and software components selection, followed by their configuration.

Methodology

When it comes to making hardware and configuration choices, it usually starts with the selection of compute, network and storage resources. The challenge is to design an architecture that achieves the required capacity and satisfactory performance, while minimising pressure on the infrastructure budget. Although the capacity requirements can be addressed relatively easily by analysing the requirements of the workloads and designing the cloud for the required amount of central processing unit (CPU) threads, random-access memory (RAM) and storage, addressing performance requirements is not so trivial. While high-end hardware usually achieves the required performance results, it may ruin the budget. In turn, the low-end hardware may not achieve the company's performance goals.

But the relation of cloud performance to cloud price is not linear. Based on Canonical's experience and observations, it rather resembles a logarithmic curve, as shown in Fig. 2a. This is because the current capabilities of the hardware far exceed capabilities of the cloud software that is used to manage the virtualisation stack and provide software-defined networking (SDN) and software-defined storage (SDS) abstraction. So starting from some point using more powerful hardware does not really make sense as it does not improve the performance of the cloud any more, while it puts an unnecessary pressure on the infrastructure budget. This is why finding an optimal architecture that satisfies both is so important. In order to facilitate this process Canonical constantly benchmarks various cloud architectures and conducts price-performance analysis to find the optimal one.

Once the optimal architecture is determined, the next challenge is to find an optimal size of components that will be used to build the cloud. The size of components does not usually have an impact on the performance of the cloud. It may have an impact on the price of the cloud, however. Let's take storage as an example. The price of storage devices varies depending on their size, but the relationship between their price and size is not linear. It rather resembles a U-curve, as shown in Fig. 2b. Therefore, it is important to find the point where it becomes uneconomical to use disks of a higher size. On the other hand, using disks of a lower size requires more devices to fulfill the cloud capacity requirements. This adds to the cost and usually translates to more servers, again resulting in an increase in total hardware costs.

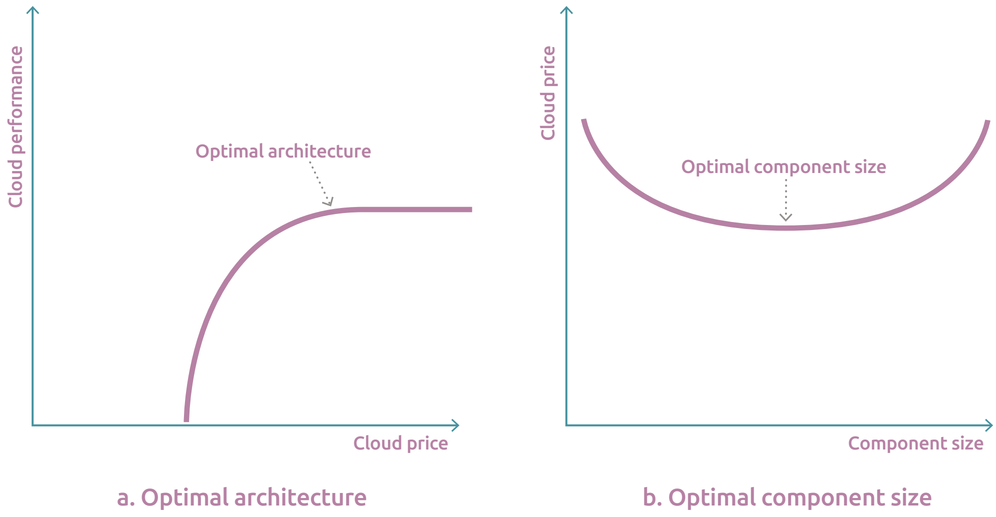

Fig. 2. Private cloud cost analysis.

The optimal component size is always in the middle of the curve. The same applies to other primary hardware components, i.e. RAM. Since the price of components is constantly evolving, such a process should be performed every time when architecting a new cloud. Canonical performs this process on a regular basis and constantly update its reference architecture for private cloud implementation to ensure maximum value for money based on the available hardware.

Additional considerations

In addition to following the methodology based on the U-curve, the following considerations have to be taken into account when architecting a price-performance private cloud.

Cloud architecture

The choice of a cloud architecture is usually the first decision organisations need to make when designing the cloud. It is also an important decision from the price-performance point of view. The architecture dictates how many nodes will be deployed and how cloud services will be distributed across those nodes. While running separate nodes for individual services usually results in a slightly better performance, it has a negative impact on the infrastructure budget. Canonical's proven approach to architecting price-performance private cloud is to use a hyper-converged architecture. This architecture assumes all nodes are equal and distributes compute, network, storage and control plane services evenly across the entire cluster. Such an approach ensures homogeneity and better resource utilisation, resulting in lower hardware costs compared to other architectures for private cloud implementation.

Server type

Once one is done with their choices of CPU, RAM and disks, choosing a server that will serve as a chassis for those resources is the next step. There are various types of servers available on the market and many different parameters should be taken into consideration. However, the most important parameter from the price-performance point of view is the form factor. Using too big servers results in more space occupied by those servers. This translates to more racks and increases the total hardware cost. In turn, too small servers may not have enough space to include a sufficient number of peripheral component interconnect express (PCIe) slots. Therefore, Canonical recommends using 2U servers for private cloud implementation. 2U servers ensure a balance between the total hardware cost and cloud performance. Such kinds of servers allow for higher-end CPUs, additional network interface cards (NICs), non-volatile memory express (NVMe) devices and other cards, while not putting significant pressure on the infrastructure budget.

Storage caching

When designing storage for a private cloud it is not sufficient to select an optimal disk size. This is because there are various types of disks: hard disk drives (HDDs), solid-state drives (SSDs, Intel Optane and various types of interfaces: serial-attached SCSI (SAS) and NVMe available. Their cost per TB varies by several orders of magnitude. Traditional SAS HDD disks are cheap, but their performance is limited. In turn, NVMe devices, such as SSDs and Intel Optane achieve much better performance results, but they are way more expensive. Implementing tiered storage is the most balanced solution in this case. While HDD disks provide the required storage capacity at low cost, NVMe devices are used to implement a caching layer in front of the actual storage. This caching layer does not drastically increase the total hardware costs, while significantly improving cloud performance.

Network topology

After addressing all potential bottlenecks on the machine level using storage caching technology, it is important to address them on the network level too. Canonical's reference architecture for private cloud implementation uses link aggregation control protocol (LACP) bonded links with a minimum speed of 10 Gbps everywhere to ensure high throughput between all nodes in the cloud. Moreover, it is based on two-layer leaf-spine network topology to minimise the number of hops for lower latency of network connections. Leaf-spine topology also maximises the throughput by performing packet forwarding based on Ethernet frames wherever possible, eliminating the overhead of internet protocol (IP) headers inspection. All of that results in a super fast network fabric across all nodes in the cloud and even more importantly, all VMs. Although the fast network increases the total hardware cost, it is a mandatory extension in performance-sensitive cloud environments. It is also not significant compared to the cost of other hardware components (i.e. servers).

Performance extensions

As the final step, one should consider additional performance extensions if required. Canonical supports all common performance extensions, including single root input/output virtualisation (SR-IOV), data plane development kit (DPDK), CPU pinning, non-uniform memory access (NUMA), hugepages and PCI passthrough, that are usually required in performance-sensitive environments. Since those extensions may put additional pressure on the infrastructure budget, it is recommended that they be raised as requirements during the design phase so that Canonical can assist with hardware choices.

Private cloud price-performance analysis

Designing the optimal architecture for private cloud implementation involves making careful hardware choices and finding a trade-off between its performance and price while ensuring the desired capacity of the cloud. However, organisations should also design their infrastructure for optimal operations and optimal price as shown in Fig. 1 in order to reduce the overall TCO associated with cloud infrastructure maintenance.

Ensuring maximum CapEx and OpEx efficiency

While choosing the optimal architecture ensures maximum efficiency of the CapEx cost, designing the cloud for optimal operations and optimal price helps to lower yearly OpEx cost. Although hardware choices are important and its price usually represents a significant part of the initial investment, organisations should not underestimate recurring costs associated with post-deployment maintenance of the cloud. Those include software licenses and services and internal operations. Choosing a platform which requires expensive licenses to be purchased and renewed or requires the entire team to operate it can quickly result in an inflating TCO. When making their choice, businesses should therefore take into account such issues as code openness, pricing structure transparency and operations automation. Optimising the above factors enables predictable budgeting and helps to avoid costly surprises post-deployment.

Open infrastructure stack

When it comes to the private cloud implementation, Canonical leverages the so-called open infrastructure stack. This stack, shown in Fig. 3, consists of open source technologies and covers all layers of the infrastructure, from bare metal to microservices. Canonical's approach is to leverage the best-of-breed technologies, let them do what they are best at, and offer full-stack support under a single subscription. In the heart of the open infrastructure stack is OpenStack. Being the most popular open source platform for private cloud implementation, OpenStack manages distributed compute, network and storage resources, and using virtualisation technologies enables allocation of them to VMs through a self-service portal. OpenStack can be easily extended with a containerisation layer, based on Kubernetes, running on top of it.

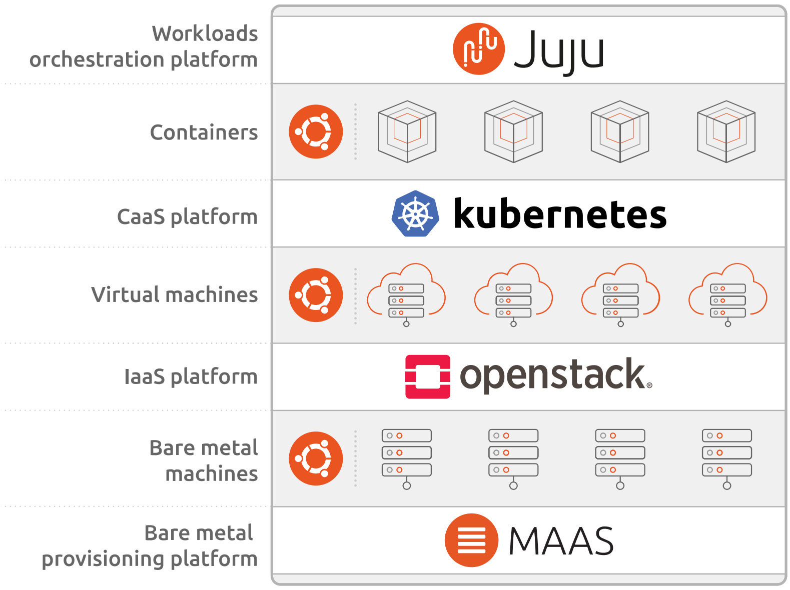

Fig. 3. Open infrastructure stack.

Four levels of commercial support for Canonical's open infrastructure stack:

Essential	Including security updates, hardening and Kernel Livepatch
Standard	Extending Essential with phone and ticket support during regular office hours
Advanced	Extending Standard with 24/7 support and more aggressive SLAs
Fully-managed	The entire stack is monitored, managed and operated by Canonical's team of cloud expert, including incident and problem resolution, upgrades and daily operations

Canonical's services for private cloud implementation

Since the adoption of open source technologies entails a number of challenges, Canonical provides a wide range of commercial services for enterprise customers. Those include consulting services, commercial support, and fully managed services. Using these services ensures organisations that they do not get stuck in OpenStack and Kubernetes deployment. It also lets them run the entire open infrastructure stack with confidence in production where aggressive availability goals and service level agreements (SLAs) have to be met. Sample Canonical's customers who leverage on commercial services from Canonical include MTS, BNP Paribas, Best Buy and Bloomberg.

Case study - Canonical’s internal cloud

In order to analyse the value for money of Canonical’s reference architecture for private cloud implementation, a sample cloud has been designed and implemented based on the presented guidelines.

Sample scenario

PCB and PCB Plus are Canonical’s fixed-price consultancy packages for private cloud implementation based on reference architecture and certified hardware.

They include cloud design and delivery, on-prem workshops, workload analysis and migration plans.

Both packages utilise Canonical’s toolset for private cloud implementation, including Metal-as-a-Service (MAAS), Charmed Operator Lifecycle Manager (OLM) and OpenStack Charms.

The cloud has been deployed using Canonical’s internal tools for private cloud implementation as a part of the [Private Cloud Build (PCB) service](). It has been deployed in a data centre, consisting of 37 cloud nodes and additional 3 infrastructure nodes for cloud automation and maintenance. In order to ensure maximum resource utilisation, hyper-converged architecture was selected. Hardware specifications from cloud and infrastructure nodes are shown in Tab. 1. Software specifications are shown in Tab. 2.

Cloud Nodes (37 x Gigabyte - R282-Z90)	
Form-factor	2U Rackmount, LFF Drive Cage
Processor	2 x AMD EPYC Rome 7502 (32c/64t 2.5GHz)
Memory	32 x 64GB 3200MHz DDR4 RDIMM (2048GB in total)
Storage	2 x 6.4TB U.2 NVMe (Samsung PM1725b) 6 x 12TB SAS 3.5” HDD 4 x Intel® Optane™ NVMe SSD 900P 480GB
Network	1 x Dual-port 100 GbE NIC (Mellanox ConnectX-5 Ex QSFP28) 1 x Dual-port 1 GbE NIC (Intel® I350-AM) 1 x BMC
Infrastructure Nodes (3 x HPE Gen10+ DL325)	
Form-factor	1U Rackmount, SFF Drive Cage
Processor	1x AMD EPYC 7302 (16c/32t 3.0GHz)
Memory	2 x 64GB 3200MHz DDR4 RDIMM
Storage	4 x 4TB SAS 2.5” SSD Hardware RAID Controller
Network	1 x 4-port 1 GbE NIC (Intel® i350-T4) 1 x BMC

Tab. 1. Hardware specifications from Canonical’s internal cloud.

Canonical’s internal cloud	
Operating System	Ubuntu 20.04 LTS
Kernel version	5.4
OpenStack version	Ussuri

Tab. 2. Software specifications from Canonical’s internal cloud.

Sample pricing

As the price of the same asset may vary from supplier to supplier, Canonical performed a bidding process to select the hardware that provides the best value for money. The desired cloud node specifications were distributed across various hardware vendors and the lowest bid that matched the requirements was selected. As a result, the hardware for this 40-node cloud was purchased by Canonical for around $1,200,000. The deployed cloud has been serving 74 AMD EPYC ROME 7502 CPUs, each with 32 cores, 74 TB of RAM, 229.4 TB of ephemeral storage and 2,664 TB of raw persistent storage, allowing for thousands of resourceful VMs.

Performance benchmarking

To measure the performance of the cloud several tests were run on the cloud using the flexible input/output tester (fio). During these tests, random read/write operations have been performed on the cloud using 200 client instances to ensure sufficient pressure on the storage. Each instance was using 8 vCPUs to ensure balanced distribution across physical CPUs and 2 persistent storage devices, each with a capacity of 200GB. To ensure that the total cache device capacity could be exceeded during tests, the total addressable storage was narrowed down to 80TB. Each test lasted for 1 hour with ramp up (and down) of client instances from 0 to 200 and back to 0 over a 15 minute period to avoid overloading all storage devices at the same time. To measure total input/output operations per second (IOPS) capacity, random 4KB data workloads were used. To measure total throughput capacity, sequential 4MB data workloads were used. All the tests were performed on a freshly deployed cloud and repeated multiple times to make sure that the results are consistent.

Benchmarking results

All collected metrics are shown in Tab. 3. Based on the presented data it is pretty clear that Canonical's internal cloud achieved similar performance results to those presented in other whitepapers about private cloud performance [1]. Since the same underlying storage platform was used in both cases and lab conditions were similar, this leads to a conclusion that Canonical's approach to architecting private clouds can satisfy performance requirements, while keeping control over budget at the same time.

4KB random workload		
Metric	Write	Read
Average throughput [KIOPS]	450	10240
Average 99%th latency [ms]	38	2
Average latency [ms]	30	1
4MB sequential workload		
Metric	Write	Read
Average throughput [GB/s]	12	258

Tab. 3. Performance results from Canonical's internal cloud.

TCO analysis

Although there are many factors to consider when selecting a platform for private cloud implementation, the cost per VM is usually the deciding one. Various platforms are based on different architectures, use different pricing structures for software licenses and services and have different requirements with regards to the platform post-deployment operations. As a result, the TCO associated with cloud infrastructure maintenance varies across different platforms of the same size. At the same time, they allow for the same capacity. Therefore, the cost per VM varies depending on the platform too.

Number of vCPUs	Amount of RAM [GB]	Amount of ephemeral storage [GB]	Amount of persistent storage [GB]	Theoretical maximum number of VMs	Hourly cost per VM [USD]
1	4	8	40	11544	0.0092
2	8	8	80	5772	0.0184
4	16	8	160	2886	0.0369
8	32	8	320	1443	0.0738
16	64	8	640	721	0.1477

Tab. 4. Cost per VM metrics from Canonical's internal cloud.

Tab. 4 shows the hourly cost per VM metrics from Canonical's internal cloud. These estimates cover all necessary costs associated with private cloud deployment and operations, including hardware costs (racks, servers, switches, cabling, etc.), hosting services (rent, electricity, network and hardware maintenance), cloud delivery and annual subscriptions, assuming fully-managed services from Canonical. Moreover, when making those calculations we assumed a 3-year hardware renewal period, 2:1 CPU overcommitment ratio, no RAM overcommitment, the default Ceph replication factor of 3 and cloud utilisation at 75%.

Conclusions

Although architecting price-performance is a non-trivial task, following certain guidelines and best practices helps to achieve maximum TCO reduction, while not affecting the quality of the infrastructure. Since private cloud implementation requires significant investments upfront due to a high total hardware cost, designing an optimal architecture is the first step organisations should focus on. Among various considerations, following certain methodology when making choices of cloud resources and applying storage caching technology is Canonical's proven method to achieve high performance of the cloud, while not putting pressure on the infrastructure budget.

In this whitepaper, we presented how Canonical's reference architecture for private cloud implementation answers the need of price-performance. Based on the data presented in the "TCO analysis" section it is clear that the cost per VM in Canonical's internal cloud is lower compared to what leading public and private cloud providers offer. We also highlighted that choosing a platform which is based on open source technologies and implements transparent pricing structure and fully automated operations helps to keep the recurring costs down.

Next steps

Get in touch with your Canonical's sales representative for the next steps on private cloud implementation. Canonical maintains a reference architecture and reference hardware documents which outline detailed recommendations for price-performance private cloud implementation. Following those recommendations is highly recommended as wrong decisions during the design phase may have a negative impact on the performance of your cloud and your infrastructure budget.

If you are not in touch with Canonical yet, you can reach out to us at https://ubuntu.com/openstack/contact-us

1. https://www.samsung.com/semiconductor/global.semi/file/resource/2020/05/redhat-ceph-whitepaper-0521.pdf

ubuntu
Delivered by Canonical