VIRTUALIZATION – CONCEPT AND DEVELOPMENT. VIRTUALIZATION ENVIRONMENT AND TOOLS

Borislav Stoyanov

Trakia University – Stara Zagora, Bulgaria

Abstract. In this paper I review virtualization and provide a comprehensive overview of it, as it's key technology that enhances resource utilization, isolation, flexibility, and scalability in modern IT infrastructures. We explore the concept and development of virtualization, highlighting its evolution from mainframe environments to contemporary cloud-based solutions and it's progress over the years. The discussion includes various types of virtualization, such as hardware, operating system, storage, and network virtualization, emphasizing their significance in optimizing resource usage and improving system performance. Additionally, we examine a range of popular virtualization platforms and tools, including Docker, Kubernetes, VMware, Microsoft Hyper-V, Xen, Amazon EC2, Google Compute Engine, OpenStack, QEMU, VirtualBox, Parallels Desktop, Hyper-V Manager, VMRC, vSphere, vCenter Server, NSX, Azure Stack, Azure Government, AWS services like S3, RDS, Lambda, ECS, and Linux distributions such as RHEL, CentOS, and Debian. Each platform's capabilities, typical use cases, and benefits are discussed, providing a clear understanding of how these tools contribute to efficient and scalable IT operations.

1. Introduction

Virtualization is a process that allows for more efficient use of physical computer hardware and is the foundation of cloud computing.

Virtualization uses software to create an abstraction layer over computer hardware, enabling the division of a single computer's hardware components—such as processors, memory and storage—into multiple **virtual machines** (**VMs**). Each **VM** runs its own operating system (**OS**) and behaves like an independent computer, even though it is running on just a portion of the actual underlying computer hardware

It follows that **virtualization** enables more efficient use of physical computer hardware and allows a greater return on an organization's hardware investment.

Today, **virtualization** is a standard practice in enterprise IT architecture. It is also the technology that drives cloud computing economics. It enables cloud providers to serve users with their existing physical computer hardware. It enables cloud users to purchase only the computing resources they need when they need it, and to scale those resources cost-effectively as their workloads grow.

2. The History of Virtualization

While invented in the 1960s, **virtualization** technology didn't become popular until the early 2000s. Back then, the tech (like **hypervisors**) aimed to give multiple users access to computers for batch processing tasks (like running payroll).

However, other solutions for sharing a single machine gained more traction, like time-sharing within operating systems. This eventually led to the development of **UNIX** and later **Linux**. **Virtualization** remained on the sidelines for many years.

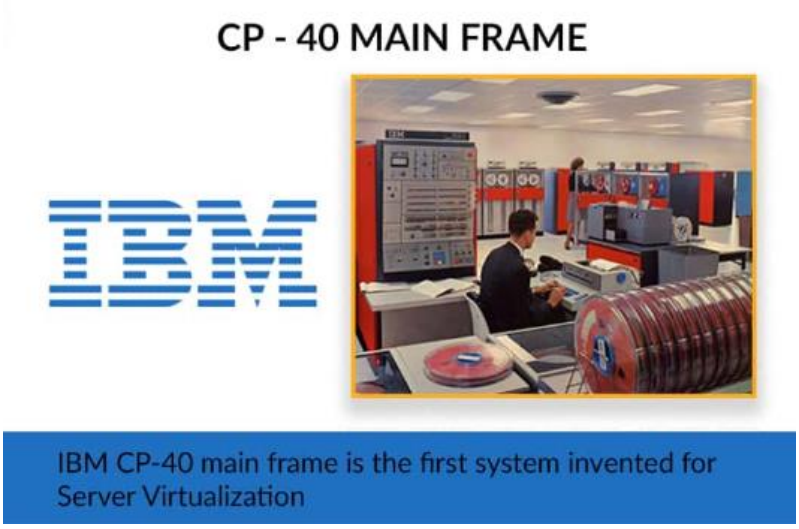
Fast forward to the 1990s. Companies relied on single-vendor IT systems, making it difficult to run older applications on hardware from different vendors. As businesses switched to cheaper, generic servers and software from various vendors, they ended up with underutilized physical servers – each server could only handle a specific task from a specific vendor.

This is when **virtualization** came to the rescue. It allowed companies to partition their servers and run older applications on different operating systems. Servers became more efficient, reducing costs associated with buying, setting up, maintaining, and cooling them.

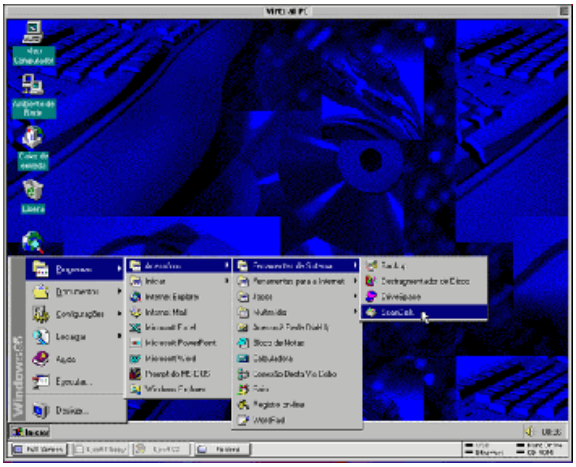
Virtualization's flexibility helped businesses avoid being locked into using a single vendor's products. It also became the foundation for cloud computing. Today, **virtualization** is so widely used that companies often need specialized software just to manage it all.

2.1 Timeline of virtualization

- **1967 – CP-40** goes into production use in January 1967. Shortly after a re-implemented version of it, which we know as CP-67 goes into production use (April).

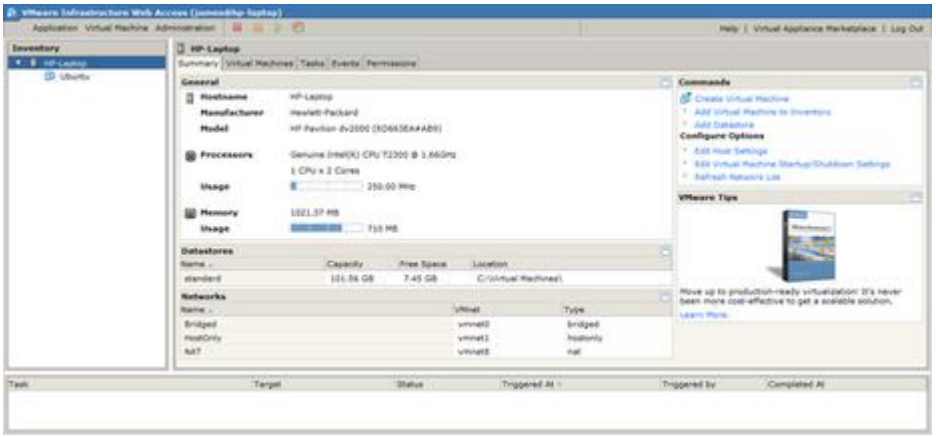


- **1987 – Merge/386** made available to **OEMs**, which supports multiple simultaneous **virtual 8086 machines**. In October it begins shipping, offered with **Microport Unix System V Release 3**.
- **1994-** Kevin Lawton leaves MIT Lincoln Lab and starts the **Bochs** project.
- **1997** - First version of **Virtual PC** for **Macintosh** platform releases in June by Connectix.



- **1999 -** On **February 8th VMware** introduced the first **x86** virtualization product for the **Intel IA-32** architecture, known as **VMware Virtual Platform**, based on earlier research by its founders at Stanford University. **VMware Virtual Platform** was based on **software emulation** with **Guest/Host OS design** that required all **Guest environments** be stored as files under the host OS filesystem.
- **2003 -** First release of first **open-source x86 hypervisor**, **Xen**.

  February 18th – **Microsoft** acquires virtualization technologies from **Connectix Corporation**.
- **2006 –** July 12th **VMware** releases **VMware Server,** a free machine-level virtualization product for the server market.



Microsoft Virtual PC 2006 is released as a free program, also in July.

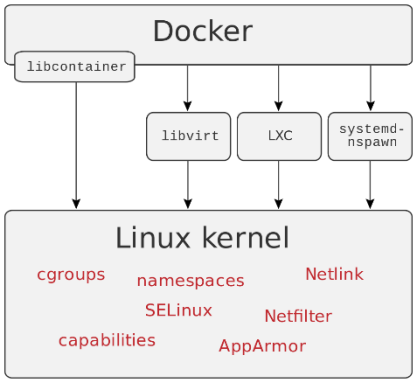
July 17th **Microsoft** bought **Softricity**.

September 26th **moka5** delivers **LivePC** technology.

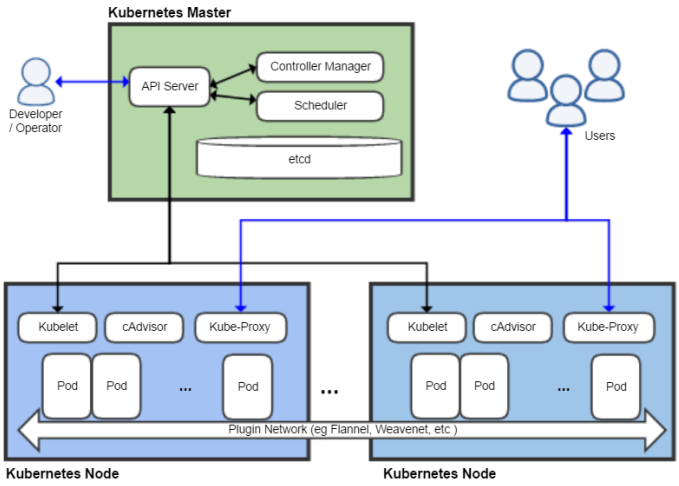
HP releases **Integrity Virtual Machines Version 2.0**, which supports **Windows Server 2003**, **CD** and **DVD** burners, tape drives and **VLAN**.

December 11th **Virtual Iron** releases **Virtual Iron 3.1**, a free bare-metal **virtualization** product for enterprise server **virtualization** market.

- **2007 January 15th - InnoTek** releases **VirtualBox Open Source Edition** (**OSE**), the first professional PC virtualization solution released as open source under the **GNU General Public License** (**GPL**).
- **2008 June 28th** – **HyperV** launched by Microsoft
- **2013 - Docker, Inc**. releases **Docker**, a series of platform as **a service (PaaS)** products that use **OS-level virtualization**.



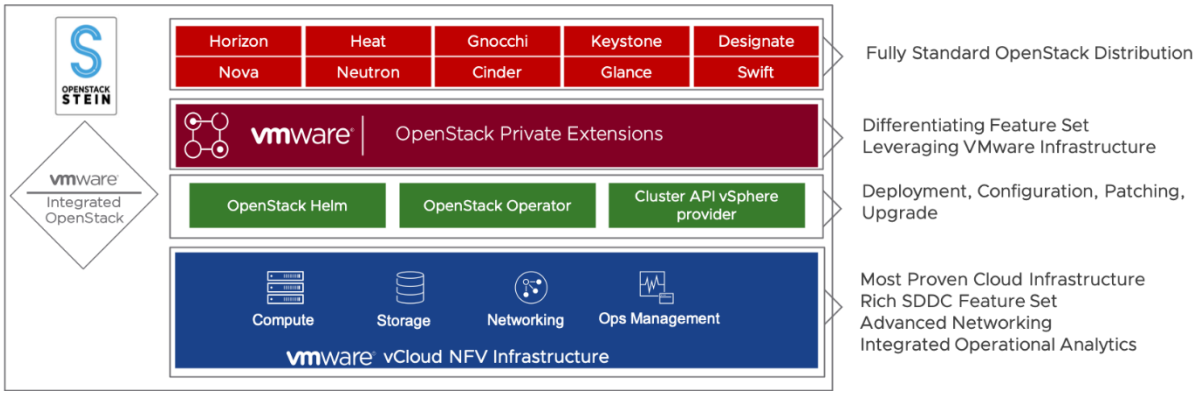
- **2014 –** Sep 8th the first public build of **Kubernetes** is released, initially designed by **Google**.



VMware releases **vSphere 6,** introducing features like **vMotion** enhancements and **Virtual Volumes (vVols)**

- **2015 - Docker** releases **Docker Engine 1.6** with support for **multi-host networking**, enhancing container orchestration capabilities

  Introduction of **VMware Integrated OpenStack** (**VIO**) for deploying and managing **OpenStack** clouds on **VMware** infrastructure.



- **2016 - Microsoft** releases **Windows Server 2016** with built-in support for **Docker** containers and **Hyper-V** enhancements.
- **2021 - VMware** introduces **VMware Anywhere Workspace**, integrating endpoint management, security, and collaboration tools.

- **2022 - VMware** introduces **VMware vSphere 7 Update 3**, focusing on security enhancements and **Kubernetes** integration.

  Expansion of edge computing deployments leads to increased use of virtualization for managing distributed computing resources.

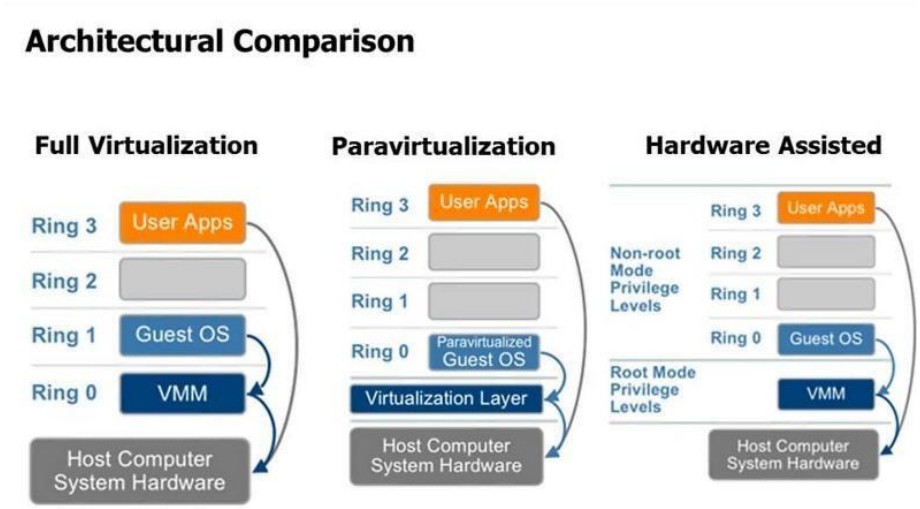
3. Basic Terms and concepts

In this section, I've written about the basic terms and concepts that define **virtualization**. We'll start with a basic explanation of **virtualization** itself, including its benefits and its different types. Next, we'll discuss some popular tools and platforms, along with commonly utilized **Linux** distributions tailored for **virtualized** settings.

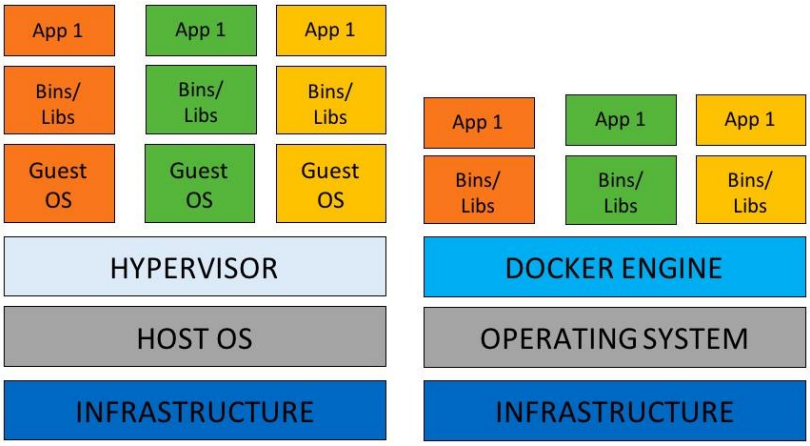
- **Virtualization: Virtualization** is a technique that allows one **physical machine** to be divided into multiple **logical** or **virtual machines**. Each **virtual machine** operates

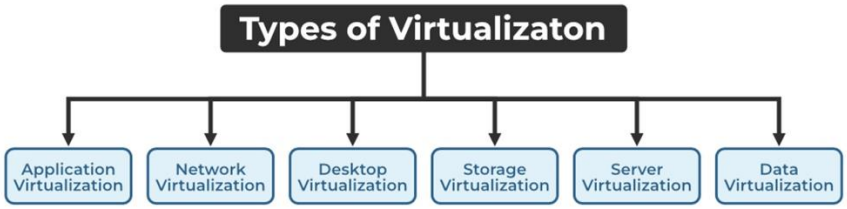
independently of others and can run its own **operating system** and **applications**. There are several types of **virtualization**, including **full virtualization, paravirtualization, containerization, application virtualization, network virtualization, desktop virtualization, storage virtualization, server virtualization and data virtualization**.

- **Full Virtualization: Full virtualization**, also known as **hardware virtualization**, is the traditional type of **virtualization** that creates a **virtual machine** that is completely independent of **the physical hardware infrastructure**. **Virtualization** software abstracts the hardware from the operating system and applications, allowing them to operate as if they have direct access to the **physical resources**.
    - **Paravirtualization**: **Paravirtualization** is a type of **virtualization** that works by creating a special layer between the **operating system** and the **physical hardware**. This layer, known as the **paravirtualization layer**, allows the operating system to interact directly with it instead of the physical hardware. Paravirtualization can offer improved performance over **full virtualization** because it eliminates the extra layer of abstraction.



- **Containerization:** A **virtualization** method that encapsulates applications and their dependencies in a **container** that can run anywhere there is a **compatible runtime environment**. **Containers** share the host operating system, making them lighter and more efficient than **full virtual machines**.

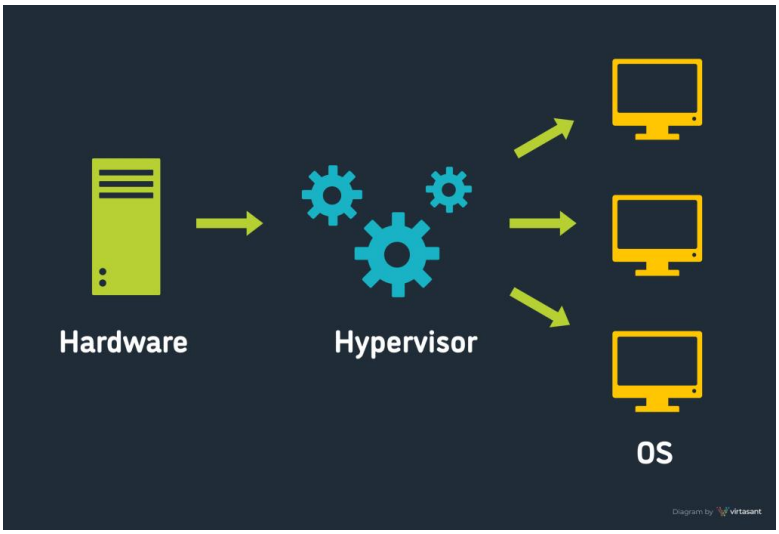




- **Application Virtualization** : Application virtualization helps a user to have remote access to an application from a server. The server stores all personal information and other characteristics of the application but can still run on a local workstation through the internet.
- **Network Virtualization** : The ability to run multiple virtual networks with each having a separate control and data plan. It co-exists together on top of one physical network. It can be managed by individual parties that are potentially confidential to each other. Network virtualization provides a facility to create and provision virtual networks, logical switches, routers, firewalls, load balancers, Virtual Private Networks (VPN), and workload security within days or even weeks.
- **Desktop Virtualization** : Desktop virtualization allows the users' OS to be remotely stored on a server in the data center. It allows the user to access their desktop virtually, from any location by a different machine. Users who want specific operating systems other than Windows Server will need to have a virtual desktop. The main benefits of desktop virtualization are user mobility, portability, and easy management of software installation, updates, and patches.

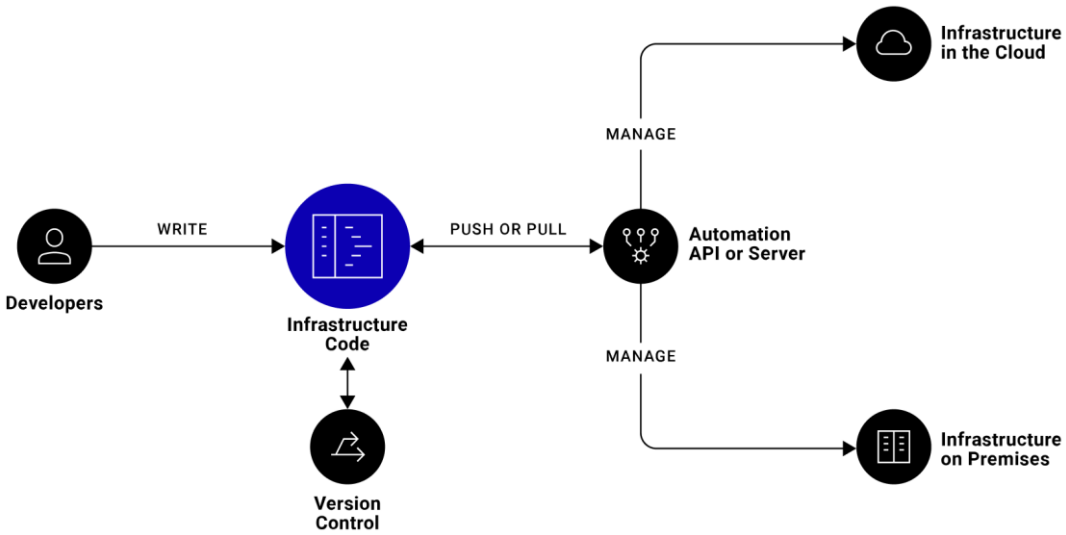
- **Storage Virtualization** : Storage virtualization is an array of servers that are managed by a virtual storage system. The servers aren't aware of exactly where their data is stored and instead function more like worker bees in a hive. It makes managing storage from multiple sources be managed and utilized as a single repository. storage virtualization software maintains smooth operations, consistent performance, and a continuous suite of advanced functions despite changes, breaks down, and differences in the underlying equipment.
- **Server Virtualization** : This is a kind of virtualization in which the masking of server resources takes place. Here, the central server (physical server) is divided into multiple different virtual servers by changing the identity number, and processors. So, each system can operate its operating systems in an isolated manner. Where each sub-server knows the identity of the central server. It causes an increase in performance and reduces the operating cost by the deployment of main server resources into a sub-server resource.
- **Data Virtualization** : This is the kind of virtualization in which the data is collected from various sources and managed at a single place without knowing more about the technical information like how data is collected, stored & formatted then arranged that data logically so that its virtual view can be accessed by its interested people and stakeholders, and users through the various cloud services remotely.

- **Hypervisor**: A **hypervisor** is a software layer that creates a virtual machine that functions as a separate computer system on a host computer. A **hypervisor** allows multiple operating systems and applications to run simultaneously on a single physical machine, effectively increasing resource utilization and enabling better management.



- **Guest operating system:** The **guest operating system** is the **operating system** that runs in the **virtual machine**.
- **Host operating system:** The **host operating system** is the **operating system** on which the **hypervisor** and **virtual machines** run.
- **Virtual Machine Isolation**: **Virtual machine isolation** ensures that each **virtual machine** operates independently of the others, preventing conflicts and improving security.
- **Resource management**: **Resource management** involves the allocation of system resources such as **CPU**, **memory**, and **storage** among **virtual machines**. This helps optimize resource usage and ensure that each **virtual machine** has enough resources to run efficiently.
- **Migration: Migration** involves moving a running virtual machine from one **host** to another. This can help improve **resource efficiency** by ensuring that virtual machines are running on the best possible **hardware configurations.**
- **Backup: Backup** involves the creation of a **virtual machine** recovery point that can be used to restore the system in case of damage or data loss. This is an important part of the disaster and recovery strategy for any virtual environment.
- **Patches: Patches** are **updates** that fix **bugs** or **vulnerabilities** in the **operating system** or **software**. It is important to apply **patches** regularly to maintain the security and stability of your **virtual environment.**
- **Cloud virtualization: Cloud virtualization** involves using **virtualization** in a **cloud environment** where **virtual machines** are created, managed, and delivered as services to users. This allows organizations to take advantage of the flexibility, scalability, and economies of scale of **cloud computing** while leveraging **virtualization** to further improve efficiency and control.
- **Data Center:** A **data center** is a facility that contains a large number of **servers** and other network equipment used to store, process and distribute **data**. **Data centers** play a crucial role in modern information technology, as they provide the necessary infrastructure to **host** and **manage virtual machines** and other services.
- **Telecommunications: Telecommunications** refers to the transmission of information over **communication channels**, such as telephone lines, fiber optic cables, and wireless networks. In the context of **virtualization**, **telecommunications** can be used to connect **virtual machines** to a **network** or to provide access to the **Internet** and other external resources.

- **Infrastructure as Code (IaC): IaC** is an infrastructure management approach where infrastructure configurations are defined as code that can be managed using versioning and source control. This enables more efficient and consistent deployment and management of virtual machines and other infrastructure components.

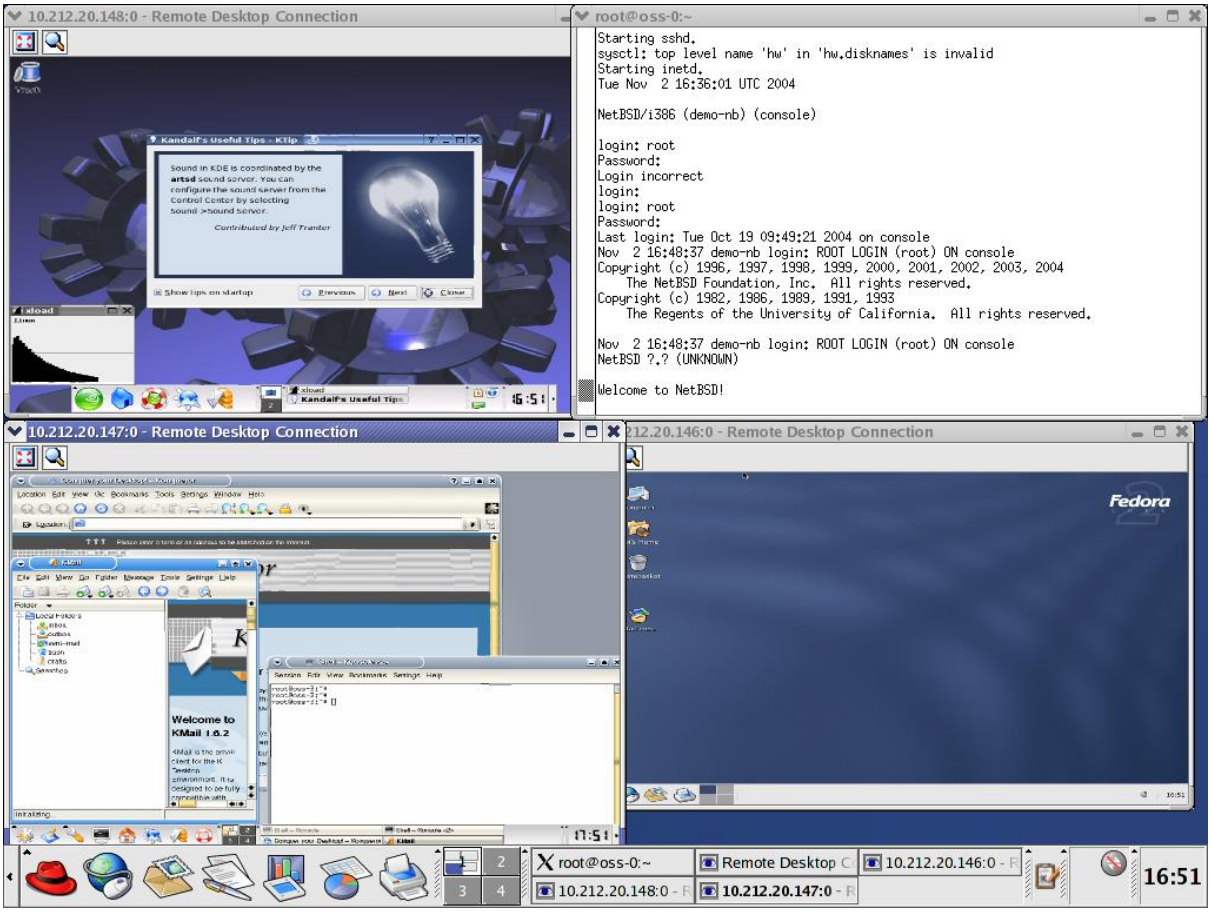


3.1 Platforms and tools

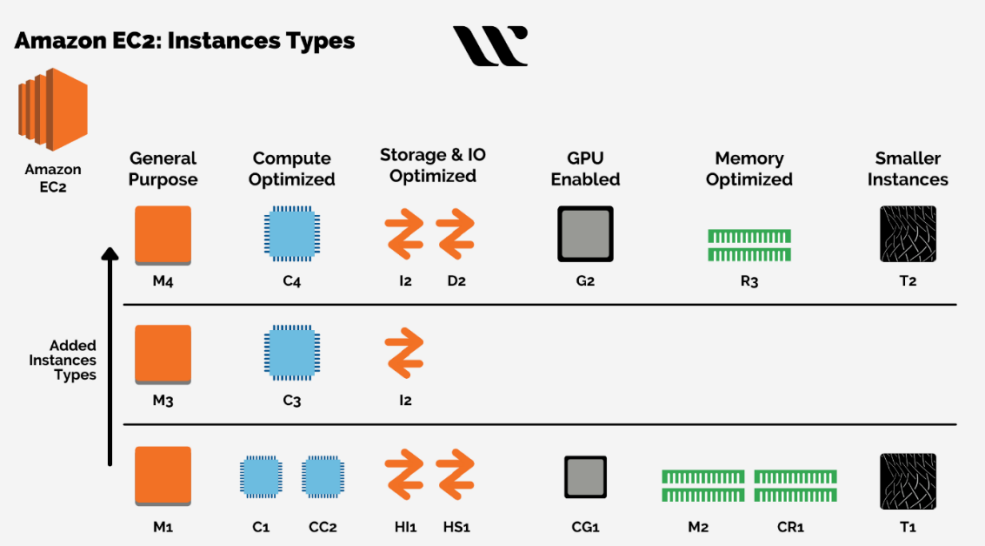
Virtualization platforms are essential in modern **computing environments** for optimizing resource utilization by allowing multiple **virtual machines** to run on the same **physical hardware**, enhancing **isolation** between **applications** and **operating systems**, ensuring **flexibility** and **scalability** through **rapid provisioning** and **deployment**, providing a safe environment for **testing** and **development** activities, consolidating infrastructure to achieve cost savings, enabling high availability and fault tolerance features, and maintaining compatibility with **legacy systems**. These **platforms** facilitate efficient use of **hardware resources**, improve security and stability, reduce **operational costs**, and support business continuity efforts, making them indispensable tools for organizations managing complex **IT infrastructures**.

These are some of the popular **virtualization platforms** that organizations commonly rely on for managing their **IT infrastructures**:

- **Docker: Docker** is an **open-source containerization platform** that automates the deployment, scaling, and management of applications. It is often used for creating and managing **containerized** applications.
- **Kubernetes: Kubernetes** is an **open-source** orchestration platform that is frequently utilized for creating and managing **clusters** of **containerized** applications similarly to **Docker**.
- **VMware: VMware** is a company that provides **virtualization solutions**, including **VMware ESXi**, a **hypervisor** widely used in **data centers**. **VMware** offers various products and services related to **virtualization**, including **VMware vCenter Server** and **VMware NSX**.
- **Microsoft Hyper-V: Microsoft Hyper-V** is a **hypervisor** included in **Windows Server**, enabling the creation and management of **virtual machines**. It is commonly used in **Windows** environments for **virtualization**.
- **Xen: Xen** is an open-source **hypervisor** widely used in **data centers** and other large organizations. It is often employed in open-source environments for virtualization.

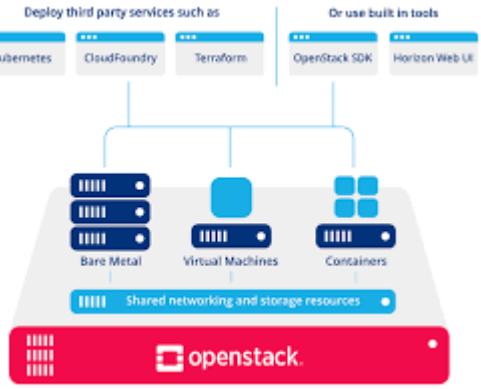


- **Amazon EC2: Amazon EC2** is an **Infrastructure as a Service** (**IaaS**) offering that allows users to create and manage **virtual machines** in the **cloud**. It is frequently utilized by organizations seeking to leverage the flexibility and scalability of **cloud computing** while using **virtualization** to further enhance efficiency and control.

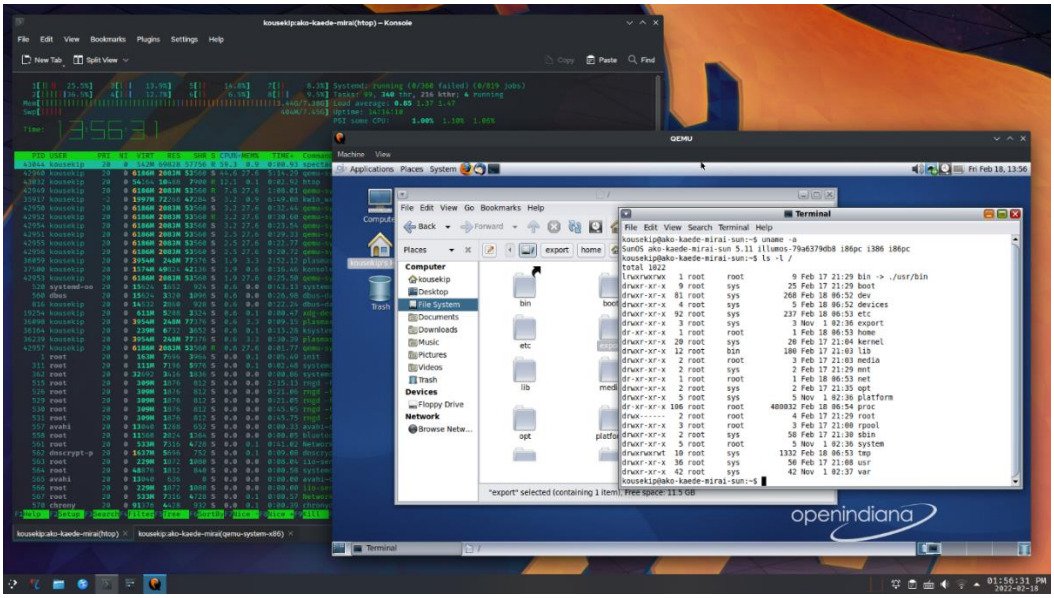


- **Google Compute Engine: Google Compute Engine (GCE)** is an **IaaS service** that enables users to create and manage **virtual machines** in the **cloud** , much like **Amazon EC2**.

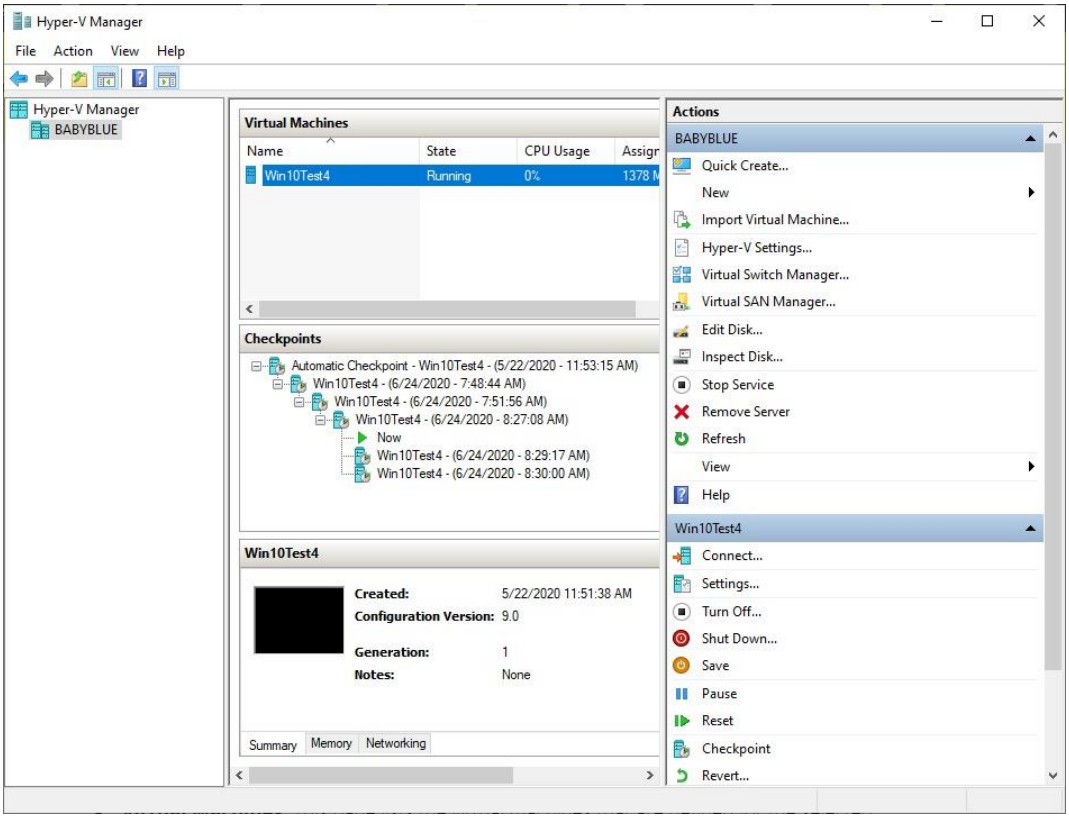
- **OpenStack: OpenStack** is an **open source platform** that uses **pooled virtual resources** to build and manage **private** and **public clouds**. The **tools** that **comprise** the **OpenStack** platform, called "**projects**," handle the core **cloud-computing** services of **compute**, **networking**, **storage**, **identity**, and **image services**.



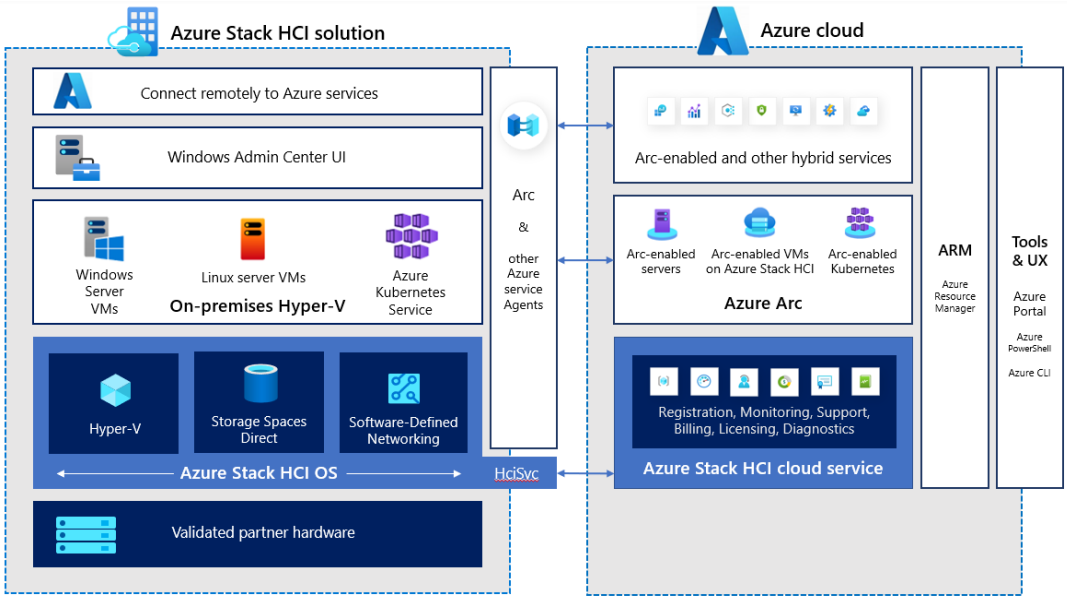
- **QEMU: QEMU** (**Quick Emulator**) is a **free** and **open-source emulator**. It **emulates** a **computer's processor** through **dynamic binary translation** and provides a set of different **hardware** and **device models** for the machine, enabling it to run a variety of guest operating systems.



- **VirtualBox:** VirtualBox is **open-source virtualization** software that allows the creation and management of virtual machines on **personal computers**. It is commonly used by individual users and small businesses for **virtualization**.
- **Parallels Desktop: Parallels Desktop** for **Mac** is software providing hardware virtualization for **Macintosh computers** with **Intel processors**, and since version 16.5 also for **Apple silicon-based Macintosh computers**.
- **Hyper-V Manager: The Hyper-V Manager console** in **Windows Server** can be used to **create VMs** on a **Hyper-V host**. It can also be used to launch **console sessions** with **existing VMs** and perform many other management tasks.



- **VMRC: VMRC (Virtual Machine Remote Console)** is a **remote management tool** included in **VMware ESXi** that allows **administrators** to interact with **virtual machines** remotely.
- **vSphere:** The **vSphere Client** acts as an **administrative interface** to access **VMware hosts**. These **hosts**, which run **VMware ESXi**, are the **servers** that run individual **virtual machines (VMs).** The **vSphere Client** attaches to the host servers. By using this interface, **admins** can create and **manage VMs**, as well as **host resources**.
- **vCenter Server: vCenter Server** is a **centralized management tool** for the **vSphere platform**, allowing **administrators** to monitor, manage, and maintain their **data centers**.
- **NSX: NSX (Network Virtualization and Security)** is **VMware**'s **network virtualization platform**, providing **network virtualization**, security, and management.
- **Azure Stack: Azure Stack** is a **hybrid cloud platform** developed by **Microsoft**, enabling organizations to extend their **on-premises data centers** to the **Azure public cloud**.



- **Azure Government: Azure Government** is a version of the **Azure** service specifically designed for government agencies, meeting the security and compliance requirements of the US government.
- **AWS (Amazon Web Services): AWS** is a **cloud computing platform** that provides a wide range of services, including **compute, storage, databases**, and **networking services**.

- **S3 (Simple Storage Service):** S3 is an object storage service provided by AWS, allowing users to store and retrieve data from a scalable and highly available array.
- **RDS (Relational Database Service):** RDS is a managed database service provided by AWS, allowing users to create, manage, and scale relational databases in the cloud.
- **Lambda:** Lambda is a serverless computing service provided by AWS, allowing users to run code without managing infrastructure.

- **ECS (Elastic Container Service): ECS** is a **container orchestration service** provided by **AWS**, allowing users to create, manage, and scale containerized applications in the cloud.
- **Red Hat Enterprise Linux (RHEL): RHEL** is a **commercial distribution** of **Linux** widely used in **data centers** and other large organizations.
- **CentOS: CentOS** is a free and open-source distribution of **Linux** based on **RHEL** and often used as an alternative to **RHEL**.
- **Debian: Debian** is a free and open-source distribution of **Linux**, one of the oldest and most widely used distributions.

4. Conclusion

In conclusion, the exploration of **virtualization** has revealed its significance as a transformative concept in the realm of **computing**.

The discussion on **virtualization** environments and tools has shown us the diverse array of options available to practitioners, each catering to specific needs and objectives. From **hypervisors** to **containerization platforms**, the landscape is rich with solutions that empower organizations to optimize resource utilization, enhance scalability, and streamline operations.

As we navigate the complexities of an increasingly digital world, **virtualization** stands as a cornerstone technology, offering a pathway to innovation and agility. By embracing its principles and leveraging its capabilities, organizations can unlock new frontiers of possibility, driving productivity, innovation, and competitiveness in the digital age.

In essence, **virtualization** represents not merely a tool or technology, but a paradigm shift—an evolution in our approach to **computing** that promises to reshape the fabric of **IT infrastructure** and redefine the boundaries of possibility.

References

https://www.ibm.com/topics/virtualization

https://www.redhat.com/en/topics/virtualization/what-is-virtualization

https://www.vmware.com/topics/glossary/content/hypervisor.html

https://www.novabackup.com/blog/virtualization-glossary-terms-know

https://www.redhat.com/en/topics/linux

https://www.debian.org

https://aws.amazon.com

https://www.britannica.com/technology/UNIX

https://www.vm.ibm.com/history/

https://aws.amazon.com/what-is/virtualization/

https://www.geeksforgeeks.org/virtualization-cloud-computing-types/