Research Assignment

Katleho Mafalela

215045849

Computer Science 3B

# Introduction

To understand virtualization, it is best to first understand what a virtual machine (VM) is. A virtual machine can be defined as a layer of abstraction that works as a communication bridge between hardware and the end-user. A computer’s operating system can accommodate multiple VM’s and share its hardware resources (CPU, memory, I/O devices etc) and software resources (Operating system and software libraries) (Daniels 2009).

Now that a definition for a virtual machine has been established, it is a lot simpler to understand what virtualization is. According to IBM, virtualization can be defined as a process that makes it possible to utilize the computer hardware and software resources and makes cloud computing possible through virtualization layers (IBM cloud education 2019). These layers are application level, library level, operating system level, hardware abstraction layer and Instruction set architecture.

The process of executing an application on the device’s operating system with installing it on the device is how application virtualization takes place (IBM cloud education 2019). To achieve this the virtualization layer is placed on top of the host’s operating system as an application. Examples of the application layer are the Java virtual machine (JVM) and .Net CLR.

Another layer of virtualization is the library level virtualization. In this form of virtualization, the application uses APIs found in the user level libraries. The application and the system share communications through API hooks, and this is how library virtualization is achievable. Tools such as WINE and LxRun are used to implement library level virtualization (Taylor 2010).

The use of containers is how operating system level virtualization is achieved. In this layer, virtualization engines and containers are allowed to operate and run their functionalities in the kernel of the host operating system. Tools such as Docker and Jail are commonly used to implement operating system level virtualization (Taylor 2010).

Next, we have the hardware abstraction layer (HAL) virtualization. HAL basically connects the software and hardware of the device (Levi 2014). With HAL virtualization software can be executed on a hardware that was not designed for it. This is done by hiding the processes of

the software and hardware from each other. HAL is commonly used in embedded systems (Mc gee).

To execute legacy code that is not compatible with the set up of the hardware can be tricky. An example would be to run a binary code in a x86 system. However, tools such as QEMU and Dynamo that make use of Instruction set architecture (ISA) which is another virtualization layer make it easier to do so (Taylor 2010).

# Virtualization

According to IBM, virtualization can be defined as a process that makes it possible to utilize the computer hardware and software resources and makes cloud computing possible through virtualization layers (IBM cloud education 2019). In the field of IT and Computer science virtualization has become common practice, this is largely due to the potential in solving many of the problems faced today.

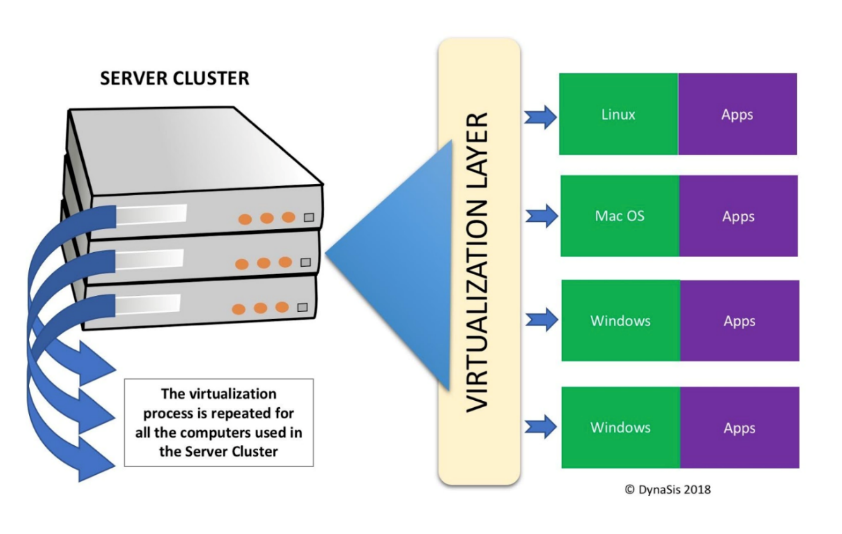


Figure : Virtualisation (dynasis.com 2008)

## 2.1 Benefits of virtualization

In network file systems, virtualisation will provide an architectural benefit. This is because virtual machines store image information as a file. Another example will be reducing the time needed for administrative duties and management of resources on virtual machine files compared to having to allocate and place files on logical units individually (Orenstein 2008, page 14).

Virtualization also has application in data storage. Consider a user whose computer tells him he has 1TB of storage he can use. It possible that storage does not physically exist on his computer but rather it is distributed over a network, or it made up of multiple different disks. It could even be divided between cache, magnetic and optical disk tapes (Matlis 2000, page 7).

According to H. Forbes in a paper he released in 2014, he theorises that virtualisation in multiprocessors could potentially improve the capacity and performance of a process controller. Also, virtualization could be applied on controllers that use real time operating systems and rich operating systems when running collaborative process automation system applications in greater application integration. These CPAS application currently run-on servers only.



Figure Benefits of virtualisation (rutter-net.com)

## 2.2 Levels of virtualisation

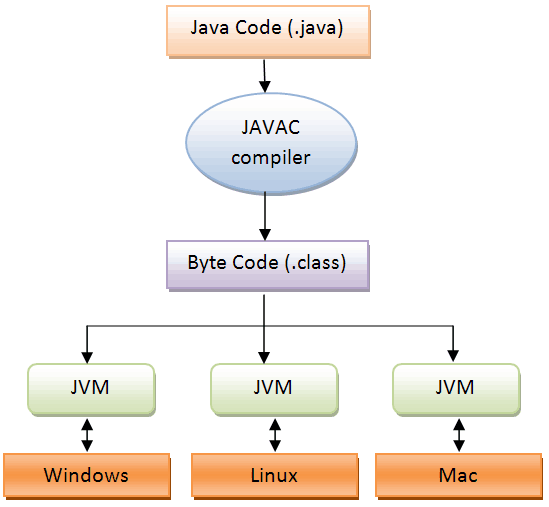
How virtualisation is implemented can divided in two five different levels of abstraction.

### 2.2.1 Application level

The process of executing an application on the device’s operating system without installing it on the device is how application virtualization takes place (IBM cloud education 2019). To achieve this, the virtualization layer is placed on top of the host’s operating system as an application.

There are a lot of application virtualisation examples. However, in this paper the focus is on the Java virtual machine and the .NET CLR.

**Java virtual machine**

******

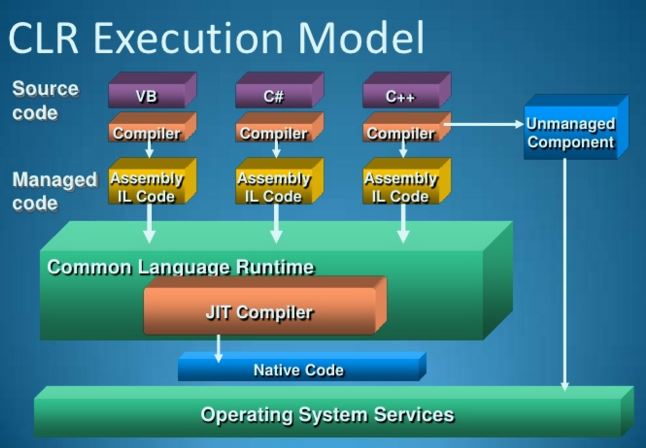
**Figure 3: Java virtual machine (Viralplanet.net)**

The java virtual machine is a virtualisation tool that takes programs written in Java, it translates these programs into bytecode, the bytecode is then in turn executed by the Java virtual machine that has JVM on its system (Arora, 2020).

**Table 1: Advantages and disadvantages of JVM**

|  |  |
| --- | --- |
| **Advantages** | **Disadvantages** |
| Secure | Platform specific features |
| Cross platform | Speed |
| Just in compiler |  |

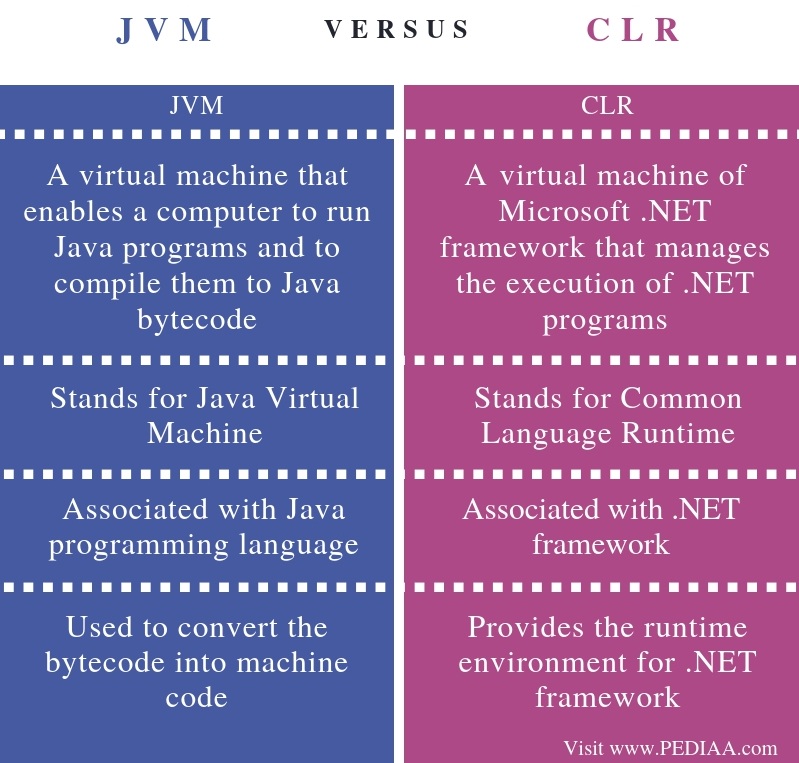
**.NET CLR**



**Figure 4: Execution model (csharpstar.com, 2016)**

CLR is a run environment for the .Net framework, on top of running .Net programs it also provides services that essentially makes development easier. This is according to Microsoft.

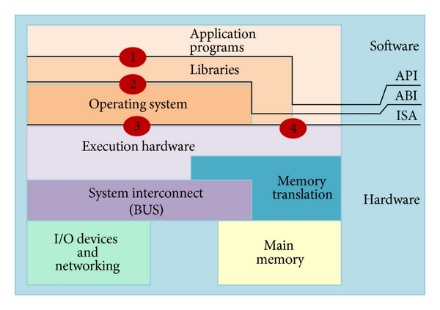
CLR executed code is managed and as a result has a managed environment which ensures that .Net programs are secure, and it also provides an improvement on the cross-language interpretation and a rich set of library classes (GeekForGeeks, 2016).



**Figure 5: Comparison of the JVM and CLR (Pediaa.com,2018).**

### 2.2.2 Library level

Another layer of virtualization is the library level virtualization. In this form of virtualization, the application uses APIs found in the user level libraries. The application and the system share communications through API hooks, and this is how library virtualization is achievable (Taylor 2010).



[This Photo](https://www.hindawi.com/journals/isrn/2013/782174/) by Unknown Author is licensed under [CC BY](https://creativecommons.org/licenses/by/3.0/)

According to WineHQ, the WINE tool is used to achieve library level virtualisation that works on any POSIX compliant systems such as Linux, macOS and BSD. It translates Windows API calls into POSIX calls, and this removes performance and memory penalties of other methods. With WINE the user can integrate their Windows applications into their Desktop.

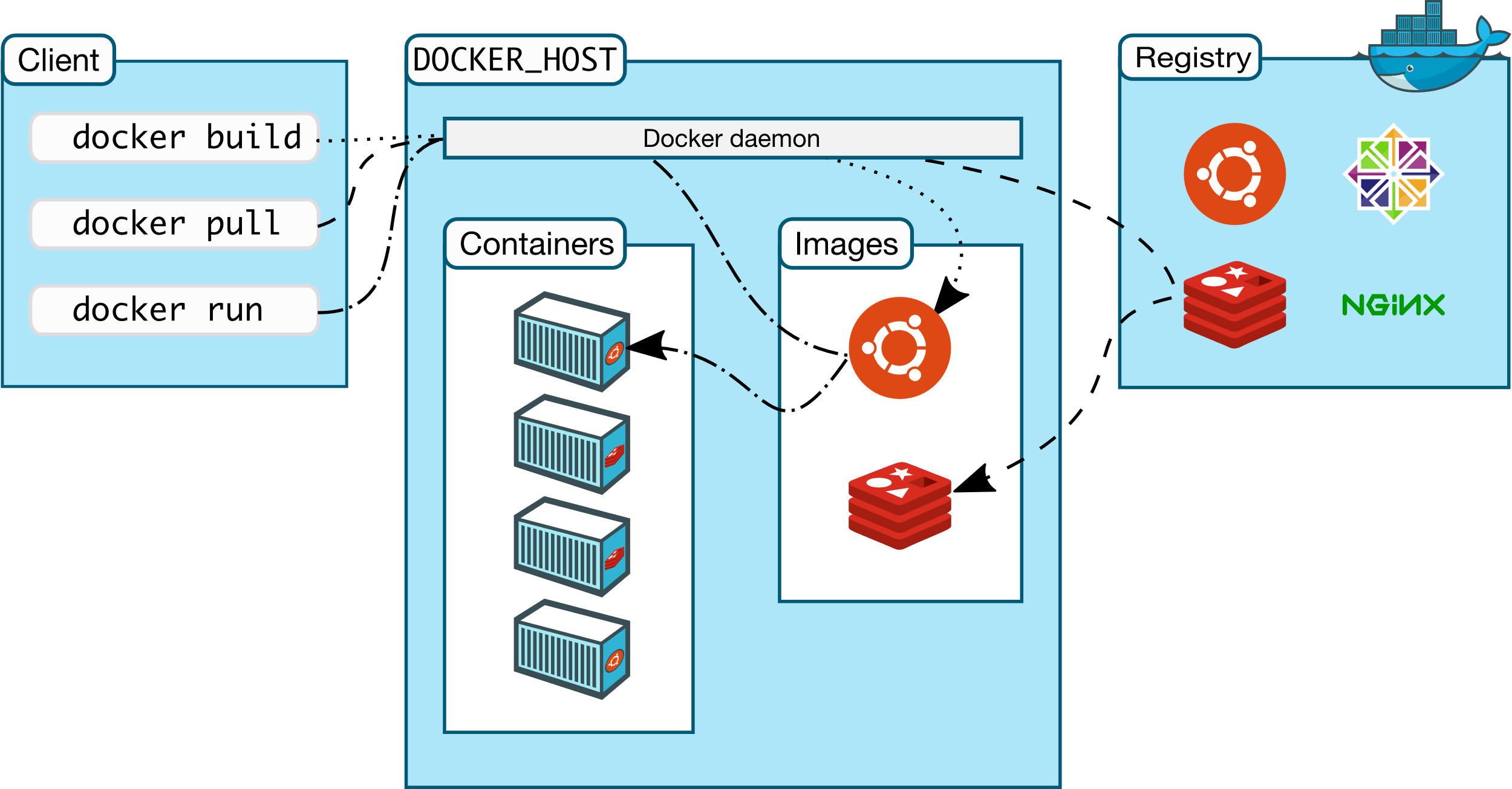
There’s also LxRun which lets users run their Intel Linux and ELF programs to run on other x86 systems. This emulator tool remaps system calls on the fly. LxRun requires copies of the Dynamic loader and any other Linux libraries required by executed application.

### Operating system level

virtualization engines and containers are allowed to operate and run their functionalities in the kernel of the host operating system, this allows for multiple isolated space instances instead of just one. The OS virtualization uses instances such as containers and software containers (Taylor,2009).

This method of virtualization is commonly used in virtual hosting environments where it is useful for securely allocating finite hardware resources amongst many mutually distrusting users. It might also be used by system admins to a lesser extent for consolidating server hardware by moving services on separate hosts into containers on the one server (Tamouri,2016). Some examples of the operating system virtualization are Docker and ENsim’s VPS.

Docker is an open platform for developing, sending, and implementing applications. With Docker, it allows the user to separate applications from the infrastructure so that software can be quickly delivered. With this tool, the developer can manage their infrastructure in the same way as they manage their applications. By exploiting the methodologies employed by docker for shipping, testing, and deploying code quickly, can reduce the delay between writing code and running it in production significantly ([www.Docker.com](http://www.Docker.com)).



**Figure 6: The docker architecture**

### Hardware abstraction layer level

Next, we have the hardware abstraction layer (HAL) virtualization. HAL basically connects the software and hardware of the device (Levi 2014). With HAL virtualization software can be executed on a hardware that was not designed for it. This is done by hiding the processes of

the software and hardware from each other. HAL is commonly used in embedded systems (Mc gee).

**Instruction set architecture level**

To execute legacy code that is not compatible with the setup of the hardware can be tricky. An example would be to run a binary code in a x86 system. However, tools such as QEMU and Dynamo that make use of Instruction set architecture (ISA) which is another virtualization layer make it easier to do so (Taylor 2010).

3 Conclusion

Virtualization is an integral part in the field of Computer science IT, and because of its great benefits it has given developers a more convenient and efficient way to deliver applications that are light, diverse, and most importantly easy to use through the five levels of abstractions, namely Application level, Library level, OS level, HAL, and IAS.

4 Bibliography

Daniels, J., 2009. Server virtualization architecture and implementation. *XRDS: Crossroads, The ACM Magazine for Students*, 16(1), pp.8-12.

Docker. 2021. *Docker Desktop for Mac and Windows | Docker*. [online] Available at: <https://www.docker.com/products/docker-desktop> [Accessed 14 September 2021].

GeeksforGeeks. 2021. *Virtualization In Cloud Computing and Types - GeeksforGeeks*. [online] Available at: <https://www.geeksforgeeks.org/virtualization-cloud-computing-types/> [Accessed 14 September 2021].

Machine?, W., 2021. *What is a Java Virtual Machine? It's Importance for Java Beginners*. [online] W3training School. Available at: <https://www.w3trainingschool.com/java-virtual-machine> [Accessed 14 September 2021].

n.d. *Virtual Machines | IBM*. [online] Available at: <https://www.ibm.com/cloud/learn/virtual-machines> [Accessed 2021].

n.d. *What is Virtualization? | IBM*. [online] Available at: <https://www.ibm.com/cloud/learn/virtualization-a-complete-guide> [Accessed 2021].

Orenstein, G., 2008. NFS offers virtualization benfits. *Business Continuity*, p.14.

Rose, R., 2004. Survey of System Virtualization Techniques. p.2.

Taylor, K., 2021. *Things to Consider for Virtualization Disaster Recovery*. [online] HitechNectar. Available at: <https://www.hitechnectar.com/blogs/virtualization-disaster-recovery-plan/> [Accessed 14 September 2021].

VMware. 2021. *Application Virtualization*. [online] Available at: <https://www.vmware.com/topics/glossary/content/application-virtualization> [Accessed 14 September 2021].

WhatIs.com. 2021. *What is .NET Framework? - Definition from WhatIs.com*. [online] Available at: <https://whatis.techtarget.com/definition/NET-Framework> [Accessed 14 September 2021].

WineHQ. 2021. *WineHQ - Run Windows applications on Linux, BSD, Solaris and macOS*. [online] Available at: <https://www.winehq.org/> [Accessed 14 September 2021].