

## 1. Introduction to Virtualization

### 1.1. What Is Virtualization?

Virtualization is consolidating computing services into an easy to provide and support solution masked away from the user. Think of Virtualization like a prism. A prism is one piece of glass that can split up many different colors and wavelengths of light. In the same sense, Virtualization can partition your computer so that one piece of hardware can support many different operating systems. And not just operating systems, but applications as well. This becomes useful if, for example, you want to run different versions of the same piece of software simultaneously on the same machine. In this course, we are going to discuss the three main types of virtualization: operating system, hardware emulation, and paravirtualization.

The key piece to virtualization is that the user never knows where or how these services are being provided. Virtualization creates a curtain behind which the average user never thinks to look. Just like in the Wizard of Oz, Virtualization allows you to create impressive IT solutions to your client's needs, and no one ever pays attention to what's actually going on behind that curtain.

### 1.2. The History of Virtualization

The history of virtualization spans nearly a half of a century. IBM developed virtualization in the 1960s. It was then called logical partitioning, and was created for the purpose of fully utilizing the mainframe hardware. This early version of a virtual machine was more reliable, offered better security, and was able to share the resources of the mainframe rather than have it split between users.

As desktop computing was introduced in the 1980s and 1990s, Virtualization became less of a priority for engineers. Desktop computing had evolved such that the application layer was broken away from the server and moved to the client desktop. This in itself was sufficient to redistribute the load and allow IT managers to adjust resource needs between x86 platform servers and the clients with greater flexibility.

In the 1980s, Insignia Solutions released a software known as SoftPC. SoftPC allowed Mac users to run Windows applications, not just DOS applications. This allowed the user to work around certain software incompatibilities. Inspired by the success of SoftPC, Apple released their own version of the product known as Virtual PC in 1997.

In the 1990s, engineers at Sun Microsystems began work on a project that became known to the world as Java. Java was targeted towards the nascent world of the Internet, and allowed applications to be supported on a number of operating systems. With Java came the Java Virtual Machine, which worked like a tiny operating system within your computer. The Java Virtual Machine's sole purpose is supporting Java.

VMWare filed for a patent in 1998. In July 2011, VMWare released a version of the first server x86 product, which found its architecture through an open source model that is still being used today by multiple vendors.

### 1.3. How Does Virtualization Work?

Think of your computer like a building. Instead of putting a space heater or air conditioning unit in every room, it is far more efficient to run a duct through the whole building, thereby centralizing and allowing for greater flexibility when allocating resources to different rooms.

In the past, each operating system needed to run on its own physical server, just like the idea of putting a small heater in each room. Virtualization uses what's called a Hypervisor to hide the hardware from the operating systems and have multiple operating systems running in memory on the same physical server.

Let's take a step back and discuss what a hypervisor is, specifically. The first thing you should know about a hypervisor is that there are two types of hypervisors: Type 1 and Type 2. A type 1 hypervisor, also known as a "native," or "bare metal hypervisor," runs directly on its host. It sits between the hardware and the operating system, and acts a liaison between the two "hiding" the hardware from the operating system, so that you are able to run more than one at a time. A Type 2 hypervisor runs within a conventional operating system. It sits between the conventional operating system, and the operating systems you wish to assign to the computer. So a type 1 hypervisor would have 3 layers: hardware, type 1 hypervisor, and operating system. A type 2 hypervisor would have 4 layers: hardware, operating system, type 2 hypervisor, and additional operating systems.