® Rose Business Technologies
Clarity - Simplicity - Productivity

# Benefits of Server Virtualization

Server virtualization is part of an overall virtualization trend that includes storage virtualization, network virtualization, and workload management. This trend is one component in the development of autonomic computing, in which the server environment will be able to manage itself based on perceived activity. Server virtualization can be used to eliminate server sprawl, to make more efficient use of server resources, to improve server availability, to assist in disaster recovery, testing and development, and to centralize server administration.

**Benefits of Server Virtualization**

- Improve server utilisation
- Reduce or contain the number of servers
- Improve security
- Improve availability and uptime
- Improve server and application management
- Improve data backup and protection

Improve server utilisation: More efficient usage of server space by consolidating multiple physical assets onto one virtual system.

Reduce total cost of ownership of IT assets: Both capital expenses (cost of buying new servers or installing a new server farm) and operating expenses (lower monthly energy bills for powering/ cooling a data center, lower labor costs for the IT department).

Improve security and reduce risk of crashes: More secure servers, less downtime and better disaster recovery / business continuity.

Reduce staff time devoted to server management: Transform IT department from a “maintenance" team performing routine tasks to a “strategic” team that can focus on higher level goals and innovation.

Additional flexibility and agility in IT asset allocation: New ways to use servers and innovative deployment of technology specialists.

**What is Server Virtualization?**

Server virtualization is the masking of server resources, including the number and identity of individual physical servers, processors, and operating systems, from server users. The server administrator uses a software application to divide one physical server into multiple isolated virtual environments. The virtual environments are sometimes called virtual private servers, but they are also known as guests, instances, containers or emulations.

There are three popular approaches to server virtualization: the virtual machine model, the paravirtual machine model, and virtualization at the operating system (OS) layer.

Virtual machines are based on the host/guest paradigm. Each guest runs on a virtual imitation of the hardware layer. This approach allows the guest operating system to run without modifications. It also allows the administrator to create guests that use different operating systems. The guest has no knowledge of the host's operating system because it is not aware that it's not running on real hardware. It does, however, require real computing resources from the host -- so it uses a hypervisor to coordinate instructions to the CPU.

The hypervisor is called a virtual machine monitor (VMM). It validates all the guest-issued CPU instructions and manages any executed code that requires addition privileges. VMware and Microsoft Virtual Server both use the virtual machine model.

The paravirtual machine (PVM) model is also based on the host/guest paradigm -- and it uses a virtual machine monitor too. In the paravirtual machine model, however, The VMM actually modifies the guest operating system's code. This modification is called porting. Porting supports the VMM so it can utilize privileged systems calls sparingly. Like virtual machines, paravirtual machines are capable of running multiple operating systems. Xen and UML both use the paravirtual machine model.

Virtualization at the OS level works a little differently. It isn't based on the host/guest paradigm. In the OS level model, the host runs a single OS kernel as its core and exports operating system functionality to each of the guests. Guests must use the same operating system as the host, although different distributions of the same system are allowed. This distributed architecture eliminates system calls between layers, which reduces CPU usage overhead. It also requires that each partition remain strictly isolated from its neighbors so that a failure or security breach in one partition isn't able to affect any of the other partitions. In this model, common binaries and libraries on the same physical machine can be shared, allowing an OS level virtual server to host thousands of guests at the same time. Virtuozzo and Solaris Zones both use OS-level virtualization.