

THE ROLE OF VIRTUALIZATION IN DEVOPS



The primary pursuit of DevOps structuring and systems is to improve the speed and quality of the software development process. DevOps looks to achieve the simultaneous increase in both quality and rate of deployments by emphasizing the importance of collaboration and promoting transparency through the use of tools that empower communication and data transparency. DevOps teams are made of cross-discipline members that bring their unique skill sets and diverse perspectives to bear to increase the team's ability to tackle their projects quickly and efficiently.

Aside from an emphasis on communication and teamwork, DevOps also looks for technologies aiding in powering up the [software development life cycle \(SDLC\)](#). Tools like the cloud and automation are leveraged to their fullest extent to provide DevOps teams with every advantage possible to help them in delivering on the [continuous integration and continuous delivery \(CI/CD\)](#) goals of the enterprise. Any piece of tech that alleviates pressure from the team and optimizes the hours they spend on each task is a valuable asset for the organization to employ.

Another such tool that helps to ensure rapid and stable deployments occur on a regular basis is virtualization.

(This article is part of our [DevOps Guide](#). Use the right-hand menu to navigate.)

Basics of Virtualization

Virtualization is the process of creating software that mimics various hardware and software environments without having to change out the physical hardware itself. Virtualization techniques create abstractions of physical hardware components to create aggregated pools of resources made up of CPUs, memory, storage, networking, and applications among other components.

Essentially, a virtual machine (VM) functions as a simulation of specific IT hardware and software configurations without requiring the physical components to be interchanged. This allows a single piece of hardware to host to numerous VM configurations simultaneously. Virtualization is not a new concept. In fact, the utilization of virtualization techniques dates back to as early as the late 1960s.

Virtualization has come a long way since its first inception, but the core concept remains the same - achieve more computing capabilities with fewer resources and running multiple, independent systems simultaneously on a single machine. Today's VMs are much more complex and capable of quickly scaling up or down to meet resource requirements on demand thanks, in part, to leveraging cloud technology in addition to virtual machine techniques. The power of modern VMs allows engineers to create environments that are virtually identical to real-world hardware configurations with software running within the VM environment.

This is especially important in today's world where computers have become incredibly complex and diverse thanks to the proliferation of various operating systems, hardware manufacturers, and the vast array of mobile devices that see regular use across the globe. As computers have continued to increase in capabilities, complexity, and diversity, creating software that runs on these various devices also grows in complexity. These factors have created a situation in which creating and testing software for the various devices that software will eventually be utilized on is a daunting if not impossible task without leveraging the power of virtualization.

Different Types of Virtualization

There are three primary types of virtualization: server virtualization, network virtualization, and desktop virtualization.

Server Virtualization

Server virtualization enables a single physical server to run multiple operating systems at once. This creates a more efficient server providing reduced operating costs, increased performance, and faster workload capacity all while reducing physical server complexity.

Network Virtualization

Network virtualization reproduces physical networks in a virtual environment which allows applications to run on the simulated network in the same way they would on a physical one. Virtual networks provide enhanced operational benefits compared to their physical counterparts and do so at reduced costs.

Desktop Virtualization

Desktop virtualization mimics the environment and settings of an instance of a desktop and the applications hosted on the device. This allows users to access enterprise resources from off-site or even from devices that would otherwise be incapable of running the hosted services. This technology allows users to leverage the processing power of more advanced machines from standard or even mobile devices - providing them with the capabilities they need to get their work done on the go.

The applications for the various forms of virtualization are numerous, but the primary benefit they all provide is the ability to access and leverage resources that would otherwise be prohibitively expensive, if not altogether impossible. Virtualization is a powerful tool with various applications for enterprises of all sizes, but it has specific uses for organizations that utilize DevOps systems.

How Virtualization and DevOps Work Together

As was mentioned above, DevOps seeks to optimize the development, testing, and deployment processes of software development through the use of collaboration and cutting-edge technologies. Virtualization enables DevOps teams to develop and test within simulated environments that run the full gamut of devices available to consumers while also testing deployment on virtual live environments. This enables development to occur alongside real-time testing of how the changes will impact the entire system. This level of accuracy in testing makes for vastly reduced deployment times and increased stability.

Containerization, another popular technology DevOps teams are adding to their toolkits, is essentially the process of taking virtualization one step further by utilizing operating system (OS) kernels to run multiple applications within a single container. Containerization furthers the concept of virtualization by not only providing a digital configuration that mimics hardware setups but also mimics the OS and libraries that encompass the entire runtime environment.

Doing this allows DevOps teams to build, test, and deploy their software solutions within live simulated environments and doing so while using vastly reduced computing resources. This results in DevOps teams being able to achieve more with less. Continuous deployment is achievable through the flexibility of virtualization and containerization technologies that allow updates to be tested and deployed on multiple servers with enhanced stability and consistency. Virtualization and containerization both play large roles in the maximization of enterprise resources to ensure development resources are optimally leveraged at each stage of the DevOps process.

DevOps: Solutions for You

If DevOps sounds like a good fit for your organization's needs but you want to make sure you get it right the first time or you're struggling with your current DevOps implementation, BMC is the IT solution partner you need. Read more about how automation and DevOps systems can help increase the rate at which you deploy products with BMC's free eBook: [*Automate Cloud and DevOps Initiatives*](#).

BMC expert consultants are available to work with you to bring their knowledge and expertise to your organization. BMC provides custom-tailored [*Deployment Services*](#) for your organization to tackle the unique challenges you face. When partnering with BMC, you get:

- **Faster service delivery:** Agile releases that keep up with rapid demand
- **Visibility across data:** Ensure compliance and data accuracy
- **Cost-effective service:** Increased productivity and performance
- **Experienced DevOps professionals:** Equip you with the tools you need for success
- **Conversion or upgrade:** Seamless modernization or total replacement
- **All tailored for the specific needs of your organization.**

Download or view the [*Solution Implementation Overview*](#) online to learn more about how [*BMC Consulting Services*](#) can help you. Check out [*The Power of Shifting Left*](#) to learn more about amping up your CI/CD pipeline by modernizing your workload orchestration. Then contact the experts at BMC to find out more about how to leverage DevOps and virtualization for enhanced building, testing, and deployment success.