



([HTTPS://WWW.ASSISTANZ.COM/](https://www.assistanz.com/))

[About Us \(https://www.assistanz.com/about-us\)](https://www.assistanz.com/about-us)

[Cloud Services](#) [IMS Consulting](#)

[Mobility \(https://www.assistanz.com/mobility\)](https://www.assistanz.com/mobility)

Home (<https://www.assistanz.com>) > Blog (<https://www.assistanz.com/blog/>) > Blog (<https://www.assistanz.com/category/blog/>) > Introduction To Windows Container And Docker Terminology

Introduction to Windows Container and Docker

🕒 March 25, 2017 Posted by: Loges Category: Blog, Containers

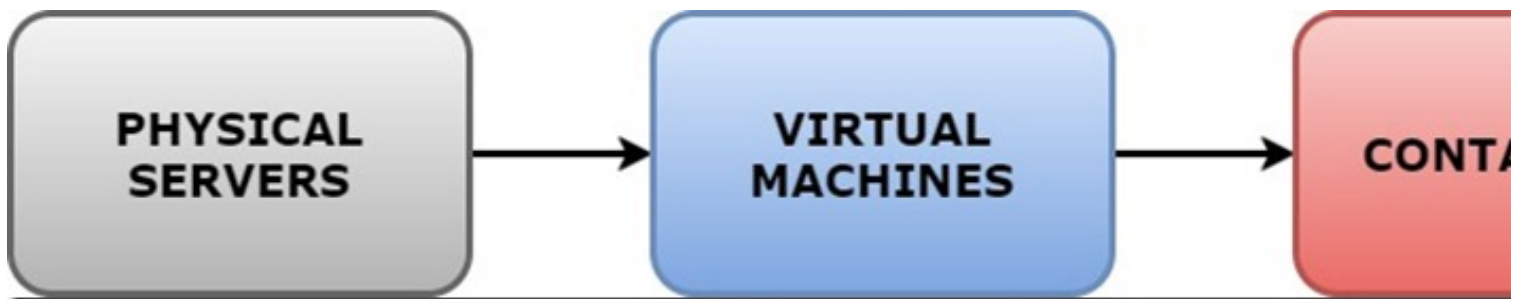
(<https://www.assistanz.com/blog/container-docker-terminology>)



Introduction to Windows Container and Docker Terminology

In this blog, we are sharing about **windows container** and **docker** terminology in windows 2016.

UNDERSTANDING THE USE CASE FOR CONTAINERS



(<https://www.assistanz.com/wp-content/uploads/2017/03/Untitled-Diagram-1.jpg>)

PHYSICAL SERVERS

In Earlier days, we use to deploy individual physical servers for each application that we need to support. So week to prepare the server. Typically there is a lot of unused physical resources. As technology advance, virt software like VMware made easy to deploy virtual machines. It helps the business to more efficient use of ha available. They can spin up a virtual machine within few minutes without any hassles.

VIRTUAL MACHINES

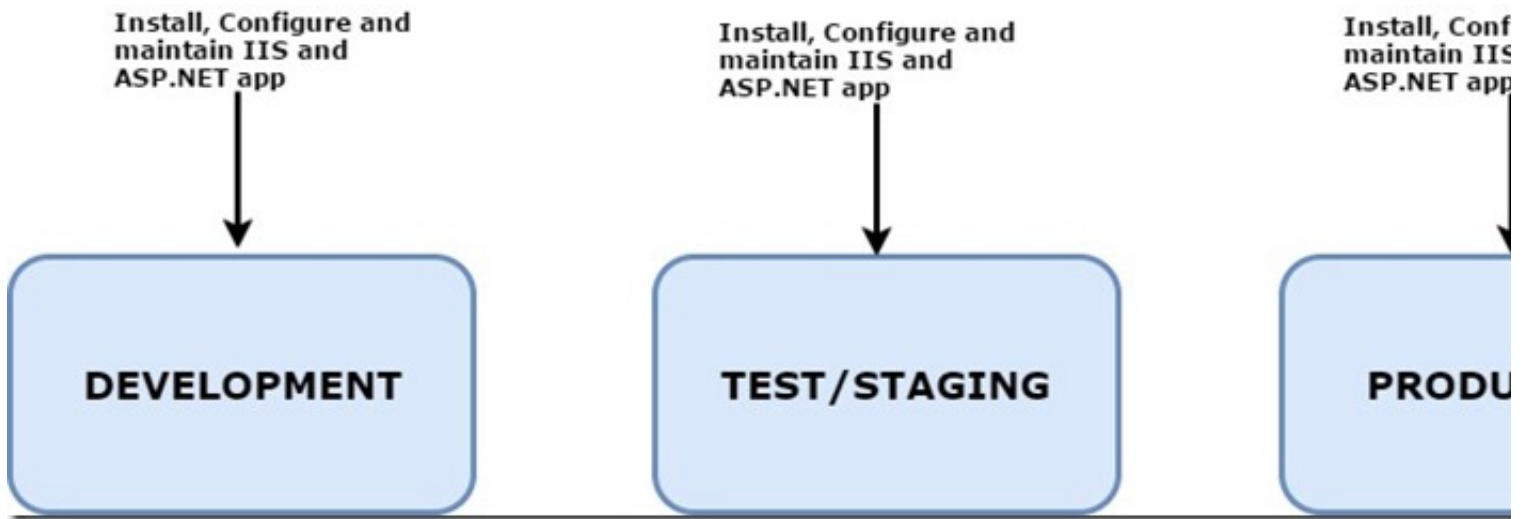
As the downside, each VM will have a full copy of OS files which leads to slow bootup. We need to take care of licensing, patching and maintenance of VM. On top of that, VM hard disk size is also too large and it's hard to manage environments. Also, it needs more CPU and memory to run VM's.

CONTAINERS

So many businesses had now turned to container technology to bypass the above limitations. Containers are not virtual machines. Containers do not require complete operating system. It actually shares the OS kernel on which it is running on. So it reduced tons of resources and also it eliminates licensing & patching operation system.

The main benefit about containers are it will boot up instantly and it's smaller in size. This **portability & stability** are the reasons for containers to gain more popularity. Developers can build and test apps locally in their container(like a laptop). They can run the same container easily on physical servers, virtual machines and also in the cloud with ease of modification. So application environment will be identical no matter where the developer works on the application.

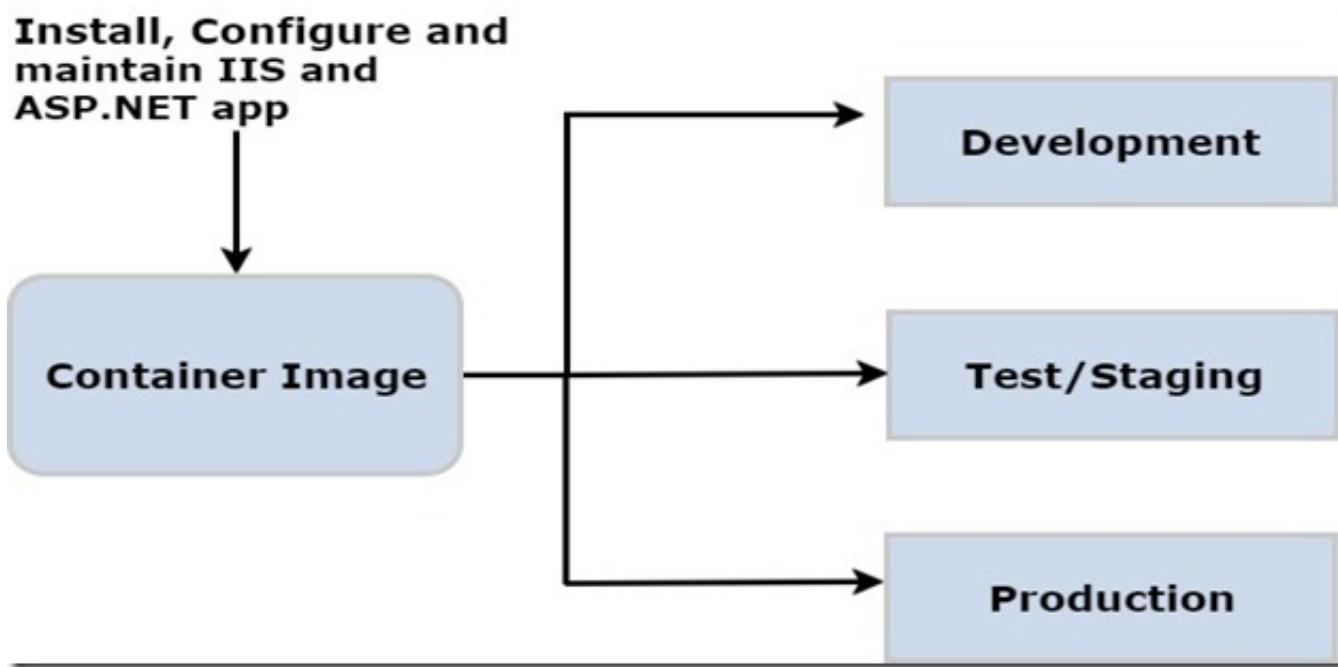
APPLICATION PROCESS IN VM



(<https://www.assistanz.com/wp-content/uploads/2017/03/Untitled-Diagram.jpg>)

In above diagram, the development process for **ASP.NET** application will take in an isolated environment. Once the process is complete, we move the application to a different environment for testing the application. If everything is working fine, we need to move the application to the production environment to make the application available to the users. In this process, we are installing **IIS & ASP.NET** every time in separate VM's which take lots of time & resources involved in this process. We experience different issues as we move the application from one environment to another.

APPLICATION PROCESS IN CONTAINER

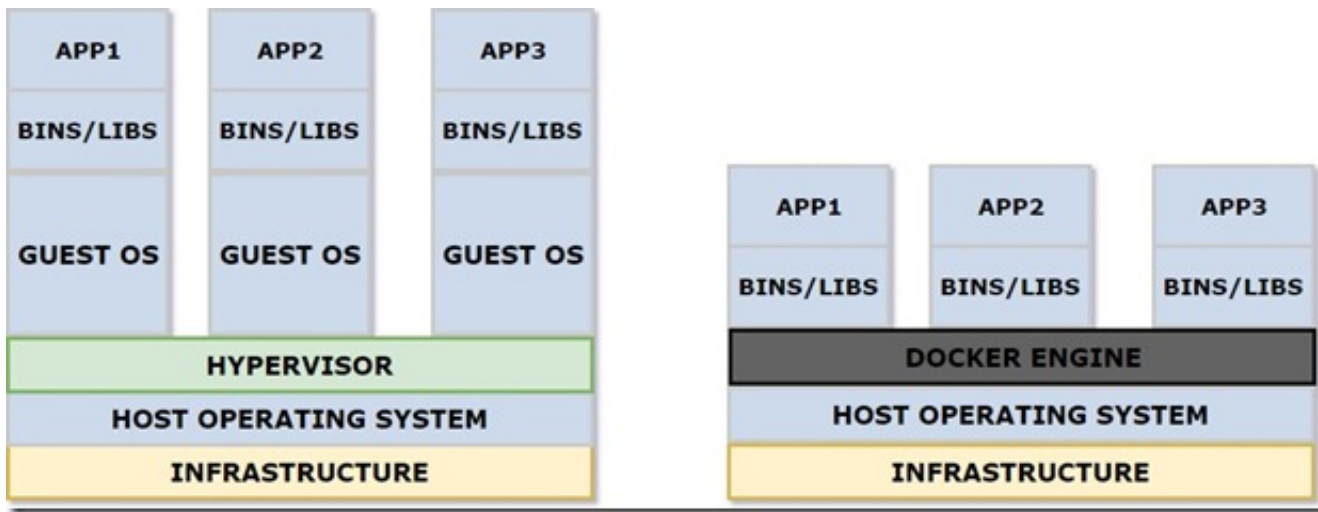


(<https://www.assistanz.com/wp-content/uploads/2017/03/Untitled-Diagram-2.jpg>)

In the container, we are packing our application and its dependencies into a **container image** and we can run it in any environment. In above diagram, we install IIS with ASP.NET app and build a container image. Now we can run the image in development, testing/staging & production environment. This is one of the biggest benefits from this technology.

DOCKER INTRODUCTION

DOCKER is an open source project and Microsoft contributed and partnered with them to manage the windows containers. They did for Linux containers. So developers, administrators can use same tools and technologies that are available for both Linux & Windows containers.



(<https://www.assistanz.com/wp-content/uploads/2017/03/Untitled-Diagram-3.jpg>)

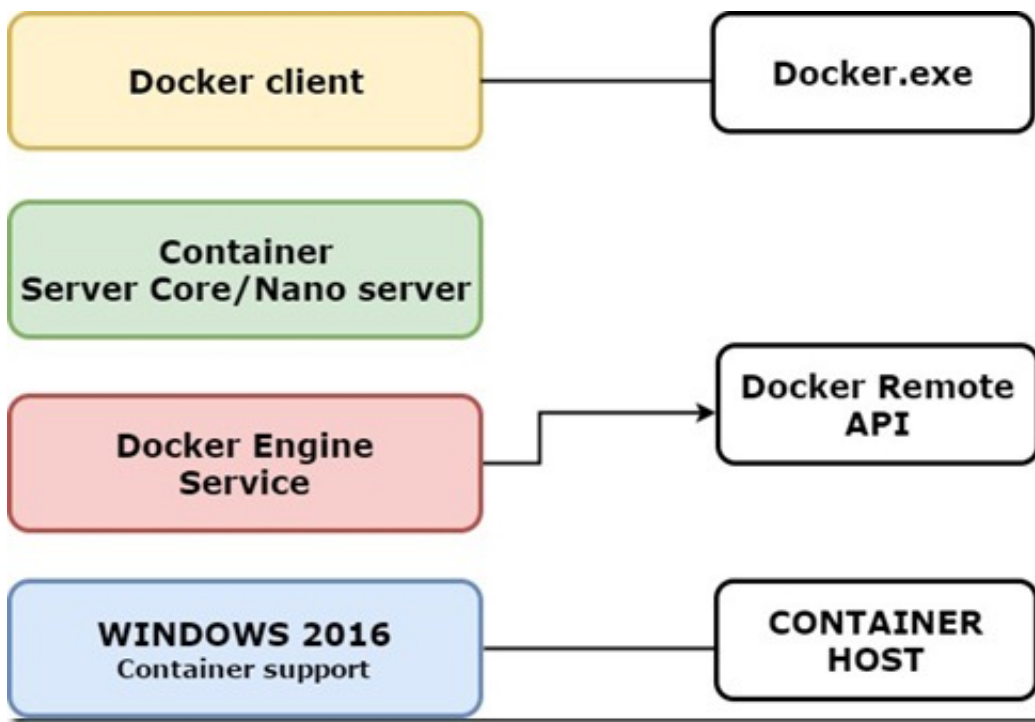
VIRTUAL MACHINES

Vs

CONTAINERS

In above diagram on the left-hand side, we have virtual machines running on top of the hypervisor like VMware. On the right-hand side instead of the hypervisor, we have a new technology called **DOCKER ENGINE**. This engine runs on Linux or Windows which help to run containers. One more component which we use is **docker client** to interact with containers to build images. Docker engine is communicating with docker client through API to execute the containers.

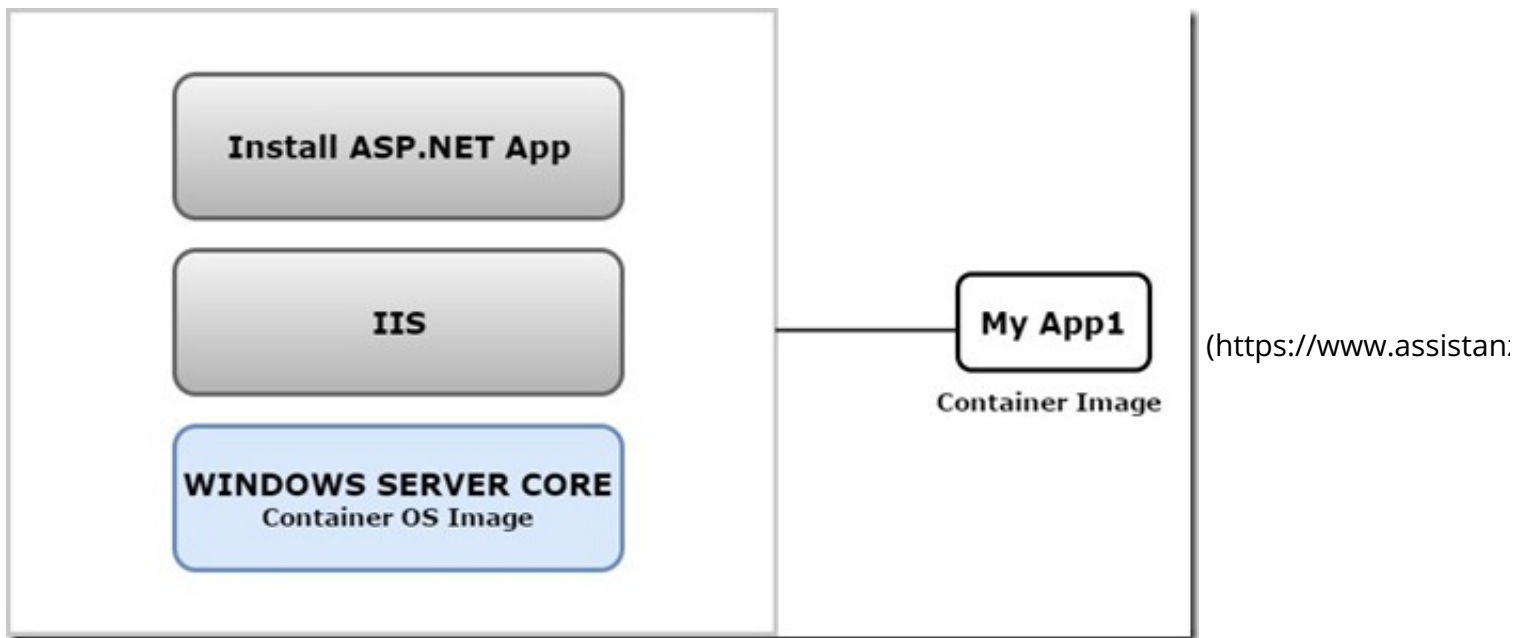
DOCKER RUN IN WINDOWS 2016



(<https://www.assistanz.com/wp->

content/uploads/2017/03/Untitled-Diagram-4.jpg)

To run the container we need windows 2016 host machine (Physical machine or VM) and native **windows co** need to be installed in it. We referred it as **Container Host**. Then we need to install **Docker Engine** and run a container host. This helps to create and launch containers. Now we can start creating containers using windc core and nano server. There is no graphical interface for this containers. In order to manage the containers, the docker client tool **docker.exe**. When we issue a command through docker.exe it talks to docker engine t



content/uploads/2017/03/Untitled-Diagram-5.jpg)

To create a new container first we need to select an OS image for your container either windows server core. As above example, we select windows server core as OS image then we install **IIS** on top of that image. Then **ASP.NET** application and we package it as customized container image. This image can be pushed to centrali; we can pull it on any other container host as desired. Also, we can launch the same container on the differen required.

Thanks for reading this blog. We hope this will be useful for you to learn the basic container and docl windows 2016.

windows 2016 container
(<https://www.assistanz.com/tag/windows-2016-container/>)

SHARE

(/ #facebook)

(/ #twitter)

(/ #google_plus)

windows container
(<https://www.assistanz.com/tag/windows-container/>)

windows container basics
(<https://www.assistanz.com/tag/windows-container-basics/>)