



.NET

DEVELOPER

DAYS

Everyone Loves Docker Containers Before They Understand Docker Containers

Alex Mang (@mangalexandru)



Sponsors and Partners

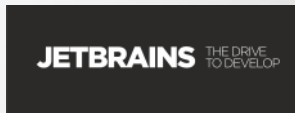
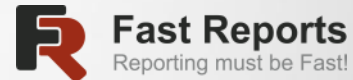
Strategic Sponsors



Gold Sponsors



Silver Sponsors



Software

Ballmer: "Linux is a cancer"

Contaminates all other software with Hippie

2 Jun 2001 at 18:19, [Thomas C Greene](#)



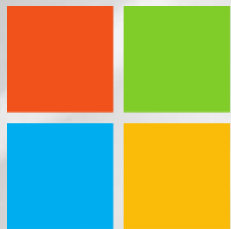
Microsoft CEO and incontinent over-stater of facts Steve Ballmer sai



Vista

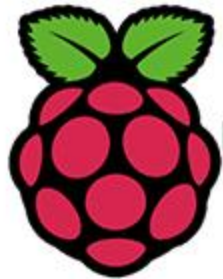
Microsoft®





Microsoft





RaspberryPi



Microsoft®
.NET



Cloud Expectations Today

Availability

Agility

Hyper-
scalability



docker

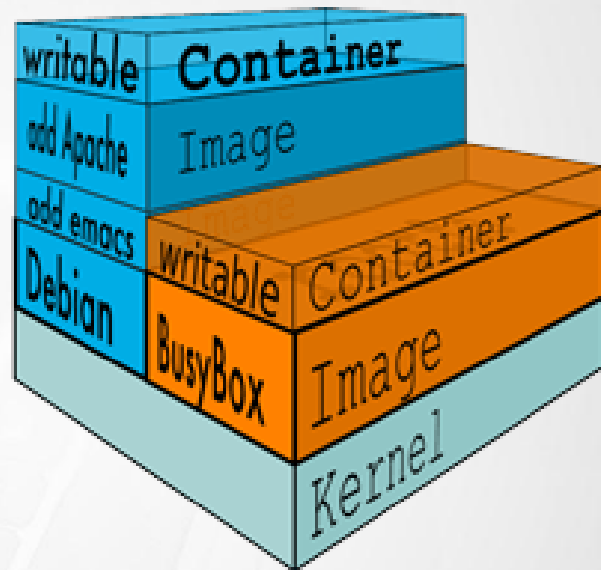
What is Docker?

What is Docker?

- Code + Runtime + System tools + System libs

→ in a single package

- Code runs the same, always, regardless of environment



Why Docker?

Lightweight

- Shared kernel
- Layered filesystem

Open

- Runs both on Linux and Windows

Secure

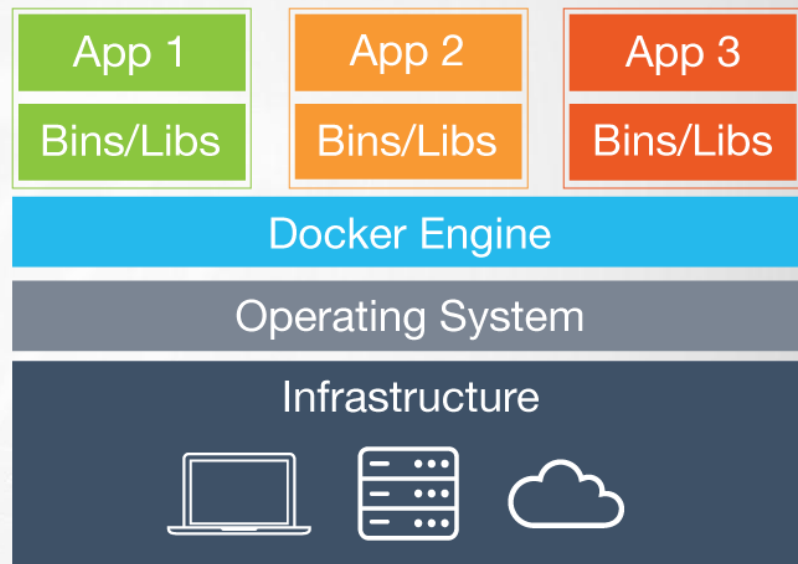
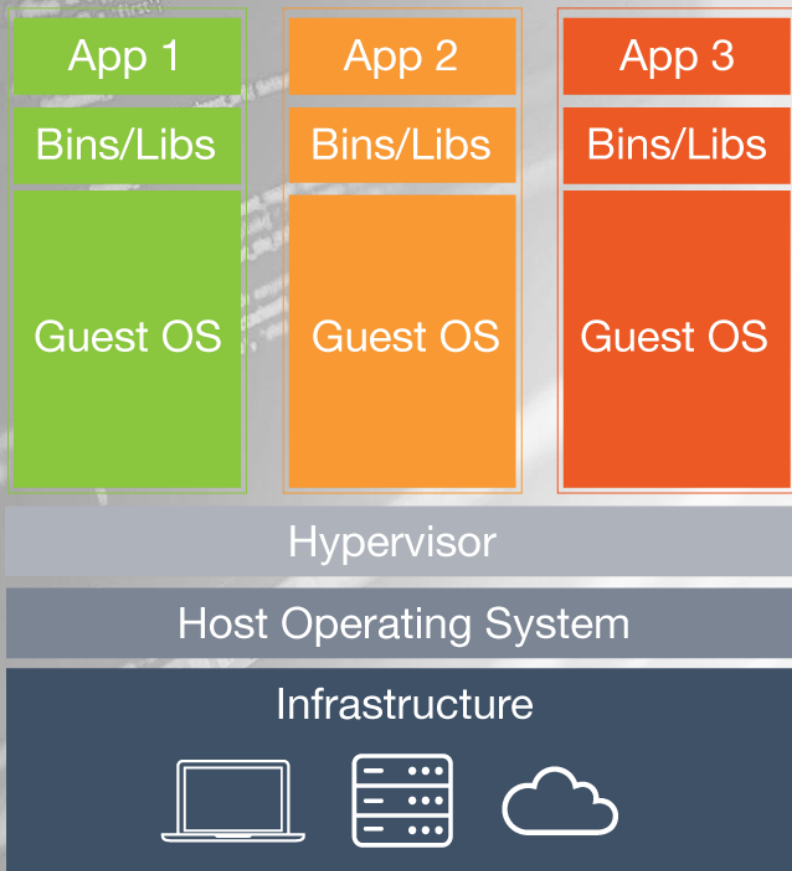
- Apps are isolated from each other and the underlying infrastructure

Why Do Developers Like It?

- Developers can build apps in ANY LANGUAGE using ANY TOOLCHAIN
- *Dockerized* apps are portable
 - They can run on your PC, on a colleague's iMac, on QA servers running Ubuntu, in production VMs running RedHat
- There's a “NuGet” for Docker with 13K+ apps to build upon
- Docker helps devs build and ship high-quality apps

Why Do Sysadmins Like It?

- Sysadmins use Docker to provide standardized environments for development, QA and production teams
- No more “works on my machine” finger-pointing
- With “Dockerized” apps, sysadmins abstract away differences in OS distributions and underlying infrastructure
- Workload deployment is not constrained by infrastructure technology



VMs vs. Docker

Virtual Machine Analogy

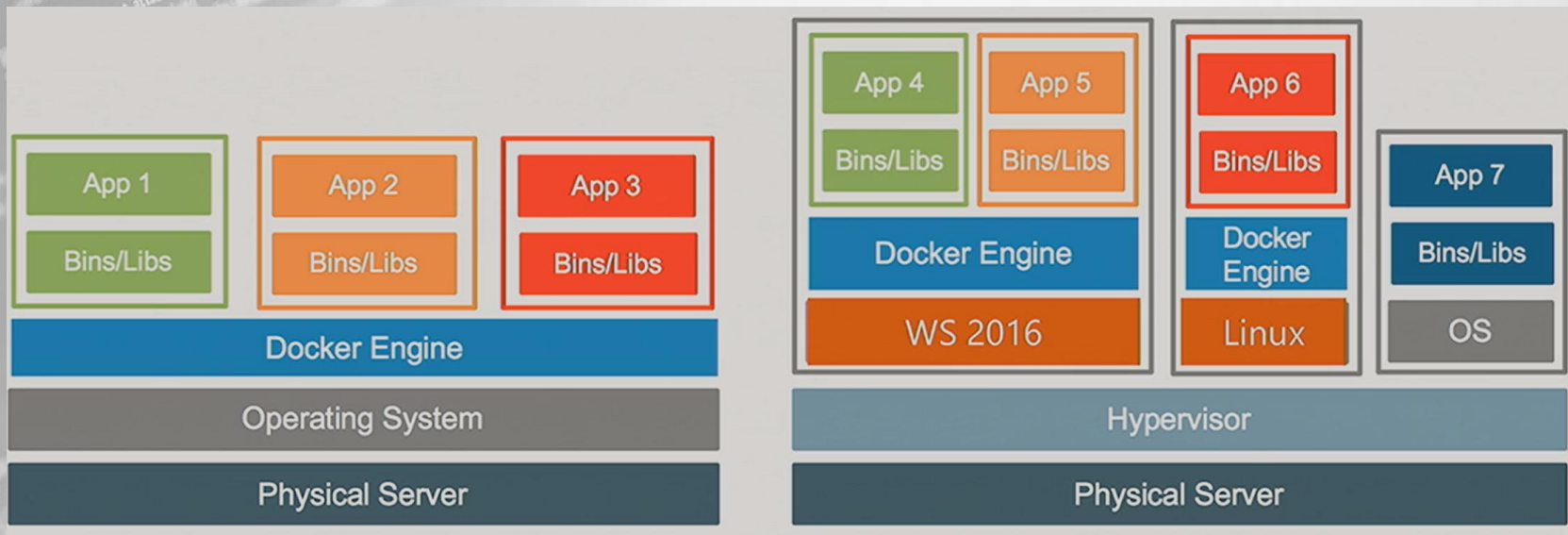


Source: <http://kentonconstruction.com/>

Containers Analogy



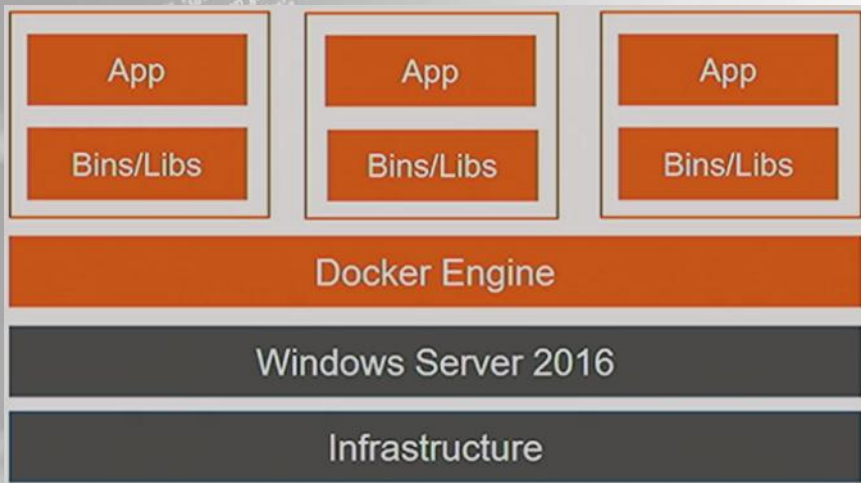
Source: <http://kentonconstruction.com/>



Source: MS Ignite 2016, BRK3189

Different, but not mutually exclusive!

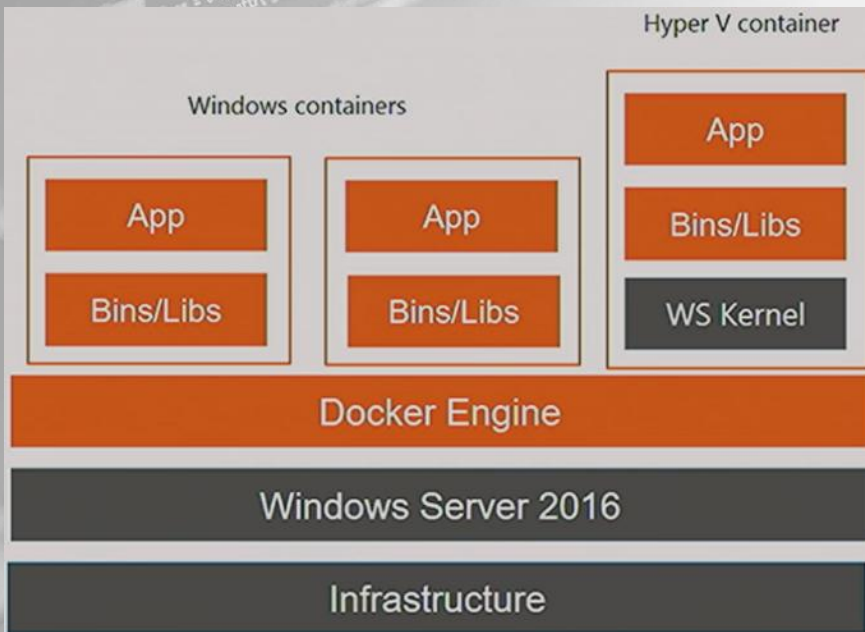
Docker + Windows Server = Windows Containers



Source: MS Ignite 2016, BRK3189

- Native Windows containers powered by Docker Engine
- Windows kernel engineered with new primitives to support containers
- Deep integration with 2y+ of engineering collaboration in Docker Engine and Windows Server

Hyper-V Containers = Docker + Windows Containers



- Leverages same Docker and Windows Server technology
- Flexible container deployment options available for IT pros
- One workload per Hyper-V container delivers added isolation for multi-tenant or hostile environments

Source: MS Ignite 2016, BRK3189

Container / VM comparison

Source: MS Ignite 2016, BRK3319

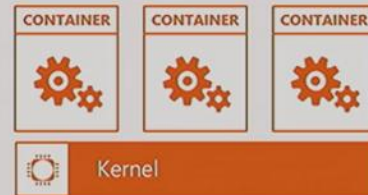
Containers = Operating system virtualization



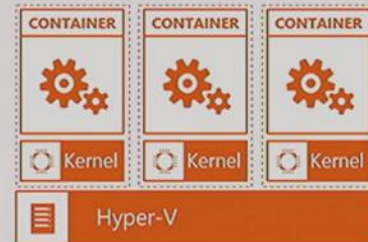
Traditional virtual machines = hardware virtualization



Windows Server Containers
Maximum speed and density



Hyper-V Containers
Isolation plus performance



Docker file – Linux example

```
1 our_base_image
2 FROM alpine:latest
3
4 # Install python and pip
5 RUN apk add --update py-pip
6
7 # upgrade pip
8 RUN pip install --upgrade pip
9
10 # install Python modules needed by the Python app
11 COPY requirements.txt /usr/src/app/
12 RUN pip install --no-cache-dir -r /usr/src/app/requirements.txt
13
14 # copy files required for the app to run
15 COPY app.py /usr/src/app/
16 COPY templates/index.html /usr/src/app/templates/
17
18 # tell the port number the container should expose
19 EXPOSE 5000
20
21 # run the application
22 CMD ["python", "/usr/src/app/app.py"]
```

Source: MS Ignite 2016, BRK3189

- Instructions on how to build a Docker image
- Looks very similar to “native” commands
- Important to optimize your Dockerfile

Docker file – Windows example

20 lines (14 sloc) | 775 Bytes

Raw

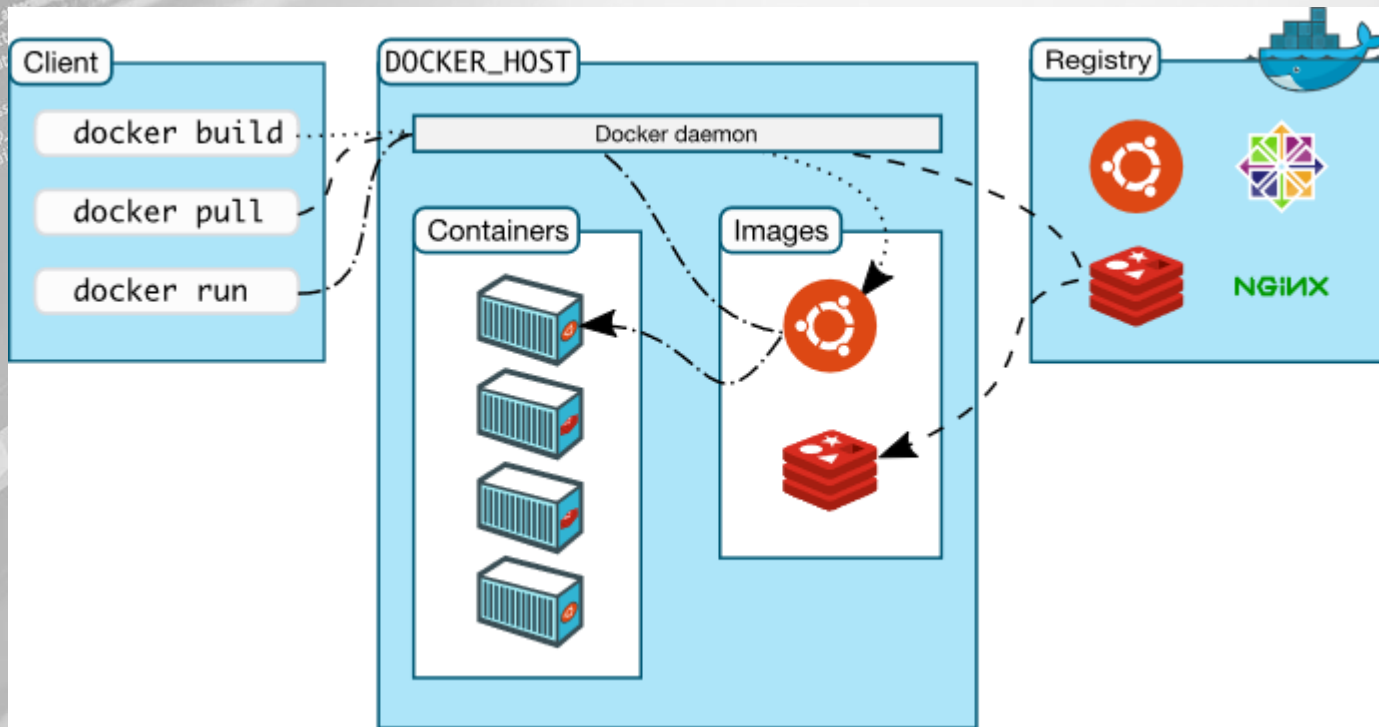
Blame

History

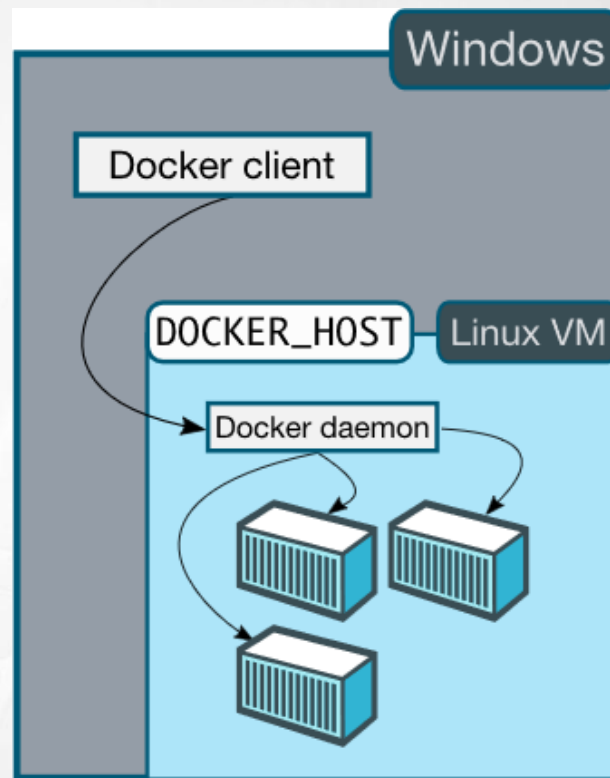
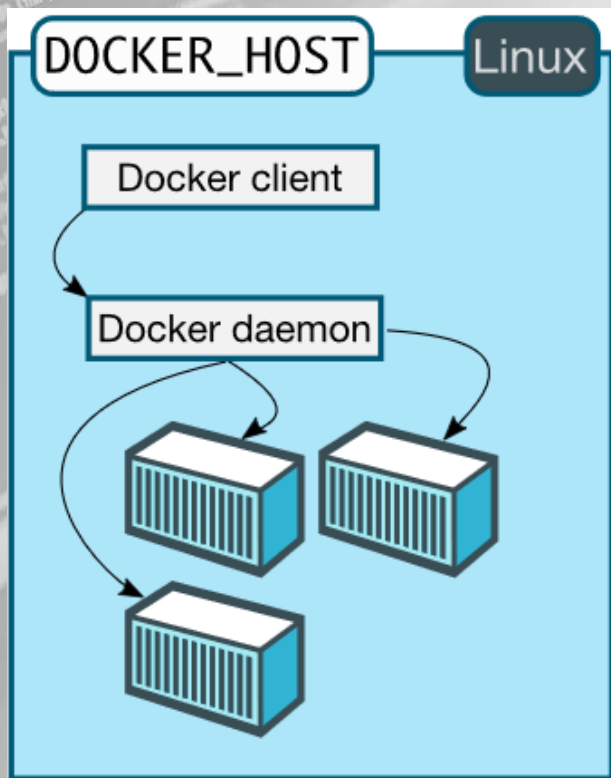


```
1 FROM microsoft/dotnet:1.0.0-preview2-windowsservercore-sdk
2
3 SHELL ["powershell", "--Command", "$ErrorActionPreference = 'Stop';"]
4
5 # REMARK: Temporary workaround for Windows DNS client weirdness
6 RUN set-itemproperty -path 'HKLM:\SYSTEM\CurrentControlSet\Services\Dnscache\Parameters' -Name ServerPriorityTimeLimit -Value 0 -Type DWord
7
8 RUN New-Item -Path \MusicStore\samples\MusicStore.Standalone -Type Directory
9 WORKDIR MusicStore
10
11 ADD samples/MusicStore.Standalone/project.json samples/MusicStore.Standalone/project.json
12 ADD NuGet.config .
13 RUN dotnet restore --no-cache .\samples\MusicStore.Standalone
14
15 ADD samples samples
16 RUN dotnet build .\samples\MusicStore.Standalone
17
18 EXPOSE 5000
19 ENV ASPNETCORE_URLS http://0.0.0.0:5000
20 CMD dotnet run -p .\samples\MusicStore.Standalone
```

Docker Architecture



Docker Installation



Docker clients

- `docker` (as a CLI)
- `docker-Machine` (as yet another CLI)
- `docker`
 - Start, stop, kill etc. containers
- `docker-machine`
 - Install and run Docker (engine!) on Mac and Windows
 - Provision and manage multiple remote Docker hosts

A photograph of three scientists in a laboratory. A woman on the left is looking through a microscope. A woman in the middle is holding a round-bottom flask containing a blue liquid. A man on the right is wearing safety glasses and holding a pipette. The scene is overlaid with a semi-transparent blue rectangle containing the text 'DEMO' and another semi-transparent blue rectangle below it containing the text 'Let's use Docker!'.

DEMO

Let's use Docker!

Windows PowerShell

```
PS C:\Users\amang> docker-machine create --driver azure --azure-subscription-id "05f2e3c3-5c8c-42f7-b6a7-6b0adf7b8b6bf" netdevdaysdocker2016
Creating CA: C:\Users\amang\.docker\machine\certs\ca.pem
Creating client certificate: C:\Users\amang\.docker\machine\certs\cert.pem
Running pre-create checks...
(netdevdaysdocker2016) Microsoft Azure: To sign in, use a web browser to open the page https://aka.ms/devicelogin. Enter the code CESME5YM2 to authenticate.
(netdevdaysdocker2016) Completed machine pre-create checks.
Creating machine...
(netdevdaysdocker2016) Querying existing resource group. name="docker-machine"
(netdevdaysdocker2016) Creating resource group. name="docker-machine" location="westus"
(netdevdaysdocker2016) Configuring availability set. name="docker-machine"
(netdevdaysdocker2016) Configuring network security group. name="netdevdaysdocker2016-firewall" location="westus"
(netdevdaysdocker2016) Querying if virtual network already exists. name="docker-machine-vnet" location="westus"
(netdevdaysdocker2016) Creating virtual network. location="westus" name="docker-machine-vnet"
(netdevdaysdocker2016) Configuring subnet. name="docker-machine" vnet="docker-machine-vnet" cidr="192.168.0.0/16"
(netdevdaysdocker2016) Creating public IP address. name="netdevdaysdocker2016-ip" static=false
(netdevdaysdocker2016) Creating network interface. name="netdevdaysdocker2016-nic"
(netdevdaysdocker2016) Creating storage account. name="vhdsu62vtbg9j8i0rxu1n0vs" location="westus"
(netdevdaysdocker2016) Creating virtual machine. username="docker-user" osImage="canonical:UbuntuServer:16.04.0-LTS:latest" name="netdevdaysdocker2016" location="westus" size="Standard_A2"
Waiting for machine to be running, this may take a few minutes...
Detecting operating system of created instance...
Waiting for SSH to be available...
Detecting the provisioner...
Provisioning with ubuntu(systemd)...
Installing Docker...
Copying certs to the local machine directory...
Copying certs to the remote machine...
Setting Docker configuration on the remote daemon...
Checking connection to Docker...
Docker is up and running!
To see how to connect your Docker Client to the Docker Engine running on this virtual machine, run: C:\Program Files\Docker\Docker\Resources\bin\docker-machine.exe env netdevdaysdocker2016
PS C:\Users\amang> docker-machine ls
NAME          ACTIVE   DRIVER   STATE   URL               SWARM   DOCKER   ERRORS
netdevdaysdocker2016 -        azure   Running  tcp://40.78.16.164:2376          v1.12.2
PS C:\Users\amang>
```

VERY Useful Resources

- Image2Docker
 - <https://github.com/docker/communitytools-image2docker-win>
- GitHub Microsoft repo. for Docker
 - <https://github.com/Microsoft/docker>
- Microsoft Windows Container Samples
 - <https://github.com/Microsoft/Virtualization-Documentation/tree/master/windows-container-samples/>

Q & A

HOW DO YOU
PLAN DISTRIBUTION
OF WATER AROUND
THE GLOBE?

I'M THINKING OF
A CLOUD-BASED
SOLUTION



The Creation – Day Two



That's all Folks!

Alex Mang

@mangalexandru

Sponsors and Partners

Strategic Sponsors



Gold Sponsors



Silver Sponsors

