

DOCKER ON WINDOWS

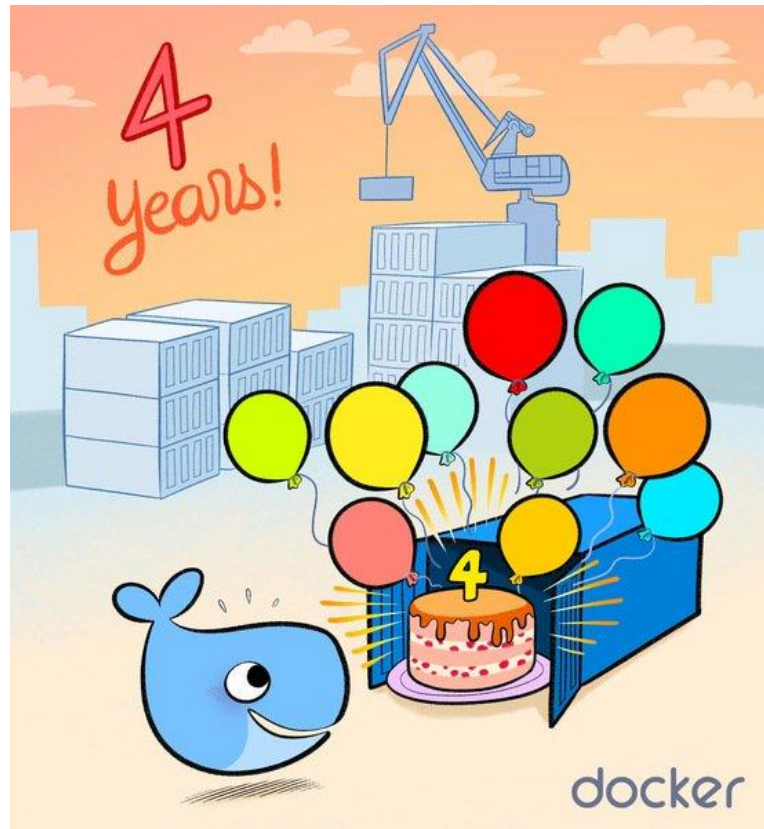
4th #DockerBDay Celebration

Marseille

15/03

Nice

16/03



- ▶ Introduction
- ▶ Architecture
- ▶ Pour finir
- ▶ Questions / Réponses
- ▶ Annexes (Ecosystème & Tools)

INTRODUCTION



- ▶ Pas Docker **for** Windows
 - <https://docs.docker.com/docker-for-windows/>
 - Ne fait pas tourner les Linux containers
- ▶ Microsoft et Docker ont formé un partenariat en 2014
- ▶ Le kernel Windows a maintenant les primitives de containerization
 - Celui de Windows Server 2016 et Windows 10 AU
- ▶ Microsoft et Docker ont travaillé pour porter (pas un fork) le Docker Engine sur Windows, en conservant l'esprit Open Source (tout est disponible sur <https://github.com/docker/docker>)
- ▶ Même remote API, même tools par dessus (Compose, Swarm, etc.)
- ▶ Les services auxiliaires comme la registry, Hub, etc. ont le support multi-arch, permet de maintenir en side by side les Docker images Linux et Windows

- ▶ Partenariat Commercial
- ▶ Commercially Supported Docker Engine aka “CS Docker Engine”
 - Docker, Inc.’s tested, validated, and supported package of Docker Engine
 - Disponible pour Windows Server 2016 sans cout additionnel
- ▶ Microsoft fournira aux utilisateurs de Windows Server 2016 un support enterprise pour le CS Docker Engine
 - Réalisé par Docker Inc au travers du support Microsoft
- ▶ Docker et Microsoft vont promouvoir conjointement Docker Datacenter pour permettre aux professionnels de l'informatique de sécuriser leur Windows Server software supply chain et de gérer les workloads Windows Server containerized, que ce soit on-prem, dans le cloud, ou hybride

Docker and Microsoft delivers integrated tooling across the application lifecycle



Build



Ship



Run



Visual Studio

Visual Studio Tools
for Docker



Docker for
Windows

Library of Microsoft
images on Docker Hub



Docker Datacenter for orchestration,
management and security



Microsoft Operations Management Suite
for hybrid cloud visibility and control



Windows Server

Docker containers available for Windows
Server running on any infrastructure



Microsoft
Hyper-V

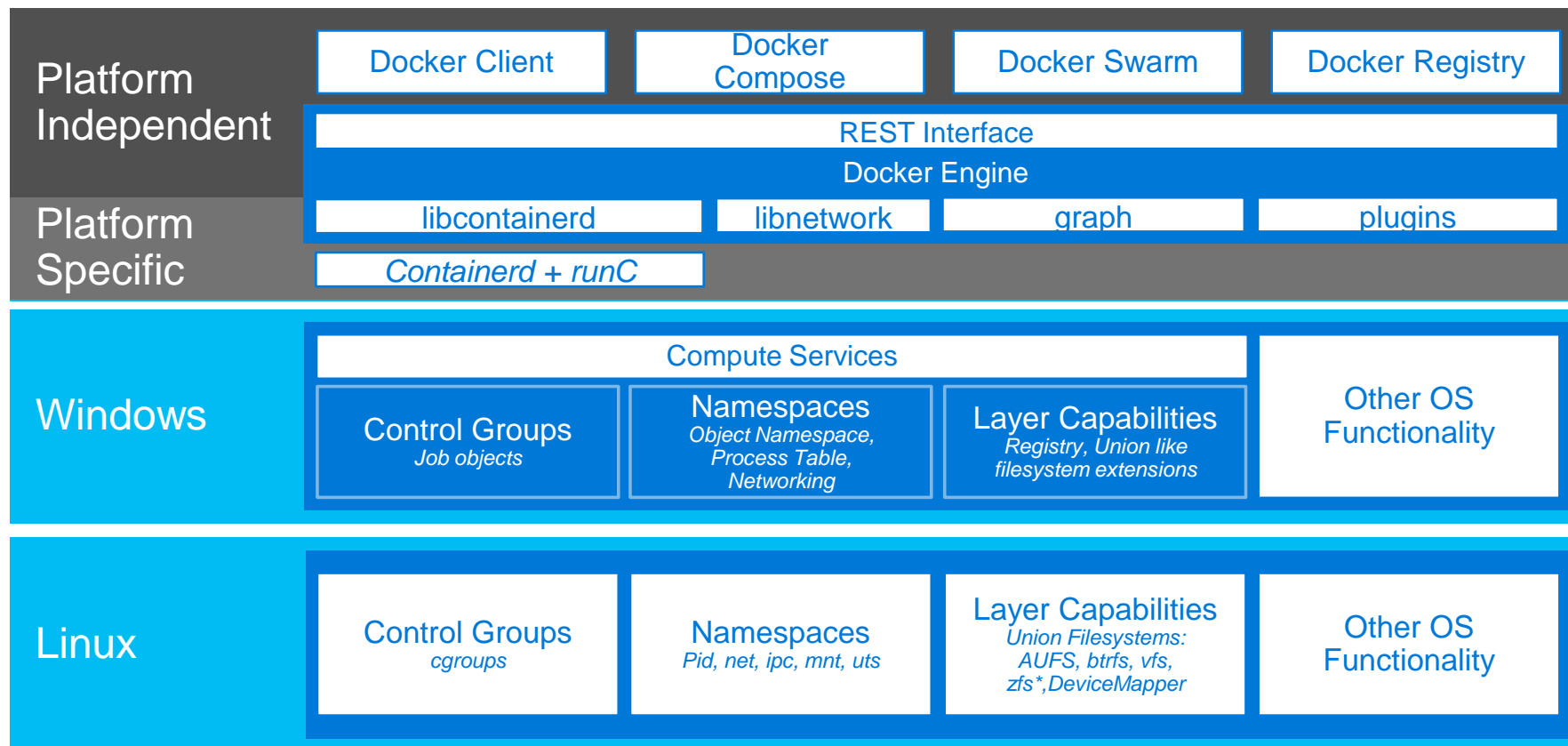


Demo Time

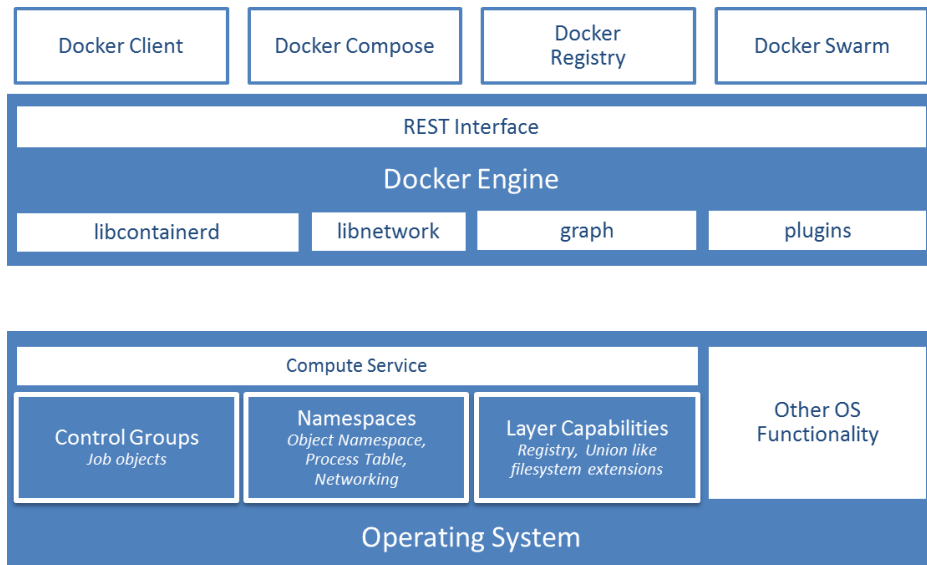


ARCHITECTURE

COMPARING OS ARCHITECTURES



- ▶ Interface publique aux containers
- ▶ Remplace containerd sur Windows
- ▶ Gère les containers en cours d'exécution
- ▶ Abstraction des capacités de bas niveau
- ▶ Language bindings disponibles
 - C#: <https://github.com/Microsoft/dotnet-computevirtualization>
 - Go: <https://github.com/Microsoft/hcsshim>



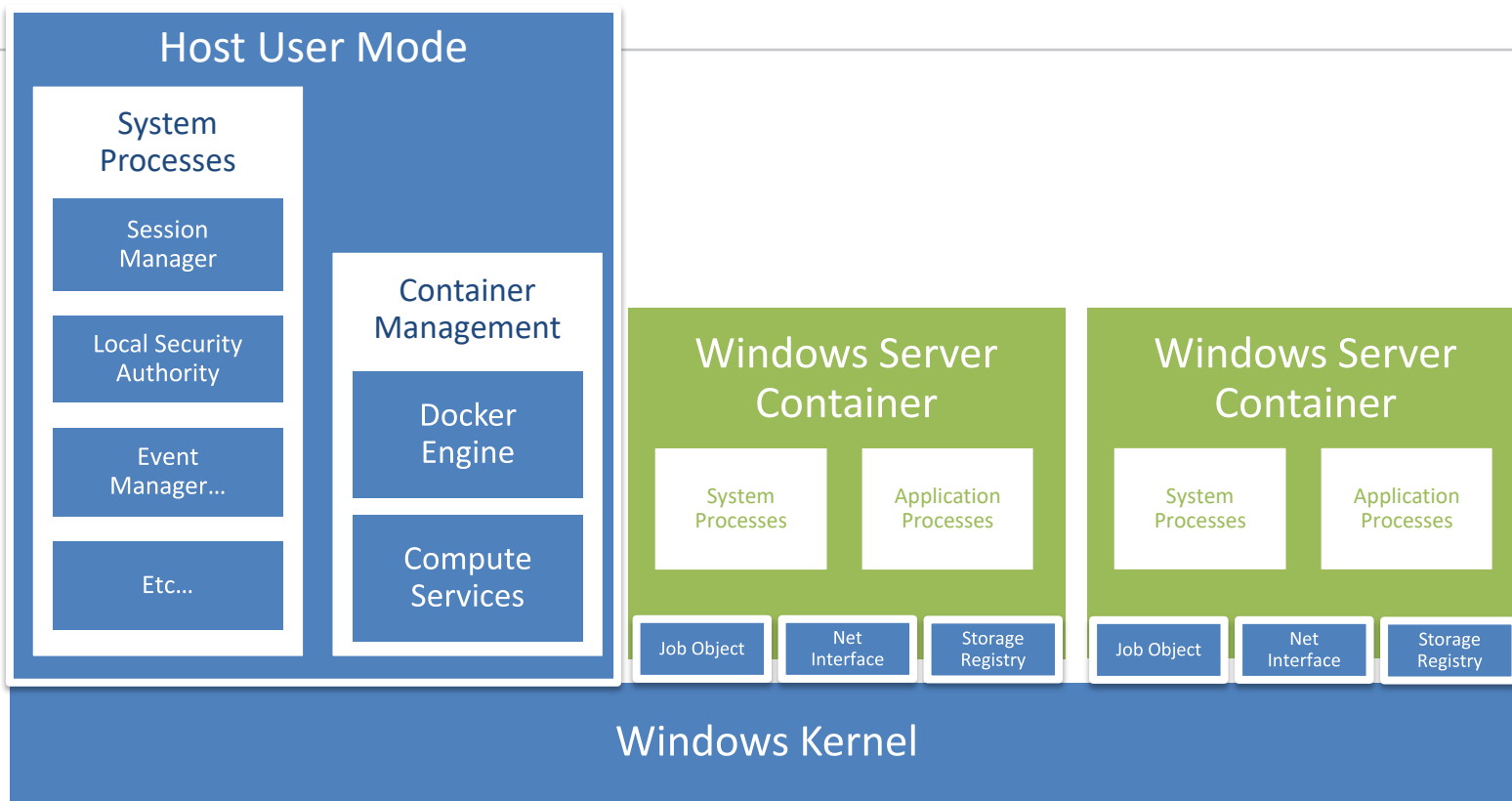
► Windows Server Containers

- Equivalent au fonctionnement dans les environnements Linux
- Application isolation using process and namespace isolation, and a shared kernel with the container host

► Hyper-V Containers

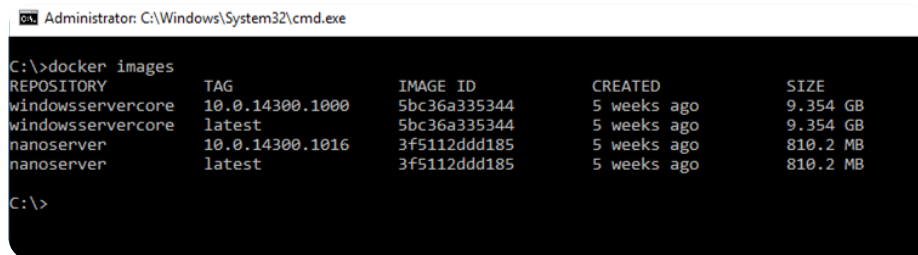
- Nouveau n'existe pas dans les environnements Linux
- Utilise Hyper-V
- Containers with a dedicated kernel and stronger isolation from other containers

WINDOWS SERVER CONTAINERS



- ▶ Les API Windows publiques sont fournies via DLLs, pas syscalls
- ▶ Beaucoup d'interdépendances
 - Très dépendant des services du système en cours d'exécution
 - RPC calls caches dans les API Win32
- ▶ Démare smss automatiquement
 - Equivalent à init
 - Lance une variété de services système
- ▶ Pas de "FROM scratch" possible

- Fournies par Microsoft
- 2 options
 - windowsservercore: large (énorme ?), hautement compatible
 - Runs most Windows Server apps, eg. IIS, SQL Server, full .NET Framework
 - nanoserver: petit, rapide, smaller API surface
 - New minimal Windows version, runs IIS and .NET Core
- docker pull
 - microsoft/windowsservercore
 - microsoft/nanoserver



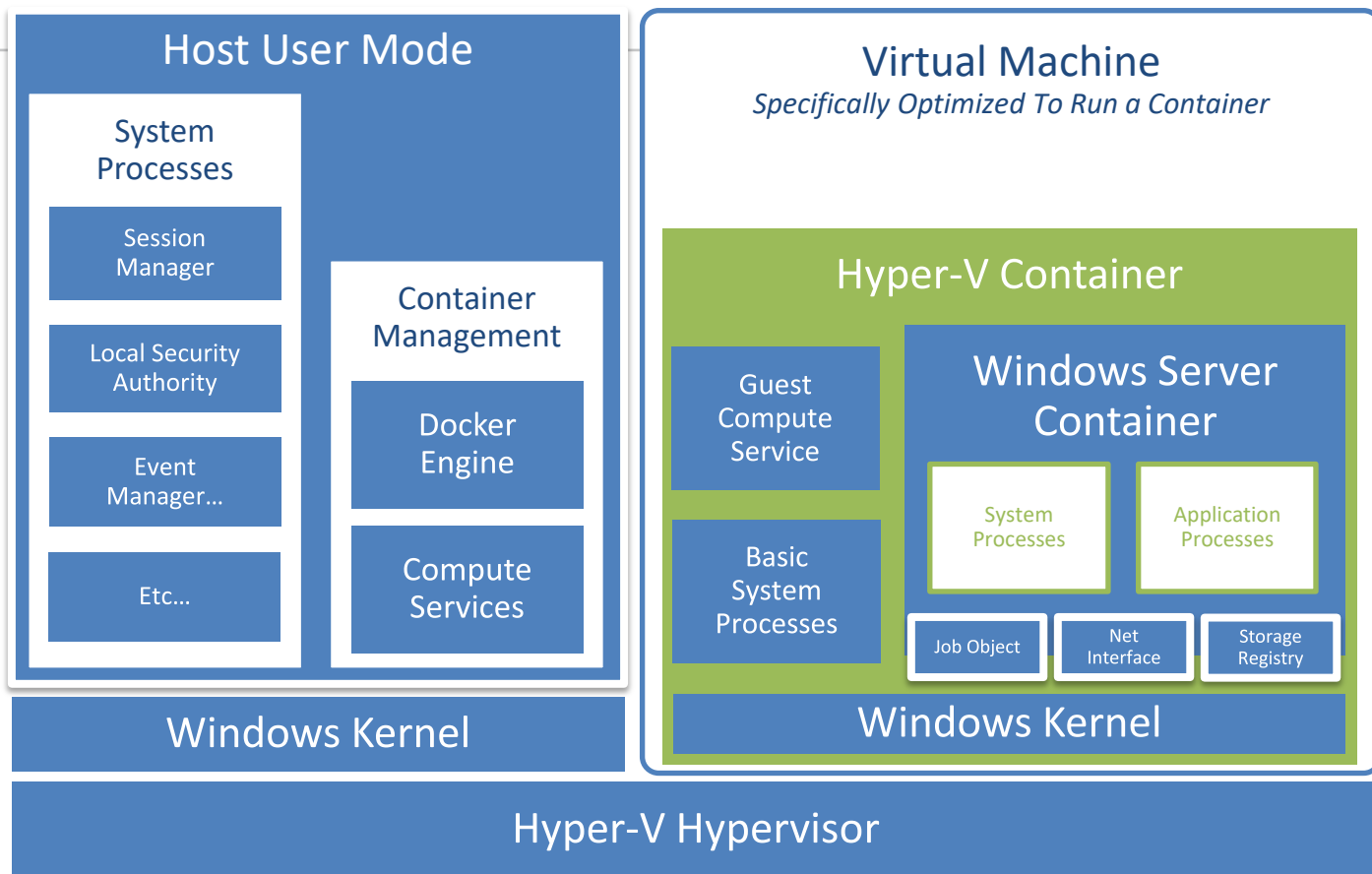
```
Administrator: C:\Windows\System32\cmd.exe
C:\>docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
windowsservercore   10.0.14300.1000    5bc36a335344        5 weeks ago         9.354 GB
windowsservercore   latest              5bc36a335344        5 weeks ago         9.354 GB
nanoserver           10.0.14300.1016    3f5112ddd185        5 weeks ago         810.2 MB
nanoserver           latest              3f5112ddd185        5 weeks ago         810.2 MB
C:\>
```

Demo Time



- ▶ Certains workload ont besoin de plus d'isolement
 - Hostile multi-tenancy
 - Regulated workloads
- ▶ Solution : exécuter de façon transparente chaque container dans une VM !
- ▶ (pratiquement) invisible pour Docker et les utilisateurs
 - `docker run --isolation=hyperv`
 - Hyper-V Containers sont la valeur par défaut sur Windows 10
- ▶ Les Images sont identiques

HYPER-V CONTAINERS



- ▶ Petit, stateless “utility VM”
 - Le plus petit Windows jusqu’à présent ?
 - Ecritures non persistantes
- ▶ Stockage attaché via SMB
 - VMBus transport
 - File cache sharing
- ▶ Réseau attaché via virtual NIC

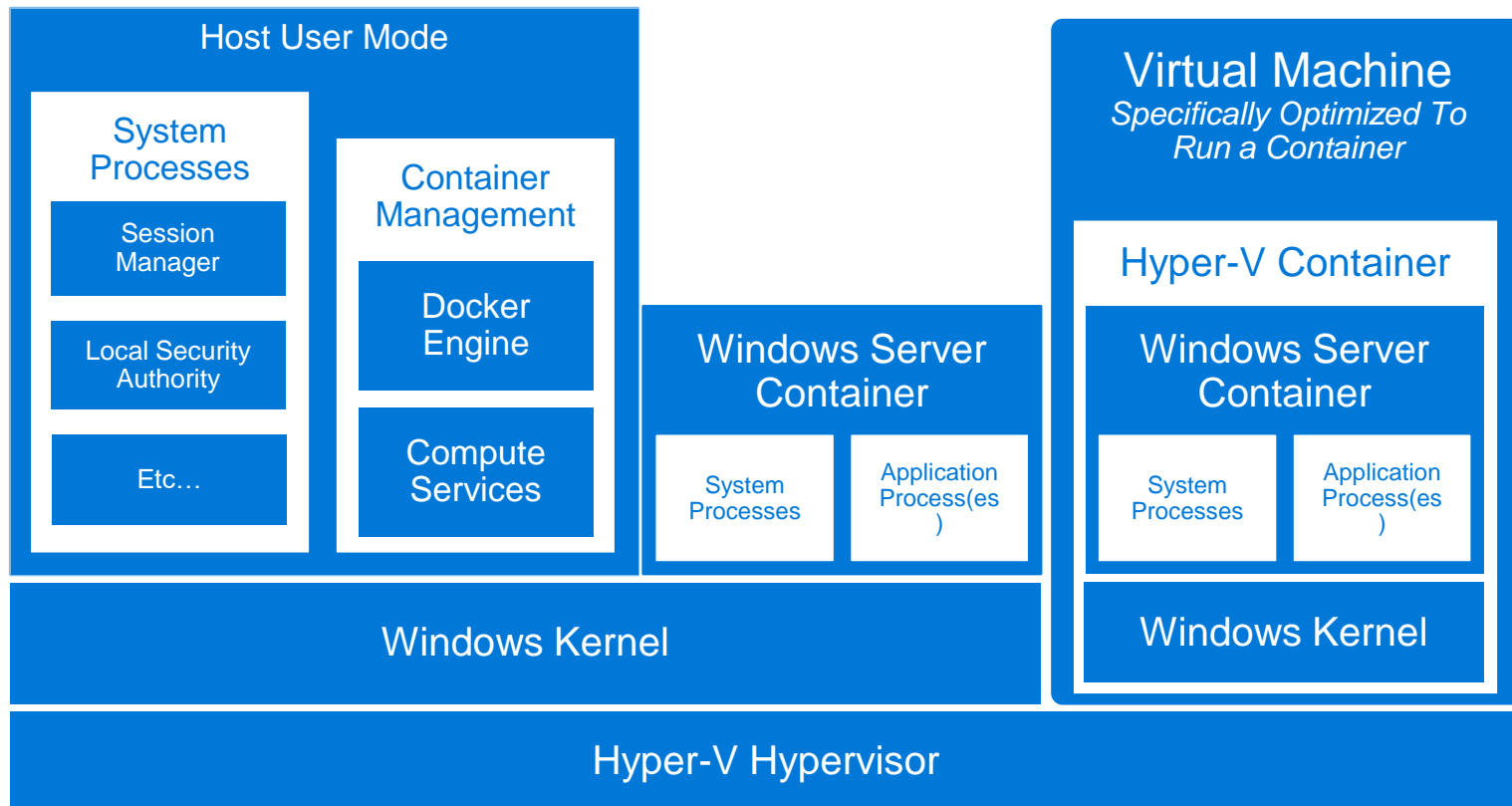
- ▶ Le lancement de l'Utility VM prend du temps et de la mémoire
- ▶ Le faire une fois, geler le résultat
- ▶ Fork de la VM pour chaque nouvelle instance
 - Élimine le temps de démarrage
 - Partage de la mémoire

Demo Time



POUR FINIR

WINDOWS AND HYPER-V CONTAINERS



MERCI
QUESTIONS / REPONSES

ANNEXES ECOSYSTÈME & TOOLS

- ▶ Environnement poste de développement
- ▶ Docker Toolbox
 - Docker Host Linux sur VM VirtualBox
- ▶ OS/Arch supporté
 - linux/amd64

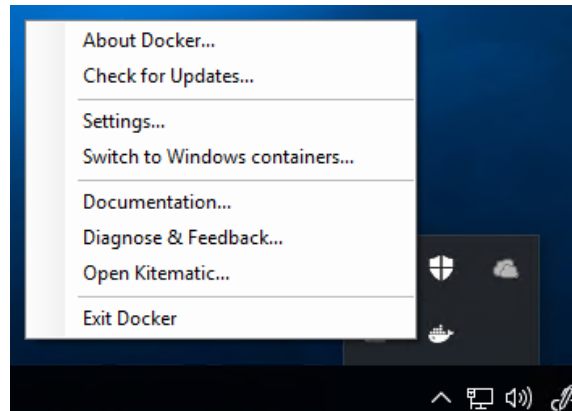
► Environnement poste de développement

► Docker for Windows

- <https://docs.docker.com/docker-for-windows/>
- <https://docs.docker.com/docker-for-windows/release-notes/>
- En date du 6/11/16 c'est à partir de la beta 26 que le switch est possible en clic droit
- Pas encore disponible sur les version stables




► OS/Arch supporté

- linux/amd64
- windows/amd64



- ▶ Environnement serveur de production
- ▶ CS Docker Engine sur Compute Service
- ▶ OS/Arch supporté
 - windows/amd64

- ▶ Infrastructure as Code avec Configuration Management possible
- ▶ Avec PoSH DSC (PowerShell Desired State Configuration)
- ▶ Blog Post
 - <https://dscottraynsford.wordpress.com/2016/10/15/install-docker-on-windows-server-2016-using-dsc/>
- ▶ Disponible sur la PowerShell Gallery
 - <https://www.powershellgallery.com/packages/Install-DockerOnWS2016UsingDSC/>

▶ 	Docker on Ubuntu Server	Canonical + Microsoft	Compute
▶	OS/Arch supporté <ul style="list-style-type: none">• linux/amd64		
▶ 	Windows Server 2016 Datacenter - with Containers	Microsoft	Compute
▶	OS/Arch supporté <ul style="list-style-type: none">• windows/amd64		
▶ 	Windows Server 2016 - Nano Server	Microsoft	Compute
▶	OS/Arch supporté <ul style="list-style-type: none">• windows/amd64		



Azure Container Service

Microsoft

Compute

▶ Technologie d'Orchestration disponible

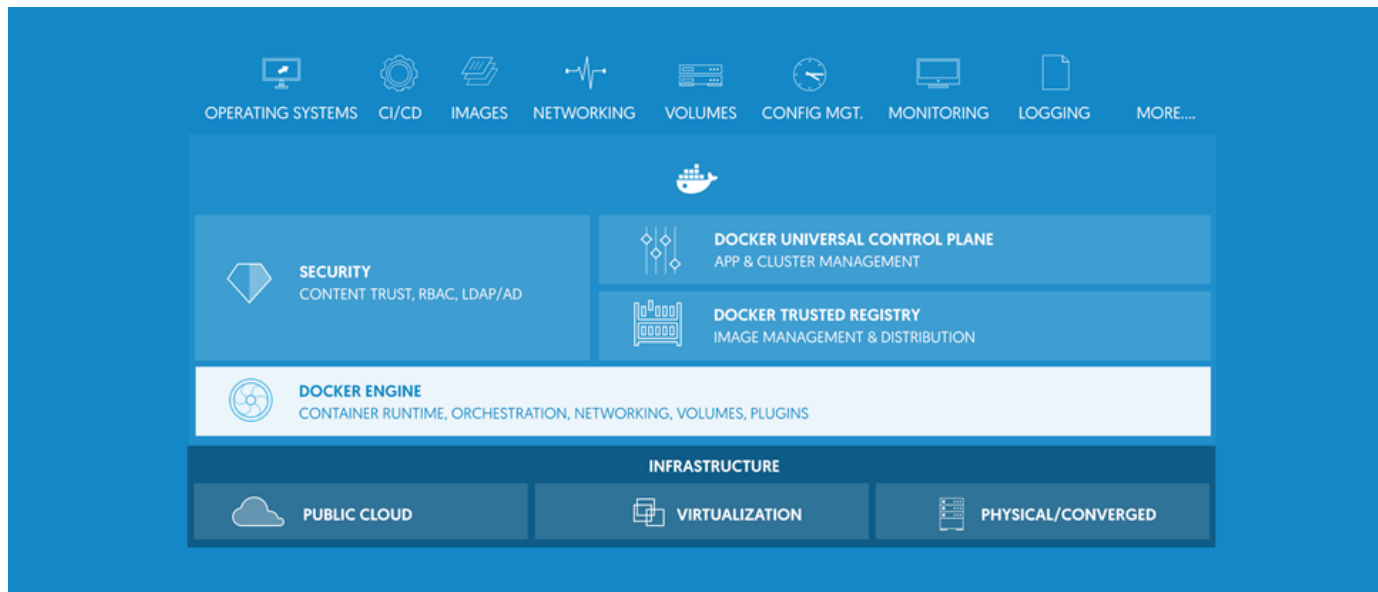
- Docker Swarm
- DC/OS
- Kubernetes

▶ OS/Arch supporté

- linux/amd64

- ▶ À base de Docker Swarm
- ▶ OS/Arch supporté
 - linux/amd64

- ▶ Docker Inc full stack (UCP / Docker Engine Nodes / DTR)
- ▶ OS/Arch supporté
 - linux/amd64
 - windows/amd64



[DOCKER] DOCKER DATACENTER SUR AZURE



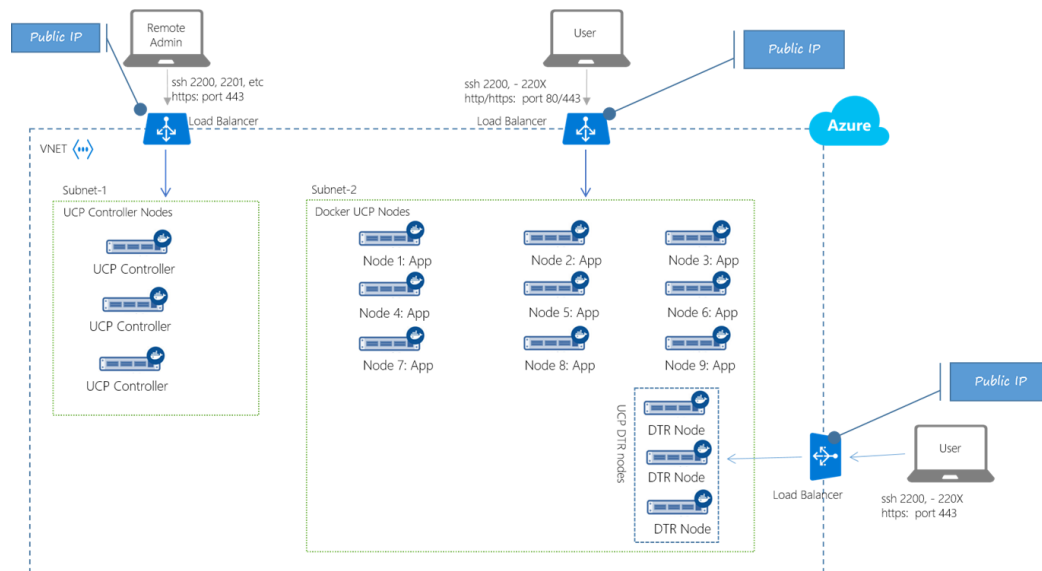
Docker Datacenter

Docker, Inc.

Compute

- Docker Datacenter hébergé sur Azure
- Déploiement à l'aide d'un template
- BYOL

Docker Datacenter Architecture



- ▶ Outil (PowerShell Module) communautaire (Open Source) pour simplifier la création de Dockerfile
- ▶ Objectif : migrer une VM Windows Legacy vers un Windows Container Dockerfile
- ▶ On part d'un VHD, VHDX, ou fichier WIM et l'outil cherche les composants installés (artefacts) et crée un fichier Dockerfile en sorti

```
### List out supported artifacts  
Get-WindowsArtifacts
```

```
### Perform scan of Windows source image  
ConvertTo-Dockerfile -ImagePath c:\docker\myimage.wim
```



THANK YOU