4tecture
empower your software solutions

ASP.NET Core mit VS 2017und Docker

Marc Müller
Principal Consultant

marc.mueller@4tecture.ch
@muellermarc
www.4tecture.ch

MVP Microsoft Most Valuable Professional

DIGICOMP

Microsoft

NOSER ENGINEERING
WE KNOW HOW

progile
productive not busy

trivadis
makes IT easier.

OFFERING SOLUTIONS SOFTWARE

Marc Müller
Principal Consultant
@muellermarc

Microsoft Most Valuable Professional

4tecture©
empower your software solutions

Agenda

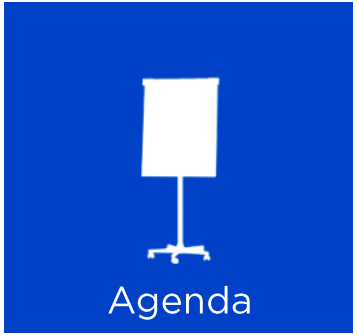
Intro

Container für Web Applikationen

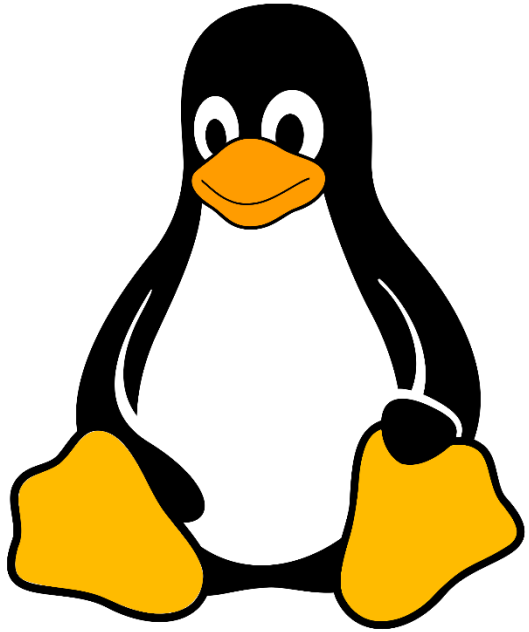
Docker Support in Visual Studio

Docker CI Build

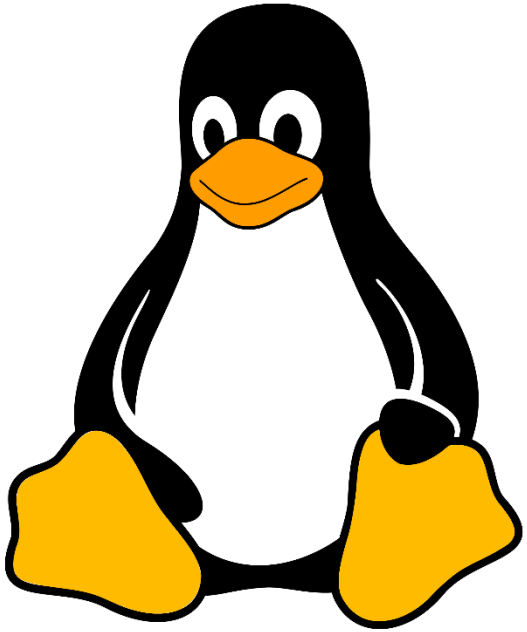
Docker Hosting



Microsoft

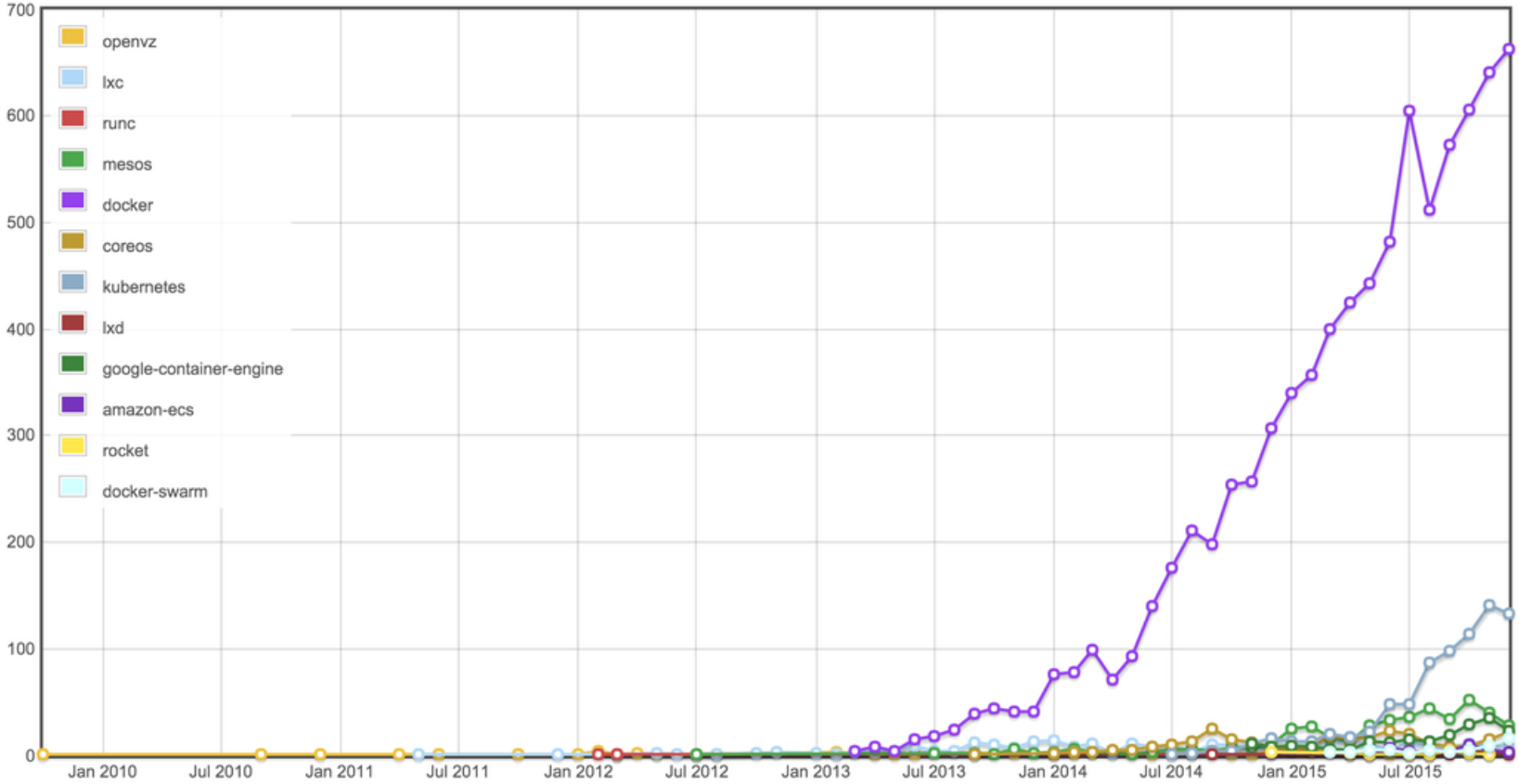


I like



…but I’m a Microsoft Fan-Boy

Docker?

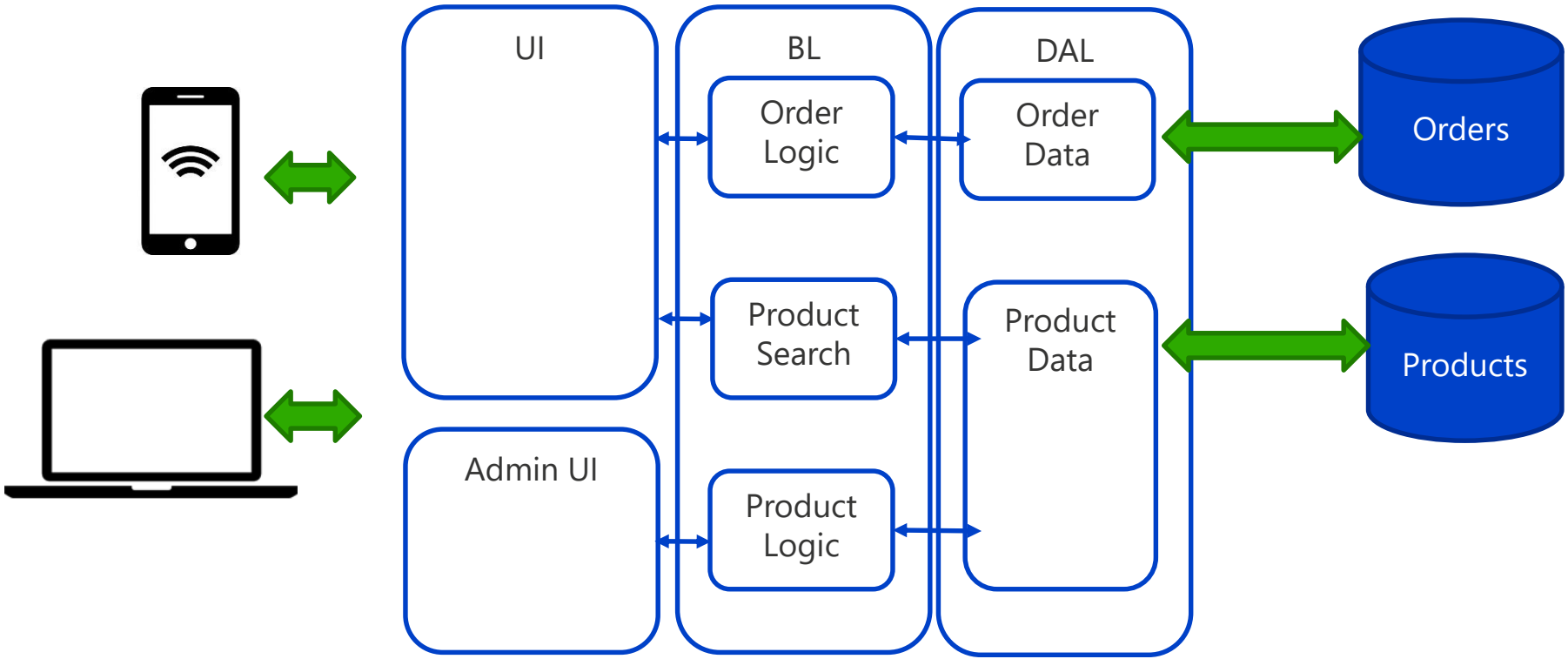


https://twitter.com/dberkholz/status/689211852157407233

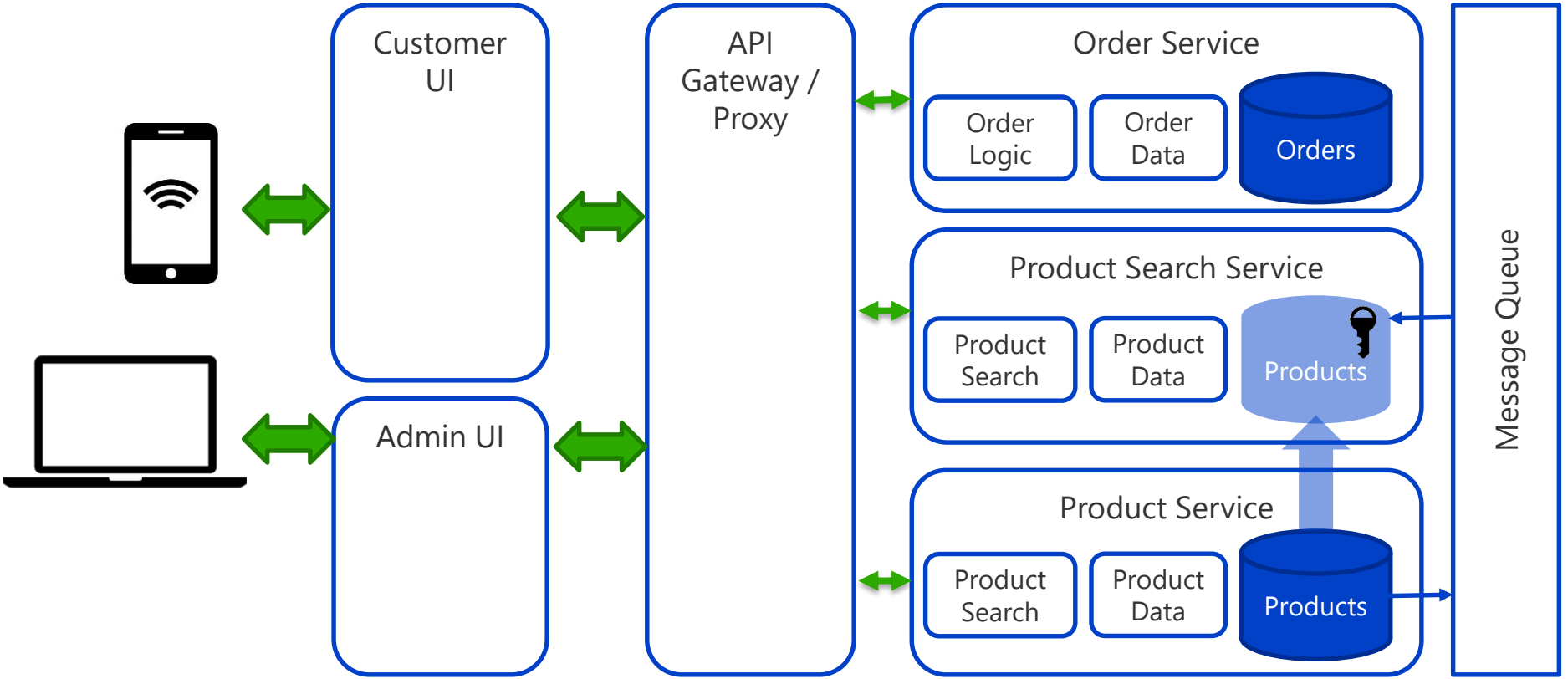

100 Deployments
pro Tag!


ReactionGIFS.me

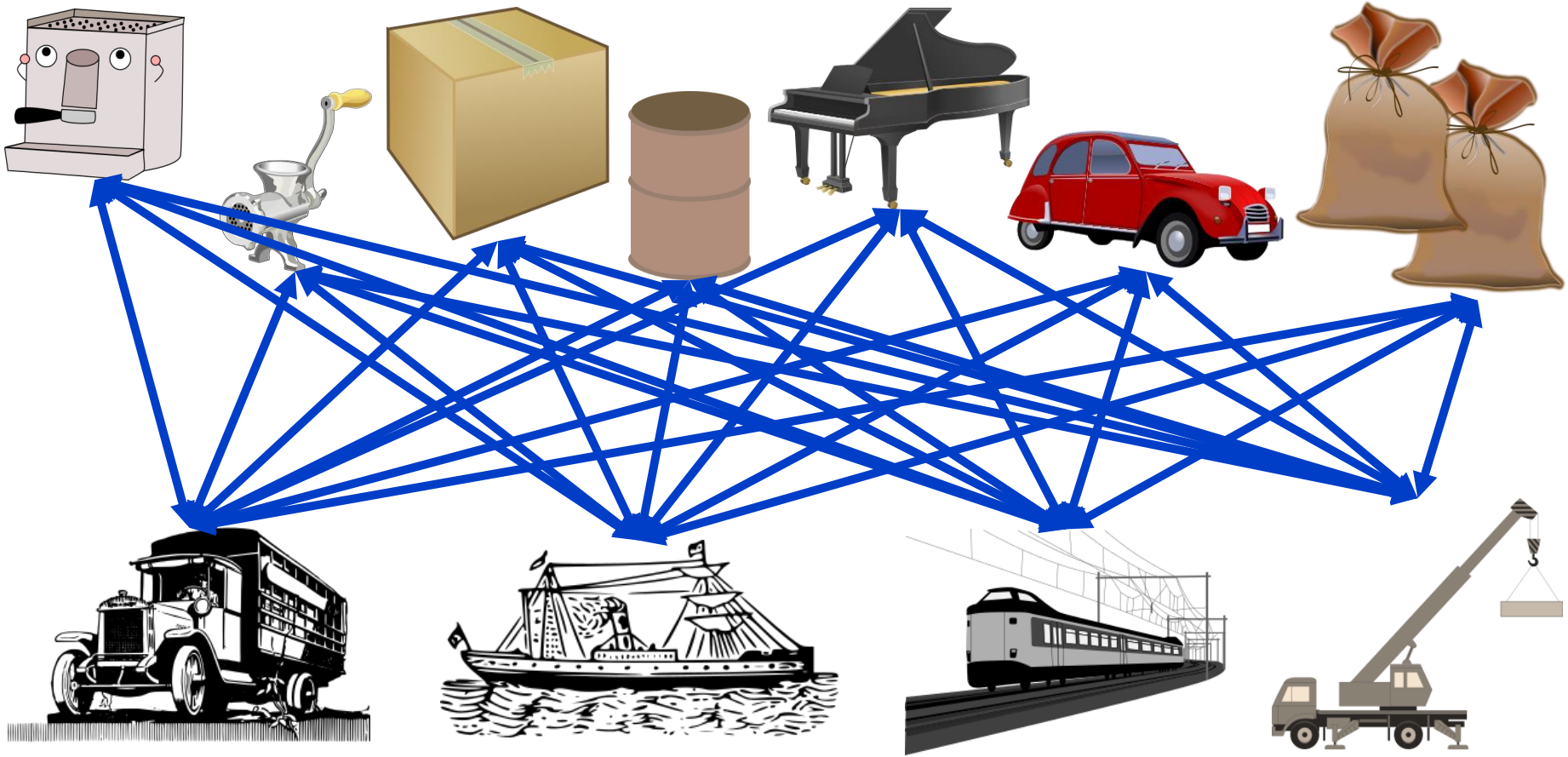
Monolithische Architektur



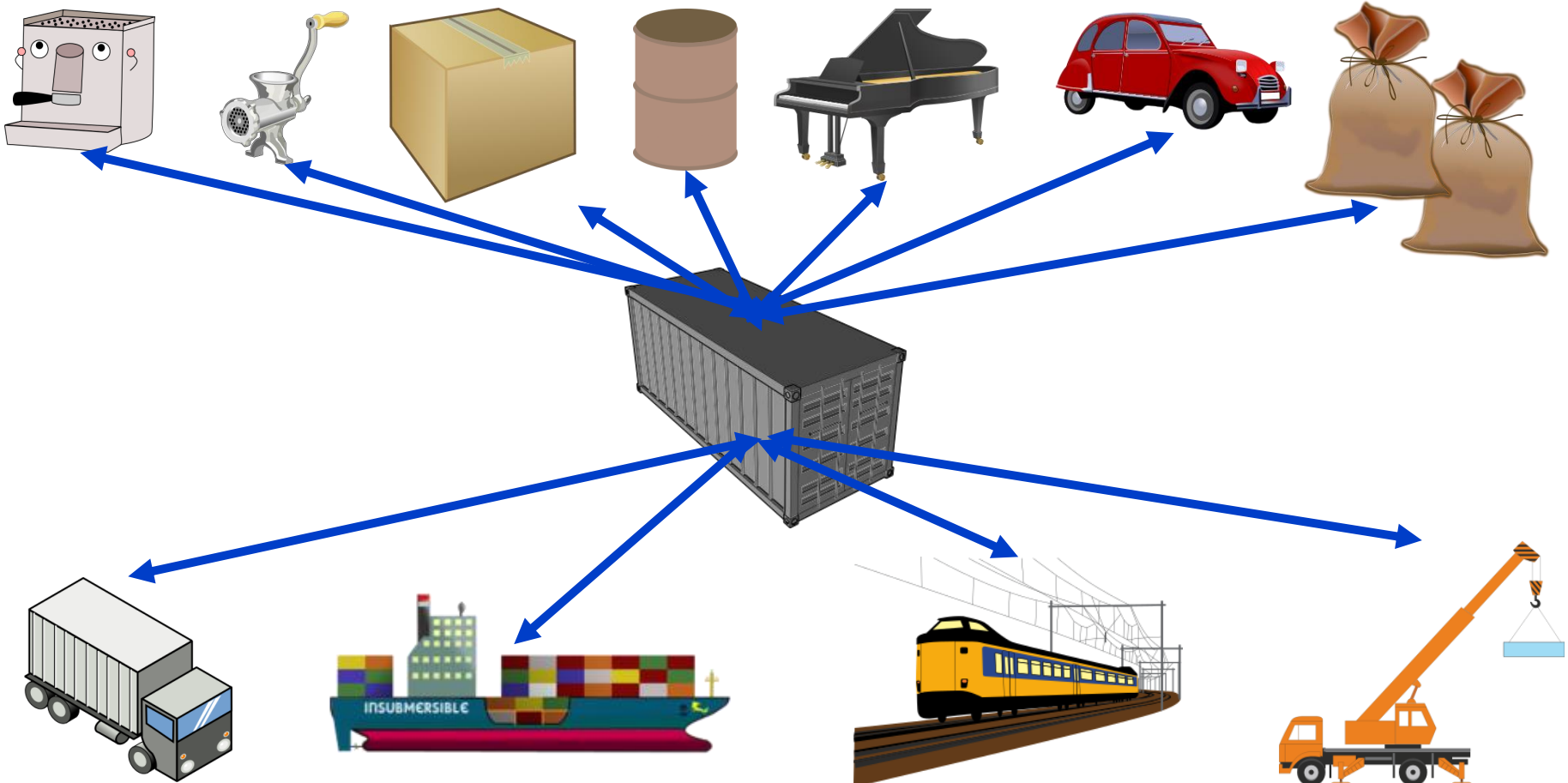
Micro-Service Architektur



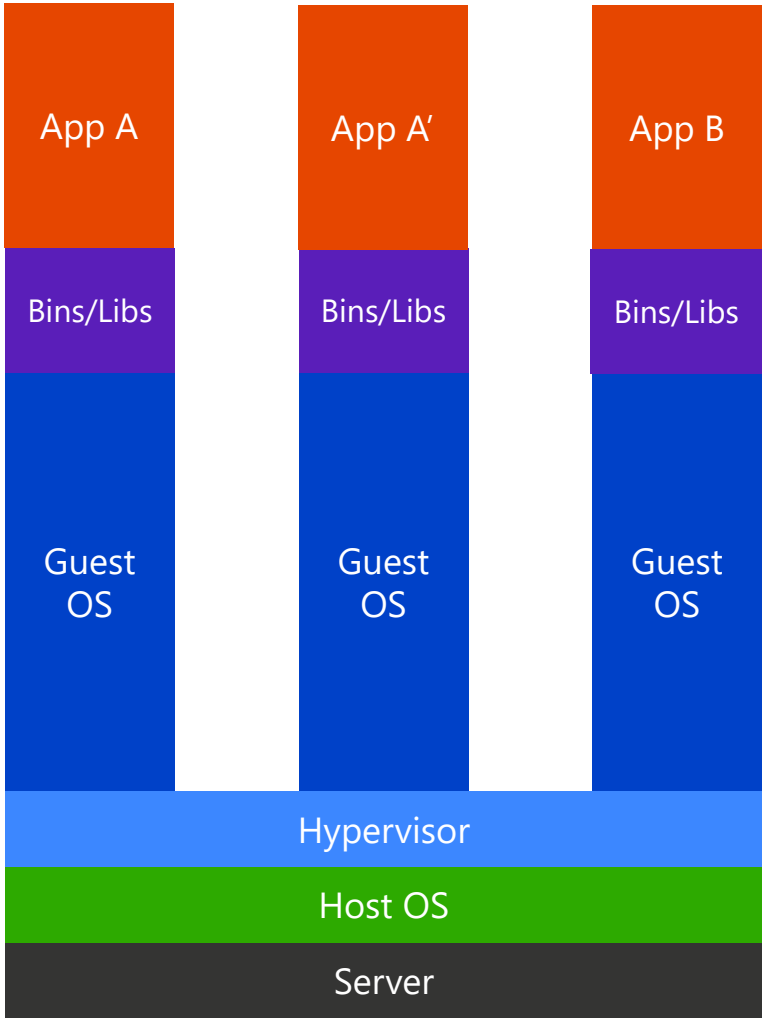
Güterverkehr vor 1960



Lösung: Frachtcontainer




Container für Web Applikationen
4tecture
empower your software solutions



Containers are isolated, but share OS and, where appropriate, bins/libraries

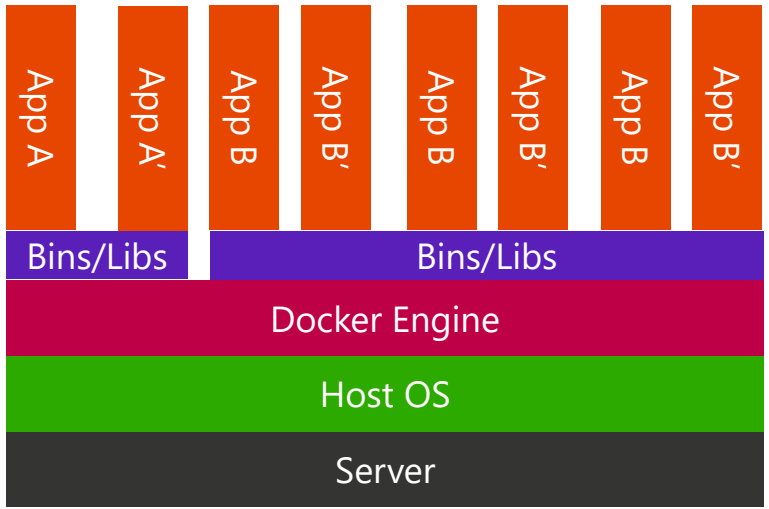
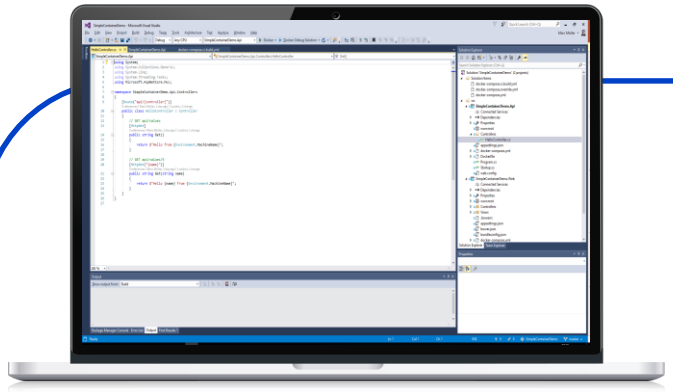


Image Source: https://sec.ch9.ms/sessions/build/2016/B822.pptx

Docker Layers

- My ASP.NET Core Application
- microsoft/aspnet
- microsoft/dotnet
- buildpack-deps:jessie-scm
- buildpack-deps:jessie-curl
- debian:jessie

Environments - Development



Developer PC / Laptop

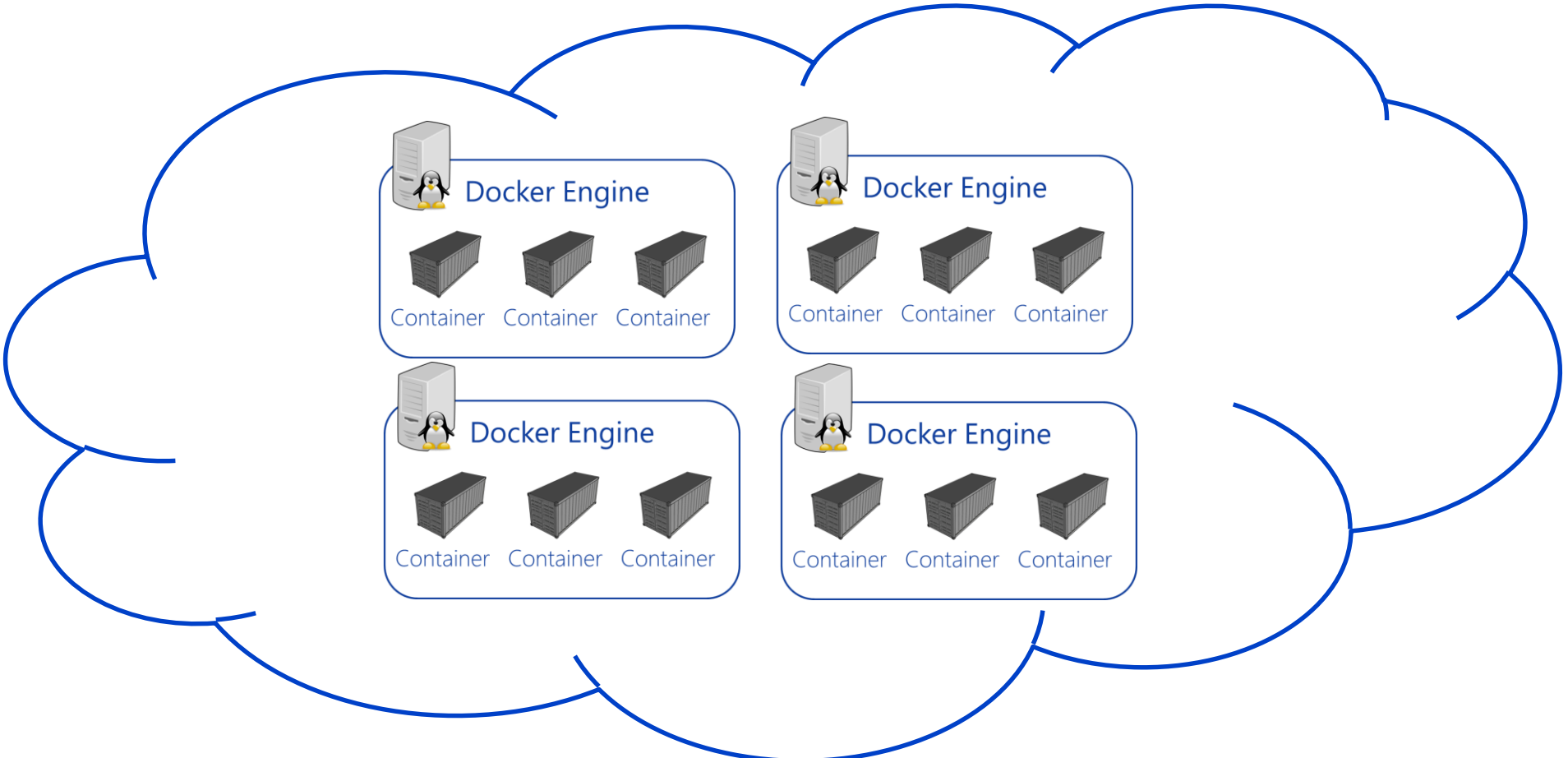
Docker Engine

Container

Container

Container

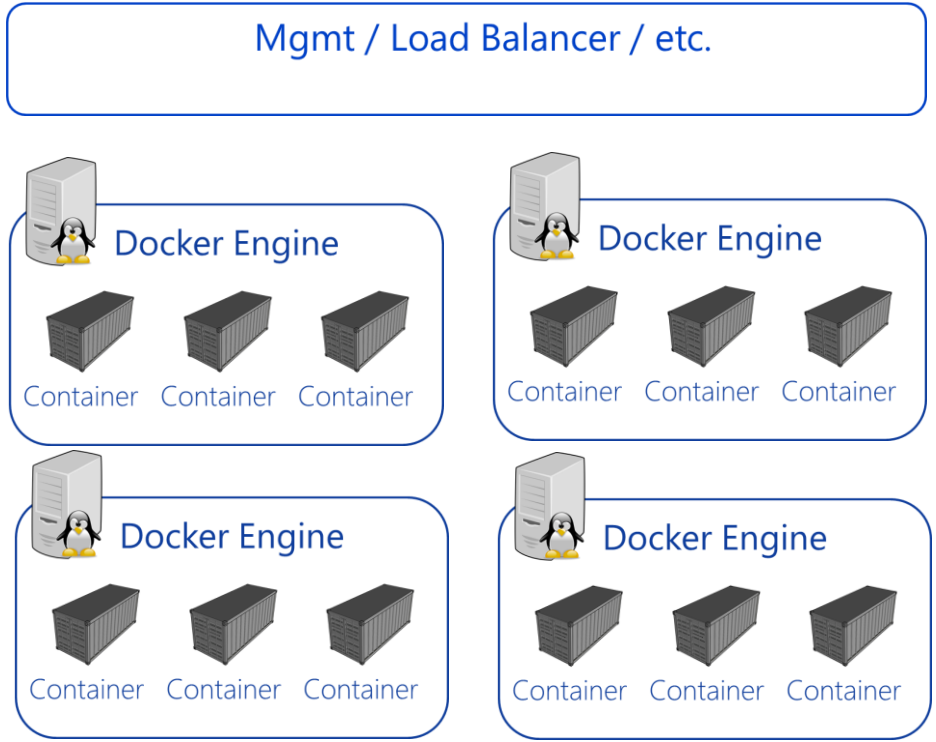
Environments – Hosting / Cloud



Cluster Management

App1

App2



Vorteile von Containern

- Schnelles Deployment
- Schneller Start
- Einfaches Scaling und Partitioning
- Isolierter, versionierter, wiederverwendbarer Code (Artefakt / Konfigurationselement)

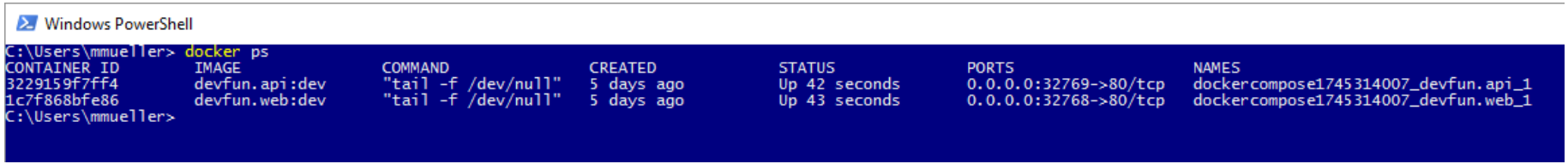
Klare Abgrenzung zwischen Entwicklung und Betrieb

- Entwicklung kümmert sich um den Inhalt des Containers
- Betrieb kümmert sich darum, wie der Container verwendet wird

Docker Support in Visual Studio

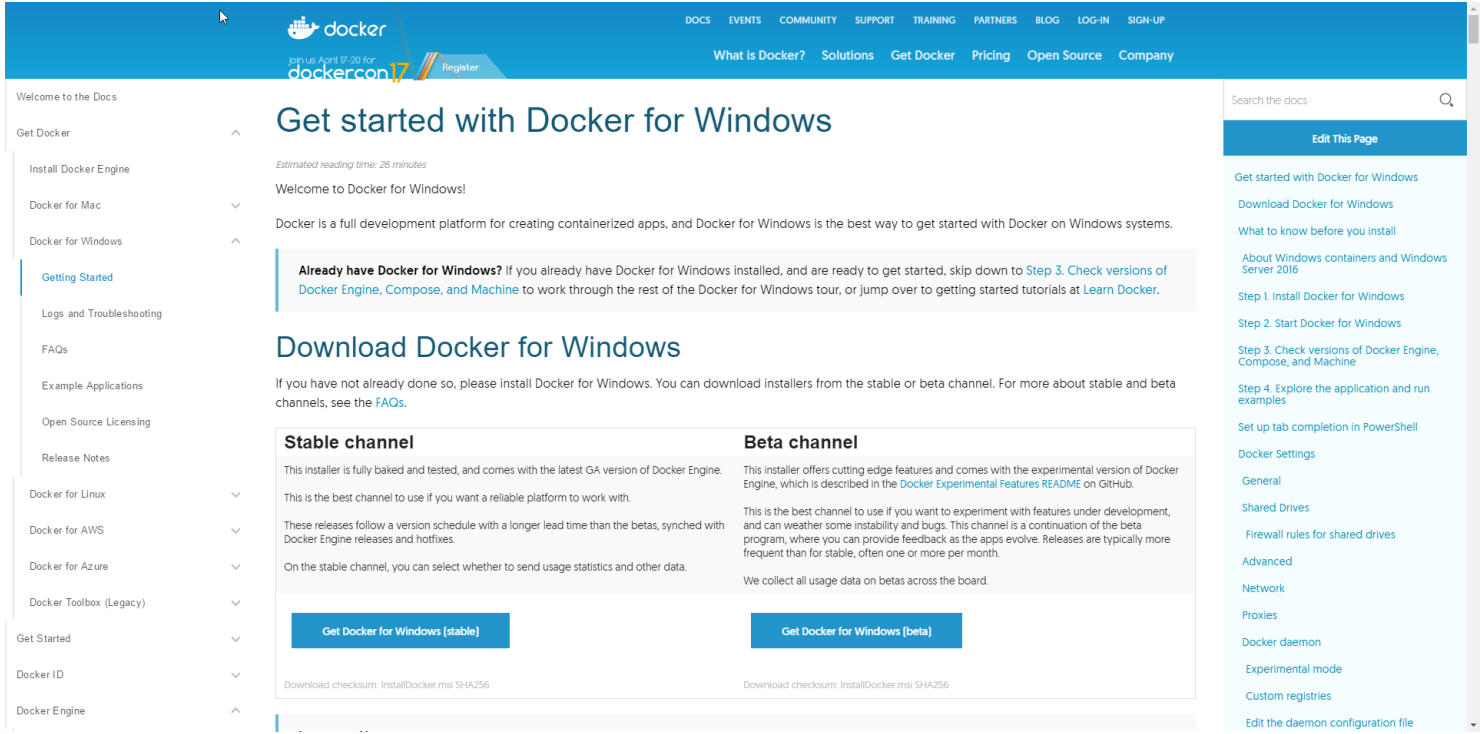
4tecture
empower your software solutions

Docker CLI

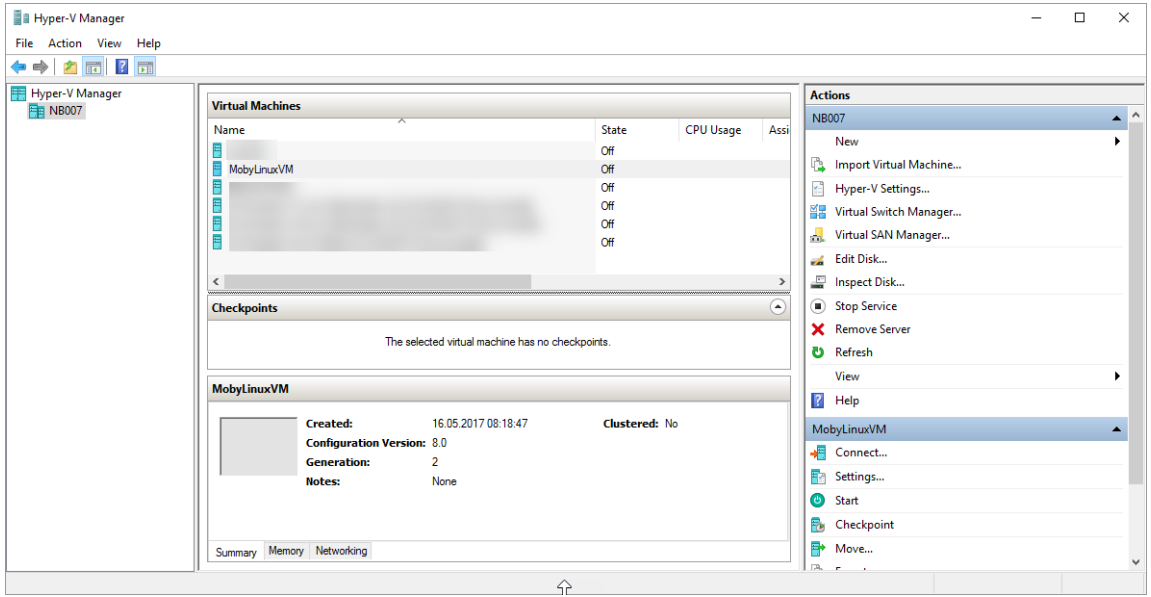

Windows PowerShell
C:\Users\mmueller> docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
3229159f7ff4 devfun.api:dev "tail -f /dev/null" 5 days ago Up 42 seconds 0.0.0.0:32769->80/tcp dockercompose1745314007_devfun.api_1
1c7f868bfe86 devfun.web:dev "tail -f /dev/null" 5 days ago Up 43 seconds 0.0.0.0:32768->80/tcp dockercompose1745314007_devfun.web_1
C:\Users\mmueller>

Docker for Windows

https://docs.docker.com/docker-for-windows/

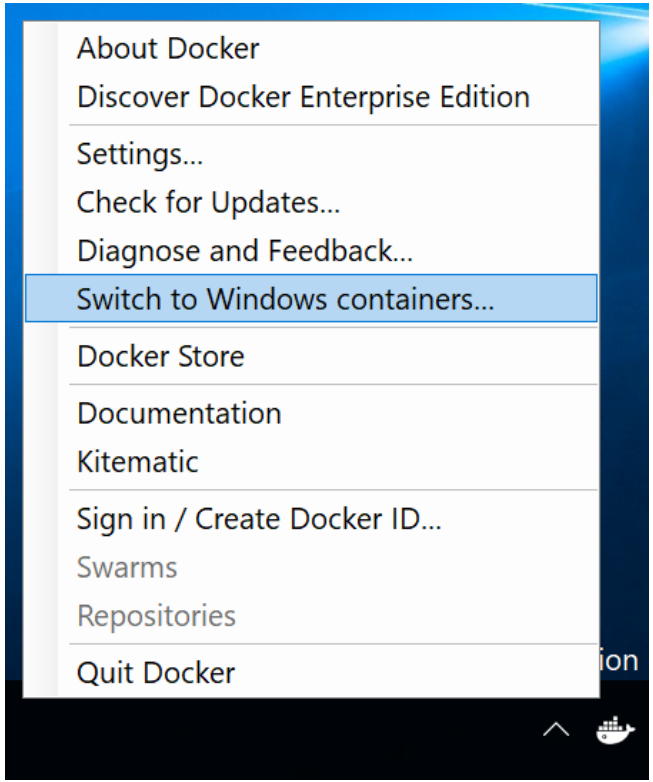


Linux Container (Default)



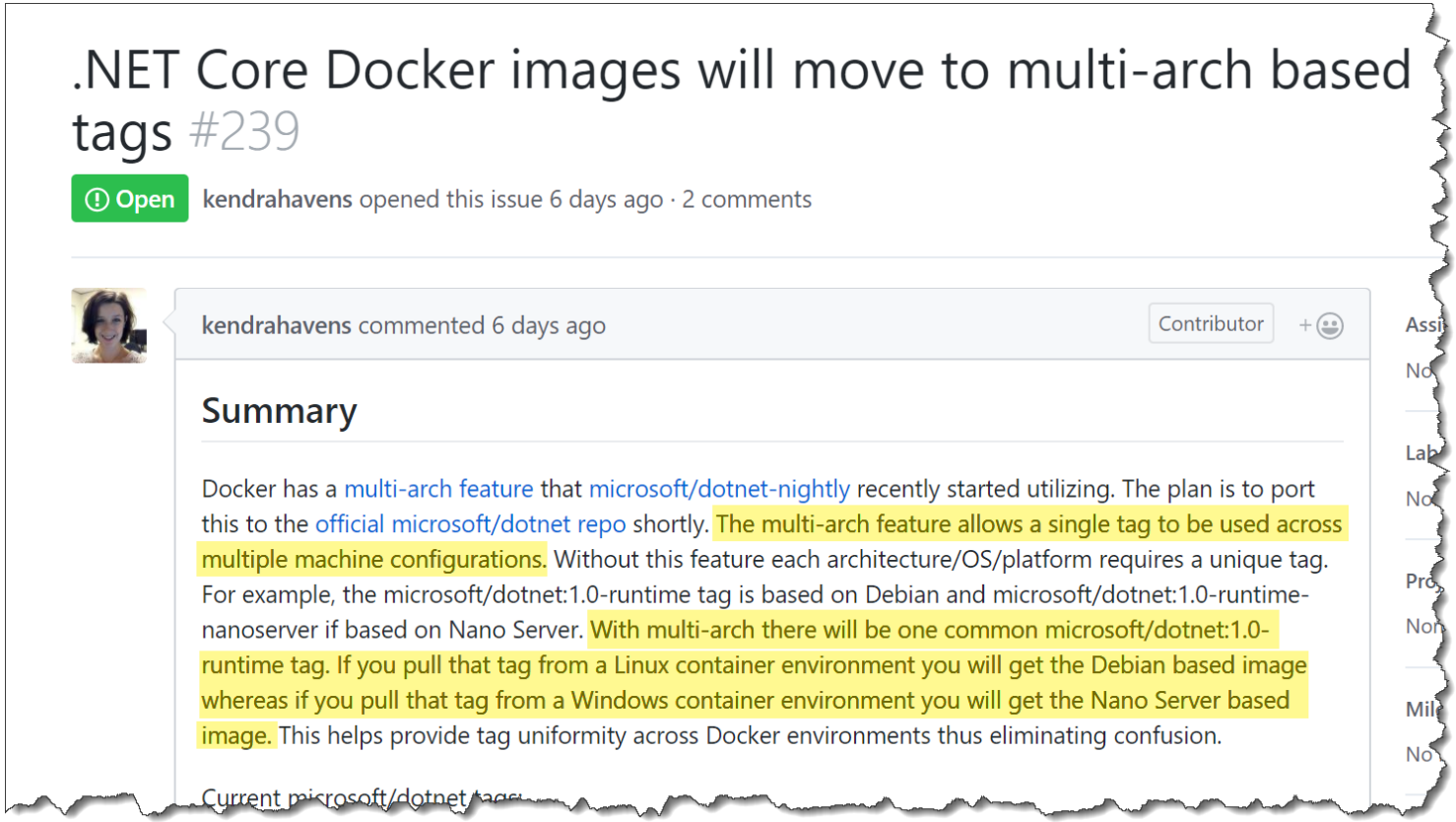
com.docker.proxy.exe
//./pipe/docker_engine

Windows Container



//./pipe/docker_engine_windows

Multi-Arch based Tags



.NET Core Docker images will move to multi-arch based tags #239

Open **kendrahavens** opened this issue 6 days ago · 2 comments

kendrahavens commented 6 days ago — Contributor

Summary

Docker has a multi-arch feature that microsoft/dotnet-nightly recently started utilizing. The plan is to port this to the official microsoft/dotnet repo shortly. The multi-arch feature allows a single tag to be used across multiple machine configurations. Without this feature each architecture/OS/platform requires a unique tag. For example, the microsoft/dotnet:1.0-runtime tag is based on Debian and microsoft/dotnet:1.0-runtime-nanoserver if based on Nano Server. With multi-arch there will be one common microsoft/dotnet:1.0-runtime tag. If you pull that tag from a Linux container environment you will get the Debian based image whereas if you pull that tag from a Windows container environment you will get the Nano Server based image. This helps provide tag uniformity across Docker environments thus eliminating confusion.

Current microsoft/dotnet tags:

https://github.com/dotnet/dotnet-docker/issues/239

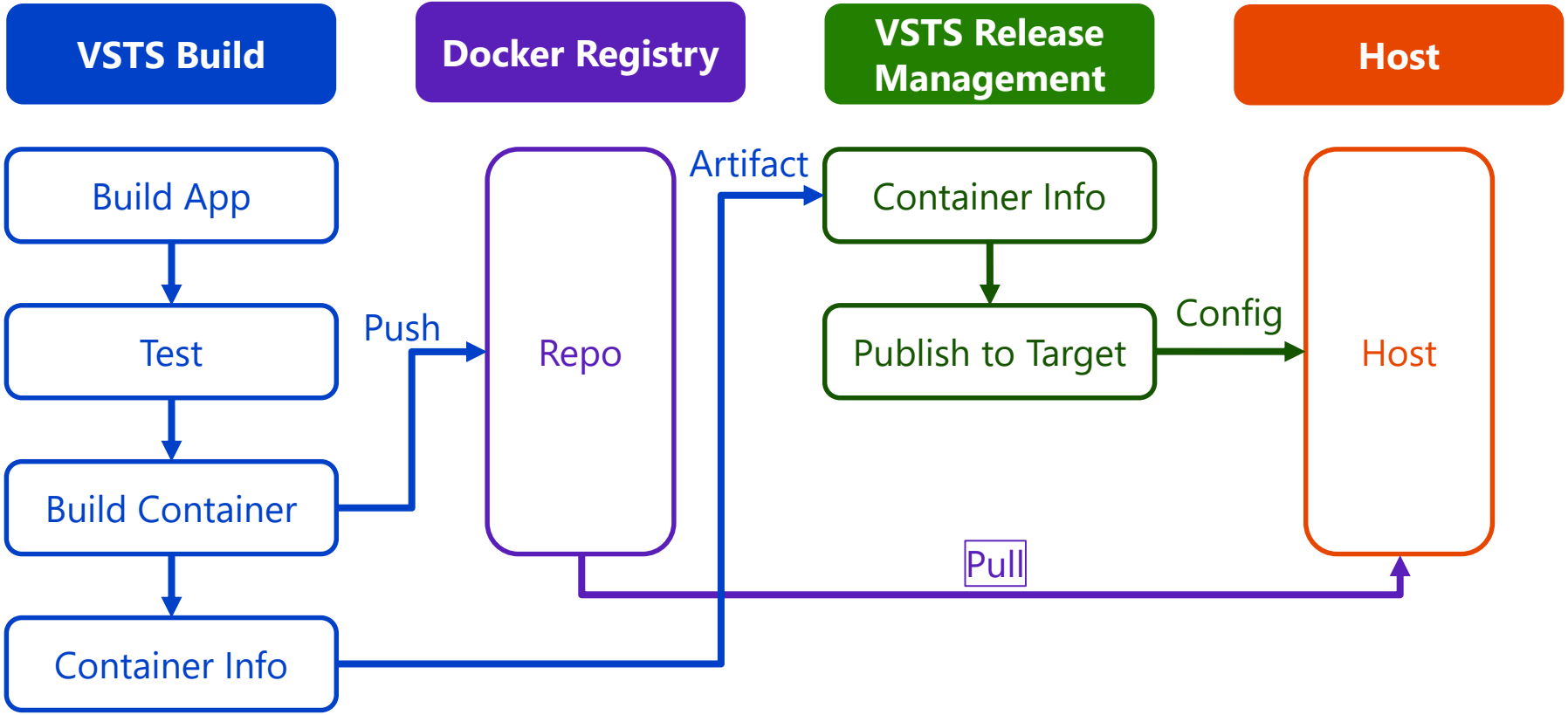
Demo

- Docker for Windows
- Visual Studio Docker Support
- Proxy Container

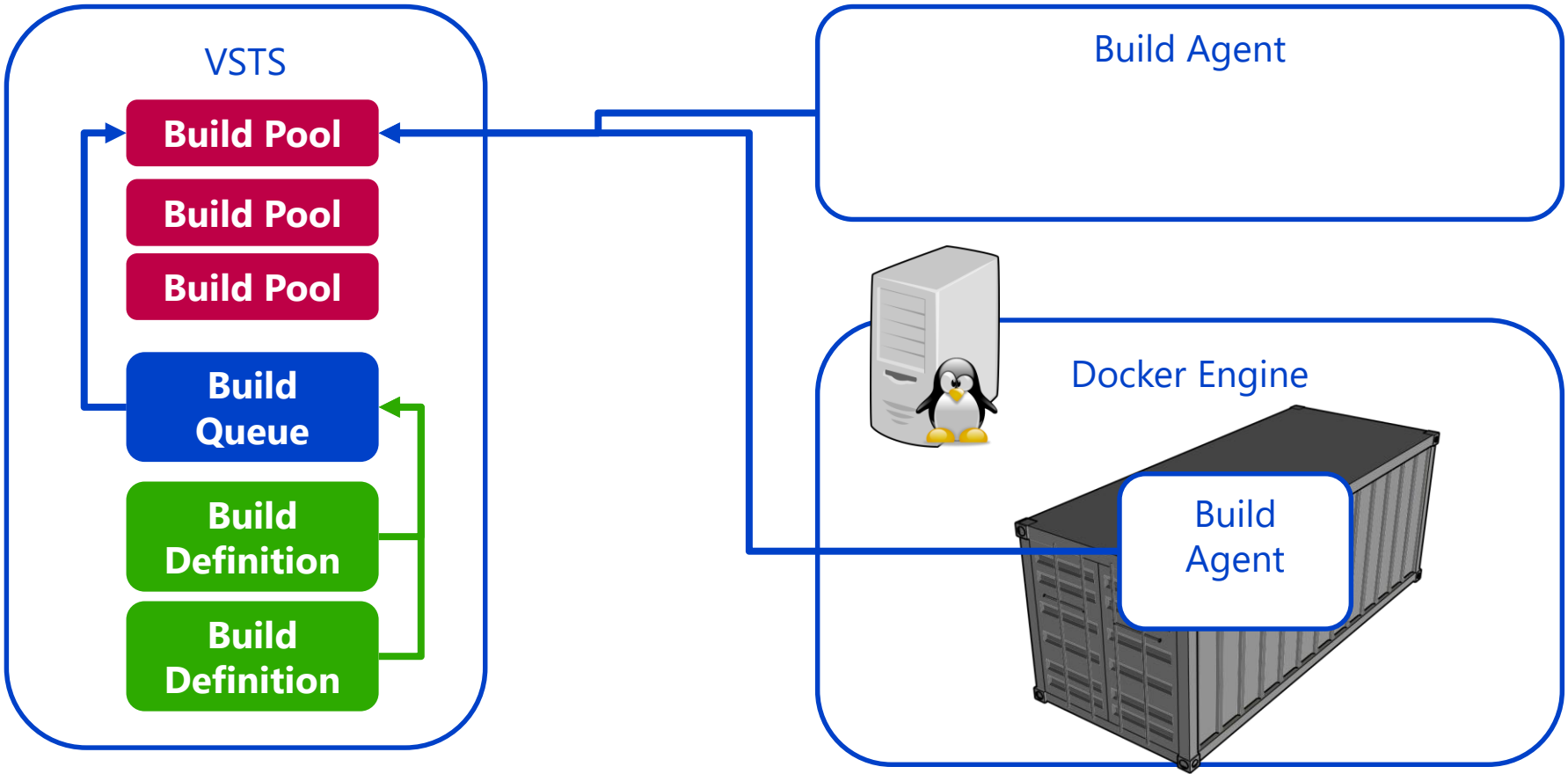
Docker CI Build

4tecture
empower your software solutions

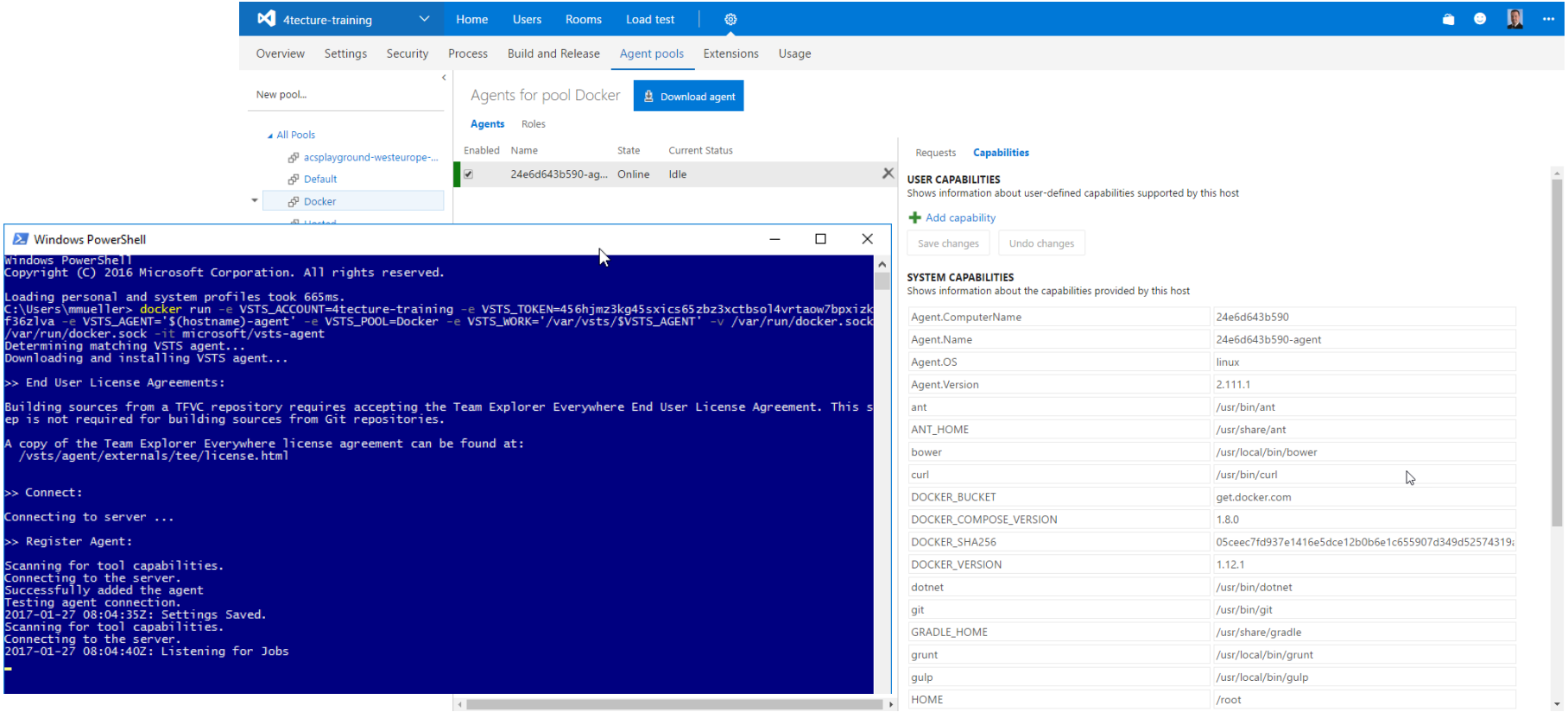
Docker CI / CD


VSTS Build
Docker Registry
VSTS Release Management
Host
Build App
Test
Build Container
Container Info
Push
Repo
Artifact
Container Info
Publish to Target
Config
Host
Pull

VSTS / TFS Build Agent



Custom Docker Build Agent

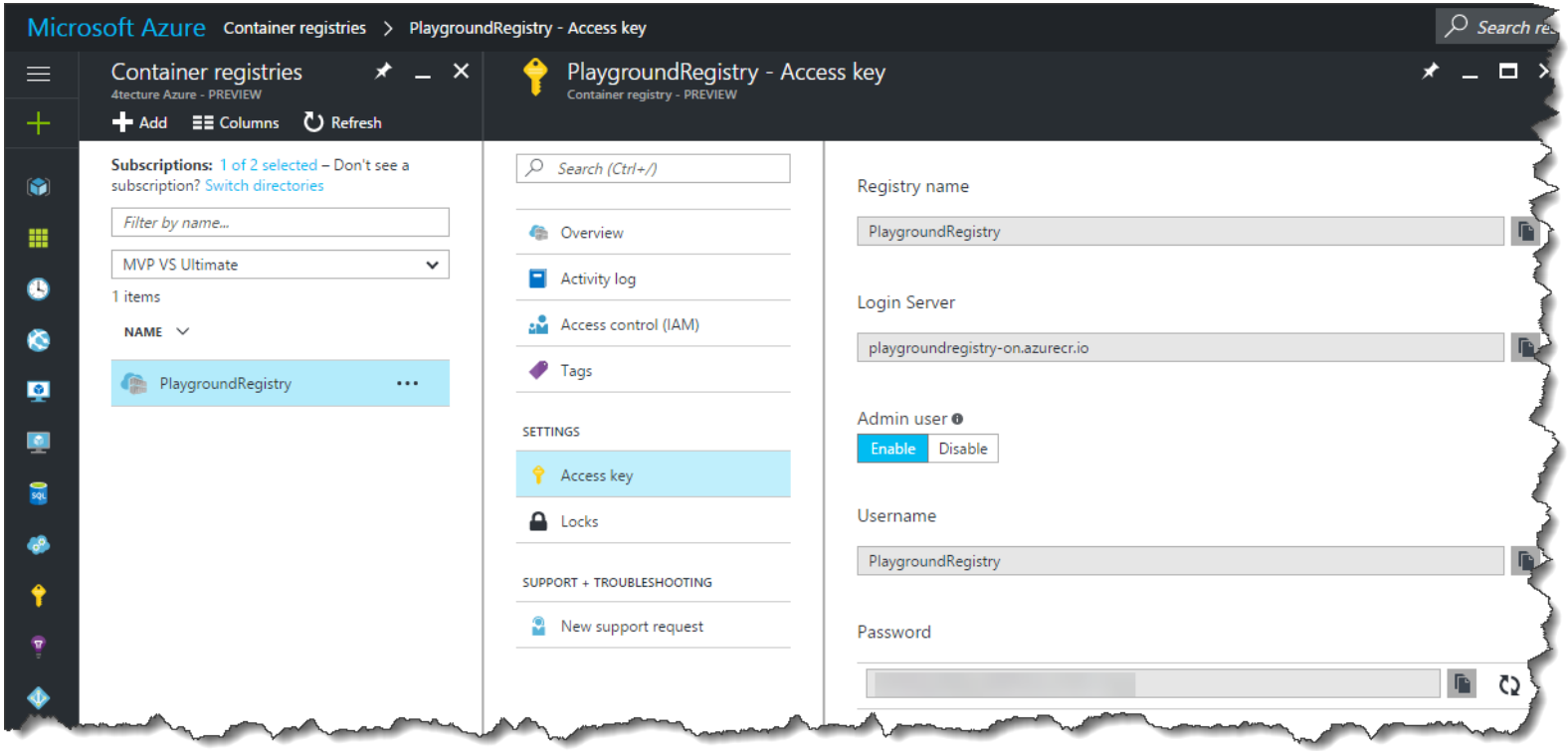


Demo

- Compile in Docker Container
- Setup Docker Build Agent


Docker Hosting
4tecture
empower your software solutions

Docker Registry im Azure Portal



Docker Registry API

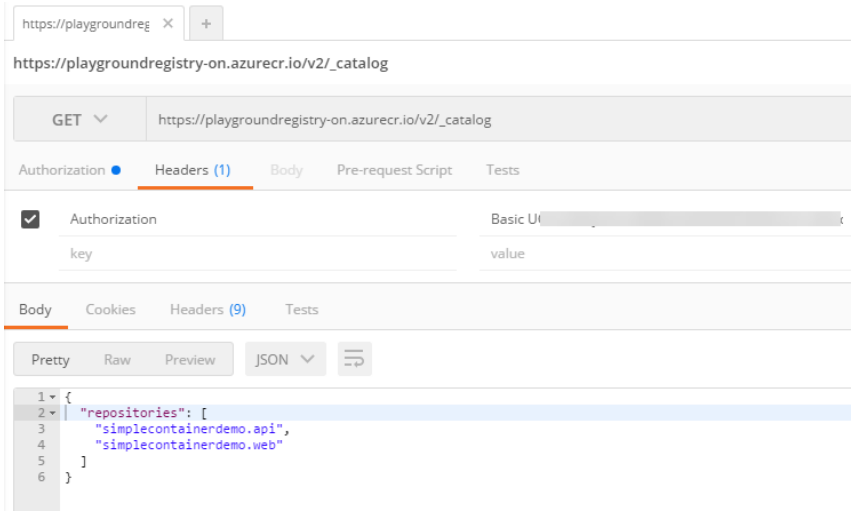
Method	Path	Entity	Description
GET	`/v2/`	Base	Check that the endpoint implements Docker Registry API V2.
GET	`/v2/<name>/tags/list`	Tags	Fetch the tags under the repository identified by `name`.
GET	`/v2/<name>/manifests/<reference>`	Manifest	Fetch the manifest identified by `name` and `reference` where `reference` can be a tag or digest. A `HEAD` request can also be issued to this endpoint to obtain resource information without receiving all data.
PUT	`/v2/<name>/manifests/<reference>`	Manifest	Put the manifest identified by `name` and `reference` where `reference` can be a tag or digest.
DELETE	`/v2/<name>/manifests/<reference>`	Manifest	Delete the manifest identified by `name` and `reference`. Note that a manifest can *only* be deleted by `digest`.
GET	`/v2/<name>/blobs/<digest>`	Blob	Retrieve the blob from the registry identified by `digest`. A `HEAD` request can also be issued to this endpoint to obtain resource information without receiving all data.
DELETE	`/v2/<name>/blobs/<digest>`	Blob	Delete the blob identified by `name` and `digest`
POST	`/v2/<name>/blobs/uploads/`	Initiate Blob Upload	Initiate a resumable blob upload. If successful, an upload location will be provided to complete the upload. Optionally, if the `digest` parameter is present, the request body will be used to complete the upload in a single request.
GET	`/v2/<name>/blobs/uploads/<uuid>`	Blob Upload	Retrieve status of upload identified by `uuid`. The primary purpose of this endpoint is to resolve the current status of a resumable upload.
PATCH	`/v2/<name>/blobs/uploads/<uuid>`	Blob Upload	Upload a chunk of data for the specified upload.
PUT	`/v2/<name>/blobs/uploads/<uuid>`	Blob Upload	Complete the upload specified by `uuid`, optionally appending the body as the final chunk.
DELETE	`/v2/<name>/blobs/uploads/<uuid>`	Blob Upload	Cancel outstanding upload processes, releasing associated resources. If this is not called, the unfinished uploads will eventually timeout.
GET	`/v2/_catalog`	Catalog	Retrieve a sorted, json list of repositories available in the registry.

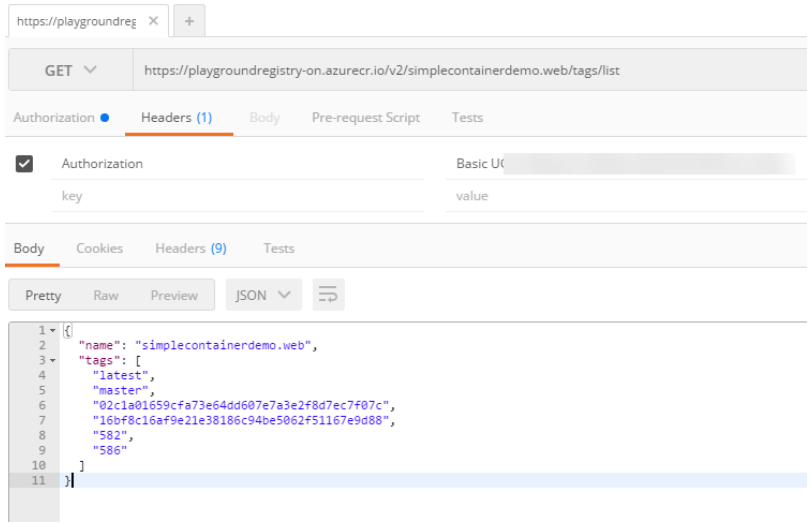
Image Source: https://docs.docker.com/registry/spec/api/

Docker Registry API - Samples

List Repositories: /v2/_catalog

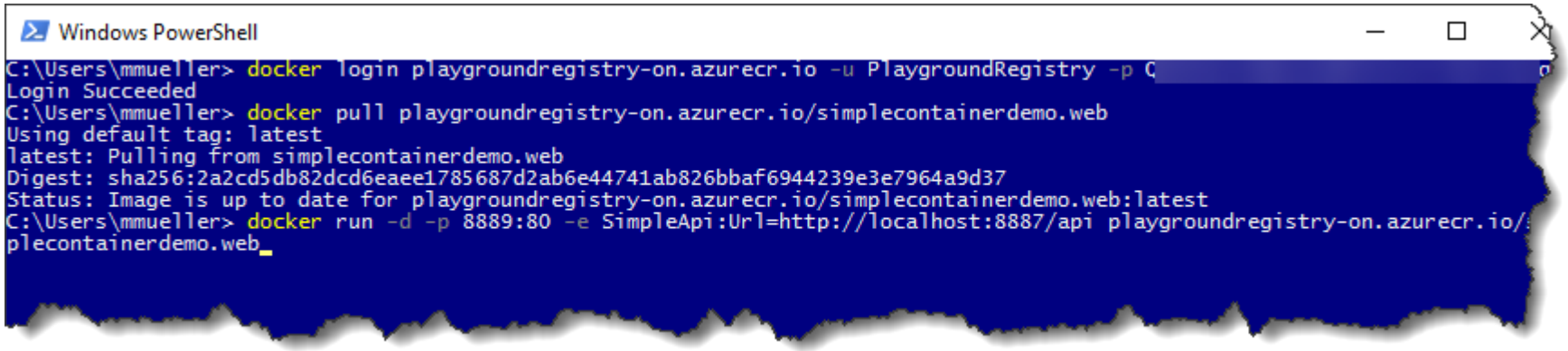
List Tags: /v2/<repository>/tags/list





Azure Container Registry und Docker

- Login to your private registry — docker login <registry> -u username -p password
- Pull the desired image — docker pull <registry>/<repository>:<tag>
- Run the container — docker run…



Containerized Workflow

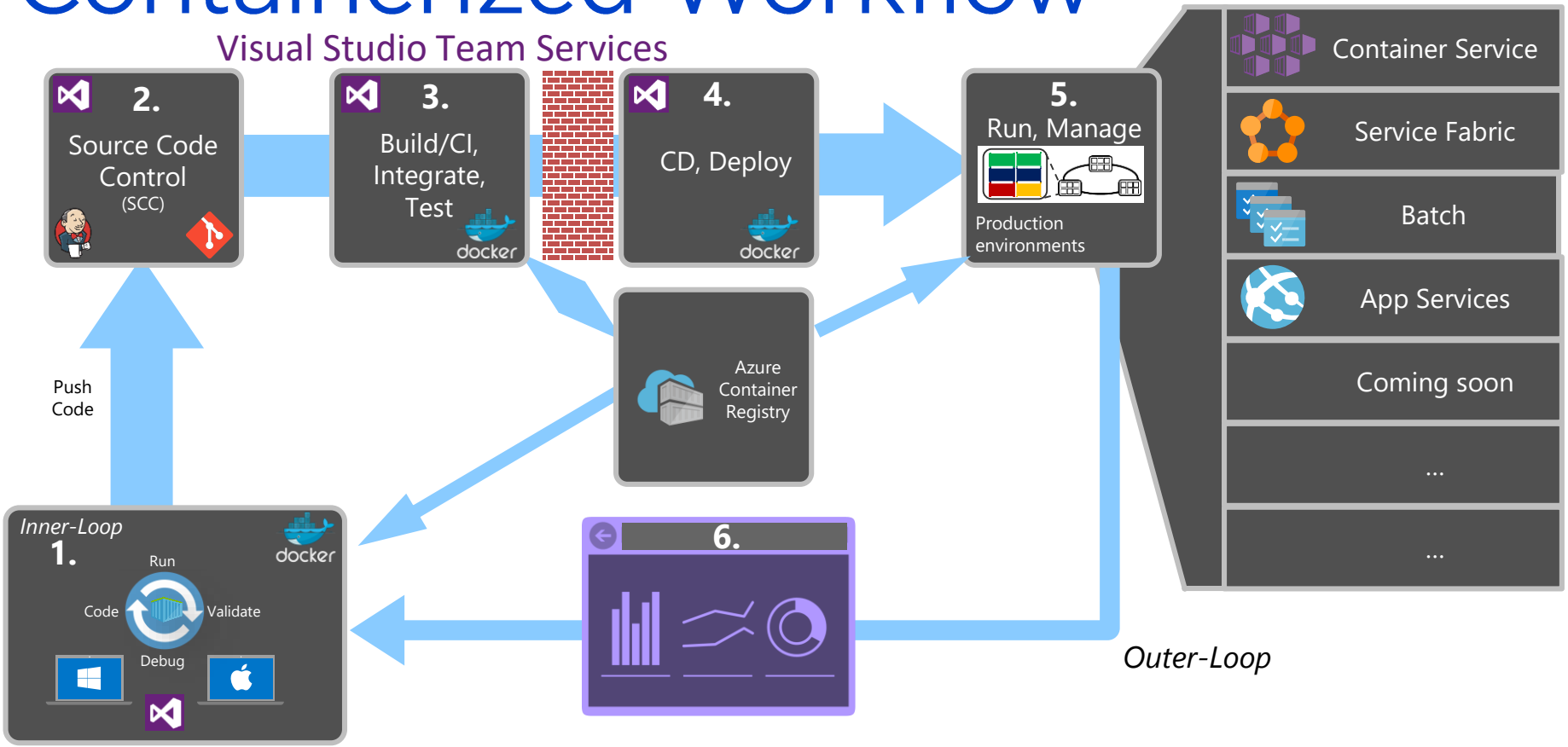


Image Source: https://sec.ch9.ms/sessions/69613110-c334-4f25-bb36-08e5a93456b5/206BuildDebugDeployASPNETCorewithDocker.pptx

Azure Container Service

ACS = ARM Templates

3 Varianten

- DC/OS with Marathon
- Docker Swarm
- Kubernetes

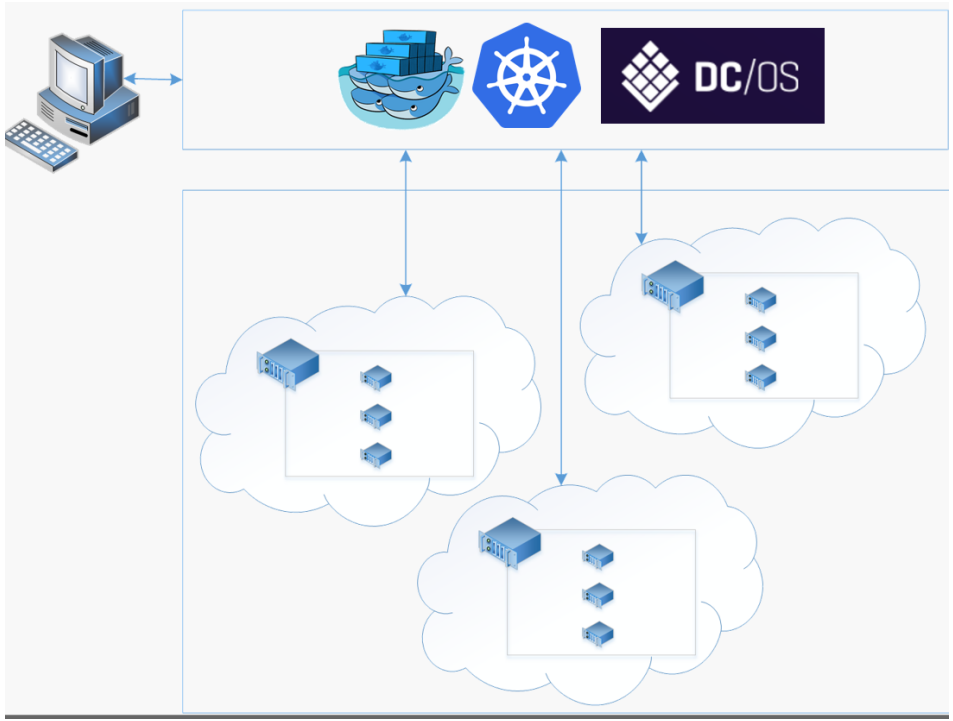
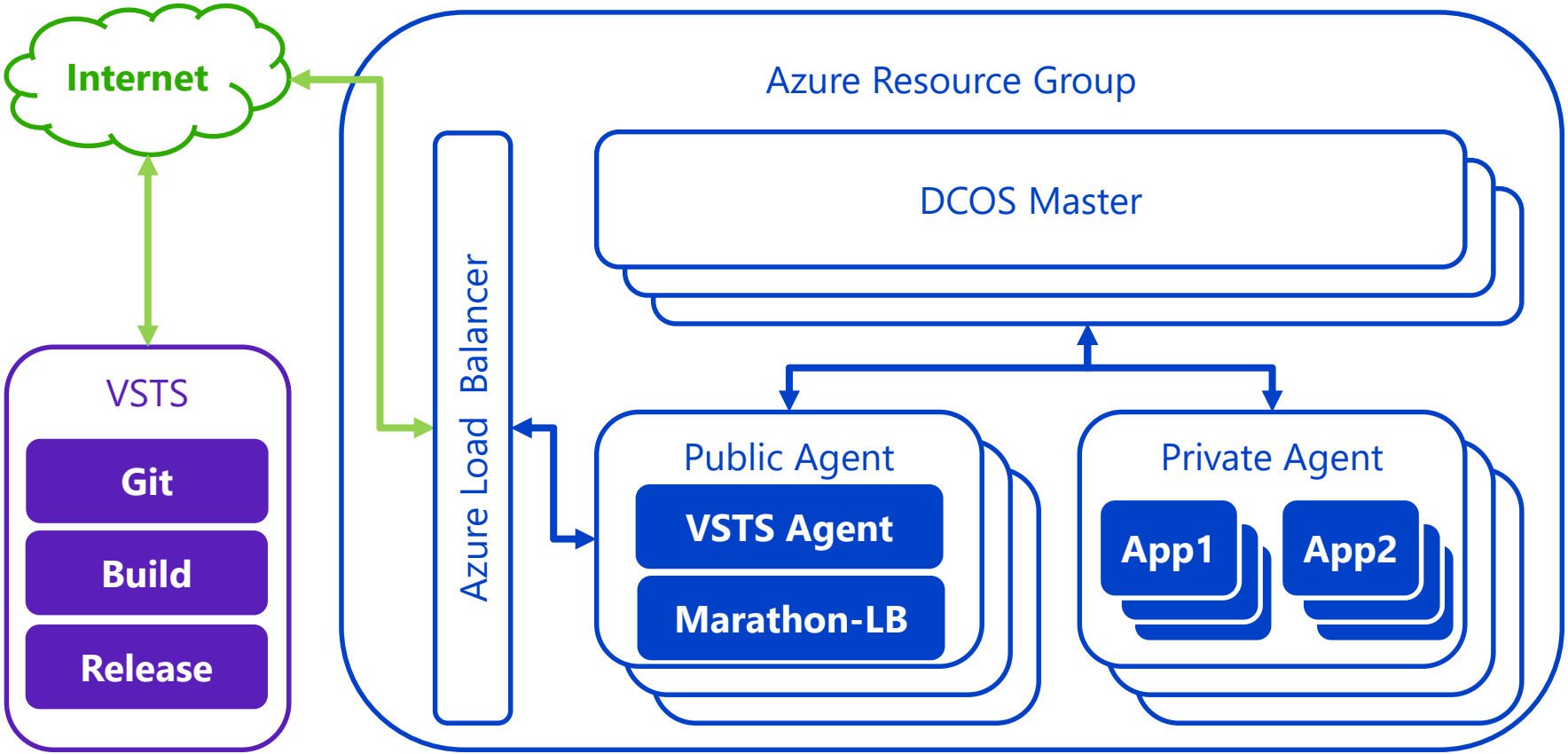
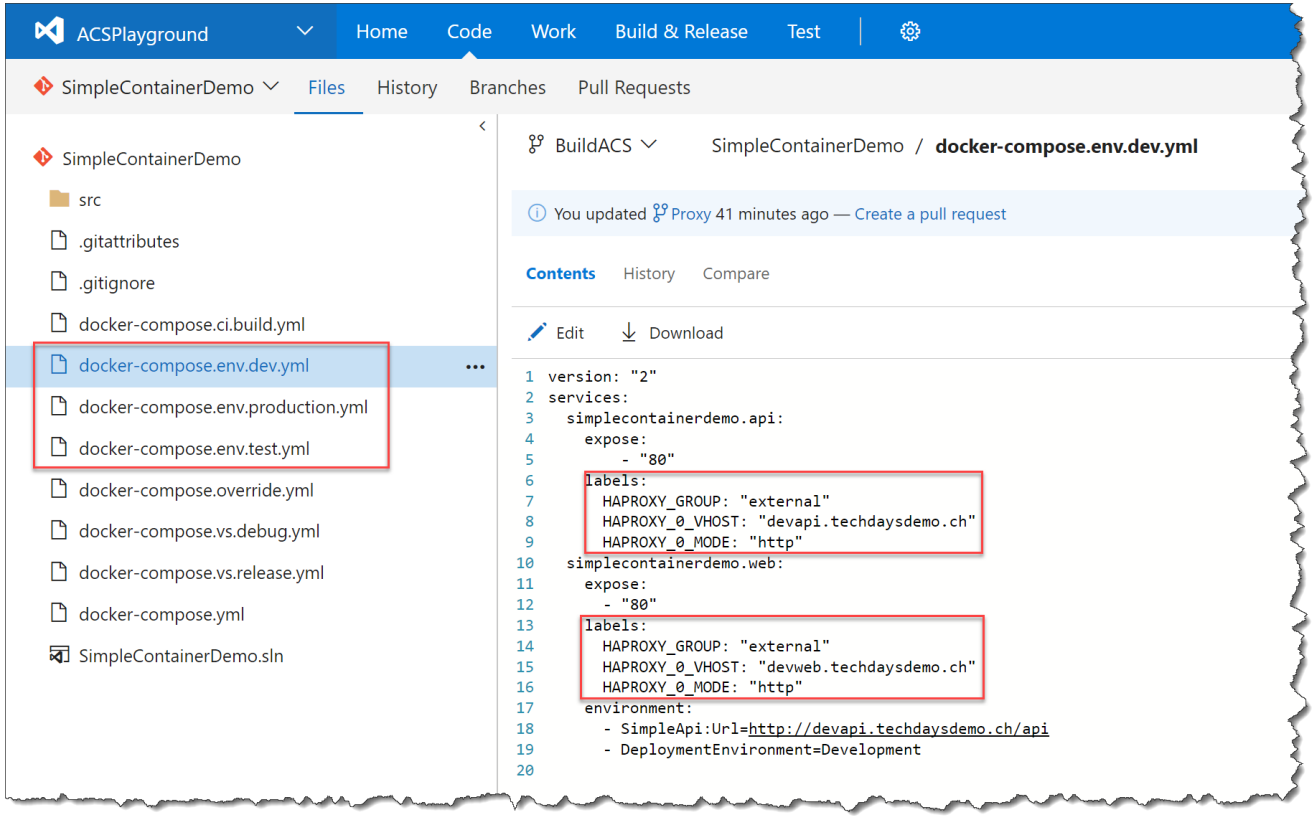


Image Source: https://docs.microsoft.com/en-us/azure/container-service/container-service-intro

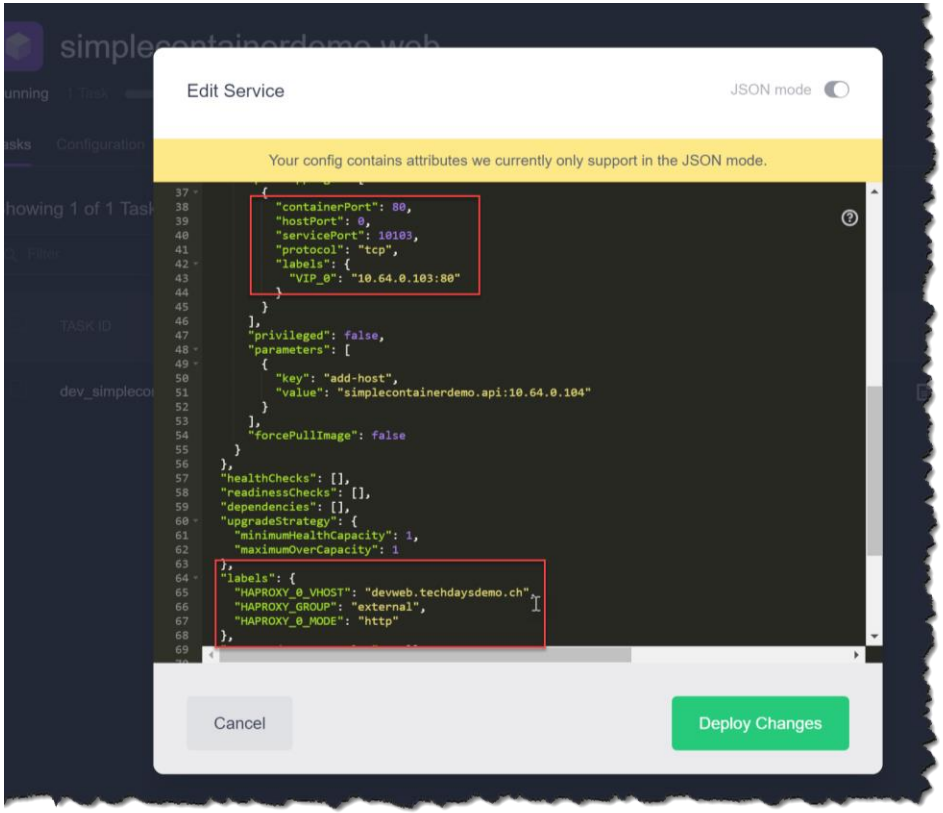
Simplified ACS DCOS Setup



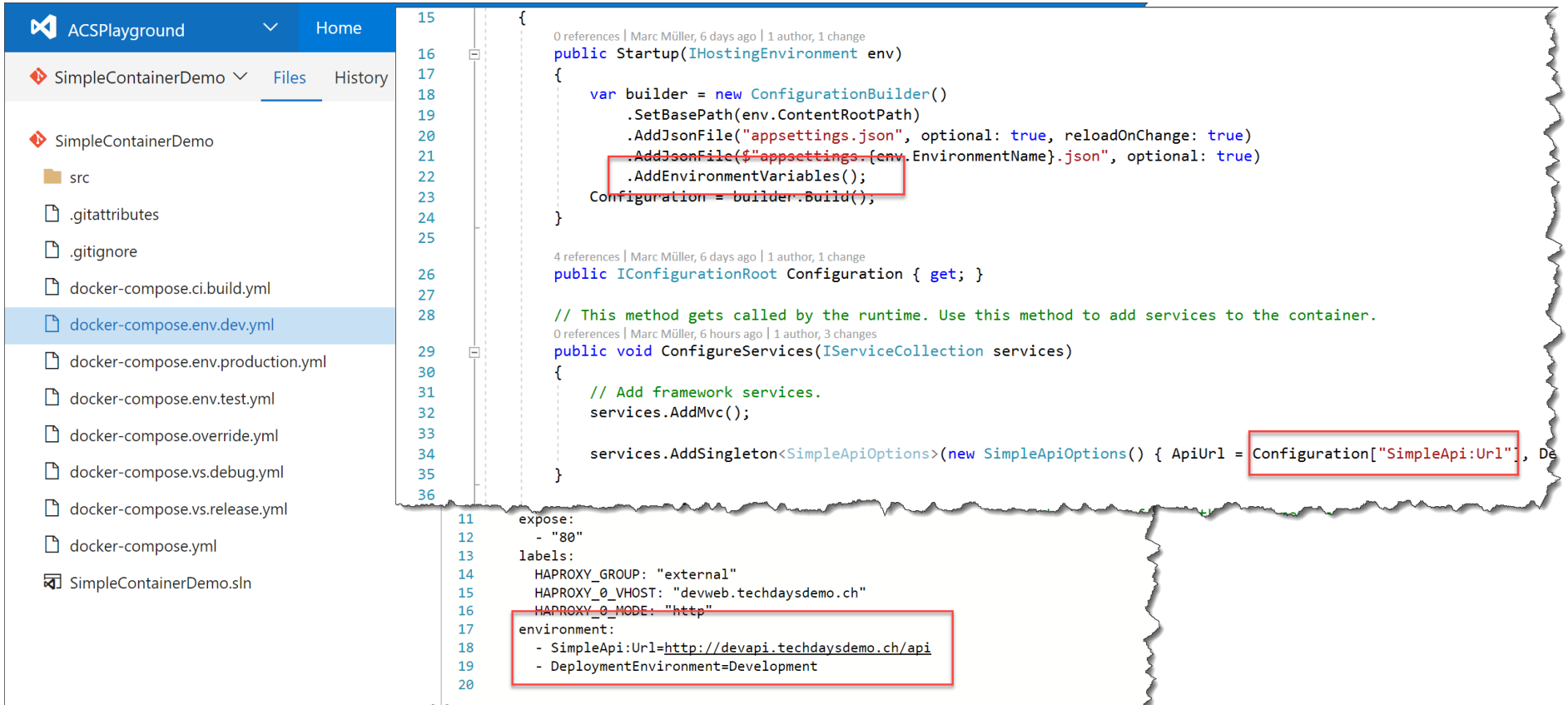
Load Balancer Settings for Services



Load Balancer Settings for Services



Use Environment Variables



Zusammenfassung

Zusammenfassung

- Micro-Service Architektur
  Reduktion der Komplexität, einfacher und schnellere Deployments
- Automatisierung der Build- und Release-Prozesse
- Isolation der Services als Container
- Isolation der Build Umgebung (Agent)
- Load Balancing mittels Service Discovery

Besten Dank für Ihre Aufmerksamkeit!

Für allfällige Fragen stehen wir Ihnen jederzeit gerne zur Verfügung:

4tecture GmbH
Aathalstrasse 84
CH-8610 Uster

+41 44 508 37 00
info@4tecture.ch
www.4tecture.ch

Marc Müller
Principal Consultant

marc.mueller@4tecture.ch
@muellermarc

DIGICOMP

Microsoft

NOSER ENGINEERING
WE KNOW HOW

progile
productive not busy

trivadis
makes IT easier.

OFFERING SOLUTIONS SOFTWARE