axians

Infoma

An introduction to Windows Containers and BC on Docker

VINCI ENERGIES

Tobias Fenster

CTO at Axians Infoma
Microsoft MVP for Business Applications

@tobiasfenster / techblog.axians-infoma.com

Agenda

- Introduction to Windows Containers
- Business Central on Docker

Part 1 – **Windows Containers**

Quick introduction to docker

Containers and images

- What is Docker?
  Leading cross platform software container environment
- What is a Docker container and a Docker image?
  - An image is a template with the minimum amount of os, libraries and application binaries needed
  - A container is an instance of an image with an immutable base and it's changes on top
  - A container is NOT a VM, you especially don't have a GUI and nothing you can connect to with RDP!
  - Images have tags for versioning

Quick introduction to docker

Host, registry and repository

- What is a Docker host?
  The (physical or virtual) machine where the containers are running
- What is a Docker registry?
  A place where you and others can upload (push) and download (pull) images. Docker offers Docker Hub as free registry
- What is a Docker repository?
  A part of a registry (like a “subfolder”) owned by a person or company

Quick introduction to docker

Flow to a running container

Identify the image you want

microsoft/mssql-server-linux

Side note: You need to specify
<registry>/<repository>/<image>:<tag>
but Docker has defaults, so
microsoft/mssql-server-linux
resolves to
docker hub/microsoft/mssql-server-linux:latest

Infoma

Quick introduction to docker

Flow to a running container

Identify the image you want

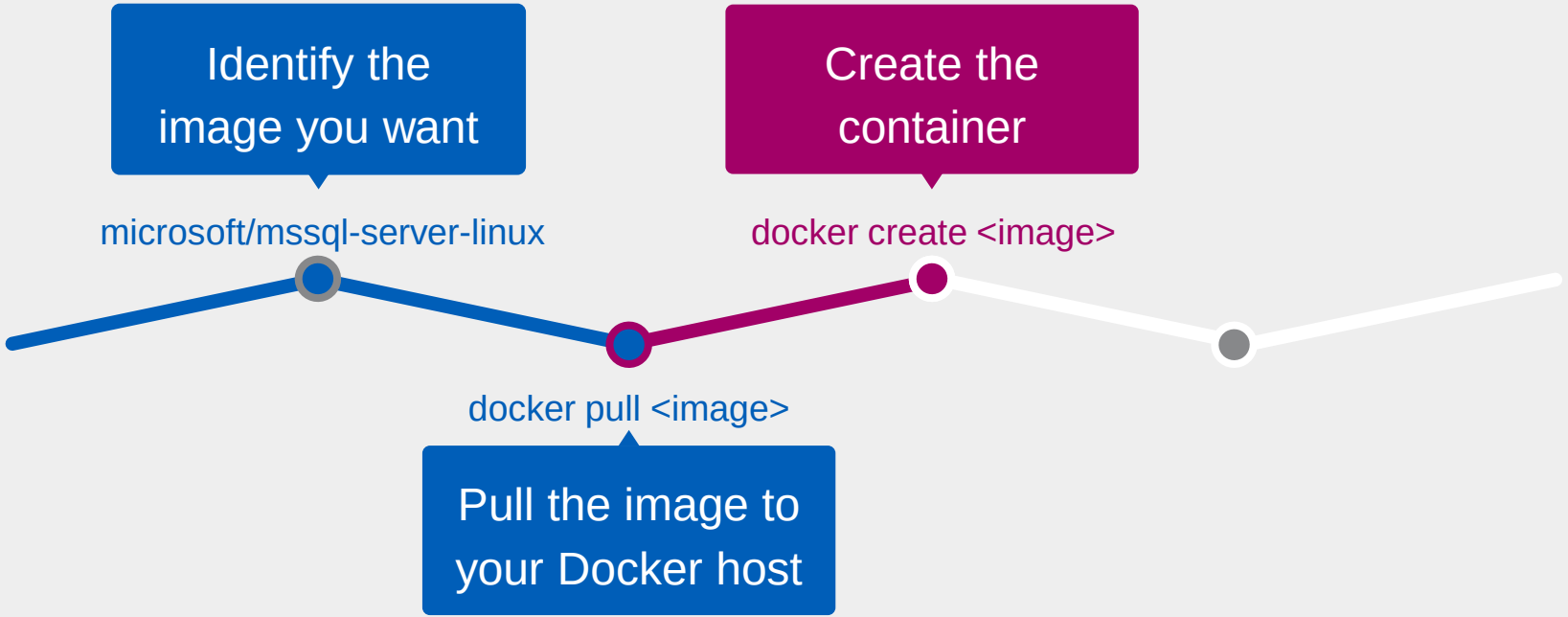
microsoft/mssql-server-linux

docker pull <image>

Pull the image to your Docker host

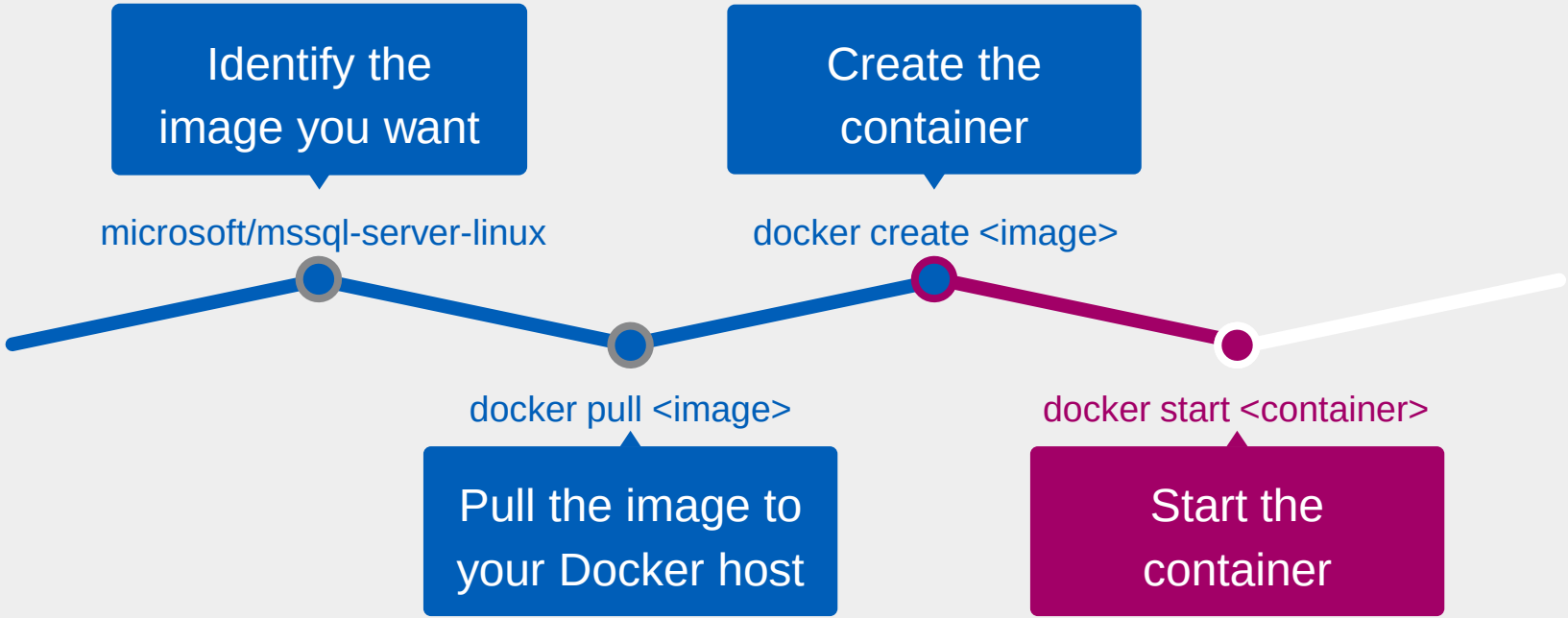
Quick introduction to docker

Flow to a running container



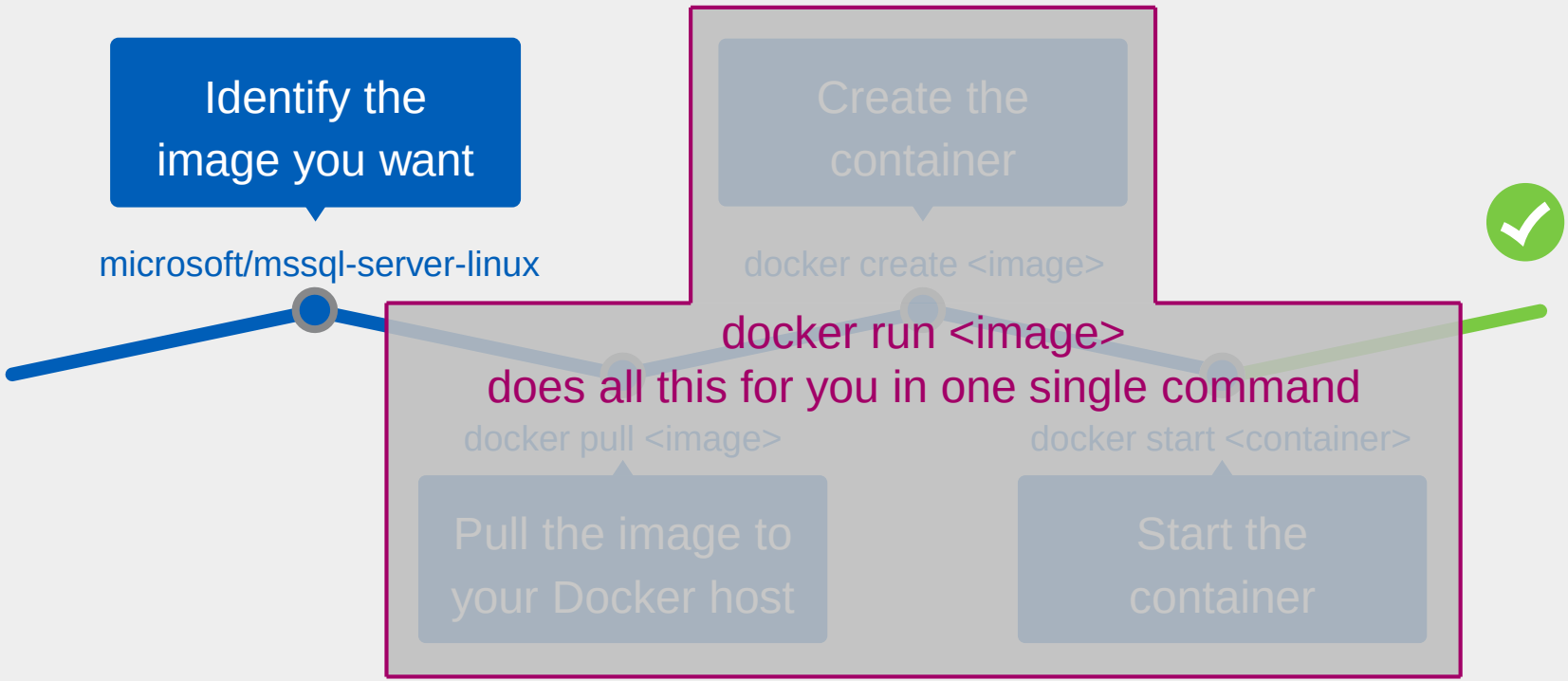
Quick introduction to docker

Flow to a running container



Quick introduction to docker

Flow to a running container



Quick introduction to docker

The basics of container handling

- Install through PowerShell (Windows Server) or installer (Win 10)
- Show running and all containers
- Run a container in interactive mode
- Get a PowerShell session inside a container
- Show resource consumption and logs
- Stop and remove containers, remove images

Quick introduction to docker

Main benefits

- Quickly install necessary software in different versions / editions
- Easy way to create deployments / configuration in a very stable and reliable way (no "works here", helps a lot to avoid gaps between dev and ops)
- Better resource usage than in VMs, especially because there is no guest OS as the host kernel is directly used
- Resource limits even if not natively supported by the application
- Big ecosystem of readily available images, primarily on Docker Hub

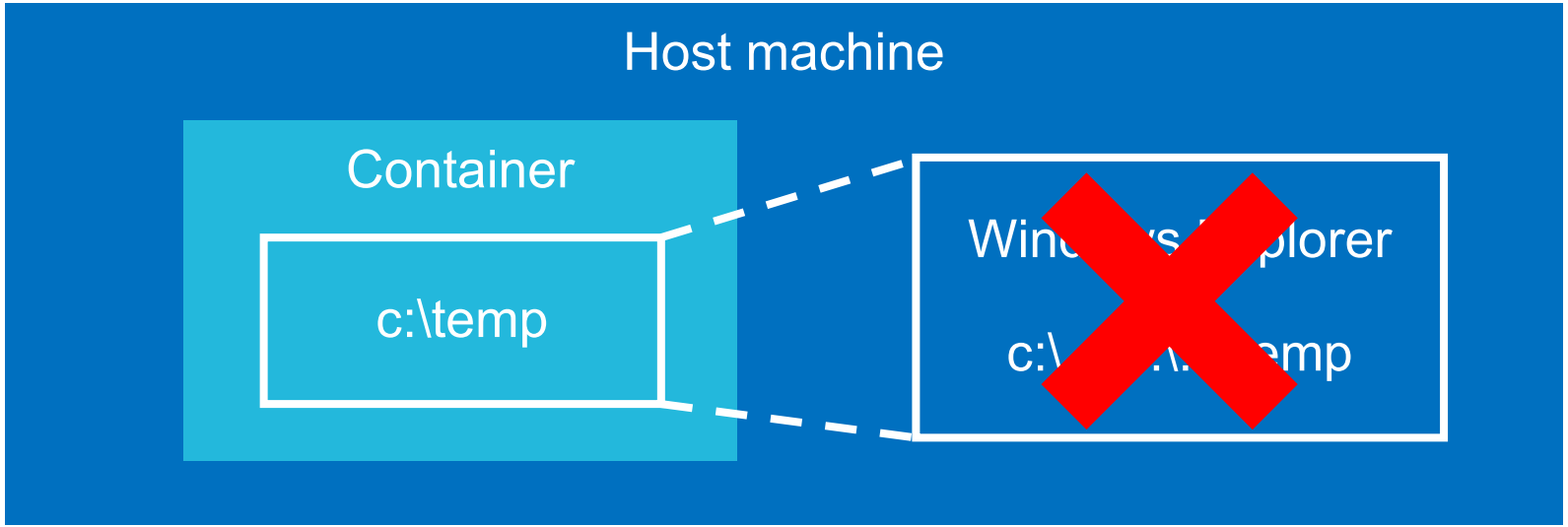
Quick introduction to docker

Editions and isolation

- Process isolation has minimal overhead, can only run on „matching“ hosts (backward compatible from Win Server 2019)
- Hyper-V isolation with a „mini VM“ for non matching containers
- Docker community edition (CE): Latest release, runs on Win 10 with Hyper-V isolation (process with 18.09.1 / 1809)
- Docker enterprise edition (EE): More stable, a bit behind, runs on Windows Server, supports both isolation types
  - Windows Server comes with EE basics for free, paid EE contains a lot of tooling around Docker for production usage

File handling

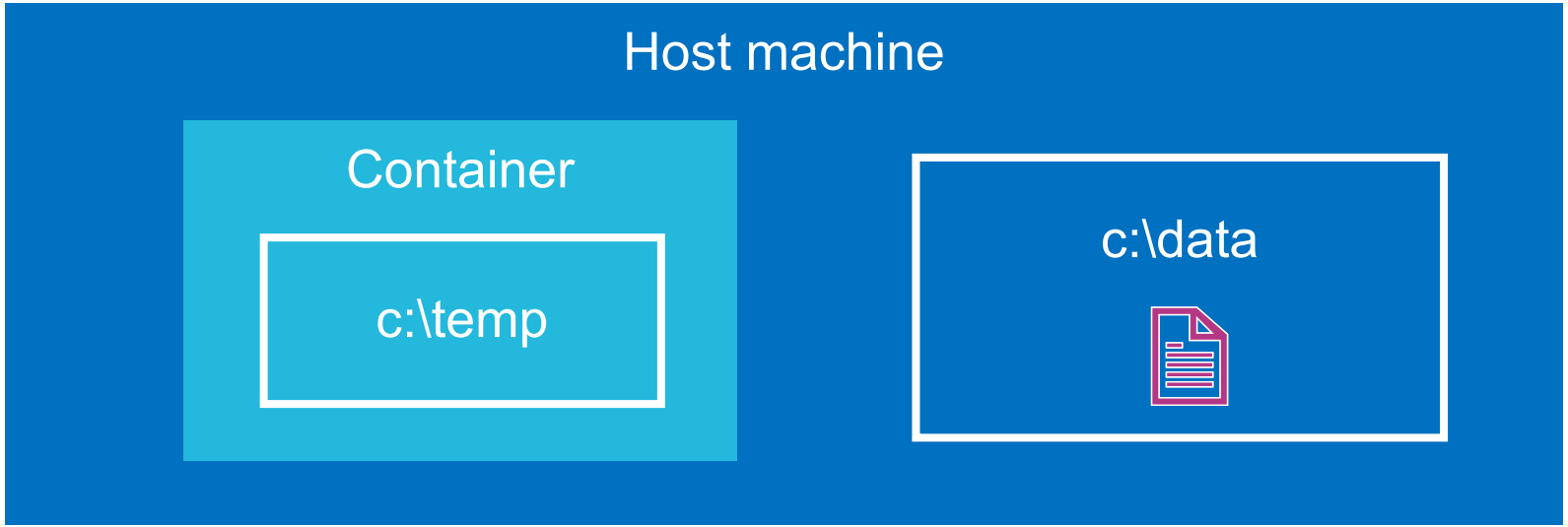
Working with Docker on Windows



Standard fs setup: nothing configured

File handling

Working with Docker on Windows

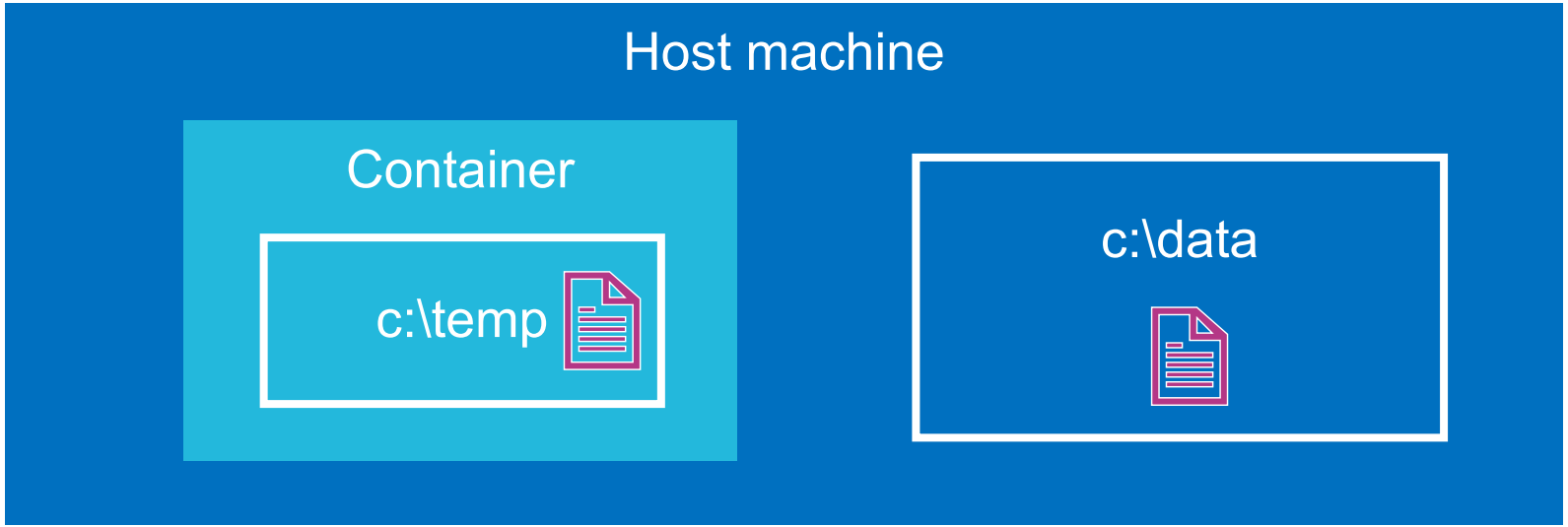


Standard fs setup: nothing configured. Use docker cp to copy files

Infoma

File handling

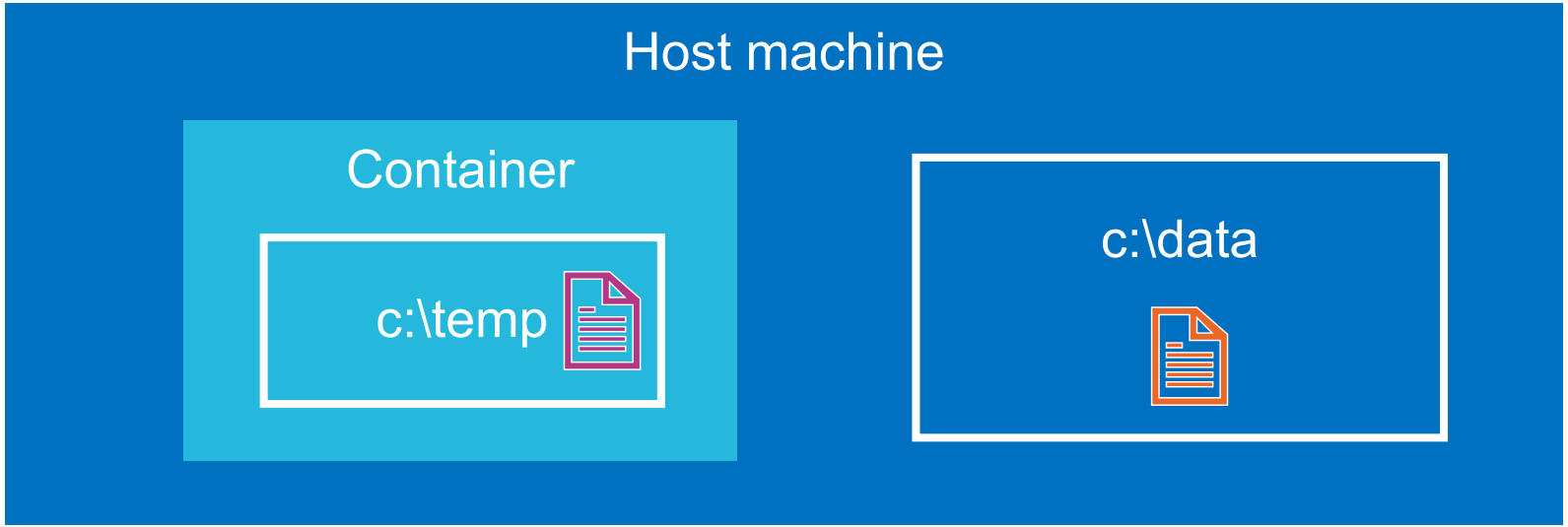
Working with Docker on Windows



Standard fs setup: nothing configured. Use docker cp to copy files

File handling

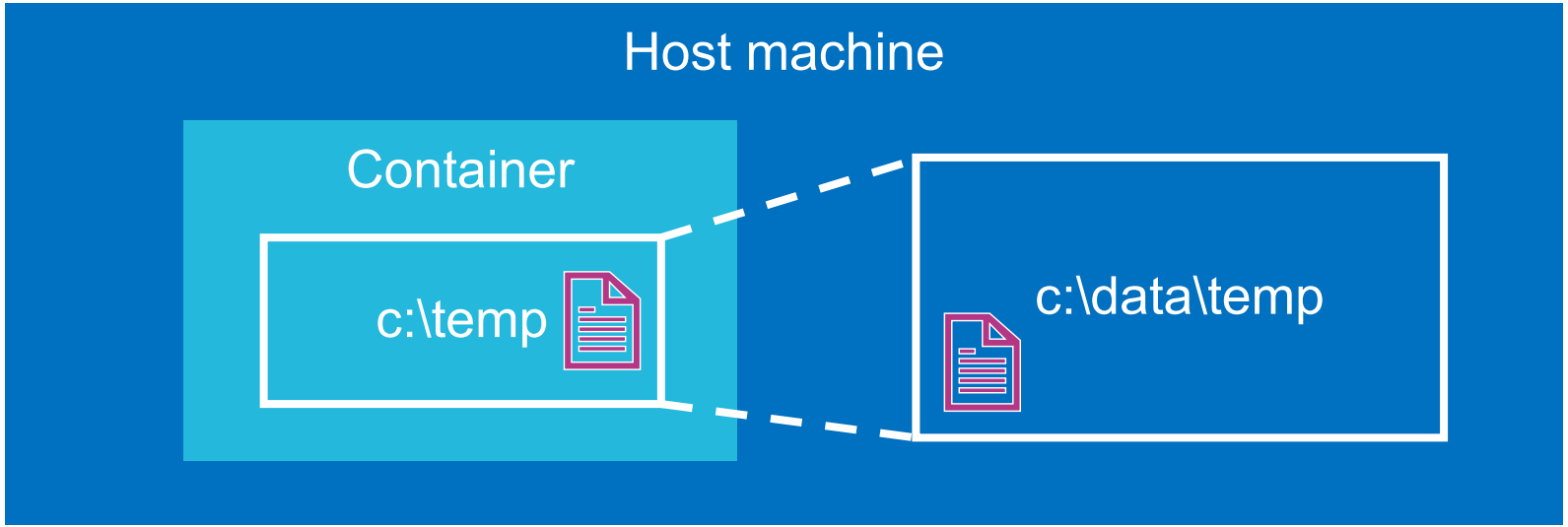
Working with Docker on Windows



Standard fs setup: nothing configured. Use docker cp to copy files

File handling

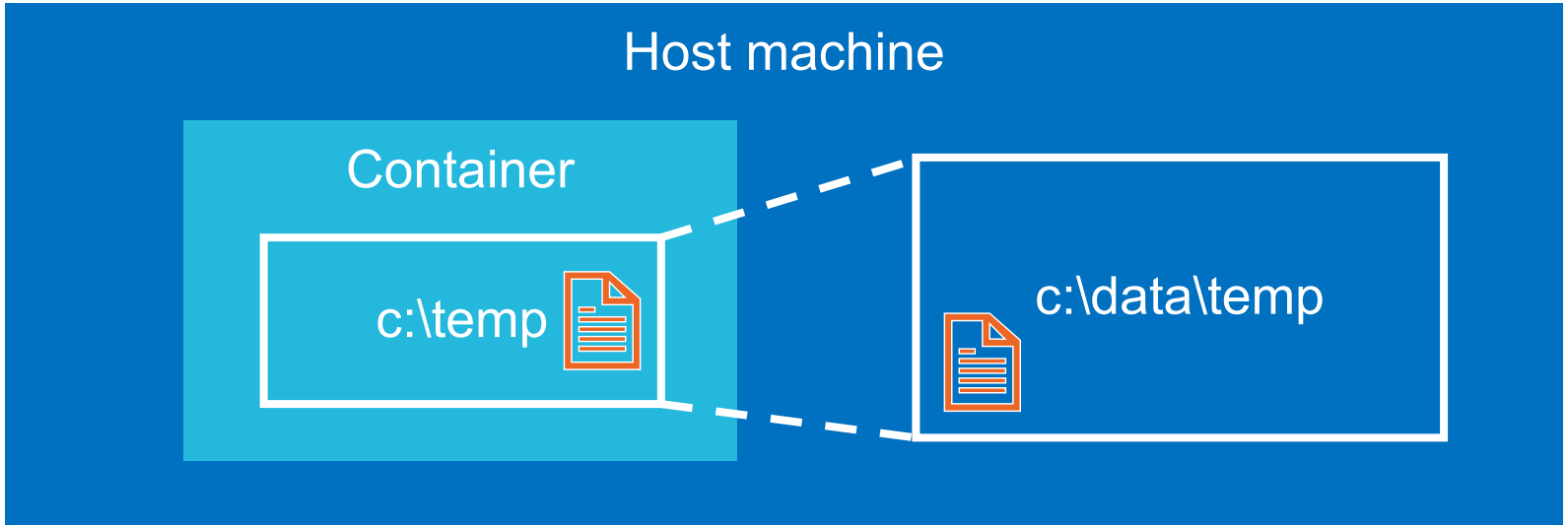
Working with Docker on Windows



fs setup with a volume, e.g. -v c:\data\temp:c:\temp

File handling

Working with Docker on Windows



fs setup with a volume, e.g. -v c:\data\temp:c:\temp

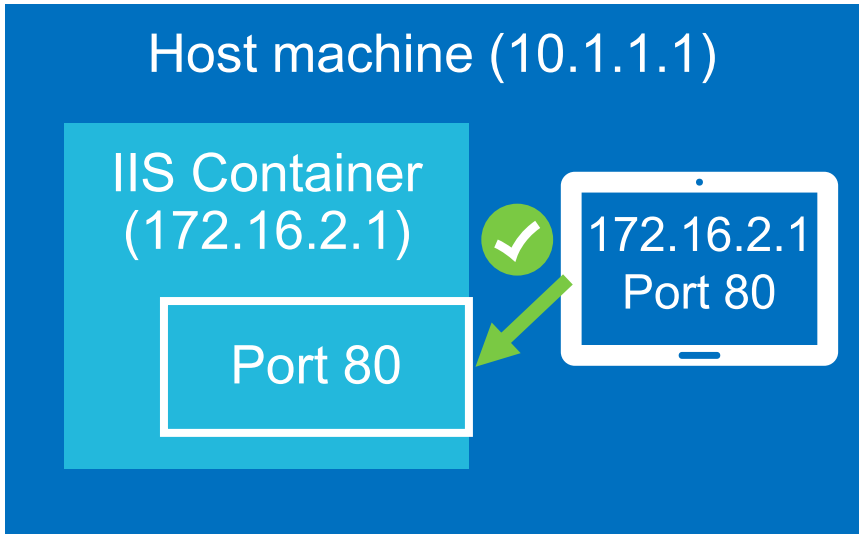
File handling

Working with Docker on Windows

- Two options for file sharing between host and container:
  - Command docker cp allows copying files, that means afterwards you have two identical but unrelated files → works anytime
  - Parameter -v for volumes allow sharing folders between host and container (only empty target folders until Server 2016) → can only be set up on startup

Networking

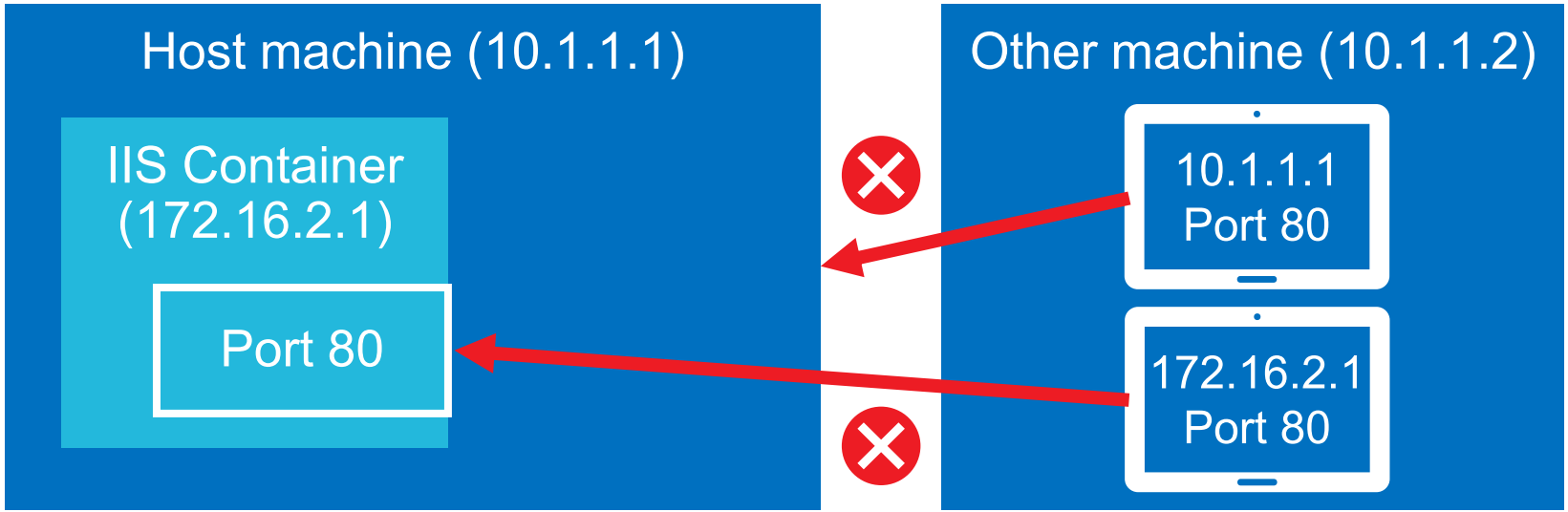
Working with Docker on Windows



Standard network setup: NAT

Networking

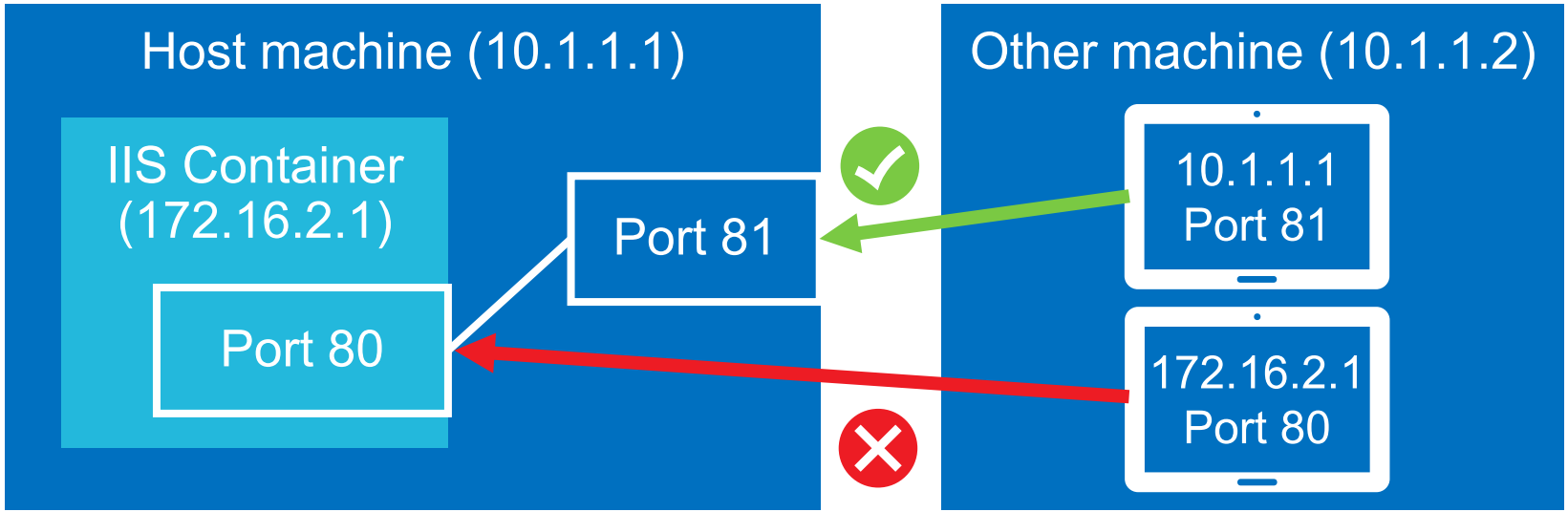
Working with Docker on Windows



Standard network setup: NAT

Networking

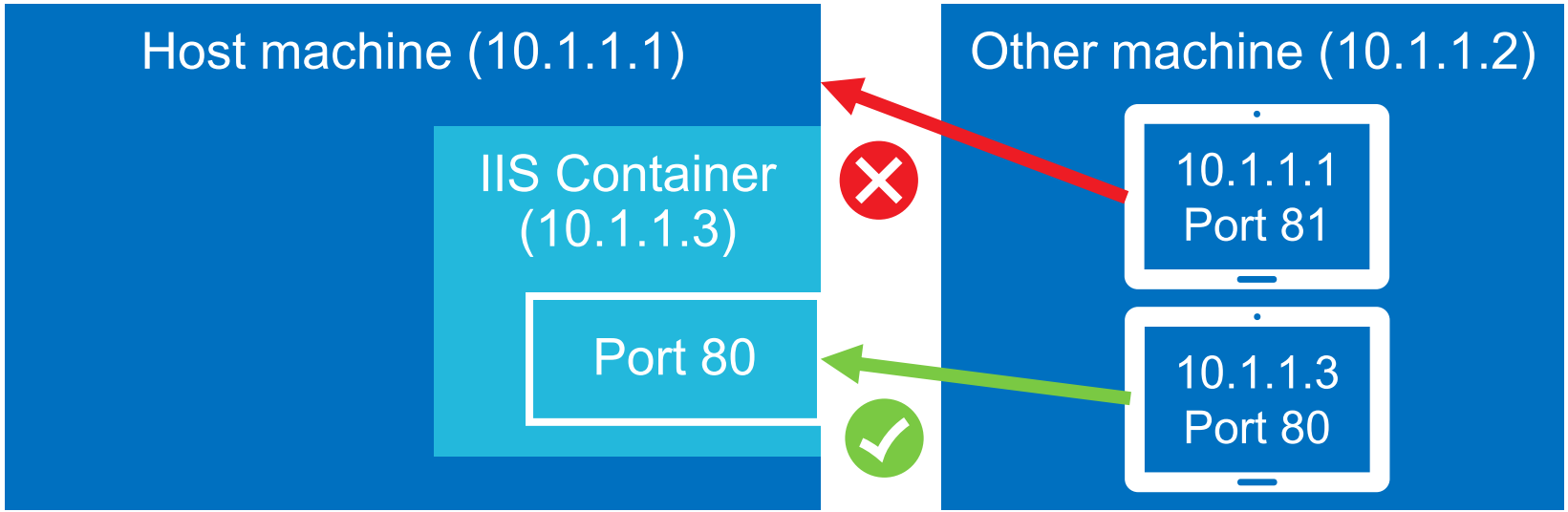
Working with Docker on Windows



Standard network setup with port mapping, e.g param -p 81:80

Networking

Working with Docker on Windows



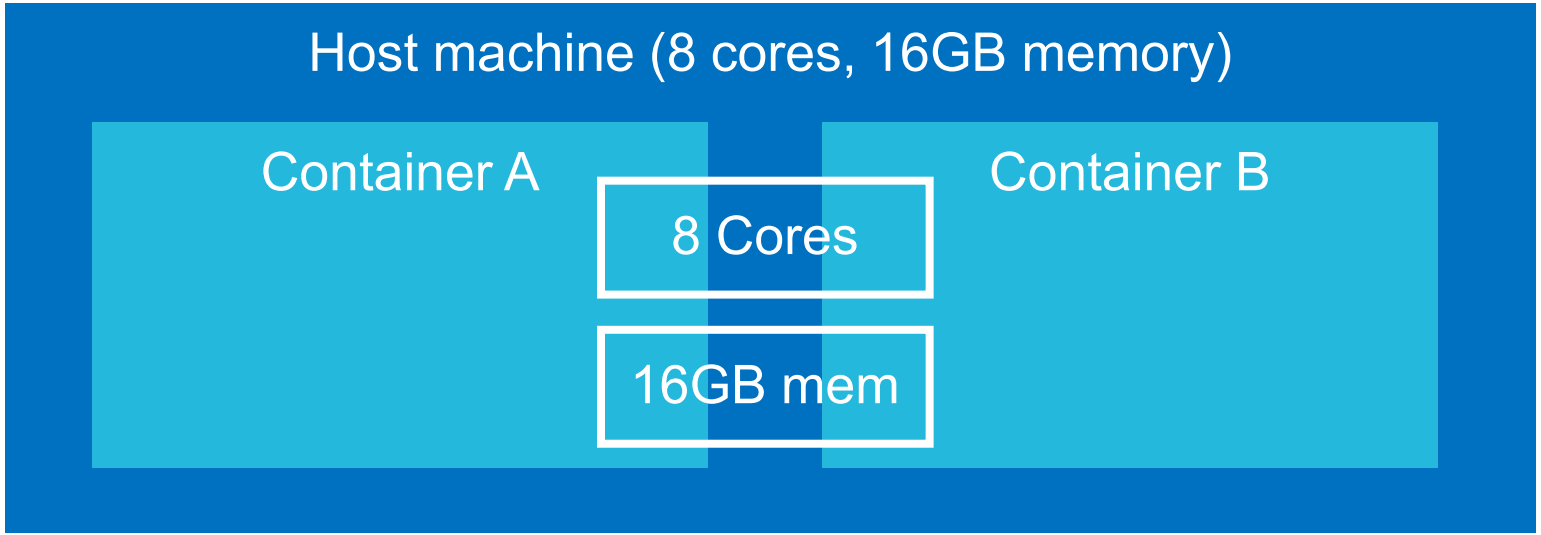
Transparent setup: host and container “share” a network adapter

Networking
Working with Docker on Windows

- Three options for network connections to the container:
  - Do nothing: Default NAT allows connections only from the host
  - Port mapping of 1-n ports on the container to 1-n possibly different ports on the hosts
  - Sharing the network through transparent config gets a dedicated IP (static or dynamic) for every container and makes it reachable on that network

Resource limits

Working with Docker on Windows

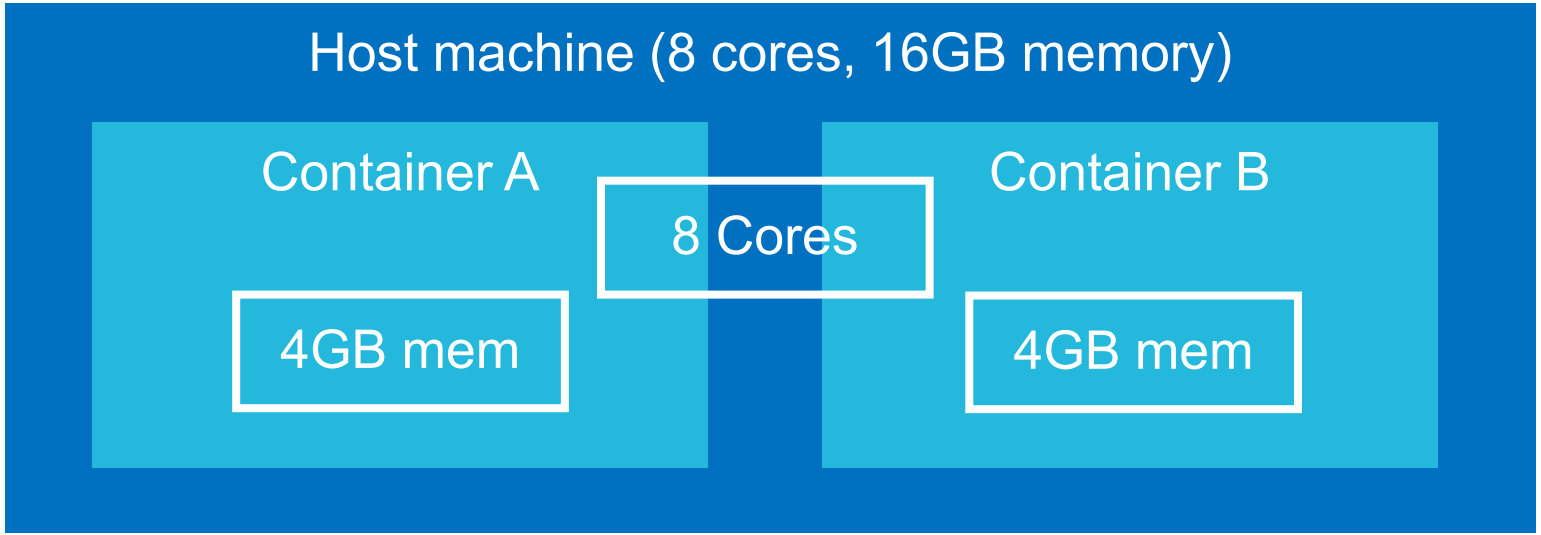


Standard resource setup: nothing configured

Infoma

Resource limits

Working with Docker on Windows



Specific resource setup: e.g. memory limit -m 4g

Resource limits

Working with Docker on Windows

- Various options to limit CPU, memory and IO
  - See docker run --help
- Can only be set up on startup

Links to know and people to follow

Working with Docker on Windows

- https://docs.microsoft.com/en-us/virtualization/windowscontainers/
- https://docs.docker.com/
- @Docker
- @EltonStoneman (dev advocate at Docker / Microsoft MVP)
- @stefscherer (Docker Captain / Microsoft MVP)
- @ManoMarks (director dev relations at Docker)

Part 2 – **BC on Docker**

Available images (only dev and test!)
Business Central on Docker

- On Premises images:
  - microsoft/dynamics-nav and mcr.microsoft.com/businesscentral/onprem
  - Tag:<ver>-<cu>-<country>-<winver>, e.g. 2017-cu22-de-ltsc2019
- SaaS images:
  - mcr.microsoft.com/businesscentral/sandbox → current
  - bcinsider.azurecr.io/bcsandbox → next minor
  - bcinsider.azurecr.io/bcsandbox-master → next major
  - Tagged with :<build>-<country>-<winver>, e.g. 13.1.25940.26323-dk-ltsc2019

Base structure for all images

Business Central on Docker

- Public repository https://github.com/microsoft/nav-docker
- Base of all: „generic“ image
  - FROM windowsservercore with .NET runtime 4.7.2 (from 1809 only windowsservercore as it then includes .NET 4.7.2)
  - Install SQL Server and IIS dependencies
  - Copy files from Run folder into the image
  - Download Report Builder and some utils
- Same for all types (dynamics-nav, bcsandbox, bconprem): „specific“ images W1 (called „base“ in bcsandbox) and local versions behave a bit different

Base structure for all images

Business Central on Docker

- W1 / base built FROM generic:
  - Download NAV DVD and .vsix (AL extension for VS Code)
  - Move the right version specific files (folders 70 to 130) in place
  - Call navinstall.ps1 which starts SQL and IIS and „installs“ NAV / BC with dependencies, restores country independent CRONUS database and generates a Service Tier
- Country specific built FROM W1 / base:
  - Uses importCountry.ps1 to restore country Cronus<lang> databases like CronusDK or CronusDE, run local installers and adjust Service Tier conf
  - Also generates AL symbols from NAV 2018 onward

Base structure for all images

Business Central on Docker

- bcsandbox-master works the same afaik
- start.ps1 is called on docker run and calls all other scripts, depending on params and whether it is the first start of the container and whether the DNS name has changed

(Almost) everything is available as parameter

Business Central on Docker

- Custom NAV settings / Web settings
- Use Windows authentication and enable ClickOnce
- Connect to an external SQL Server
- Change the ports for the different services
- Good way to find all possible parameters: Check SetupVariables.ps1

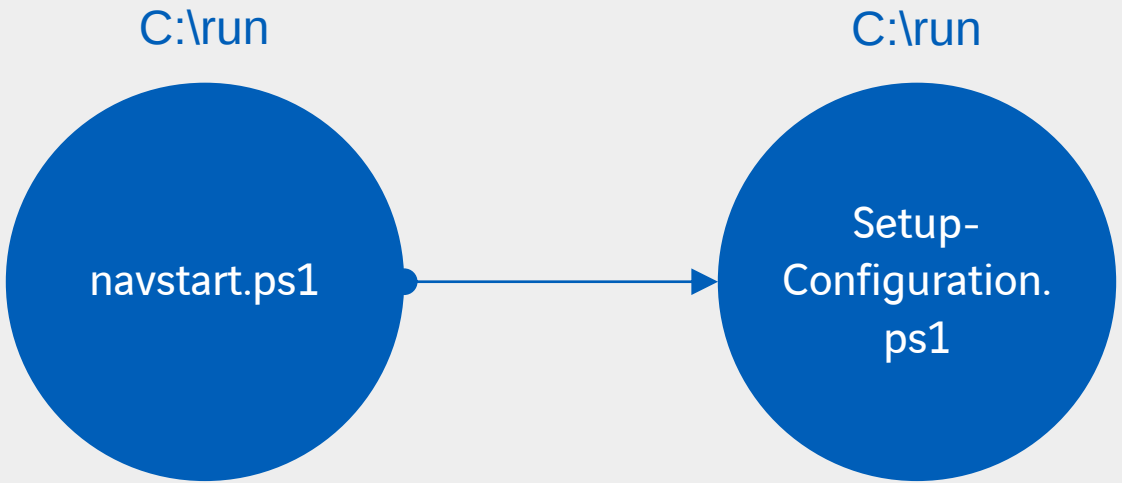
Overwrite scripts
Business Central on Docker

- Before calling standard scripts in c:\run, c:\run\my is checked for a script with the exact same name
- If the script exists in c:\run\my, it is called instead of the standard
- Make sure to call the standard script as well if necessary, before / during / after your lines:
```
. (Join-Path $runPath $MyInvocation.MyCommand.Name)
```
- Use if there is no param for your requirement

Infoma

Overwrite scripts

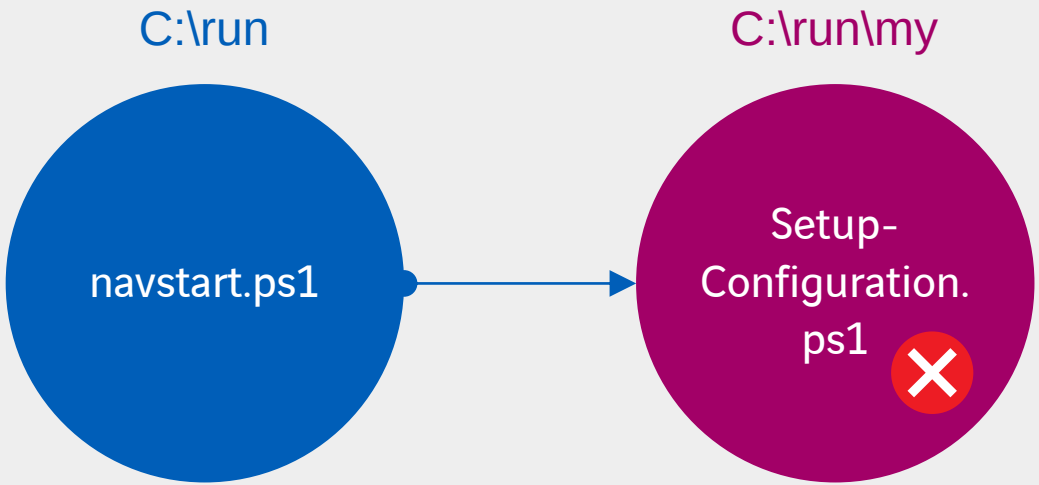
Business Central on Docker



Infoma

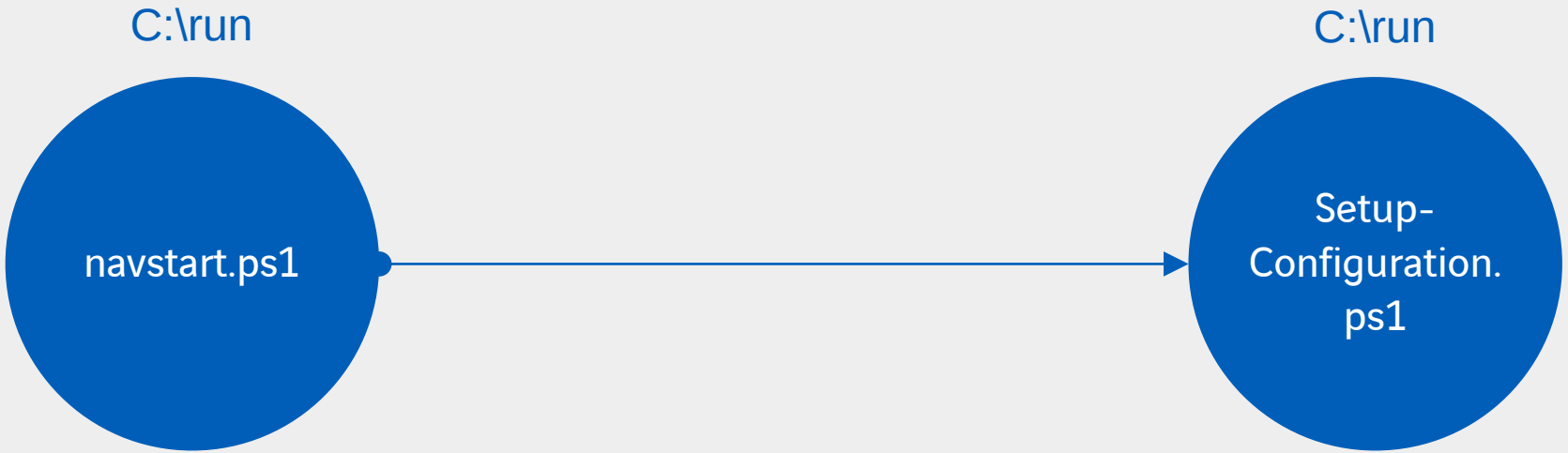
Overwrite scripts

Business Central on Docker



Overwrite scripts

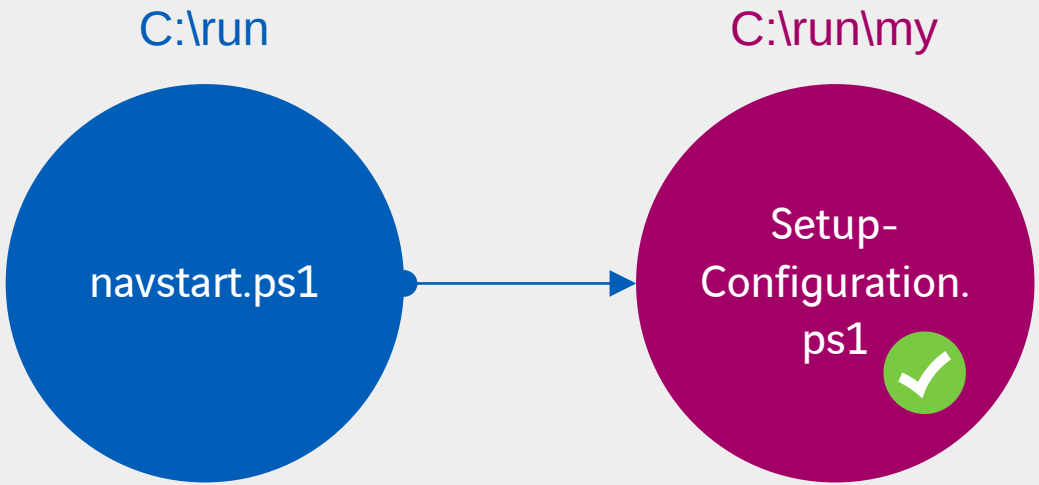
Business Central on Docker



Infoma

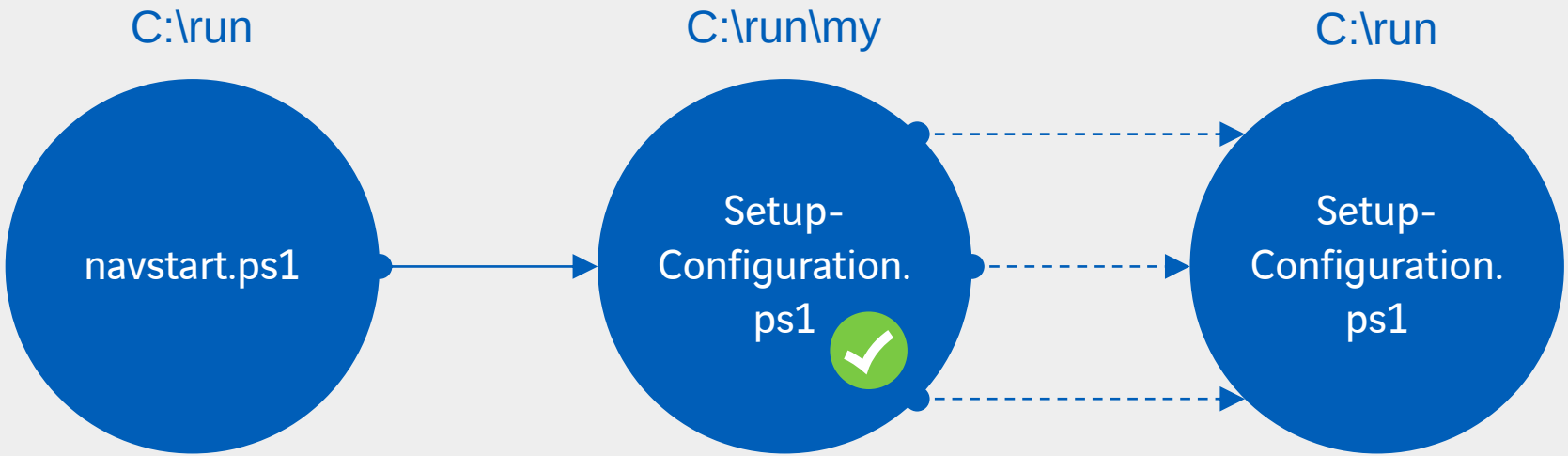
Overwrite scripts

Business Central on Docker



Overwrite scripts

Business Central on Docker



Getting files into a BC container

Business Central on Docker

- Docker cp and volumes work as expected: Overwrite start.ps1 through a volume bound to c:\run\my
- Download script on startup
  - Param „folders“ download files and puts them in the container, e.g. -e folders= “c:\temp=https://xyz.de/font.ttf“ → No need to have it locally on the host
  - Can also be used for public scripts e.g. on GitHub to put in c:\run\my

navcontainerhelper

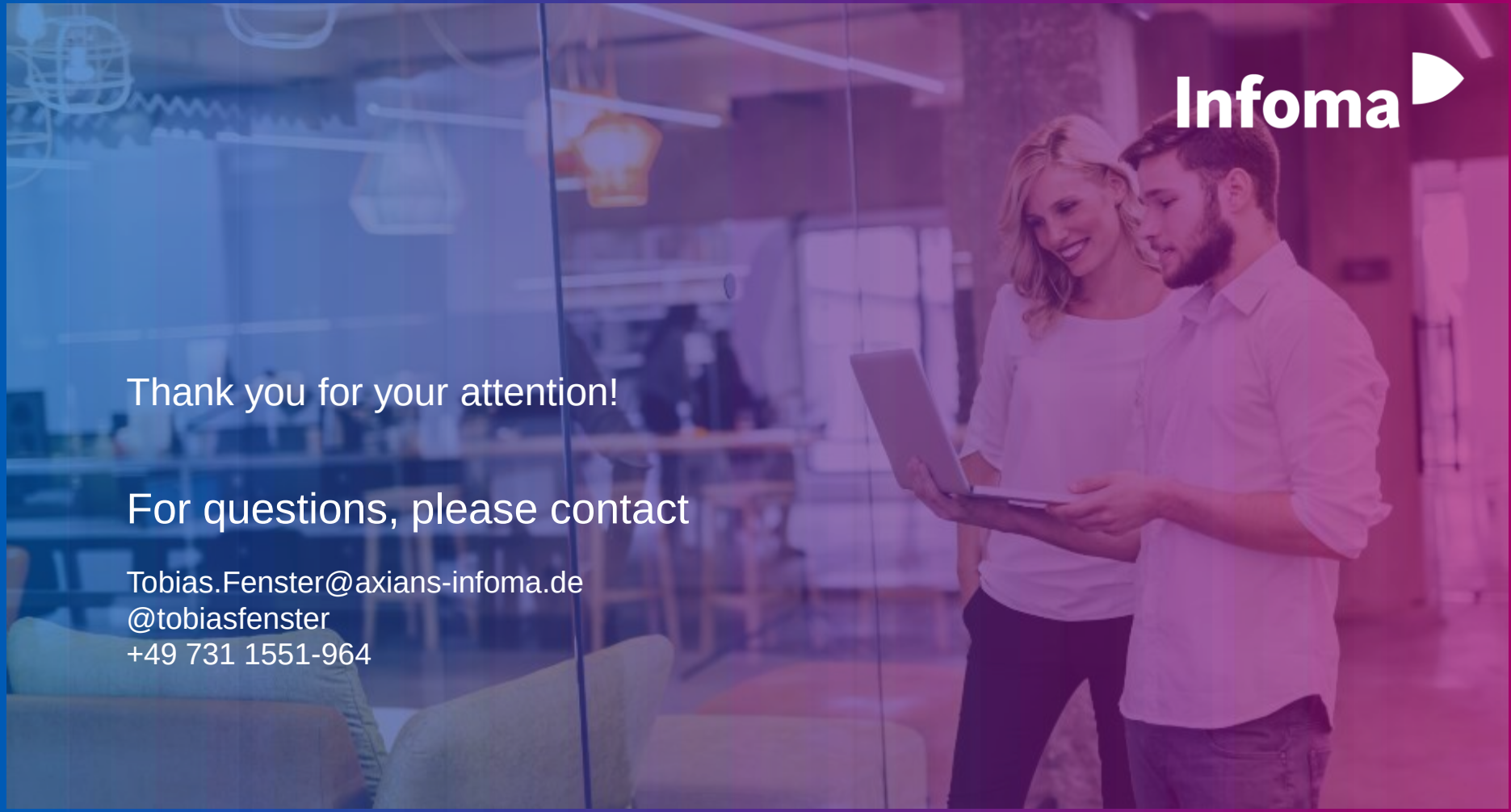
Business Central on Docker

- Collection of helper Cmdlets and Scripts to ease container usage mainly for NAV / BC development and devops
- Also base for Freddy's CI/CD scripts, aka.ms/getbc and others
- No "magic", but extensive set of common use cases like
  - New-NAVContainer, Replace-NavServerContainer
  - Convert-ModifiedObjectsToAl
  - Compile-AppInNavContainer, Compile-ObjectsInNavContainer
  - Install-NavContainerApp, Publish-NavContainerApp
  - Convert-AlcOutputToAzureDevOps, …

Links to know and people to follow

Business Central on Docker

- https://blogs.msdn.microsoft.com/freddyk/ and @freddydk
- https://github.com/Microsoft/nav-docker/
- https://github.com/Microsoft/navcontainerhelper



Thank you for your attention!

For questions, please contact

Tobias.Fenster@axians-infoma.de
@tobiasfenster
+49 731 1551-964