



APPLIED IAC: DEPLOYING BUSINESS CENTRAL IN AN AZURE CONTAINER INSTANCE USING BICEP

SPEAKER INTRO

TOBIAS FENSTER

- Managing Partner at 4PS Germany
- Microsoft Regional Director and MVP for Azure and Business Central
- tobiasfenster on Twitter and LinkedIn
- Blog URL tobiasfenster.io



WHAT DO WE TALK ABOUT TODAY?

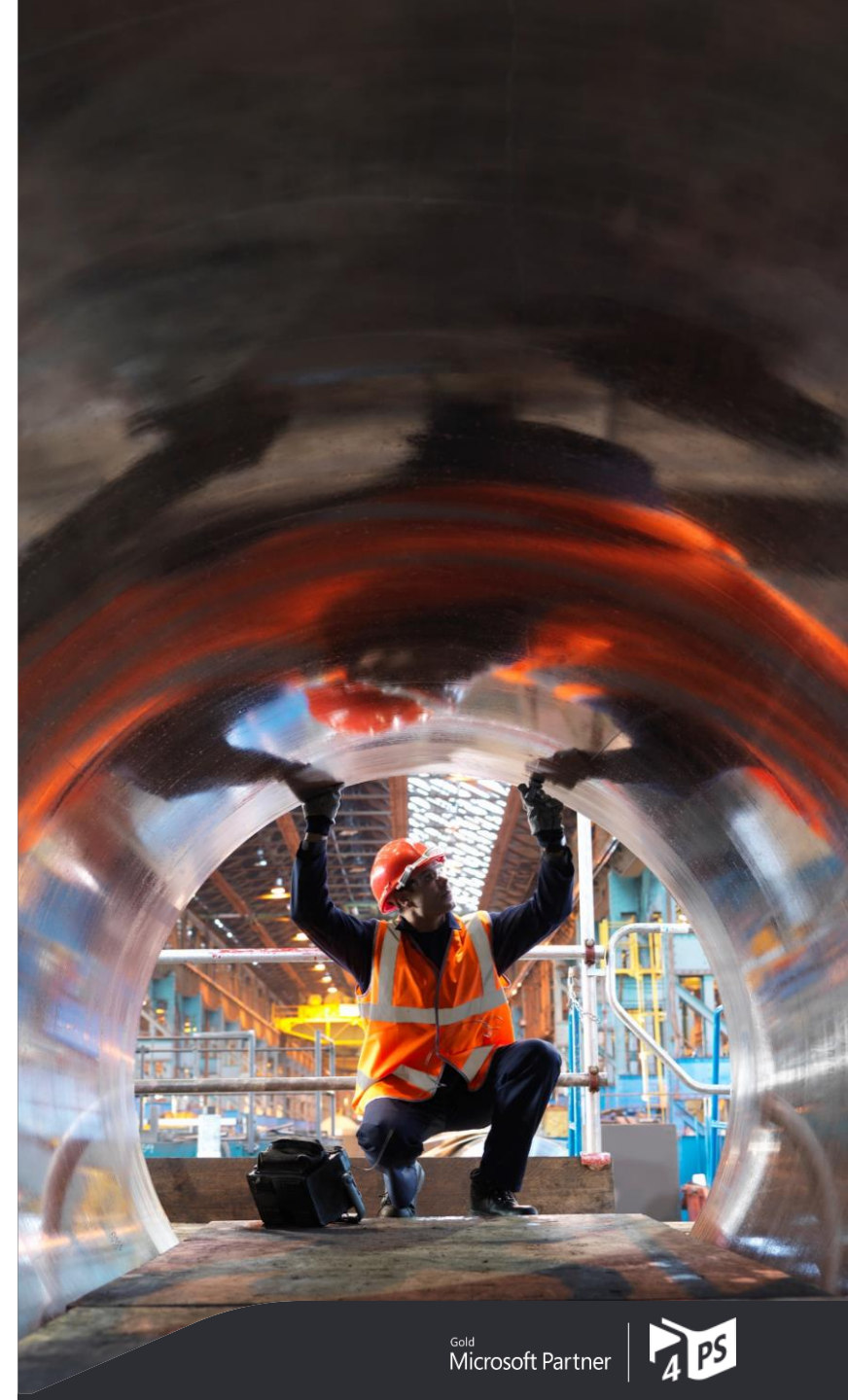
AGENDA

- Introduction to **Infrastructure as Code (IaC)** with a focus on **Bicep**
- Usage of **Azure Container instances** in the context of Microsoft Dynamics 365 **Business Central**
- Other tools to consider



WHAT IS IAC AND WHY SHOULD YOU CARE?

- What is the problem? We already have enough solutions without matching problems...
 - Infrastructure is growing **more complex** → Handling it manually takes **too long** and is **too error prone** but at the same time, **larger teams** are working on infrastructure
 - How the infrastructure **changes over time** must be tracked and you might have to **go back**
 - Development and operations (the teams creating infrastructure for software created by development) need to **work together more closely** → DevOps! But how to do that?
 - New infrastructure needs to be created **quickly**, maintenance must be done **without interruption** while at the same time needing **less human interaction**



SOUNDS LIKE A FAMILIAR PROBLEM...

- Basically: Growing complexity, working in teams, tracking changes → Adopt **best practices of development** for infrastructure as well
 - **Source control management** including version history
 - Changes are done through **pull request with reviews**
 - **Automated builds** and **automated tests**
- Conclusion: **Treat your infrastructure as code!** → IaC
- My preferred flavor: **Declarative** IaC
 - Describe **what you want**, not how you get there
 - PowerShell (or other) traditional scripts describe how you set it up, not the result



I need a VM

The VM needs to have 4 CPU cores and 8 GB of memory

The VM needs to have 2 disks with 500 GB storage each

The VM needs to have Windows Server 2022 Core installed

The VM needs to have Containers enabled and Docker installed

I need a firewall setup

The firewall needs to open port 443 and 22

The VM needs to be secured by the firewall

Or in our case:

I need an Azure Container Instance (ACI)

The ACI needs to have 2 CPU cores and 4 GB of memory

The ACI needs to run in North Europe with a given DNS name

The BC version should be the latest German sandbox

INTRODUCTION TO BICEP

BICEP

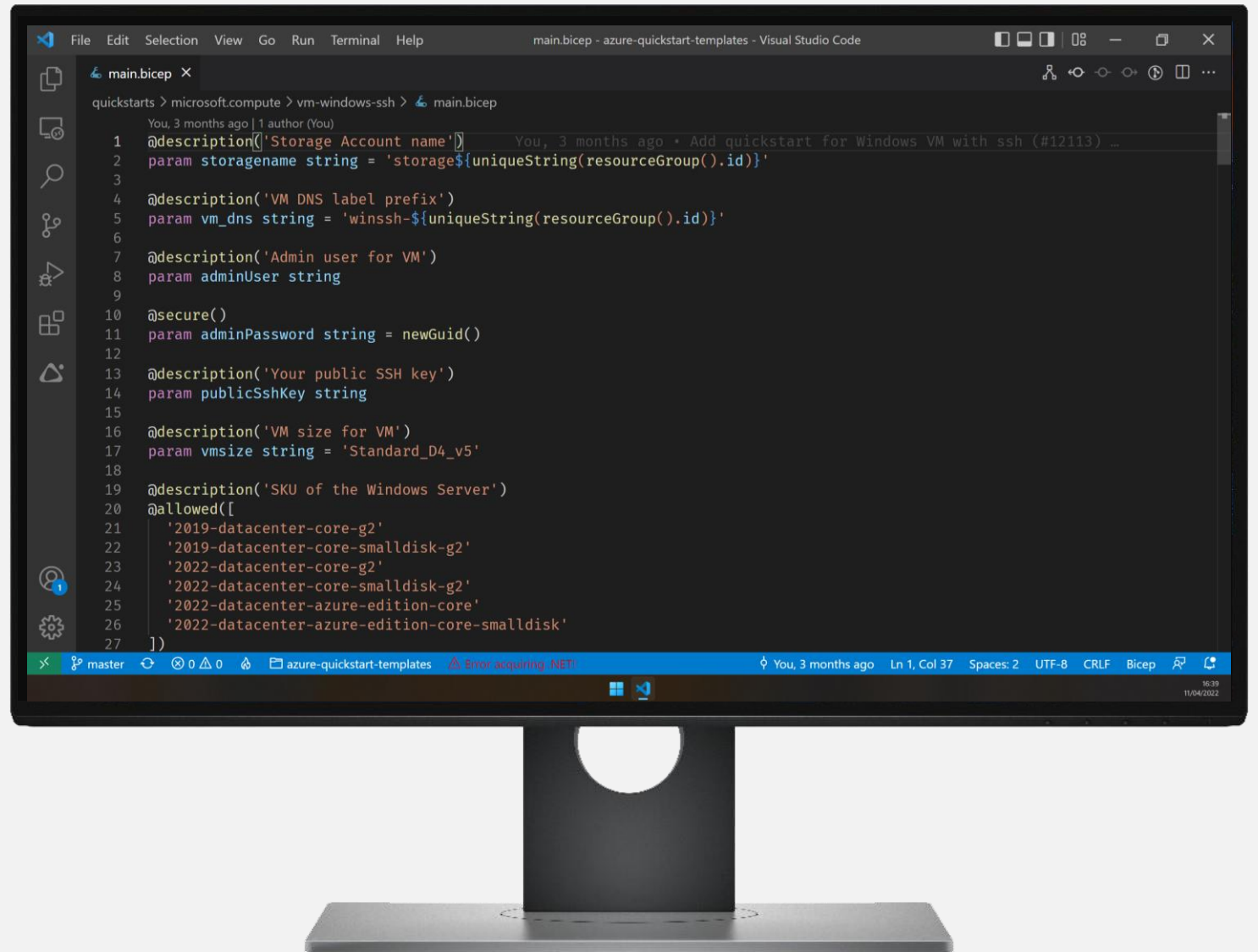
- IaC tool **native to Azure**: Might be a limitation, but also ensures quick and comprehensive support for Azure services
- **Simple** syntax (JSON-like but shorter)
- Good support in **VS Code** through an extension
- **Repeatable** deployments with **auto-managed orchestration**
- Support for **modules** and **"what if"** deployments
- No separate state file ("Azure is the state")
- No cost and open source
- Very good documentation: <https://docs.microsoft.com/en-us/azure/azure-resource-manager/bicep/overview>



DEMO

VIRTUAL MACHINE

- Deploy a Windows Azure VM that also runs some scripts on startup
- Use the Azure cloud shell to avoid any local dependencies



AZURE CONTAINER INSTANCES

- Managed Azure Container service targeted at deploying simple containerized workloads (like 1 BC container)
- Benefits compared to VMs:
 - Faster startup
 - Fully managed containerization stack
 - Easy, name-based access from the internet (but limited to 5 ports)
- Drawbacks:
 - No resource-sharing between containers
 - No access to the host, so e.g. no bccontainerhelper (but e.g. PowerShell access to the container)
- Very good documentation: <https://docs.microsoft.com/en-us/azure/container-instances/container-instances-overview>



BRIEF DETOUR

BC ARTIFACT URL PROXY

- Running BC containers requires **BC artifacts** since June 2020
- URL to be retrieved through PowerShell → **Requires PowerShell... and some time**
- BC artifact URL proxy: (unsupported) alternative to use "static URL", e.g.

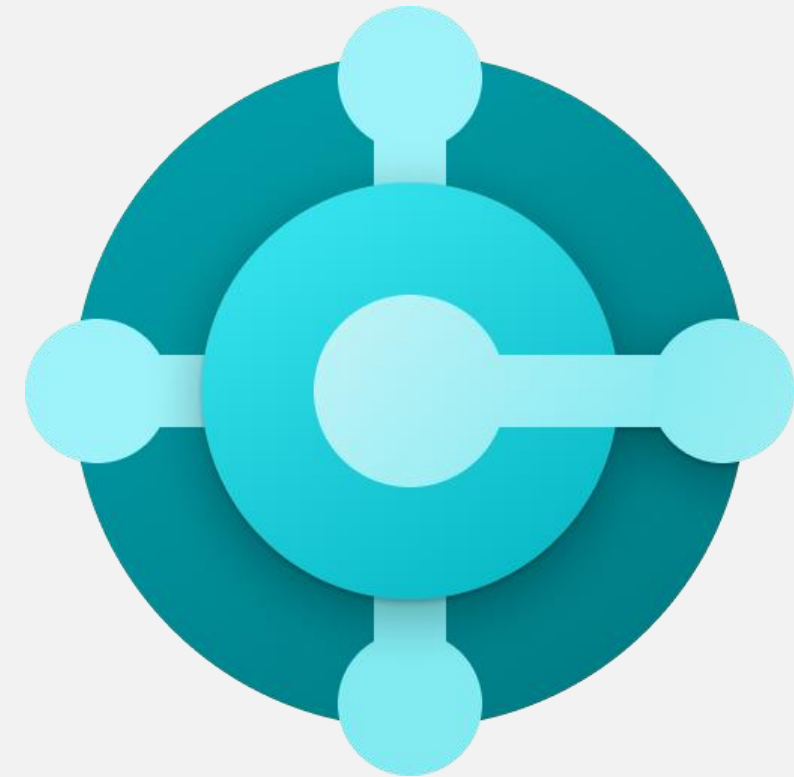
`https://bca-url-proxy.azurewebsites.net/bca-url/sandbox/de/20?DoNotRedirect=true`

returns

`https://bcartifacts.azureedge.net/sandbox/20.0.37253.38995/de`

or redirects there if you don't add **DoNotRedirect=true**

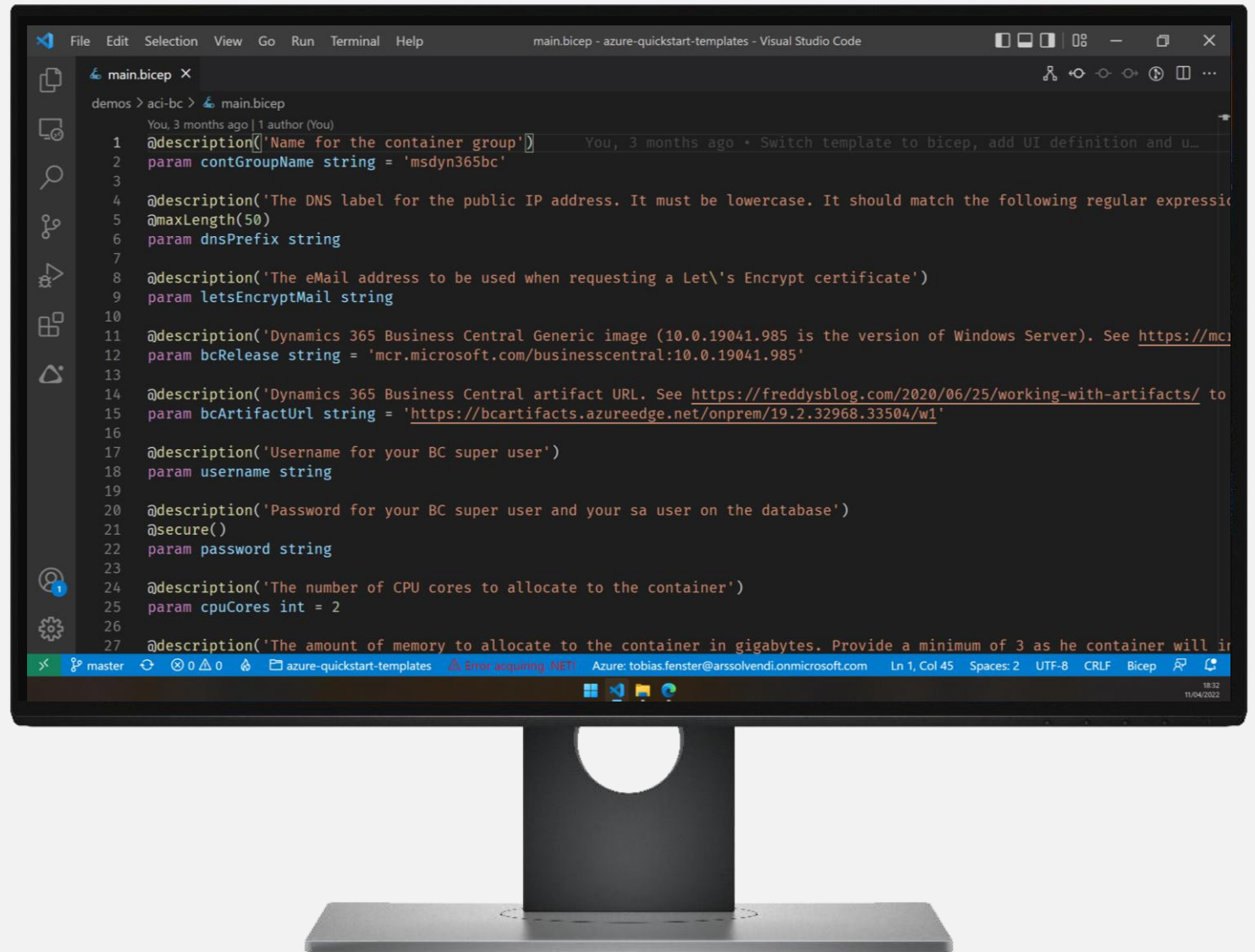
- If you want to learn more: <https://tobiasfenster.io/get-your-bc-artifact-urls-without-powershell>



DEMO

BC ACI

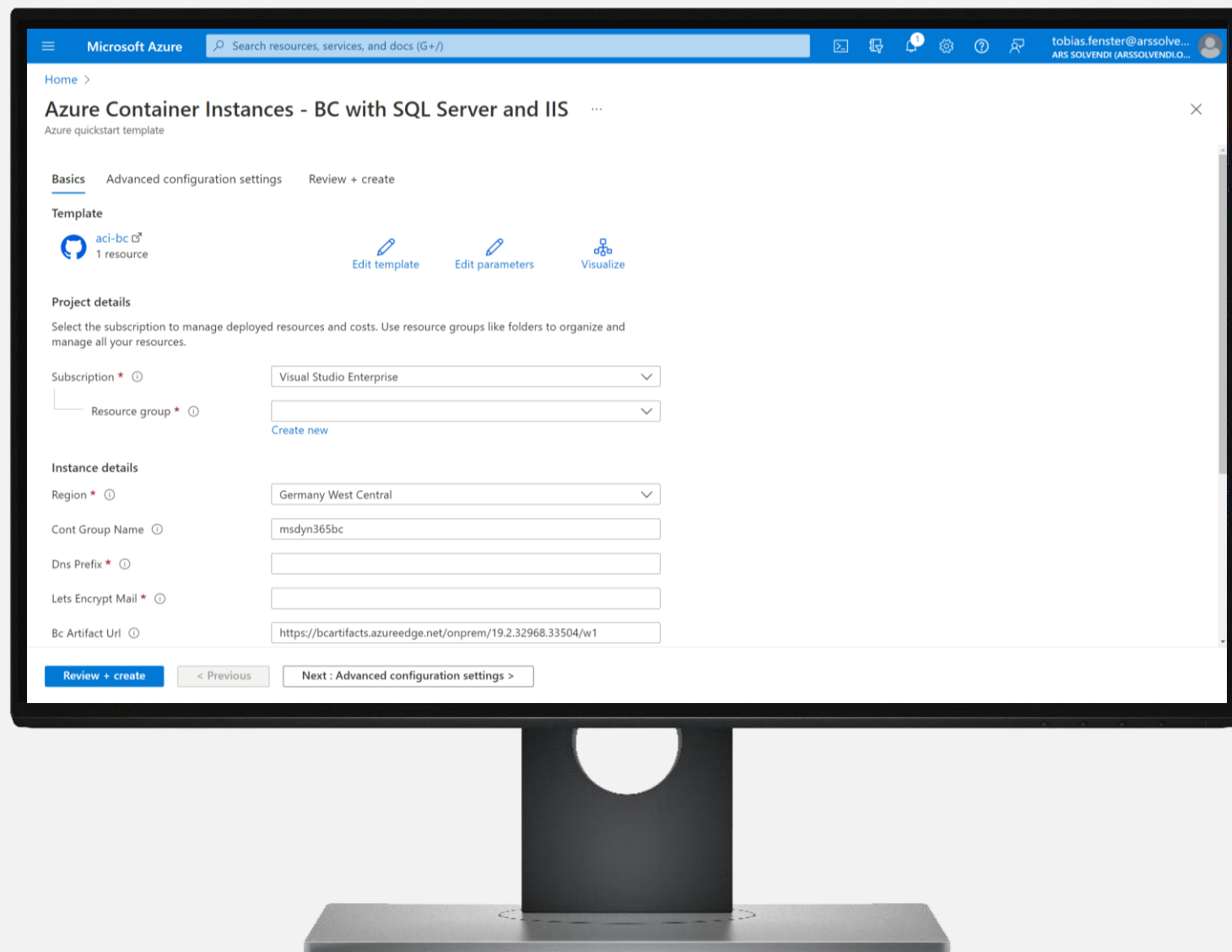
- Deploy a Microsoft Dynamics 365 Business Central container as an Azure Container Instance
- Use the Azure cloud shell to avoid any local dependencies



DEMO

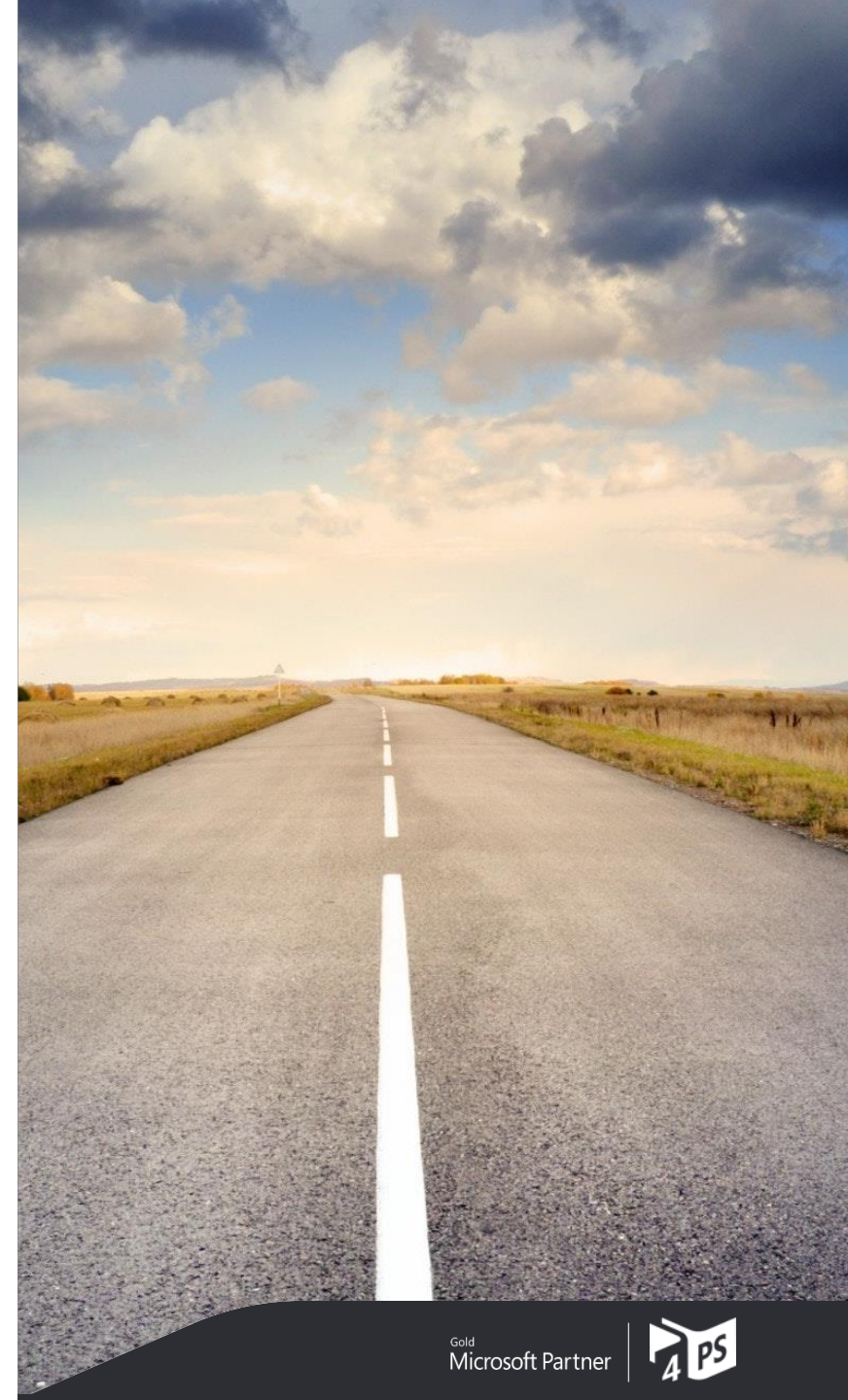
AZURE PORTAL

- “No code” IaC: Use a template from the Azure Portal
- Include UI definition



OTHER OPTIONS

- ARM Templates: Microsoft's original offering, a lot more complex
- Terraform: Open source offering by Hashicorp with a similar syntax as Bicep and multi-cloud support
- Pulumi: Support for a wide range of popular programming languages, multi-cloud support
- Others with a similar, but wider scope: Ansible, Chef, Puppet



THANK YOU FOR
YOUR ATTENTION

