



Microsoft Azure Cloud Adoption Framework in Practice

Who am I?

Sudesh Jethoe

Freelance Cloud Native Architect

sudesh@dynamicautomation.nl

[@sudeshjethoe](https://www.linkedin.com/in/sudeshjethoe/)

<https://www.linkedin.com/in/sudeshjethoe/>

2012 Linux Engineer (Managed Hosting)

2014 Middleware Specialist (Managed Infrastructure)

2016 Site Reliability Engineer (On-prem)

2018 Cloud Architect





Overview

1. Why use a “Cloud Adoption Framework”?
2. The Cloud Adoption Framework for Microsoft Azure
3. CAF Project Phases
4. Using CAF landingzones to accelerate your cloud migration

Let's make it interactive, feel free to ask questions during the presentation



Why CAF?

How many of you have previous experience with starting up cloud migrations projects?

How did that work out for you?

What went well?

What didn't go well?



Why CAF?

Many possibilities

Every company is different

No one size fits all

Where to start?



Why CAF?

Generalised approach

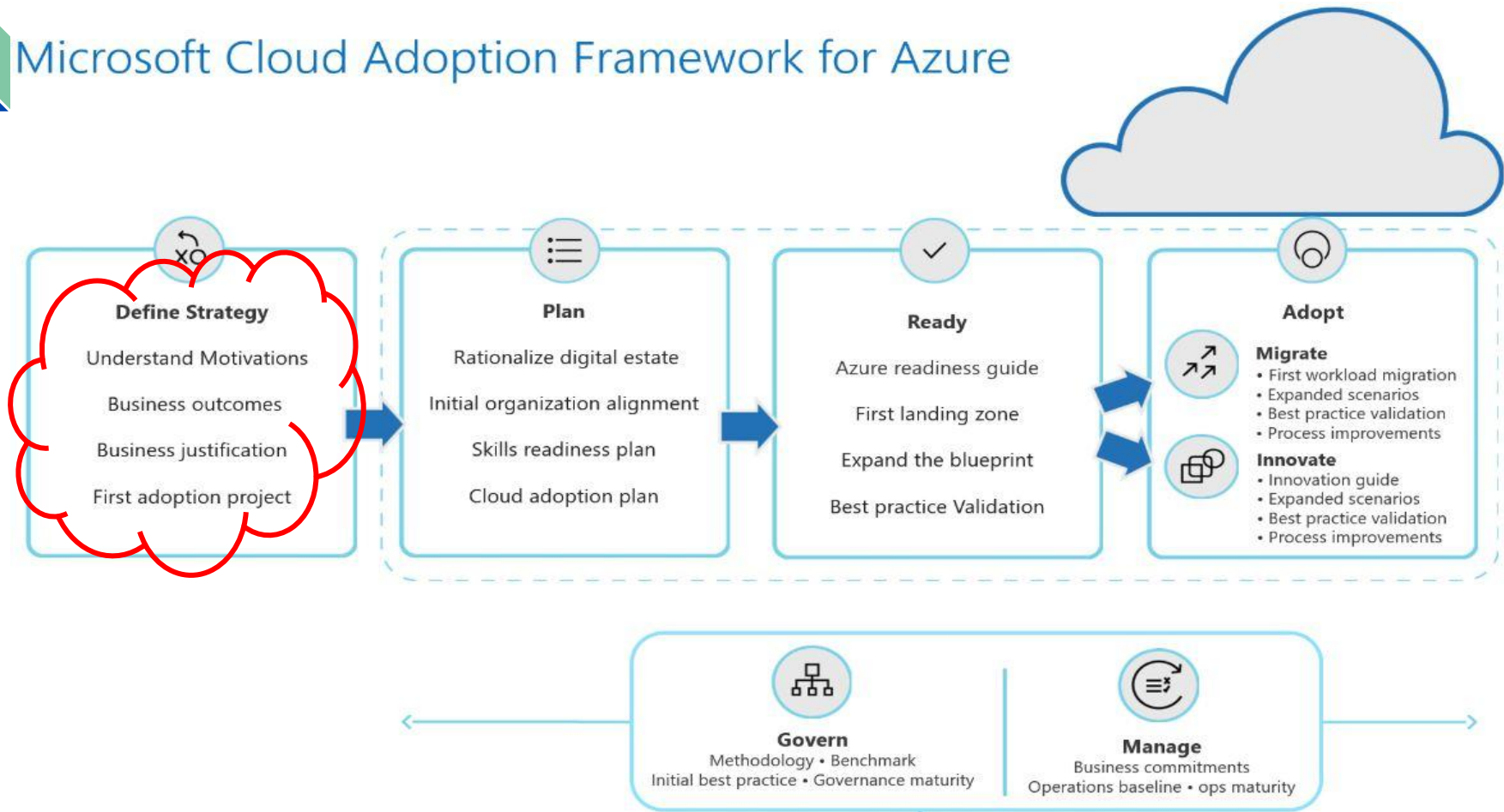
Answer the important questions

Helps define your own strategy

Based on real world experiences of Microsoft,
employees and customers



Microsoft Cloud Adoption Framework for Azure

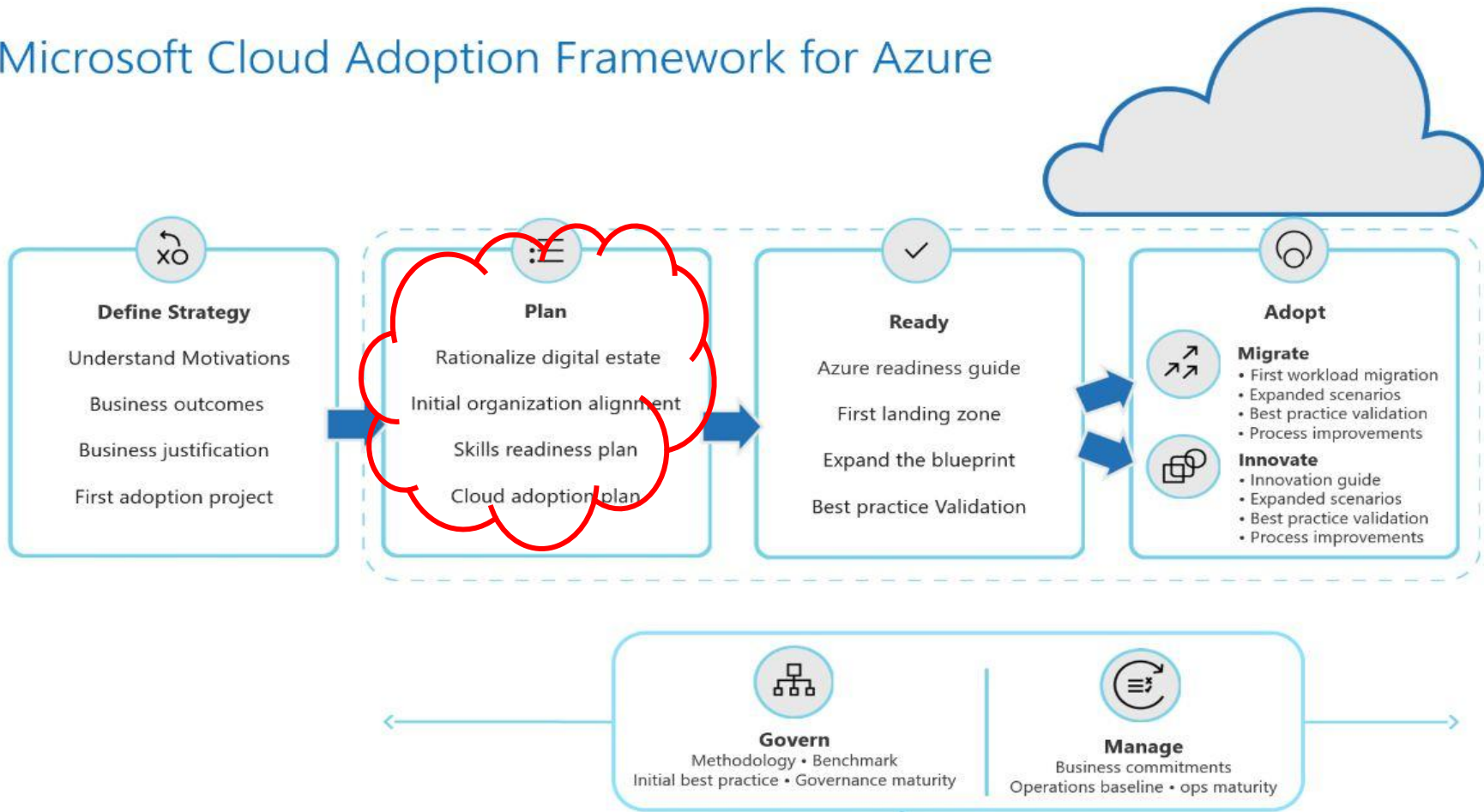




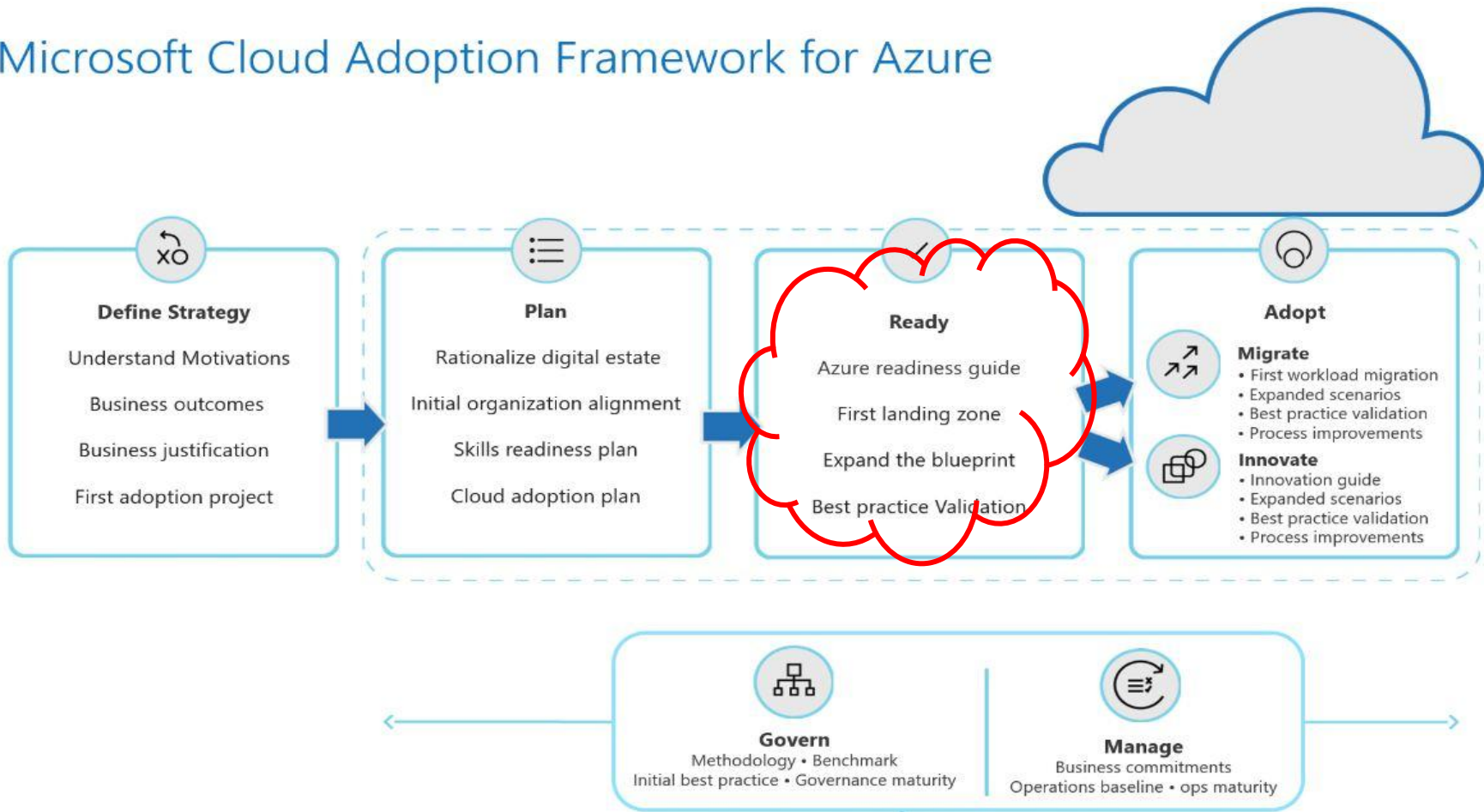
Defining your Strategy (steps)

- 01 Motivations (why?)
1. Response to critical business events
 2. Migration
 3. Leveraging of innovative capabilities
- 02 **Business Outcomes** (expectations?)
- Financial
 - Efficiency
 - Agility
 - Performance
 - Customer Experience
- 03 Building the **Business Case**
- [Cloud Myths](#)
 - [TCO calculator](#)
 - Internal Charging (OpEx vs CapEx)
- 04 Identify starting **project**

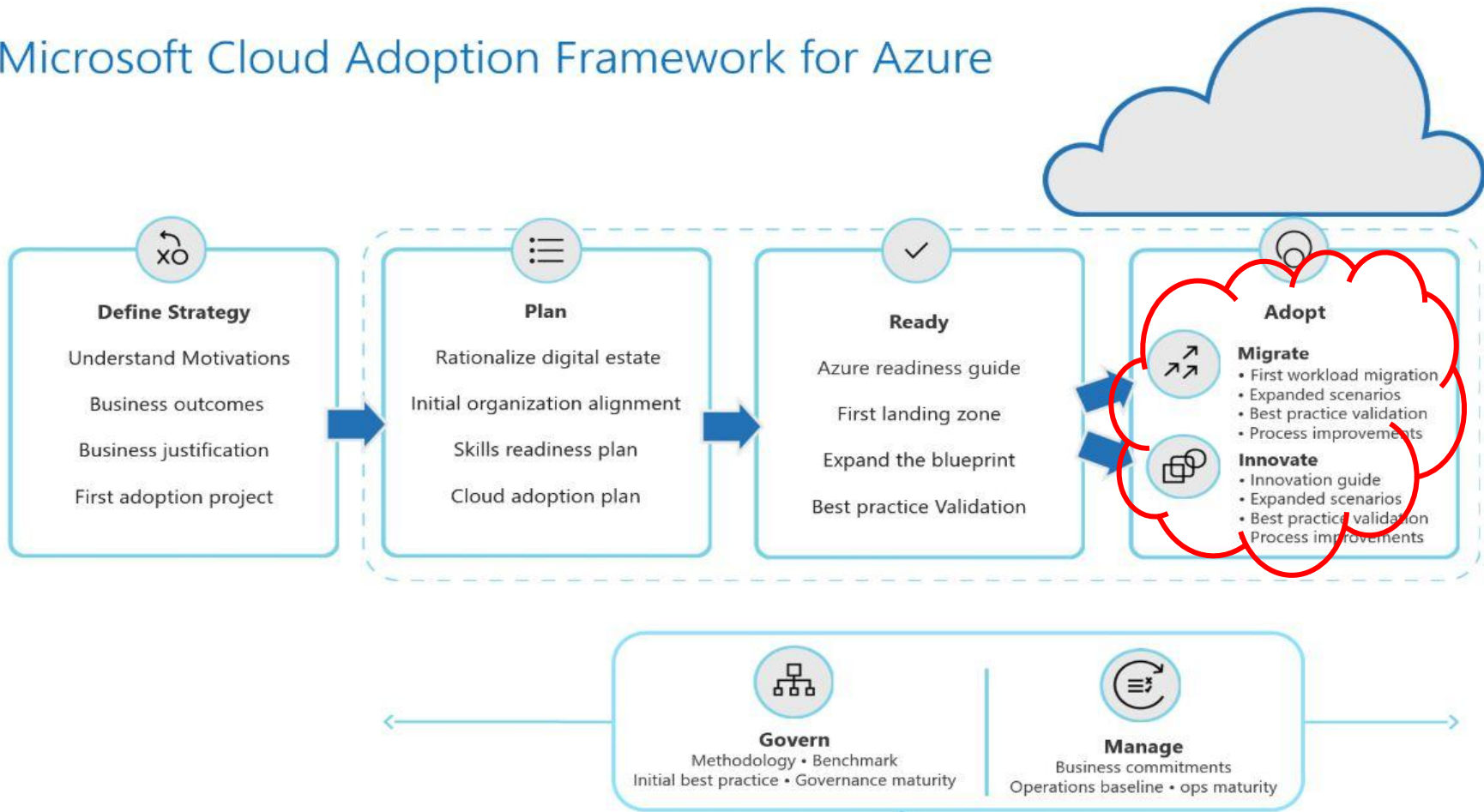
Microsoft Cloud Adoption Framework for Azure



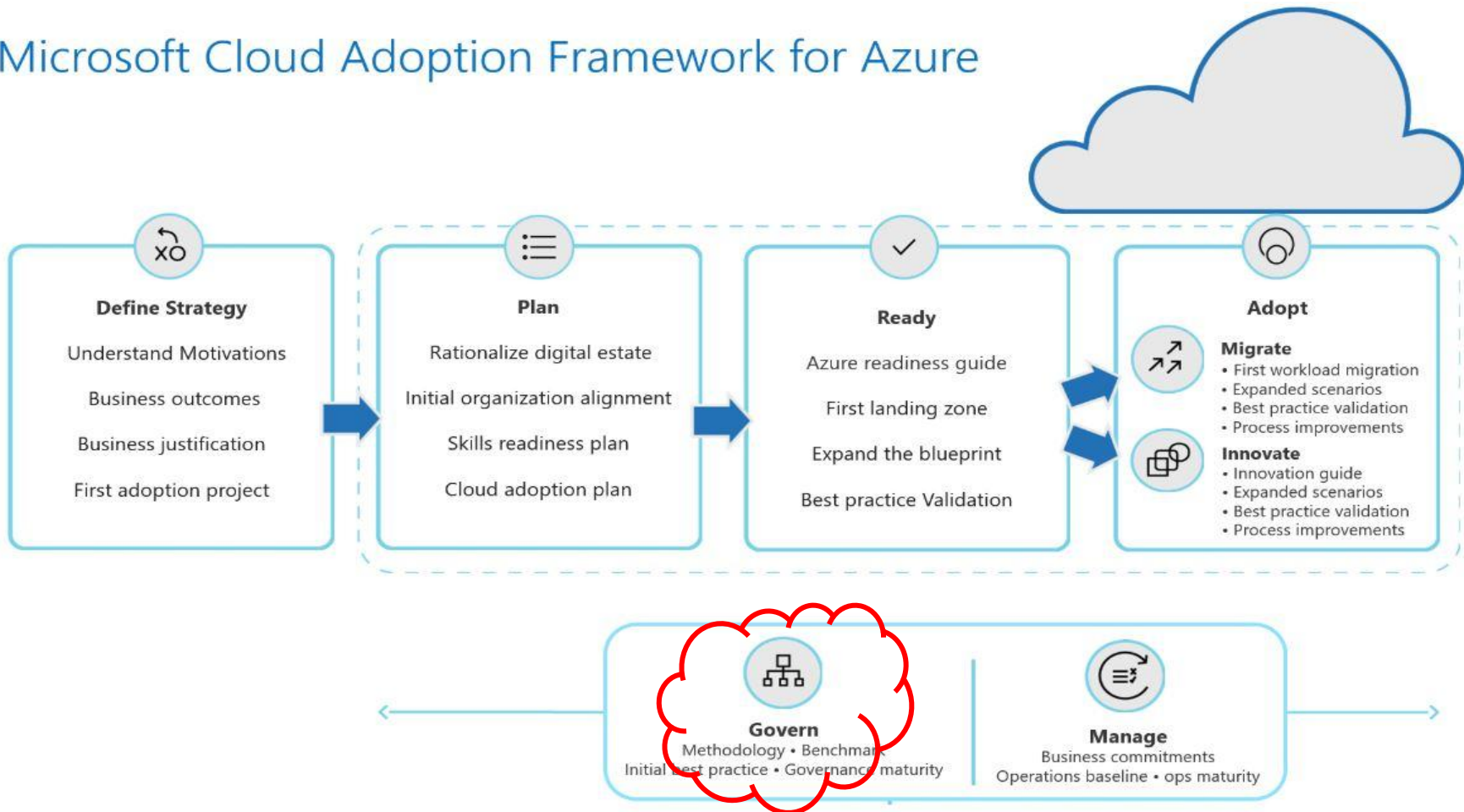
Microsoft Cloud Adoption Framework for Azure



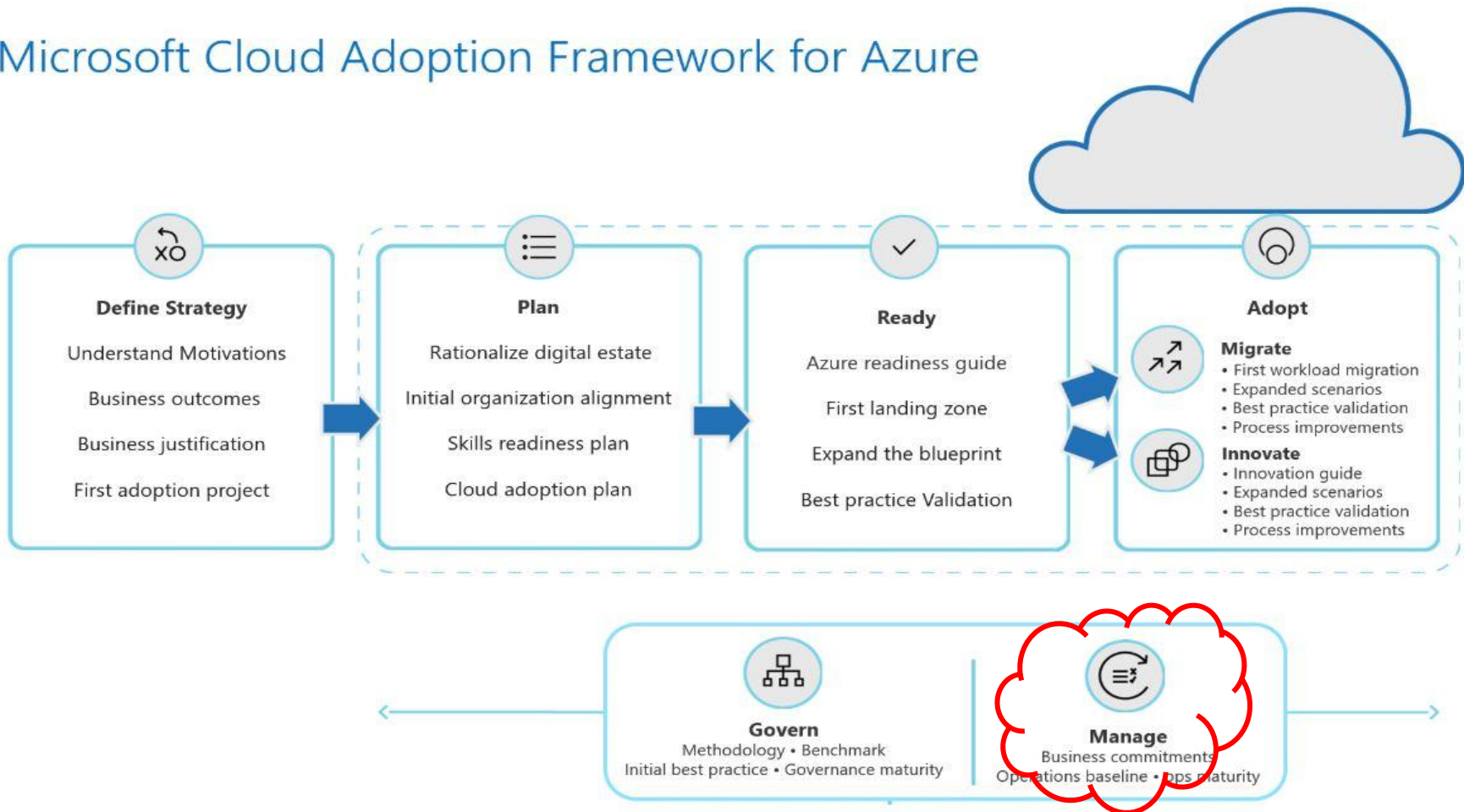
Microsoft Cloud Adoption Framework for Azure



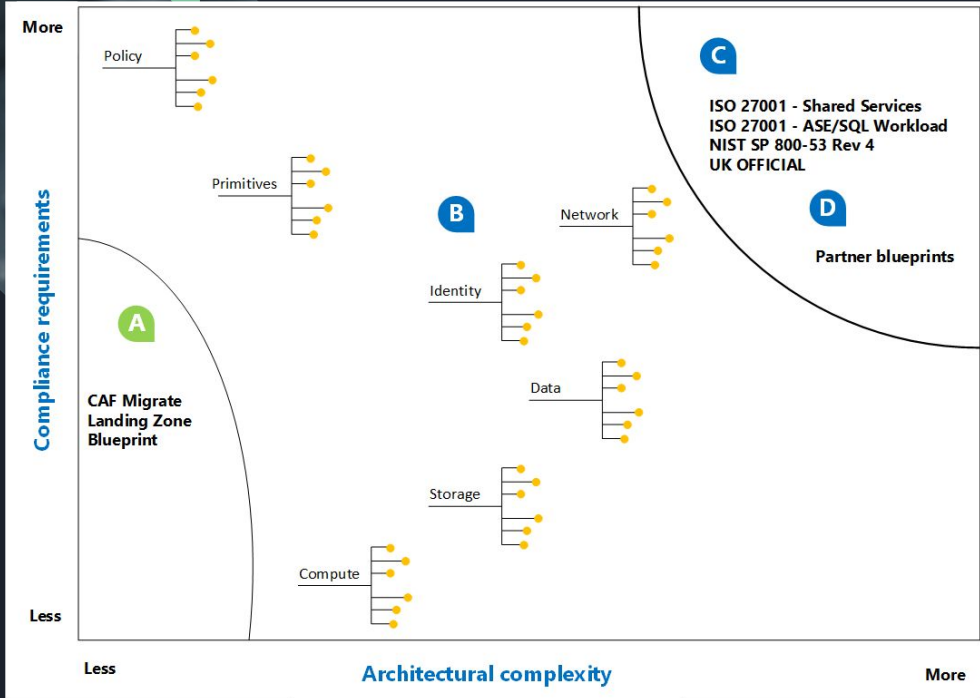
Microsoft Cloud Adoption Framework for Azure



Microsoft Cloud Adoption Framework for Azure



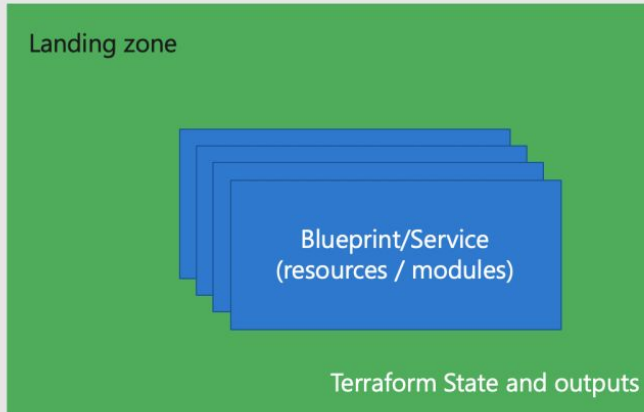
Landingzones



The need of composable architectures to fulfill regulatory and architectural demands

Layered approach using Terraform

Anatomy of a landing zone



Examples:

- Virtual datacenter level1
- Virtual datacenter level2
- Hub-Spoke
- Lambda architecture with Databricks
- Mobile application development platform
- Citrix cloud

A landing zone orchestrates blueprints to build a solution. It owns the Terraform state for all the component it deploys.

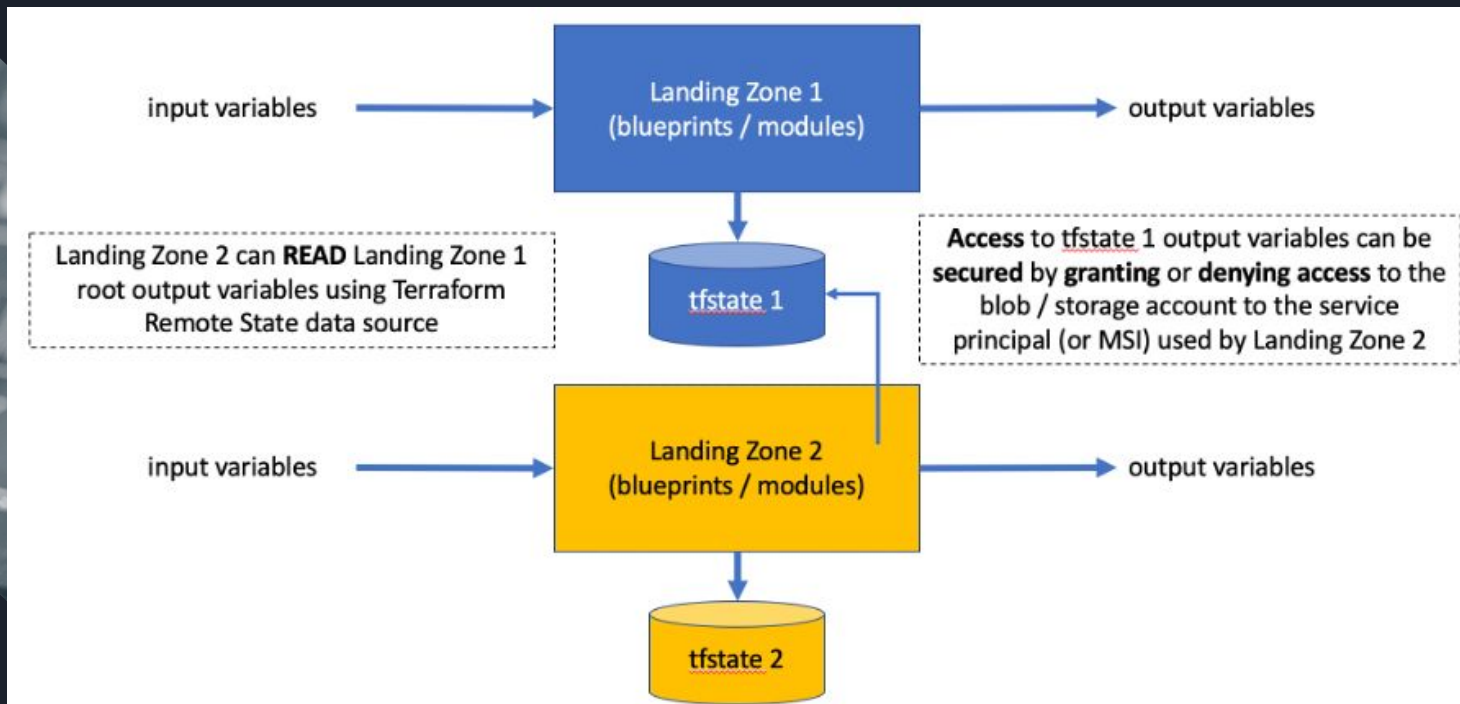


Landingzones are composed of blueprints

- ✓ `landingzone_vdc_demo`

- > `blueprint_networking_shared_egress`
- > `blueprint_networking_shared_services`
- > `blueprint_networking_shared_transit`

Infrastructures are composed by stacking landingzones



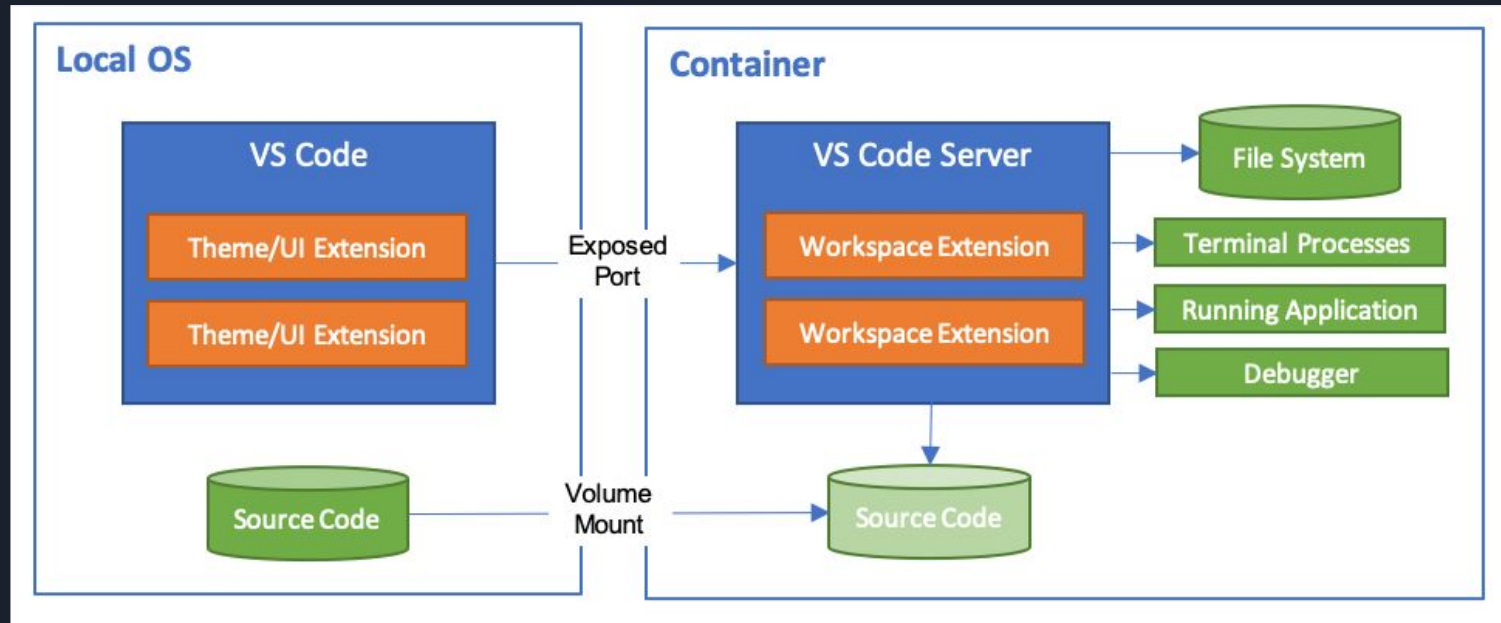


Stacking as implemented by CAF

1. Level0: components required to run terraform
 - storage account
 - keyvault
 - service principal
2. CAF_Foundation: generic elements for any infrastructure
 - Generic Policies
 - SIEM (Azure Sentinel)
 - Monitoring
 - Auditing
3. Landingzone_Application: required for the service to be run
 - Loadbalancer
 - AppService
 - Databases

Deployment using CAF Rover

CAF Rover: Generalized development environment using VSCode and Docker





Thanks!

Sudesh Jethoe
Freelance Cloud Native Architect
sudesh@dynamicautomation.nl
[@sudeshjethoe](https://www.linkedin.com/in/sudeshjethoe/)
<https://www.linkedin.com/in/sudeshjethoe/>





Reference Materials

[Microsoft Learn: Cloud Adoption Framework for Azure](#)

[Microsoft Docs: Cloud Adoption Framework for Azure](#)

[Github CAF Rover](#)

[Github CAF Rover Level0 Launchpads](#)

[Github CAF Landingzones](#)

[Github CAF Creating Composable Landingzones](#)