Getting the right Azure Foundation - What makes a **great** Landing Zone?

Please note – the views/opinions in this presentation are entirely my own. This presentation will not be kept updated after Technical Summit 2024 (October 2024) – so may be outdated if downloaded afterwards.

If in any doubt, please check latest documentation and MS Links for updated info!

technicalSUMMIT

Jake Walsh

jakewalsh.co.uk

@jakewalsh90

Who am I?

- Senior Solution Architect – Microsoft Azure
  - Microsoft MVP
  - Microsoft MCT
  - HashiCorp Ambassador
  - Citrix CTA

- Blog @ jakewalsh.co.uk
- X @jakewalsh90

Please note: the Views and Opinions in this Presentation are entirely my own. If in any doubt – please refer to vendor documentation.

Agenda

What is a Landing Zone?

Key Areas / Benefits / Why?

A **GREAT** Landing Zone!

Resources

The key to success = A solid foundation

Remember...

Landing Zone Design is complex but structured – ~30 mins = basic overview.

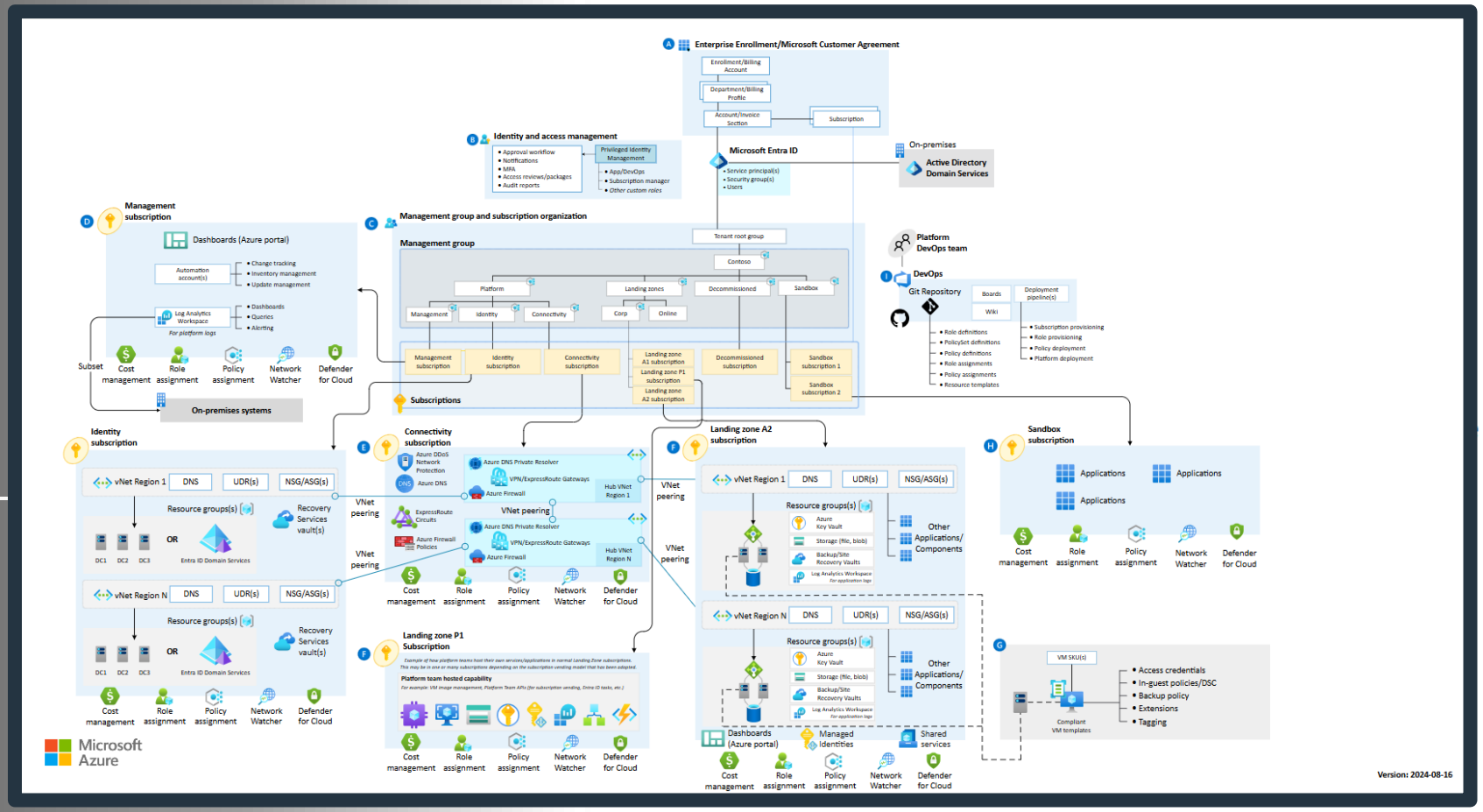
Please review guidance!

What is an Azure Landing Zone?

What is an Azure landing zone?

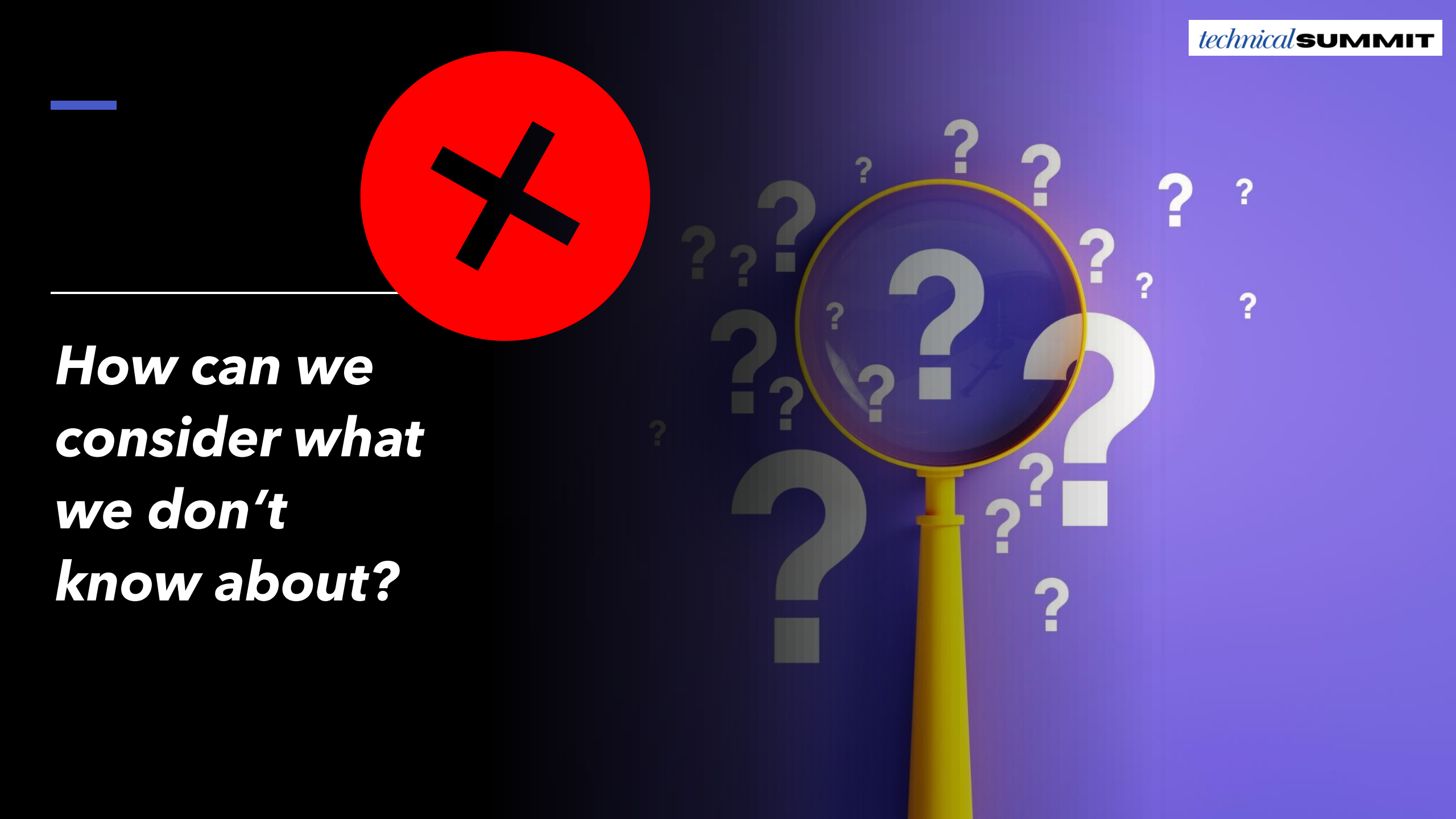
Article • 07/22/2024 • 28 contributors

Feedback



The Azure landing zone conceptual architecture (*see figure 1*) represents an opinionated target architecture for your Azure landing zone. You should use this conceptual architecture as a starting point and tailor the architecture to meet your needs.

The Challenge



How can we consider what we don't know about?

What we can do:

A few of the things we can do…

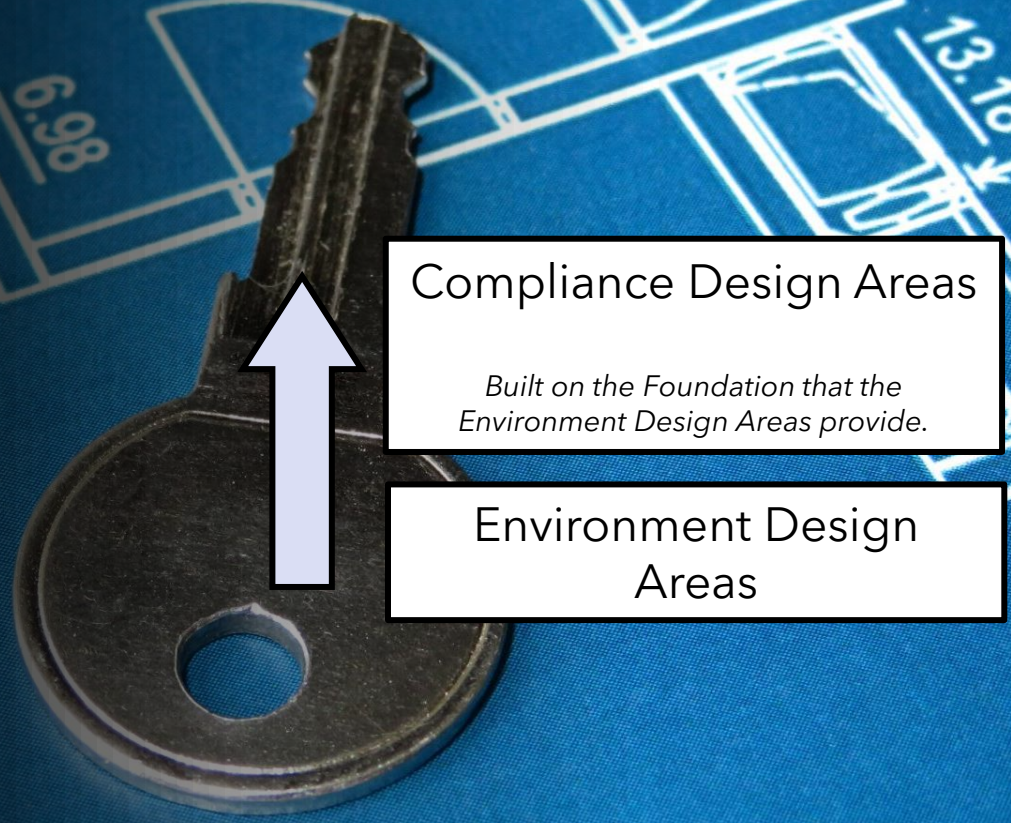
- Consider Design Principles and Areas – **plan!**
- Design for **flexibility** and **modularity**
- Design with the **future in mind**
- Design for **change** – sizing and services
- Follow appropriate **documentation & standards**
- Consider our **business** and **application** needs and likely changes
- Use all of the **GREAT** Resources available to help

Design Principles

- Subscription democratization
- Policy-driven governance
- Single control and management plane
- Application-centric service model
- Alignment with Azure-native design and roadmaps

Key Concept – Design Areas

- A set of Key Considerations and Areas that provide guidance on the areas that are **essential** to consider when starting and building your LZ in Azure.
- These are the foundation for a successful journey in Azure.
- The good news – there is extensive help and guidance via Microsoft Documentation.
- Important to note – these require ongoing consideration!



technicalSUMMIT

Environment Design Areas

The Foundation for Growth

Environment Design Areas

- Azure Billing and Microsoft Entra Tenant
- Identity and Access Management
- Network Topology and Connectivity
- Resource Organization

Environment Design Areas

Azure Billing & Microsoft Entra ID

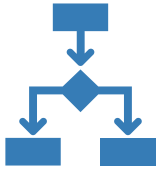
- Agreement Type – MCA/CSP/etc.
- Tenant(s)
- Subscriptions
- Management / Support

Network Topology & Connectivity

- Topology
- VPNs
- ExpressRoute
- SD-WAN
- Firewalling
- User VPN
- NSGs/ASGs

Identity & Access Management

- Entra ID
- Active Directory
- 3rd Party Authentication
- Licensing
- MFA
- RBAC
- Break Glass

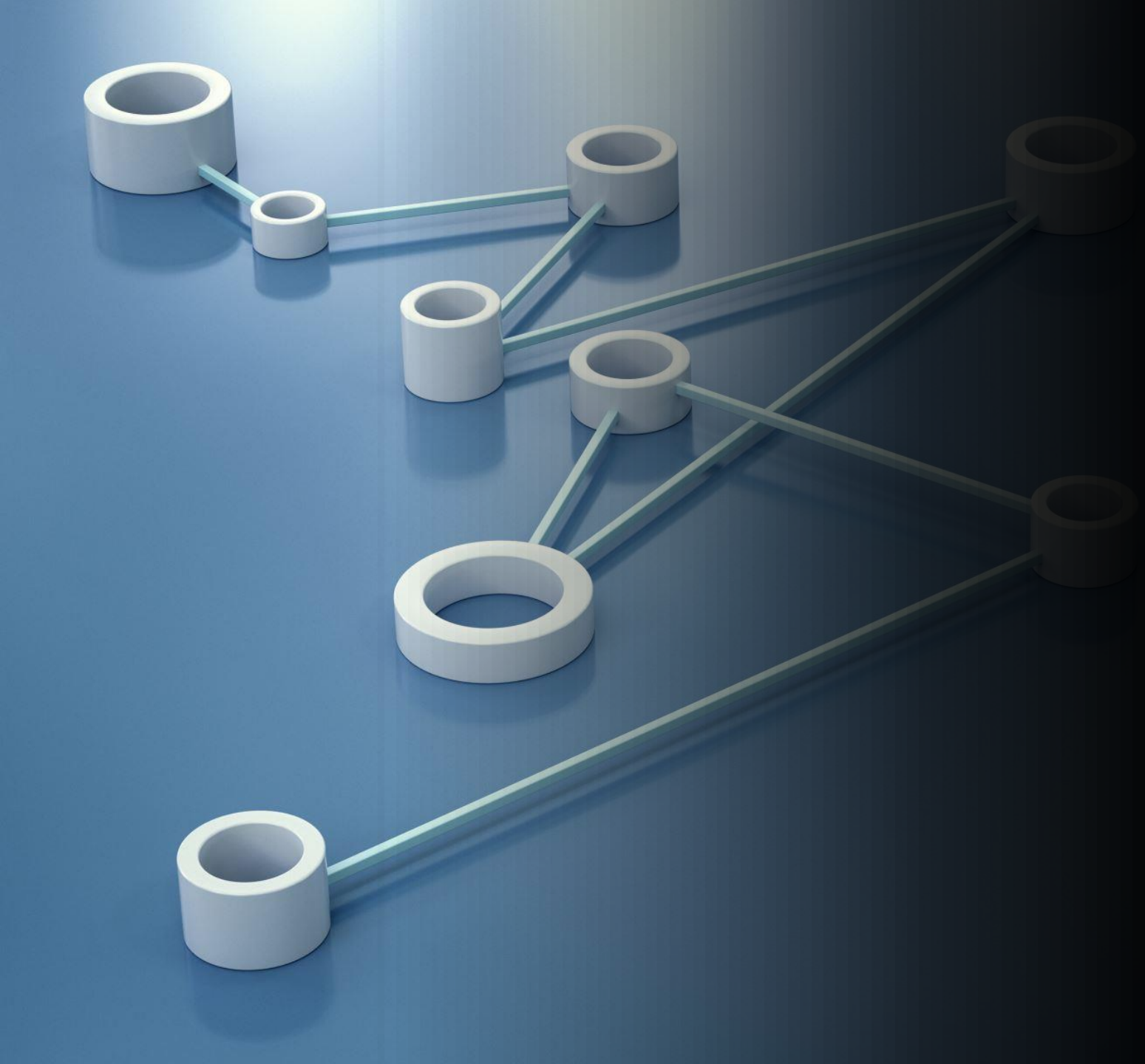


Resource Organization

- Management Groups
- Resource Groups
- Naming
- Tagging
- Alignment to Business Structure
- Global Regions

Compliance Design Areas

Building on the Foundation



Compliance Design Areas

- Security
- Management
- Governance
- Platform Automation and DevOps

Compliance Design Areas

Security

- Logging, Monitoring, Alerting
- Shared Responsibility
- Encryption
- SIEM / SOC
- Defender
- Vulnerability Management

Governance

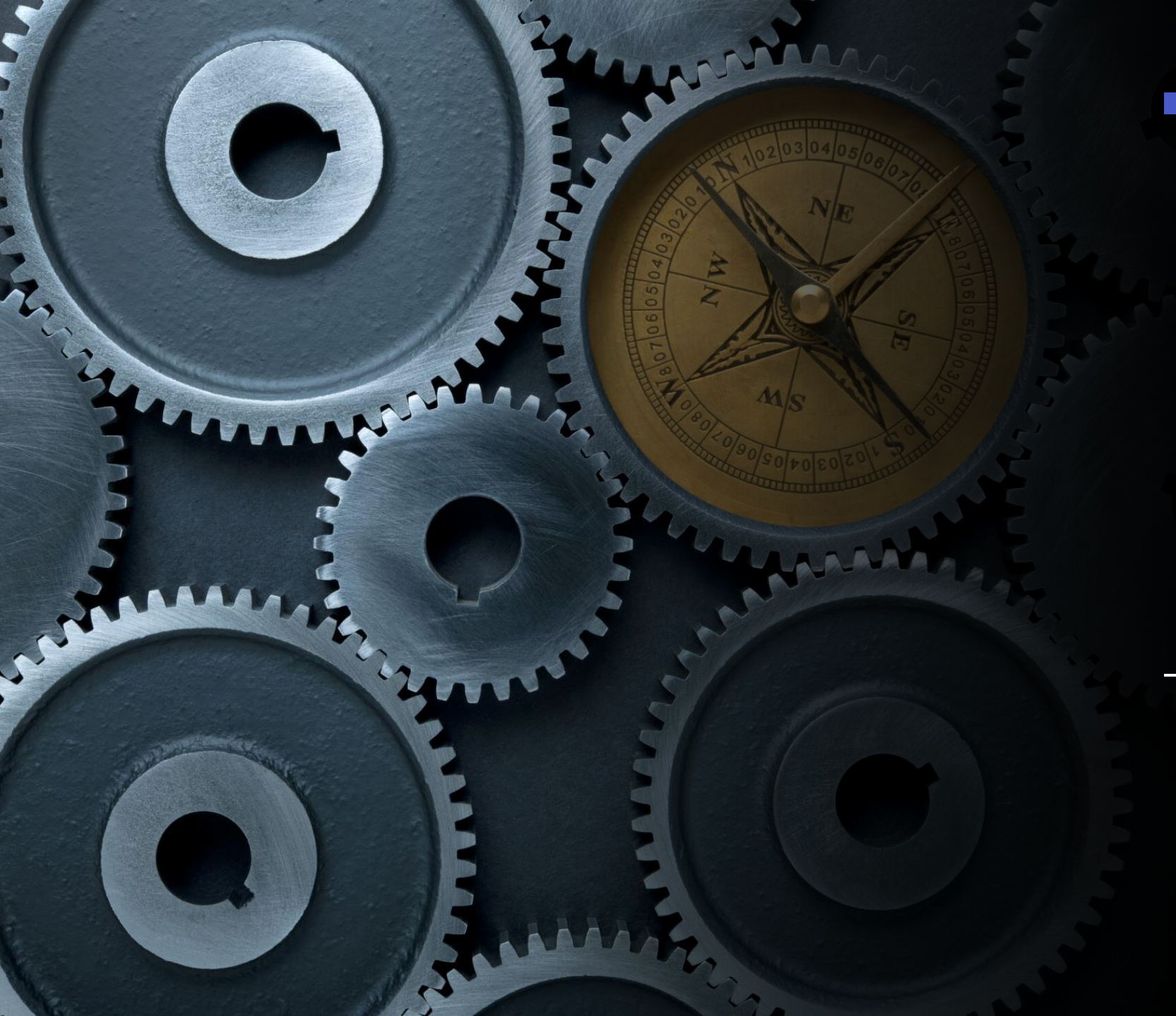
- Management Baseline
- Cost Management
- Azure Policy
- Entra Entitlement Management
- Advisor

Management

- Inventory & Visibility
- Operational Compliance
- Optimization
- Protection and Recovery
- Platform Management
- Workload Management

Platform Automation and DevOps

- Automation
- DevOps – Teams and Topologies
- Infrastructure as Code
- Lifecycle



Implementation Options

Implementation option	Description	Deployment instructions
Enterprise-scale foundation	Enterprise-ready platform foundation with all the necessary shared services to support the full IT portfolio, where connectivity can be added later as needed. Design principles Design areas	Deploy to Azure Readme: foundation Readme: Network topology (Virtual WAN) Readme: Network topology (hub-spoke)
Azure landing zones modular	Modular approach using Bicep for deploying the core platform capabilities.	Readme: Bicep modules
Enterprise-scale for small enterprises	This reference implementation is meant for organizations that don't have a large IT team and do not require fine grained administration delegation models.	Deploy to Azure Readme
Enterprise-scale for Azure Government	Reference implementation that can be deployed to Azure Government Cloud.	Deploy to Azure Readme
CAF enterprise-scale landing zone (Azure China 21Vianet regions)	Reference implementation that can be deployed to Azure clouds in China.	Deploy to Azure Deploy
Azure landing zones Terraform module	Deploys an enterprise-ready platform foundation using Terraform. Use this option when managing your platform using Terraform and need to accelerate delivery of the recommended resource hierarchy and governance model. Shared services, network connectivity, and application workloads can be integrated into your deployment or managed independently.	Readme
Microsoft Cloud for Sovereignty	A sovereign landing zone uses the same code base as the Azure landing zone Bicep approach but has more orchestration and deployment automation capabilities. It also has Azure Policy initiatives and assignments to help meet sovereignty requirements for public-sector customers, partners, and independent software vendors (ISVs).	Readme

Accelerator vs Customized

Customize approach

Expand table

Implementation option	Description	Deployment instructions
Partner landing zones	Partners who provide offerings aligned to the Ready methodology of the Cloud Adoption Framework can provide their own customized implementation option. Design principles	Find a partner

Accelerator vs Customized

Going beyond design areas...

Beyond the Design Areas – 3 Key Considerations

Future Considerations

- *How might our needs change over time?*
- *What needs could arise in the next 6,12,24 months?*
- *Divestment / M&A?*

Application Changes

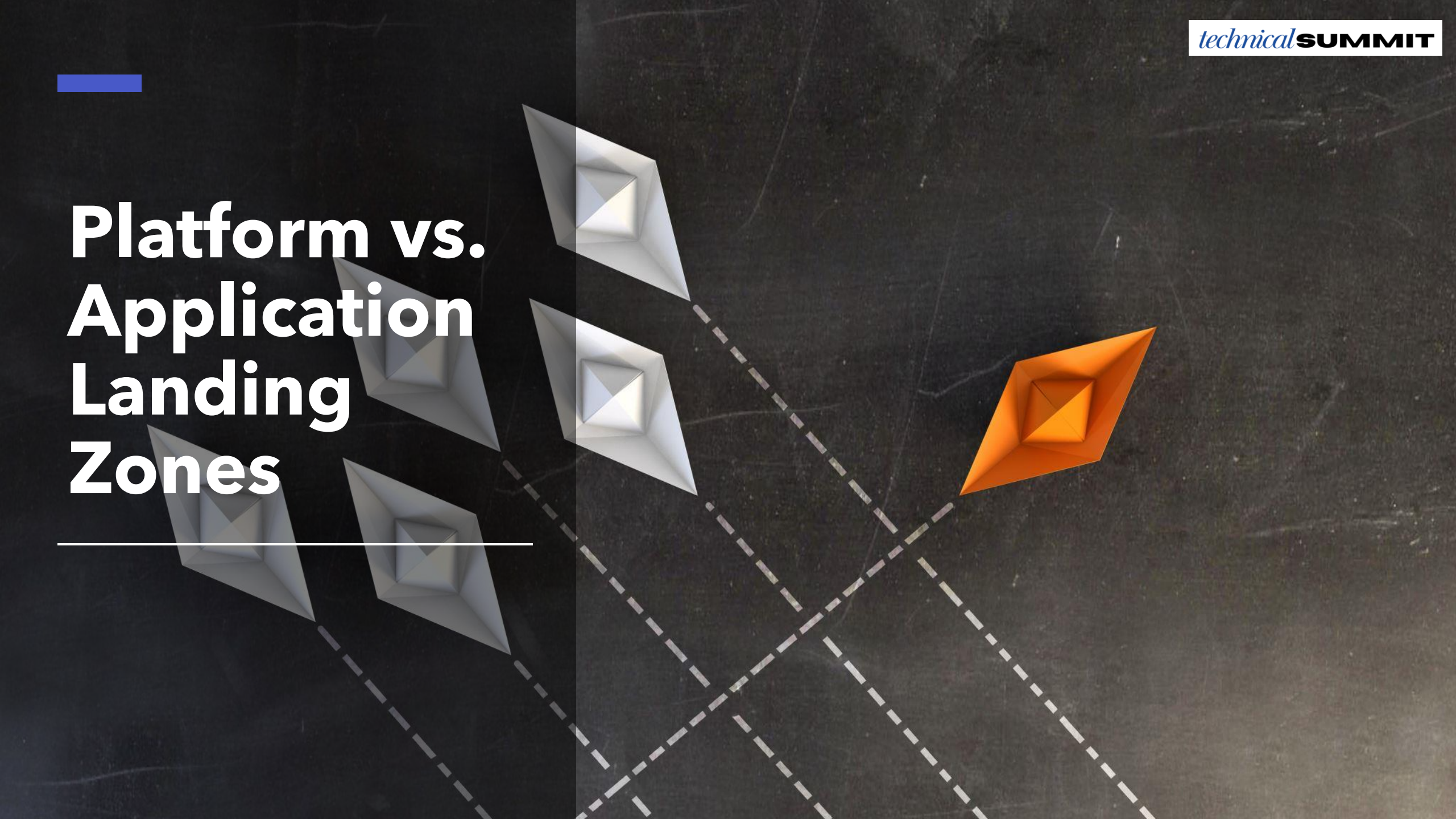
- *What changes to core LOB applications may arise?*
- *What new apps do we need to plan for?*

Organisational Changes

- *Staffing?*
- *Geographies?*
- *Compliance?*

technicalSUMMIT

Platform vs. Application Landing Zones



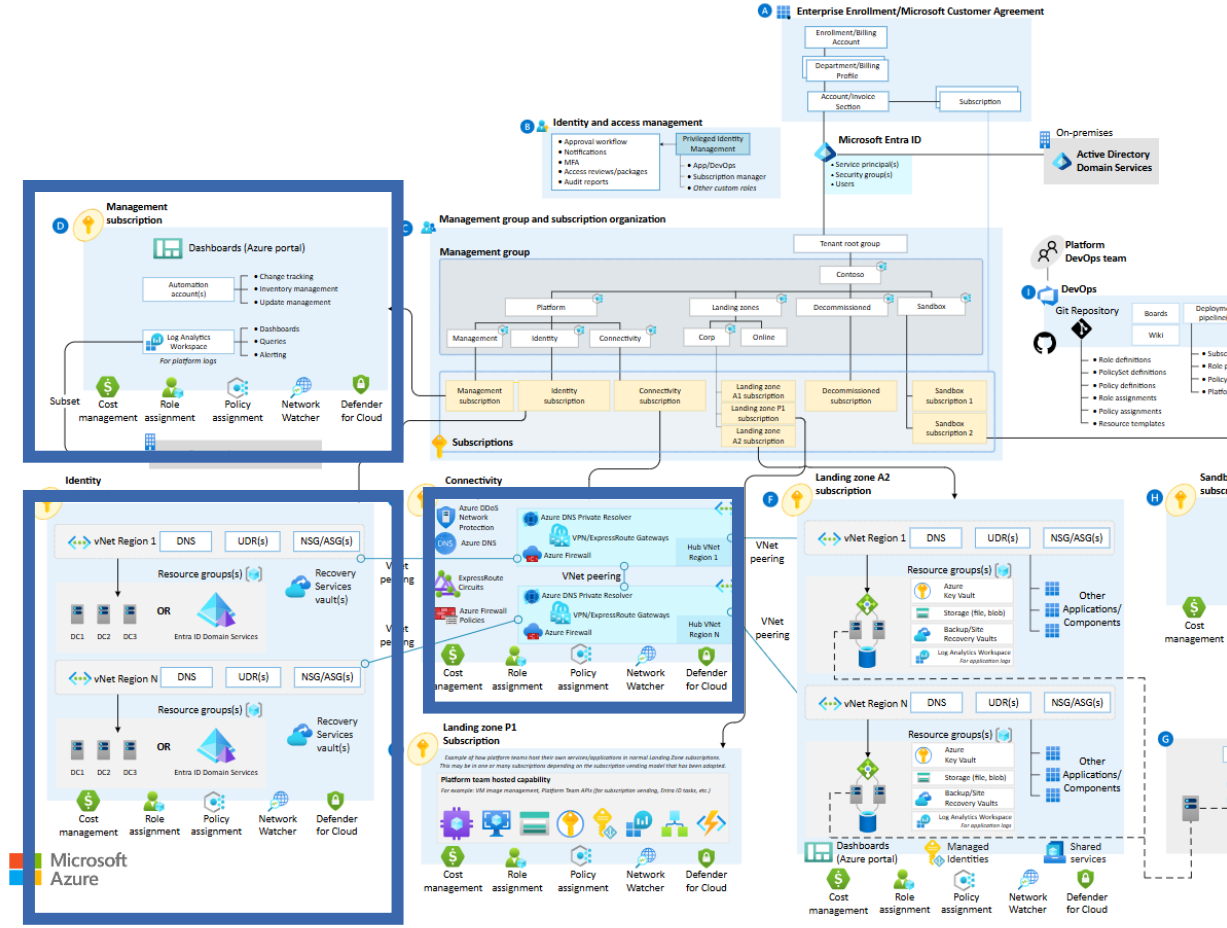
Platform Landing Zones

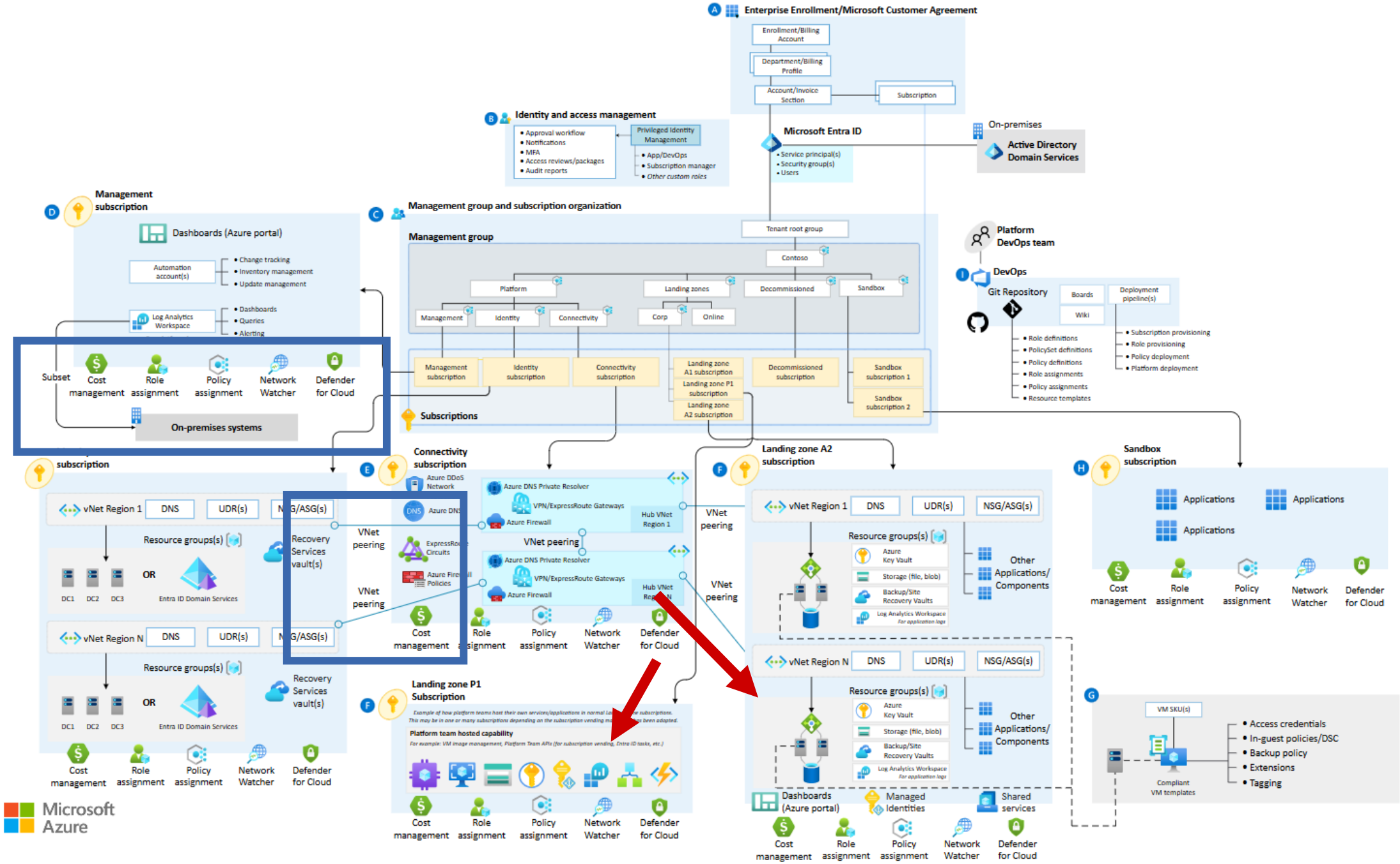
technicalSUMMIT

Platform Landing Zone

- **Shared Services** – Identity, Connectivity, Management
- **Consolidation is key** – operational efficiency through removal of unnecessary replication.
- **Platform Services** – provide to other areas/systems/Subscriptions.

Platform Services – provide the “City” and most of the supporting infrastructure (key services).




A Enterprise Enrollment/Microsoft Customer Agreement
Enrollment/Billing Account
Department/Billing Profile
Account/Invoice Section
Subscription
B Identity and access management
Approval workflow
Notifications
MFA
Access reviews/packages
Audit reports
Privileged Identity Management
App/DevOps
Subscription manager
Other custom roles
Microsoft Entra ID
Service principal(s)
Security group(s)
Users
On-premises
Active Directory Domain Services
D Management subscription
Dashboards (Azure portal)
Automation account(s)
Change tracking
Inventory management
Update management
Log Analytics Workspace
Dashboards
Queries
Alerting
Subset
Cost management
Role assignment
Policy assignment
Network Watcher
Defender for Cloud
On-premises systems
C Management group and subscription organization
Management group
Tenant root group
Contoso
Platform
Landing zones
Decommissioned
Sandbox
Management
Identity
Connectivity
Corp
Online
Management subscription
Identity subscription
Connectivity subscription
Landing zone A1 subscription
Landing zone P1 subscription
Landing zone A2 subscription
Decommissioned subscription
Sandbox subscription 1
Sandbox subscription 2
Subscriptions
Platform DevOps team
I DevOps
Git Repository
Boards
Deployment pipeline(s)
Wiki
Role definitions
PolicySet definitions
Policy definitions
Role assignments
Policy assignments
Resource templates
Subscription provisioning
Role provisioning
Policy deployment
Platform deployment
subscription
vNet Region 1
DNS
UDR(s)
NSG/ASG(s)
Resource groups(s)
OR
DC1
DC2
DC3
Entra ID Domain Services
Recovery Services vault(s)
vNet Region N
Cost management
Role assignment
Policy assignment
Network Watcher
Defender for Cloud
E Connectivity subscription
Azure DDoS Network
Azure DNS Private Resolver
VPN/ExpressRoute Gateways
Azure Firewall
Hub VNet Region 1
Azure DNS
ExpressRoute Circuits
Azure Firewall Policies
VNet peering
F Landing zone A2 subscription
Other Applications/ Components
Azure Key Vault
Storage (file, blob)
Backup/Site Recovery Vaults
Log Analytics Workspace
Dashboards (Azure portal)
Managed Identities
Shared services
H Sandbox subscription
Applications
F Landing zone P1 Subscription
Platform team hosted capability
G
VM SKU(s)
Compliant VM templates
Access credentials
In-guest policies/DSC
Backup policy
Extensions
Tagging
Microsoft Azure
Version: 2024-08-16

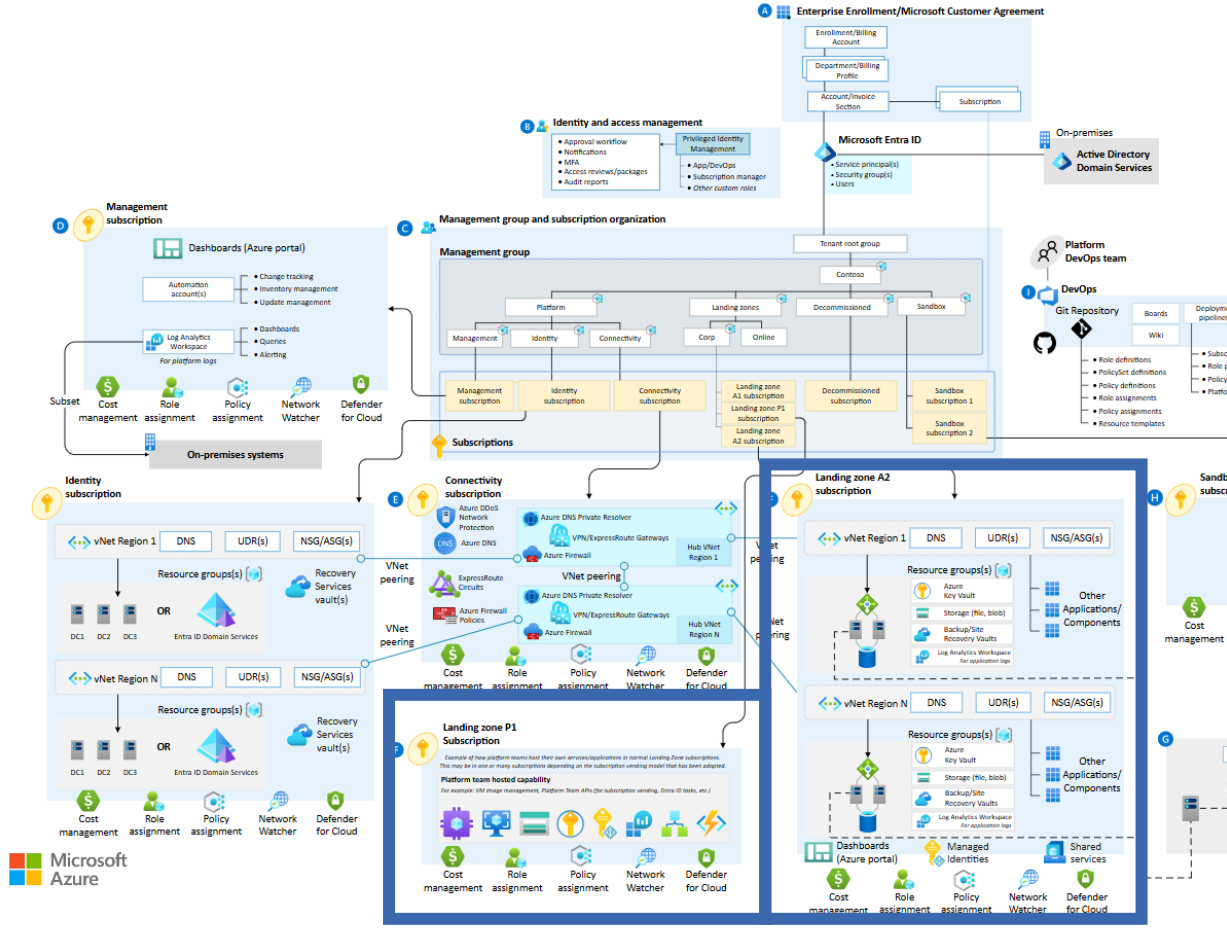
technicalSUMMIT

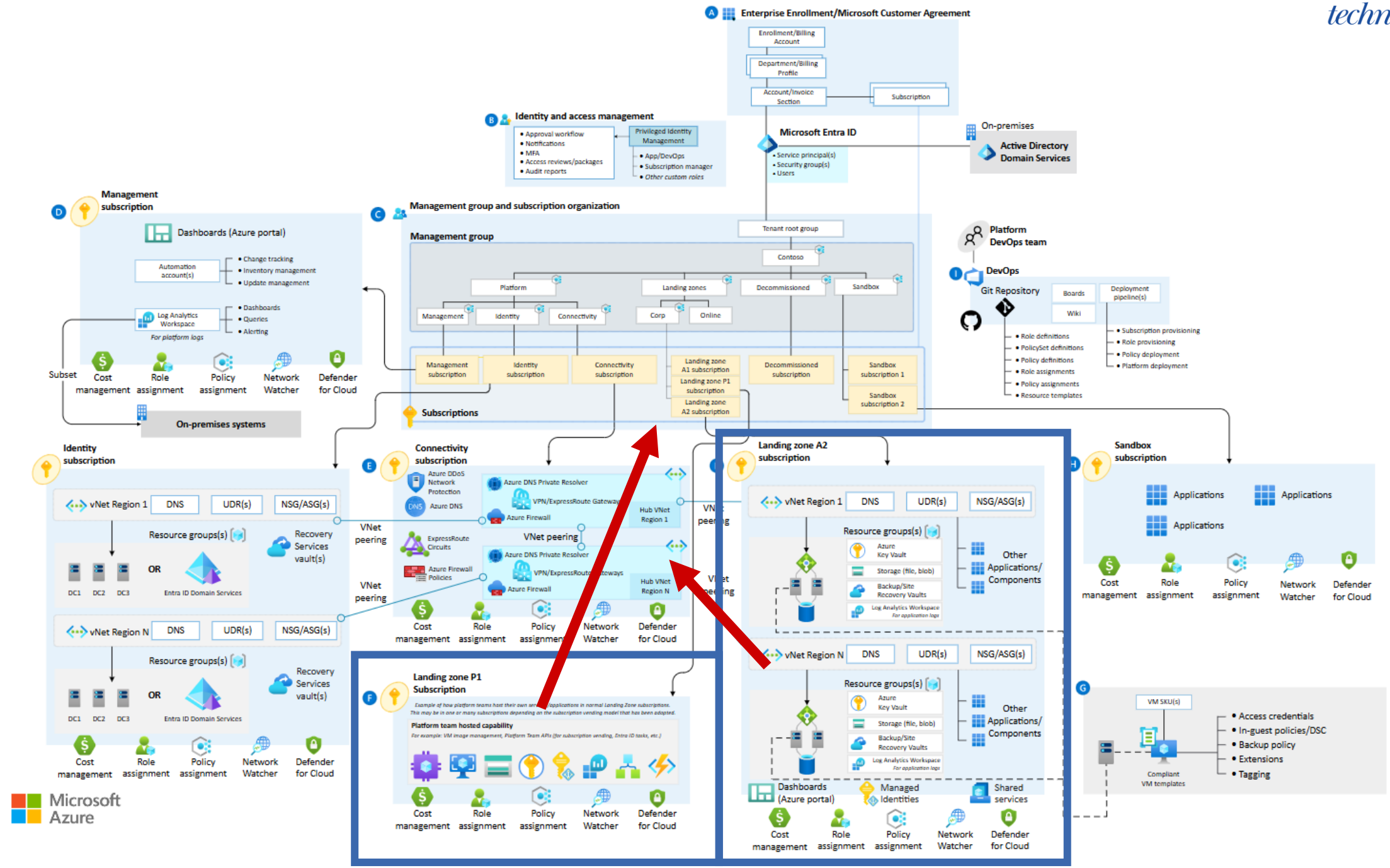
Application Landing Zones

Application Landing Zone

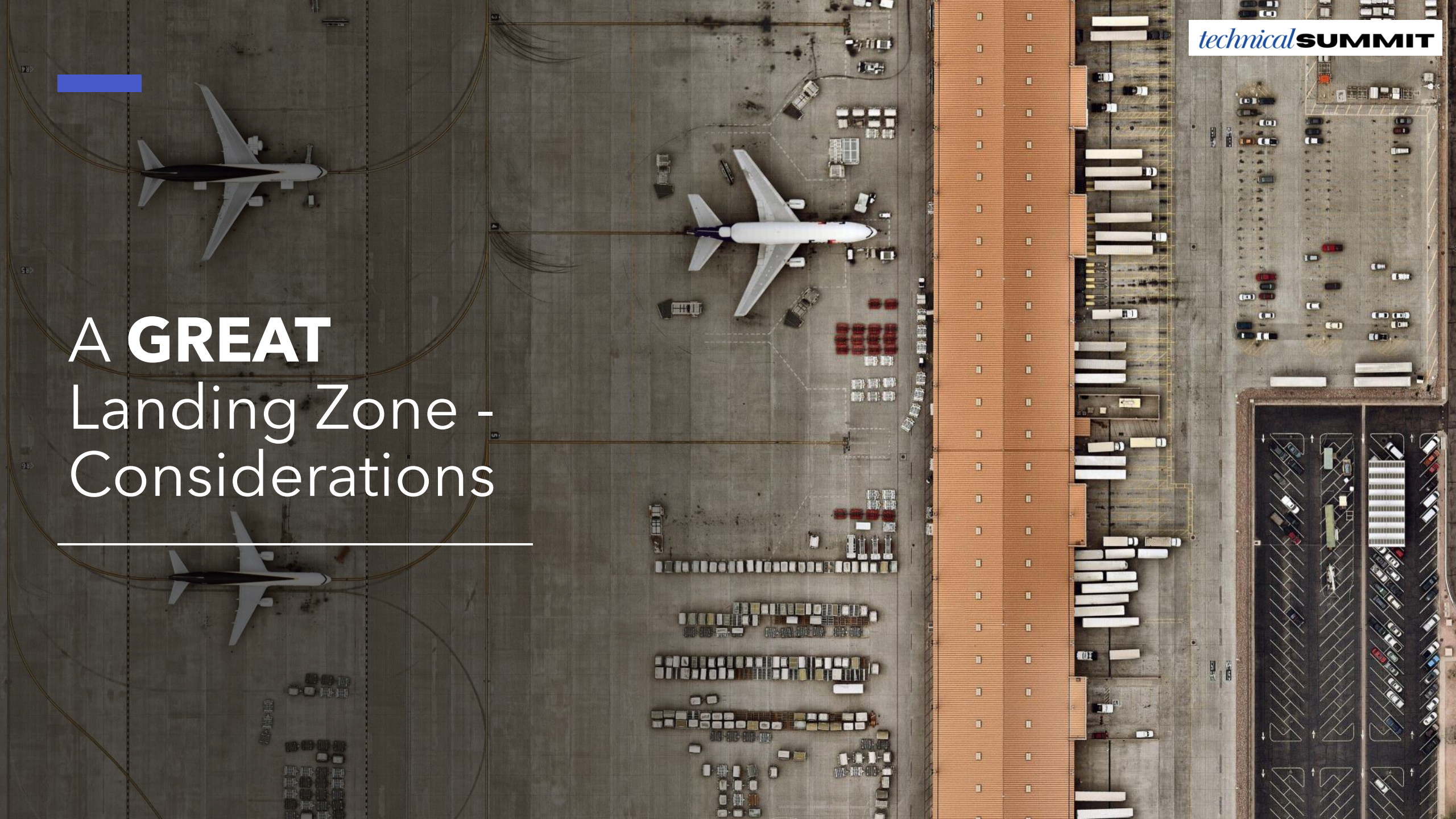
- A Landing Zone for a **specific Application**.
- Usually, Pre-Provisioned through **code**.
- **Consumes** services from Platform Landing Zones.

Think of these as the "buildings/offices/facilities" in the City.





Version: 2024-08-16



technicalSUMMIT

A **GREAT** Landing Zone - Considerations

technicalSUMMIT

Journey to **GREAT:**

Guidance is available!

---

*And it is **AWESOME!***

Use the Accelerators!

Azure landing zone accelerators

Accelerators are infrastructure-as-code implementations that help you deploy an Azure landing zone correctly. We have a platform landing zone accelerator and several application landing zone accelerators you can deploy.

Platform landing zone accelerator

There's a ready-made deployment experience called the **Azure landing zone portal accelerator**. The Azure landing zone portal accelerator deploys the conceptual architecture (*see figure 1*) and applies predetermined configurations to key components such as management groups and policies. It suits organizations whose conceptual architecture aligns with the planned operating model and resource structure.

You should use the Azure landing zone portal accelerator if you plan to manage your environment with the Azure portal. If you want to use Bicep or Terraform, see the Bicep and Terraform deployment options. Deploying the Azure landing zone portal accelerator requires permissions to create resources at the tenant (`/`) scope. Follow the guidance in Tenant deployments with ARM templates: Required access to grant these permissions.

Deploy to Azure

Application landing zone accelerators

Application landing zone accelerators help you deploy application landing zones. Use the list of available application landing zone accelerators in the Azure Architecture Center and deploy the accelerator that matches your scenario.

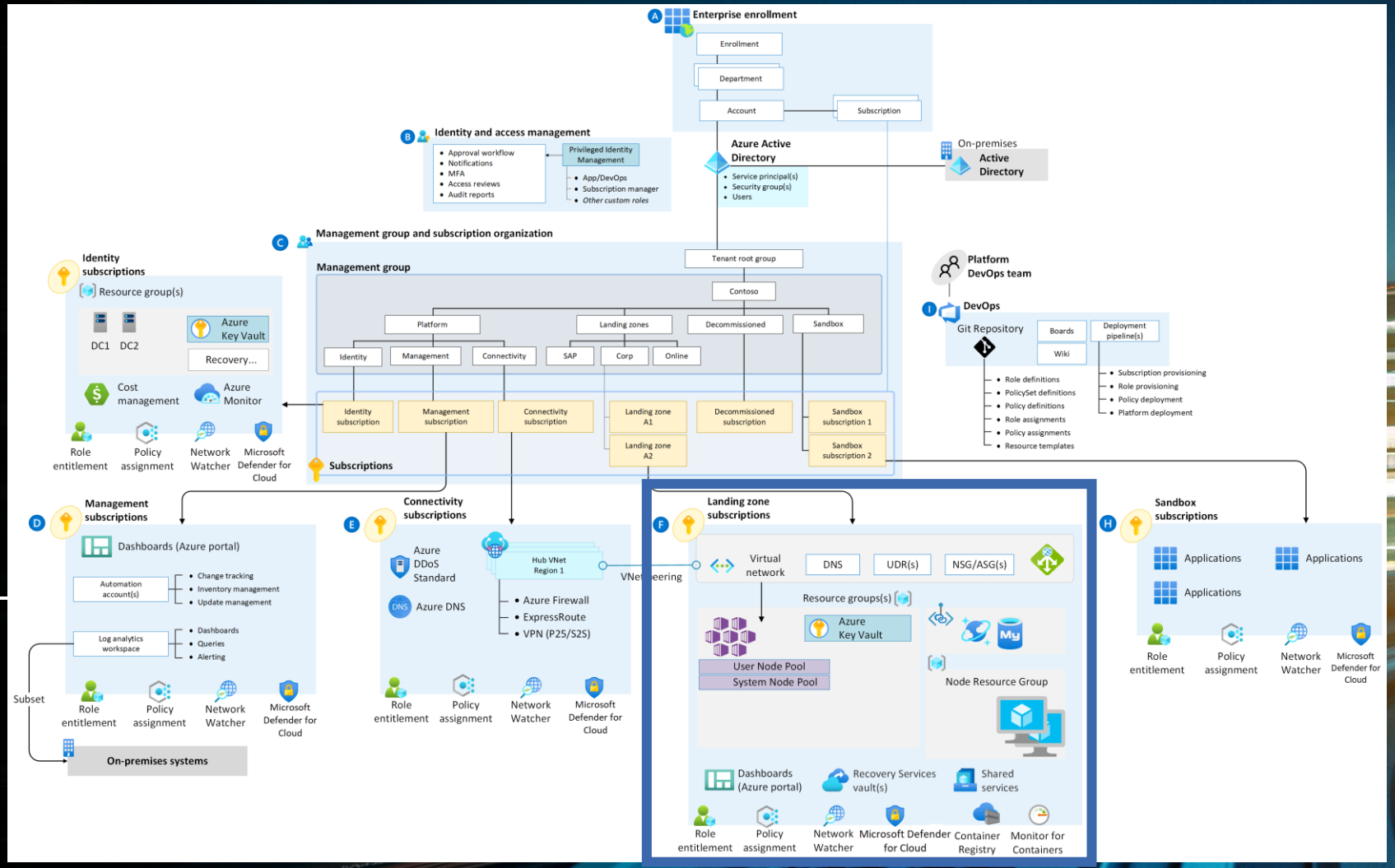
Application Landing Zone Accelerators

Foundations for many applications and deployments – already done, and ready to be used!

Application	Description
AKS landing zone accelerator	An open-source collection of Azure Resource Manager (ARM), Bicep, and Terraform templates that represent the strategic design path and target technical state for an AKS deployment.
Azure OpenAI chat baseline architecture in an Azure landing zone	Outlines how to integrate an Azure OpenAI chat application within Azure landing zones to utilize centralized shared resources while adhering to governance and cost efficiency, offering guidance for workload teams on deployment and management.
Azure App Service landing zone accelerator	Proven recommendations and considerations across both multitenant and App Service environment use cases with a reference implementation for ASEv3-based deployment.
Azure API Management landing zone accelerator	Proven recommendations and considerations for deploying APIM management with a reference implementation showcasing Azure Application Gateway with an internal APIM instance-backed Azure Functions as back end.
SAP on Azure landing zone accelerator	Terraform and Ansible templates that accelerate SAP workload deployments by using Azure landing zone best practices, including the creation of infrastructure components like compute, networking, storage, monitoring, and build of SAP systems.
HPC landing zone accelerator	An end-to-end HPC cluster solution in Azure that uses tools like Terraform, Ansible, and Packer. It addresses Azure landing zone best practices, including implementing identity, jumpbox access, and autoscale.
Azure VMware Solution landing zone accelerator	ARM, Bicep, and Terraform templates that accelerate VMware deployments, including Azure VMware Solution private cloud, jumpbox, networking, monitoring, and add-ons.
Azure Virtual Desktop landing zone accelerator	ARM, Bicep, and Terraform templates that accelerate Azure Virtual Desktop deployments, including creation of host pools, networking, storage, monitoring, and add-ons.
Azure Red Hat OpenShift landing zone accelerator	An open-source collection of Terraform templates that represent an optimal Azure Red Hat OpenShift deployment that includes Azure and Red Hat resources.
Azure Arc landing zone accelerator for hybrid and multicloud	Azure Arc-enabled servers, Kubernetes, and Azure Arc-enabled SQL Managed Instance. See the Jumpstart ArcBox overview.
Azure Spring Apps landing zone accelerator	Azure Spring Apps landing zone accelerator is intended for an application team that builds and deploys Spring Boot applications in a typical landing enterprise zone design. As the workload owner, use architectural guidance provided in this accelerator to achieve your target technical state with confidence.
Enterprise-scale landing zone for Citrix on Azure	Design guidelines for the Cloud Adoption Framework for Citrix Cloud in an Azure enterprise-scale landing zone cover for many design areas.
Azure Container Apps Landing Zone Accelerator	This Azure Container Apps landing zone accelerator outlines the strategic design path and defines the target technical state for deploying Azure Container Apps. It is owned and operated by a dedicated workload team.

AKS Example

Foundations for many applications and deployments – ready made and easy to deploy.



Cloud Readiness **Antipatterns**

Cloud readiness antipatterns

Article • 03/22/2023 • 9 contributors

Feedback

In this article

Customers often experience antipatterns during the readiness phase of cloud adoption. These antipatterns can lead to unexpected downtime, disaster recovery problems, and availability issues.

Antipattern: Assume released services are ready for production

Because cloud computing is evolving rapidly, companies often release preview versions of new services. Customers tend to assume that they can use any available cloud service in a production environment. But, problems can result, for these reasons:

- Preview services usually don't provide uptime service-level agreements (SLAs).
- New services often aren't as mature as cloud services that are already available.

Example: Use a preview service in production

A research institute uses a preview cloud service in production. The service seems to be a good fit for its use case. But, the institute doesn't perform due diligence on the service. The institute also doesn't follow its reference architecture's requirements and guidelines.

Problems come up with the preview service that lead to unexpected downtime. The institute begins to think that cloud services in general aren't as mature or resilient as promised.

Patterns

Cloud Design Patterns

Article • 04/13/2023 • 27 contributors

Feedback

In this article

These design patterns are useful for building reliable, scalable, secure applications in the cloud.

Each pattern describes the problem that the pattern addresses, considerations for applying the pattern, and an example based on Microsoft Azure. Most patterns include code samples or snippets that show how to implement the pattern on Azure. However, most patterns are relevant to any distributed system, whether hosted on Azure or other cloud platforms.

Cloud workloads are prone to the fallacies of distributed computing. Some examples of cloud design fallacies are:

- The network is reliable
- Latency is zero
- Bandwidth is infinite
- The network is secure
- Topology doesn't change
- There is one administrator
- Component versioning is simple
- Observability implementation can be delayed

Key Skills

Skills readiness path during the readiness phase of a migration journey

Article • 12/01/2022 • 12 contributors

Feedback

In this article

During the readiness phase of a migration journey, the objective is to prepare for the journey ahead. This phase is accomplished in two primary areas: organizational readiness and environmental (technical) readiness. Each area might require new skills for both technical and nontechnical contributors. The following sections describe a few options to help build the necessary skills.

Organizational readiness learning paths

Depending on the motivations and business outcomes associated with a cloud adoption effort, leaders might be required to establish new organizational structures or virtual teams to facilitate various functions. The following articles help to develop the skills that are necessary to structure those teams in accordance with desired outcomes:

The Importance of Policy

What is Azure Policy?

Article • 04/17/2024 • 19 contributors

Feedback

In this article

Azure Policy helps to enforce organizational standards and to assess compliance at-scale. Through its compliance dashboard, it provides an aggregated view to evaluate the overall state of the environment, with the ability to drill down to the per-resource, per-policy granularity. It also helps to bring your resources to compliance through bulk remediation for existing resources and automatic remediation for new resources.

> **Note**
>
> For more information on remediation, see Remediate non-compliant resources with Azure Policy.

Common use cases for Azure Policy include implementing governance for resource consistency, regulatory compliance, security, cost, and management. Policy definitions for these common use cases are already available in your Azure environment as built-ins to help you get started.

Specifically, some useful governance actions you can enforce with Azure Policy include:

- Ensuring your team deploys Azure resources only to allowed regions
- Enforcing the consistent application of taxonomic tags
- Requiring resources to send diagnostic logs to a Log Analytics workspace

technicalSUMMIT

Infrastructure as Code

Azure Bicep & Terraform

Azure landing zones - Terraform module design considerations

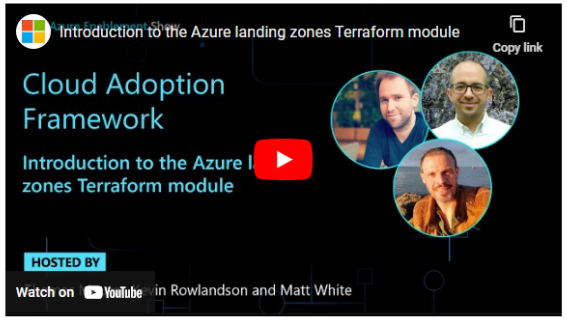
Article • 03/30/2023 • 7 contributors

Feedback

In this article

This article discusses important areas to consider when using the Azure landing zones Terraform module. The module provides an opinionated approach to deploy and operate an Azure platform based on the Azure landing zone conceptual architecture as detailed in the Cloud Adoption Framework (CAF).

Terraform is an open-source Infrastructure as Code (IaC) tool, created by HashiCorp, that uses declarative syntax to deploy infrastructure resources. It is extensible, has cross-platform support and enables immutable infrastructure through state tracking.



Azure landing zones - Bicep modules design considerations

Article • 11/07/2023 • 6 contributors

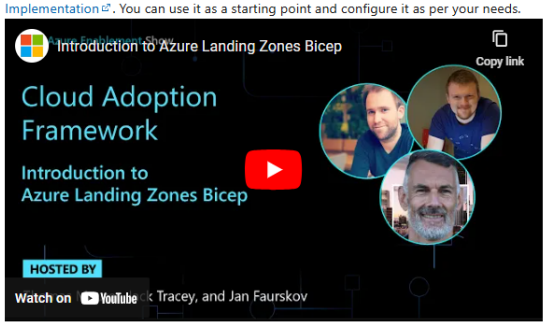
Feedback

In this article

This article discusses the design considerations of the modularized Azure Landing Zones (ALZ) - Bicep solution you can use to deploy and manage the core platform capabilities of the Azure landing zone conceptual architecture as detailed in the Cloud Adoption Framework (CAF).

Bicep is a domain-specific language (DSL) that uses declarative syntax to deploy Azure resources. It has concise syntax, reliable type safety, and support for code reuse.

An implementation of this architecture is available on GitHub: Azure Landing Zones (ALZ) - Bicep Implementation. You can use it as a starting point and configure it as per your needs.



Subscription Vending

Rapid deployment for application team Subscriptions

Design common product lines for subscription vending

Now that you understand that platform teams must provide multiple Azure subscription types and styles, or product lines, to consumers of their Azure platform, this section describes several common product lines that you can use across industries and countries or regions.

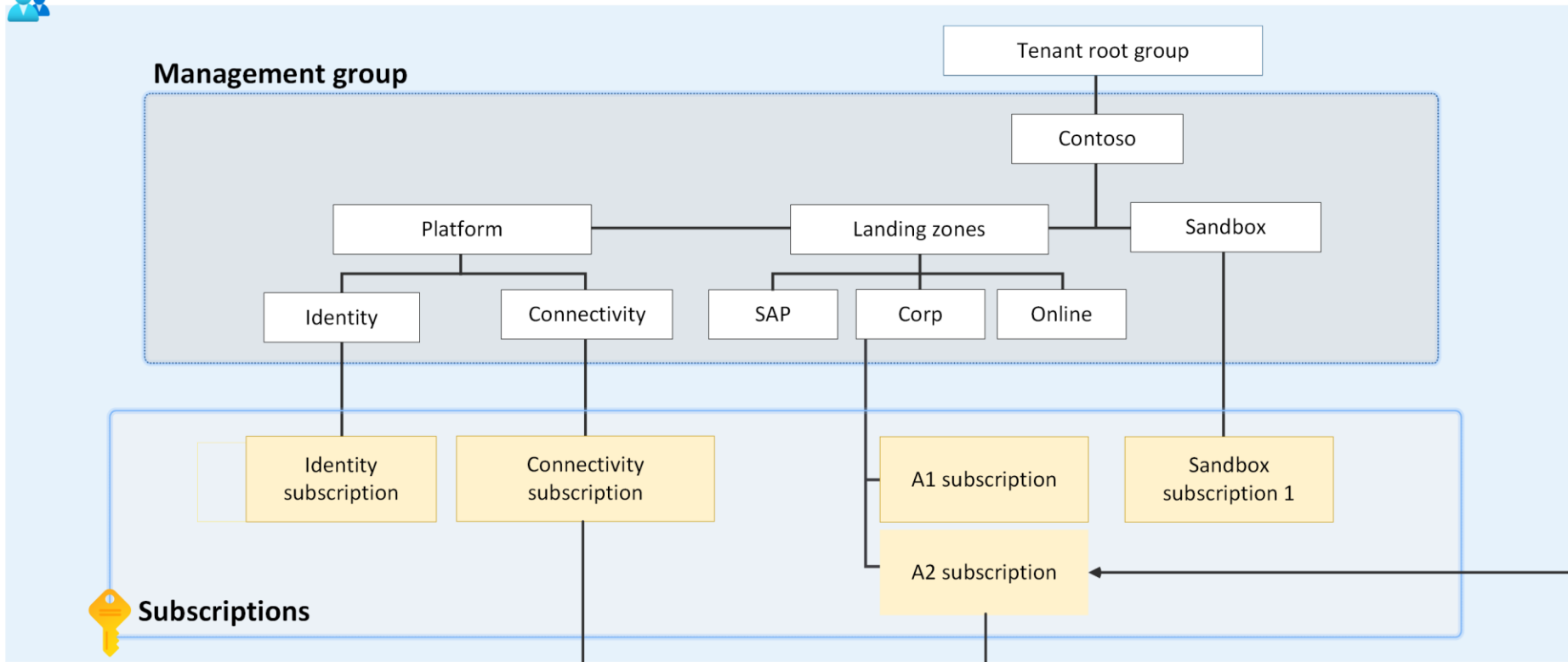
Your platform team should use these common subscription vending product lines as a baseline. Your team can provide multiple options to its consumers out of the box, which aligns with the *prioritize customers* platform engineering principle. This approach gives internal customers the freedom to use Azure landing zone design principles and design area recommendations to deliver their workloads and service and also provides Azure platform governance.

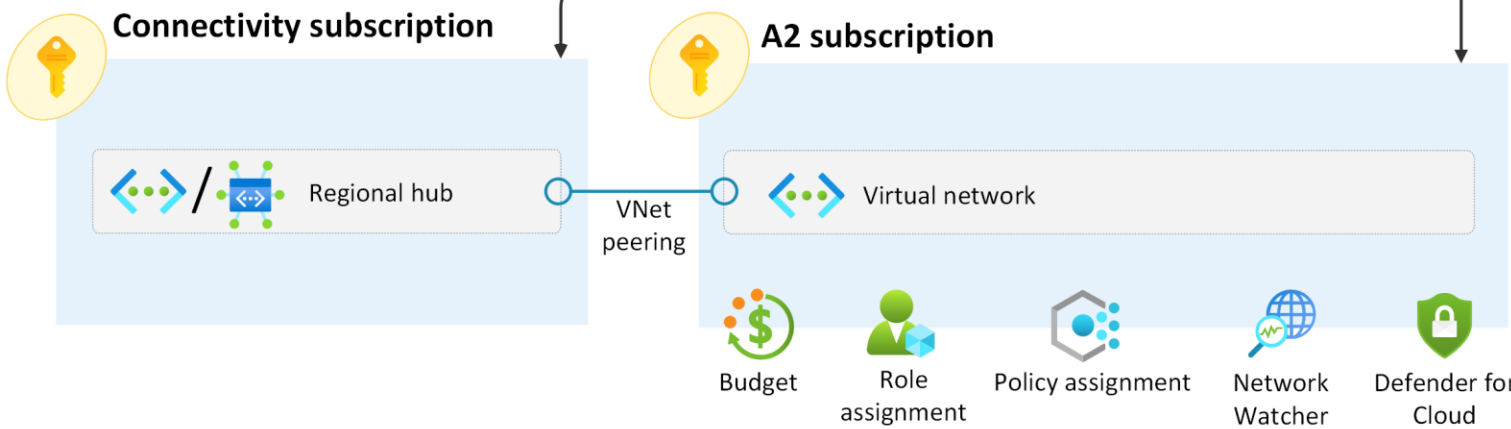
> ⓘ Note
>
> Use these examples as a starting point. You can customize and expand these product lines to cater to the needs of your organization.

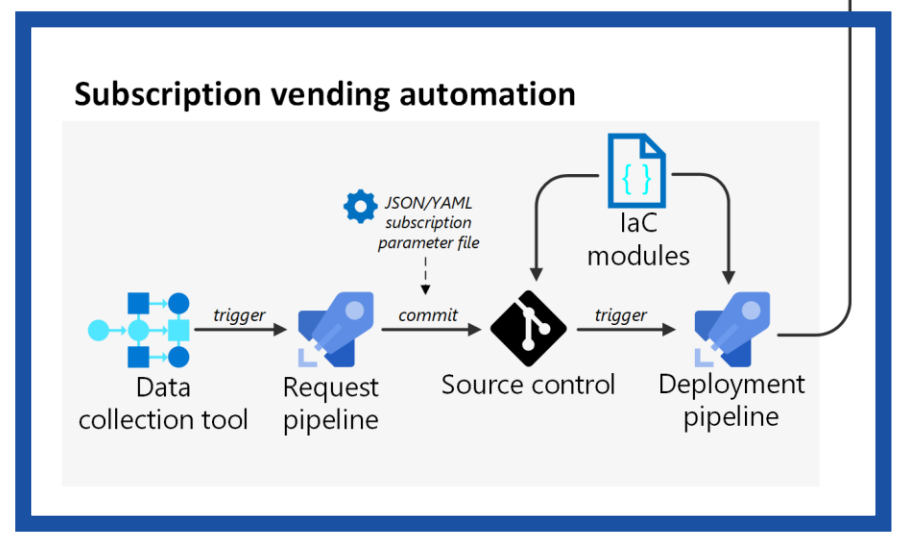
Common product lines for subscription vending include:

- Corp connected: Workloads that require traditional Layer-3 IP routing connectivity to other applications and on-premises environments via the Connectivity subscription.
- Online: Workloads that connect with other applications through modern connectivity services and architectures, such as Azure Private Link or interaction via exposed APIs or endpoints from each application.
- Tech platform: Workloads that build a platform on which you can build other applications. For example, an Azure Kubernetes Service (AKS) fleet of clusters that an AKS platform team manages can host other applications within its AKS clusters on behalf of other application teams.
- Shared application portfolio: Shared workloads among the same application teams for a common set of closely coupled applications. You don't want to host the applications on their own or with any specific workload.
- Sandbox: An area where application teams can build a proof of concept (PoC) or minimum viable product (MVP) and impose fewer controls, so the team can promote development, invention, and freedom to build the best possible application from the catalog of available Azure services.

Management group and subscription organization

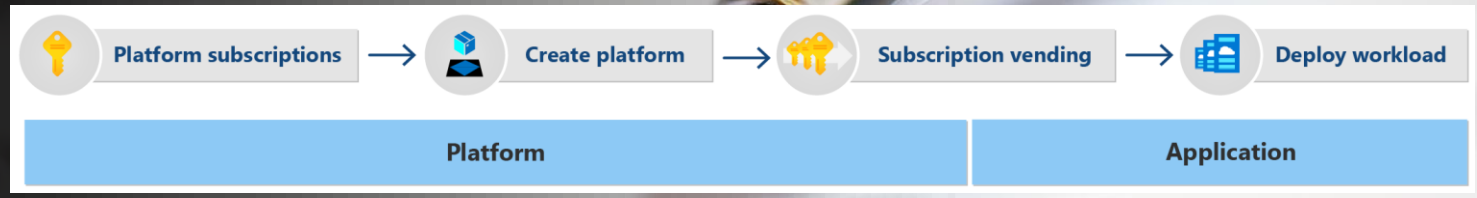




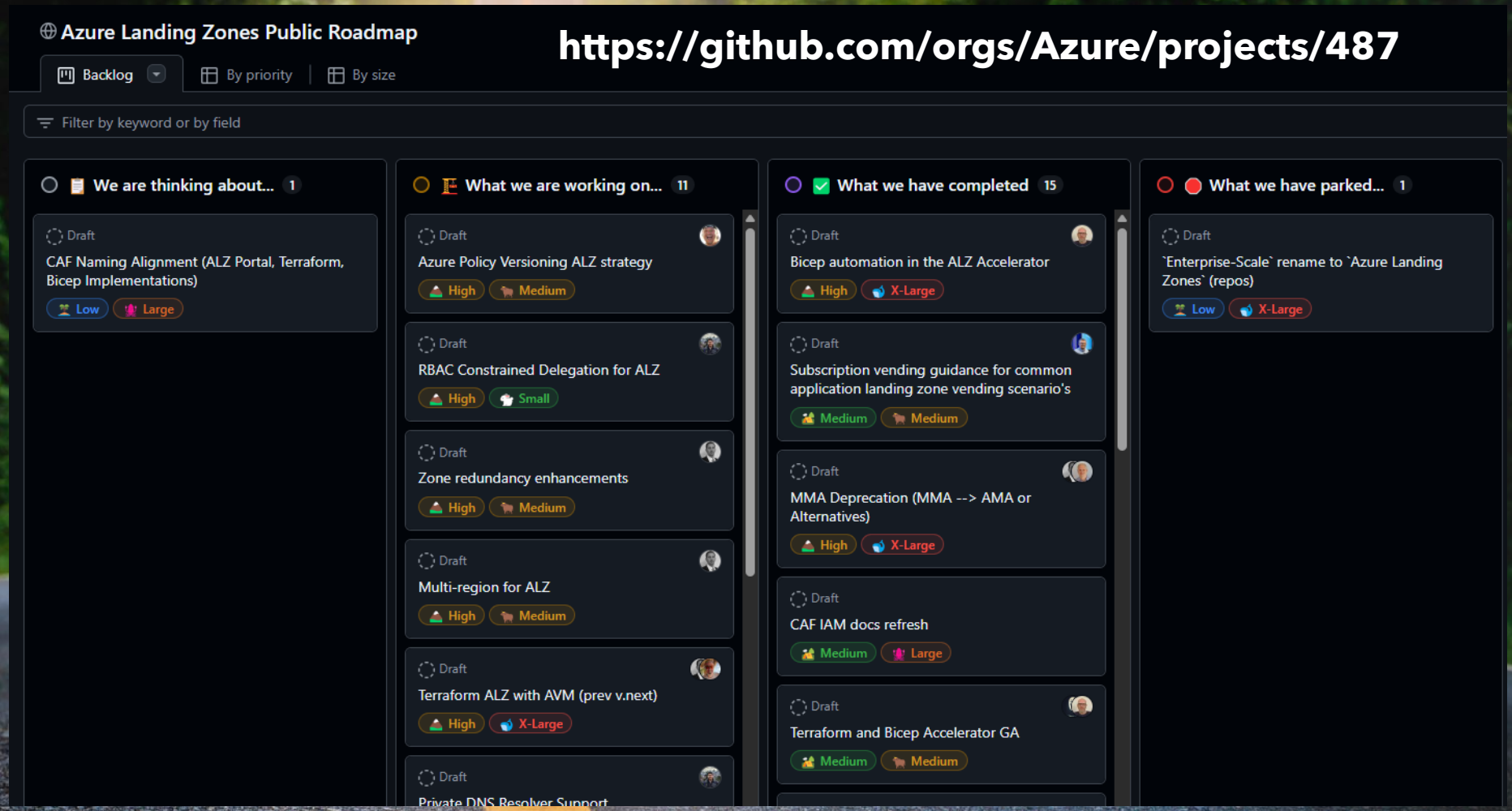


Subscription Vending

Rapid deployment for application team Subscriptions



Roadmap


Azure Landing Zones Public Roadmap
https://github.com/orgs/Azure/projects/487
Backlog
By priority
By size
Filter by keyword or by field
We are thinking about... 1
Draft
CAF Naming Alignment (ALZ Portal, Terraform, Bicep Implementations)
Low
Large
What we are working on... 11
Draft
Azure Policy Versioning ALZ strategy
High
Medium
Draft
RBAC Constrained Delegation for ALZ
High
Small
Draft
Zone redundancy enhancements
High
Medium
Draft
Multi-region for ALZ
High
Medium
Draft
Terraform ALZ with AVM (prev v.next)
High
X-Large
Draft
Private DNS Resolver Support
What we have completed 15
Draft
Bicep automation in the ALZ Accelerator
High
X-Large
Draft
Subscription vending guidance for common application landing zone vending scenario's
Medium
Medium
Draft
MMA Deprecation (MMA --> AMA or Alternatives)
High
X-Large
Draft
CAF IAM docs refresh
Medium
Large
Draft
Terraform and Bicep Accelerator GA
Medium
Medium
What we have parked... 1
Draft
`Enterprise-Scale` rename to `Azure Landing Zones` (repos)
Low
X-Large

Summary

A few closing thoughts…

- Before the process – evaluate, consider, and plan. Note – the plan may change, so flexibility is key.
- Guidance is key – use the resources, materials, and code available to help.
- Democratize Cloud – A Great Landing Zone architecture and implementation should free up IT Teams.
- Ensure appropriate levels of Dev/Test access to enable Cloud Adoption at speed.

Getting the right Azure Foundation - What makes a **great** Landing Zone?

Jake Walsh

jakewalsh.co.uk

@jakewalsh90

technicalSUMMIT

MVP Microsoft® Most Valuable Professional

HashiCorp AMBASSADOR 2024

Microsoft CERTIFIED TRAINER MCT

CTA Citrix Technology Advocate