Microsoft

Azure Landing Zones

27th April 2023 - External Community Call

Registration:
https://aka.ms/ALZ/CommunityCallRegister

Agenda (please add suggestions):
https://aka.ms/ALZ/CommunityCallAgenda

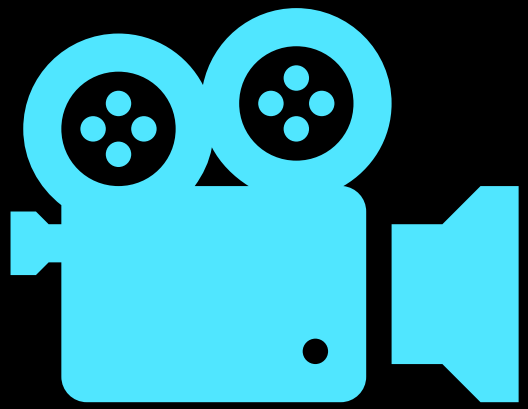
Microsoft

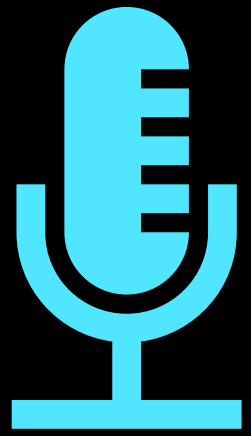
Azure Landing Zones

27th April 2023 - External Community Call

This month's presenters:

Microsoft





This meeting is being recorded

Agenda

- New Docs Published
- Multi-Tenant Guidance Released
- Policy Refresh Launched
- Subscription Vending Updates
- Azure Monitor Updates – Baseline, AMA & Diagnostic Settings
- ALZ TF Updates
- ALZ Bicep Updates
- CARML Updates
- Microsoft Build
- Q & A

Before we get started...

At any point, if you have a question please put it in the chat!

(we have members of the team here to help 🤓)

Also we may stop and discuss your question/point at that time, we want this to be an open discussion with all of you ☺

ALZ What's New?

https://aka.ms/ALZ/WhatsNew

Single place to stay up-to-date

Updates

Here's what's changed in Enterprise Scale/Azure Landing Zones:

April 2023

We are pleased to announce that we are starting regular Azure Policy reviews for Azure Landing Zone. This includes a review of new built-in policies released and their suitability for ALZ, built-in policies that can replace custom ALZ policies, built-in policies that have been deprecated and addition of new ALZ custom policies and initiatives as identified based on best practices, issues raised and customer feedback. Most importantly, we have also provided default assignments for all the new policies at the appropriate ALZ Management Group level. This will ensure that all new policies are automatically assigned to the appropriate scope and will be in compliance with the ALZ baseline. This will also ensure that the ALZ is always up to date with the latest Azure Policy definitions.

This update includes many ALZ Azure Policies and Initiatives that have been added or updated to enhance the security, governance, and management of ALZ. As part of our commitment to continuous improvement, we have also enhanced our policy review process, with a focus on transitioning away from deprecated policies where possible, move from custom to built-in policies providing the same or enhanced functionality, and implementing new policies to keep ALZ as part of the current review cycle.

This is the first major review and refresh of Azure Policy since ALZ was GA'd. Since GA many new built-in policies and initiatives have been released which has driven the need for this review. We believe that a regular review cycle will allow us to stay on top of emerging trends and new policies, ensuring that our Azure environment remains secure and compliant. Should you identify policies or initiatives that should be considered for ALZ, kindly submit an GitHub issue. For more information, please refer to the ALZ Policies or the new Excel spreadsheet version.

We strongly advise staying up-to-date to ensure the best possible security posture for your Azure environment, see Keep your Azure landing zone up to date. For those with existing deployments or policies, we have provided Brownfield guidance to help you navigate the process of updating to the latest policies. We recognize that there may be breaking changes when upgrading an existing deployment or policies and for details follow our recently released guidance to support you in this process:

- Update Azure landing zone custom policies
- Migrate Azure landing zone policies to Azure built-in policies

> **Please note** that, in some cases, moving to the new Built-In Policy definitions, deploying changes to existing custom policies or removing deprecated policies will require a new Policy Assignment and removing the previous Policy Assignment, which will mean compliance history for the Policy Assignment will be lost. However, if you have configured your Activity Logs and Security Center to export to a Log Analytics Workspace, Policy Assignment historic data will be stored here as per the retention duration configured. Thank you for your cooperation, and we look forward to continuing to work with you to ensure the security and compliance of our Azure environment.

> While we've made every effort to test the stability of this release, should you have any issues and the guidance provided does not resolve your issue, please open a GitHub issue so we can do our best to support you and document the fix for

New AAC Content

- Azure Architecture Center
- Browse all Architectures
- Architecture icons
- What's new
- Landing zones
  - **Deployment Options**
  - Design guides
    - Landing zone implementations
      - Bicep landing zone implementation
      - Terraform landing zone implementation
    - Subscription vending implementation

Cloud operating model roles and responsibilities

The Cloud Adoption Framework describes four common cloud operating models. The Azure identity and access for landing zones recommends five role definitions (Roles) you should consider if your organizations cloud operating model requires customized Role Based Access Control (RBAC). If your organization has more decentralized operations, the Azure built-in roles may be sufficient.

The table below outlines the key roles for each of the cloud operating models.

Role	Decentralized operations	Centralized operations	Enterprise operations	Distributed operations
Azure platform owner (such as the built-in Owner role)	Workload team	Central cloud strategy	Enterprise architect in CCoE	Based on portfolio analysis - see Business alignment and Business commitments
Network management (NetOps)	Workload team	Central IT	Central Networking in CCoE	Central Networking for each distributed team + CCoE
Security operations (SecOps)	Workload team	Security operations center (SOC)	CCoE + SOC	Mixed - see: Define a security strategy
Subscription owner	Workload team	Central IT	Central IT + Application Owners	CCoE + Application Owners
Application owners (DevOps, AppOps)	Workload team	Workload team	Central IT + Application Owners	CCoE + Application Owners

Platform

The options below provide an opinionated approach to deploy and operate the Azure landing zone conceptual architecture as detailed in the Cloud Adoption Framework (CAF). It's important to note that, depending upon customizations, the resulting architecture might not be the same for all the options listed below. The differences between the options are how you deploy the architecture. They use differing technologies, take different approaches and are customized differently.

Deployment option	Description
Azure landing zone Portal accelerator	An Azure portal-based deployment that provides a full implementation of the conceptual architecture, along with opinionated configurations for key components such as management groups and policies.
Azure landing zone Terraform accelerator	This accelerator provides an orchestrator module, but also allows you to deploy each capability individually or in part.
Azure landing zone Bicep accelerator	A modular accelerator where each module encapsulates a core capability of the Azure landing zone conceptual architecture. While the modules can be deployed individually, the design proposes the use of orchestrator modules to encapsulate the complexity of deploying different topologies with the modules.

In addition, after deploying the landing zone, you will need to plan to operate it and maintain it. Review the guidance on how to Keep your Azure landing zone up to date.

Subscription Vending

Once the platform landing zone is in place, the next step is to create and operationalize application landing zones for workload owners. Subscription democratization is a design principle of Azure landing zones that uses subscriptions as units of management and scale. This approach accelerates application migrations and new application development.

Subscription vending standardizes the process for requesting, deploying, and governing subscriptions, enabling application teams to deploy their workloads faster. To get started, see subscription vending implementation guidance, then review the following infrastructure-as-code modules. They provide flexibility to fit your implementation needs.

Deployment option	Description
Bicep Subscription Vending	The Subscription Vending Bicep module is designed to accelerate deployment of the individual landing zones (aka Subscriptions) within an Azure Active Directory Tenant on EA, MCA & MPA billing accounts.
Terraform Subscription Vending	The Subscription Vending Terraform module is designed to accelerate deployment of the individual landing zones (aka Subscriptions) within an Azure Active Directory Tenant on EA, MCA & MPA billing accounts

Application

Application landing zones are one or more subscriptions that are deployed as environments for workloads or applications. These workloads can take advantage of services deployed in platform landing zones. The application landing zones can be centrally managed applications, decentralized workloads, or technology platforms such as Azure Kubernetes Service that host applications.

You can use the options below to deploy and manage applications or workloads in an application landing zone.

Application	Description
AKS landing zone accelerator	An open-source collection of ARM, Bicep, and Terraform templates that represent the strategic design path and target technical state for an Azure Kubernetes Service (AKS) deployment.
Azure App Service landing zone accelerator	Proven recommendations and considerations across both multi-tenant 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 App Gateway with internal APIM instance backed Azure Functions as backend.
SAP on Azure landing zone accelerator	Terraform and Ansible templates that accelerate SAP workload deployments using Azure Landing Zone best practices, including the creation of Infrastructure components like Compute, Networking, Storage, Monitoring & build of SAP systems.
HPC landing zone accelerator	An end-to-end HPC cluster solution in Azure using tools like Terraform, Ansible, and Packer. It addresses Azure Landing Zone best practices, including implementing identity, Jump-box access, and autoscale.
Azure VMware Solution landing zone accelerator	ARM, Bicep, and Terraform templates that accelerate VMware deployments, including AVS 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 (ARO) deployment that is comprised of both Azure and Red Hat resources.
Azure Arc landing zone accelerator for hybrid and multicloud	Arc enabled Servers, Kubernetes, and Arc-enabled SQL Managed Instance see the Jumpstart ArcBox overview.

aka.ms/ALZ/AAC

New CAF Doc

[aka.ms/ALZ/IaC](aka.ms/ALZ/IaC)

Use infrastructure as code to update Azure landing zones

Article • 03/31/2023 • 3 contributors

Feedback

This article describes the benefits of using infrastructure as code (IaC) to update Azure landing zones. Organizations need to update their landing zones as they operate to ensure that configurations are correct and they respond to the need for changes.

IaC can manage the whole life cycle, and it excels at managing the resources that it deploys. Organizations should plan to deploy their Azure landing zones with IaC. It requires planning to align existing non-IaC resources with IaC resources that are backed with state management. You need to map the existing resources to the desired state.

For more information, see Keep your Azure landing zone up to date.

How infrastructure as code works

IaC refers to the practice and tools for managing the lifecycle of infrastructure resources by using machine-readable definition files. The definition for the infrastructure is written, versioned, deployed through pipelines, and then it becomes a part of the deployment for workloads.

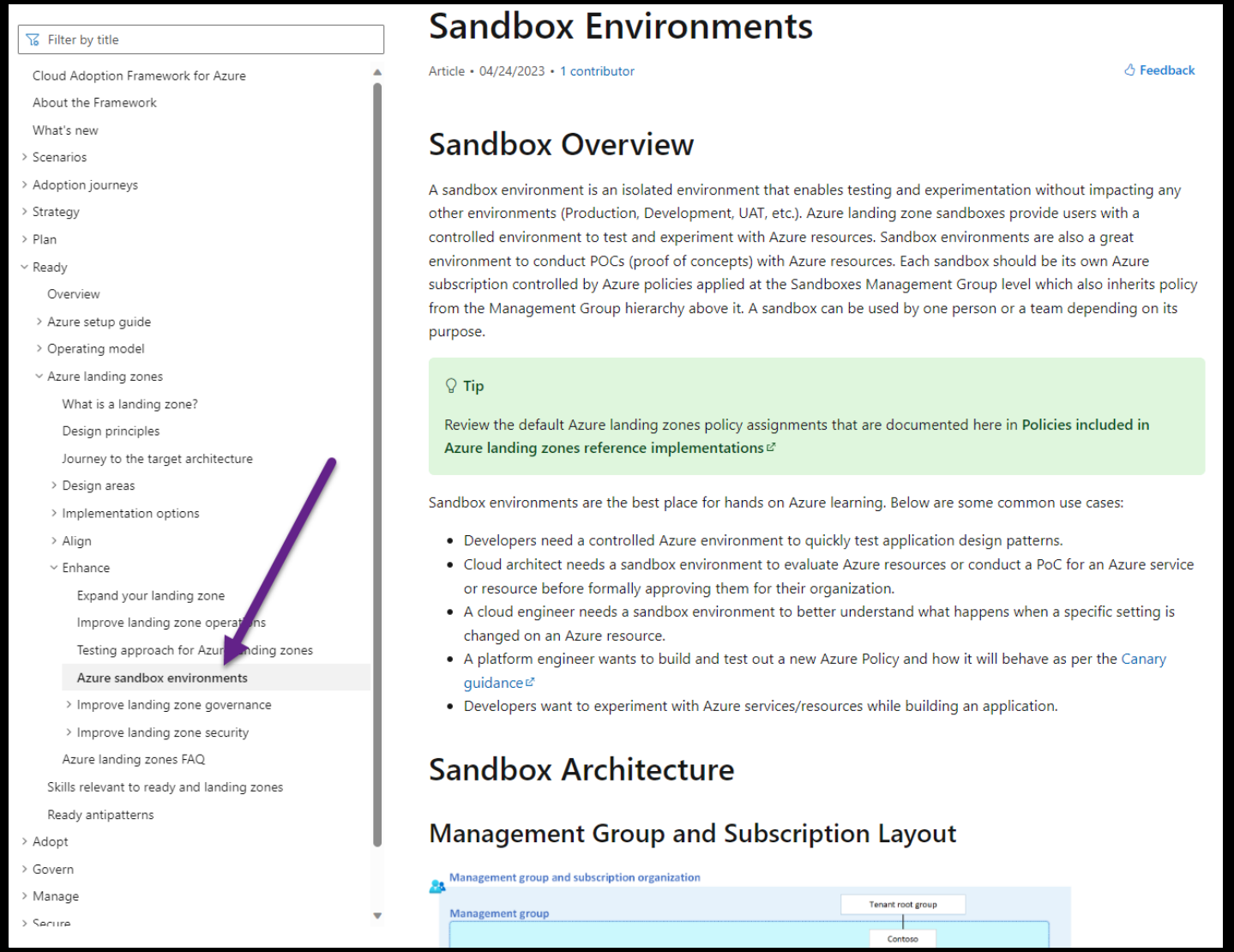
IaC technologies are *declarative*, which means when IaC runs, it sets the configuration to what's described in the code, regardless of its current state. When you configure infrastructure through scripts, such as the Azure CLI or Azure PowerShell, they're *imperative*. Imperative scripts perform a set of actions, and the result depends on the current state plus the state after the actions.

So, if you have an infrastructure as code definition for an Azure resource, you can run that definition as often as you want, and it only creates a change if:

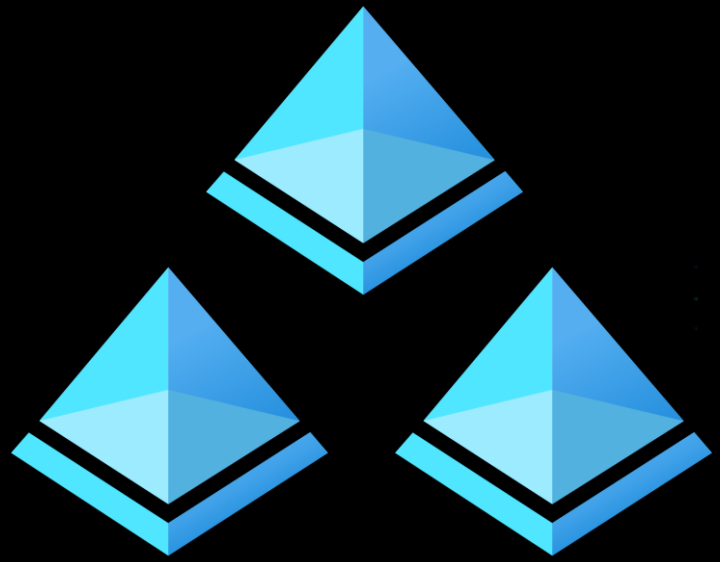
- The definition changes to add new resources, remove resources previously deployed, or modifies resources that were previously deployed.
- The deployed resource drifts from the configuration to reset the configuration to the defined one.

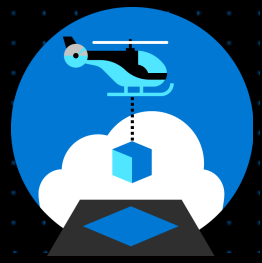
New CAF Doc SANDBOX

Next Month



ALZ Multi-Tenant Updates



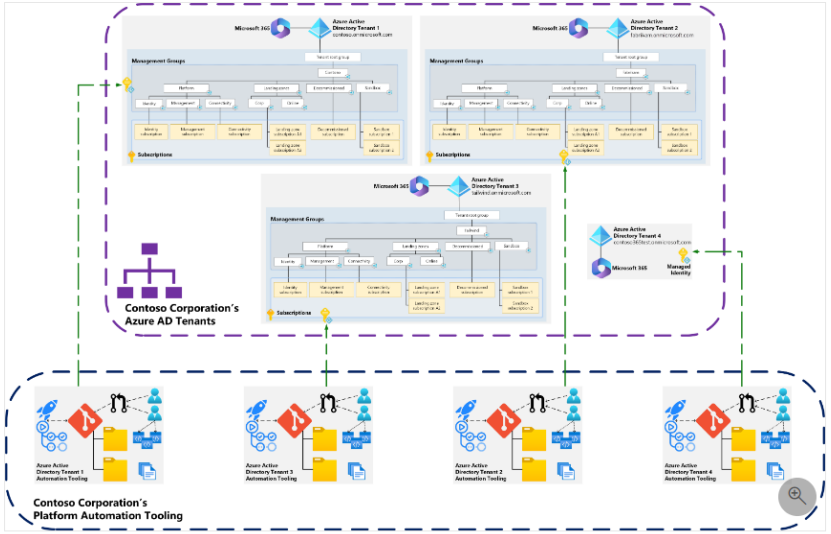


New CAF docs are live

Approach 1 – Complete isolation

In this approach, the primary objective is to keep each Azure Active Directory tenant isolated from each other across all automation components, like:

- A Git repository.
- GitHub Actions or Azure Pipelines (including self-hosted runners, if being utilized).
- Identities that are used for performing tasks from automation, like managed identities assigned to self-hosted runners, service principal names (SPNs), users, or administrators.



aka.ms/ALZ/MultiTenant

Scenarios for multiple Azure Active Directory tenants

Article • 02/28/2023 • 7 minutes to read • 1 contributor

Feedback

There are a few reasons why an organization might need, or might want to investigate, multiple Azure Active Directory tenants. The most common scenarios are:

- Mergers and acquisitions
- Regulatory or country/region compliance requirements
- Business unit or organizational isolation and autonomy requirements
- Independent software vendor (ISV) delivering SaaS applications from Azure
- Tenant level testing / Microsoft 365 testing
- Grassroots / Shadow IT / start-ups

Important

Multiple Azure Active Directory tenants **are not the recommended** approach for most customers. A single Azure Active Directory tenant, typically the corporate Azure Active Directory tenant, is recommended for most customers because it provides the necessary separation requirements.

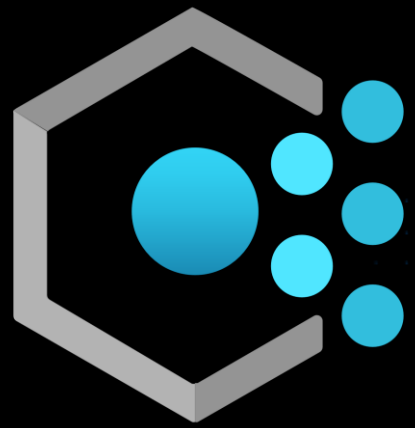
For more information, see:

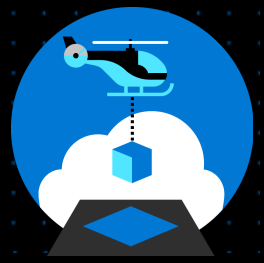
- Define Azure Active Directory tenants
- Architecture: Standardize on a single directory and identity
- Testing approach for Azure landing zones
- Introduction to delegated administration and isolated environments
- Resource isolation in a single tenant
- Your Microsoft 365 for enterprise tenants

Next steps

Scenarios for multiple Azure Active Directory tenants

ALZ Policy Refresh Update





ALZ Policy Refresh

Objective

- Review current policy/initiative assignments and validate relevance
- New best practice guardrails for core services (e.g., Key Vault) and specialized landing zones (Decommissioned/Sandbox)
- "Remove" or replace deprecated policies/initiatives
- Remove default workload specific assignments no longer in scope for ALZ (e.g., Databricks, Machine Learning)
- Update assignments to leverage built-in policies (instead of custom ALZ policies) where possible

Update

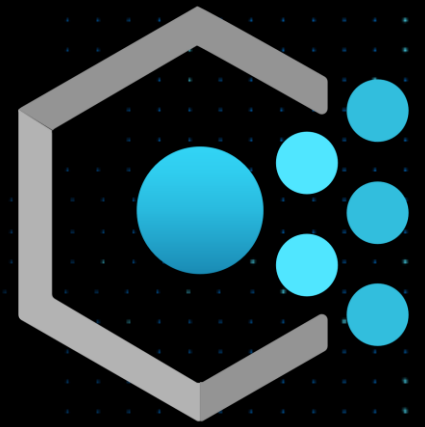
- Code complete - preparing for merge into ‘main’ ALZ branch
- Summary of changes:
  - Added 13 net new policy and initiative default assignments, which includes:
    - Added 46 new policies, 8 new initiatives
    - Updated 11 initiatives
    - Retired 4 policies
    - Portal Accelerator RI updated

ALZ Policy Refresh

Ask

- Submit Azure Policy suggestions on GitHub
  - These must be policies that all customers across all industries/countries/verticals would benefit from

Next Steps



- Next review cycle of policy/initiative assignments for ALZ
- Update Terraform and Bicep RIs to support new policies/initiatives

Going Forward

- Quarterly reviews of policy/initiative assignments and regular release of policy/initiative and assignment updates

Relevant Links

- Official Release: aka.ms/alz/whatsnew and aka.ms/alz/policies

Subscription Vending Updates

Blog

Announcing our new Subscription Vending IaC Modules for Bicep & Terraform

Subscribe

By Matt White

Published Mar 20 2023 09:38 AM

5,772 Views

Listen

We're excited to announce the release of the subscription vending IaC Modules! These modules are designed to help you streamline and automate the process of provisioning Azure subscriptions, making it easier than ever to get started with Azure.

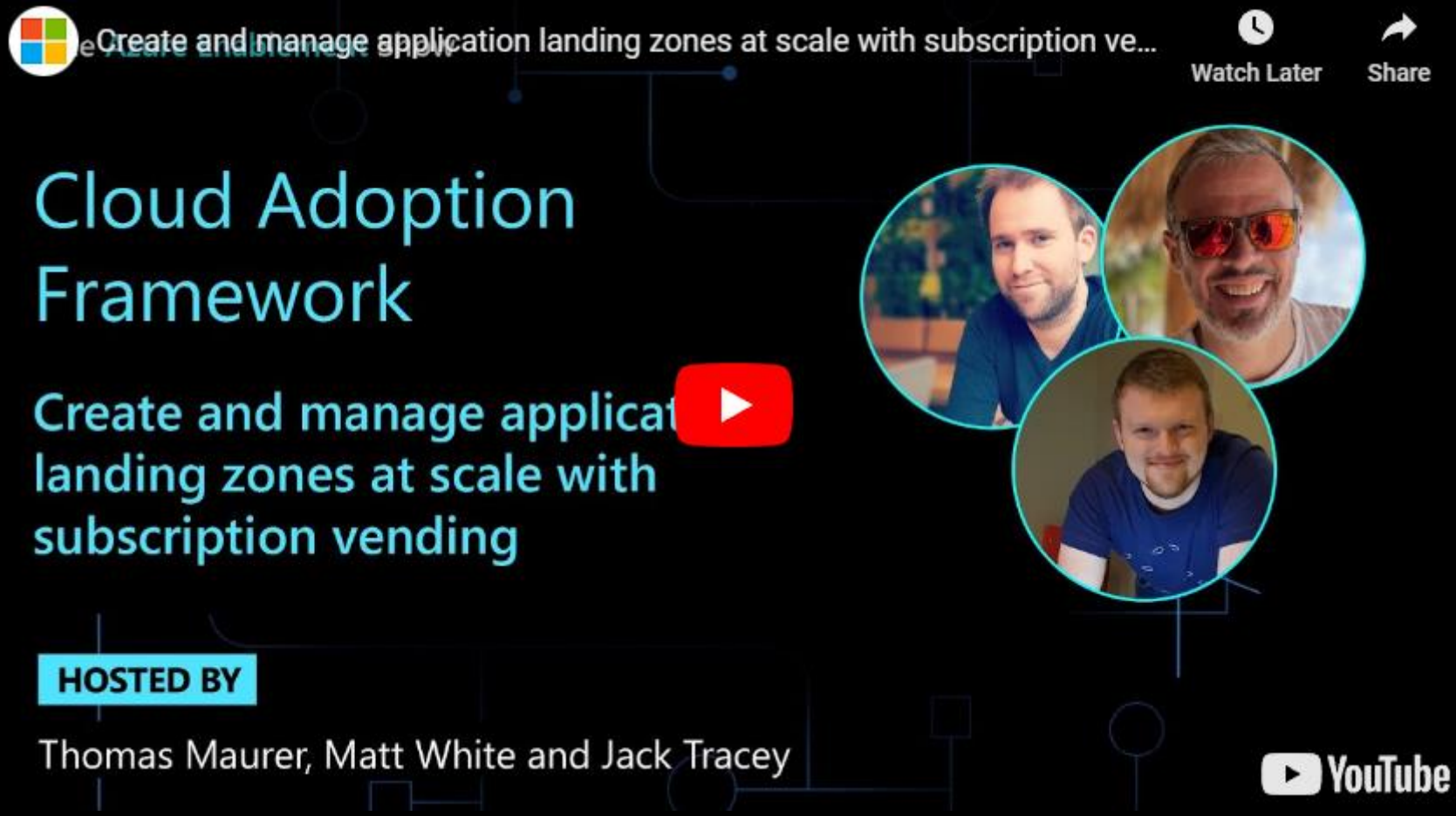
We heard loud and clear feedback from our customers & partners that we did a great job helping them accelerate their Cloud Platforms with Azure Landing Zones. And that we also helped accelerate the deployment of complex workloads like Azure Virtual Desktop, Azure VMware Solution, and more via our Application Landing Zones Accelerators. But, we didn't help customers easily achieve the design principle of Subscription Democratization. Until now!

The Subscription Vending IaC Modules are available for use with two popular infrastructure-as-code (IaC) tools: Bicep and Terraform. These modules have been created by the Customer Architecture & Engineering (CAE) team within Microsoft's Global Customer Success (GCS) organization (the same team behind Azure Landing Zones) and are designed to help you implement the best practices for subscription provisioning.

Using these modules, you can quickly and easily provision new Azure subscriptions that are pre-configured to meet your organization's specific needs and help your application teams get access to the Azure subscriptions they need, faster. The modules include parameters/variables for Role-Based Access Control, Networking, Tags and much more.

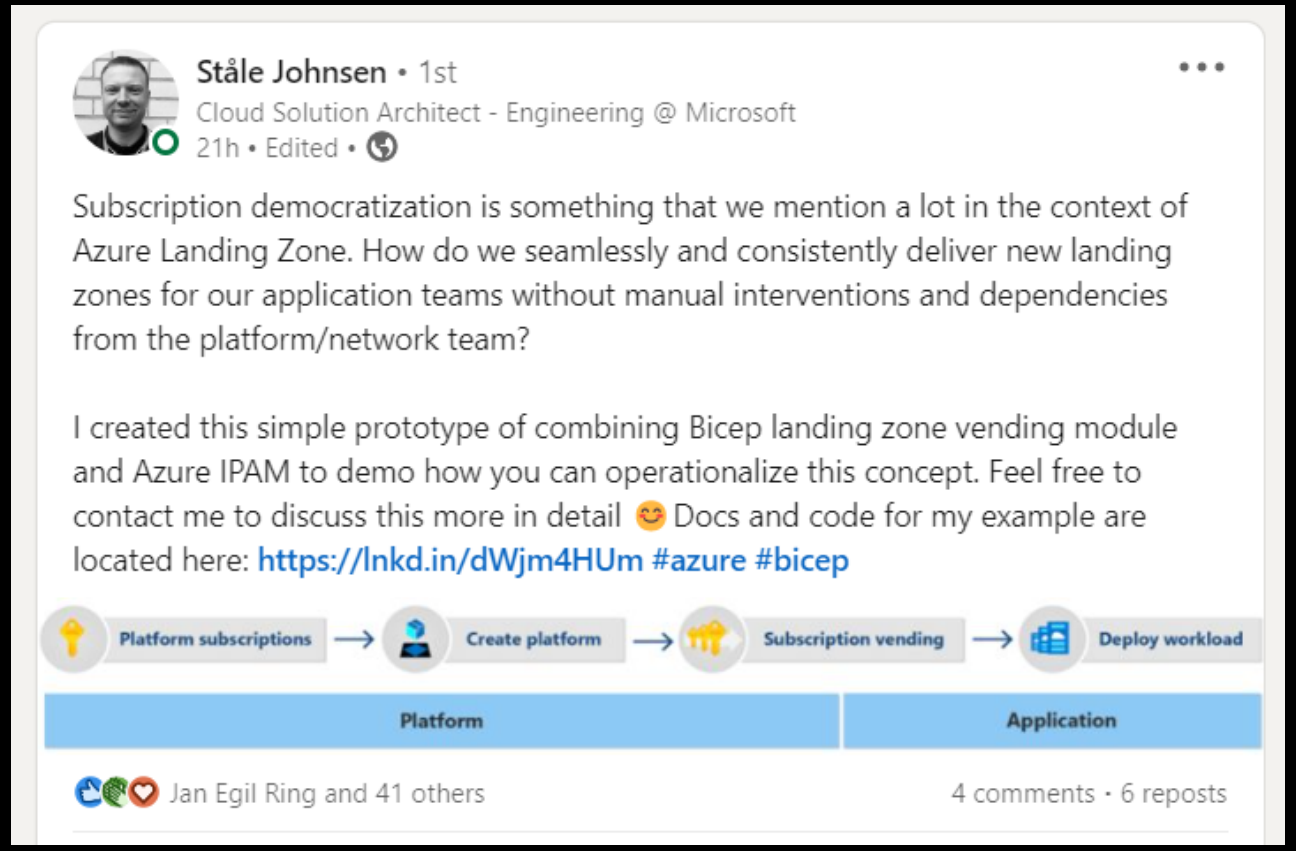
aka.ms/sub-vending/blog

New Azure Enablement Show Video



aka.ms/sub-vending/aes

Community Call Out! 📣 📣 📣



[Checkout Ståle Johnsen's Example Repo Here]

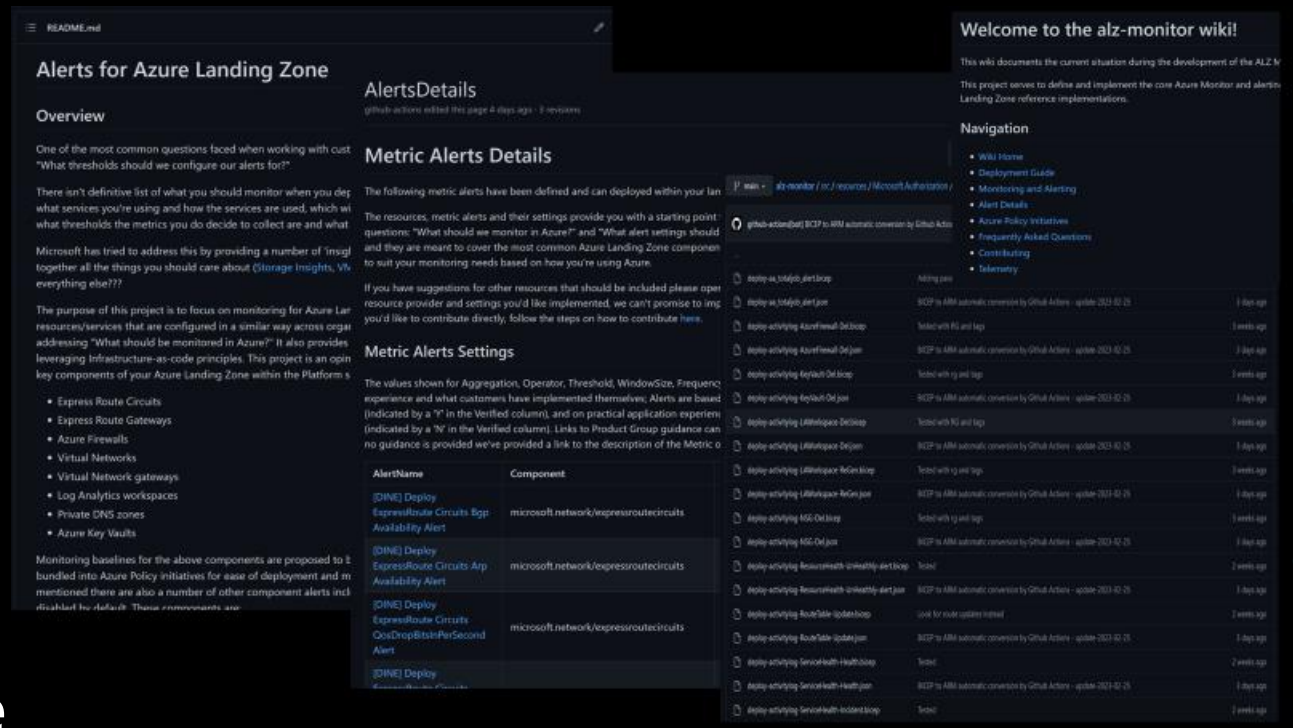
ALZ Azure Monitor Baseline

Azure Monitor Baseline Alerts

- **What is it and what does it contain?**

1. Detailed table of alerts / metrics and the recommended values working
2. Azure Policies and Policy Initiatives both in Bicep and json format
3. Extensible and flexible – chose what you deploy
4. Comprehensive deployment guidance and scripts for ALZ / (Brownfield) non-ALZ customers Sample GitHub Pipeline (ADO Pipeline on its way)

- **We're currently in preview! Please test and give your feedback (raise [issues](#))**



https://aka.ms/alz/monitor/repo

ALZ Migration to AMA Update

Added to FAQ

Why hasn't Azure landing zones migrated to the Azure Monitor Agent yet?

Great question! Don't worry we are aware of this required migration and change to Azure landing zones with the Log Analytics Agent (Microsoft Monitoring Agent - MMA) being retired in August 2024 as detailed here: Migrate to Azure Monitor Agent from Log Analytics agent.

We are working hard internally with the Azure Monitor Product Group (PG) to ensure everything that Azure landing zones requires and gets from the Log Analytics Agent (Microsoft Monitoring Agent - MMA) approach today is covered and has a path for migration to the Azure Monitor Agent (AMA) approach. This has been underway for sometime and continues to progress.

The AMA agent brings a number of new concepts, resources and changes to existing integrations with other services, such as Microsoft Defender for Cloud, that all require validation by each of the associated PGs as well as the Azure landing zone team, prior to migrating to AMA from MMA.

We will, when ready, provide Azure landing zones specific migration guidance that supports existing and to be created PG documentation. We will also make the relevant changes to each of the implementation options (Portal, Bicep, Terraform) to support the migration, especially for greenfield scenarios.

> We have an existing GitHub Issue (#1055) opened for this feature request. Please feel free to give it a 👍 or add a comment.

What if we are not ready to make the switch and migrate, right now?

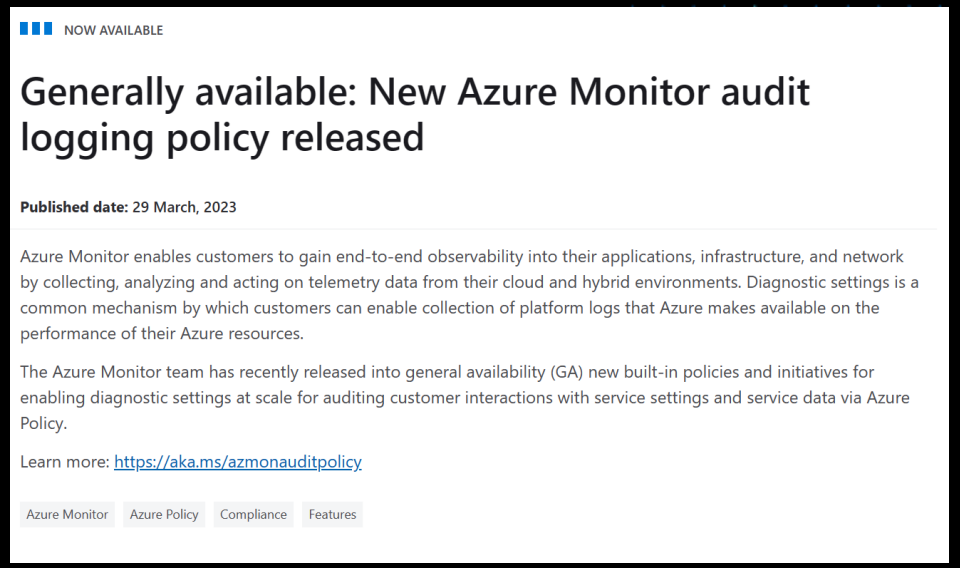
Another good question. You will need to plan, and complete, the migration to the Azure Monitor Agent before the Log Analytics Agent is retired as documented here.

Read in FAQ here

Azure Monitor Audit Diagnostic Policies

Built-in Policies have been released!

- Built-in Audit Diagnostic Policies [Generally available: New Azure Monitor audit logging policy released | Azure updates | Microsoft Azure]
- Contains policies for sending to:
  - Storage account
  - Log Analytics Workspace
  - Event Hub
- 33 Policies per initiative/destination!
- Not in ALZ yet!
- What are we doing about ALZ integration ☺?



NOW AVAILABLE

Generally available: New Azure Monitor audit logging policy released

Published date: 29 March, 2023

Azure Monitor enables customers to gain end-to-end observability into their applications, infrastructure, and network by collecting, analyzing and acting on telemetry data from their cloud and hybrid environments. Diagnostic settings is a common mechanism by which customers can enable collection of platform logs that Azure makes available on the performance of their Azure resources.

The Azure Monitor team has recently released into general availability (GA) new built-in policies and initiatives for enabling diagnostic settings at scale for auditing customer interactions with service settings and service data via Azure Policy.

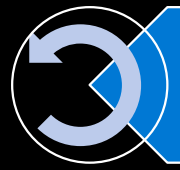
Learn more: https://aka.ms/azmonauditpolicy

Azure Monitor | Azure Policy | Compliance | Features

ALZ Terraform Updates

v4.0 of ALZ Terraform ETA May 2023

Azure Firewall Basic

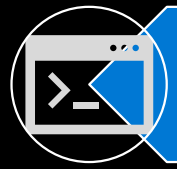


Policy Refresh

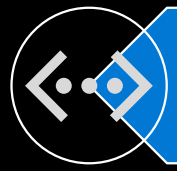
Enforcement mode

Current state

>1,200 unique customers



>7,000 deployments for core module



>2,900 deployments for connectivity

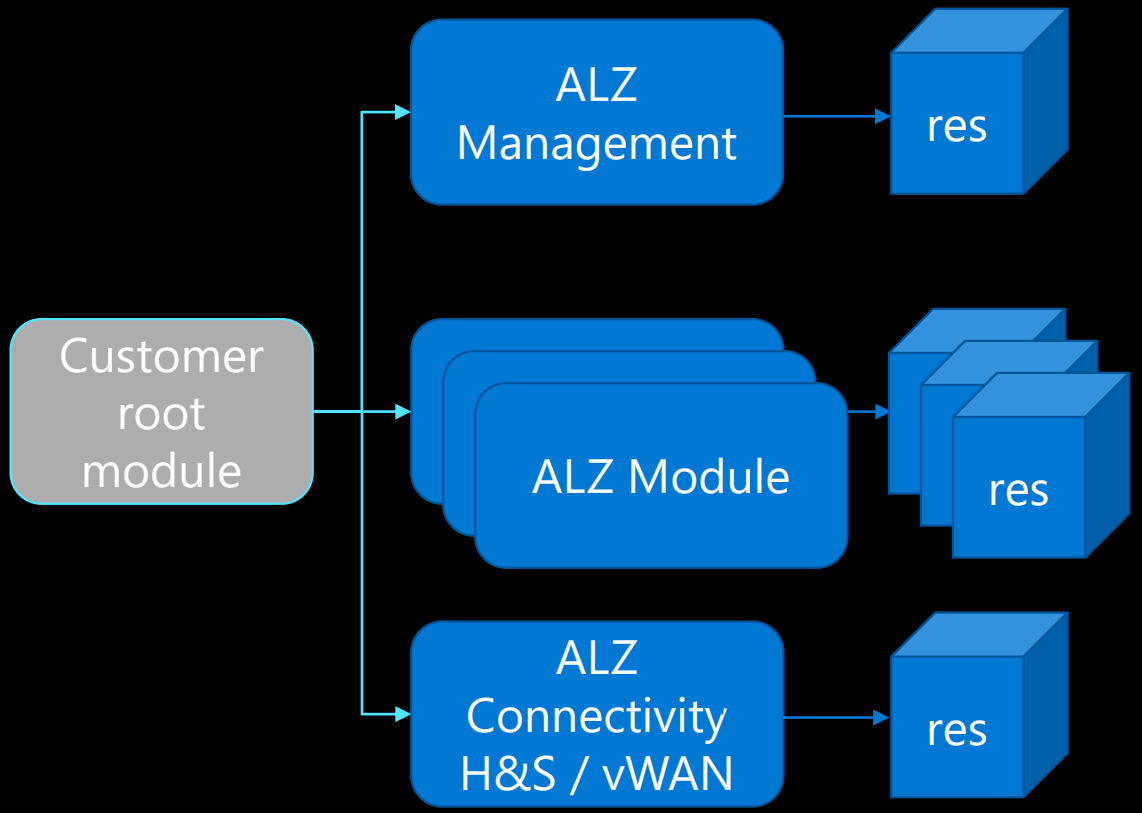
59% of customers only deploy core

Evidence

- GH issue comments - Request for feedback: future direction for this module · Issue #630 · Azure/terraform-azurerm-caf-enterprise-scale (github.com)
- Customer & partner interviews
  - **Nobody** said keep it as it is

asks

Proposed v.next architecture 🤓



Sub-modules that were previously part of the ALZ module, will become independent.

All to become TFVM

Other stuff...

- Provider
  - Written in go
  - Allows more advanced data processing
  - Can debug, set breakpoints
  - Testing is 1st class feature
  - Massively simplifies Terraform code
  - Still uses ‘standard’ Azure providers to deploy resources via a module
- Migration
  - SxS for MGs and Policy
  - State migrate for networking/mgmt resources
  - Support current module with policy updates

ALZ Bicep Updates

ALZ Bicep Updates

v0.14.0

ETA 1st May 2023

Policies

ALZ Bicep Accelerator

Beta release

Platform Management Group Flexibility

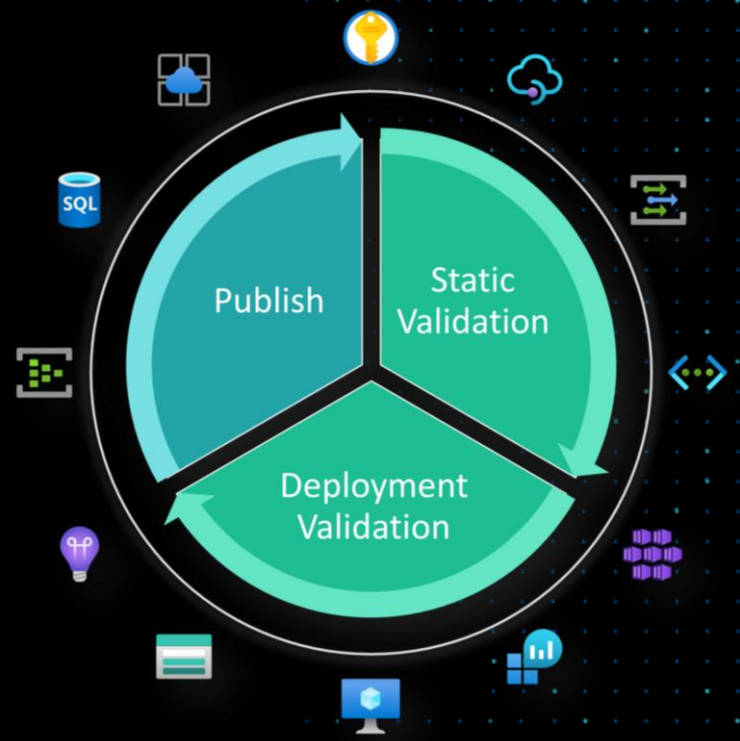
Private DNS Zone Linking Support in Hub Peered Spoke Module

- Latest from Upstream ALZ Repo Refresh
- New Policy Assignment Exclusion Parameter for ALZ Defaults
- Uses New PowerShell Module (available from PowerShell Gallery)
- Bootstrap your GitHub repo with actions/workflows to deploy ALZ
- Guidance on branching strategy and how to stay up-to-date with ALZ Bicep releases
- Will become recommended way to deploy and manage your ALZ Bicep deployment
- Same as available in Landing Zones Management Group

Microsoft

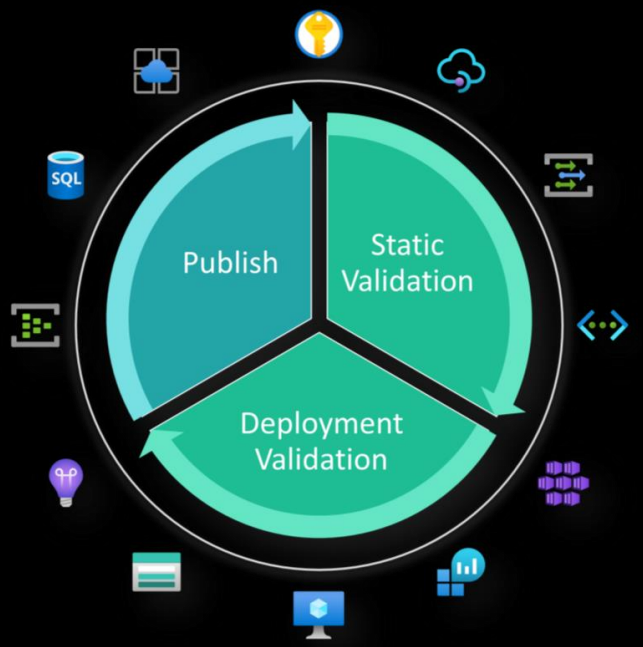
Common Azure Resource Modules Library (CARML)

Release v0.10.0



CARML

Release v0.10.0 highlights

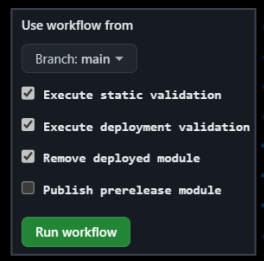


- 516 ★ *(+13%)*
- 351 forks *(+ 8%)*
- 200+ unique visitors daily
- 129 merged PRs
- 34 contributors *(6 new / 82 total)*

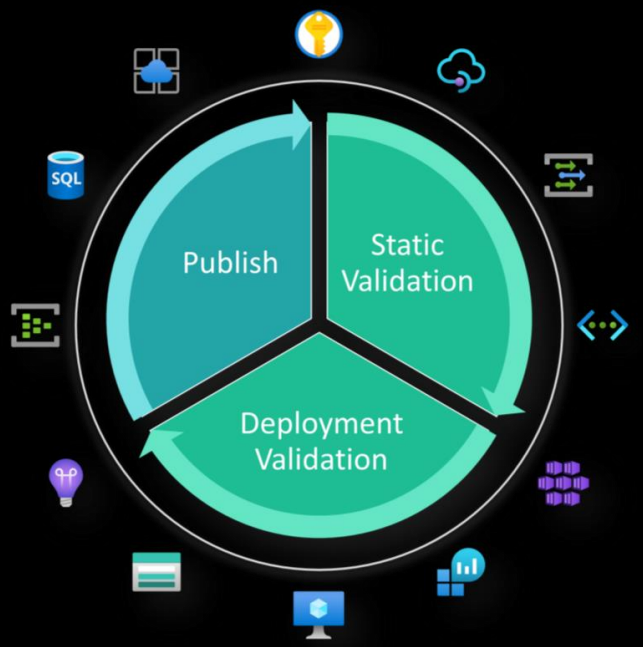
- **Modules**
  - 21 **new modules** including 7 top-level services such as
    - *Microsoft.Insights/dataCollectionRules*
    - *Microsoft.Purview/accounts*
    - *Microsoft.DBforMySQL/flexibleServers*

    alongside other features & new test cases
  - Addressed several **PS-Rule** checks
  - Optimized input parameters (e.g., *vaultSku* to *sku*)
- **CI Environment**
  - Implemented the GitHub **reusable workflows** feature to reduce code duplication & and make maintenance easier
  - Introduced **runtime parameters** to both GitHub and Azure DevOps

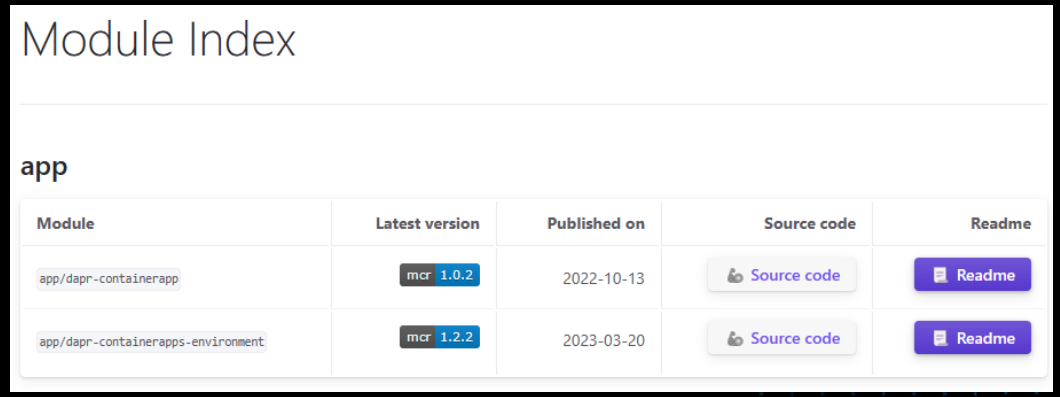
```
call-workflow-passing-data:
  name: 'Module'
  uses: ./.github/workflows/template.module.yml
  with:
    workflowInput: '${{ needs.job_initialize.pi
```



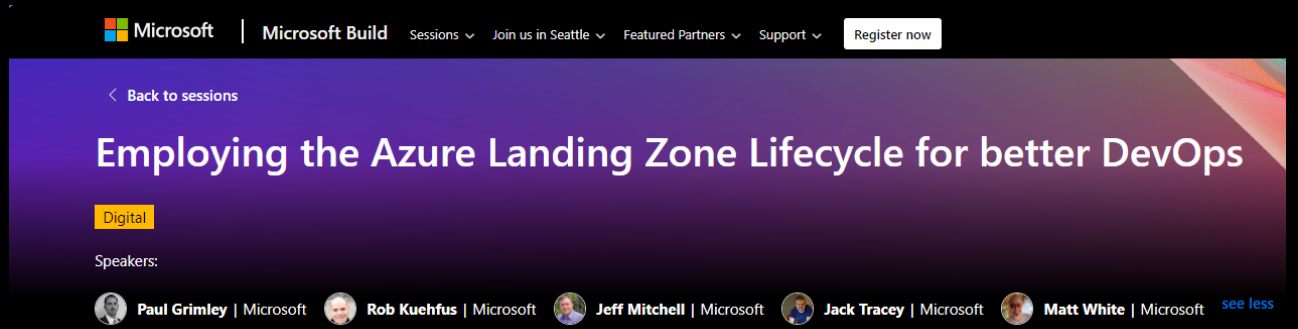
Outlook

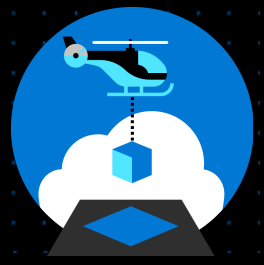


- Implementing User-Defined Types
- Alignment with the Public Bicep Registry (PBR)
  - Adjusting folder & files names
    - e.g., *Microsoft.Network/virtualNetworks* → *network/virtual-networks*
    - e.g., *deploy.bicep* → *main.bicep*
  - Adding PBR files such as the *metadata.json*
- Enhancing the *BRM* tool with the capabilities CARML's `*Set-ModuleReadme.ps1*` script
- Publish top-level modules into the PBR



We are hosting a roundtable at...





Next Community Call will be on 25th September

Back to a US friendly time slot for this occurrence and then the one after will be back to this time slot

Stay tuned to issue #1287 (ALZ/ESLZ Repo)

Recordings will be available at aka.ms/ALZ/Community

Microsoft

Q & A

Microsoft

Thank You!

Stay up-to-date:
https://aka.ms/ALZ/WhatsNew