10 Things Every Dev Should Know about Azure

Slides at scottsauber.com

@scottsauber

Audience

- Developers (examples in C# but if you're not that's okay)
- Haven't had much exposure to Azure

Polls

- How often are people in the Azure portal or in an IAC tool?
- Every day?
- Every week?
- Every month?
- Longer?
- How many people are responsible for Azure?
- How many people have a team or a person responsible for Azure who isn’t them?

Agenda

- Structured as a series of lightning talks
- Account Organization
- DefaultAzureCredential (and why you shouldn’t be using it)
- Naming Standards
- Reservations and Savings Plans
- Managed Identities
- Federated Credentials
- Azure Key Vault
- Azure Monitor
- Azure App Services
- Azure Container Apps
- Infrastructure As Code

Goals

- Give you practical tips that can be applied immediately
- Further understanding how Azure works and things you should be leveraging

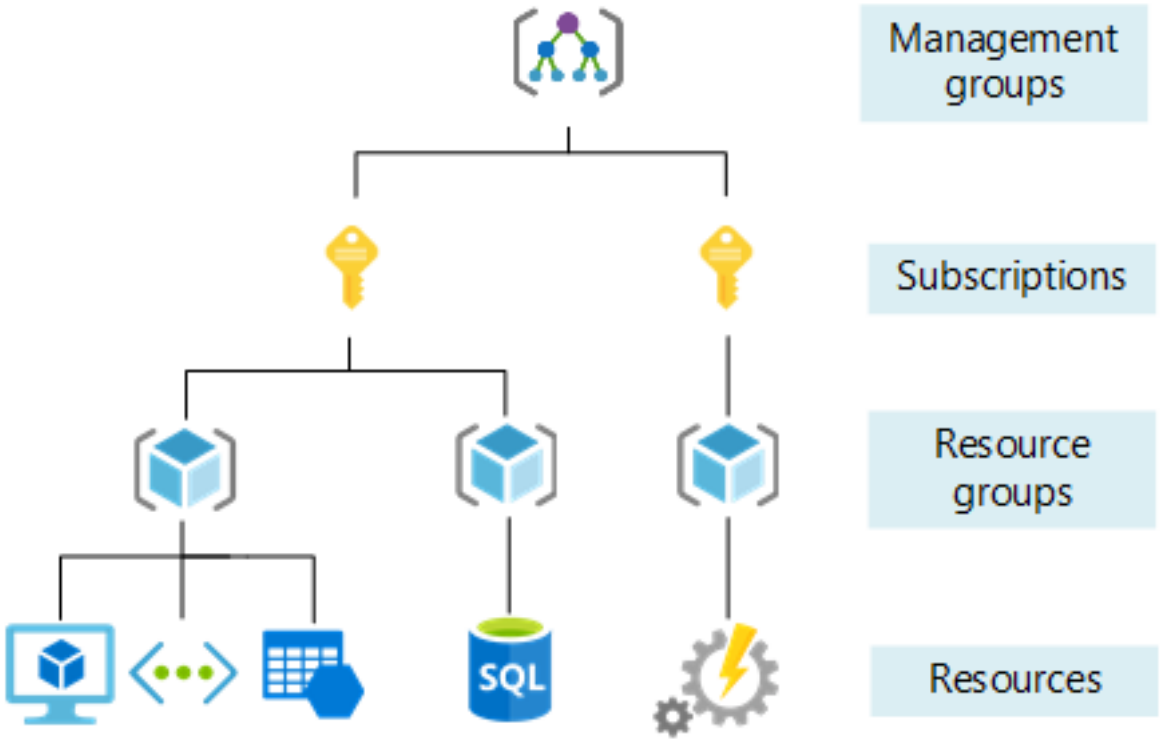
Who am I?

- Director of Engineering at Lean TECHniques
- Microsoft MVP
- Dometrain Author
- Redgate Community Ambassador
- Co-organizer of Iowa .NET User Group

Account Organization

Subscriptions vs Resource Groups

- Ignoring Management Groups bc you probably don’t care about those



Develop Account Organization Strategy

- Subscriptions - It Depends™
  - Minimum: per environment boundaries
  - Maximum: per team per environment boundaries
- Resource Groups – always per app per environment
- Service Groups – new as of May 2025
  - Fancy tags
  - Overlay to pull together across subscriptions, resource groups, etc – ie “Production apps” or “PCI Scope” or “MyApp”

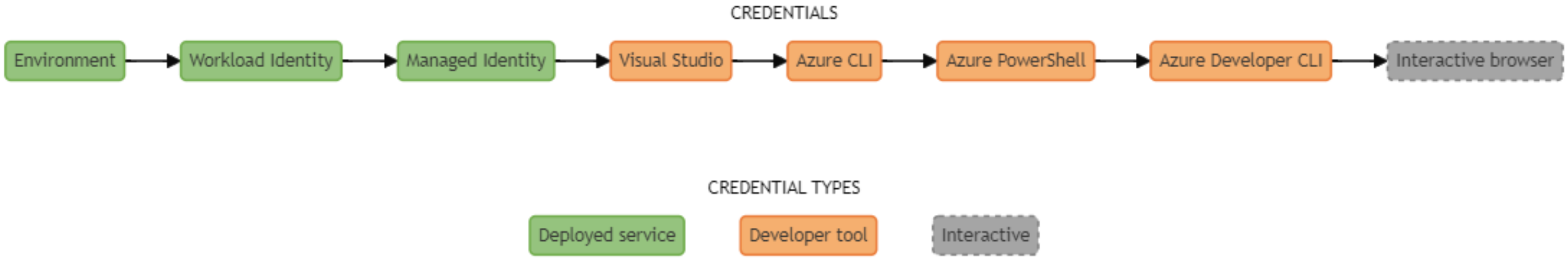
Questions?

DefaultAzureCredential (and why you should NOT be using it)

Problem: People use DefaultAzureCredential…

- …without knowing the tradeoffs
- DefaultAzureCredential is meant to get you into Azure as quickly as possible
- The downside is – performance

Problem: People use DefaultAzureCredential…



Solution: Stop - Be explicit about credentials

- Locally – use AzureCli or VisualStudioCredential… else use ManagedIdentity
- … or use Excludes on DefaultAzureCredential
- This will save you ~2-10 seconds when interacting with the credential
- DefaultAzureCredential Best Practices
- My blog post on this in 2022

Solution: Be explicit about credentials

```
// Old
builder.Configuration.AddAzureKeyVault(
    new Uri($"https://kv-myapp-{builder.Environment.EnvironmentName}.vault.azure.net"),
    new DefaultAzureCredential());

// New
TokenCredential tokenCredential;
if (builder.Environment.IsEnvironment("Local"))
{
    tokenCredential = new ChainedTokenCredential( params sources: new AzureCliCredential(),
      new VisualStudioCredential());
}
else
{
    tokenCredential = new ManagedIdentityCredential();
}
builder.Configuration.AddAzureKeyVault(
    new Uri($"https://kv-myapp-{builder.Environment.EnvironmentName}.vault.azure.net"),
    tokenCredential);
```

Microsoft trying to fix this

Restart the app using Azure CLI, PowerShell, or the Azure portal.

The sample app creates an instance of the DefaultAzureCredential class. The credential attempts to obtain an access token from environment for Azure resources:

C# Copy

```
using Azure.Identity;

var builder = WebApplication.CreateBuilder(args);

if (builder.Environment.IsProduction())
{
    builder.Configuration.AddAzureKeyVault(
        new Uri($"https://{builder.Configuration["KeyVaultName"]}.vault.azure.ne
        new DefaultAzureCredential());
}
```

ⓘ Note

The preceding example uses DefaultAzureCredential to simplify authentication while developing apps that deploy to Azure by combining credentials used in Azure hosting environments with credentials used in local development. When moving to production, an alternative is a better choice, such as ManagedIdentityCredential. For more information, see Authenticate Azure-hosted .NET apps to Azure resources using a system-assigned managed identity.

Key Vault name example value: `contosovault`

Reuse Credential Instances

- Improve app resilience
- Reduce number of token requests to Entra
- Microsoft.Extensions.Azure package

Reuse Credential Instances

C# Copy

```
builder.Services.AddAzureClients(clientBuilder =>
{
    clientBuilder.AddSecretClient(
        new Uri($"https://{keyVaultName}.vault.azure.net"));
    clientBuilder.AddBlobServiceClient(
        new Uri($"https://{storageAccountName}.blob.core.windows.net"));

    TokenCredential credential;

    if (builder.Environment.IsProduction() || builder.Environment.IsStaging(
    {
        string? clientId = builder.Configuration["UserAssignedClientId"];
        credential = new ManagedIdentityCredential(
            ManagedIdentityId.FromUserAssignedClientId(clientId));
    }
    else
    {
        // local development environment
        credential = new ChainedTokenCredential(
            new VisualStudioCredential(),
            new AzureCliCredential(),
            new AzurePowerShellCredential());
    }

    clientBuilder.UseCredential(credential);
});
```

Resources

- https://learn.microsoft.com/en-us/dotnet/azure/sdk/authentication/best-practices?tabs=aspdotnet

Questions?

Naming Standards

Problem: Naming is a mess

- Who here feels good about their Azure naming conventions?
- Naming has no consistency across apps, teams, divisions

Solution: Implement a Naming Standard

- Microsoft has a recommended naming convention based on recommended abbreviations
- Example: app-navigator-dev-001 for an application called Navigator, that's an App Service running in the Dev env
- Azure has a naming tool you can run
- AzureNamingTool
- More important that you pick something and be consistent
- Enforce with Azure Policy (more soon)

Questions?

Reservations && Savings Plans

Problem: Cost is getting too high

- Cloud isn’t always cheaper up front
- Sometimes cloud is about speed, lower TCO, and more visibility

Solution: Azure Reservations

- Commit to using certain resources for 1 year or 3 years, get a discount between 20% - 70% depending on the resource (App Service Plans, SQL, VMs, Cosmos, etc)
- Downside – what if you get rid of the resource after 2 years but you committed to 3
- Still pay monthly or up front (if you want)

Solution: Azure Savings Plans

- Commit to using certain spend $ for 1 yr or 3 yrs, get a discount
- Can be used however you want across eligible resource types

Comparing Savings Plans vs Reservations

- Reservations give you more savings, but more risk if you need to change your plans
- I recommend everyone use at least Savings Plans
- Go tell your boss you can cut the Azure bill by at least 20% and get a promotion
- Comparison of a single P0V3 App Service Plan:

Premium v3 Service Plan	vCPU(s)	RAM	Storage	Pay as you go	1 year savings plan *	3 year savings plan*	1 year reserved	3 year reserved
P0v3	1	4 GB	250 GB	£89.246/month	£74.339/month ~17% savings	£66.042/month ~26% savings	£64.089/month ~28% savings	£52.217/month ~41% savings

Questions?

Managed Identities

Problem: We are managing credentials

- Credentials to manage – create, store, rotate, revoke
- Azure SQL still using SQL Auth – username + password
- Key Vault still using Client ID + Client Secret
- Someone might accidentally commit a secret

Solution: Use Managed Identities

- Managed Identities allow you to give a resource an “identity” (user to run as essentially)
- These Identities are managed completely by Azure – no username/password/client id/client secret needed
- Assign a Managed Identity to an App Service/Function/etc
- Give that Managed Identity access to read secrets, connect to a DB, etc
- Completely passwordless
- In your code you can use the ManagedIdentityCredential (not DefaultAzureCredential)

Questions?

Federated Credentials

Problem: Not using Federated Credentials

- Your CI/CD pipelines need to auth to Azure to provision resources, deploy, etc
- Without Federated Credentials, you're managing a Client ID + Client Secret (most likely)
- All the same issues managing credentials (create, store, rotate, revoke)
- If someone gets these credentials, they can manipulate your Azure environment

Solution: Use Federated Credentials

- Federated Credentials allows you to “trust” repositories to push to Azure
- Passwordless
- Short-lived tokens (minutes)
- How to do this with GitHub Actions

Configure an Microsoft Entra ID managed identity or an identity from an external OpenID Connect Provider to get tokens as this application and access Azure resources.

Federated credential scenario * GitHub Actions deploying Azure resources

Connect your GitHub account

Please enter the details of your GitHub Actions workflow that you want to connect with Microsoft Entra ID. These values will be used by Microsoft Entra ID to validate the connection and should match your GitHub OIDC configuration. Issuer has a limit of 600 characters. Subject Identifier is a calculated field with a 600 character limit.

Issuer ⓘ https://token.actions.githubusercontent.com

Edit (optional)

Organization * scottsauber

Repository * workshop-dotnet-azure-github-bicep

Entity type * Branch

Based on selection * main

Questions?

Azure Key Vault

Azure Key Vault

- Where you can store Secrets, Keys, Certificates
- Most of the time
- You should be using RBAC not Access Policies
  - Key Vault Secrets User can read secrets
  - Key Vault Secrets Officer can read/write secrets
- You should be using Managed Identities for your applications to talk to Azure Key Vault
  - ie app-myapi-dev can talk to Azure Key Vault with Secrets User RBAC role
- Create groups to assign your developers permission to those Key Vaults
- Developers need at least ability to list secrets to see what's there
  - Key Vault Reader == list, but Key Vault Secrets Officer for r+w

Azure Key Vault in .NET

- $0.03 per 10,000 requests
- Load secrets on startup
- Azure.Identity Package
- Azure.Extensions.AspNetCore.Configuration.Secrets

Azure Key Vault in .NET

```
var builder = WebApplication.CreateBuilder(args);

builder.Configuration.AddAzureKeyVault(
    new Uri($"https://{builder.Configuration["KeyVaultName"]}.vault.azure.net/"),
    new ChainedTokenCredential(
        new ManagedIdentityCredential(new ManagedIdentityCredentialOptions()),
        new AzureCliCredential()));
```

Use Azure Key Vault for Local Secrets

- If you have local secrets… API keys… etc
- Don’t use `dotnet-secrets`
- It’s ok for 1 or 2 person teams, not ok for large teams
- Need to hand around magic JSON files or values
- Instead create a Key Vault for local development
- Shared by everyone
- “But I need offline”
- …what are you doing offline that you need a secret for?

Questions?

Azure Monitor

Azure Monitor

- Conglomerate of Azure services
- Azure Application Insights
- Azure Log Analytics
- Alerts
- Dashboards
- Workbooks

Azure Application Insights

- APM tool
- Answers questions like “Why is this page slow?” or “How is the system being used in Production?”
- Also stores logs
- Leverage Managed Identity to talk to Application Insights

CH1-RetailAppAI | Application map

Application Insights | Last hour - CH1-RetailAppAI

Search

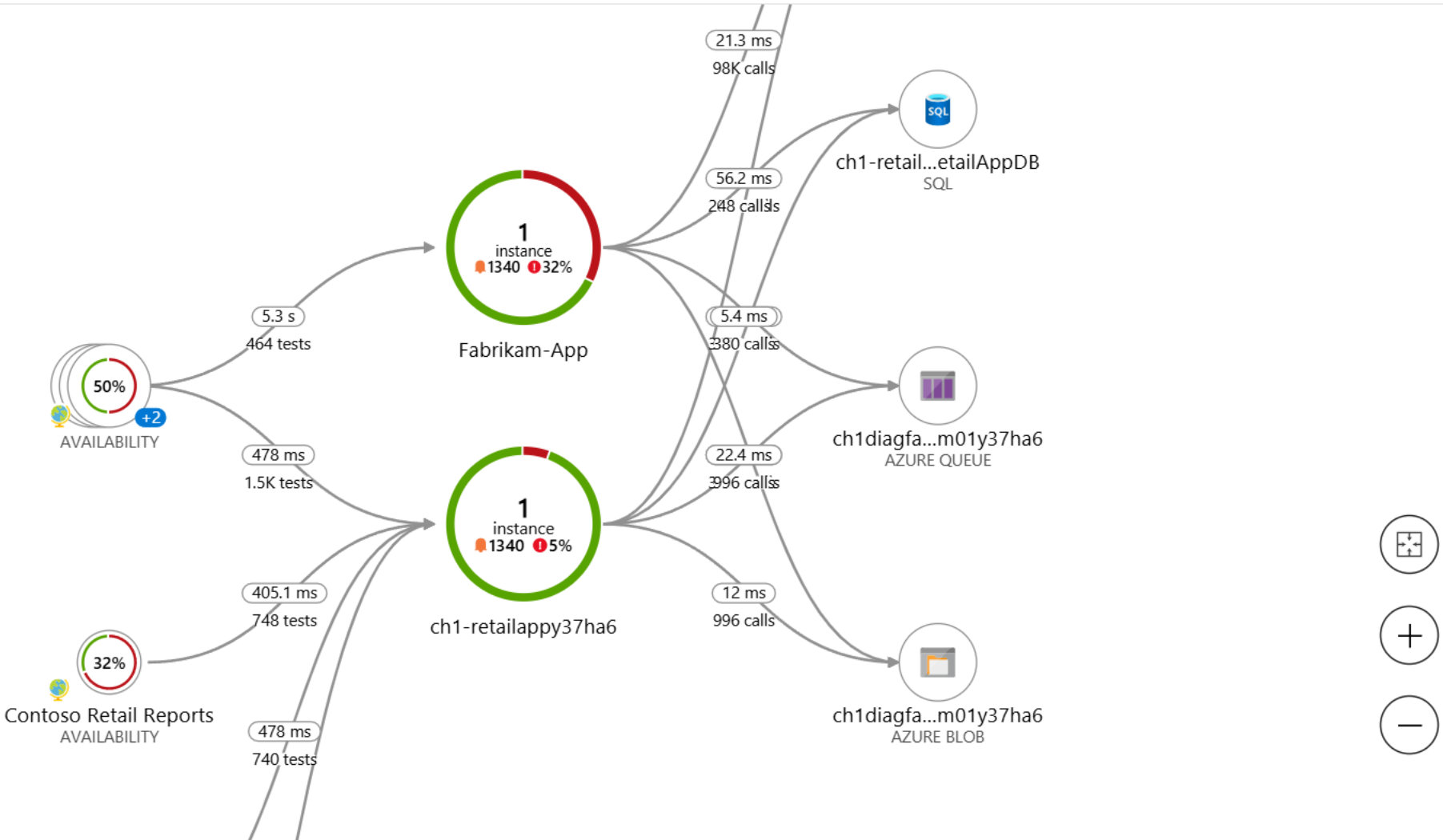
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Resource visualizer
Investigate
Application map
Smart detection
Live metrics
Search
Availability
Failures
Performance
Agents (preview)
Monitoring
Alerts
Metrics

Add or remove favorites by pressing Ctrl+Shift+F

Refresh | Save view | Load view | Copy link | Learn more | Troubleshooting | Leave preview | Feedback

Last hour | Add filter

Update map components | Intelligent view: On Off | Low Medium High | Exclude 4xx: On Off | Layout:



AppInsightsDemo | Performance

Application Insights | Directory: Microsoft

Search

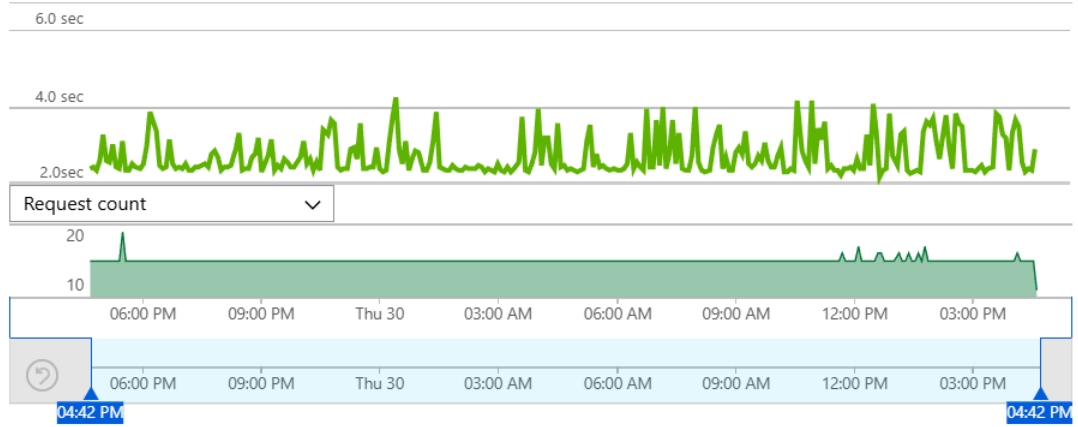
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Resource visualizer
Investigate
Application map
Smart detection
Live metrics
Search
Availability
Failures
Performance
Agents (preview)
Monitoring
Usage
Configure
Settings
Automation
Help

Server | Browser | Local Time: Last 24 hours | Roles = All

Operations | Dependencies | Roles

Operation times: zoom into a range

Avg | 50TH | 95TH | 99TH



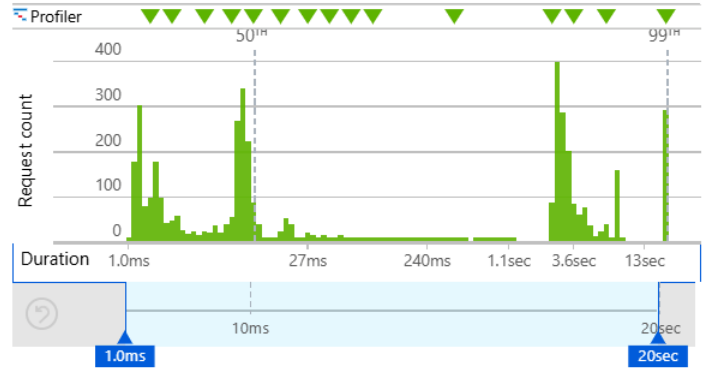
Select operation

Search to filter items...

OPERATION NAME	DURATION (AVG)	COUNT	PIN
Overall	**2.70 sec**	**4.33k**	
GET /perfCPU/HighCPU/ArraySort/20000	20.2 sec	290	
GET /perfCPU/HighCPUAsync/2000	4.01 sec	1.45k	
GET /	69.9 ms	574	
GET /exception/Throw	6.27 ms	578	
GET /exception/ThrowIndexOutOfBounds	5.47 ms	1.44k	
GET /exception/ThrowIndexOutOfBound	2.87 ms	1	

Overall

Distribution of durations: zoom into a ... Scale



Insights (2)

Top 3 Code Optimizations

TYPE	IMPACT	PERFORMANCE ISSUE	CLOUD ROLE
CPU	5%	Array.Sort is causing high CPU	diagservicecw

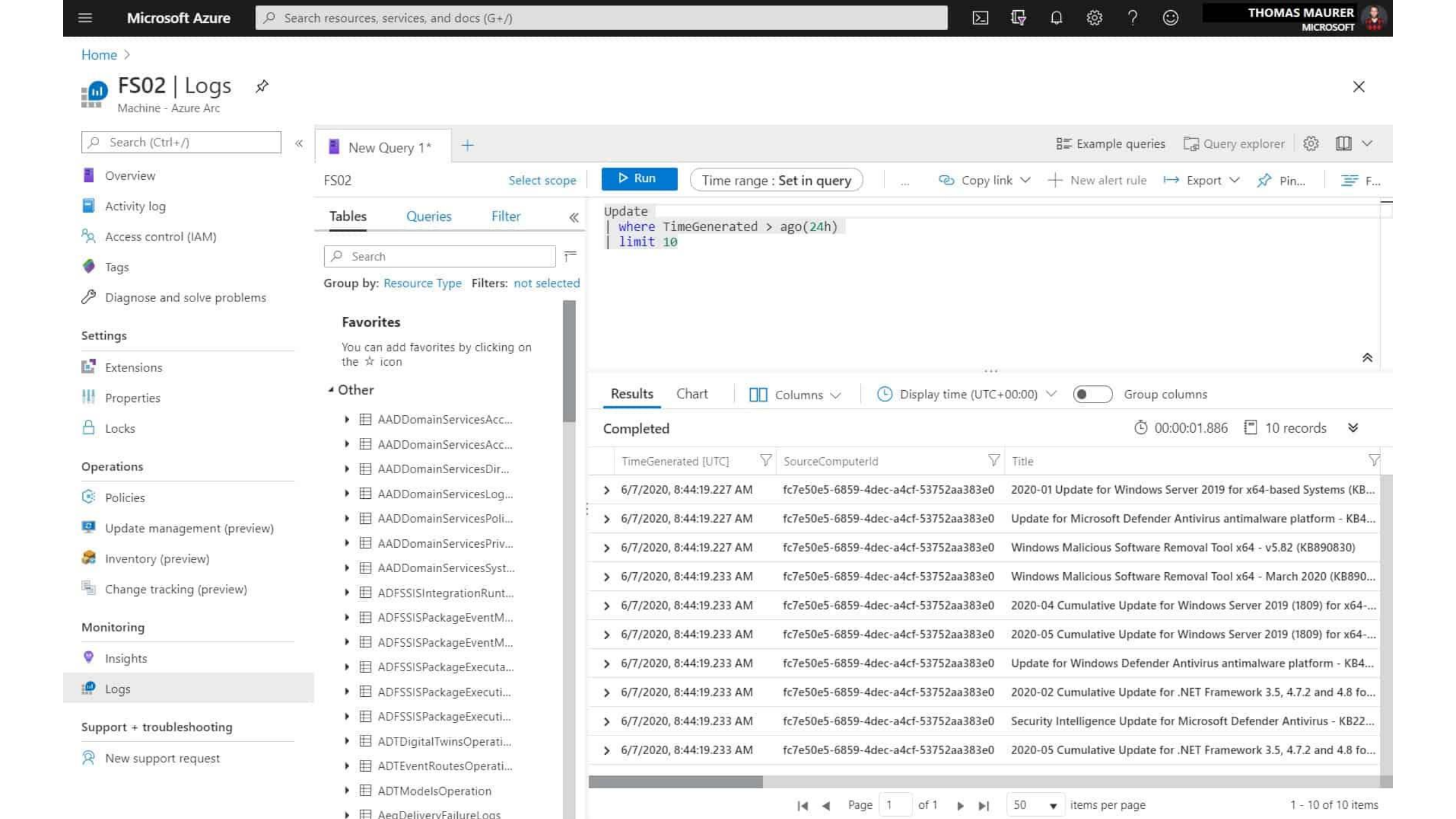
53% COMMON PROPERTIES:
performanceBucket, cloud_RoleName, cloud_RoleI...

Drill into...

4.33k Samples | 131 Profiler traces

Azure Log Analytics

- Ingest in logs in centralized log aggregator – no log files
- Be able to query those logs using KQL
- Alert off logs


Microsoft Azure
Search resources, services, and docs (G+/)
THOMAS MAURER
MICROSOFT
Home
FS02 | Logs
Machine - Azure Arc
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Settings
Extensions
Properties
Locks
Operations
Policies
Update management (preview)
Inventory (preview)
Change tracking (preview)
Monitoring
Insights
Logs
Support + troubleshooting
New support request
New Query 1*
Example queries
Query explorer
FS02
Select scope
Run
Time range : Set in query
Copy link
New alert rule
Export
Pin...
Tables
Queries
Filter
Search
Group by: Resource Type
Filters: not selected
Favorites
You can add favorites by clicking on the ☆ icon
Other
AADDomainServicesAcc...
AADDomainServicesAcc...
AADDomainServicesDir...
AADDomainServicesLog...
AADDomainServicesPoli...
AADDomainServicesPriv...
AADDomainServicesSyst...
ADFSSISIntegrationRunt...
ADFSSISPackageEventM...
ADFSSISPackageEventM...
ADFSSISPackageExecuta...
ADFSSISPackageExecuti...
ADFSSISPackageExecuti...
ADTDigitalTwinsOperati...
ADTEventRoutesOperati...
ADTModelsOperation
AegDeliveryFailureLogs
Update
| where TimeGenerated > ago(24h)
| limit 10
Results
Chart
Columns
Display time (UTC+00:00)
Group columns
Completed
00:00:01.886
10 records
TimeGenerated [UTC]
SourceComputerId
Title
6/7/2020, 8:44:19.227 AM fc7e50e5-6859-4dec-a4cf-53752aa383e0 2020-01 Update for Windows Server 2019 for x64-based Systems (KB...
6/7/2020, 8:44:19.227 AM fc7e50e5-6859-4dec-a4cf-53752aa383e0 Update for Microsoft Defender Antivirus antimalware platform - KB4...
6/7/2020, 8:44:19.227 AM fc7e50e5-6859-4dec-a4cf-53752aa383e0 Windows Malicious Software Removal Tool x64 - v5.82 (KB890830)
6/7/2020, 8:44:19.233 AM fc7e50e5-6859-4dec-a4cf-53752aa383e0 Windows Malicious Software Removal Tool x64 - March 2020 (KB890...
6/7/2020, 8:44:19.233 AM fc7e50e5-6859-4dec-a4cf-53752aa383e0 2020-04 Cumulative Update for Windows Server 2019 (1809) for x64-...
6/7/2020, 8:44:19.233 AM fc7e50e5-6859-4dec-a4cf-53752aa383e0 2020-05 Cumulative Update for Windows Server 2019 (1809) for x64-...
6/7/2020, 8:44:19.233 AM fc7e50e5-6859-4dec-a4cf-53752aa383e0 Update for Windows Defender Antivirus antimalware platform - KB4...
6/7/2020, 8:44:19.233 AM fc7e50e5-6859-4dec-a4cf-53752aa383e0 2020-02 Cumulative Update for .NET Framework 3.5, 4.7.2 and 4.8 fo...
6/7/2020, 8:44:19.233 AM fc7e50e5-6859-4dec-a4cf-53752aa383e0 Security Intelligence Update for Microsoft Defender Antivirus - KB22...
6/7/2020, 8:44:19.233 AM fc7e50e5-6859-4dec-a4cf-53752aa383e0 2020-05 Cumulative Update for .NET Framework 3.5, 4.7.2 and 4.8 fo...
Page 1 of 1
50 items per page
1 - 10 of 10 items

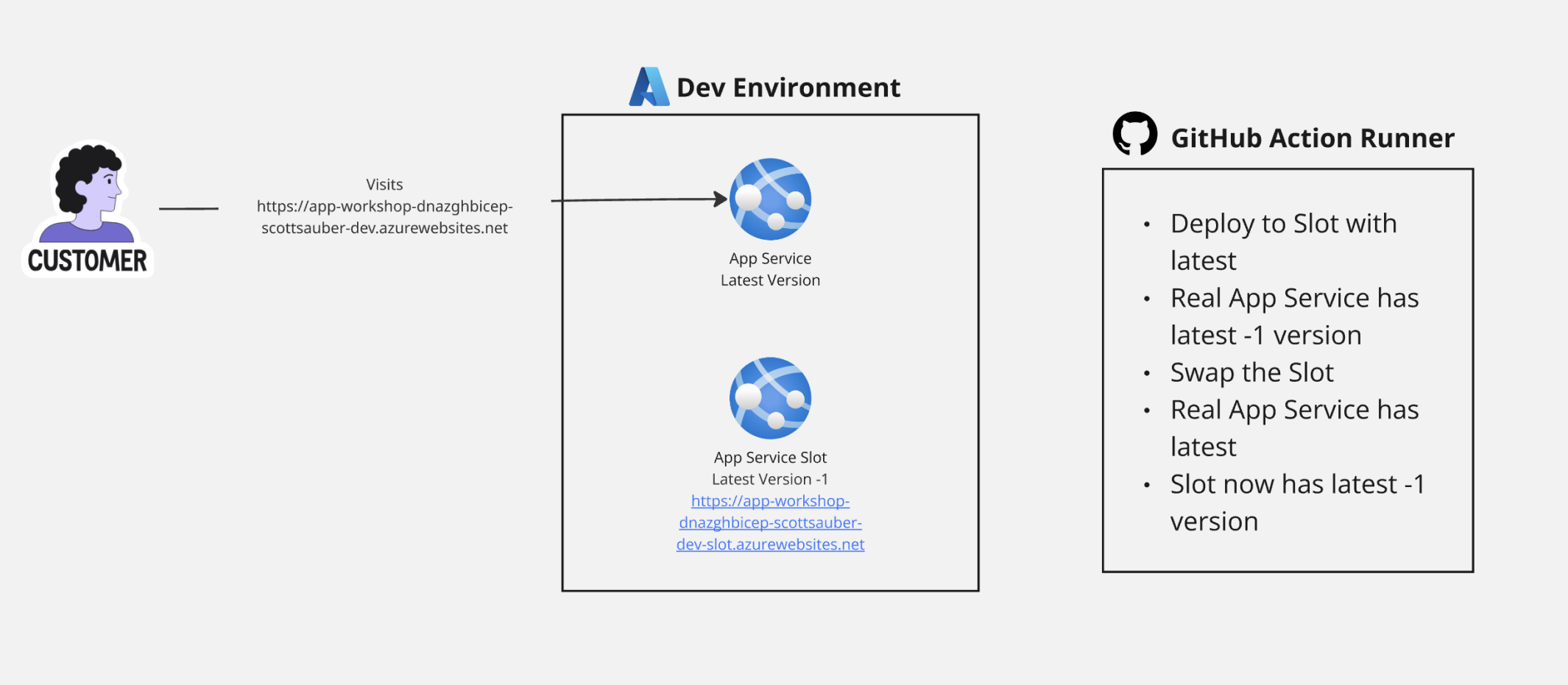
Questions?

Azure App Services

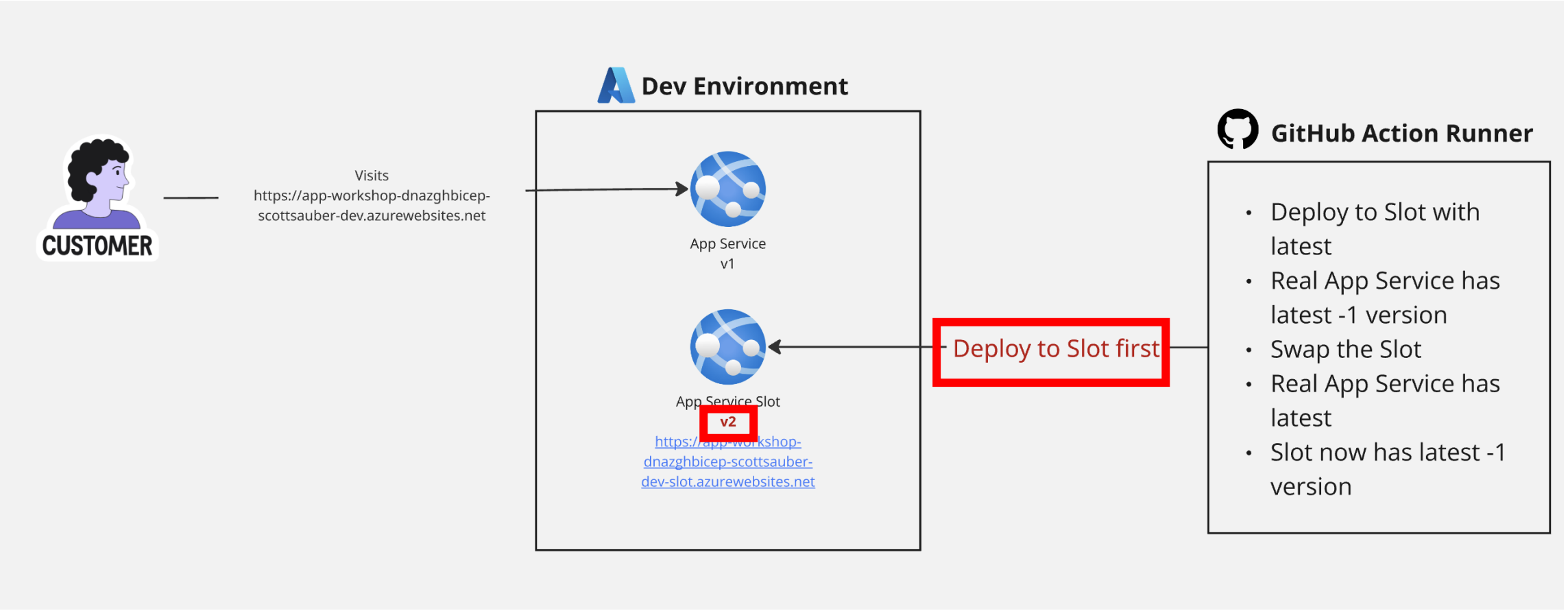
Azure App Service

- Simplest way of hosting a web app or API (imo) across major clouds
- PaaS
- Upload zip, select runtime, and go
- Customize environment variables, custom domains, and more
- Also able to use containers instead of a zip, just set an env var
- Zero Downtime Deployments with Deployment Slots
- Try to avoid Slots per environment

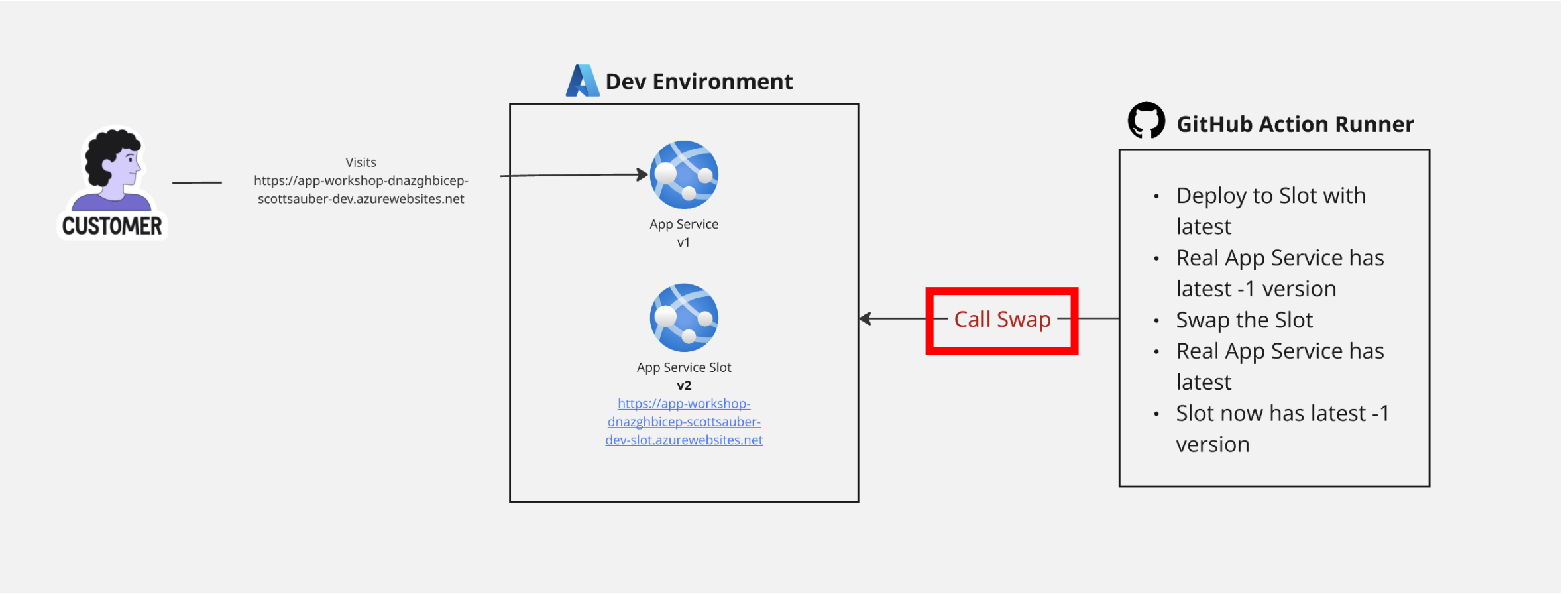
Azure App Service Slots - ZDD



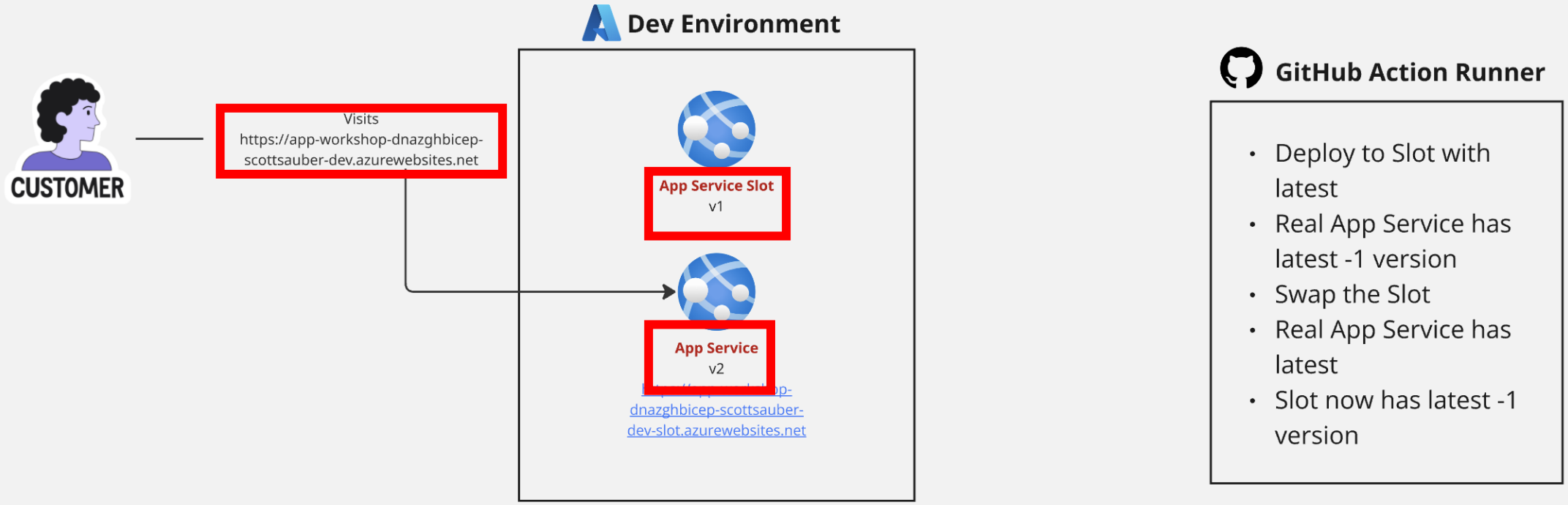
Azure App Service Slots - ZDD



Azure App Service Slots - ZDD



Azure App Service Slots - ZDD



Azure App Service Plans

- PaaS but it’s a managed VM
- You pick memory, CPU, storage, and operating system
- Host many App Services on a single App Service Plan
- Should you? Noisy Neighbor
- You get charged by the App Service Plan not the App Service

Questions?

Azure Container Apps

Azure Container Apps

- PaaS
- Run a container as a service
- Allows you to spin down to zero vs App Services
- A Simpler Kubernetes

Azure Container Apps vs Azure App Service

- Azure App Service way simpler
- Azure Container Apps more powerful, in the right hands

Questions?

Infrastructure as Code

What is IAC?

- Source code that defined provisioning resources that’s configurable and repeatable across all environments
- Stored in version control
- Declarative – what resources to create, not how to create them
- Deployed via automation (ideally pipeline, but could be local)
- Resource properties can be linked together

Why IAC?

- Promotes consistency and standardization
- “It Works On My Machine” goes away
- “It works in Dev, but not in Production” goes away
- Audit trail of who did what and when
- Resources are Transient not Permanent
- Can be deleted and re-created with ease (besides your DB)
- Standing up a new environment is a few lines of code and minutes
- ClickOps doesn’t scale

What can I create with IAC?

- Virtual Machines
- Web Servers
- Databases
- Secret Stores
- Networking
- IAM Policies
- Monitoring
- DNS
- …pretty much everything

Ok great... But how?

What is Azure Bicep?

- Used to configure Azure resources
- Built and maintained by Microsoft
- Domain-specific language (fancy word for custom)
- Provides intellisense, error checking, “whatif,” and orders the resource creations
- Built on top of Azure Resource Manager (ARM) – don’t use ARM directly
- No state file

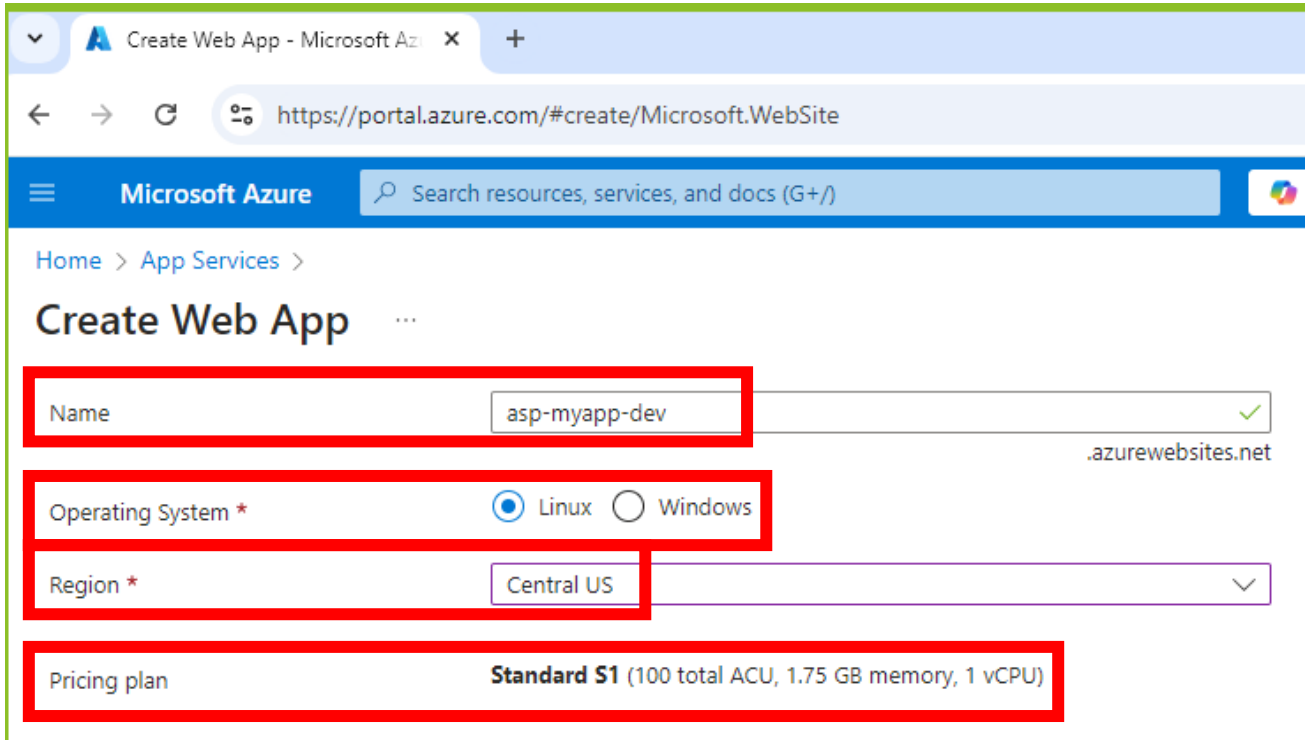
What is Azure Bicep?

appservice.bicep

```
1  resource appServicePlan 'Microsoft.Web/serverfarms@2022-09-01' = {
2    name: 'asp-myapp-dev'
3    location: 'centralus'
4    sku: {
5      name: 'S1'
6    }
7    kind: 'linux'
8  }
```

appservice.bicep

```
1  resource appServicePlan 'Microsoft.Web/serverfarms@2022-09-01' = {
2    name: 'asp-myapp-dev'
3    kind: 'linux'
4    location: 'centralus'
5    sku: {
6      name: 'S1'
7    }
8  }
```



Terraform

- Created by HashiCorp
- Domain-specific language
- Configure different clouds (Azure, AWS, GCP, VMware, etc) via Providers
- BUT – can’t take same Terraform and run it on both Azure and AWS
- Maintains state of infrastructure via state file
- Likely the most popular IAC tool in the world
- Semi-recently acquired by IBM
- OpenTofu – drop-in replacement

appservice.tf

```
1  resource "azurerm_service_plan" "myapp" {
2    name                = "asp-myapp-dev"
3    resource_group_name = "rg-myapp-dev"
4    location            = "centralus"
5    os_type             = "Linux"
6    sku_name            = "S1"
7  }
```

appservice.bicep

```bicep
resource appServicePlan 'Microsoft.Web/serverfarms@2022-09-01' = {
  name: 'asp-myapp-dev'
  location: 'centralus'
  sku: {
    name: 'S1'
  }
  kind: 'linux'
}
```

appservice.tf

```terraform
resource "azurerm_service_plan" "myapp" {
  name                = "asp-myapp-dev"
  resource_group_name = "rg-myapp-dev"
  location            = "centralus"
  os_type             = "Linux"
  sku_name            = "S1"
}
```

Pulumi

- Created by Pulumi Corporation
- Configure different clouds (Azure, AWS, GCP, etc)
- You write your language of choice – TypeScript, JavaScript, Python, Java, C#, Go, or Visual Basic (yes really)
- First big player to market with IAC using an existing language

```
using System.Collections.Generic;
using System.Linq;
using Pulumi;
using Azure = Pulumi.Azure;

return await Deployment.RunAsync(() =>
{
    var example = new Azure.Core.ResourceGroup("example", new()
    {
        Name = "example-resources",
        Location = "West Europe",
    });

    var examplePlan = new Azure.AppService.Plan("example", new()
    {
        Name = "example-appserviceplan",
        Location = example.Location,
        ResourceGroupName = example.Name,
        Sku = new Azure.AppService.Inputs.PlanSkuArgs
        {
            Tier = "Standard",
            Size = "S1",
        },
    });
```

Configuration Files vs Code Files

- Bicep, ARM, Terraform, and Cloudformation are DSL config files
- CDK and Pulumi are libraries of existing languages
- CDK for Terraform is CDK but built on Terraform, supported by Hashicorp
- Configuration popular with Ops-focused teams
- Code popular with Dev-focused teams
- Code leverages existing skills – packages, syntax, autocomplete, etc

Questions?

Takeaways

- Ideas on what you can apply immediately back at work
- Better understanding of how different pieces of Azure works

Resources

- This slide deck at https://scottsauber.com

Questions?

ssauber@leantechniques.com

Add me on LinkedIn:

@scottsauber

@scottsauber.com

Slides at scottsauber.com

Thanks!

Add me on LinkedIn:

@scottsauber

@scottsauber.com

Slides at scottsauber.com