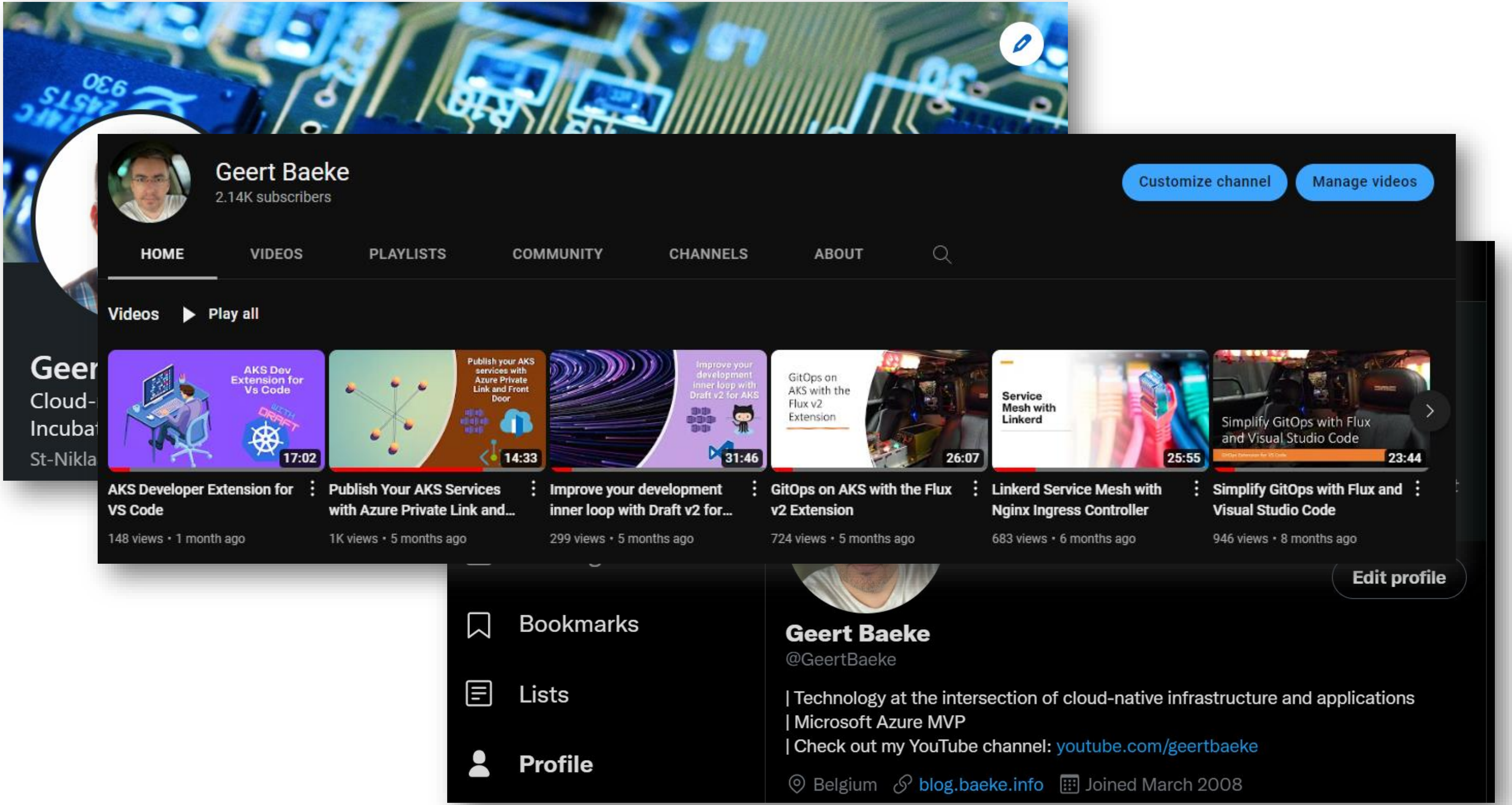
MC²

Goodbye Azure Kubernetes Service!

Hello Azure Container Apps!

vanroey.be



Geer
Cloud-
Incubat
St-Nikla

Geert Baeke
2.14K subscribers

Customize channel
Manage videos

HOME VIDEOS PLAYLISTS COMMUNITY CHANNELS ABOUT

Videos ▶ Play all

AKS Dev Extension for Vs Code
WITH DRAFT
17:02
Publish your AKS services with Azure Private Link and Front Door
14:33
Improve your development inner loop with Draft v2 for AKS
31:46
GitOps on AKS with the Flux v2 Extension
26:07
Service Mesh with Linkerd
25:55
Simplify GitOps with Flux and Visual Studio Code
23:44

AKS Developer Extension for VS Code
148 views • 1 month ago

Publish Your AKS Services with Azure Private Link and...
1K views • 5 months ago

Improve your development inner loop with Draft v2 for...
299 views • 5 months ago

GitOps on AKS with the Flux v2 Extension
724 views • 5 months ago

Linkerd Service Mesh with Nginx Ingress Controller
683 views • 6 months ago

Simplify GitOps with Flux and Visual Studio Code
946 views • 8 months ago

Bookmarks
Lists
Profile

Edit profile

Geert Baeke
@GeertBaeke

| Technology at the intersection of cloud-native infrastructure and applications
| Microsoft Azure MVP
| Check out my YouTube channel: youtube.com/geertbaeke

Belgium blog.baeke.info Joined March 2008

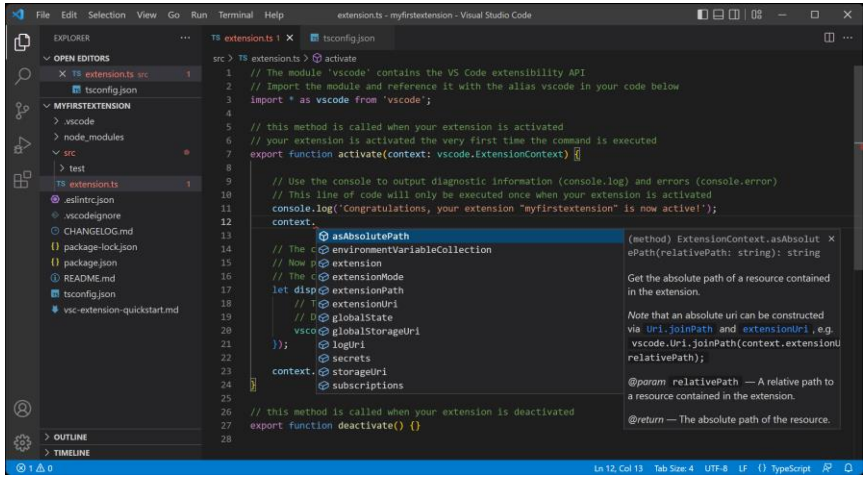
Containers

Lightweight

Fast

Package once, run anywhere

Windows/Linux



```
// The module 'vscode' contains the VS Code extensibility API
// Import the module and reference it with the alias vscode in your code below
import * as vscode from 'vscode';

// this method is called when your extension is activated
// your extension is activated the very first time the command is executed
export function activate(context: vscode.ExtensionContext) {

    // Use the console to output diagnostic information (console.log) and errors (console.error)
    // This line of code will only be executed once when your extension is activated
    console.log('Congratulations, your extension "myfirstextension" is now active!');
    context.

    // The c
    // Now p
    // The c
    let disp
        // T
        // D
        vsco
    });

    context.
}

// this method is called when your extension is deactivated
export function deactivate() {}
```

```
asAbsolutePath
environmentVariableCollection
extension
extensionMode
extensionPath
extensionUri
globalState
globalStorageUri
logUri
secrets
storageUri
subscriptions
```

(method) ExtensionContext.asAbsolutePath(relativePath: string): string

Get the absolute path of a resource contained in the extension.

Note that an absolute uri can be constructed via `Uri.joinPath` and `extensionUri`, e.g. `vscode.Uri.joinPath(context.extensionU relativePath);`

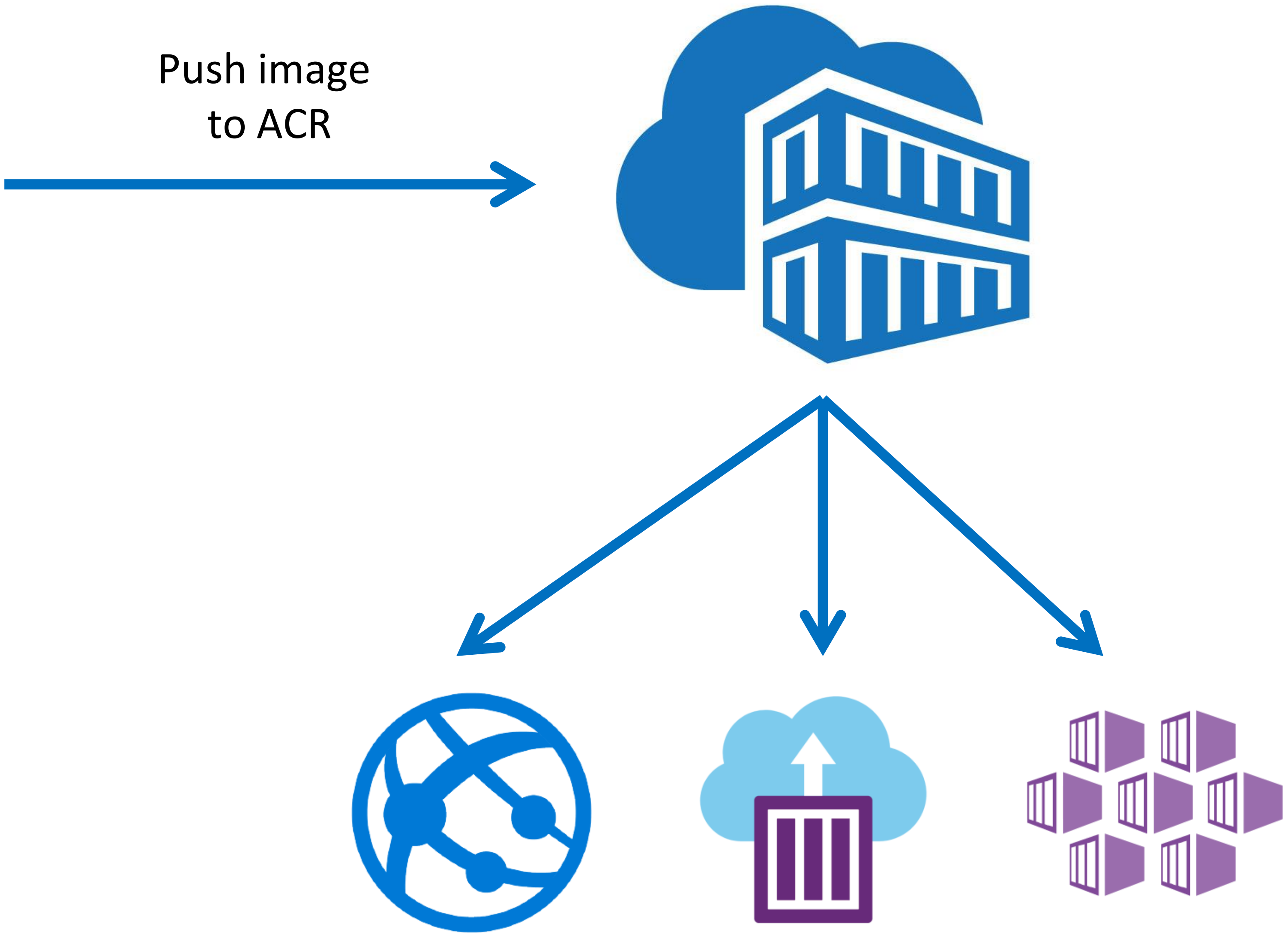
@param `relativePath` — A relative path to a resource contained in the extension.

@return — The absolute path of the resource.

Code

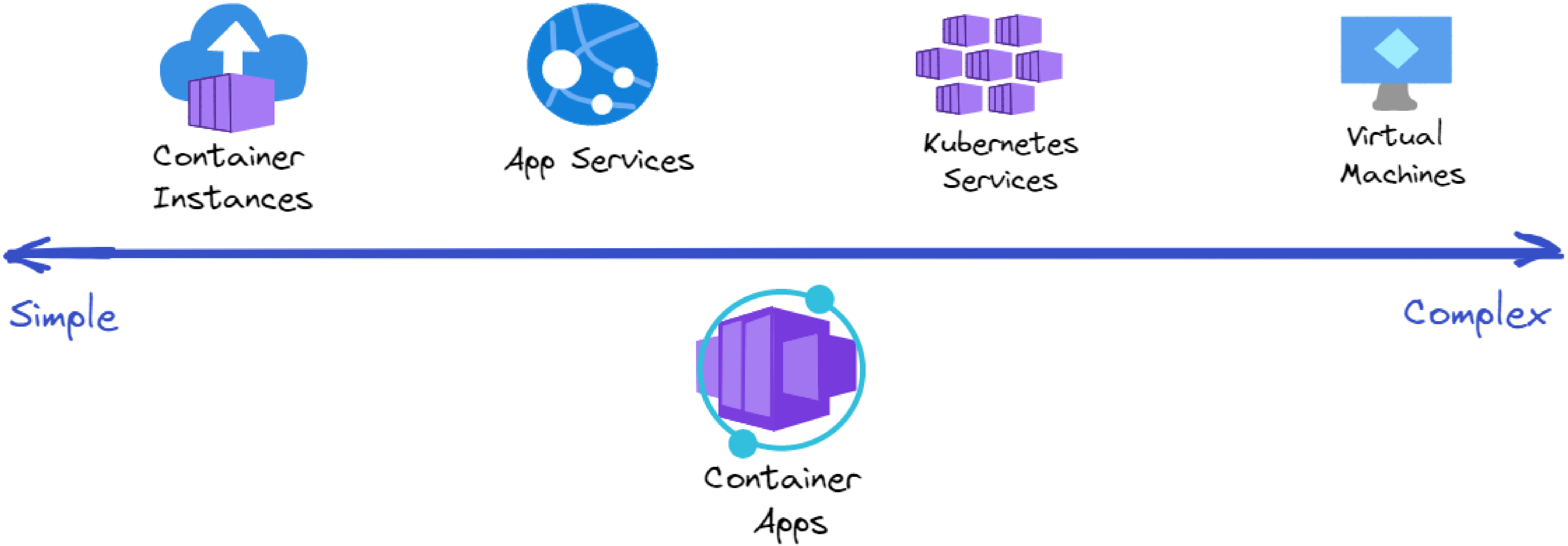
Dockerfile

docker build



Push image to ACR

Pull image to compute



Container Instances

App Services

Kubernetes Services

Virtual Machines

Simple

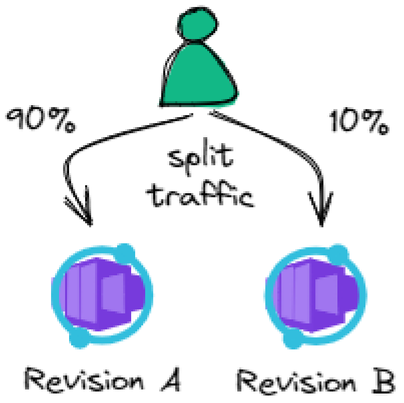
Complex

Container Apps

Container Apps

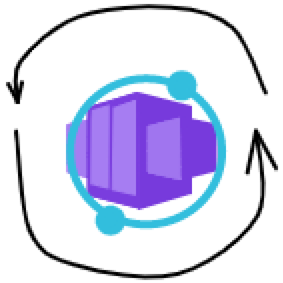
- Serverless containers for microservices
- Focus on apps, not infrastructure
- Scale dynamically based on events
- Built on open source

APIs and web apps



Scale based on concurrent HTTP requests Default = 10

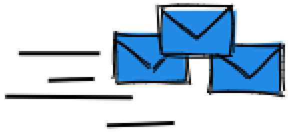
Background Processing



Continuously running background process

Scale based on CPU and memory (Keda scalers)

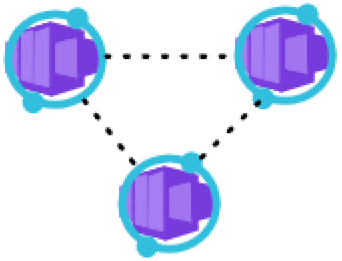
Event Processing



Reading messages from a topic or queue

Scale based on messages in the topic/queue (Keda scalers)

Microservices

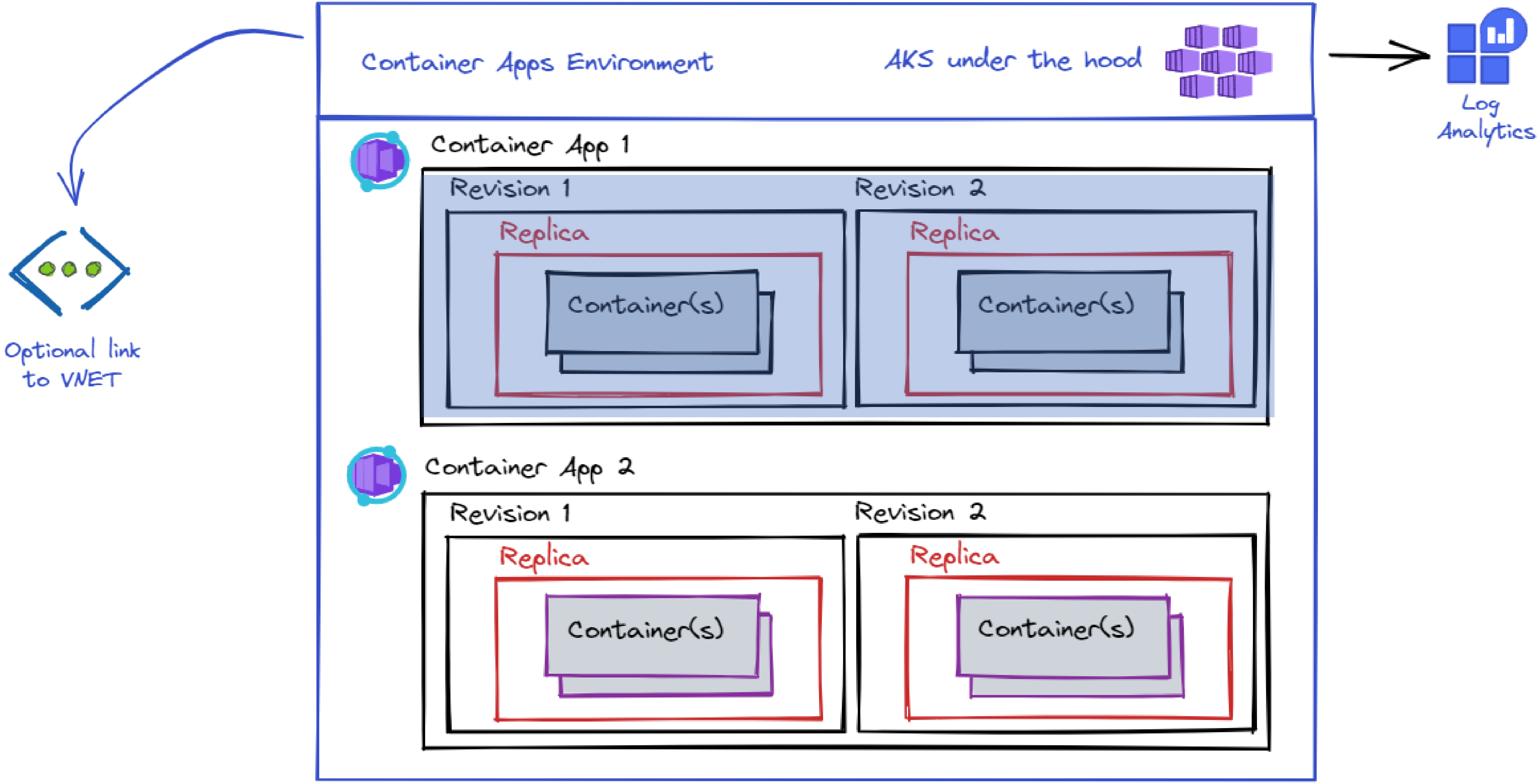


Microservices

Optional Dapr Integration

dapr

Scale based on any Keda scalers



Container Apps Environment

AKS under the hood

Log Analytics

Optional link to VNET

Container App 1

Revision 1

Replica

Container(s)

Revision 2

Replica

Container(s)

Container App 2

Revision 1

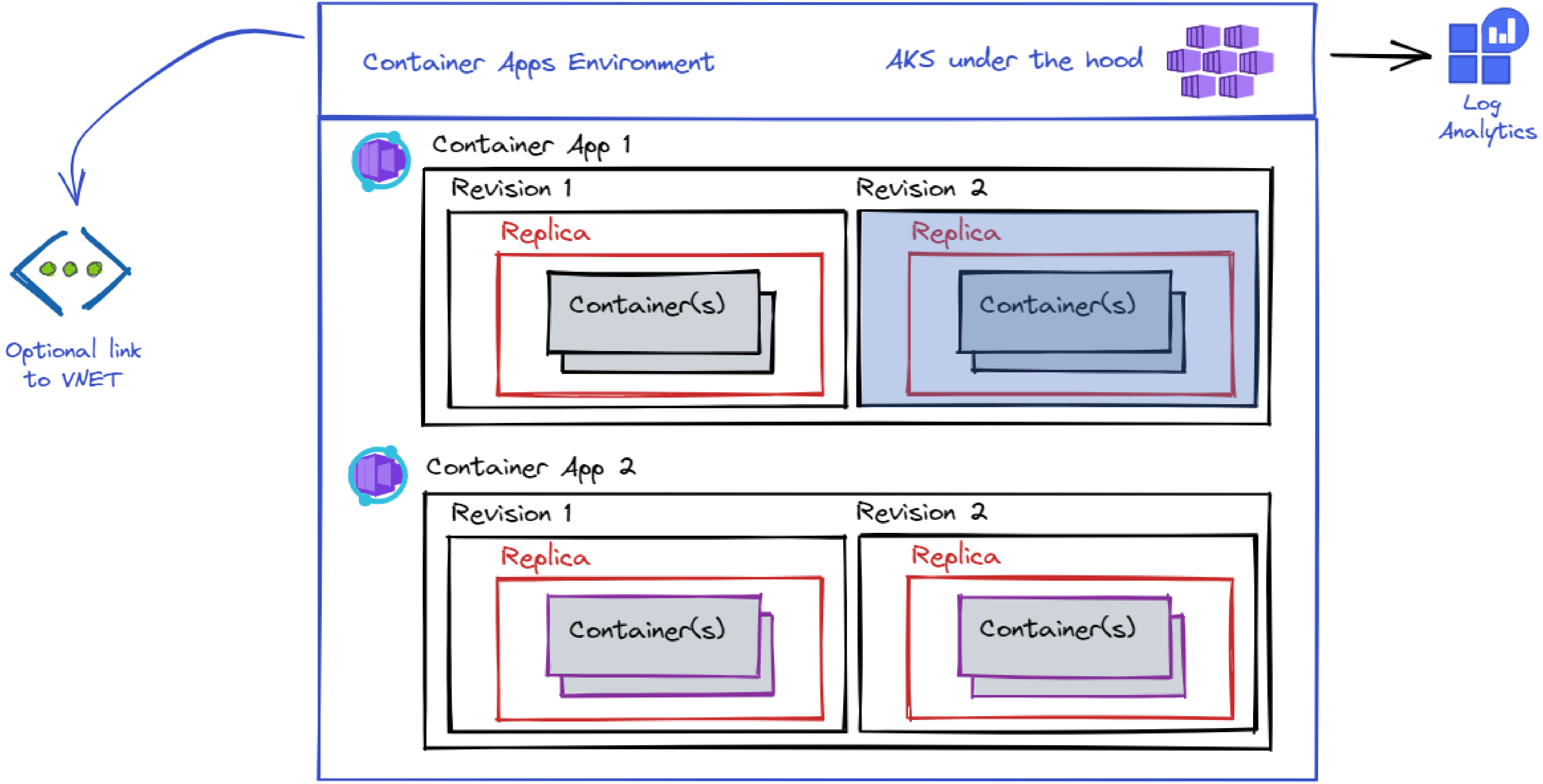
Replica

Container(s)

Revision 2

Replica

Container(s)


Container Apps Environment
AKS under the hood
Log Analytics
Optional link to VNET
Container App 1
Revision 1
Replica
Container(s)
Revision 2
Replica
Container(s)
Container App 2
Revision 1
Replica
Container(s)
Revision 2
Replica
Container(s)

Some rules

- Linux/amd64 container images only
- No privileged containers
- Maximum 2 cores and 4GB of RAM with fixed combinations
  - 0.25 cores per 0.5 Gi ⚠
- You can have more than one container in a container app

Azure Container Apps on Azure Arc

In public preview

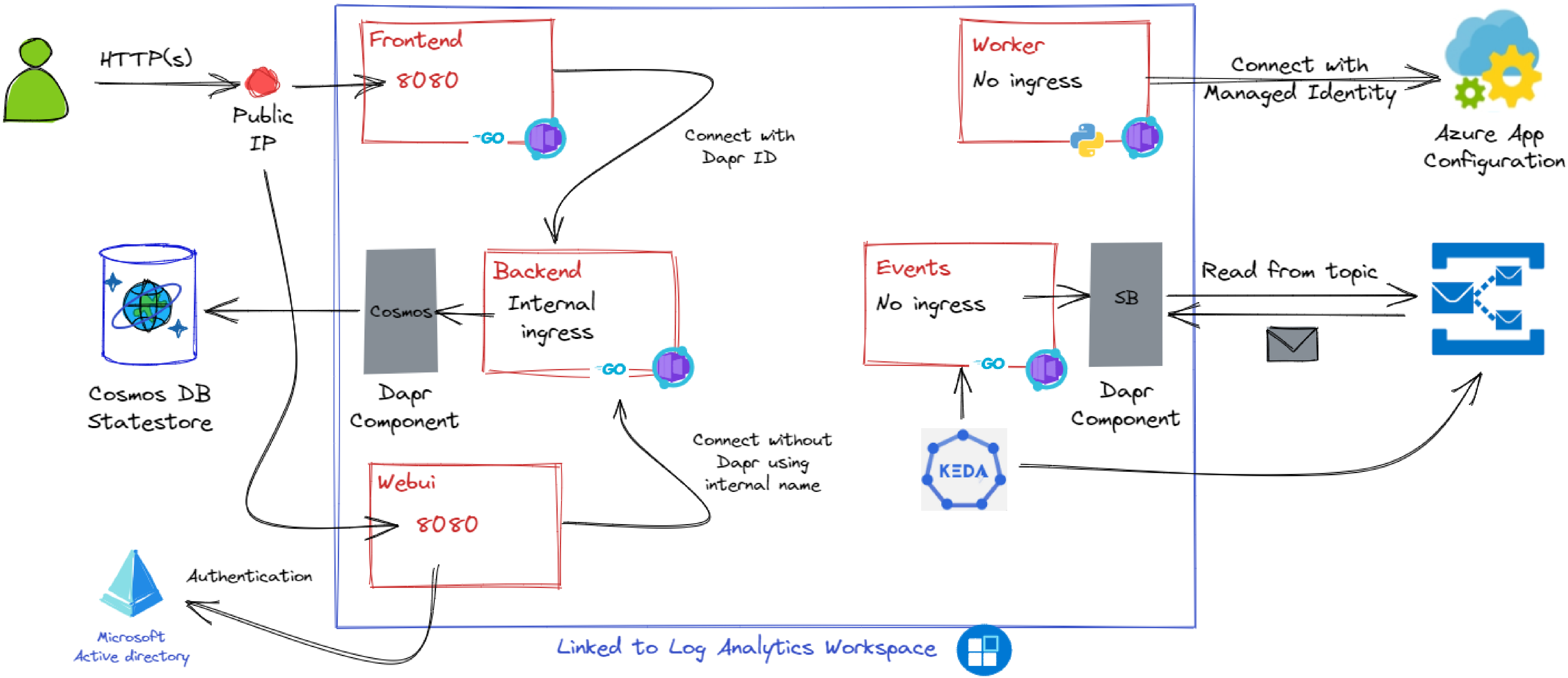
Some limitations:

Limited regions

LoadBalancer service type required

Managed Identity

Container App Environment - No VNET - powered by



HTTP(s)

Public IP

Frontend

8080

Connect with Dapr ID

Worker

No ingress

Connect with Managed Identity

Azure App Configuration

Cosmos

Dapr Component

Backend

Internal ingress

Cosmos DB Statestore

Events

No ingress

SB

Dapr Component

Read from topic

Connect without Dapr using internal name

Webui

8080

Authentication

Microsoft Active directory

Linked to Log Analytics Workspace

Extra resources

https://github.com/gbaeke/aca

https://aca.super.site