

# Microservices Made Easy!

Jessica Deen @jldeen | Principal Cloud Advocate



# Agenda



Microservice Landscape



Making Microservices Easy...er

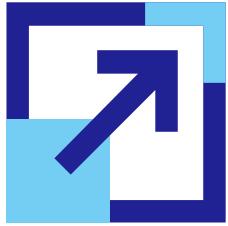


Review, Next Steps, Q&A

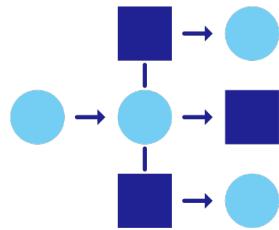
# Microservice Development

## Current Landscape

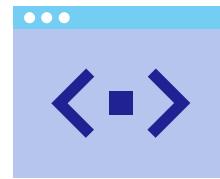
# State of enterprise developers



Deploying scale-out apps for flexibility, cost, and efficiency



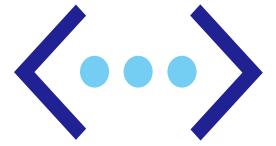
Developing resilient, scalable, microservice-based apps that interact with services



Focusing on building applications, not infrastructure



Trending toward serverless platforms with simple code to cloud pipelines



Using multiple languages and frameworks during development

# What is holding back microservice development?



Limited tools and runtimes to build distributed applications



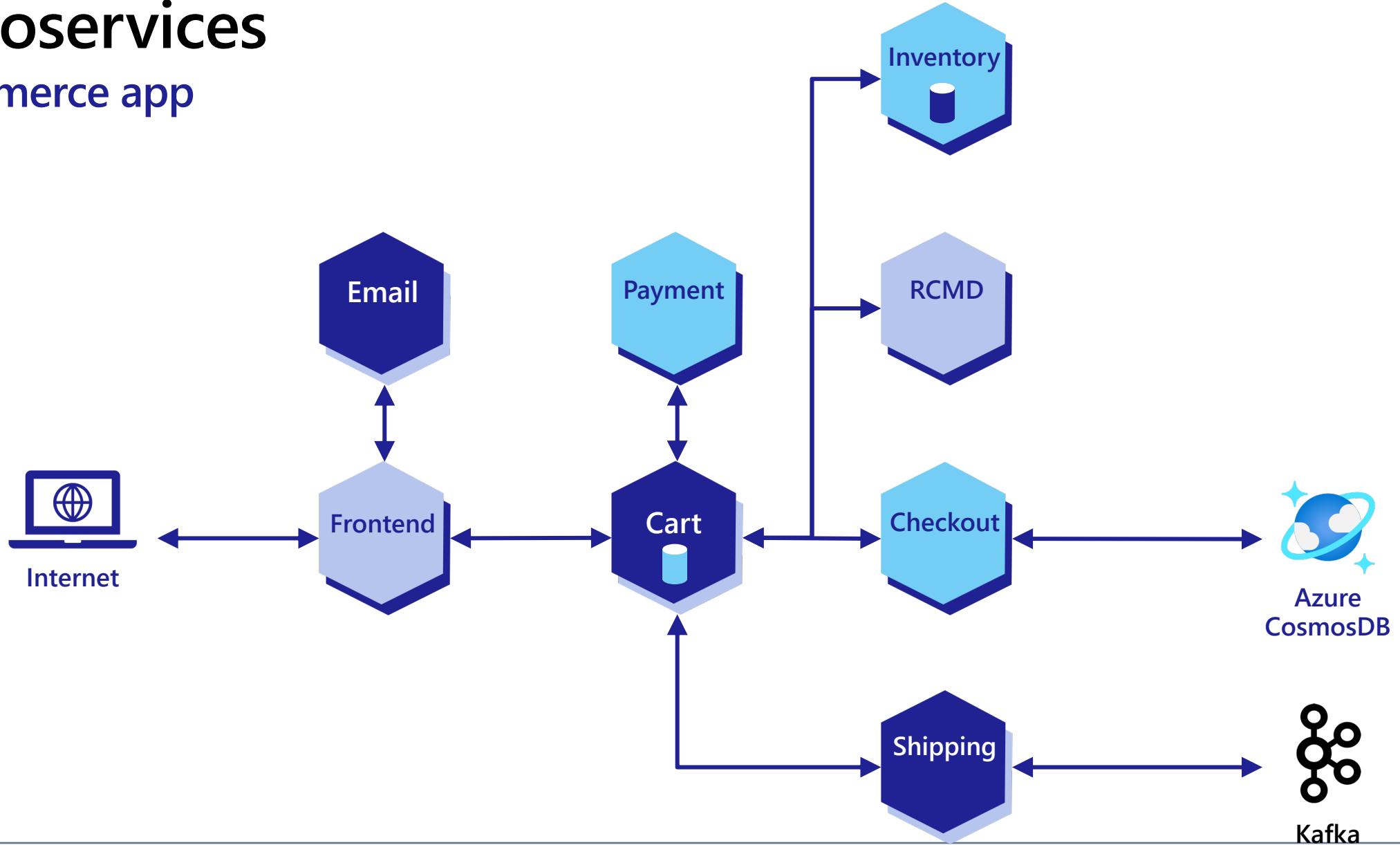
Runtimes have limited language support and tightly controlled feature sets



Runtimes only target specific infrastructure platforms with limited portability

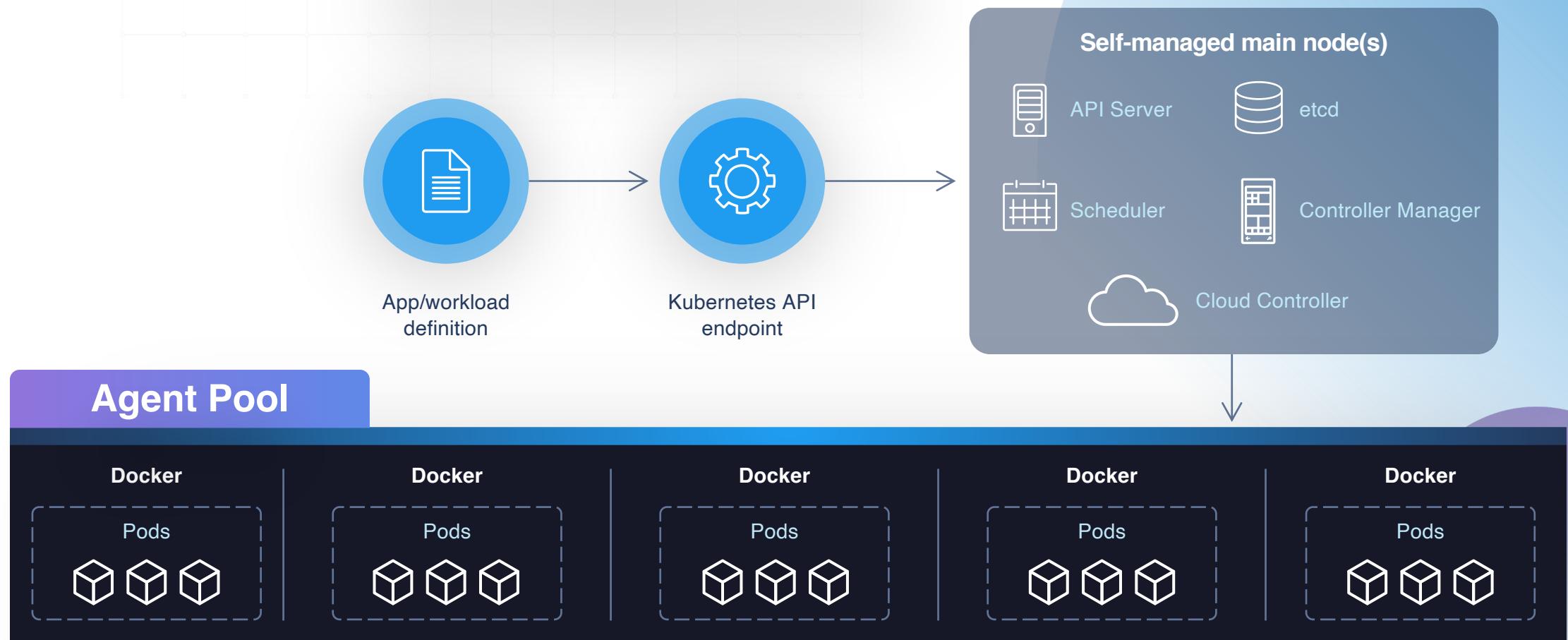
# Microservices

## E-commerce app

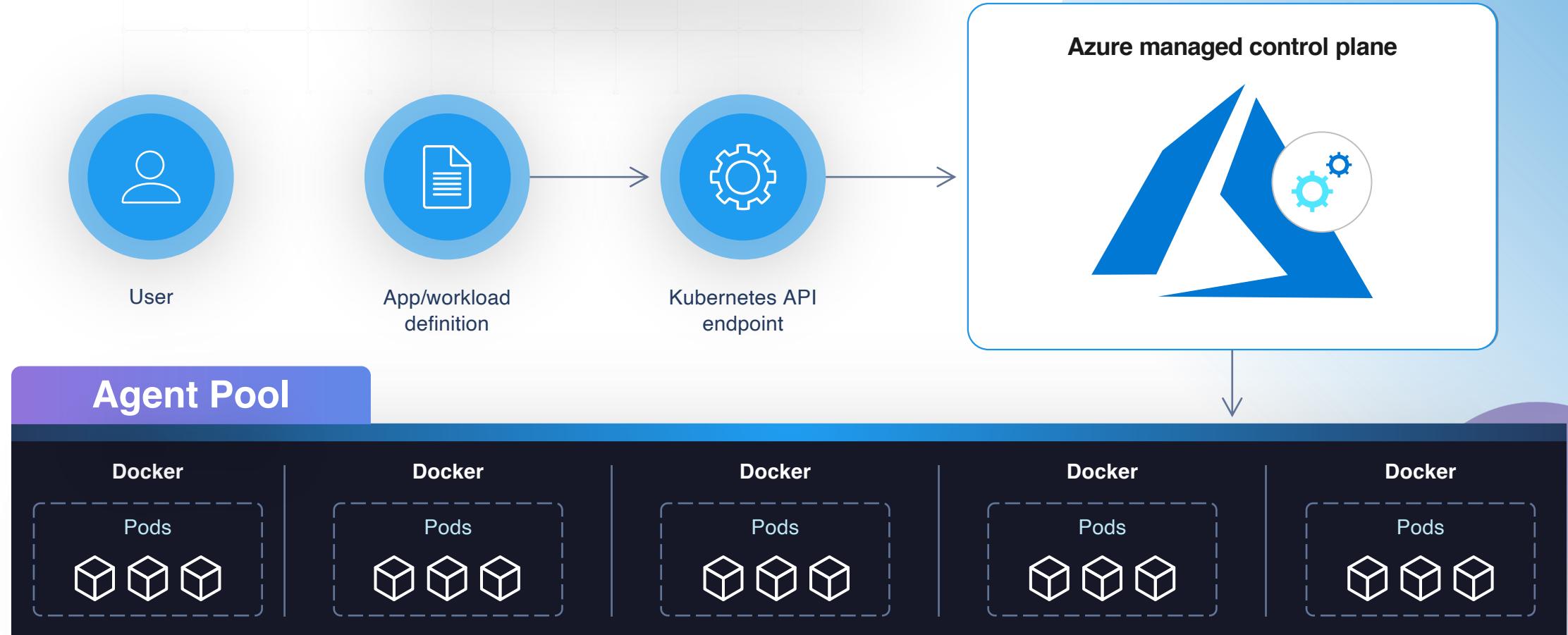


# Microservice Architecture Kubernetes

# Kubernetes Architecture



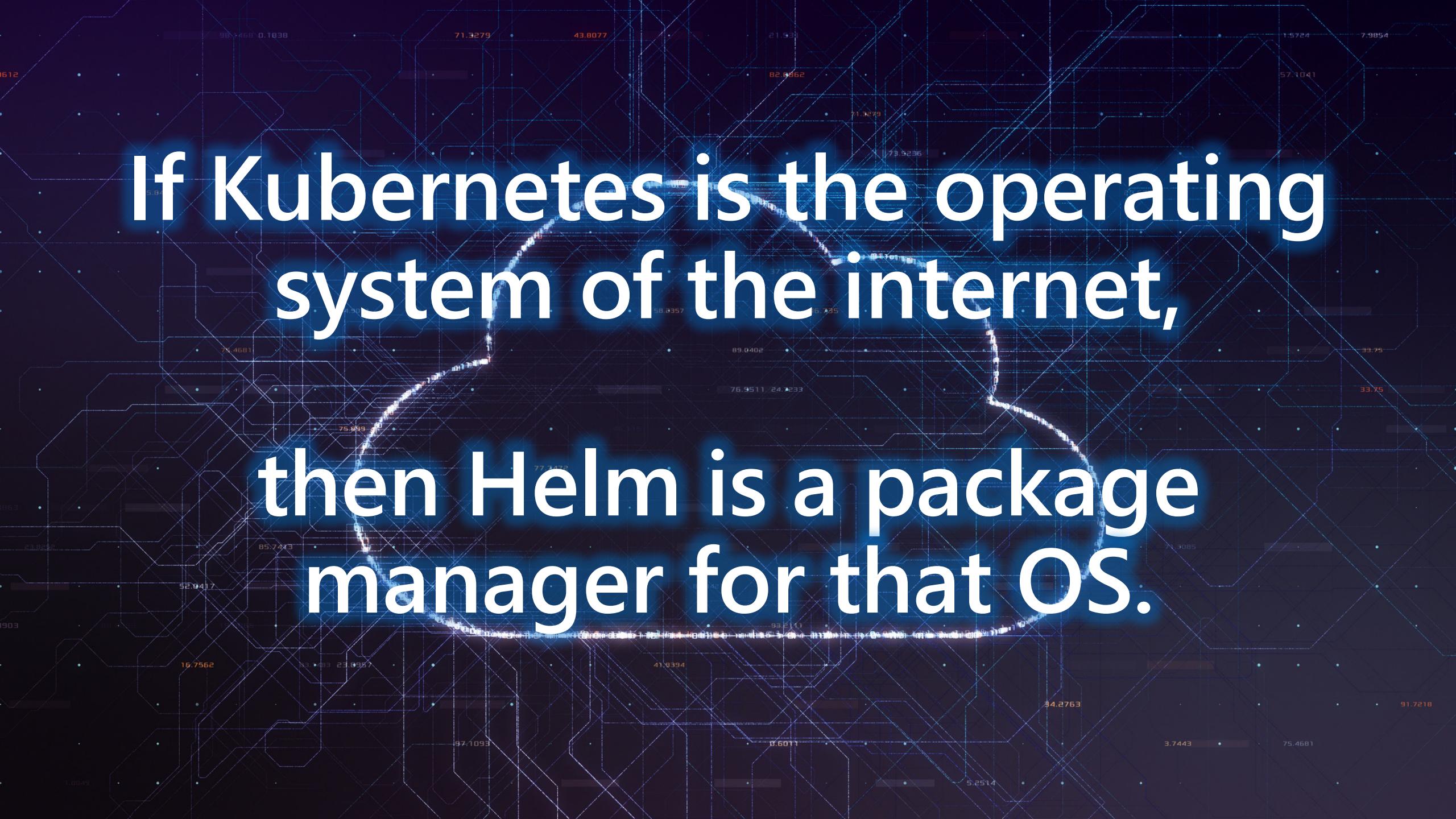
# Kubernetes Architecture



# How do we **simplify**? I'm so glad you asked.

# Helm

## Manage Kubernetes



If Kubernetes is the operating  
system of the internet,  
then Helm is a package  
manager for that OS.

# Why use Helm?

---



**Helm automatically maintains a versioned history of your releases**

- If something goes wrong, it's as simple as running `helm rollback`



**Combined with CI/CD, you can easily integrate actions into your pipeline**

- Before installation begins
- After an upgrade has finished



# Benefits of using HELM

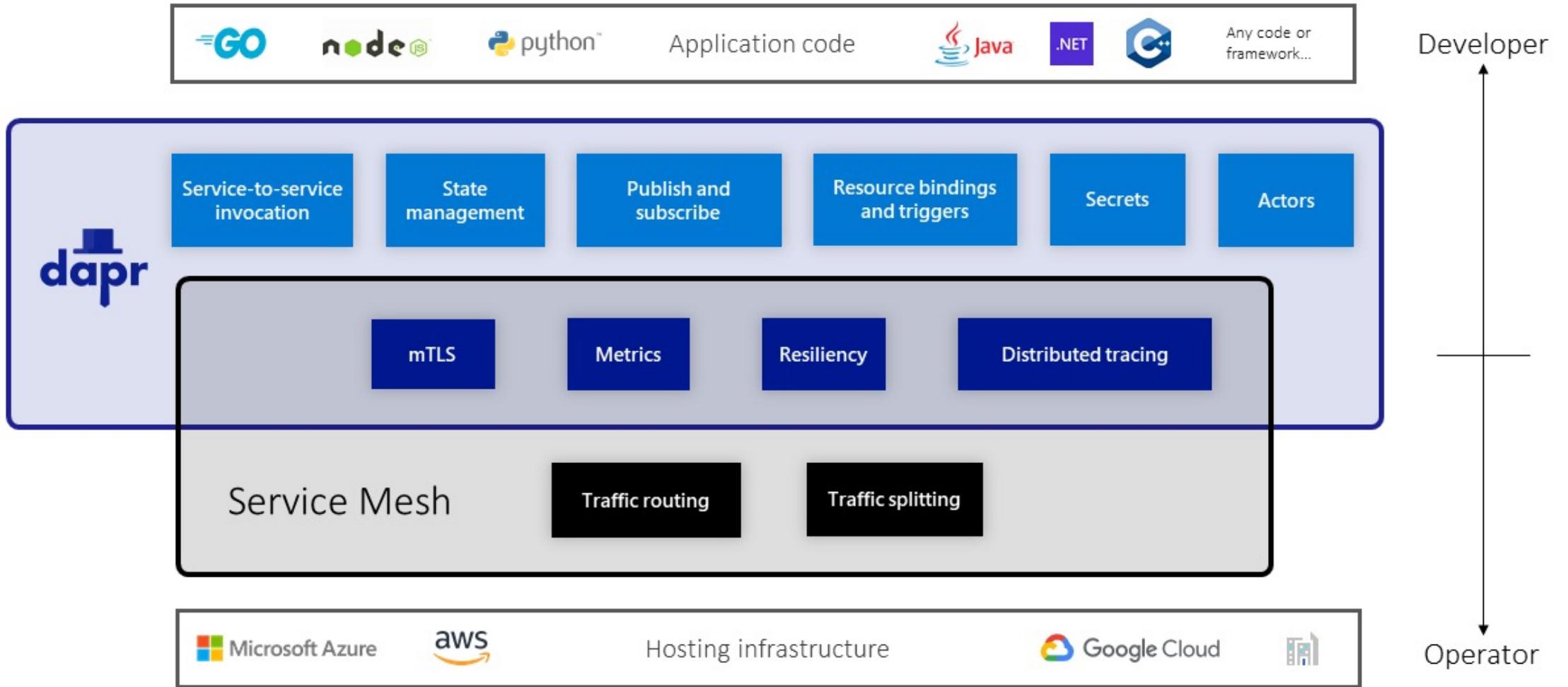


- Simple install and upgrade
- Tracks versions and changes
- Easy rollbacks
- Distribute dependencies
- Templating with dynamic values

# Dapr

## Distributed Application Runtime

*(Not another service mesh)*



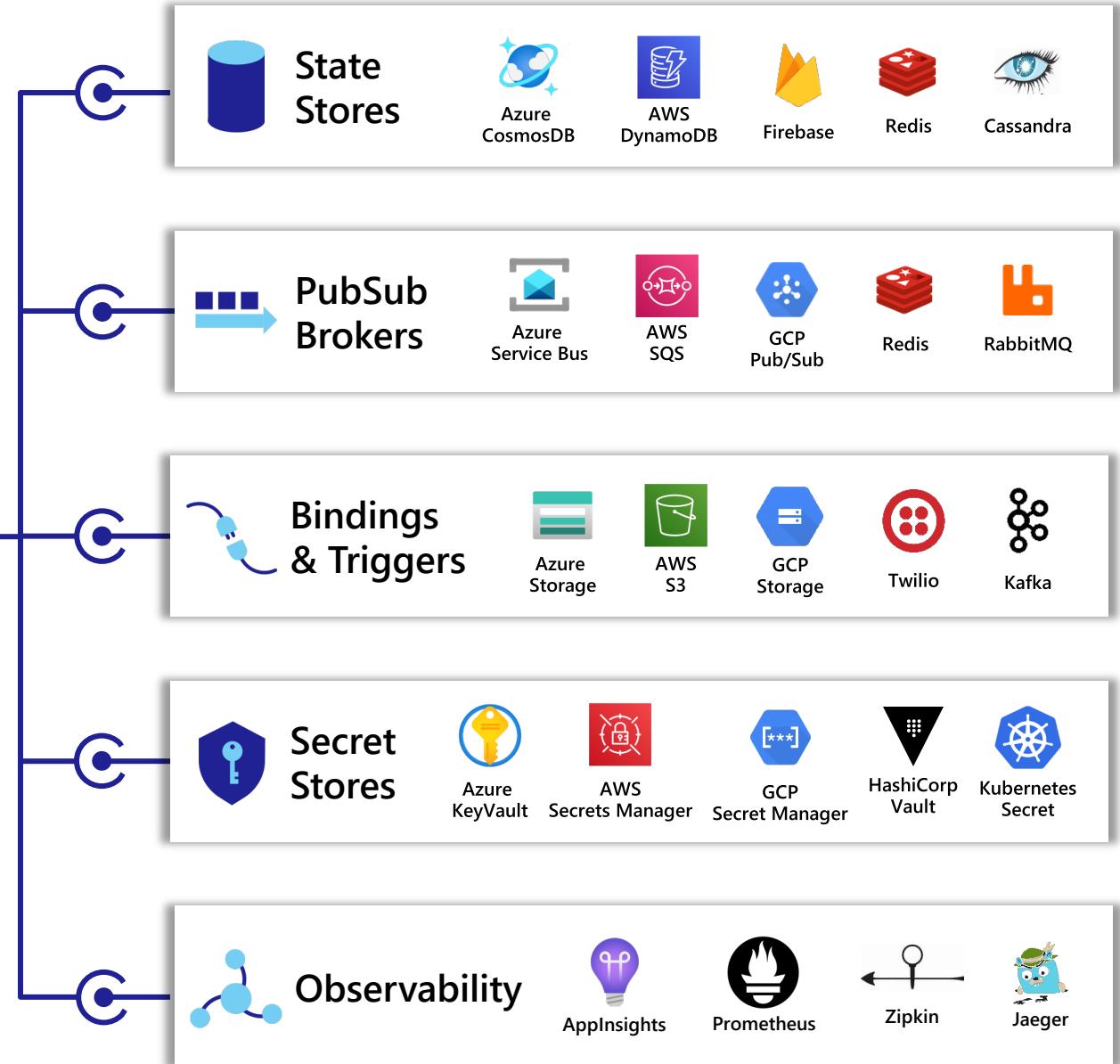
# Dapr Components



Swappable YAML files with  
resource connection details

Over 80 components available

Create components for your resource at:  
[github.com/dapr/components-contrib](https://github.com/dapr/components-contrib)



# Dapr Goals



Best-practices building blocks



Any language or framework



Consistent, portable, open APIs



Adopt standards



Extensible and pluggable components



Platform agnostic cloud + edge



Community driven, vendor neutral

# Bridge to Kubernetes

## Making Debugging Easy

# Bridge to Kubernetes



## Simplifies microservice development

Eliminates the need to manually source, configure, and compile external dependencies



## Streamlines application development

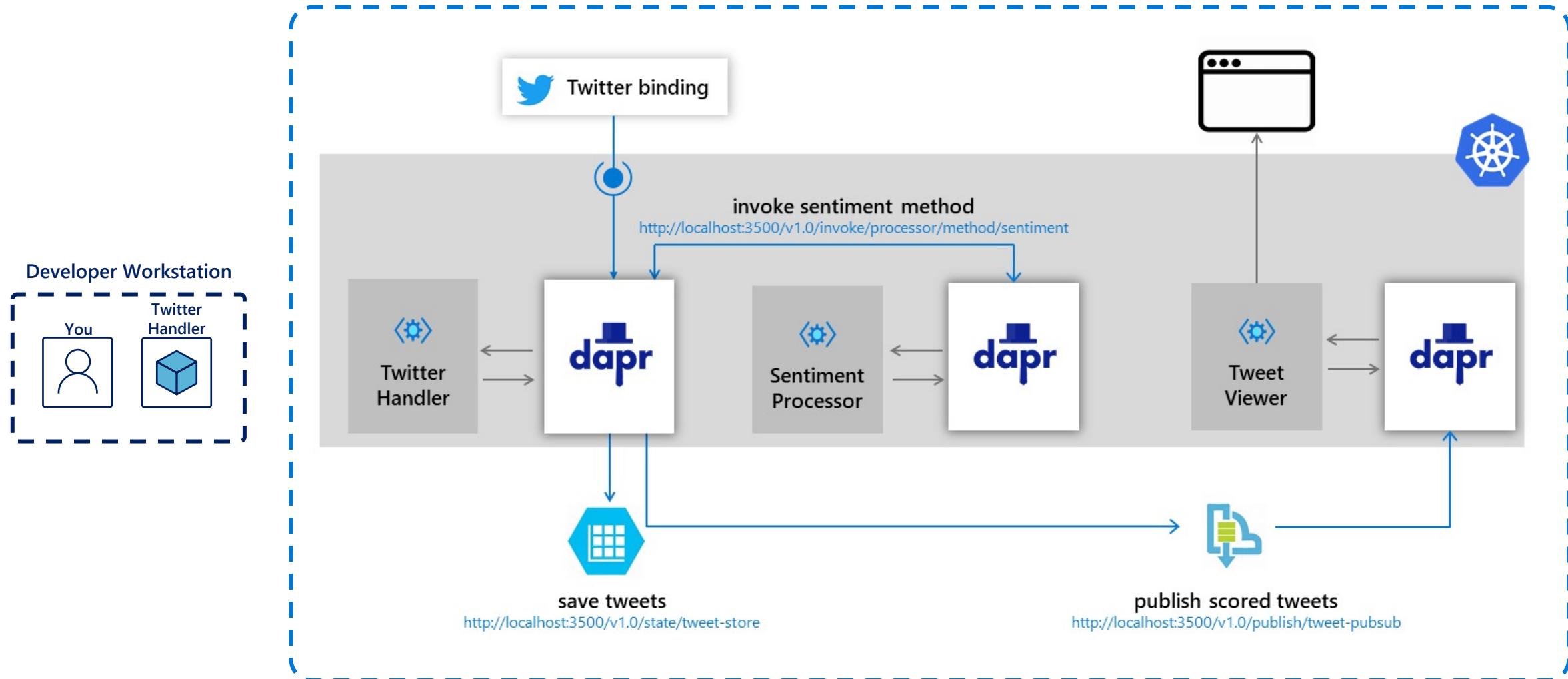
Sidestep operational complexities of building and deploying code into the cluster to test and debug



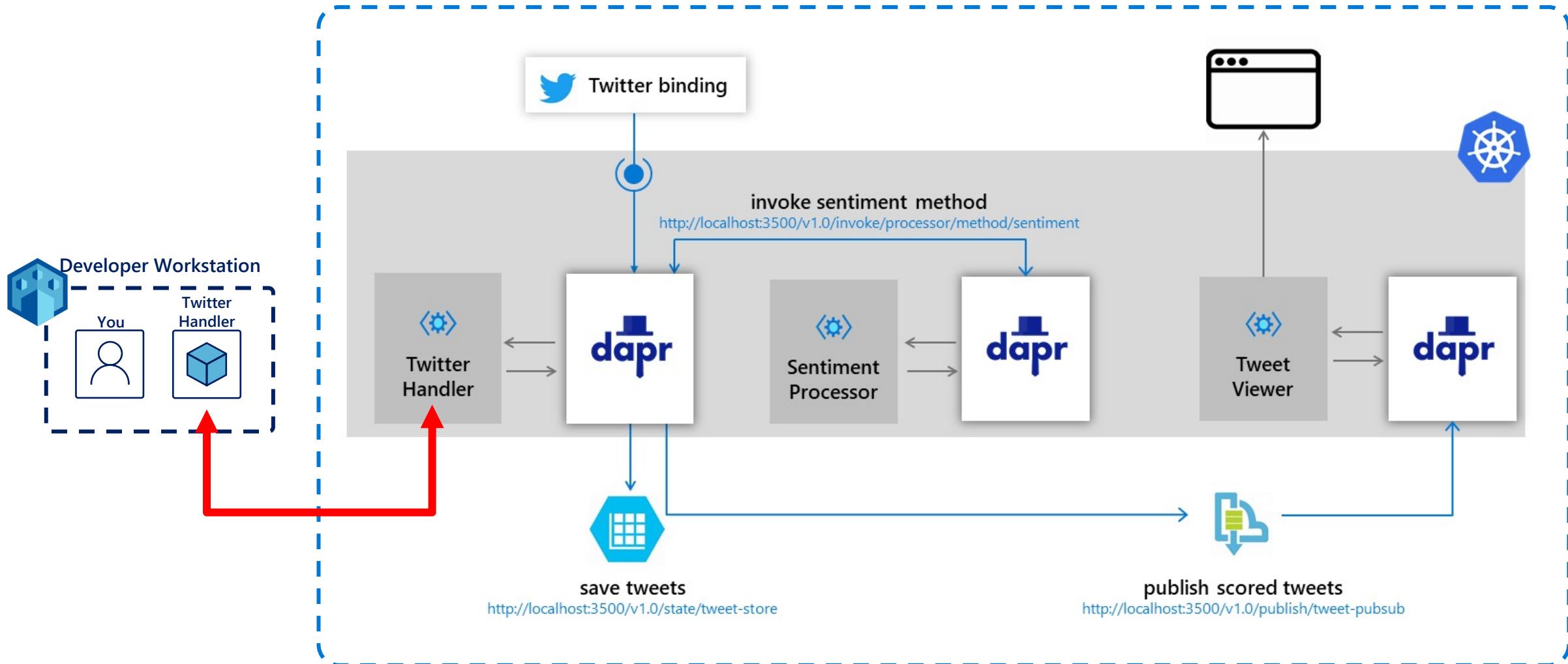
## Work in isolation in shared development environment

Work in a private “sandbox” environment by routing specific traffic locally

# Bridge to Kubernetes, Dapr, and DevOps



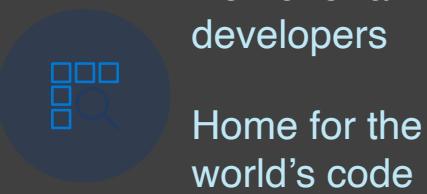
# Bridge to Kubernetes, Dapr, and DevOps



# **GitHub Actions**

## Automate workflows from idea to production

# End-to-end, code-to-cloud DevOps



Home for all  
developers  
  
Home for the  
world's code

- Elastic, to any scale
- Fully managed
- Packages always the latest
- Supports all OS for CI/CD
- Largest ecosystem
- Community-led automation



Deploy anywhere,  
including your own  
data centers

- On-prem
- Azure
- AWS
- Google Cloud Platform

# GitHub Actions



Automate



Build, test and deploy with  
confidence



Customizable

The screenshot shows a GitHub pull request page for the repository 'pied-piper-inc/fib-module'. The pull request is titled 'Create nodejs.yml #22' and is from the user 'jeremyepling'. It has 1 commit and 3 checks. The checks status is 'Some checks haven't completed yet' with 3 in progress checks: 'Node CI / build (8.x) (push)', 'Node CI / build (10.x) (push)', and 'Node CI / build (12.x) (push)'. A note says 'This branch has no conflicts with the base branch'. On the right, there are sections for 'Reviewers', 'Assignees', 'Labels', 'Projects', 'Milestone', and 'Notifications'.

DEMO

# Bridge to Kubernetes | GitHub Actions

## Debug and DevOps

Twitter Handler Updates by jldeen · Zipkin · Fixed spelling error · jldeen/samples · Pull requests · Issues · Marketplace · Explore

github.com/jldeen/samples/pull/16

Search or jump to... / Pull requests Issues Marketplace Explore

jldeen / samples forked from dapr/samples Watch 0 Star 0 Fork 72

Code Pull requests 1 Actions Projects Security Insights Settings

# Twitter Handler Updates #16

**Open** jldeen wants to merge 1 commit into main from twitter-kubecon

Conversation 0 Commits 1 Checks 1 Files changed 1 +1 -2

jldeen commented 1 hour ago  
No description provided.

Update twitter handler ✓ a7a2c22

github-actions bot commented 1 hour ago  
You can see a private version of the changes made in this pull request here:  
<http://twitter-kubecon.viewer.40.88.250.252.nip.io/>

Reviewers  
No reviews  
Still in progress? Convert to draft

Assignees  
No one—assign yourself

Labels  
None yet

Projects  
None yet

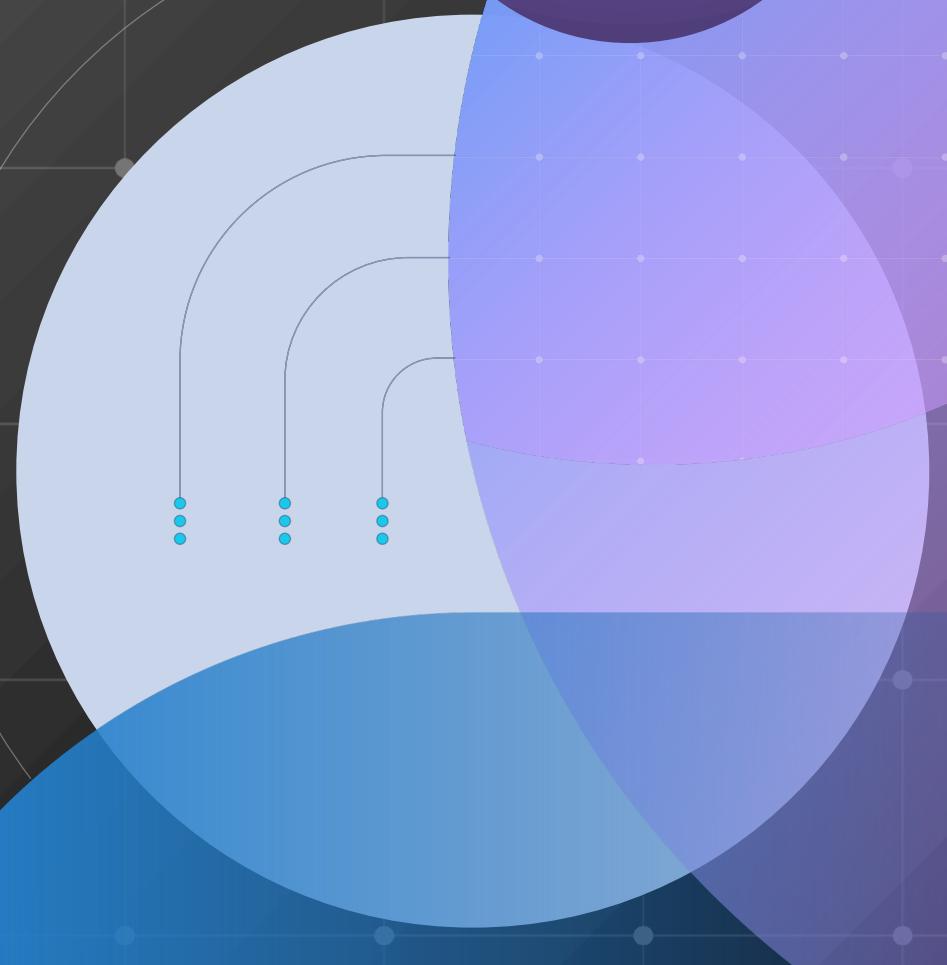
# What did we learn?

Helm: De facto Package Manager for Kubernetes

Dapr: portable, event-driven runtime making development easier

Bridge to Kubernetes: simplifies debugging microservice architecture

GitHub Actions: makes it easy to automate all software workflows



# Q&A and Next Steps



## Checkout Dapr & .NET Session @ 4:30PM PST

Paul Yuknewicz will show tips and tricks to get more gains out the APIs in your existing code, while learning how to use Dapr in microservice communication.



## Bridge to Kubernetes Getting Started

[aka.ms/jldeennetconf2021](http://aka.ms/jldeennetconf2021)



## Follow me on Twitter!

@jldeen

# Thank you!