May13th, 2023

#GlobalAzureAthens

BY COMMUNITY • FOR COMMUNITY

#GLOBALAZURE

2023

#GlobalAzureAthens

Distributed Applications …simplified

Vasilis Zisiadis
Cloud Solutions Architect App Innovation @Microsoft

Nikolaos Antoniou
Cloud Solutions Specialist App Innovation @Microsoft

Dear Global Azure Athens 2023 sponsors,
your support made all the difference — **thank you!**

kaizen GAMING

uni.systems

Microsoft

Info Quest TECHNOLOGIES

Office line envision . empower . evolve

CUBE

CANDI ADVANCED BUSINESS AND DIGITAL SOLUTIONS

BlueStream SOLUTIONS

INFOLAB Enterprise Training

Code.Hub

SIGNAL™

BY COMMUNITY • FOR COMMUNITY • #GLOBALAZURE 2023

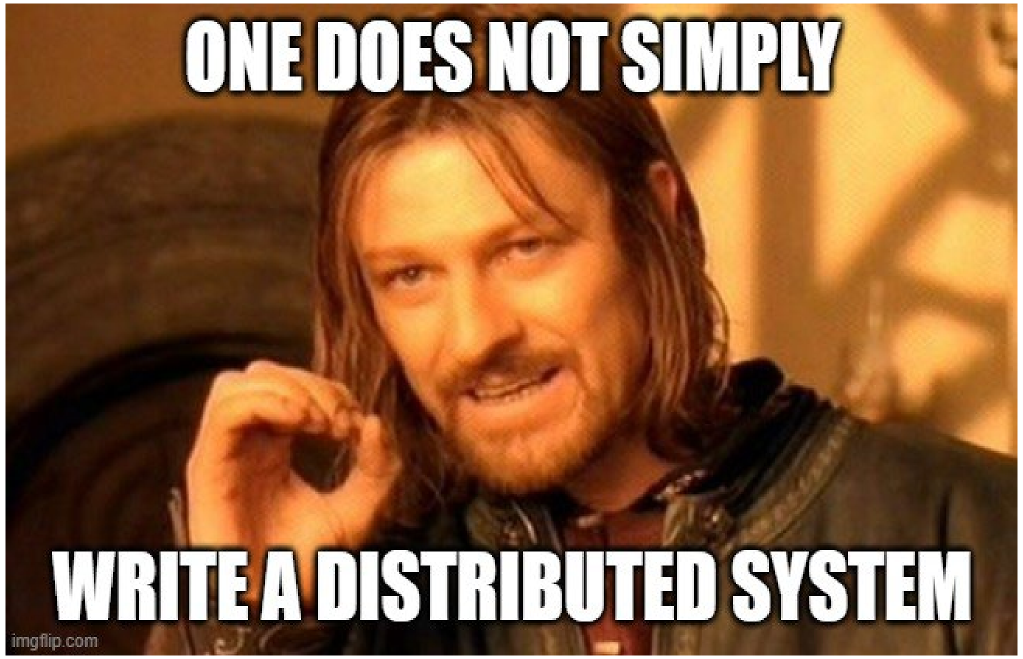
#GlobalAzureAthens

dapr

Distributed Application Runtime

Portable, event-driven, runtime for building distributed applications across cloud and edge

State of Enterprise Developers



- Being asked to develop resilient, scalable, microservice-based apps
- They write in many languages
- They want to leverage existing code

What is holding back microservices development?

Frequently need to incrementally migrate from existing and legacy code

Runtimes have narrow language support with tightly controlled feature sets

Runtimes don't have composable and incrementally adoptable equivalents that can run anywhere

Introducing Dapr

A portable, event-driven, serverless runtime for building distributed applications across cloud and edge

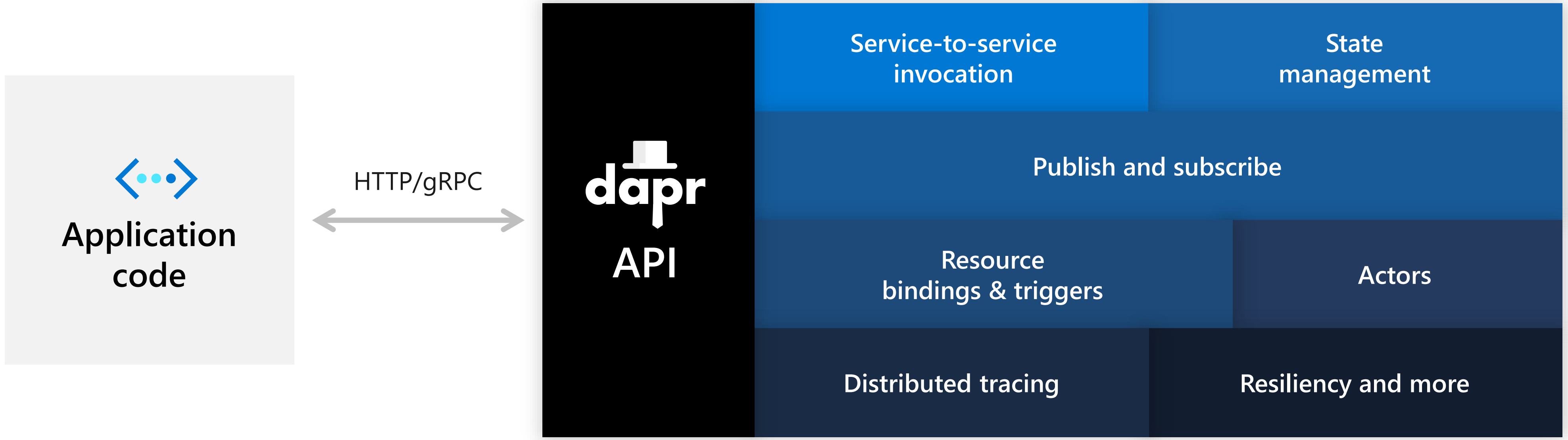
Microservice Building Blocks	Make it easy for developers to create microservice applications without being an expert in distributed systems, including migrating existing code
Sidecar Architecture	Developer first, standard APIs used from any programming language or framework
Cloud + Edge	Runs on multiple environments for cloud, on-prem, and small-edge including any Kubernetes

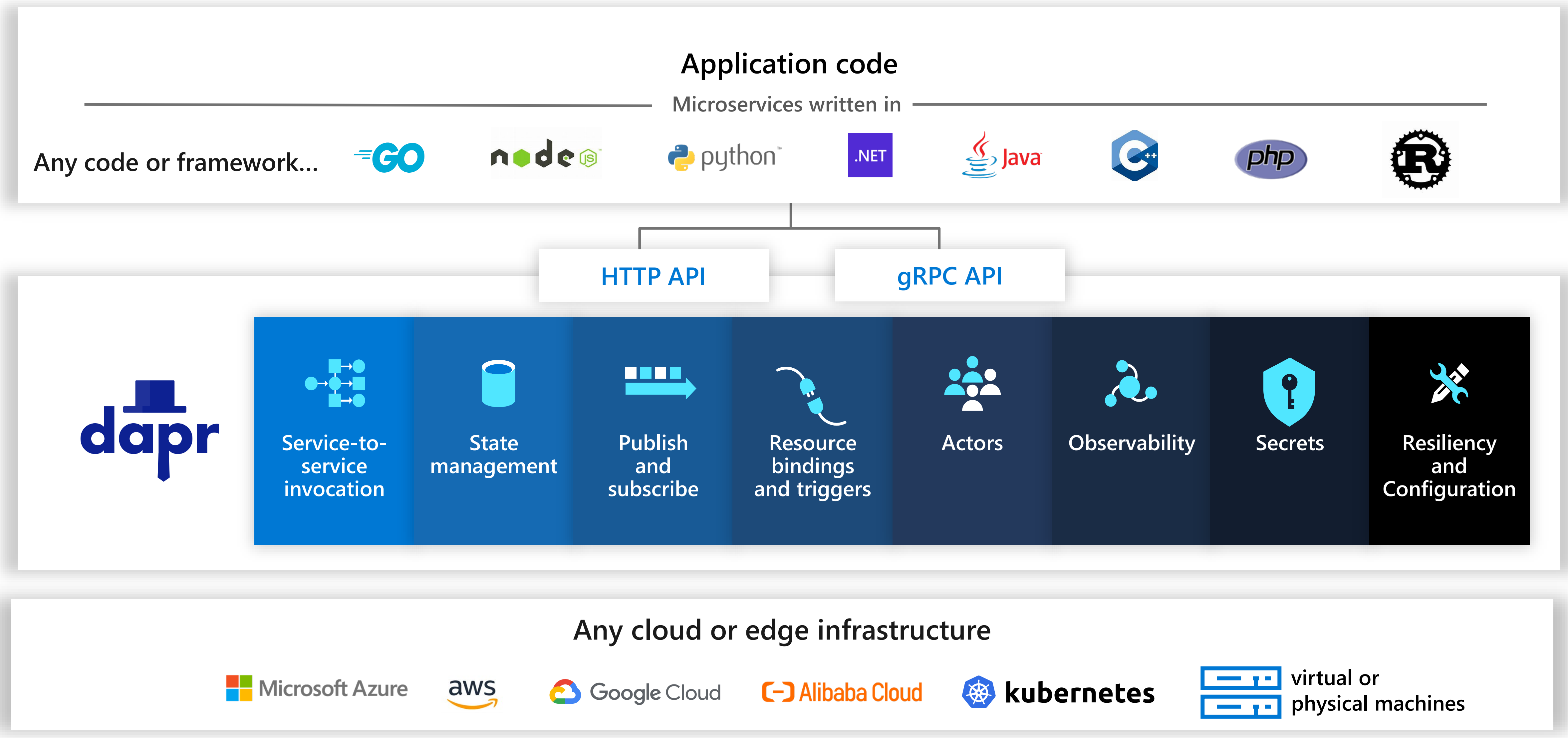
Sidecar architecture

Standard APIs accessed over http/gRPC protocols from user service code

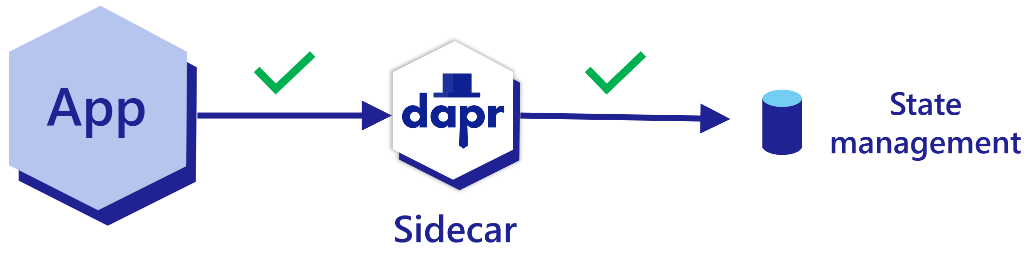
`e.g. http://localhost:3500/v1.0/invoke/myapp/method/neworder`

Dapr runs as local “side-car library” dynamically loaded at runtime for each service

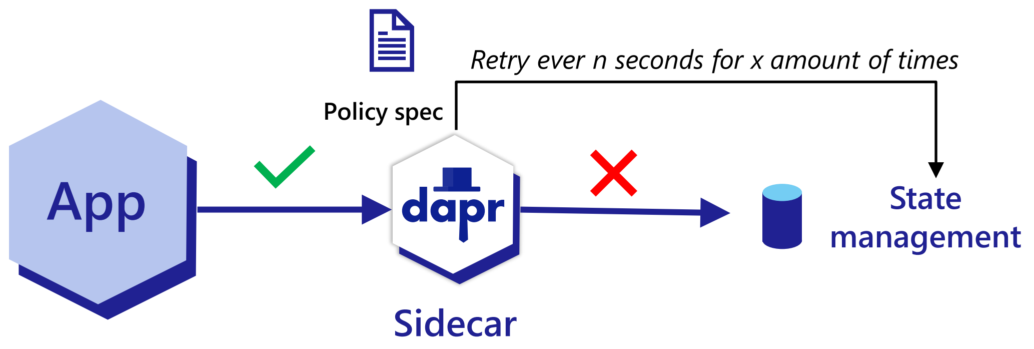



Application code
Microservices written in
Any code or framework...
GO
node
python
.NET
Java
php
HTTP API
gRPC API
dapr
Service-to-service invocation
State management
Publish and subscribe
Resource bindings and triggers
Actors
Observability
Secrets
Resiliency and Configuration
Any cloud or edge infrastructure
Microsoft Azure
aws
Google Cloud
Alibaba Cloud
kubernetes
virtual or physical machines

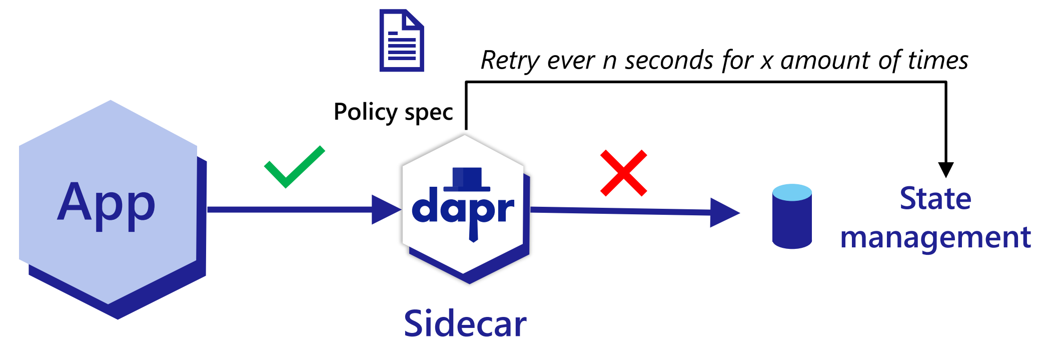
Resiliency



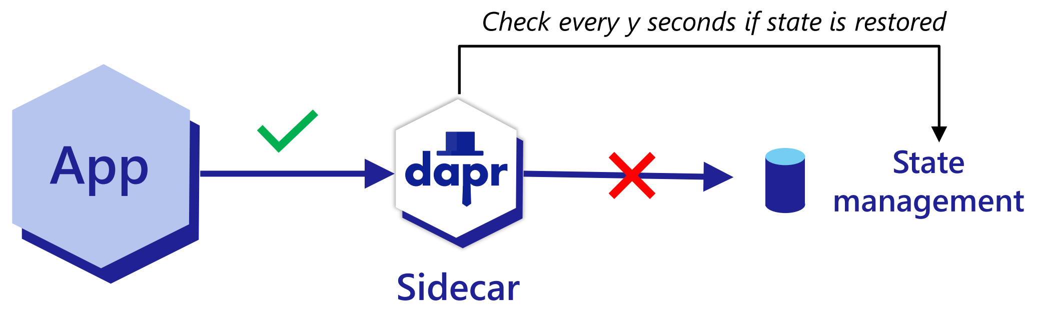
Resiliency


Policy spec
Retry ever n seconds for x amount of times
App
dapr
Sidecar
State management

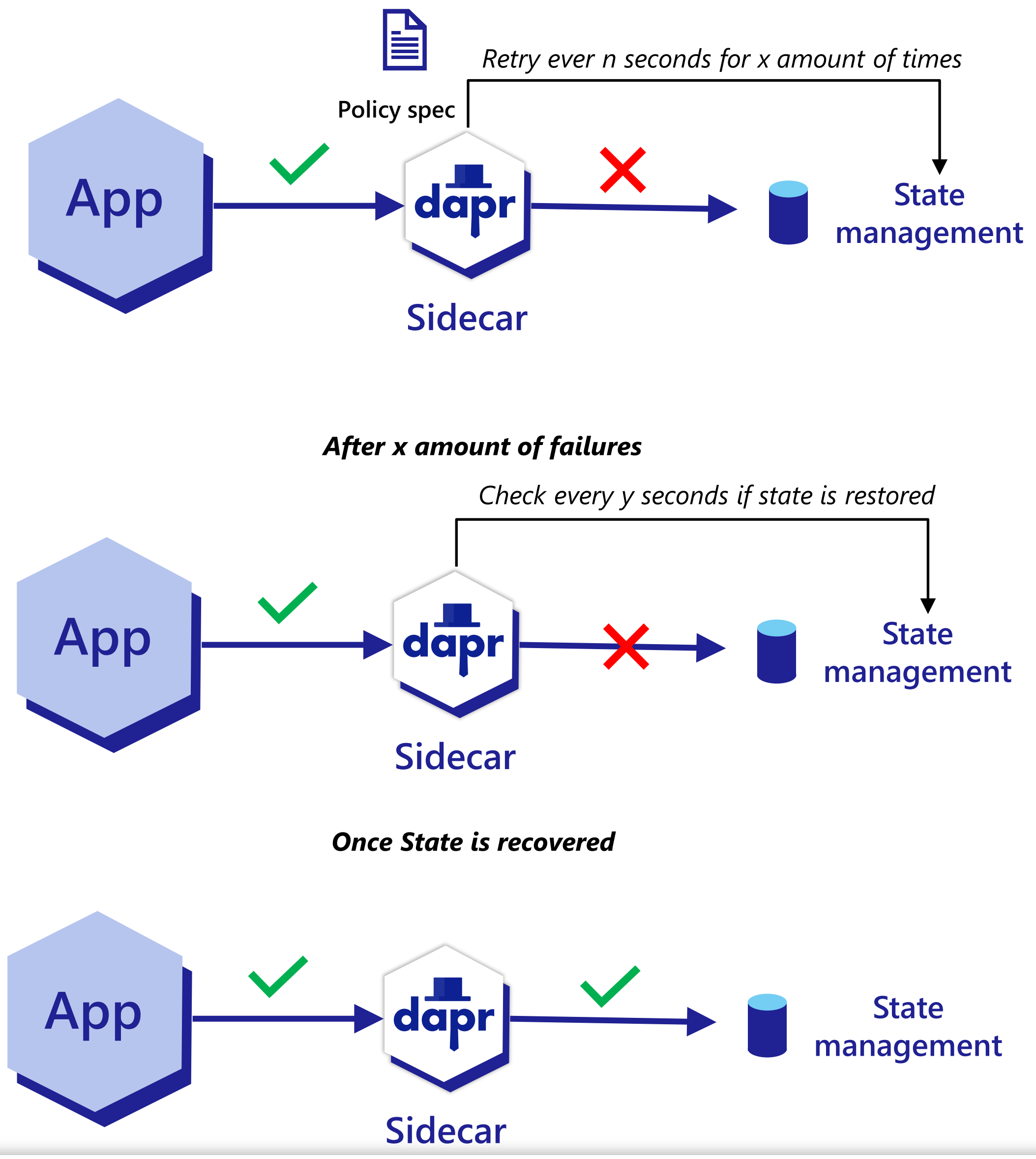
Resiliency



After x amount of failures

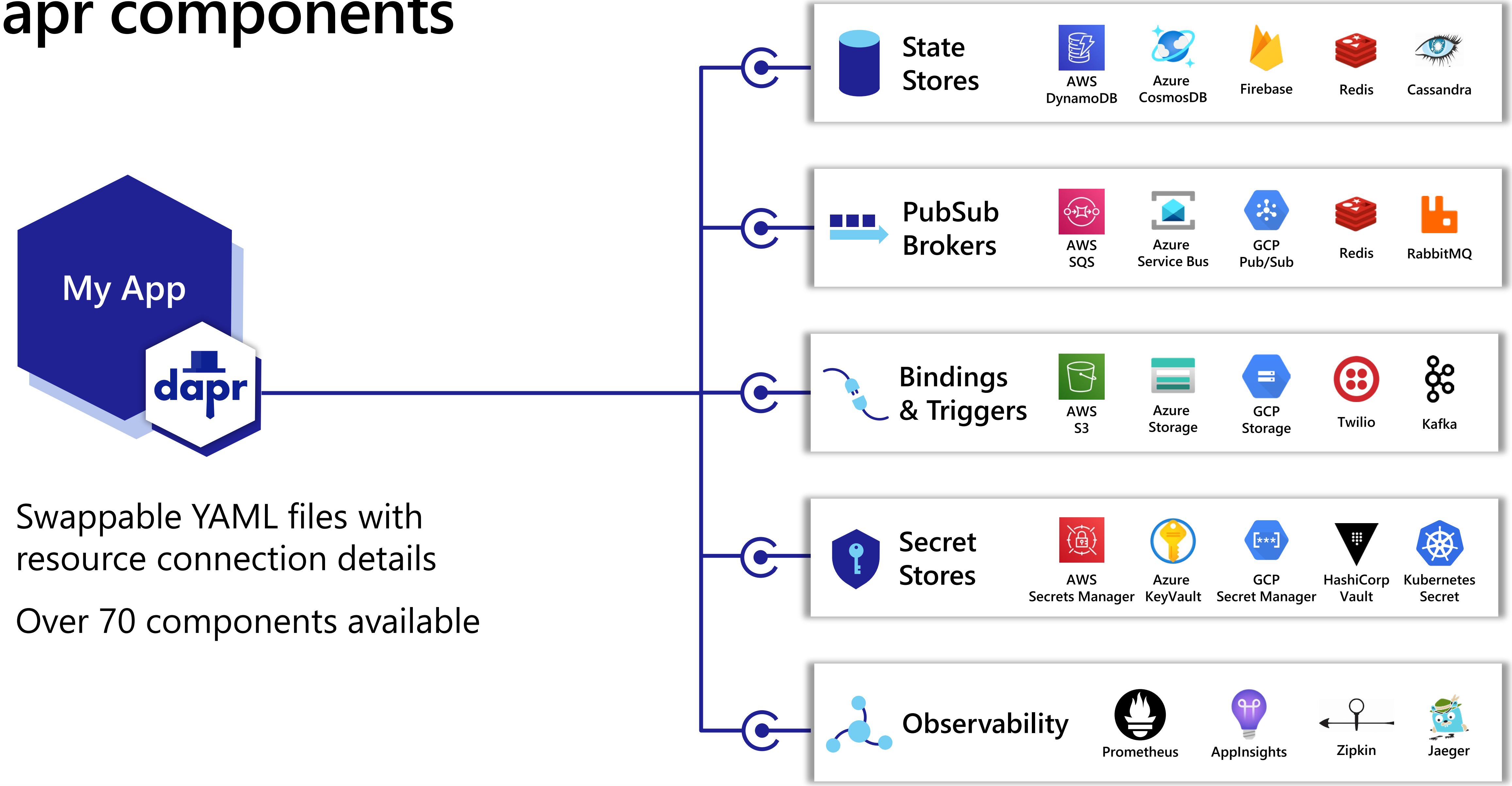


Resiliency



Policy spec
Retry ever n seconds for x amount of times
App
dapr
Sidecar
State management
After x amount of failures
Check every y seconds if state is restored
App
dapr
Sidecar
State management
Once State is recovered
App
dapr
Sidecar
State management

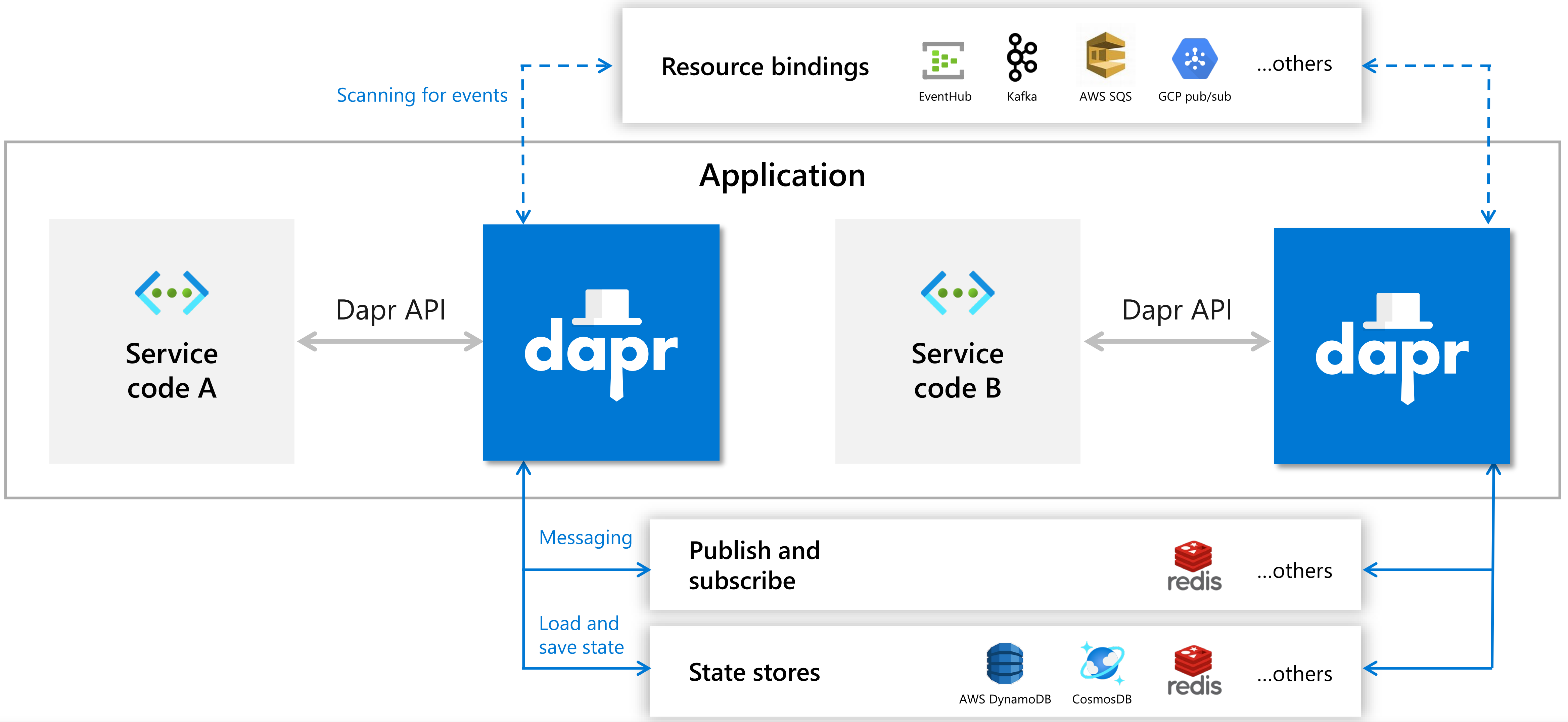
Dapr components



Swappable YAML files with resource connection details

Over 70 components available

Dapr Mechanics



Microservice Building Blocks

State Management

Create long running, stateless and stateful services

Service Invocation & Fault Handling

Perform direct, secure, service-to-service method calls

Resource Bindings

Trigger code through events from a large array of input and output bindings to external resources including databases and queues

Publish & Subscribe

Secure, scalable messaging between services

Actors

Encapsulate code and data in reusable actor objects as a common microservices design pattern

Distributed Tracing & Diagnostics

See and measure the message calls across components and networked services

Azure Container Apps

Serverless containers for microservices

Build modern apps on open source

Focus on apps, not infrastructure

Scale dynamically based on events

Kubernetes KEDA dapr DAPR envoy Envoy

Get in the fast lane with Azure Container Apps!

ACA Landing Zone Accelerator offers architectural guidance, reference architecture, reference implementation and automation packaged to deploy workloads on Azure at scale and aligned with industry-proven practices

Authoritative
Framework for holistic design decisions on Azure

Proven
Based on customer experiences with large-scale Azure migration projects at-scale

Prescriptive
Apply standards to clearly plan and design Azure environments

Enterprise-scale for ACA architecture

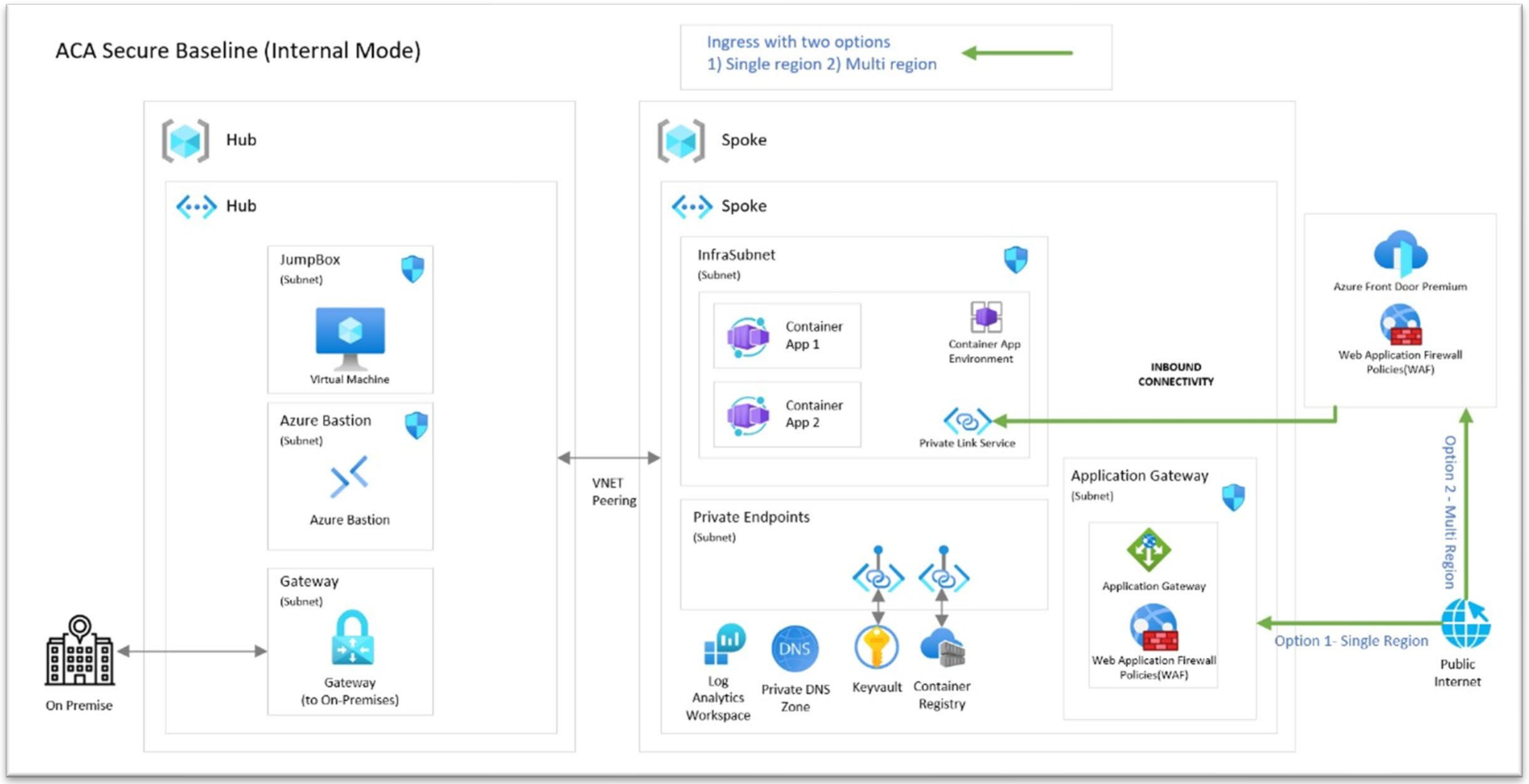
Construction set design guidelines

Guidelines (decisions and recommendations) for the 4 major components of enterprise-scale architecture

Enterprise-scale for ACA reference implementation

Reference implementation of shared (**network, security, identity, and governance**) services—required to construct and operationalize an enterprise-scale landing zone

Enterprise-grade environment in (single digit) minutes



https://aka.ms/aca-lza

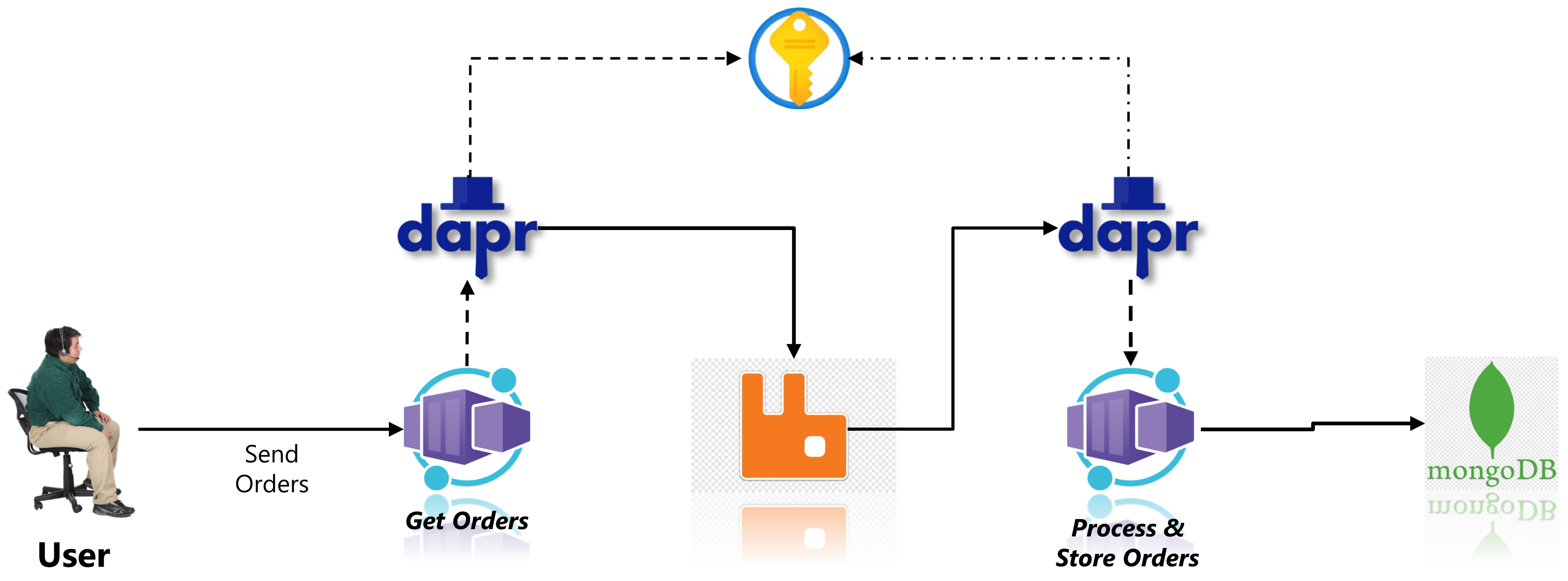
SCAN ME

DEMO

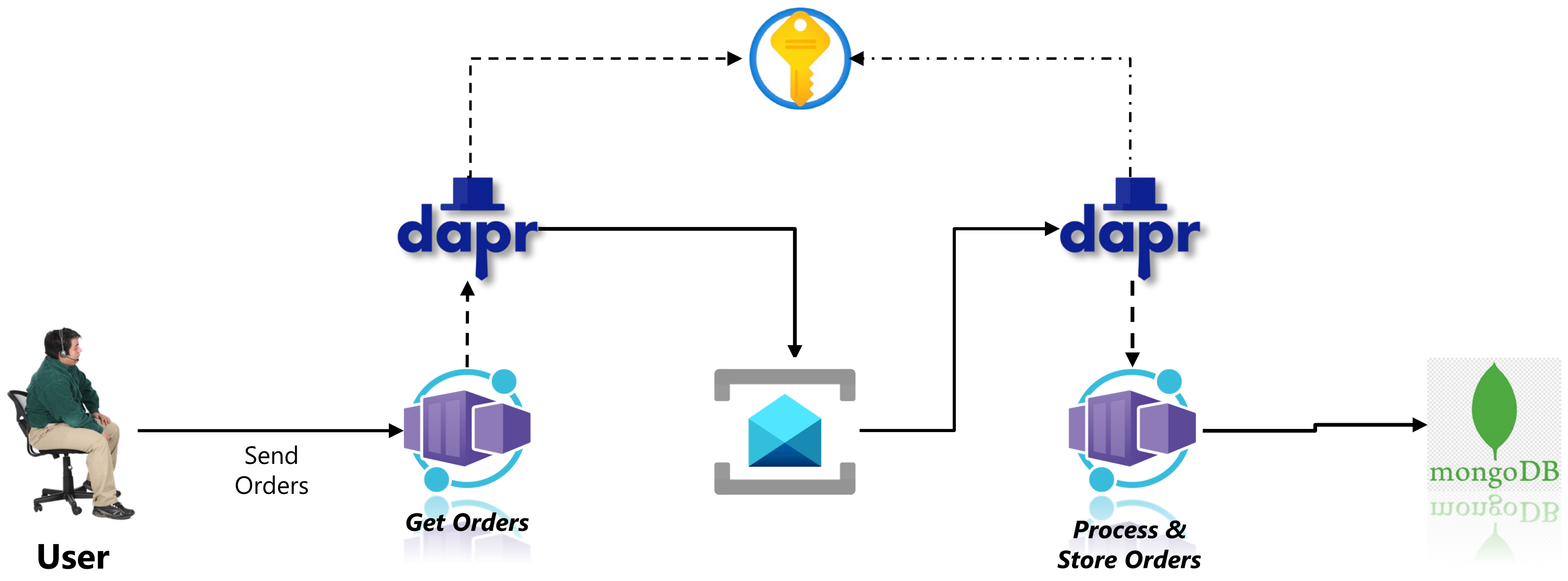
Dapr Invocation and State Management

RabbitMQ –> Service Bus –>Event Hubs
MongoDB –> Cosmos DB

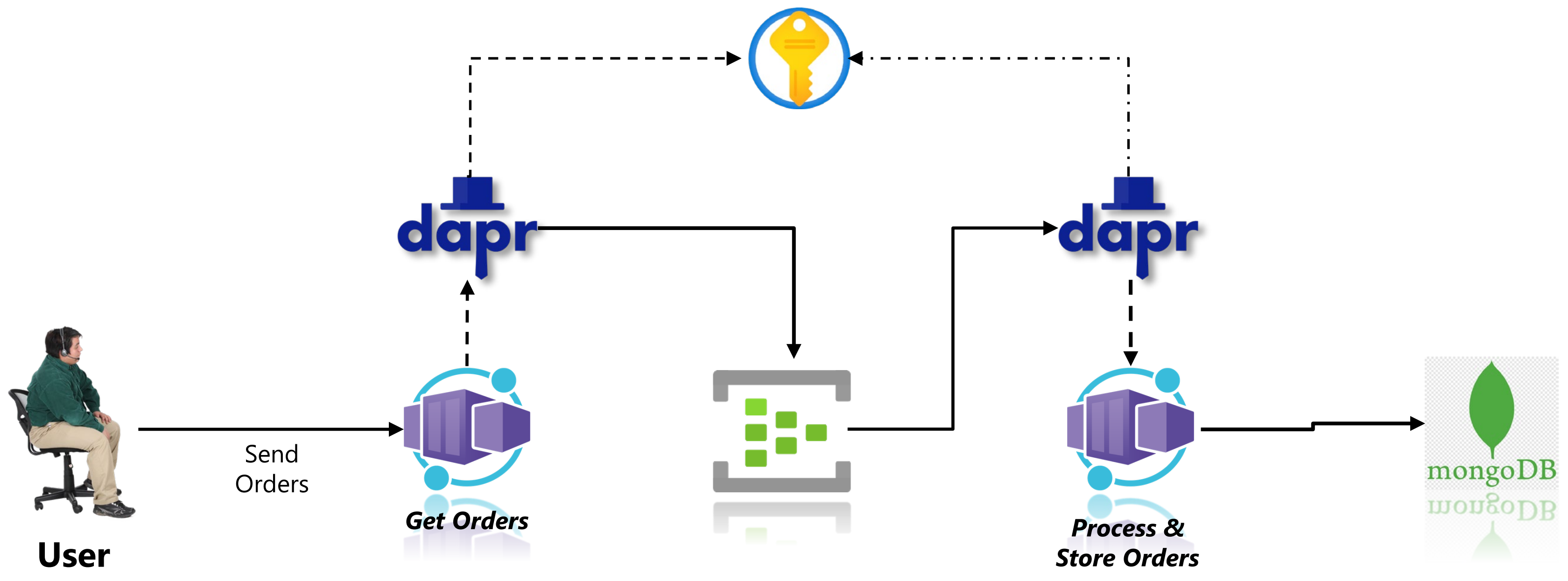
Sample Orders Architecture



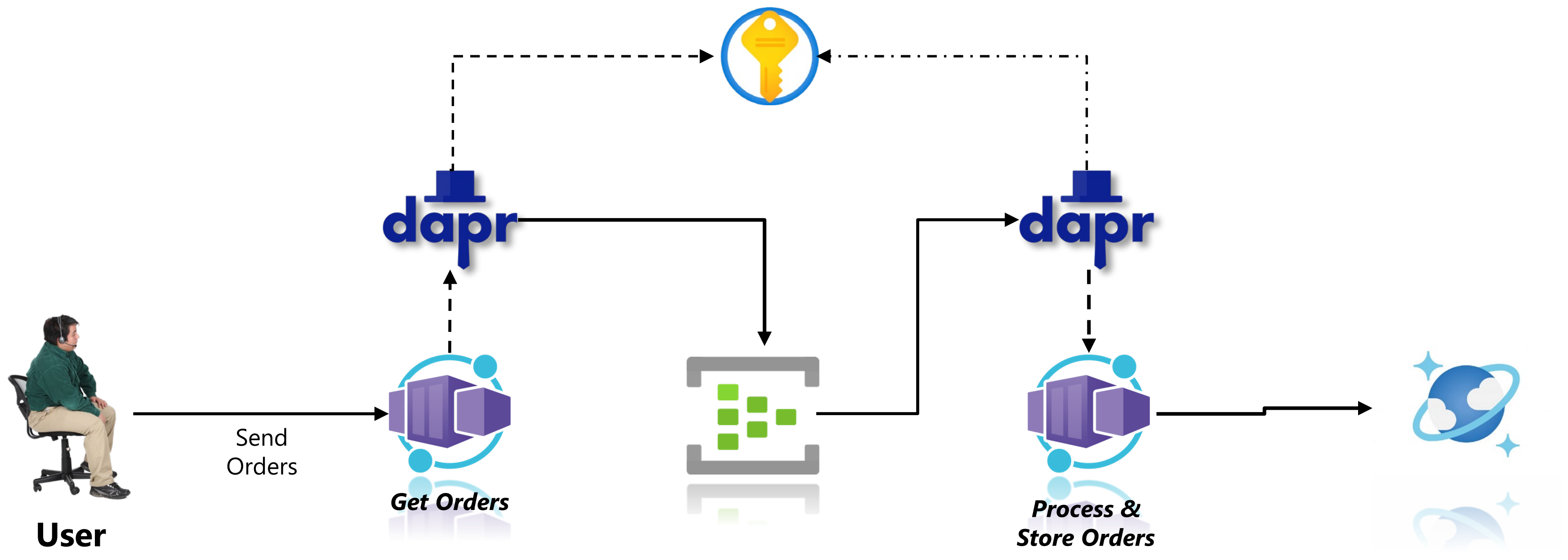
Sample Orders Architecture



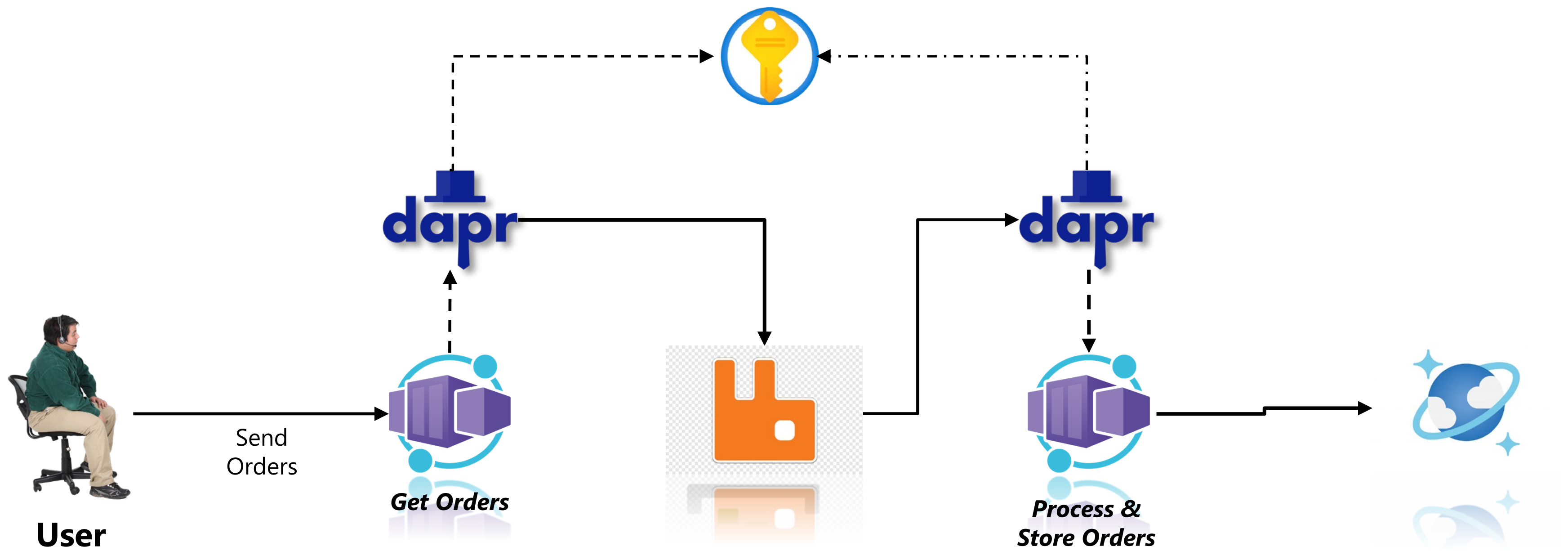
Sample Orders Architecture



Sample Orders Architecture



Sample Orders Architecture



Please evaluate !

A big **thank you** to our sponsors**!**

https://bit.ly/GA23Evaluation

#GlobalAzureAthens

Learn more

Dapr on GitHub

Dapr Docs

Azure Container Apps and Dapr integration

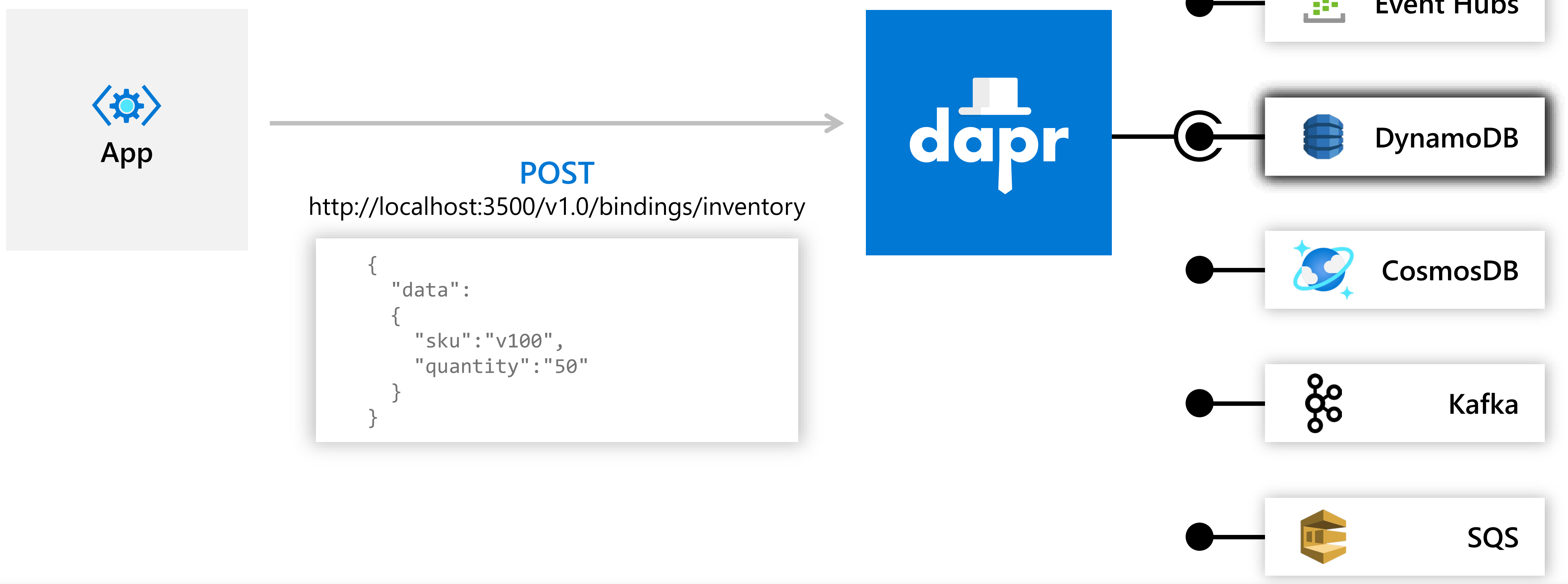
Dapr for .NET Developers

dapr

dapr.io

Appendix

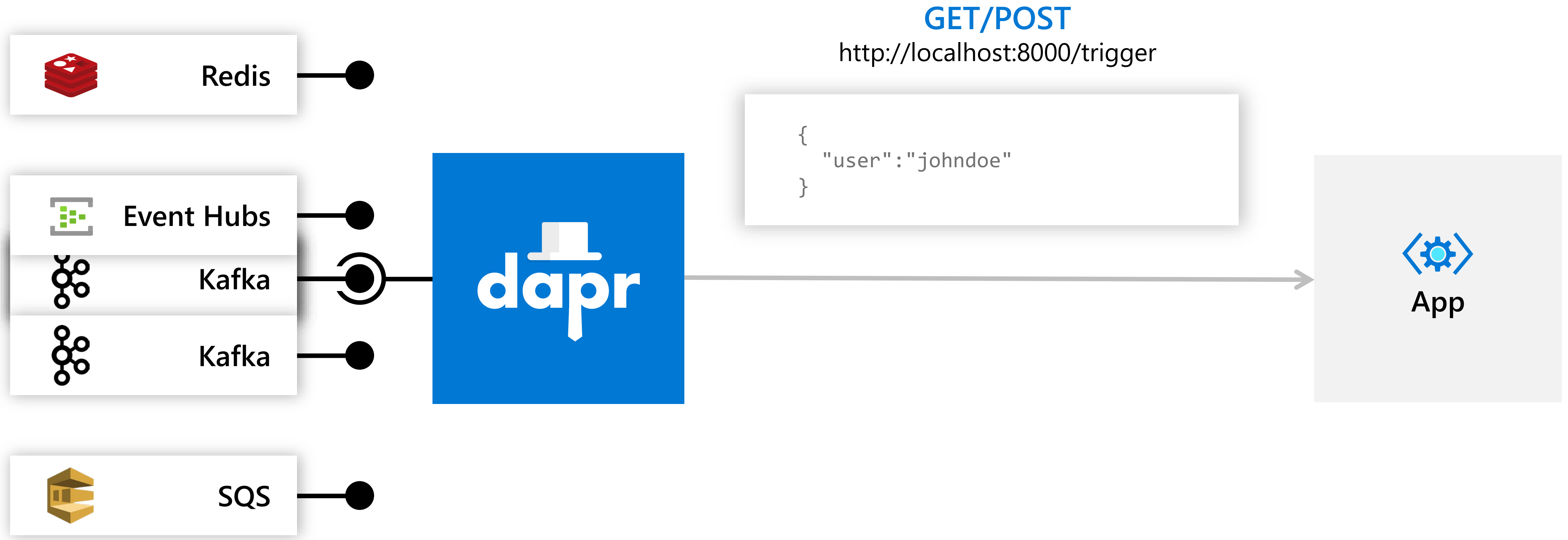
Output bindings



POST
http://localhost:3500/v1.0/bindings/inventory

```
{
  "data":
  {
    "sku":"v100",
    "quantity":"50"
  }
}
```

Input bindings

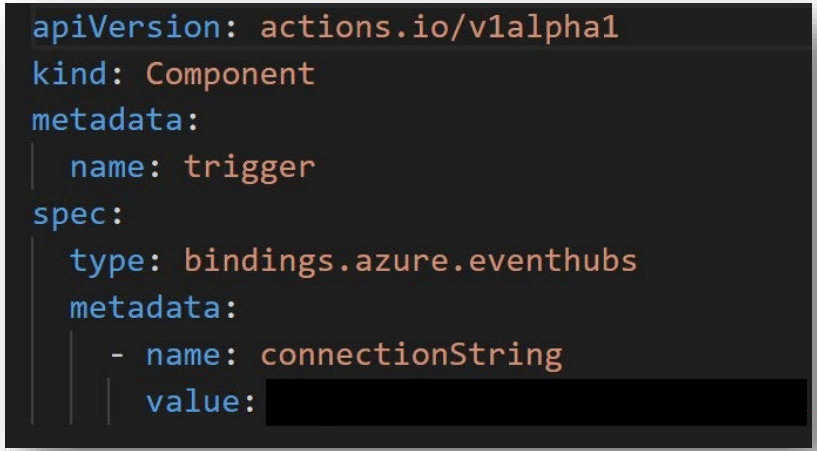


```
{
  "user":"johndoe"
}
```

Resource Bindings

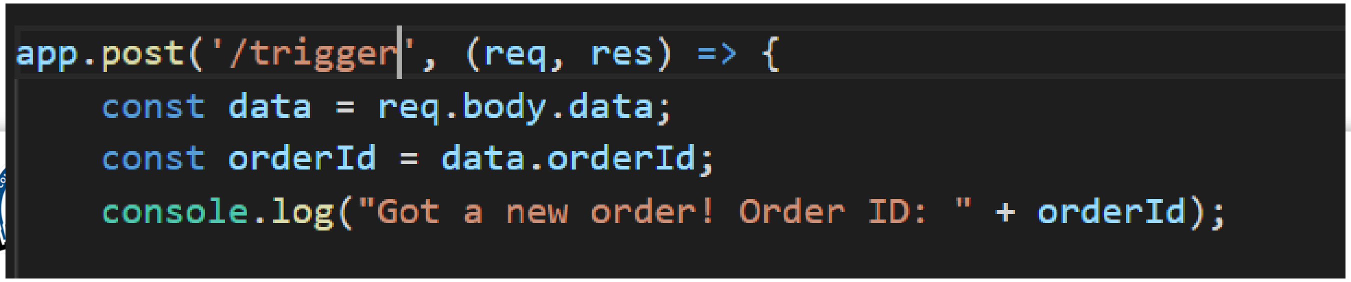
- Dapr enable events to be sent and received from specific resources for any cloud provider
  - Examples: Azure EventHubs, AWS SNS, Google storage

Configure binding

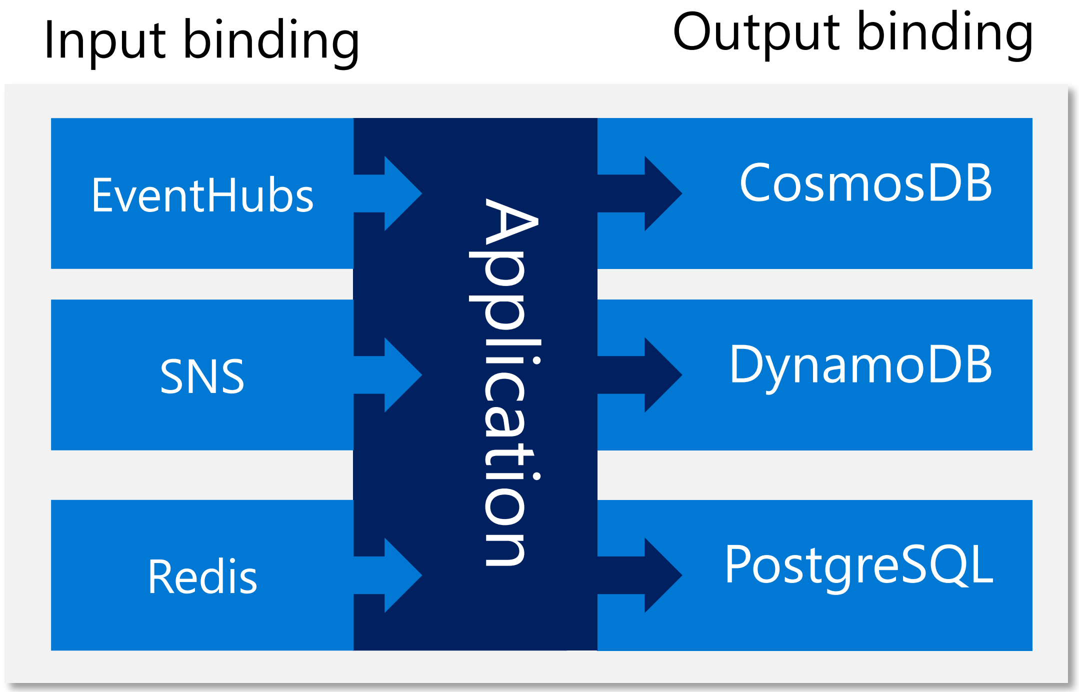


```yaml
apiVersion: actions.io/v1alpha1
kind: Component
metadata:
  name: trigger
spec:
  type: bindings.azure.eventhubs
  metadata:
    - name: connectionString
      value: [redacted]
```

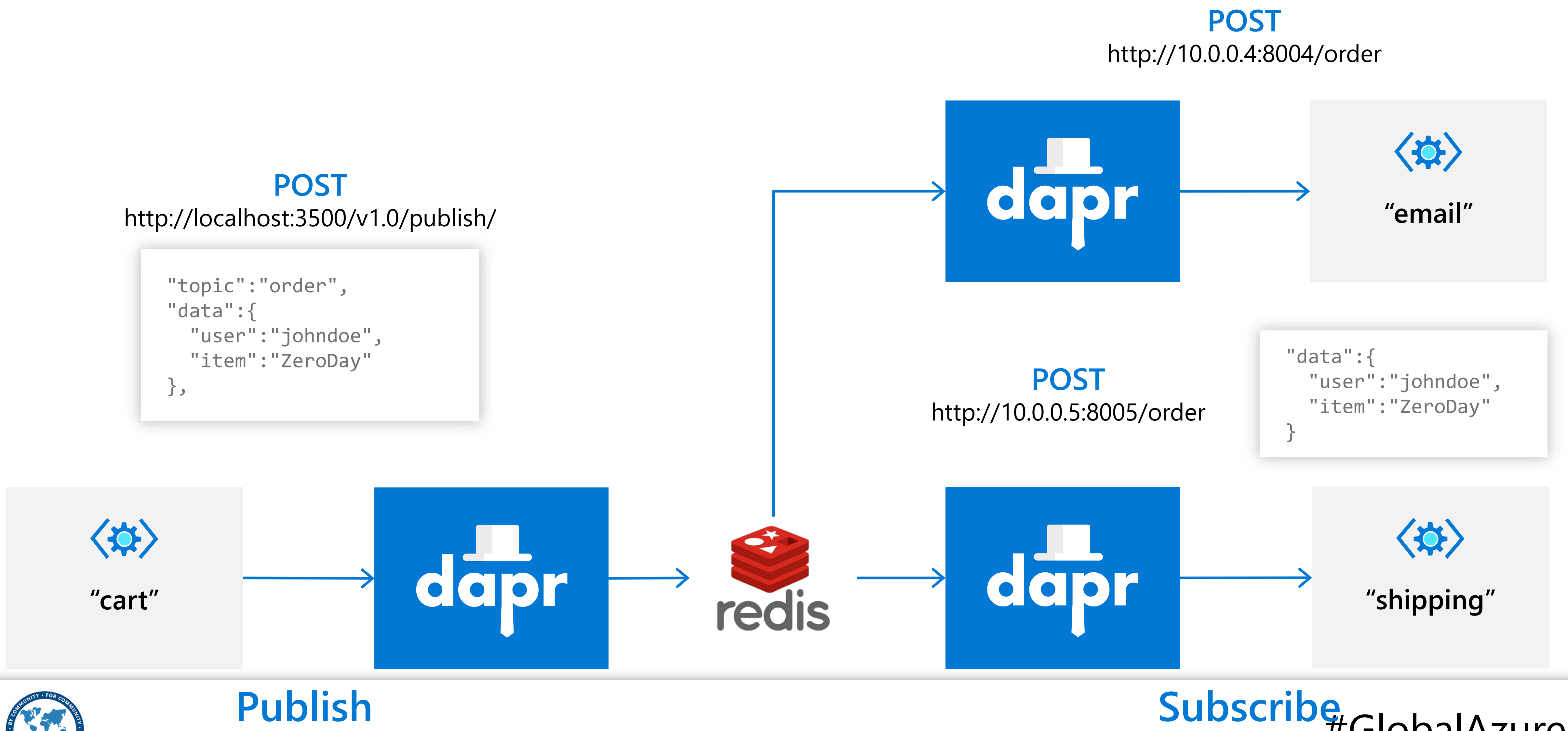
Receive events from binding



```javascript
app.post('/trigger', (req, res) => {
    const data = req.body.data;
    const orderId = data.orderId;
    console.log("Got a new order! Order ID: " + orderId);
```



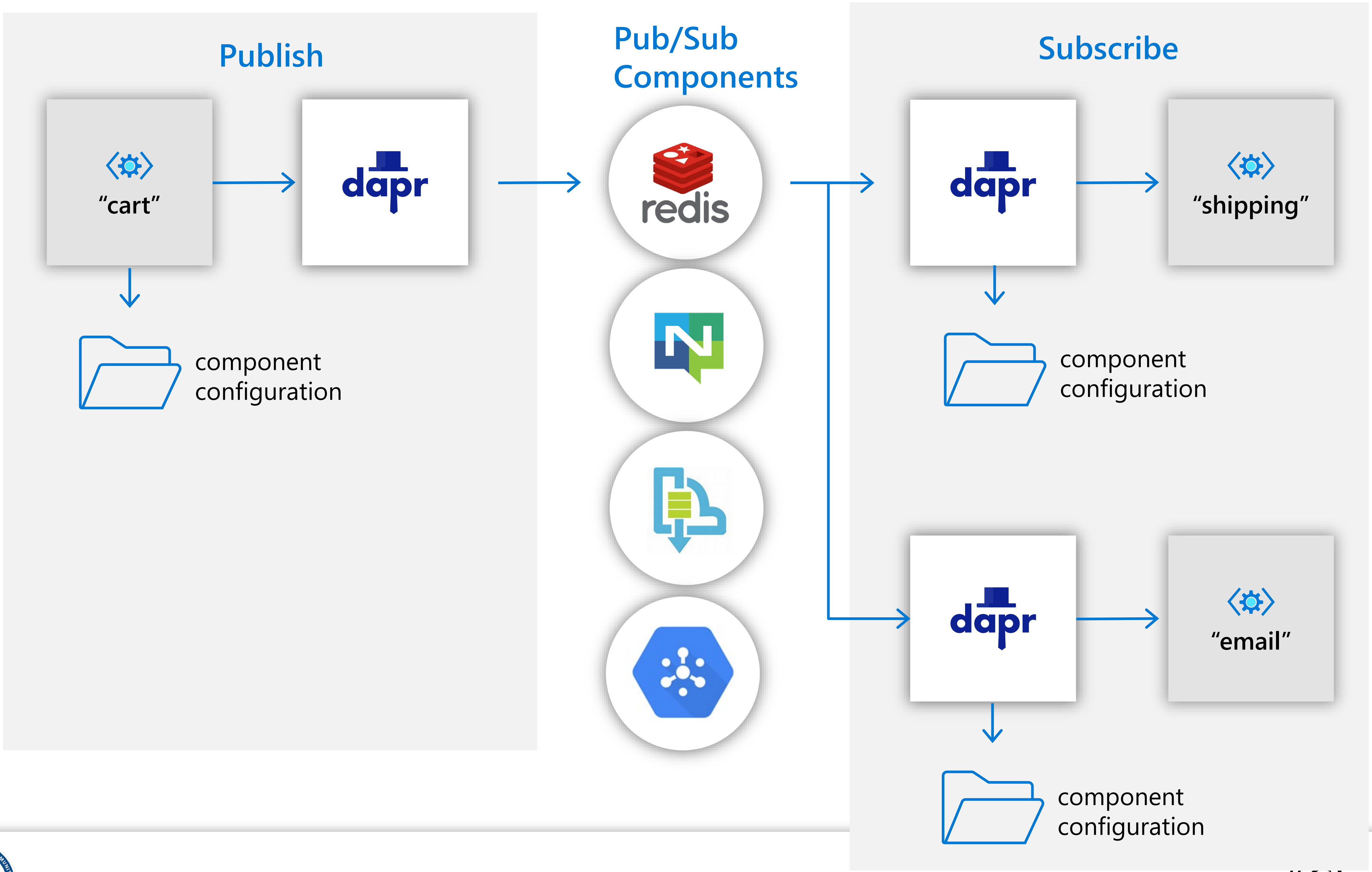
Publishing & Subscribing

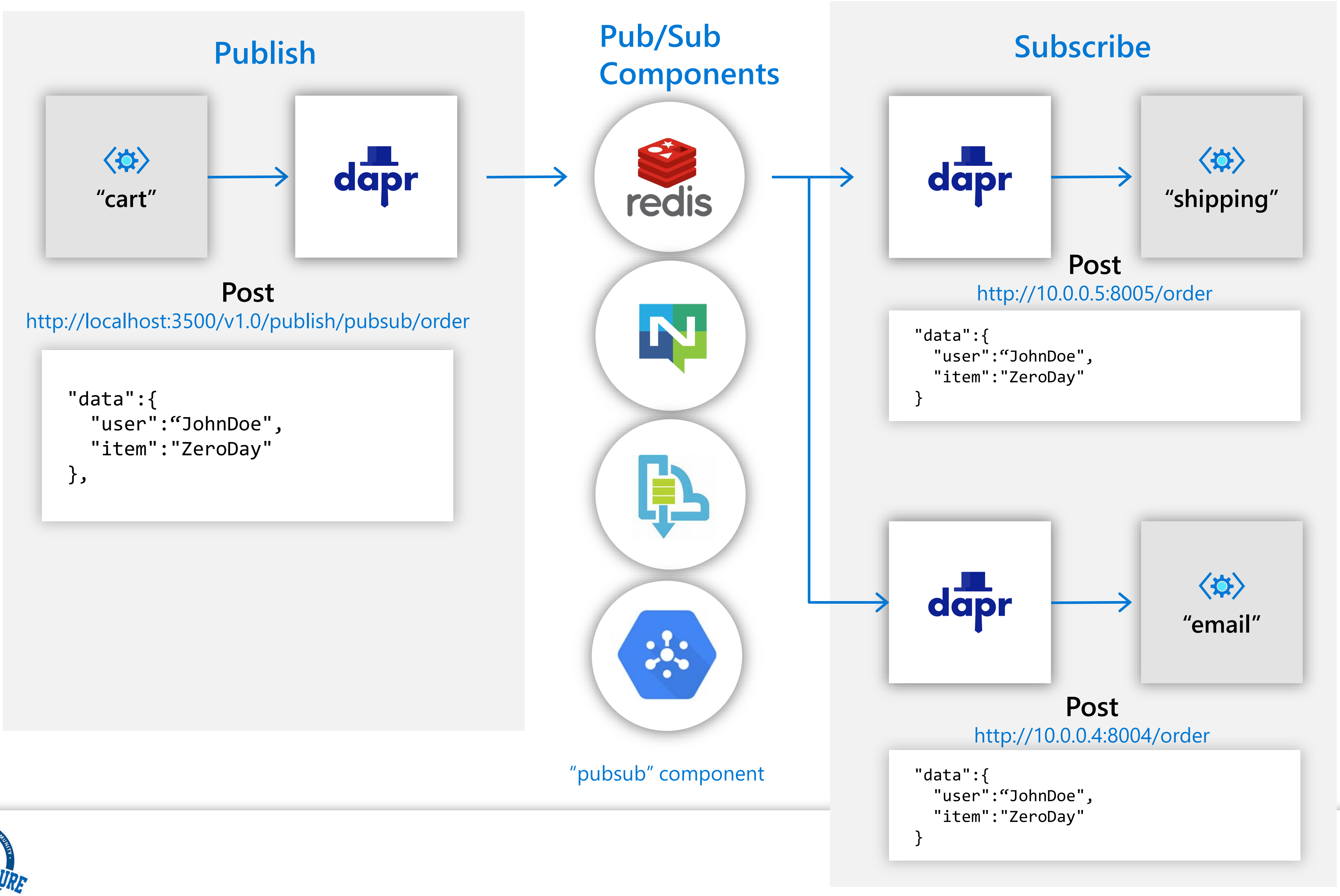

POST
http://localhost:3500/v1.0/publish/
POST
http://10.0.0.4:8004/order
POST
http://10.0.0.5:8005/order
"cart"
dapr
redis
"email"
"shipping"
Publish
Subscribe

```
"topic":"order",
"data":{
  "user":"johndoe",
  "item":"ZeroDay"
},
```

```
"data":{
  "user":"johndoe",
  "item":"ZeroDay"
}
```

#GlobalAzure


Publish
Pub/Sub Components
Subscribe
"cart"
dapr
component configuration
redis
dapr
"shipping"
component configuration
dapr
"email"
component configuration



```
"data":{
  "user":"JohnDoe",
  "item":"ZeroDay"
},
```

```
"data":{
  "user":"JohnDoe",
  "item":"ZeroDay"
}
```

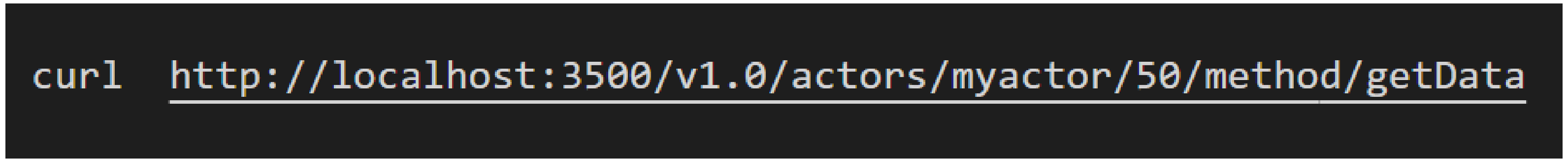
```
"data":{
  "user":"JohnDoe",
  "item":"ZeroDay"
}
```

Actors

- Actor pattern is good for solutions involving small, independent units of state and logic
- Actor runtime which provides concurrency, activation, deactivation, timers, reminders, and partitioning
- Standard API

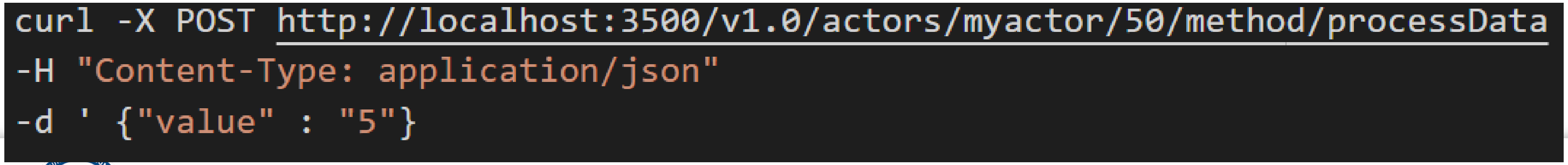
```
http://localhost:3500/v1.0/actors/<actorType>/<actorId>/method/<method>
```

Invoke the ***getData*** method on ***myactor*** with **id=50**



```
curl  http://localhost:3500/v1.0/actors/myactor/50/method/getData
```

Invoke the ***ProcessData*** method on **myactor** with **id=50**, providing the value **5**



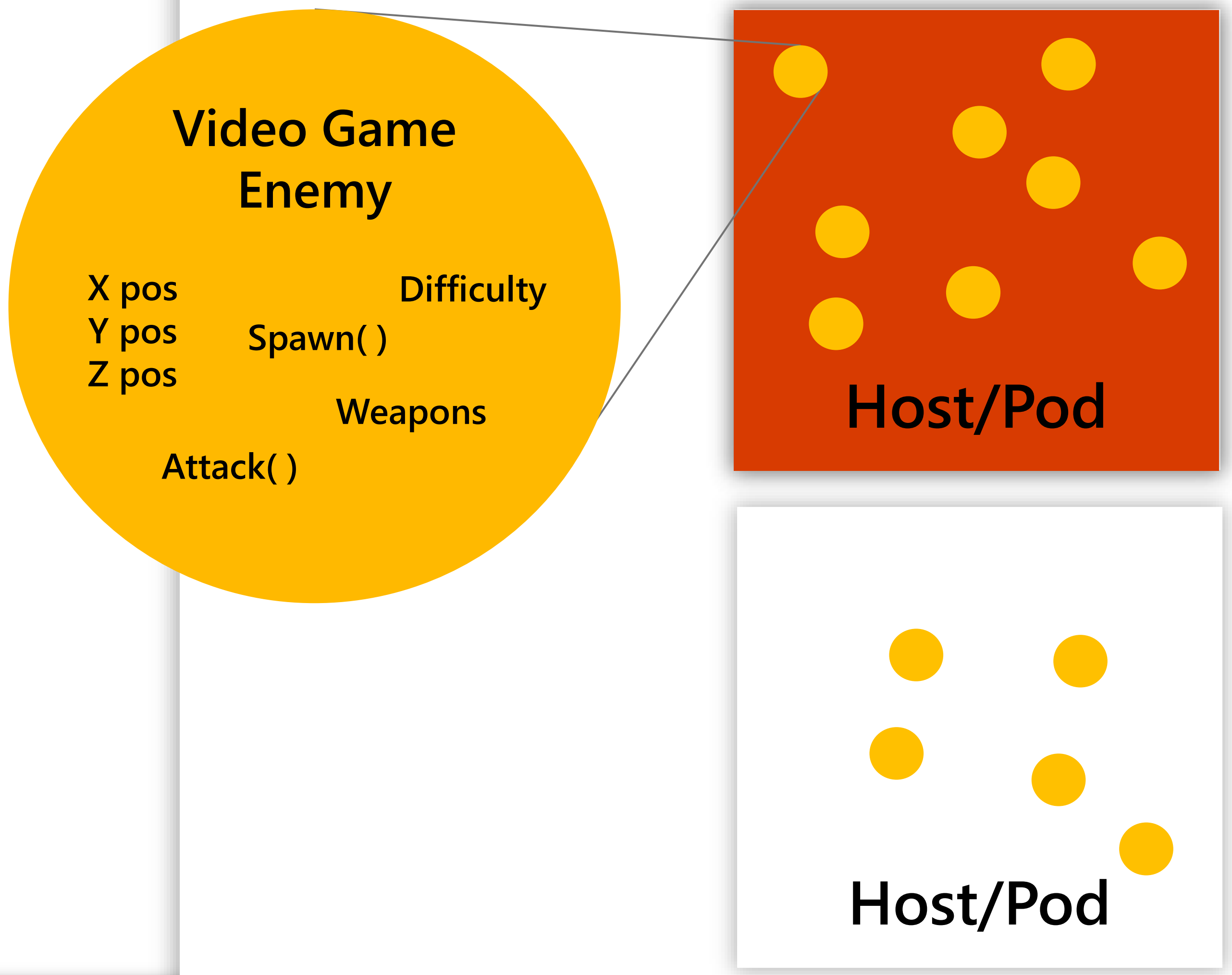
```
curl -X POST http://localhost:3500/v1.0/actors/myactor/50/method/processData
-H "Content-Type: application/json"
-d ' {"value" : "5"}
```

Virtual Actors with Dapr

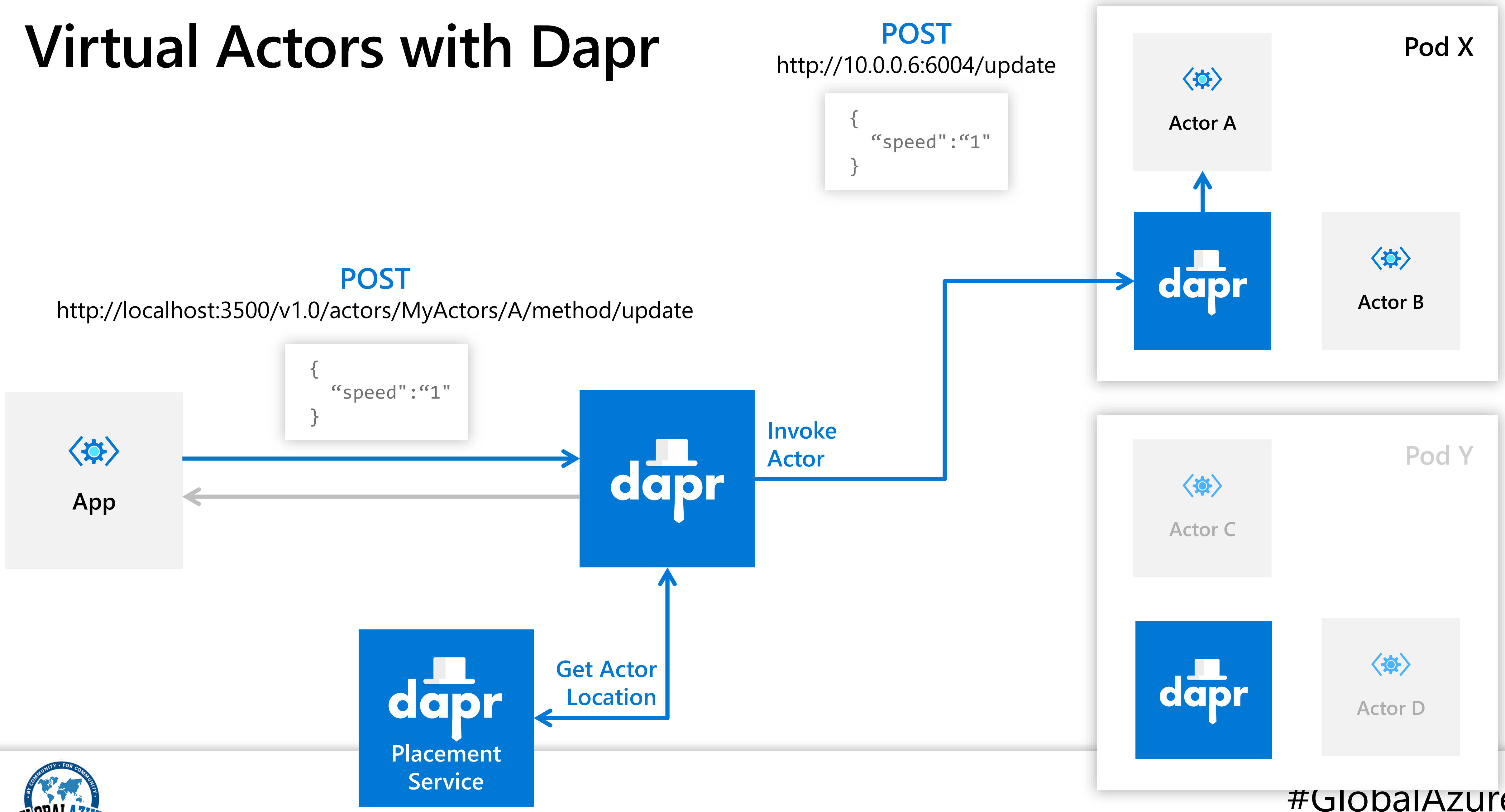
Stateful, objects of storage and compute

Dapr Actor Features:

- Distribution & failover
- Turn-based concurrency
- State management
- Timers
- Reminders



Virtual Actors with Dapr

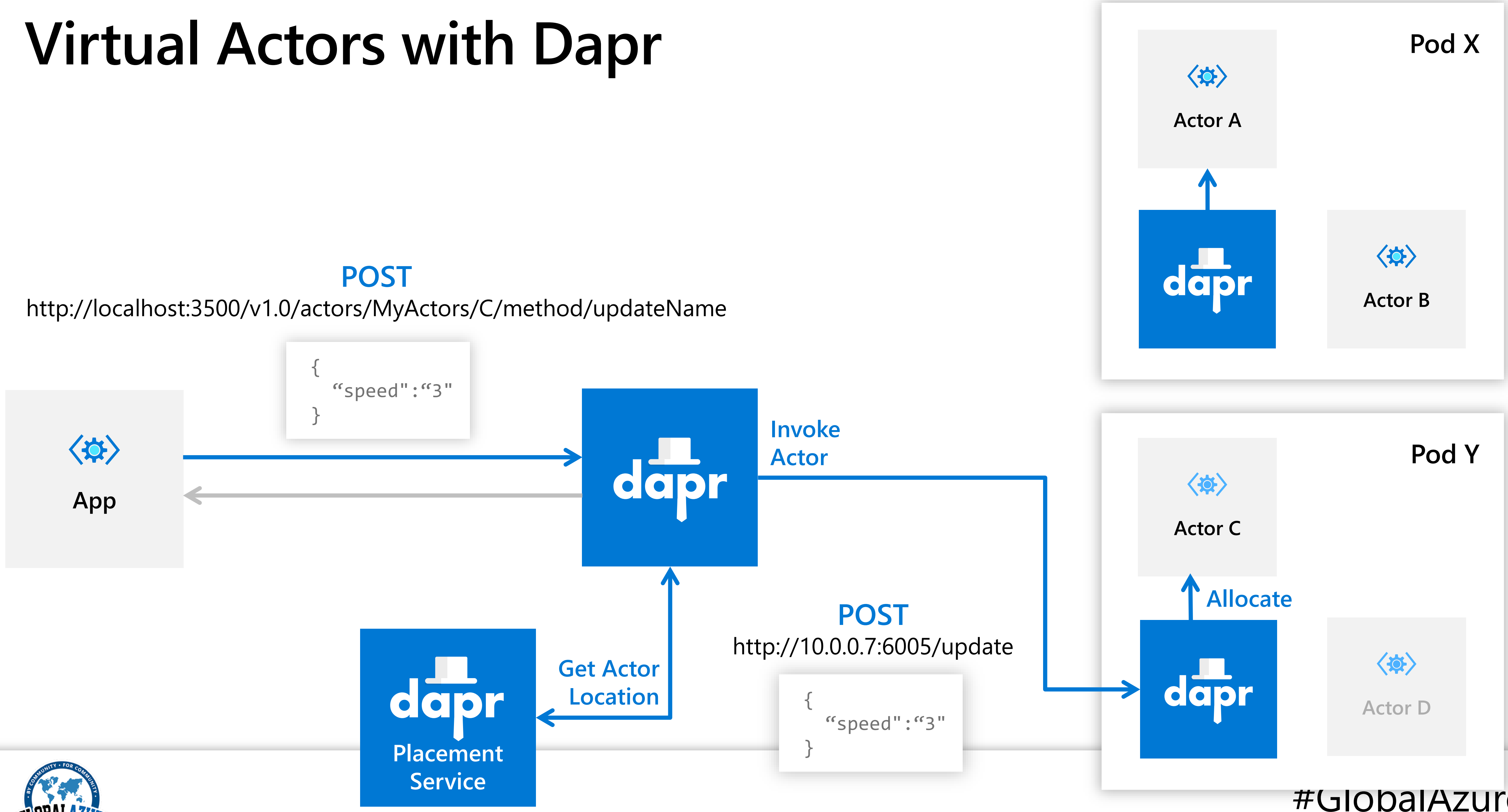


```
{
  "speed":"1"
}
```

```
{
  "speed":"1"
}
```

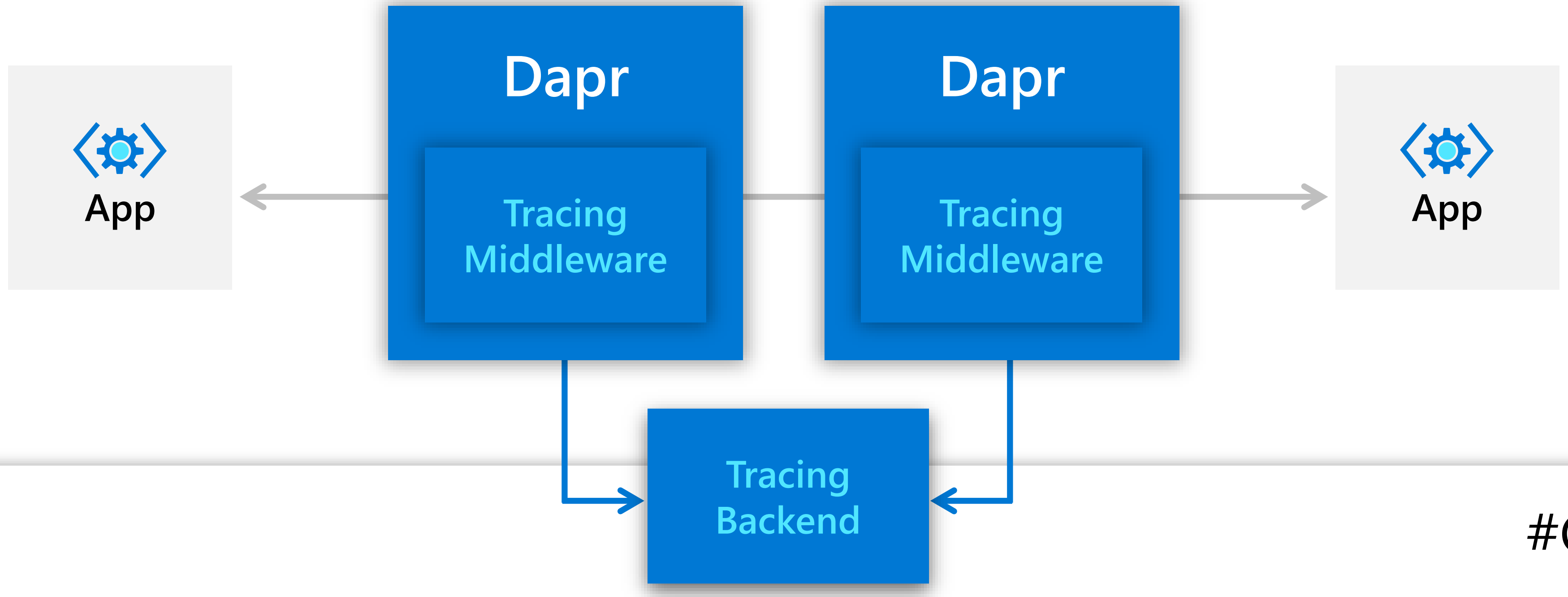
#GlobalAzure

Virtual Actors with Dapr

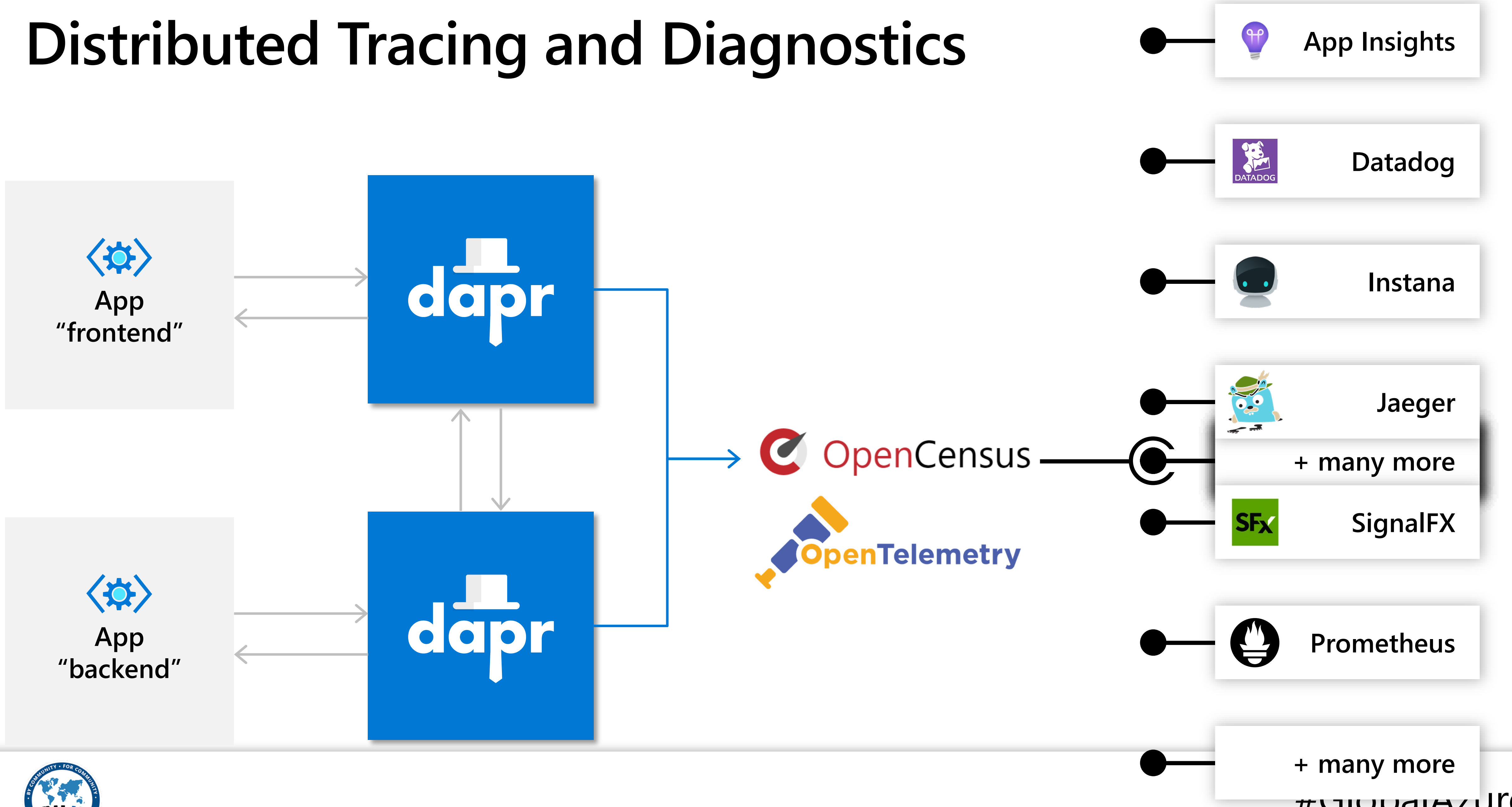


Distributed Tracing & Diagnostics

- See the message calls across between components and networked services
- Provides timing and performance information
- Integration with cloud services such as Azure Monitor



Distributed Tracing and Diagnostics


App Insights
Datadog
Instana
Jaeger
+ many more
SignalFX
Prometheus
+ many more
App "frontend"
App "backend"
dapr
dapr
OpenCensus
OpenTelemetry