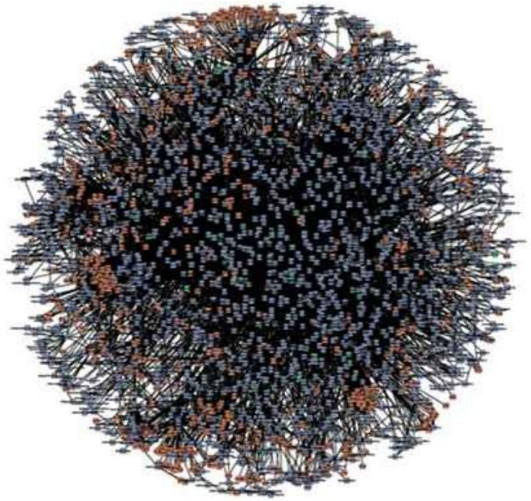
Architetture a Microservizi con DAPR e Azure Container Apps

Alessandro Melchiori - @amelchiori
CodicePlastico

Did you say “microservice”?

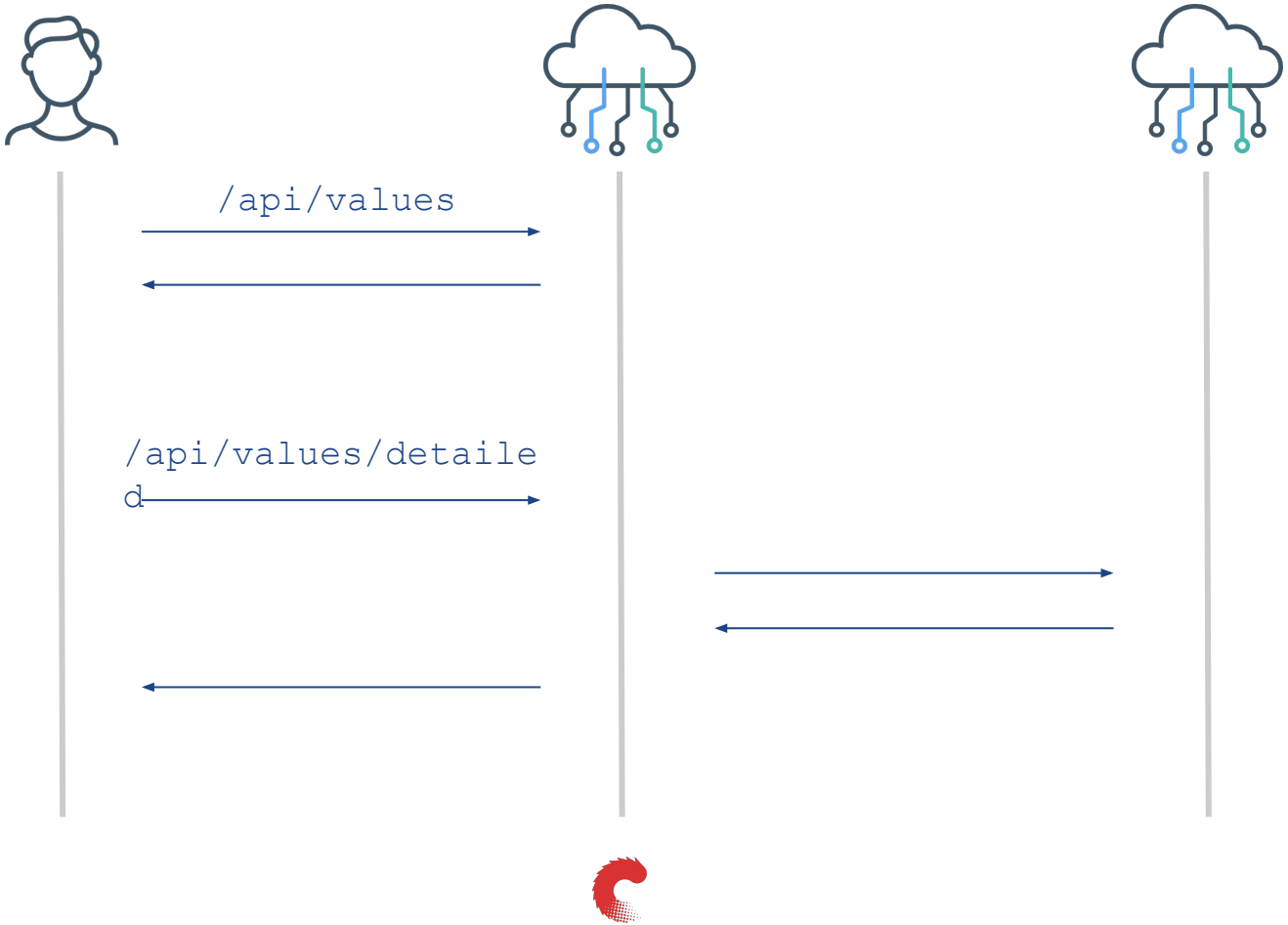


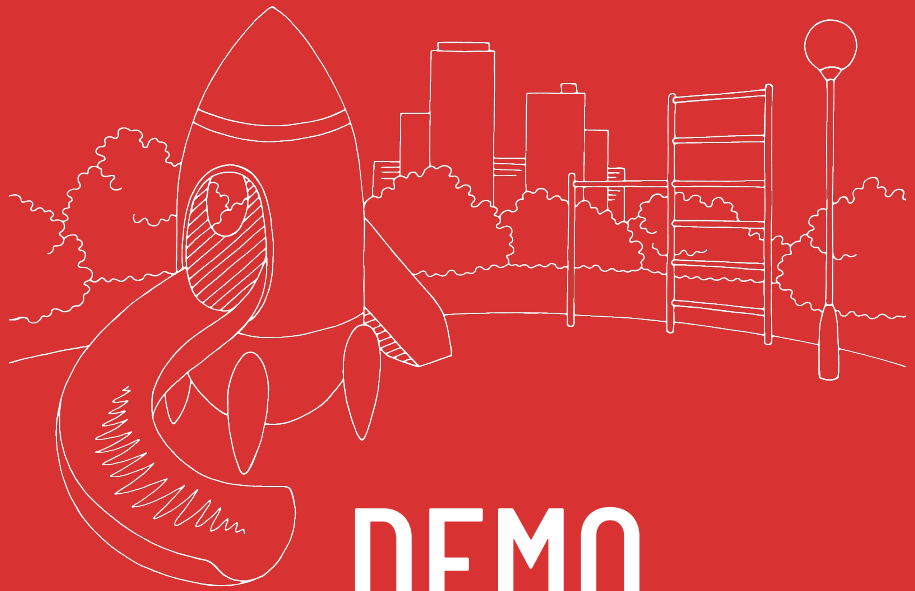
amazon.com

NETFLIX

With the network of microservices - service-to-service communication can become challenging

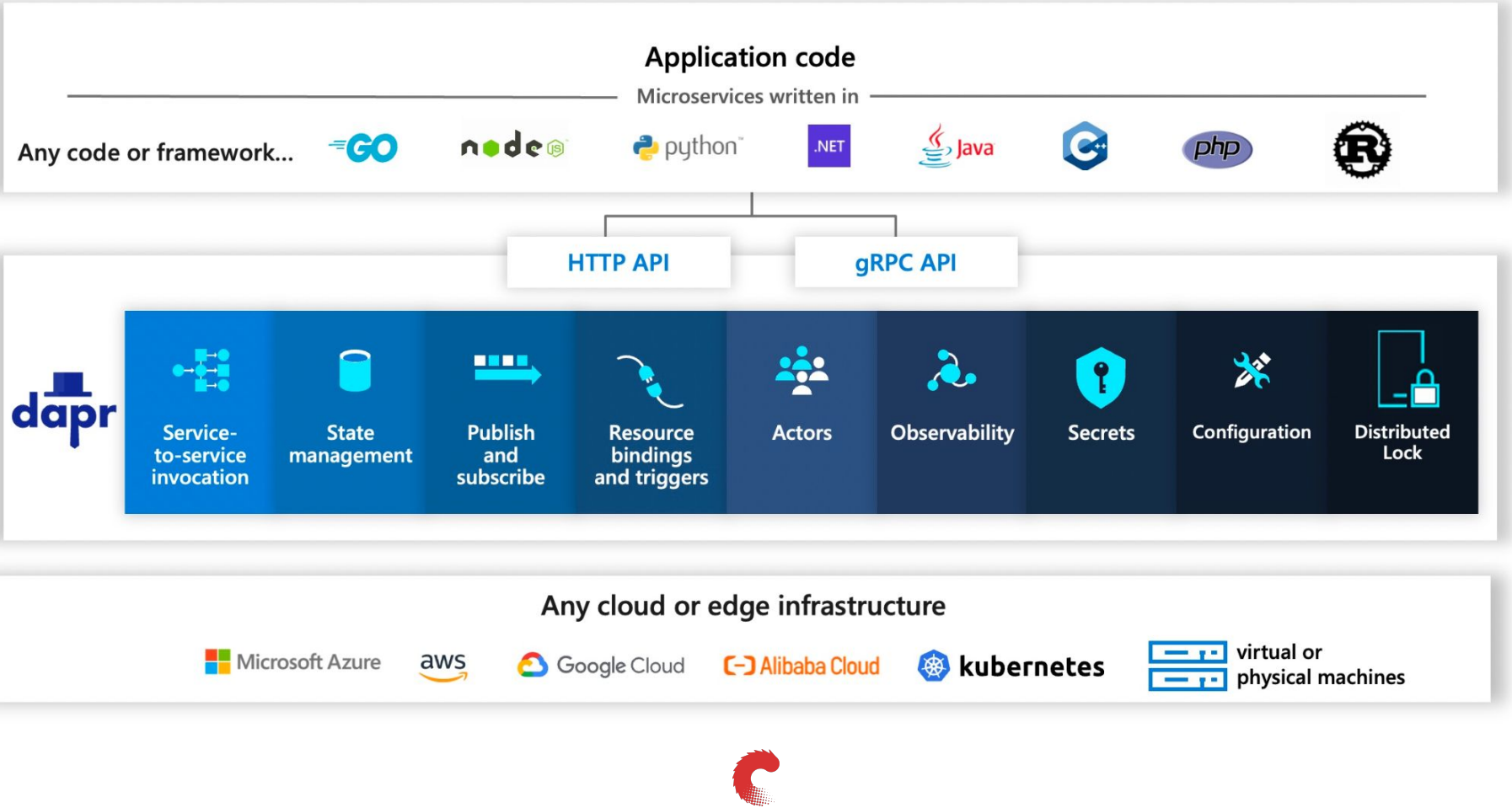
Instead of trying to prevent failures altogether, the goal is to **minimize the effects of a single** failing component.


/api/values
/api/values/detailed



DEMO

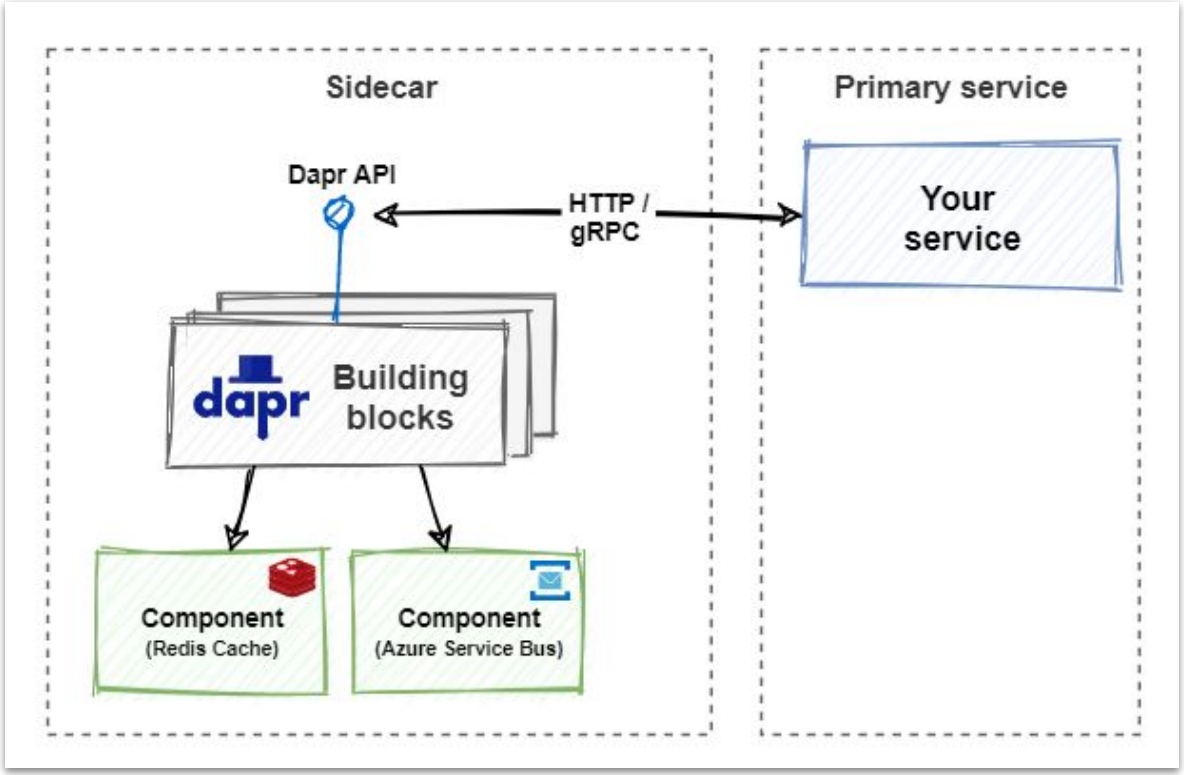
DAPR 101

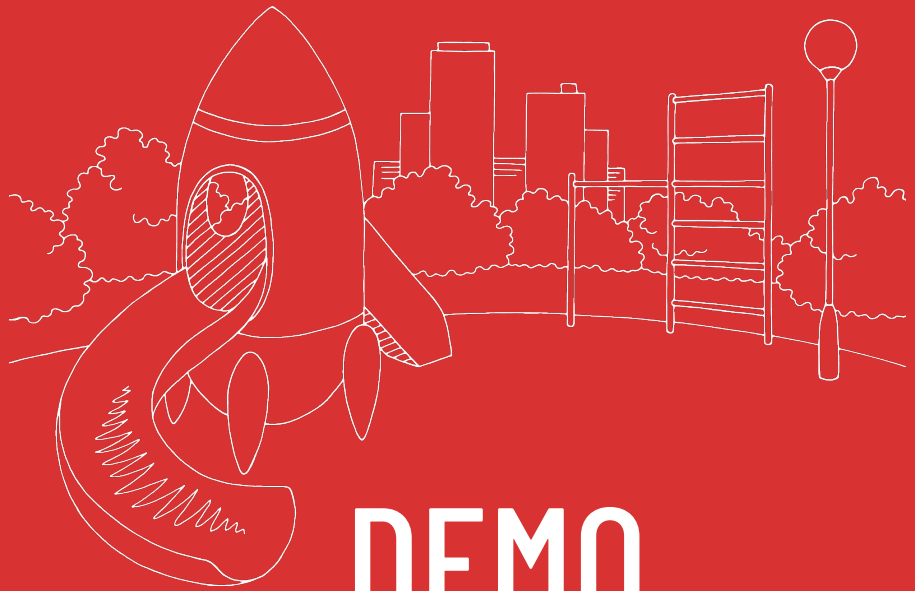

Application code
Microservices written in
Any code or framework...
HTTP API
gRPC API
dapr
Service-to-service invocation
State management
Publish and subscribe
Resource bindings and triggers
Actors
Observability
Secrets
Configuration
Distributed Lock
Any cloud or edge infrastructure
Microsoft Azure
aws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

SIDECAR PATTERN

The sidecar pattern consists of a main application plus a helper container with a responsibility that is essential to your application, but is not necessarily part of the application itself.

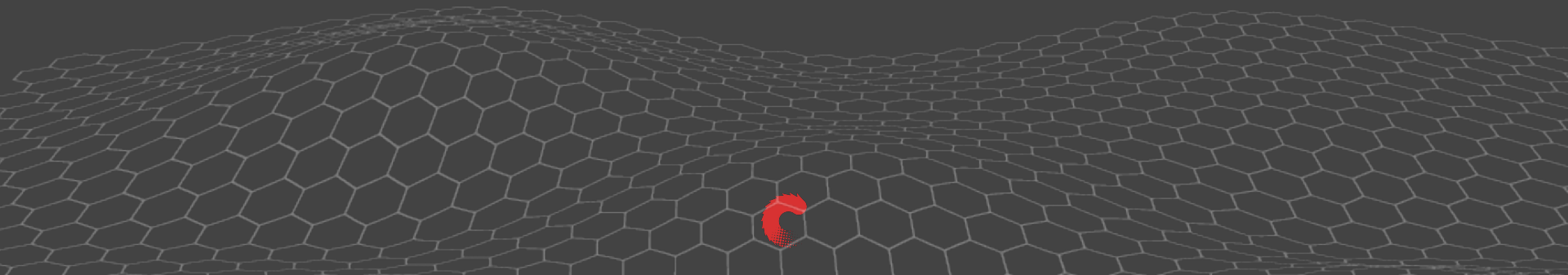
DAPR as a sidecar

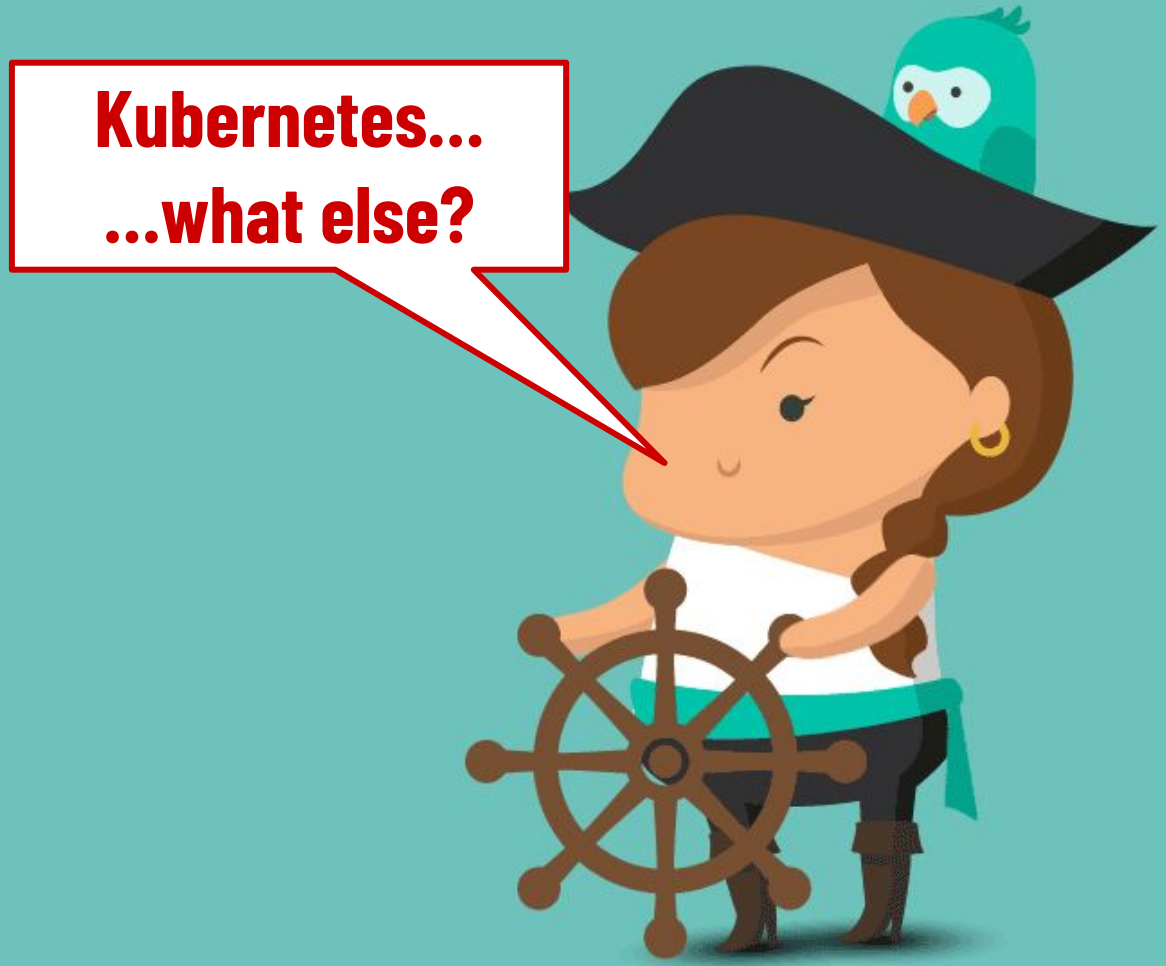




DEMO

Azure Container Apps




Kubernetes...
...what else?

Features

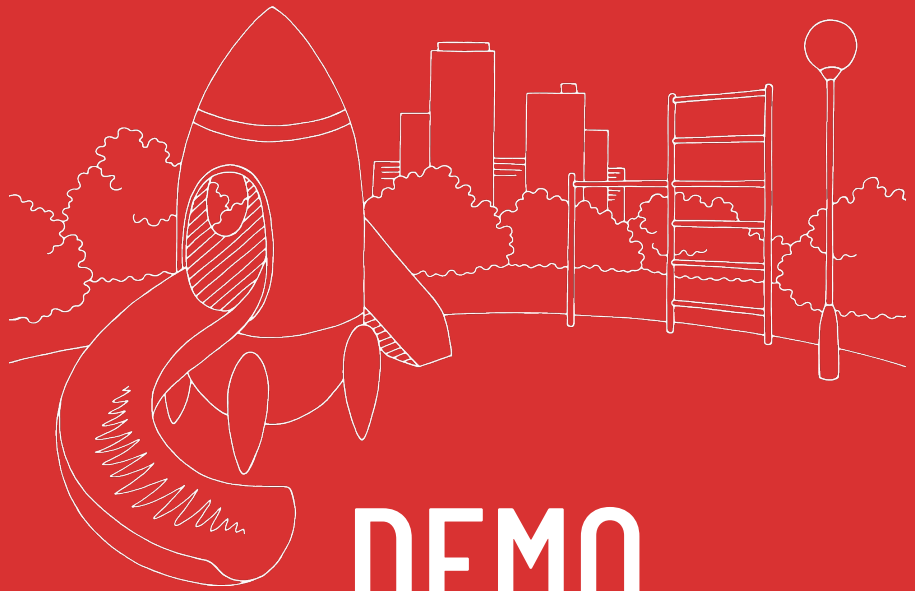
Support for a **variety of application types**, including HTTP APIs, microservices, event processing, and background tasks

Flexibility to write code using your language, framework, or SDK of choice

Robust **autoscaling** capabilities based on HTTP traffic or event triggers

Simple **configurations** to perform modern application lifecycle tasks

FULLY MANAGED SERVERLESS CONTAINER SERVICE FOR BUILDING AND DEPLOYING MODERN APPS AT SCALE



DEMO

ALESSANDRO MELCHIORI

Founder & Software developer @CodicePlastico
alessandro@codiceplastico.com
@amelchiori

Grazie!