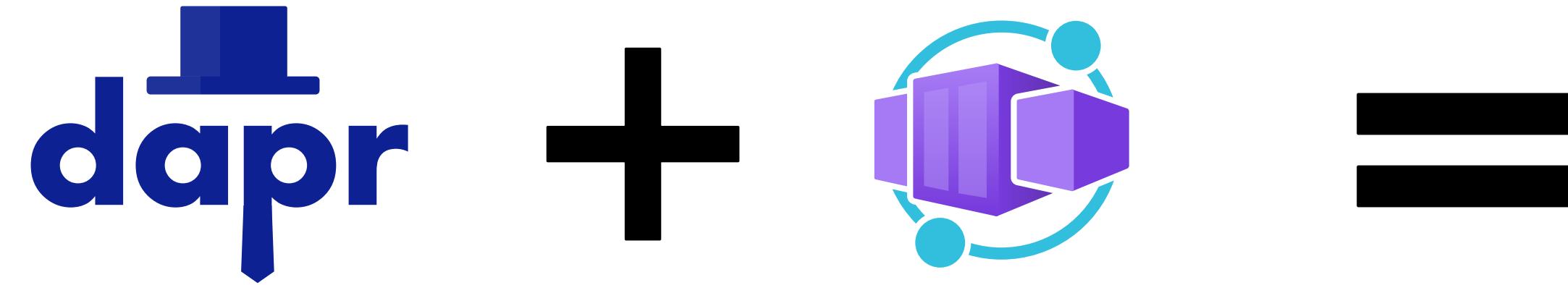


Cloud Native applications with Dapr and Azure Container Apps



Anssi Pannula

Agenda

- What is Dapr and how to use it?
- Azure Container Apps
- Demo - show me the code!

What is Dapr?

- Dapr stands for Distributed Application Runtime
- It was created at Microsoft in 2019 as part of an incubation project.
- Cloud Native Computing Foundation (CNCF) Incubating project since 2021
- Definitions:



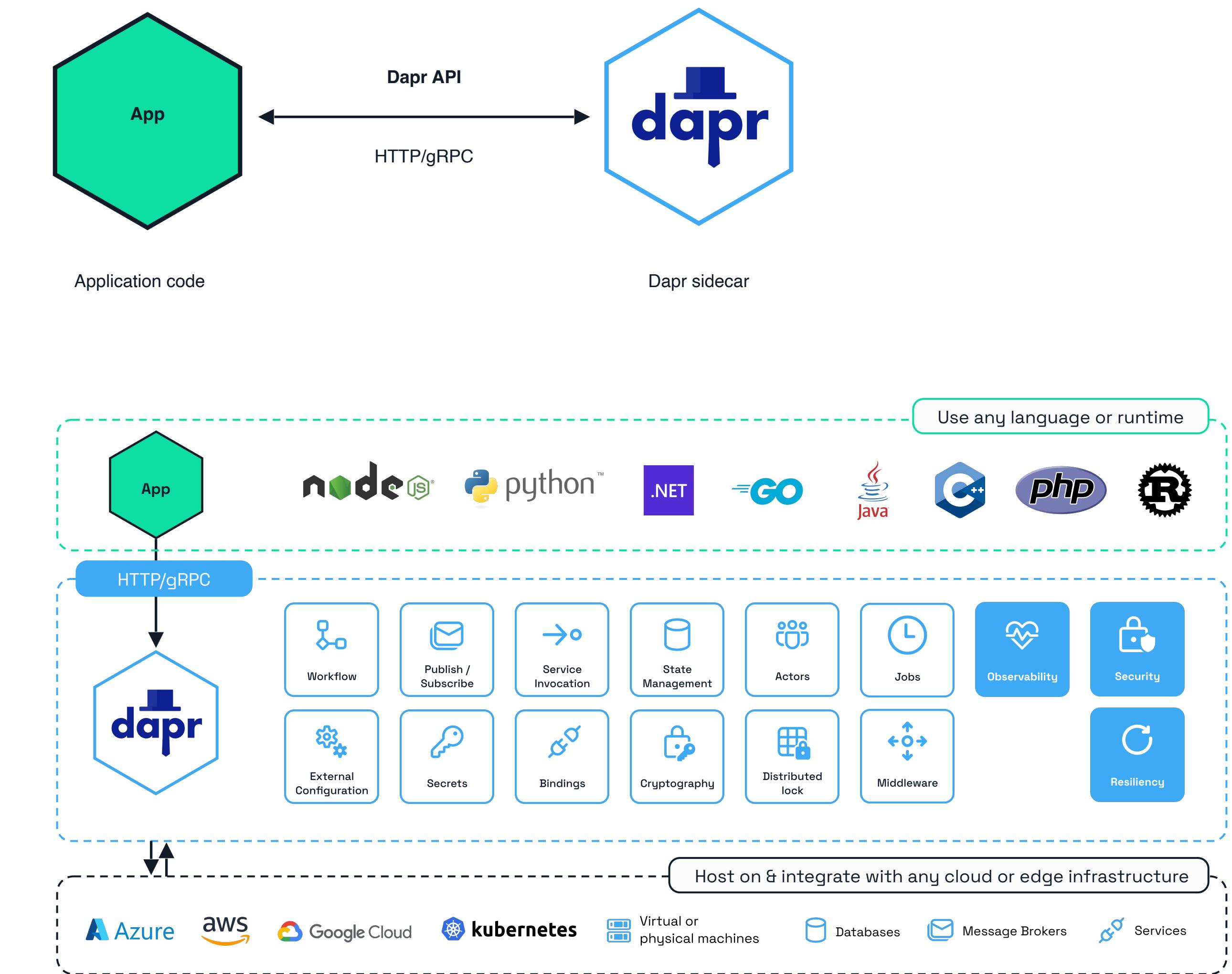
Dapr is a portable, event-driven **runtime** that makes it easy for any developer to **build resilient, stateless and stateful applications** that run on the cloud and edge and embraces the diversity of languages and developer frameworks. Leveraging the benefits of a **sidecar architecture**, Dapr helps you tackle the challenges that come with building **microservices** and keeps your code platform agnostic.



Dapr is a **toolkit** that helps different parts of an application communicate and work together smoothly, making development **easier** and more **efficient**. It handles tasks like service calls, state management, and messaging, so you can **focus on building features**.

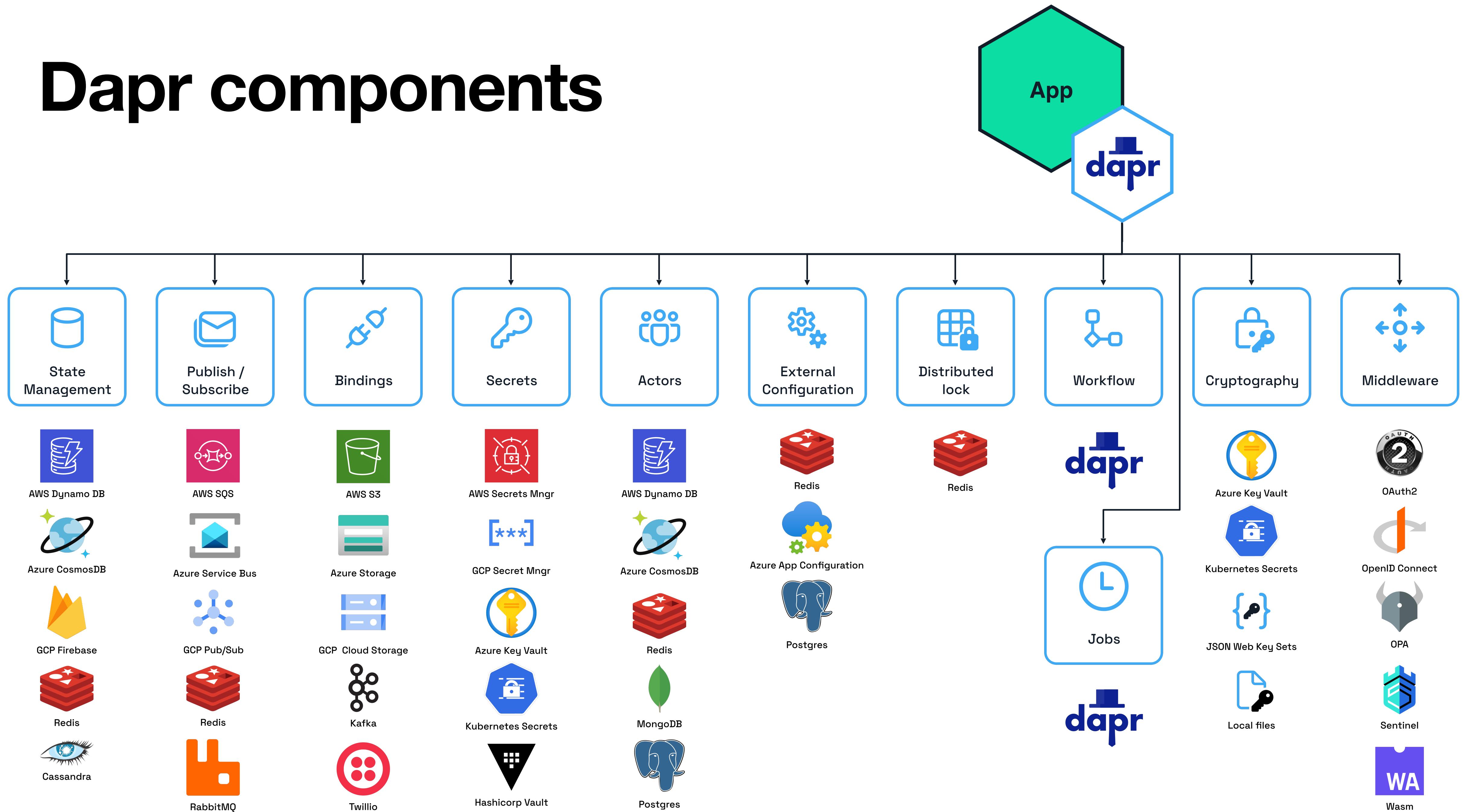
What is it really?

- Tool for building cloud native microservices
- A sidecar process
 - Sidecar pattern
- Use any language, code and framework
- Offers building blocks that enable us to focus on implementing business logic
- Speeds up application development by providing a single set of APIs for communication
- Attempts to address and resolve challenges related to distributed applications
 - Security (mTLS, access policy), service discovery, inter-service communication, resilience (timeout, retry, circuit breaker)



<https://dapr.io/>

Dapr components



How to use Dapr API in your application?

- Using HTTP client
 - POST <http://localhost:3500/v1.0/invoke/service-a/method/mymethod>
 - POST <http://localhost:3500/v1.0/publish/mybroker/mytopic>
 - GET <http://localhost:3500/v1.0/secrets/myvault/mytopsecret>
- DaprClient from SDK
 - .NET, Python, Java, GO, PHP, JavaScript
- Use your existing gRPC service definition

Hosting modes

- Self-hosted
 - Ideal for local development and testing
 - Run Dapr on your local machine using the Dapr CLI
 - **dapr init** and **dapr run**
- Kubernetes
 - Suitable for production use
 - Deploy dapr on a Kubernetes using **dapr init -k** or use Helm charts
 - Deploys placements, operator, sentry and injector pods.
- Azure Container Apps
 - Suitable for production use
 - Enable Dapr with a single boolean flag in your container app configuration.

Why you should use Dapr?

- Makes it easy for any developer to build cloud-native microservices
 - Increases your developer productivity by 20-40%*
- Language, framework and cloud agnostic
- Cloud Native Computing Foundation (CNCF) project
 - Vendor-neutral
 - Reduces the risk of the project being abandoned
- Large and active community
 - 120 community components
 - 3k contributors
 - 12/157 Largest CNCF project
- Dapr is secure, audited, and enterprise-ready
 - Dapr completed a comprehensive security audit in 2023 and no critical or high-severity issues found.
 - <https://blog.dapr.io/posts/2023/09/05/dapr-completes-2023-security-audit-increasing-enterprise-confidence/>

*<https://dapr.io>

Case Studies

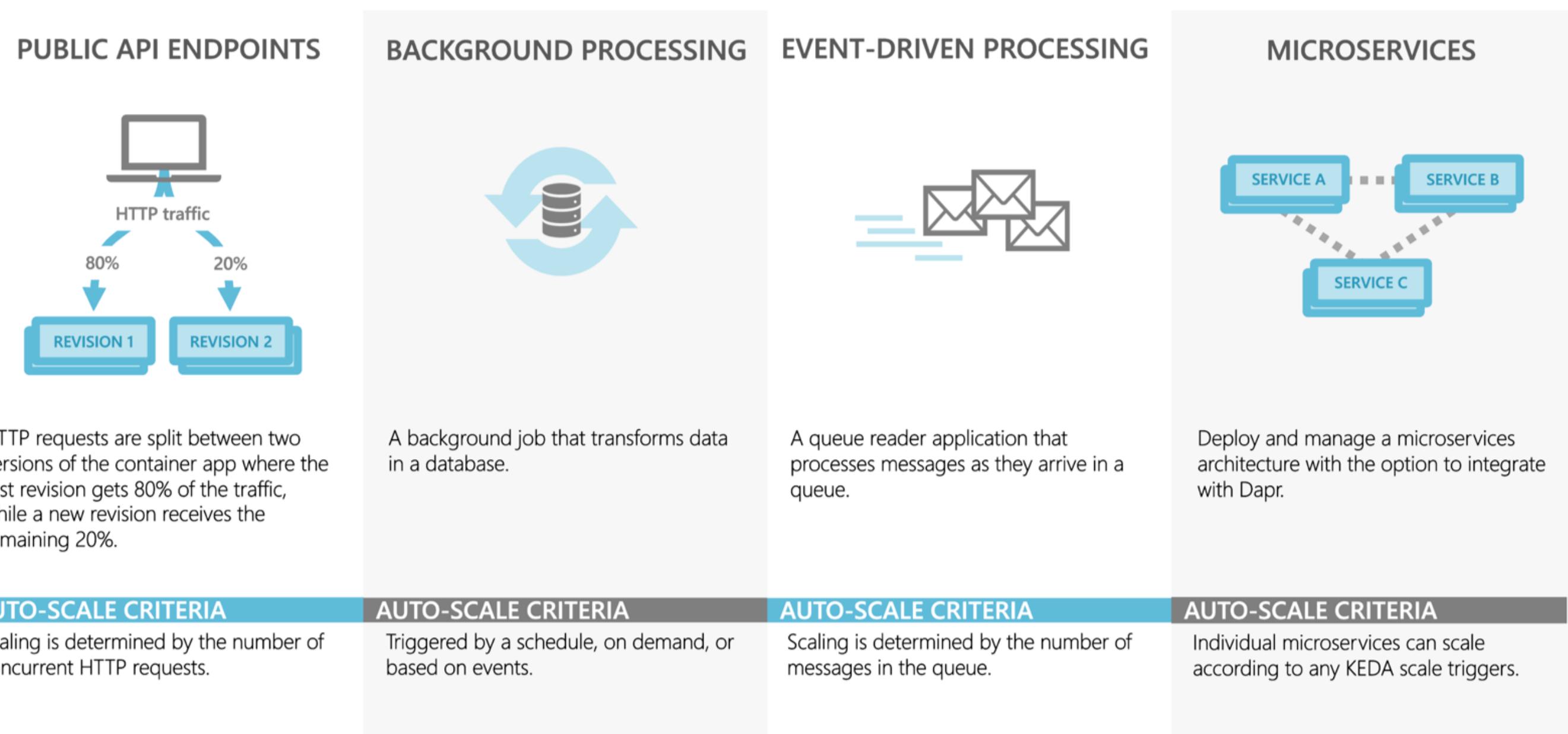
- HDFC Bank (<https://www.cncf.io/case-studies/hdfc-bank/>)
 - “Development began in early 2022 and the system went live after one year, becoming one of the **fastest applications to be developed and deployed** to a Tier One Bank.”
- DeFacto (<https://www.cncf.io/case-studies/defacto/>)
 - “As a result of adopting Dapr, DeFacto’s developers rapidly deployed a significant number of services into all environments.”
 - “Dapr’s service invocation API building block **streamlined the development process** by simplifying the discoverability and calling between multiple services.”
- Wortell (<https://www.cncf.io/case-studies/wortell/>)
 - “The team were able to **remove several software packages and a considerable amount of code** from all of their microservices.”
- Roadwork (<https://blog.dapr.io/posts/2021/02/09/running-dapr-in-production-at-roadwork/>)
 - “We moved the entire solution from Digital Ocean (was missing service accounts, crucial for security purposes) to AWS and finally to Azure with **migrations taking only a couple of hours**, since all we had to do is just configure Kubernetes, Dapr and KEDA again and we were all set-up and running again!”

Azure Container Apps - Focus on apps, not infrastructure

- Fully managed service that allows you to run microservices and containerized applications on a serverless platform
 - KEDA (Kubernetes Event-driven Autoscaling)
 - Provides horizontal scaling (scale out)
 - Envoy proxy
 - Network proxy for all HTTP requests
 - Dapr is “first-class citizen”
- “Abstraction layer” on top of Kubernetes
 - Hides the Kubernetes complexities
 - No need for kubectl
- Core Azure resources
 - Container Apps Environment:
 - Secure boundary around one or more container apps
 - Provides infrastructure (CPU, memory, network etc.) resources
 - Container App
 - Runs and manage containerized applications



Azure Container Apps: Example scenarios



<https://learn.microsoft.com/en-us/azure/container-apps>

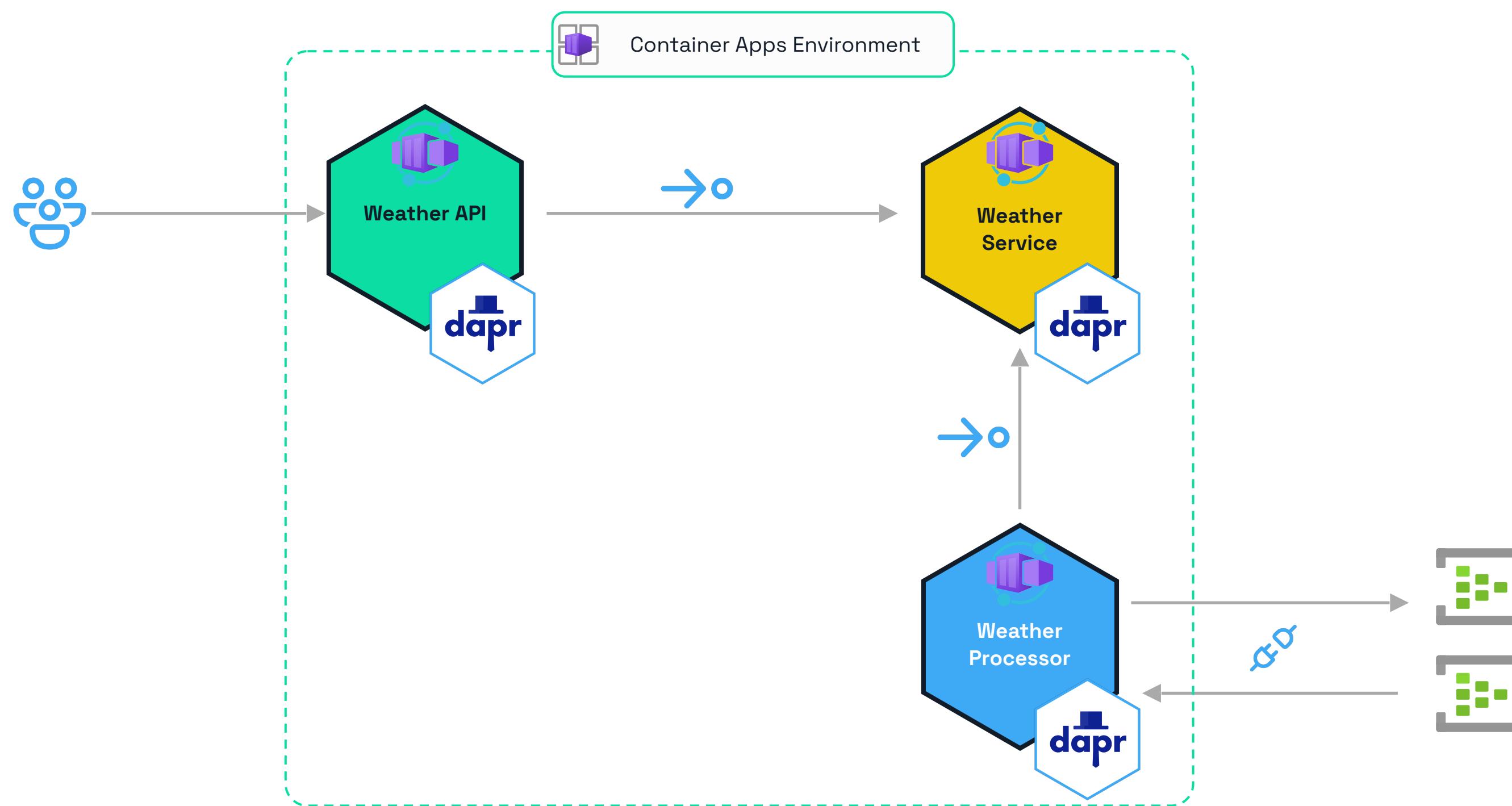
Supported Dapr components

- All the common components are supported
 - Actors
 - Bindings
 - Configuration
 - Publish and subscribe
 - Secrets
 - Service Invocation
 - State management

							
Service-to-service invocation — Perform direct, secure, service-to-service method calls	State management — Create long running, stateless and stateful services	Publish and subscribe — Secure, scalable messaging between services	Bindings (input/output) — Input and output bindings to external resources, including databases and queues	Actors — Encapsulate code and data in reusable actor objects as a common microservices design pattern	Observability — See and measure the message calls across components and networked services	Secrets — Securely access secrets from your application	Configuration — Access application configuration and be notified of updates

<https://learn.microsoft.com/>

Demo



Questions?

Thank you!