Build

Distributed Multi-cloud Applications With **Dapr**

Jabar Asadi | Software Engineer

D2IQ

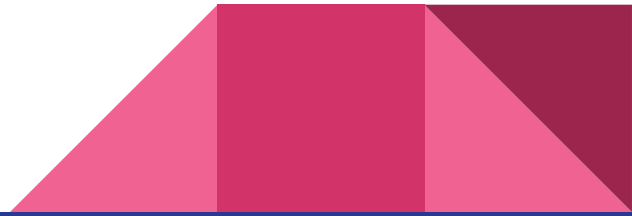
Distributed Application Challenges

- Service Resilience and Fault Tolerance
- Security and Access Control
  - Authentication and authorization **//** Communication channel **//** Auditing and logging
- Monitoring and Observability and Tracing
  - Infra **//** Application **//** Service level
- Service Coordination
  - Service Discovery **//** Load Balancing **//** Workflow execution, etc
- Configuration Management
  - Security **//** Consistency **//** scalability
- Compatibility and Portability
- Lack of Expertise (oracle)

What is Dapr?

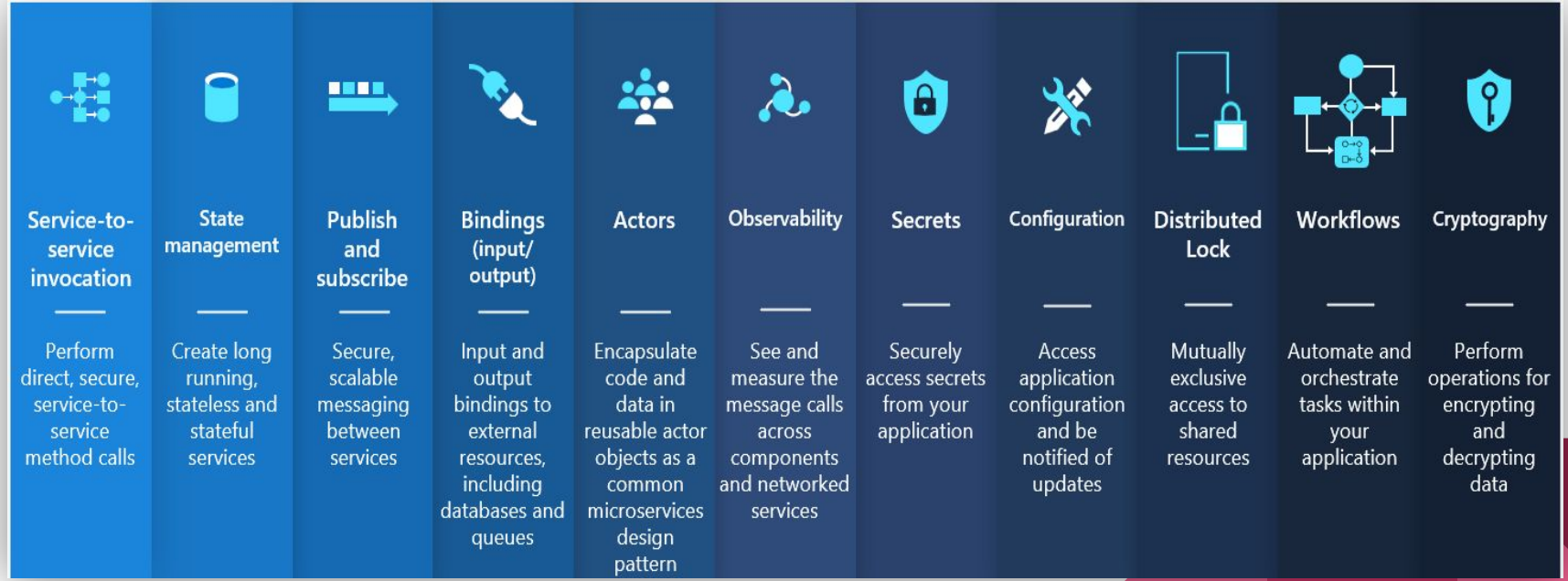
Open-Source distributed application framework.

- **Standardize** Distributed Application Development.
  - Allows developers to focus on writing business logic
  - Make developers productive
- **Consistent**, Portable and Platform-Independent
- Microservices and Event-Driven Style
- Language-Agnostic and Decentralized
- Built With **Best Practices** in Mind



Dapr Building blocks

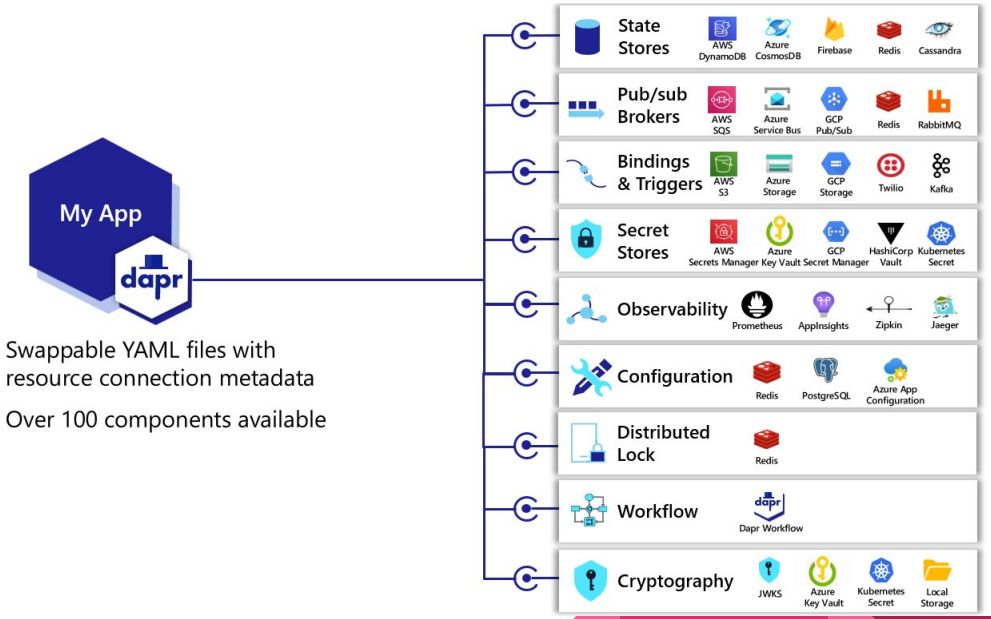
- HTTP // GRPC API
- Expose A Specific Interface



Component Schema

```
apiVersion: dapr.io/v1alpha1
kind: Component
metadata:
  name: [COMPONENT-NAME]
  namespace: [COMPONENT-NAMESPACE]
spec:
  type: [COMPONENT-TYPE]
  version: v1
  initTimeout: [TIMEOUT-DURATION]
  ignoreErrors: [BOOLEAN]
  metadata:
  - name: [METADATA-NAME]
    value: [METADATA-VALUE]
  ...
```

More details

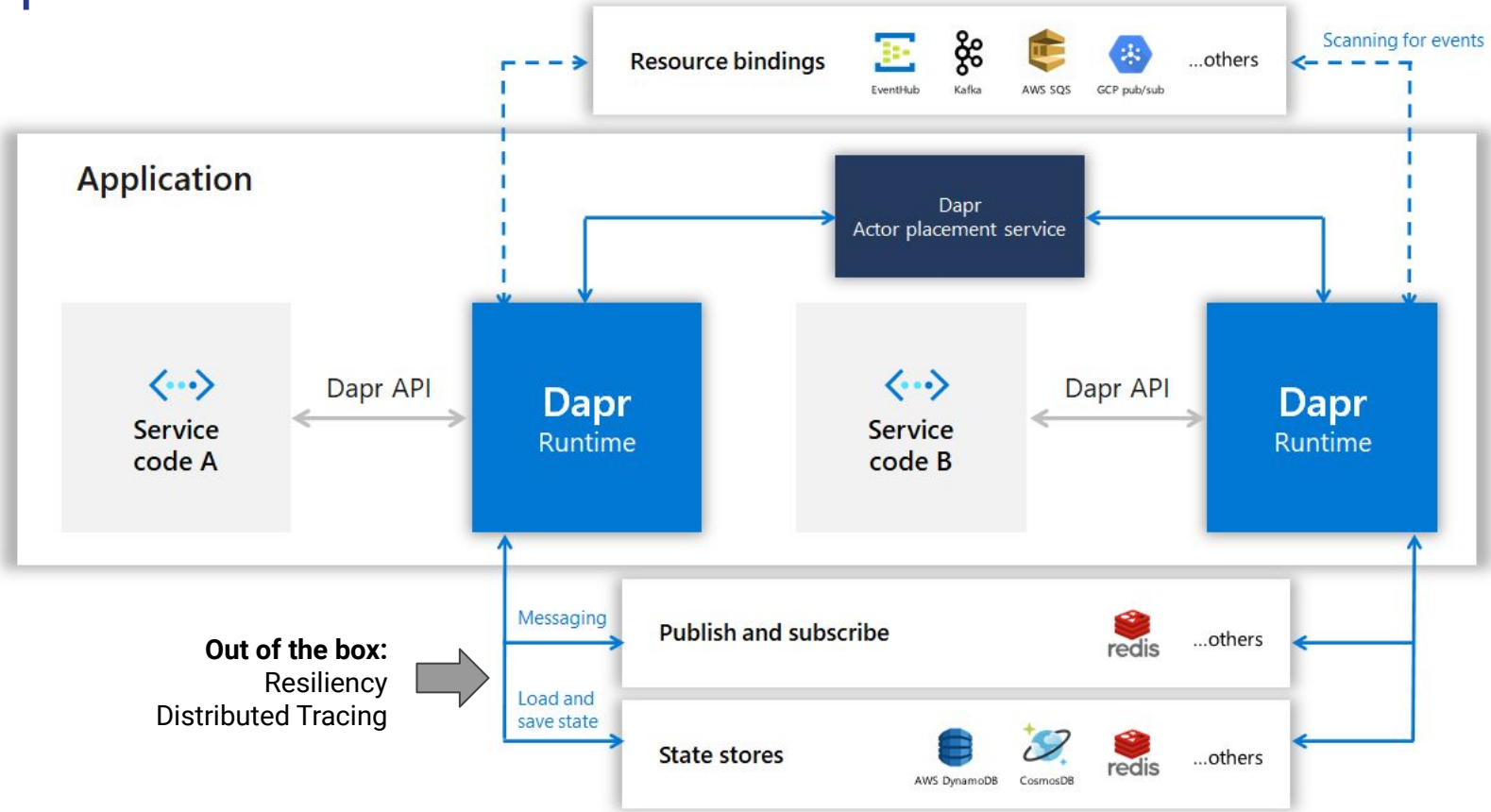


Swappable YAML files with resource connection metadata

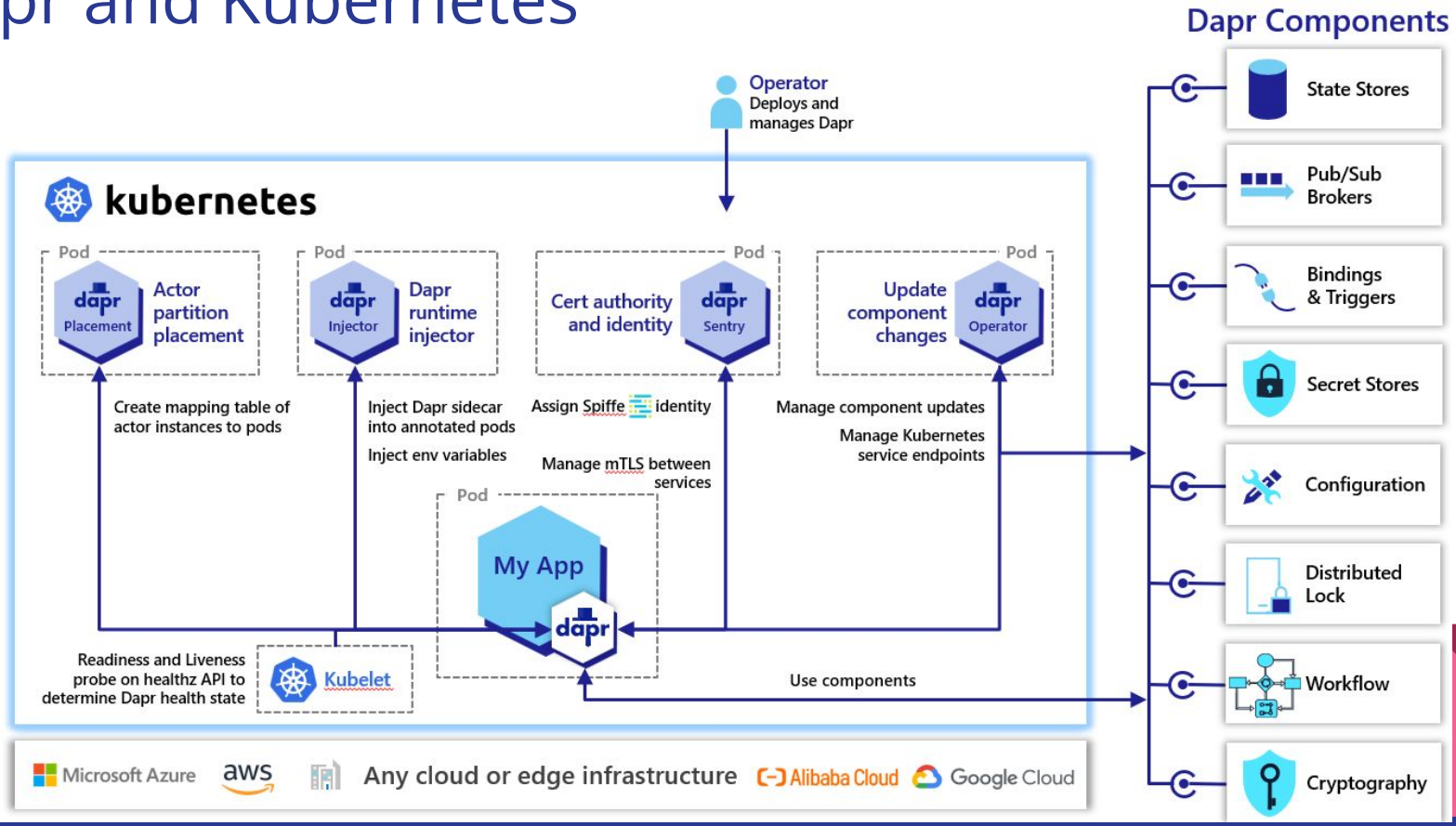
Over 100 components available

POST/GET /v1.0/{componentTYPE}/{componentName}/…

Dapr Sidecar



Dapr and Kubernetes



Using Dapr

- Dapr CLI
- Helm
- Github actions

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: myapp
  labels:
      app: go
spec:
  replicas: 2
  selector:
      matchLabels:
        app: go
  template:
    metadata:
       labels:
         app: node
       annotations:
         dapr.io/enabled: "true"
         dapr.io/app-id: "myapp"
         dapr.io/app-port: "3000"

...
...
```

Kubernetes

```
dapr run --app-id serving \
   -P http \
   -p 8080 \
   -H 3500 \
   -d ./config \
   --log-level debug \
   go run .
```

Self-hosted

```yaml
- name: Install Dapr
 uses: dapr/setup-dapr@v1
 with:
   version: '1.11.0'
```

GHA

Demo