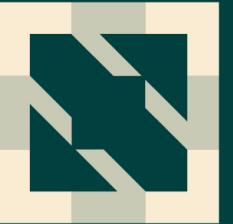


KubeCon



CloudNativeCon

**S OPEN SOURCE SUMMIT**

China 2023



KubeCon



CloudNativeCon



OPEN SOURCE SUMMIT

China 2023

# Run Serverless Workloads on Any Infrastructure with OpenFunction

*Benjamin Huo, QingCloud Technologies  
Yifei, Wang, QingCloud Technologies*

# 霍秉杰

青云科技可观测性与函数计算平台负责人

[KubeSphere](#) Maintainer

[OpenFunction](#) 发起人

[Fluent Operator](#) 发起人

[Notification Manager](#) 发起人

[Kube-Events](#) 发起人



# 王翼飞

青云科技可观测性与函数计算平台工程师

[OpenFunction](#) Maintainer

[KubeSphere](#) Maintainer



# CONTENTS

- ❑ 构建开源 FaaS 平台的必要性与可行性
- ❑ OpenFunction 简介
- ❑ 冷启动优化：Dapr proxy 模式及 Wasm 支持
- ❑ OpenFunction 在自动驾驶领域的应用
- ❑ 社区、路线图与 Demo

# 构建开源 FaaS 平台的必要性与可行性

# 构建开源 FaaS 平台的必要性

□ Kubernetes 带来了云厂商中立的可能性:

- Multi Cloud
- Distributed Cloud

□ 但 FaaS 领域很难实现云厂商中立

- 每个云厂商都有自己的 FaaS 平台
- 这些云厂商的 FaaS 平台通常是和自己云上的后端服务绑定的

□ CNCF Serverless 白皮书 (2018) 对 Serverless 缺点的分析:

- 因各厂商编程模型、事件/消息接口、后端服务的差异而导致存在厂商锁定的风险
- Potential for platform lock-in due to the platform's programming model, eventing/  
message interface and BaaS offerings.

# 构建开源 FaaS 平台的必要性

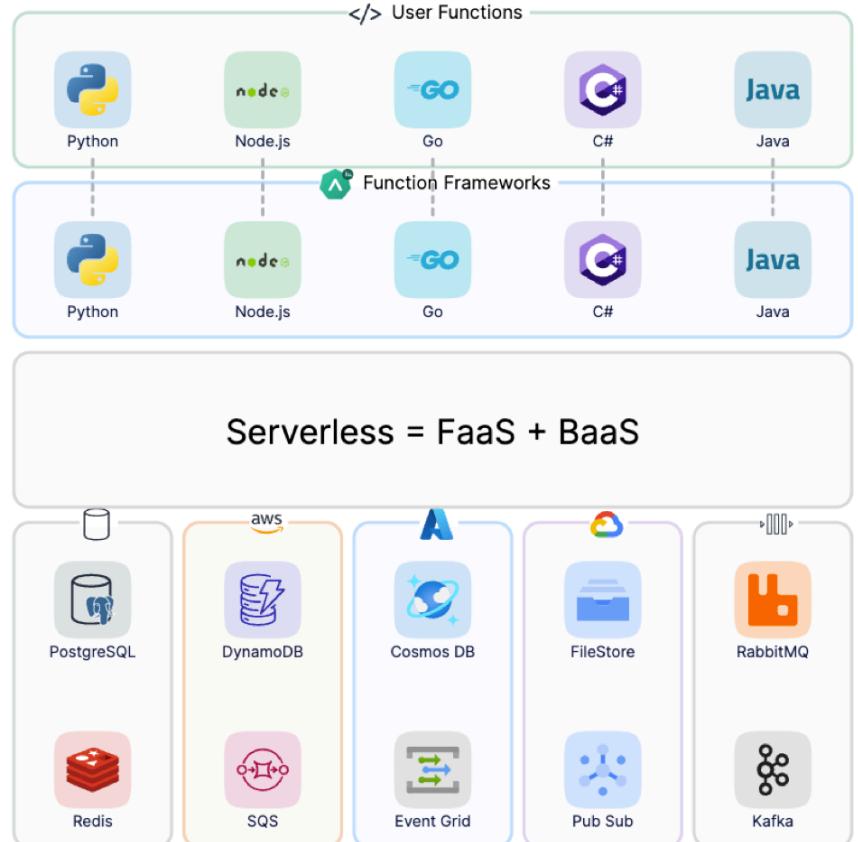
- 不开源很难实现厂商中立
- CNCF Serverless 白皮书 (2018) 对 Serverless 缺点的分析：
  - 标准化和生态成熟度方面有所欠缺
- 云原生 Serverless 领域的最新进展：
  - Dapr
  - KEDA
  - Knative Serving
  - Tekton, Shipwright, Cloud Native Buildpacks

# 构建开源 FaaS 平台的必要性

**Serverless Computing = FaaS + BaaS**

— [A Berkeley View on Serverless Computing](#)

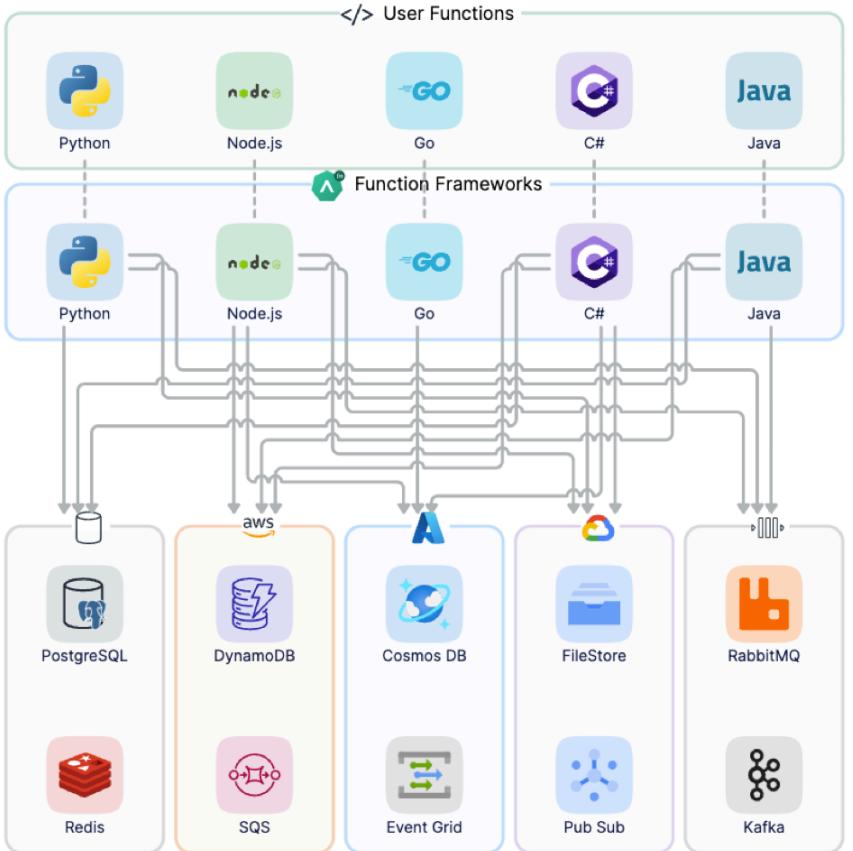
- 云函数是函数计算的主体
- 丰富的后端服务是函数的重要依托
- 云厂商通过提供托管 (Hosted) 的函数计算 (FaaS) 和各类后端中间件服务 (BaaS) 把开发者锁定在自己的云平台之上
- 如何解决下云及跨云迁移的问题?
- 跨云迁移后如何解决各云厂商 BaaS 服务接口的差异?



# 构建开源 FaaS 平台的必要性

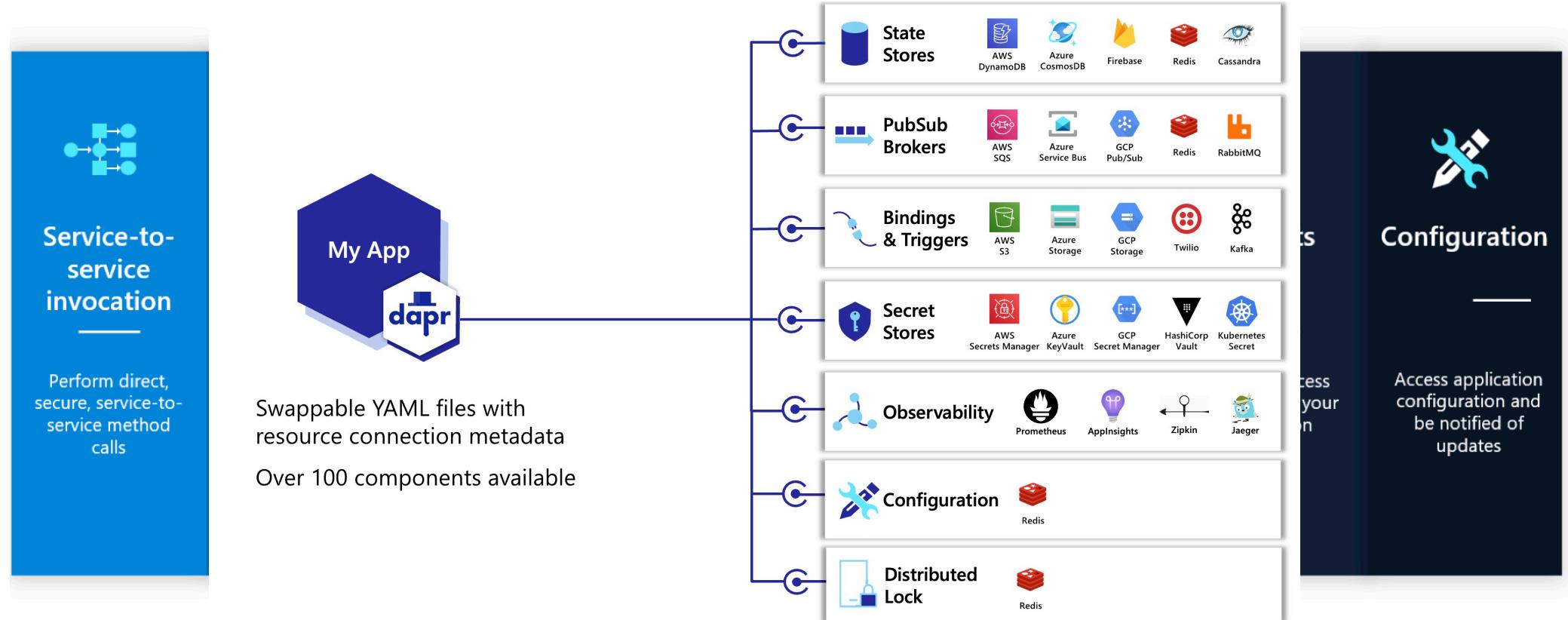
多语言的函数与众多后端服务交互的复杂性：

- FaaS 平台通常需要支持多种语言
- FaaS 平台通常也会利用到众多后端服务
- $N \times M$  的复杂性



# 构建开源 FaaS 平台的可行性

## □ 用 Dapr 来解耦分布式应用对后端支撑服务的访问

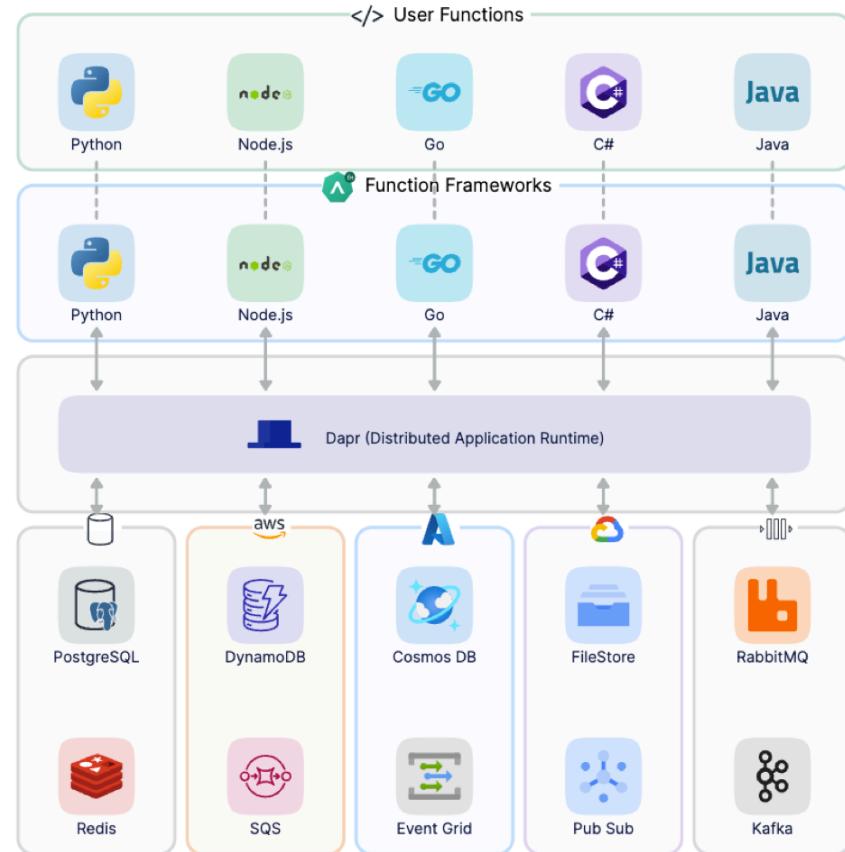


Dapr: <https://docs.dapr.io/concepts/components-concept/>

# 构建开源 FaaS 平台的可行性

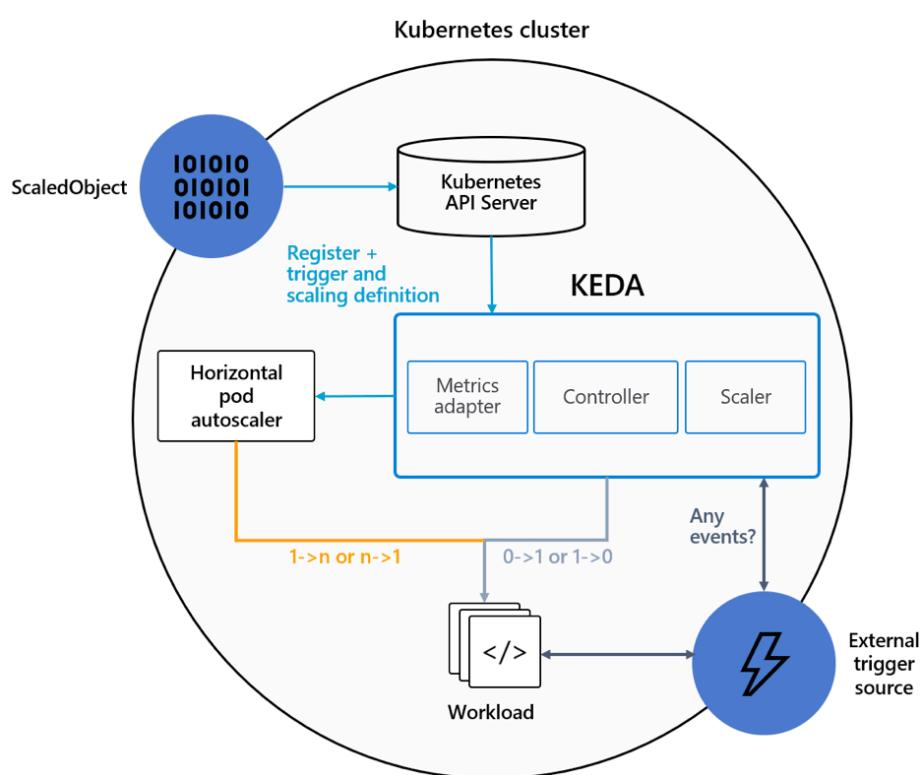
引入分布式应用运行时 Dapr 简化函数与众多后端服务的交互：

- FaaS 平台需要支持 N 种语言
- 每种语言的函数只需要和 Dapr 交互，复杂度降为  $N \times 1$
- 由 Dapr 去处理与众多后端服务的交互
- 解耦了 FaaS 平台与各云厂商的 BaaS，做到了云厂商中立



# 构建开源 FaaS 平台的可行性

## 用 KEDA 来解耦海量事件源的特有指标与应用的自动伸缩



### Event sources and scalers

KEDA has a wide range of **scalers** that can both detect if a deployment should be activated or deactivated, and feed custom metrics for a specific event source. The following scalers are available:

ActiveMQ	ActiveMQ Artemis	Apache Kafka	AWS CloudWatch	AWS Kinesis Stream	AWS SQS Queue
Azure Application Insights	Azure Blob Storage	Azure Event Hubs	Azure Log Analytics	Azure Monitor	
Azure Pipelines	Azure Service Bus	Azure Storage Queue	Cassandra	CPU	Cron
Elasticsearch	External	External Push	Google Cloud Platform Pub/Sub	Graphite	Datadog
IBM MQ	InfluxDB	Kubernetes Workload	Liikus Topic	Memory	Metrics API
MySQL	NATS Streaming	New Relic	OpenStack Metric	OpenStack Swift	PostgreSQL
Prometheus	RabbitMQ Queue	Redis Lists	Redis Lists (supports Redis Cluster)		Predictkube
Redis Lists (supports Redis Sentinel)	Redis Streams	Redis Streams (supports Redis Cluster)			
Redis Streams (supports Redis Sentinel)	Selenium Grid Scaler	Solace PubSub+ Event Broker			

KEDA: <https://keda.sh/docs/2.7/scalers/>

# OpenFunction 简介

# OpenFunction 是什么？

云原生开源函数计算平台 (2022 年 4 月成为 CNCF Sandbox 项目)

□ 云厂商中立:

- 与各云厂商的 BaaS 服务松耦合
- 通过 Dapr，简化了与各云厂商或开源 BaaS 服务的集成

□ 同时支持同步与异步函数

- 同步函数基于 [Kubernetes Gateway API](#) 实现了 [OpenFunction Gateway](#) 作为函数入口
- 异步函数可直接从事件源消费事件，并可根据事件源特有的指标自动伸缩

□ 支持直接从函数代码生成符合 OCI 标准的函数镜像

- 基于 [Cloud Native Buildpacks](#) 实现

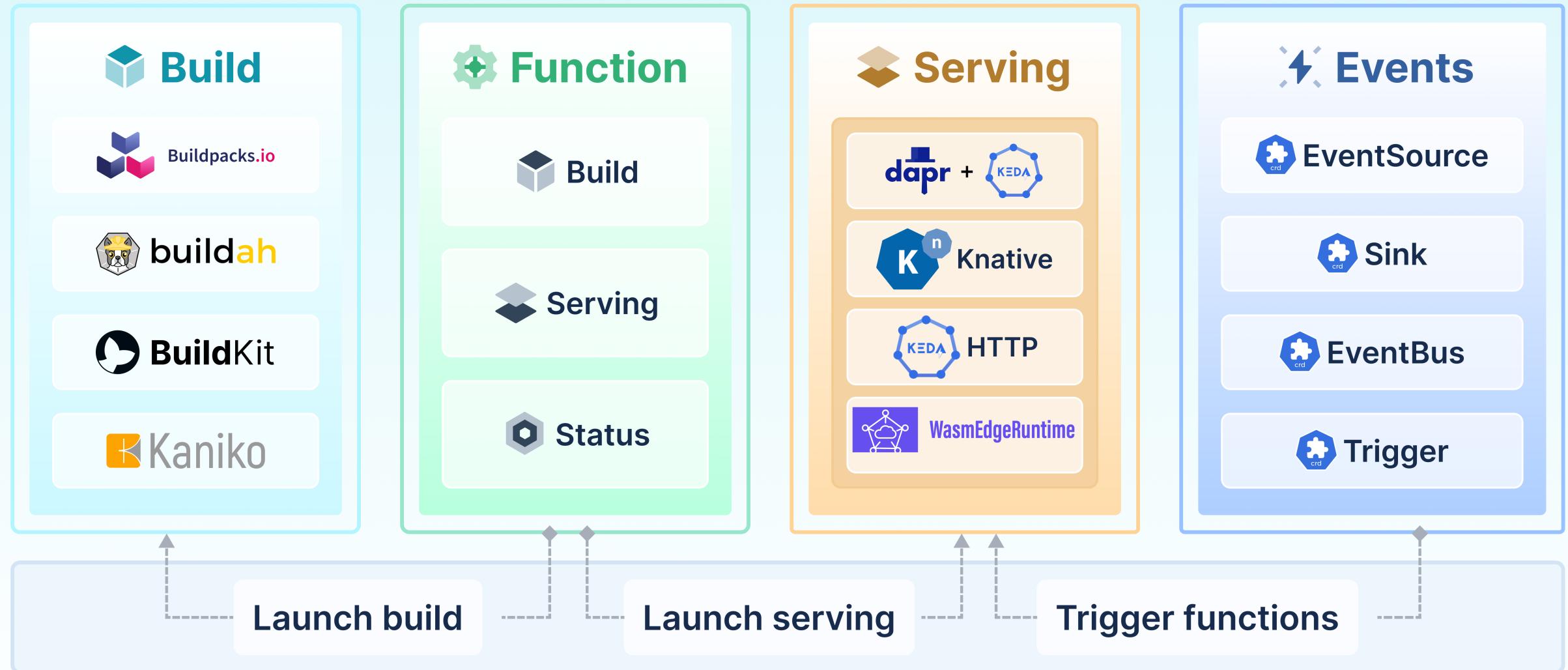
□ 支持 0 与 N 之间的自动水平伸缩

□ 既能运行函数，也能运行 Serverless 应用

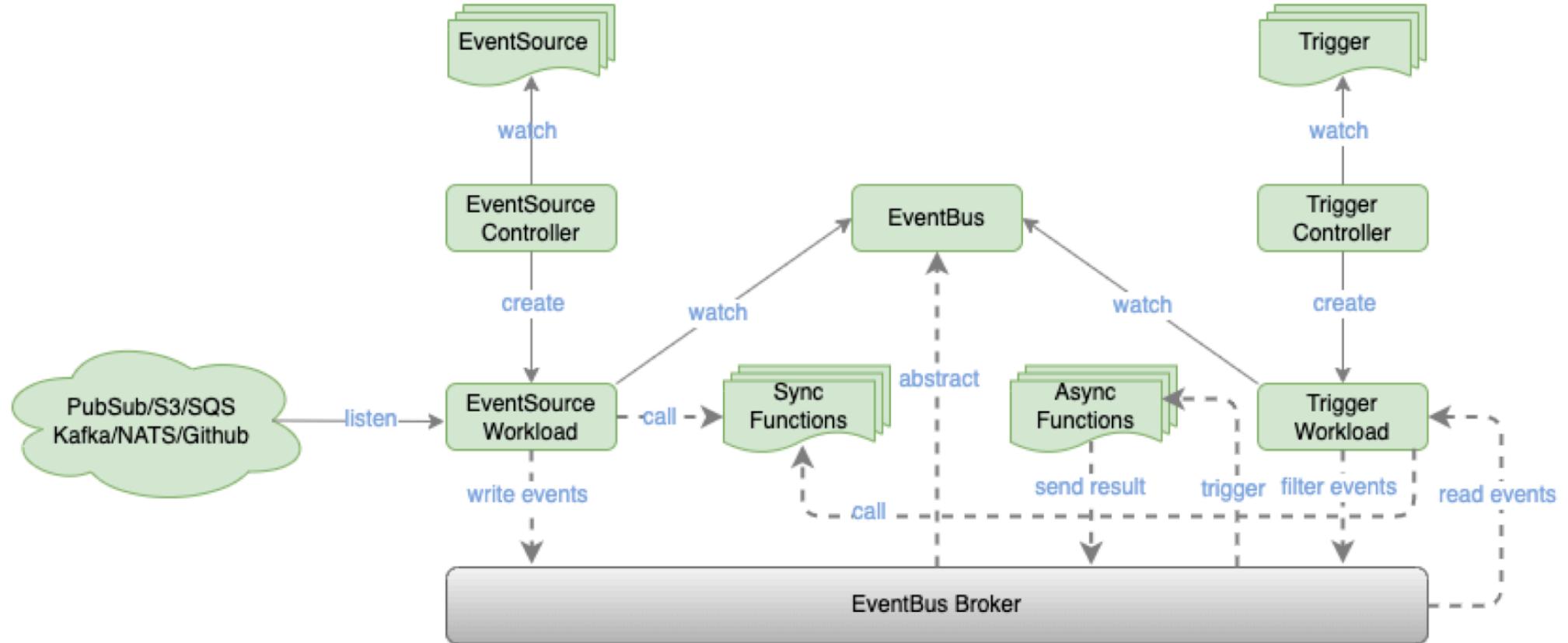
□ 支持 Wasm Runtime

□ 更完善的 CI/CD

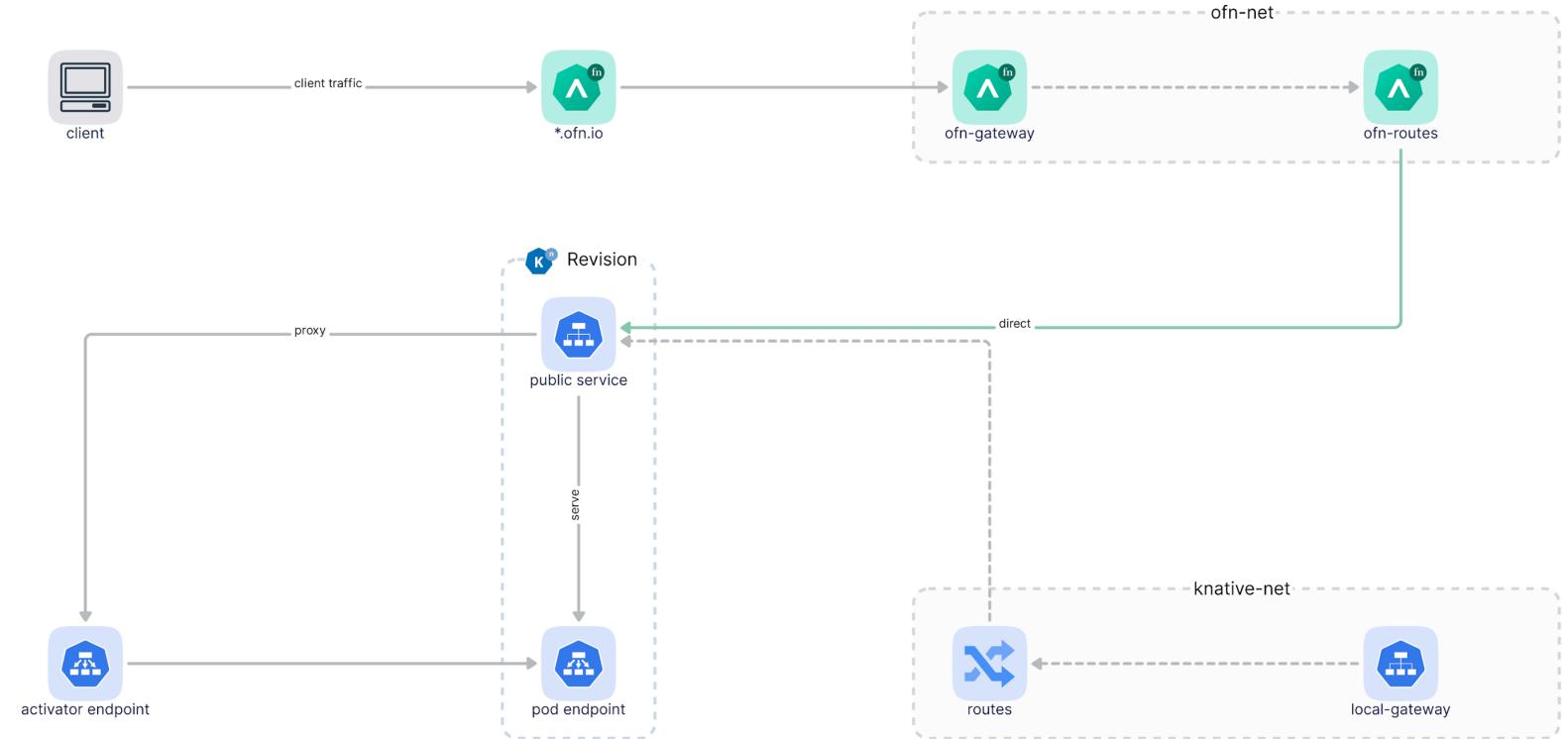
# Cloud-Native FaaS Platform



# OpenFunction Events



# OpenFunction Gateway



- ❑ Based on K8s Gateway API
- ❑ Can use any gateway implementation that implements K8s Gateway API as

OpenFunction Gateway:

- Nginx Kubernetes Gateway
- GKE
- Traefik
- Istio
- Contour
- Apache APISIX

# 为什么引入 OpenFunction Gateway

□ 通过 Knative Gateway 访问同步函数需指定一个由随机串组成的 service url, 不可预测/对用户不友好:

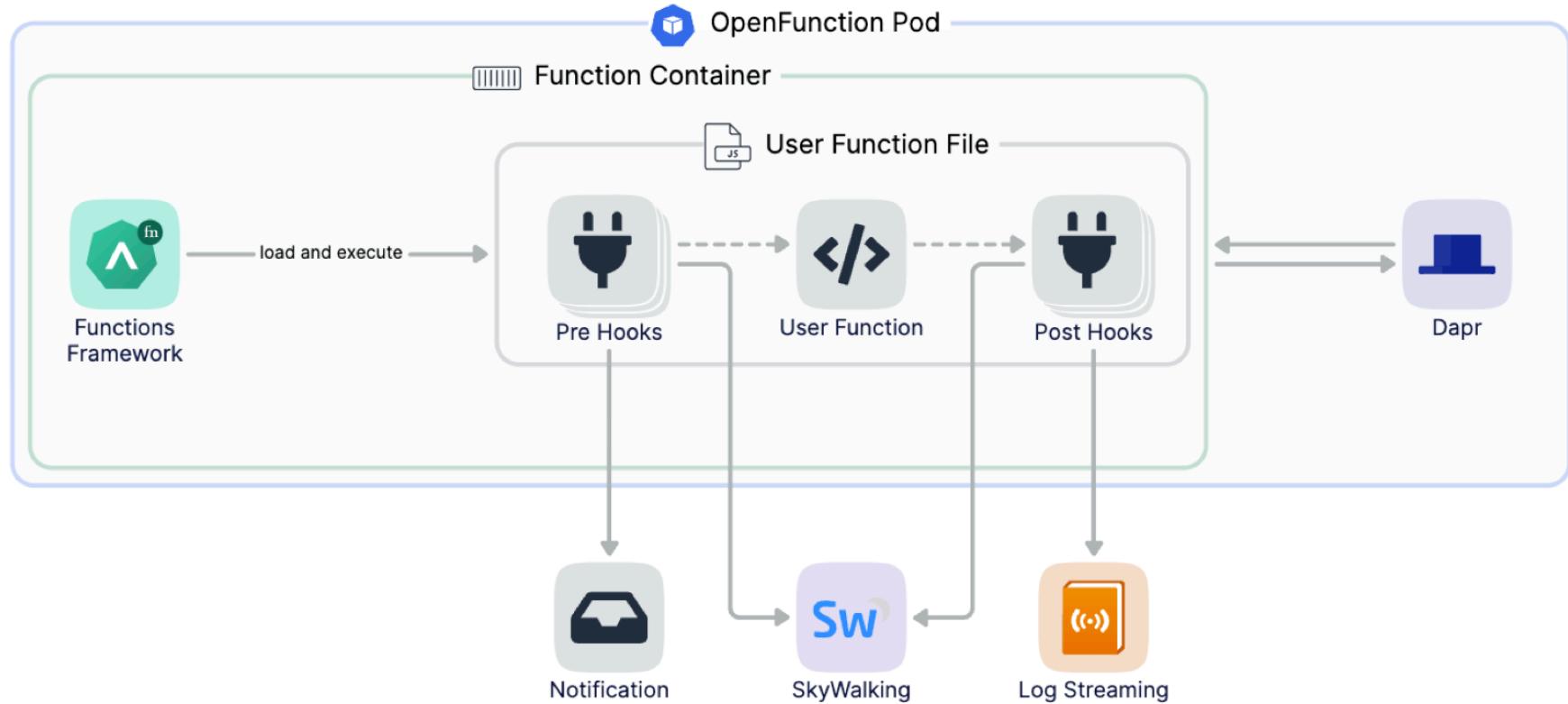
- curl -sv -I "curl http://**serving-q6wdp-ksvc-wk6mv.default.<external-ip>.sslip.io**"

□ 通过 OpenFunction Gateway 访问同步函数, 可通过函数名及 namespace 生成函数访问 url:

- curl -sv -I http://function-sample.default.svc.cluster.local
- curl -sv -I <http://gateway.openfunction.svc.cluster.local/default/function-samle>
- curl -sv -I -HHost:function-sample.ofn.io http://<openfunction gateway IP>/function-samle
- curl http://function-sample.default.172.31.73.53.sslip.io/

```
status:  
addresses:  
- type: External  
  value: http://function-sample.default.172.31.73.53.sslip.io/  
- type: Internal  
  value: http://function-sample.default.svc.cluster.local/
```

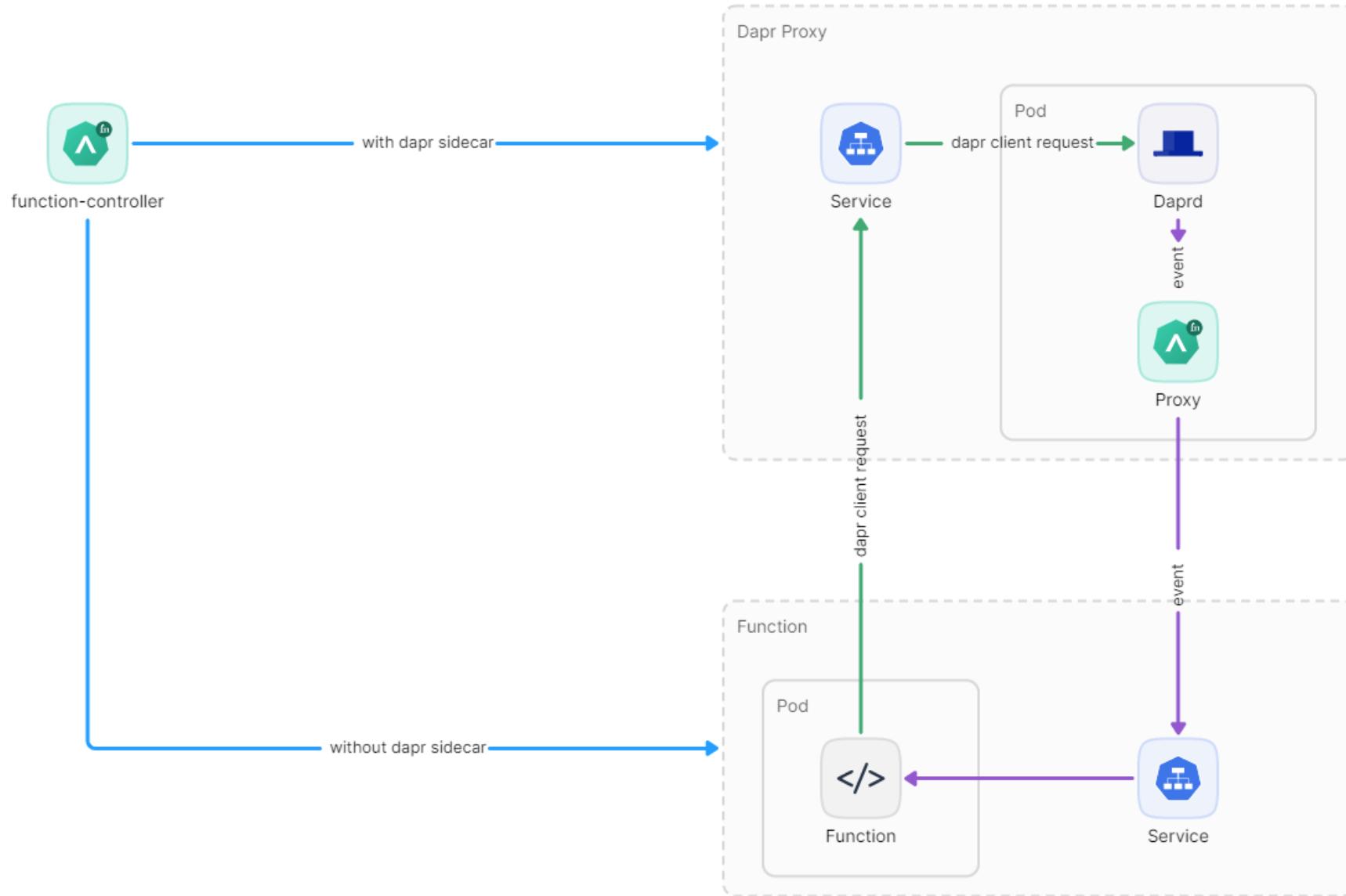
# OpenFunction Functions Framework



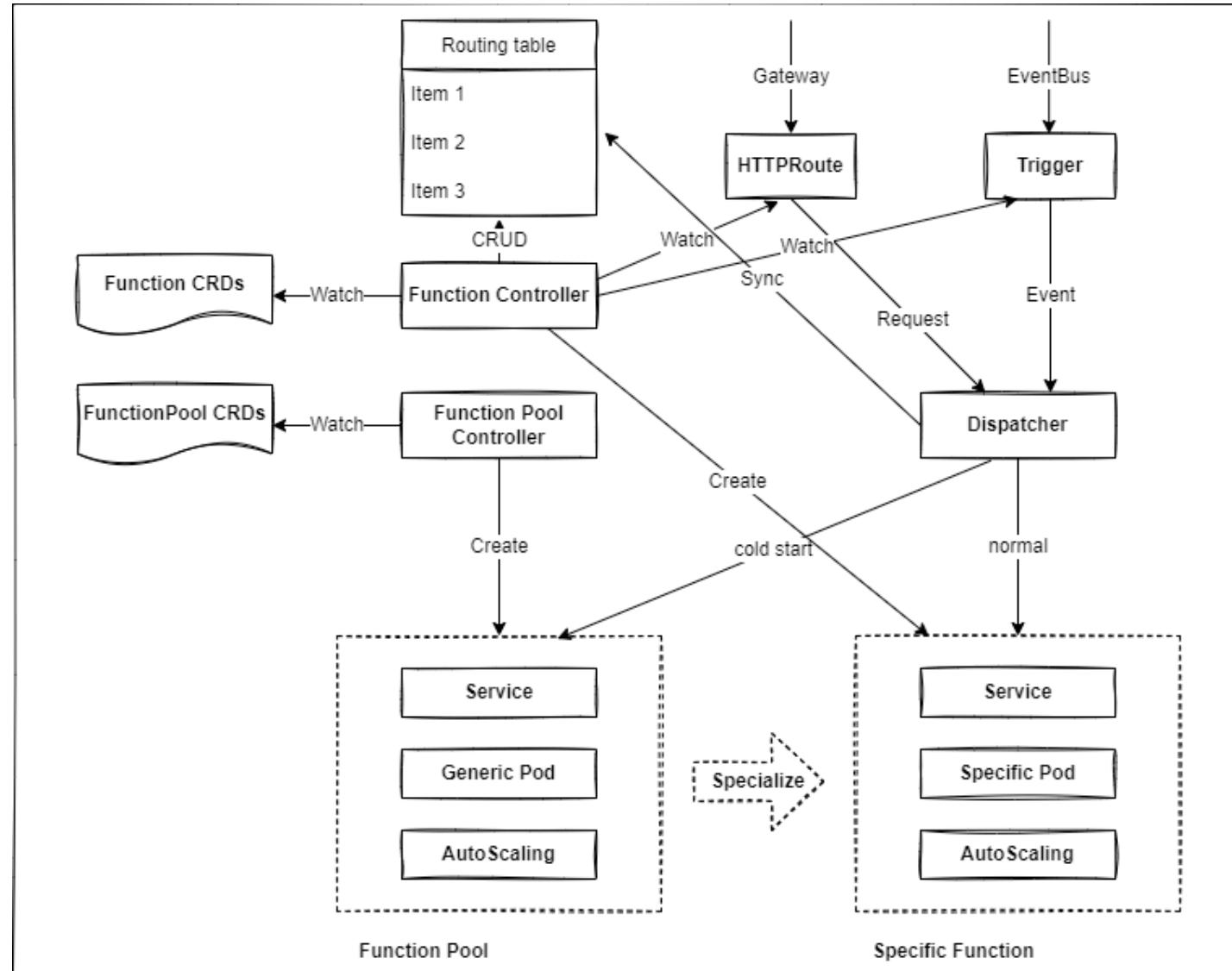
functions-framework: <https://github.com/OpenFunction/functions-framework>  
builder: <https://github.com/OpenFunction/builder>

# 冷启动优化

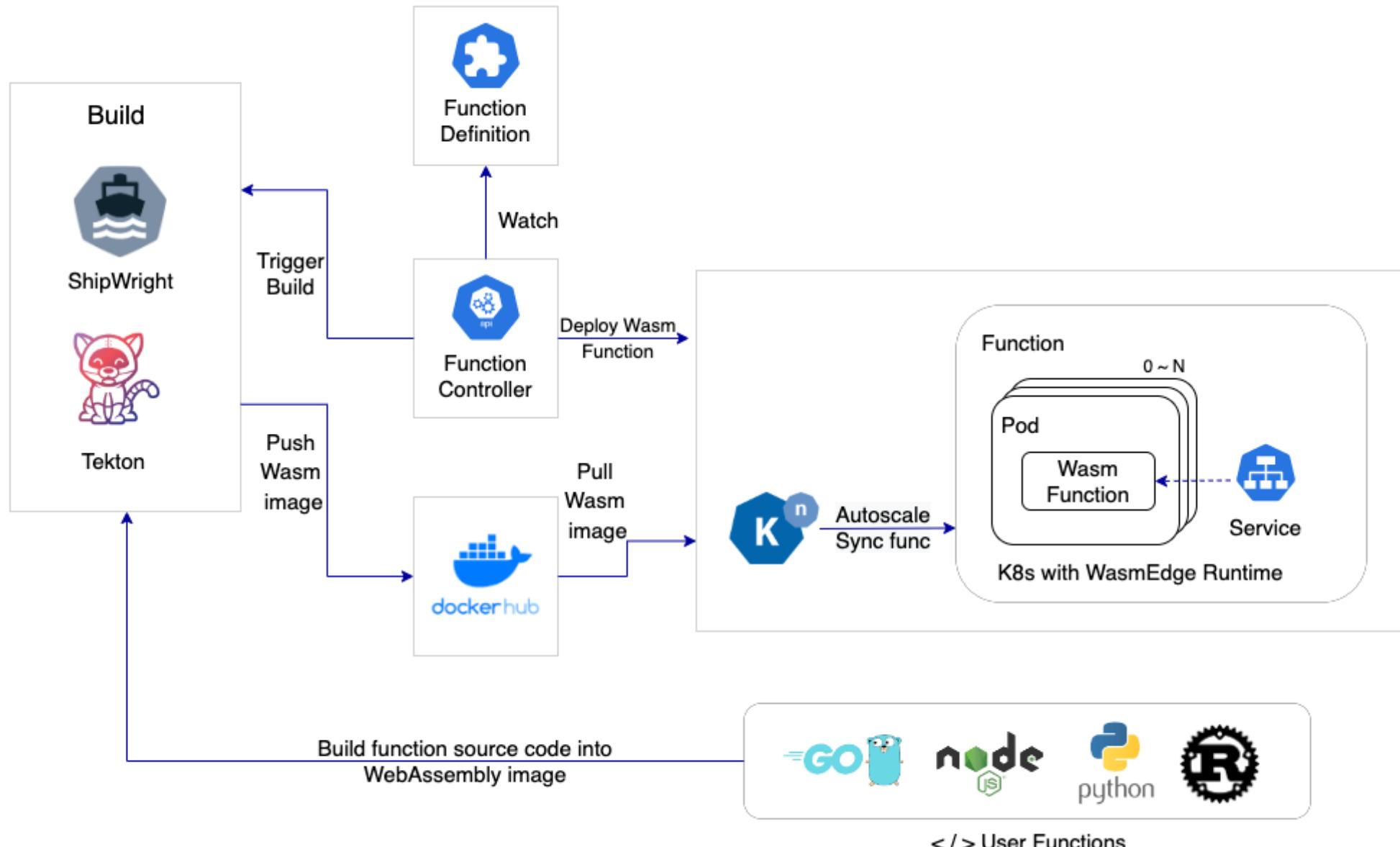
# 从 Dapr sidecar 模式到多副本共用 Dapr 服务的 Proxy 模式



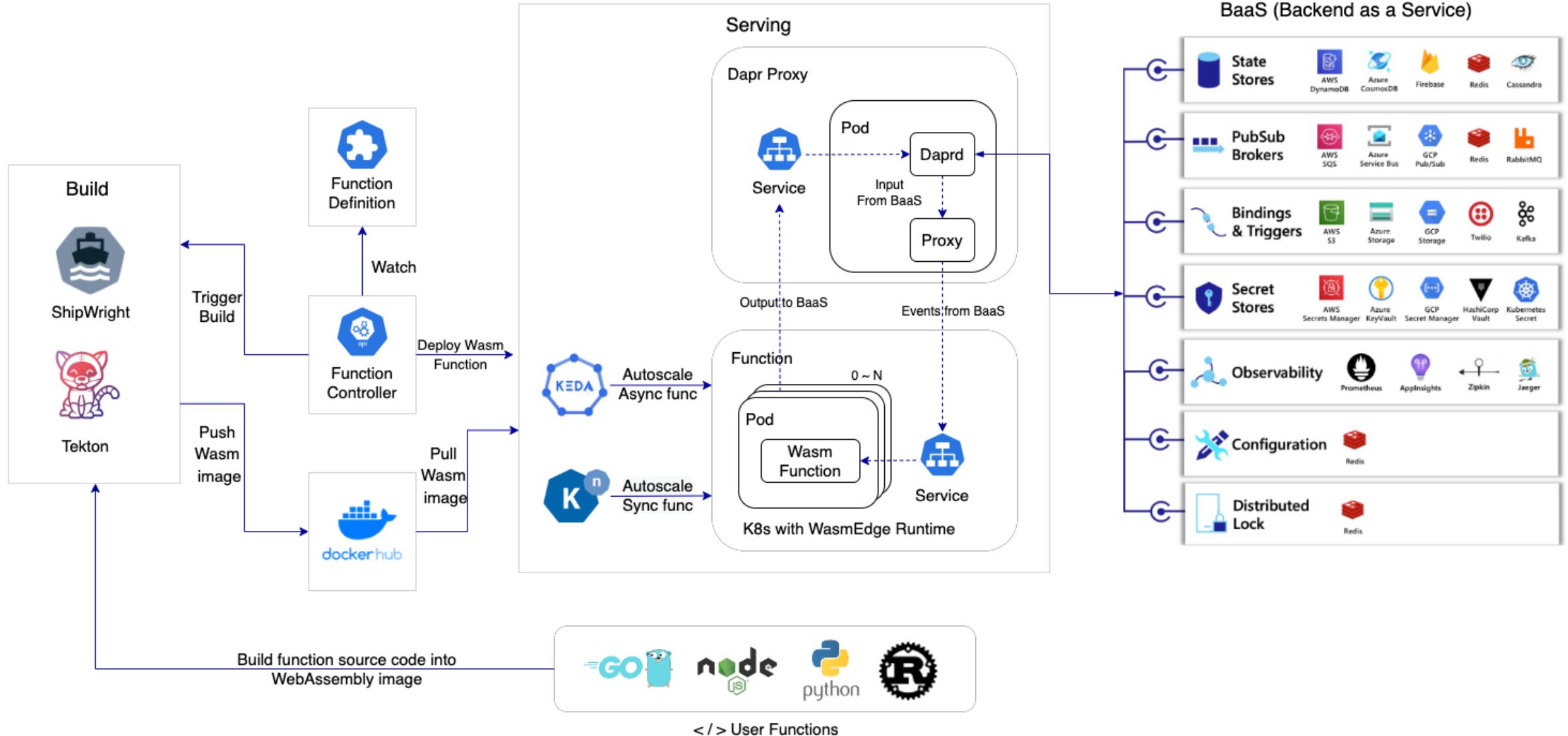
# 基于 Pool 的冷启动优化 (计划中)



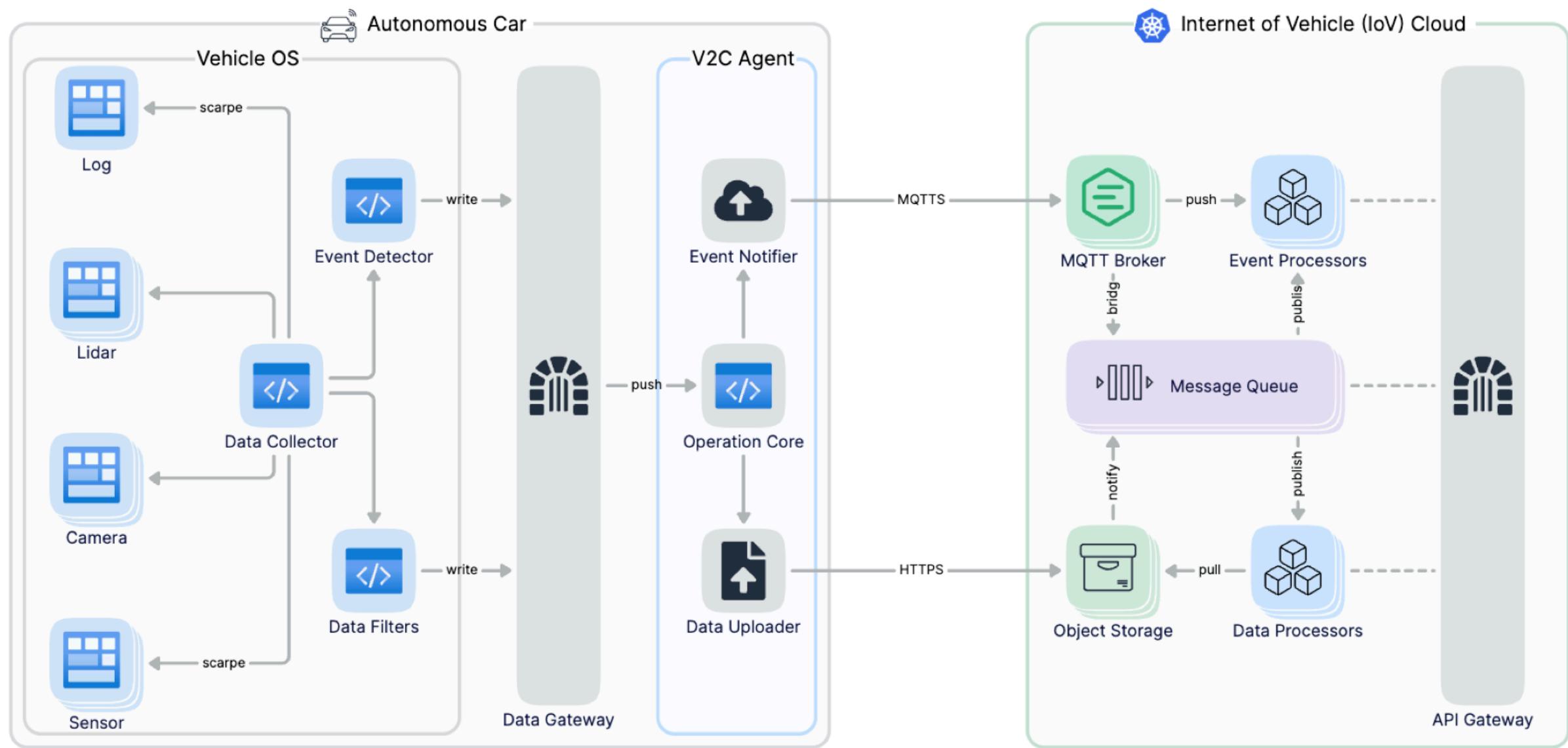
# 支持 WasmEdge 作为 Wasm 运行时 - v1.0.0

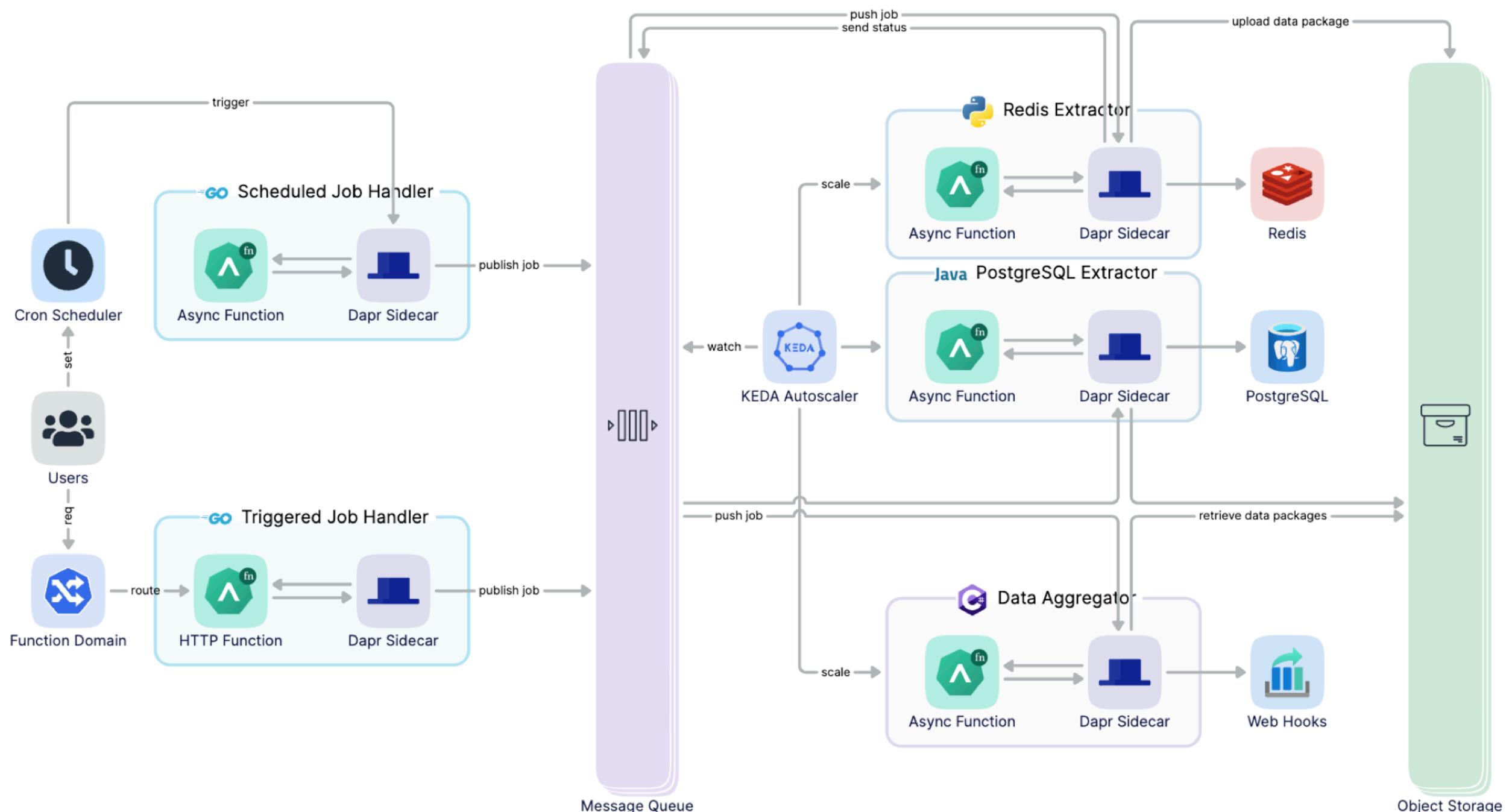


# 支持 WasmEdge 作为 Wasm 运行时 - The Future



# OpenFunction 在自动驾驶领域的应用





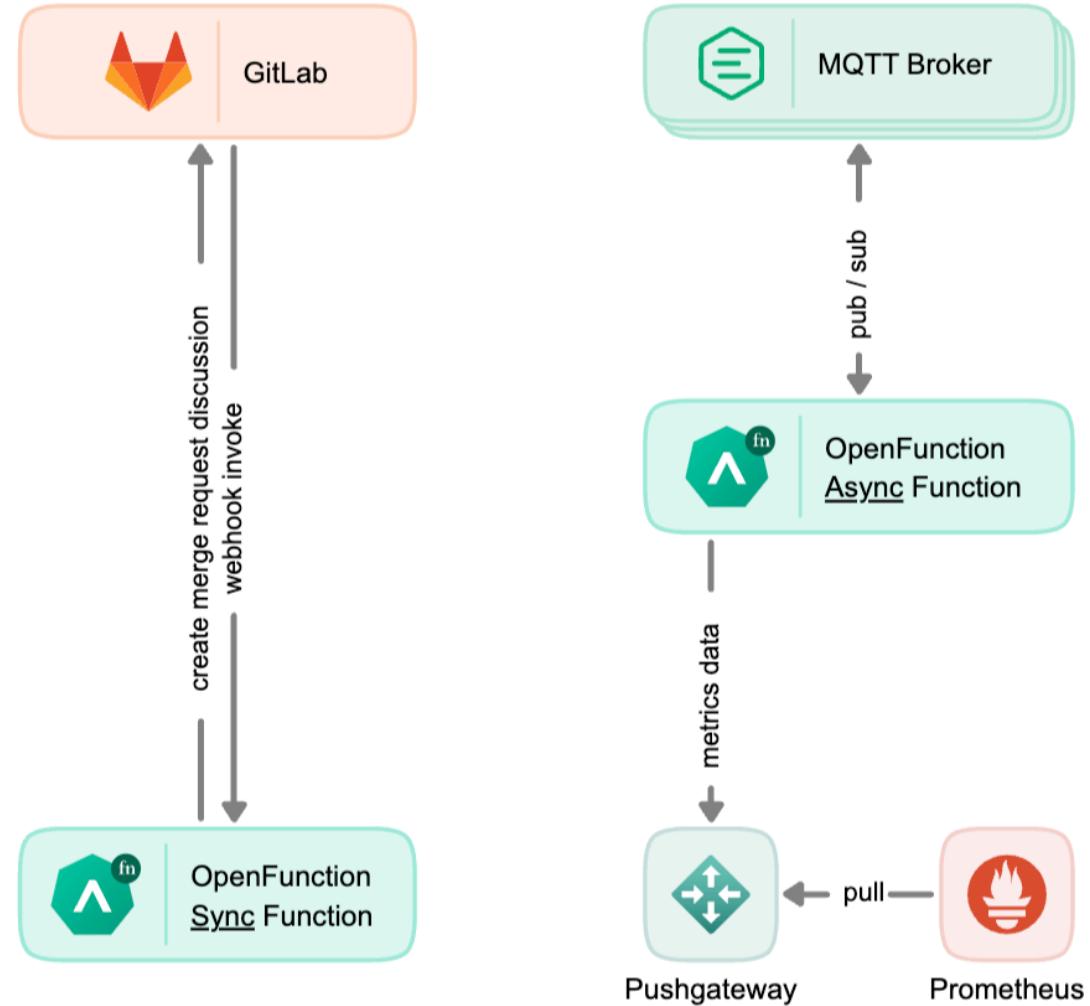
## 异步：消息队列实时数据 ➡ PROMETHEUS 指标

```
import (
    ofctx "github.com/OpenFunction/functions-frame
    "github.com/prometheus/client_golang/prometheus
)

var speedGauge prometheus.Gauge

func init() {
    speedGauge = prometheus.NewGauge(prometheus.Gau
        Namespace: namespace,
        Subsystem: subsystem,
        Name:      "speed",
        Help:      "vehicle speed",
    )
}

func trackingHandler(ctx ofctx.Context, vehicleTr
    // Helper to push data to gauge
    setAndPushGauge(speedGauge, vehicleTracking.Par
    return ctx.ReturnOnSuccess(), nil
}
```



# 为什么自动驾驶需要云厂商中立的 Serverless 平台？

## □ 对于云商中立的需求

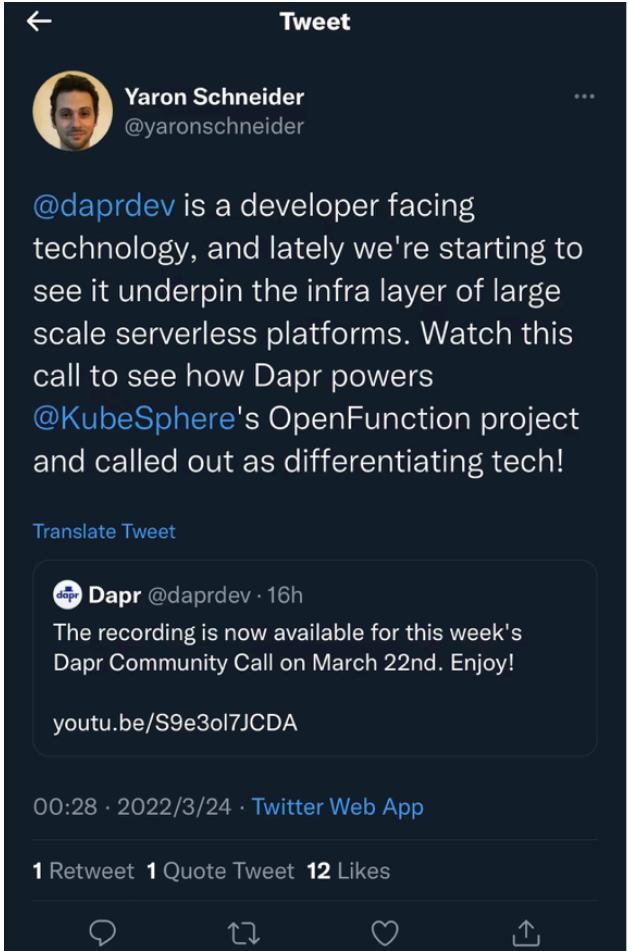
- 不同的客户要求部署到不同的云厂商
- 一些客户的车端数据比较敏感，要求放到和公有云隔离的环境
- 不同的云厂商有不同的后端服务，如果没有一个云厂商中立的云平台，对于同一处理逻辑则需要为对接的每一个云厂商都实现相似的服务

## □ 对于 Serverless 的需求

- 数据处理逻辑多样同时经常变化，来自同一数据源的数据在不同场景下的处理逻辑不尽相同
- 自动驾驶涉及的模块较多，不同的模块由不同的团队负责，需要多语言支持
- 大量车端数据需要实时处理；自动驾驶车辆也有潮汐的特性，数据处理需求有高峰和低谷

# 社区、路线图与 Demo

# 积极参与上游社区



**Tweet**

 **Yaron Schneider** @yaronschneider ...

@daprdev is a developer facing technology, and lately we're starting to see it underpin the infra layer of large scale serverless platforms. Watch this call to see how Dapr powers @KubeSphere's OpenFunction project and called out as differentiating tech!

[Translate Tweet](#)

 **Dapr** @daprdev · 16h

The recording is now available for this week's Dapr Community Call on March 22nd. Enjoy!

[youtu.be/S9e3oI7JCDA](https://youtu.be/S9e3oI7JCDA)

00:28 · 2022/3/24 · Twitter Web App

1 Retweet 1 Quote Tweet 12 Likes

Reply Share Like Retweet



**Tweet**

 **Mark Fussell** @mfussell ...

It is interesting to see Dapr increasingly being used in many serverless Function runtimes

[Translate Tweet](#)

 **Yaron Schneider** @yaronschneider · 4h

@daprdev is a developer facing technology, and lately we're starting to see it underpin the infra layer of large scale serverless platforms. Watch this call to see how Dapr powers @KubeSphere's OpenFunction project and called out as different...

00:50 · 2022/3/24 · Twitter Web App

2 Retweets 6 Likes

Reply Share Like Retweet

OpenFunction 在 Dapr 社区会议的分享: <https://www.youtube.com/watch?v=S9e3oI7JCDA?start=188>

Dapr case study: OpenFunction <https://blog.dapr.io/posts/2022/05/14/how-dapr-helps-to-build-a-cloud-agnostic-faas-platform/>

### □ 函数框架

- 支持 Dapr State Management 与 Dapr Workflow
- 支持更多语言的异步函数框架包括 Python、Rust
- 支持将 Java 函数编译成 Native 程序运行在 Quarkus 环境中

### □ 函数运行时

- 实现 Serverless 工作流 (Workflow)
- 预研基于 Pod Pool 的冷启动优化方案
- 支持 OpenTelemetry 作为另一个函数追踪方案

### □ 用户工具

- 增加 OpenFunction 控制台

### □ 与 AIGC 结合

- .....

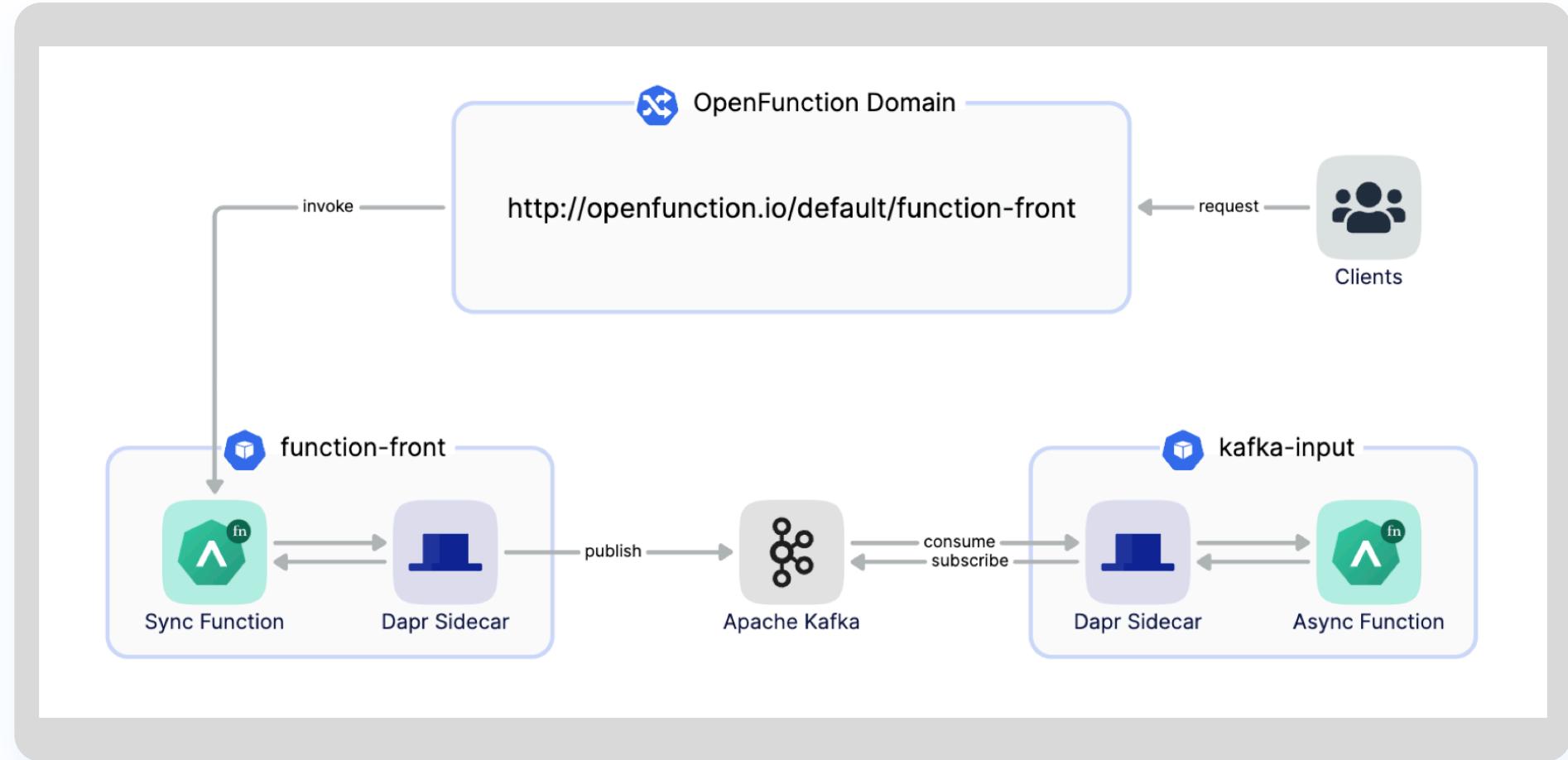
### □ 越来越多的社区贡献者

- 主要 Maintainer 来自 KubeSphere 团队
- SkyWalking PMC 成员 @arugal 实现了 SkyWalking 和 OpenFunction Go Functions Framework 的集成
- 驭势科技 (UISEE) @webup @kehuili 以及印度的贡献者正在参与 Node.js 和 Python Functions Framework 的开发
- SAP @lizzze 参与 functions-framework-go 的开发
- 来自阿里云、微众银行等社区贡献者也在积极参与
- 乌克兰的贡献者在帮忙维护 openfunction.dev

### □ 越来越多的公司开始采用

- 国内某电信公司采用 OpenFunction 构建云函数计算平台
- 驭势科技 (UISEE) 采用 OpenFunction 处理车云数据
- 微众银行
- 某证券公司
- 喜马拉雅
- 云学堂

# 同步函数触发异步函数示例



同步函数: <https://github.com/OpenFunction/samples/tree/main/functions/knative/with-output-binding>  
异步函数: <https://github.com/OpenFunction/samples/tree/main/functions/async/bindings/kafka-input>

## 更多函数示例

Wasm 函数:

<https://github.com/OpenFunction/samples/tree/main/functions/knative/wasmedge/http-server>

Redis state store:

<https://github.com/OpenFunction/java-samples/blob/main/src/main/resources/functions/redis-state-store.yaml>  
<https://github.com/OpenFunction/java-samples/blob/main/src/main/java/dev/openfunction/samples/StateStore.java>

KEDA HTTP Engine:

<https://github.com/OpenFunction/OpenFunction/blob/main/config/samples/function-kedahttp-sample-serving.yaml>

# 参与 OpenFunction 社区

OpenFunction: <https://github.com/OpenFunction/OpenFunction>

Website: <https://openfunction.dev/>

Samples: <https://github.com/OpenFunction/samples>

加入 OpenFunction 微信群请回复 OpenFunction:



小KK@kubesphere.io

湖南 岳阳



扫一扫上面的二维码图案，加我为朋友

Discord: <https://discord.gg/hG3yc6uAyC>

Slack: <https://cloud-native.slack.com/archives/C03ETDMD3LZ>

Email: [benjamin@kubesphere.io](mailto:benjamin@kubesphere.io)



**LESS CODE  
MORE FUNCTIONS**