

Press Space for next page →

OpenFunction 101 

OVERVIEW



Haili Zhang

CNCF OpenFunction TOC Member,
KubeSphere Ambassador.

Cloud Platform Director of UISEE©, a
leading autonomous driving company.

Cloud Native focuses: Kubernetes, DevOps,
Observability, Service Mesh, Serverless.



webup



zhanghaili0610




haili.zhang@uisee.com



ServiceUP · 语雀

Table of Content

- Learnings from CNCF and Datadog Reports
- Why need cloud agnostic Serverless platform?
- How to build a cloud agnostic Serverless platform?
- OpenFunction: Less codes. More Functions!
- Architecture Overview of OpenFunction
-  Use Case: Online Data Processing

CNCF Survey 2021

- Container Adoption and Kubernetes have **truly gone mainstream** – usage has risen across organizations globally, particularly in large businesses.
- According to CNCF's respondents, **96% of organizations are either using or evaluating Kubernetes** – a record high since our surveys began in 2016.
- Kubernetes has demonstrated impressive growth over the past 12 months with **5.6 million developers** using Kubernetes today.



CNCF

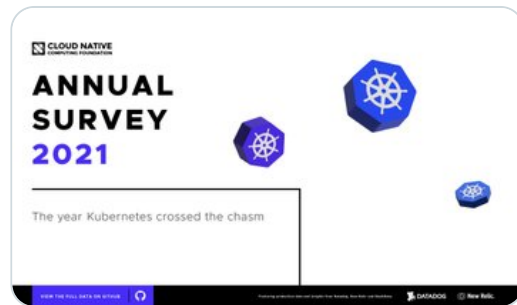
@CloudNativeFdn · Follow



[NEWS] CNCF sees record Kubernetes and container adoption in the 2021 Cloud Native Survey 📈

A record number of organizations are using or evaluating @kubernetesio and are now beginning to move up the stack.

Read more: [cncf.io/announcements/...](https://cncf.io/announcements/)



1:01 AM · Feb 11, 2022



90



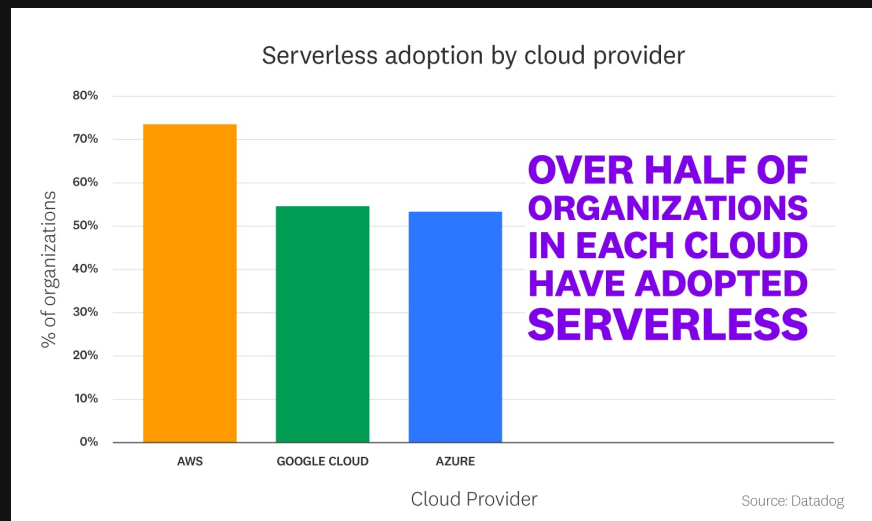
Reply



Copy link

Serverless Boosts 🚀

"As the popularity of FaaS products like AWS Lambda continues to grow, we have also seen a significant increase in adoption of other types of serverless technologies offered by Azure, Google Cloud, and AWS."



Datadog, Inc. ✓

@datadoghq · [Follow](#)



Datadog's 2022 State of Serverless report is now live! We examined telemetry data from thousands of companies' serverless applications to uncover nine eye-opening facts about how [#serverless](#) offerings from AWS, Azure, and Google Cloud are used today: dtdg.co/serverless-2022

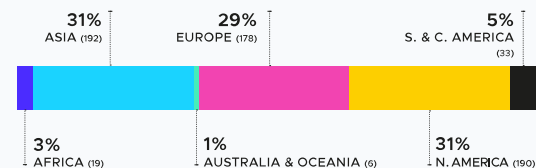


Serverless Platform War?

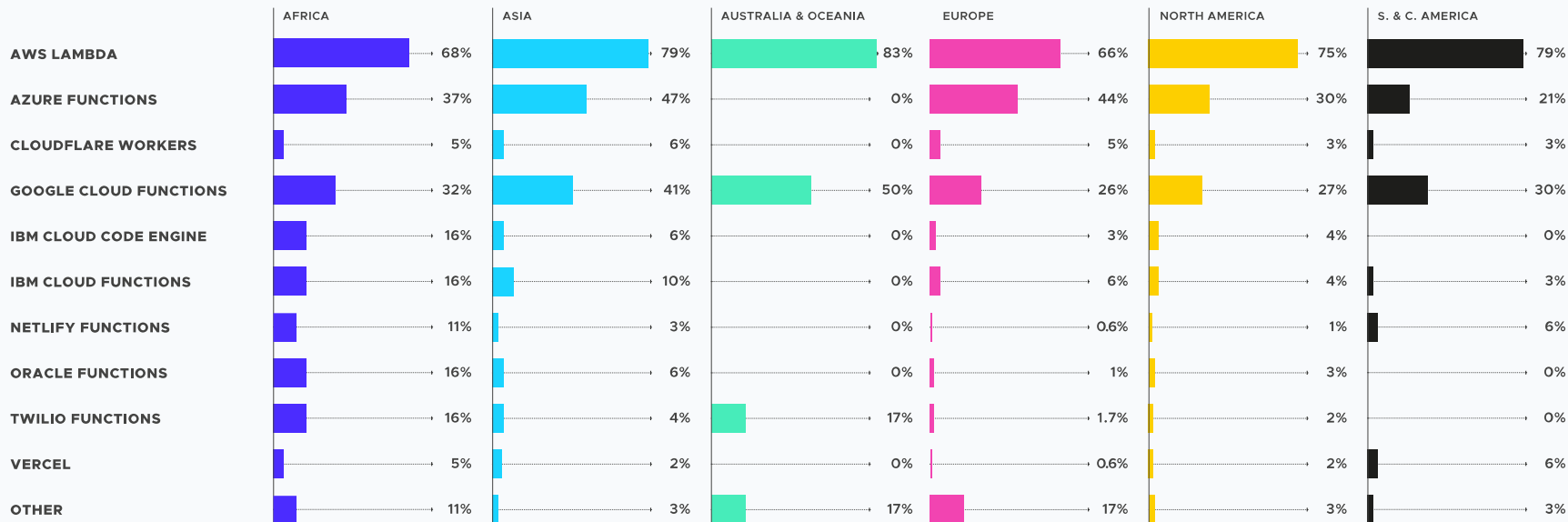
"Hosted platforms (75%) are the most popular"

618
RESPONDENTS

respondents could select
more than one platform



IF YOUR ORGANIZATION IS USING SERVERLESS VIA A 'HOSTED PLATFORM', WHICH HOSTED SERVERLESS PLATFORMS DO YOU USE?



Why need cloud agnostic Serverless platform?

Kubernetes brings the possibility of cloud-agnostic: multiple, distributed cloud.

But it's difficult to be cloud-agnostic for Serverless

- Each cloud provider has its own Serverless platform
- Further more, these platforms are coupled with their own cloud backend services

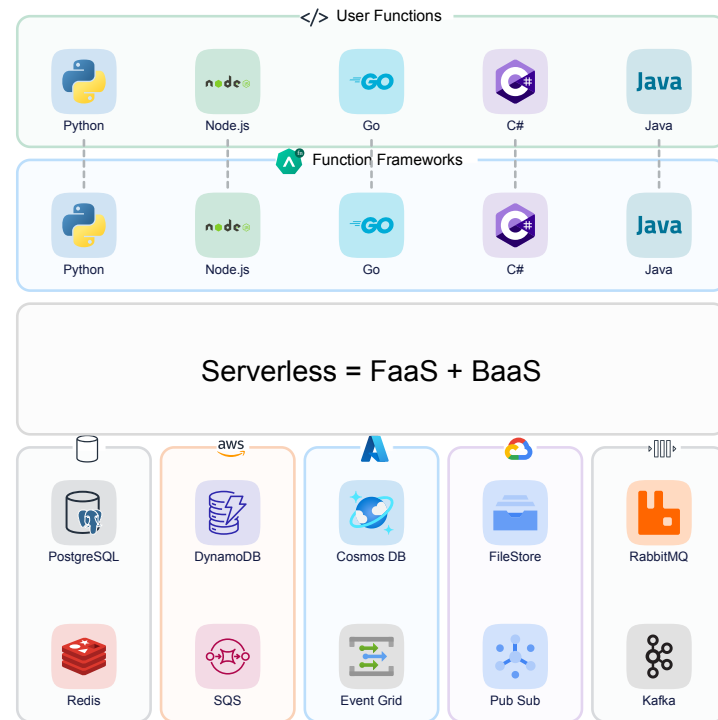
Is it possible to build a cloud agnostic Serverless platform? 🤔

What is Serverless?

Let's make core concepts clear first.

"Serverless computing = FaaS + BaaS." [1]

- Cloud functions — packaged as FaaS (Function as a Service) offerings — represent the core of serverless computing
- Cloud platforms also provide specialized serverless frameworks that cater to specific app requirements as BaaS (Backend as a Service) offerings

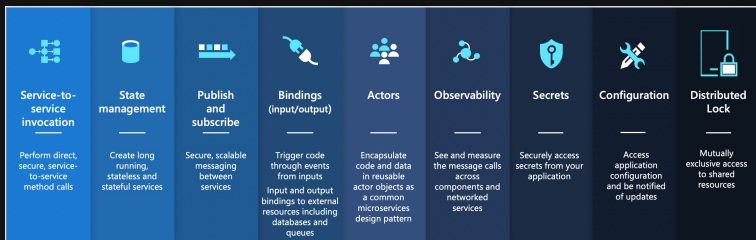


1. Cloud Programming Simplified: A Berkeley View on Serverless Computing. Feb 10, 2019.

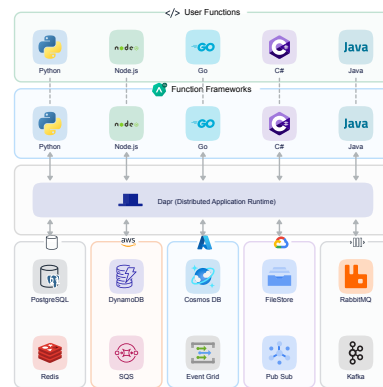
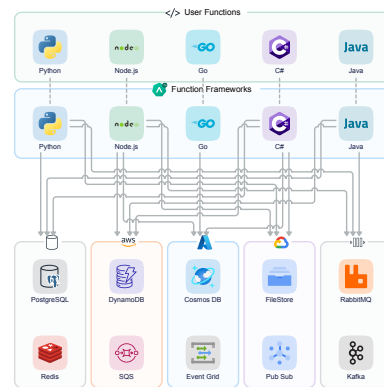
How to construct?

We need adapt various backend services.

Dapr **decouples** the distributed apps with **underlying backend services**.



Dapr provides you with APIs that abstract away the complexity of common challenges developers encounter regularly when building distributed applications. These API building blocks can be leveraged as the need arises – use one, several or all to develop your application faster and deliver your solution on time.



How to construct?

We need achieve "zero-scale".

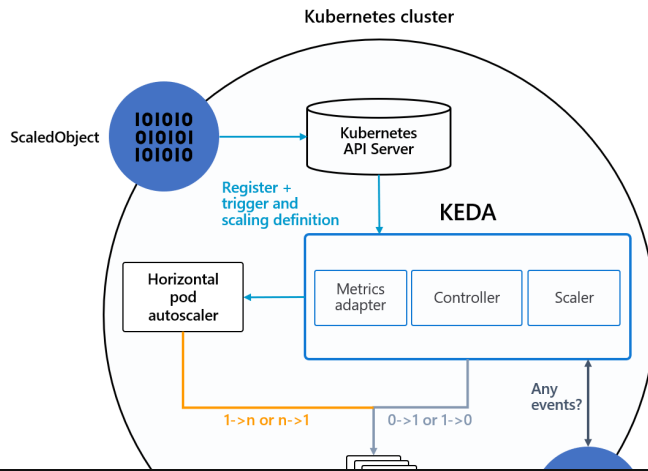
KEDA is a Kubernetes-based Event Driven Autoscaler. With KEDA, you can **drive the scaling of any container in Kubernetes based on the number of events** needing to be processed.

- KEDA is a single-purpose and lightweight component that can be added into any Kubernetes cluster.
- KEDA works alongside standard Kubernetes components like the Horizontal Pod Autoscaler and can extend functionality without overwriting or duplication.



Architecture

The diagram below shows how KEDA works in conjunction with the Kubernetes Horizontal Pod Autoscaler, external event sources, and Kubernetes' [etcd](#) data store:





OpenFunction is a cloud-native open source FaaS (Function as a Service) platform aiming to let you focus on your business logic without having to maintain the underlying runtime environment and infrastructure.

You can concentrate on developing business-related source code in the form of functions.



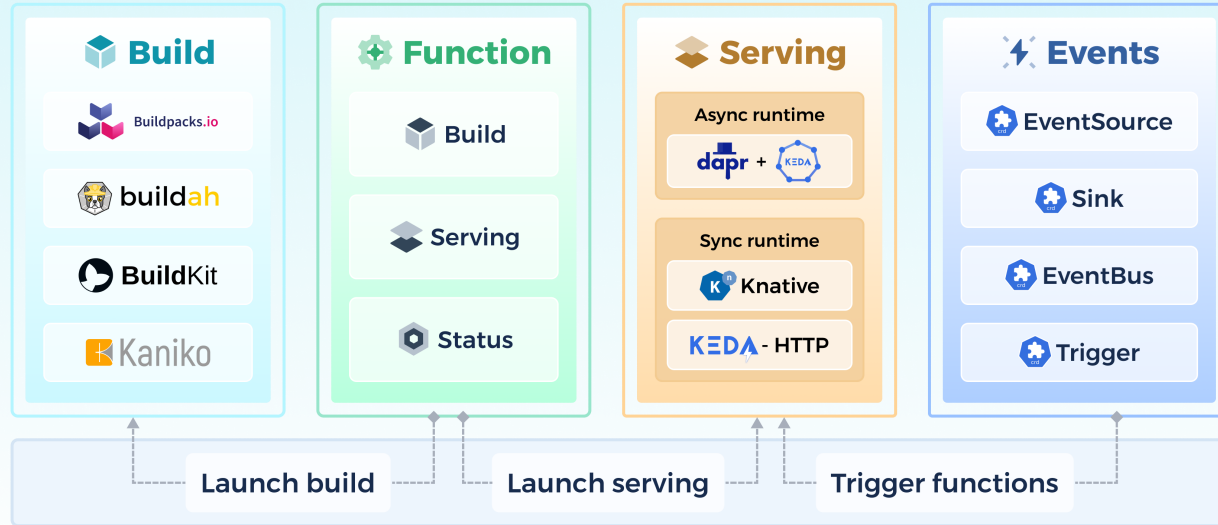
Keynote: Landscape Sustainability: The Pillars of Cloud Native Growth

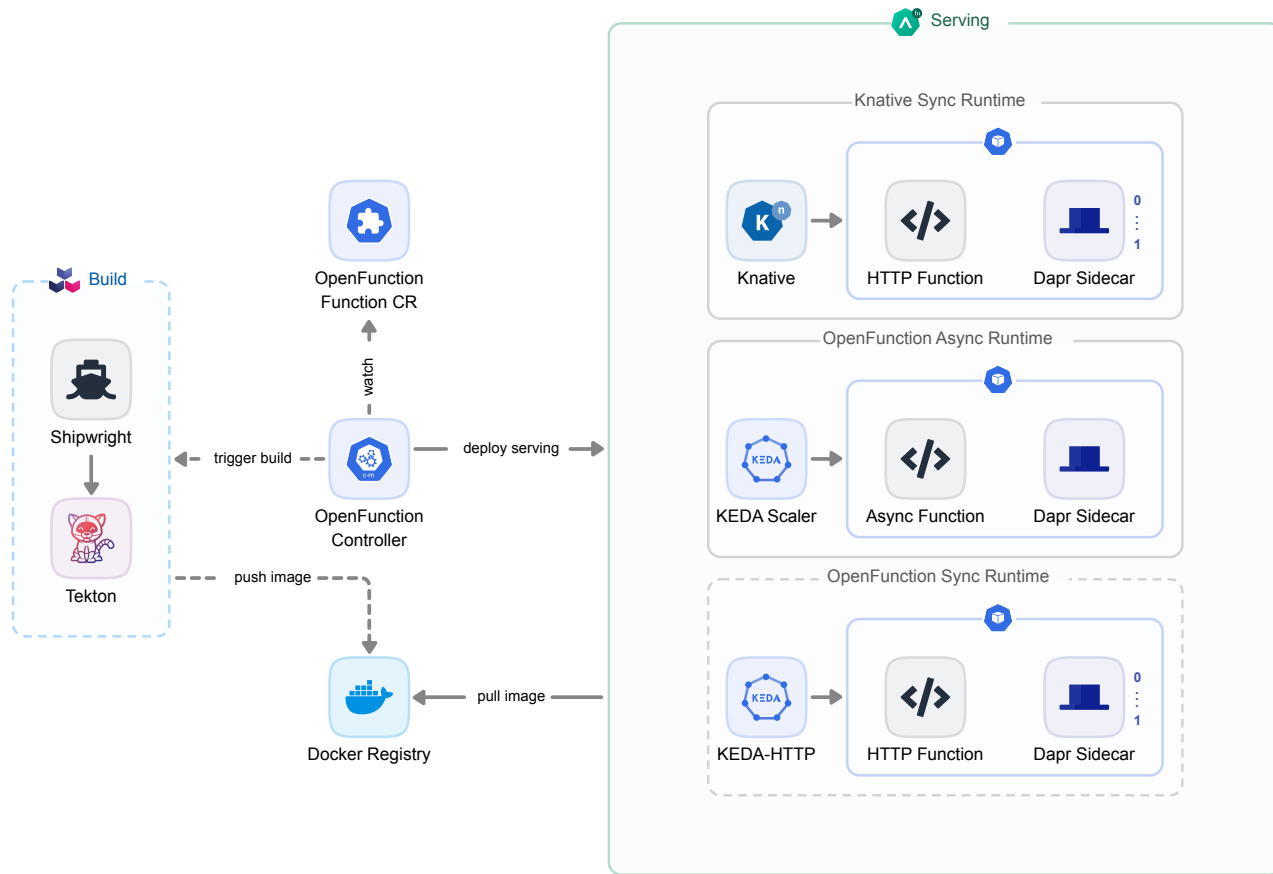
Dave Zolotusky, Software Engineer, Spotify & Katie Gamanji, Senior Kubernetes Field Engineer, Apple



Architecture Overview of OpenFunction

Cloud-Native FaaS Platform

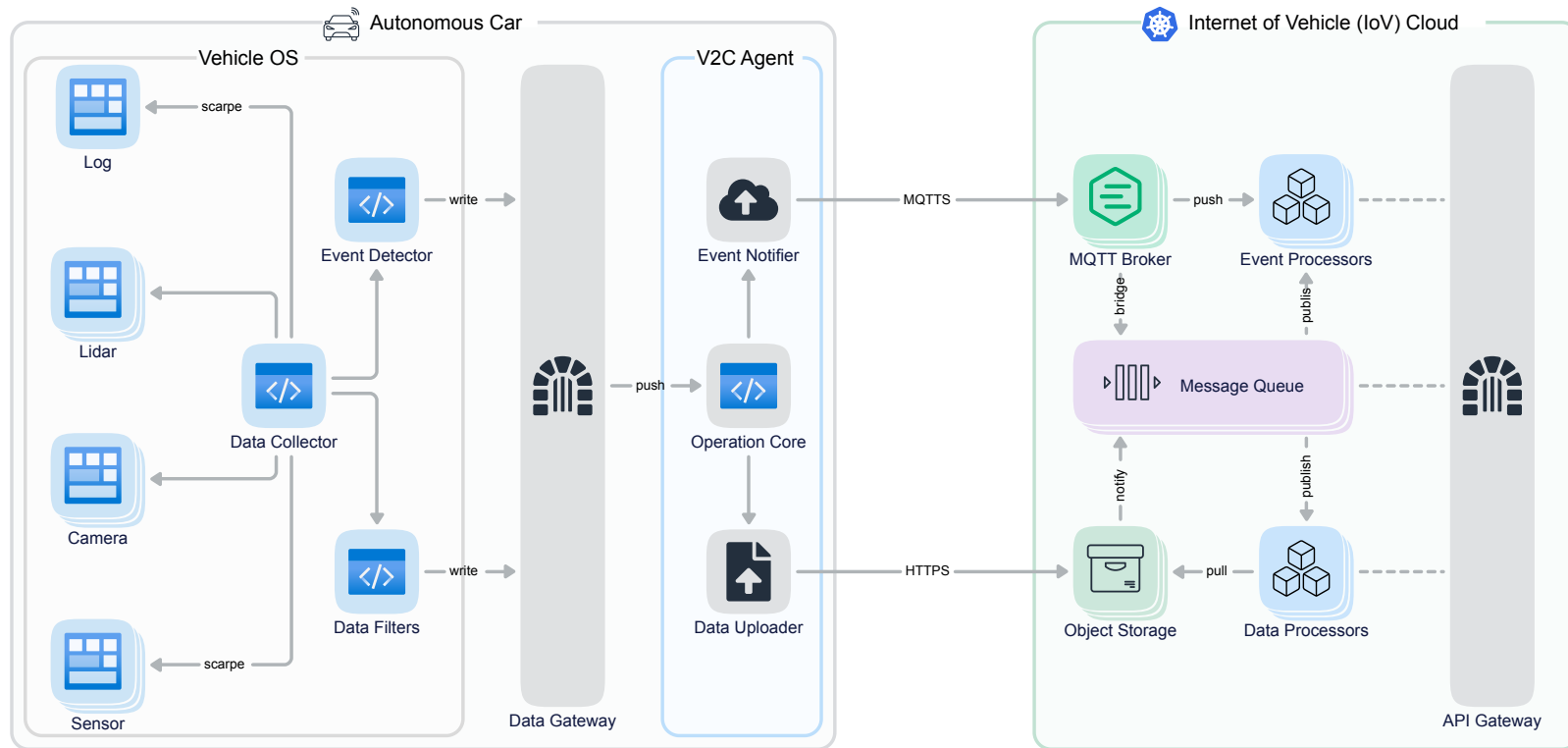






Use Case: Online Data Processing

Using OpenFunction in Autonomous Driving



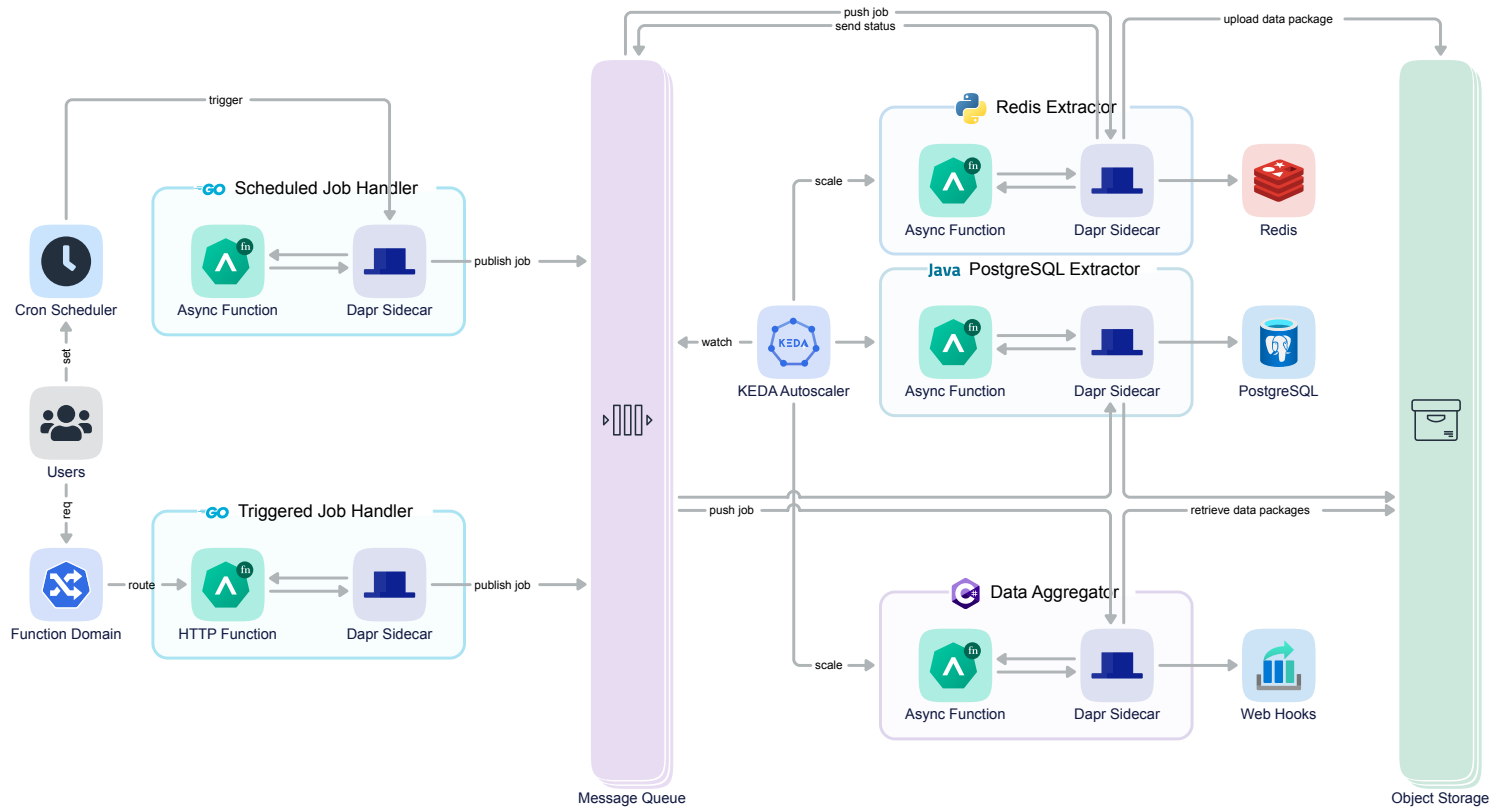
Why need a cloud agnostic Serverless platform?

WHY CLOUD AGNOSTIC

- Different customers require to use different cloud providers
- Some customers' vehicle data is sensitive and are required to be isolated from public clouds

WHY SERVERLESS

- Autonomous driving has so many application scenarios, there're requirements to use different processing logics for different scenarios even for the same data source
- Data types and processing modules are complex, multi-language support is required
- Massive events need to be handled in real time
- Data processing logics tend to be changed frequently based on rapidly changing requirements



Dapr Matters

- Data from different sensors and processing modules are complex
- Different cloud providers have different backend services that may result in duplicated implementations over the same data processing logic
- Pub/Sub and Bindings make it easy to set up event-driven architecture for data processing with async functions

KEDA Matters

- AI drivers need to retain human working habits to a certain extent, so there are obvious peaks and valleys in traffic based on working schedules
- For security reasons, data refresh frequently and sensors produce massive data, so workload should be scalable
- The usage of compute resources directly affects project cost

EXPLORE MORE

Documentation | Contribution