Observability in CloudNative

SkyWalking, Istio
and Kubernetes

Gao Hongtao

Table of Contents

01

Glance at SW & CN

02

Observing Istio

03

Running on Kubernetes

04

Future

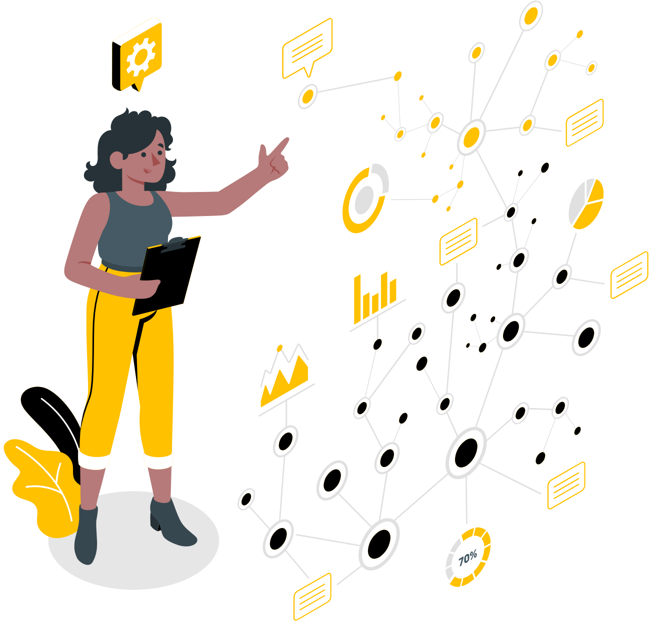
Gao Hongtao

The engineer of tetrate.io and the former Huawei Cloud expert.

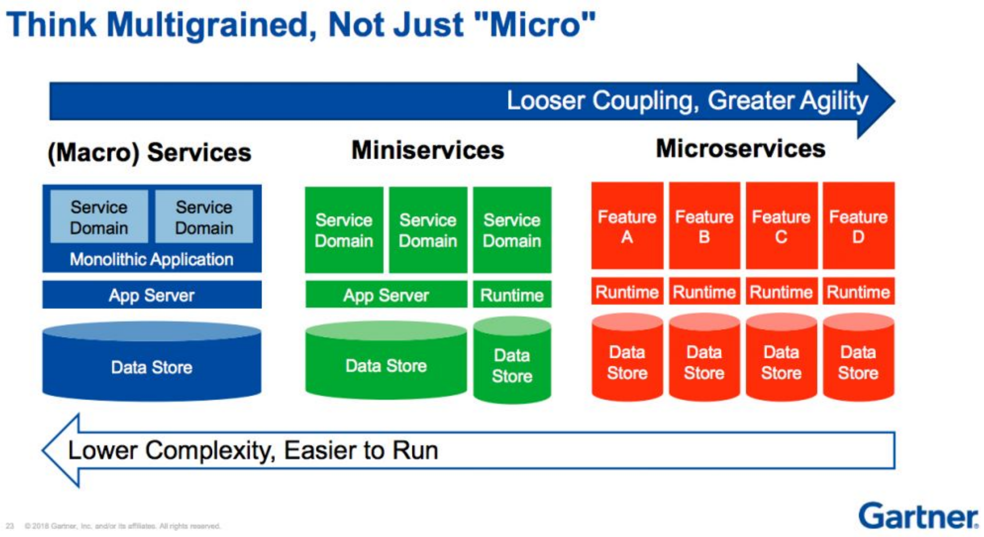
One of PMC members of Apache SkyWalking & ShardingShpere

Glance At SW & CN

01



Services of Cloud Native



Think Multigrained, Not Just "Micro"

Looser Coupling, Greater Agility

(Macro) Services

Service Domain | Service Domain

Monolithic Application

App Server

Data Store

Miniservices

Service Domain | Service Domain | Service Domain

App Server | Runtime

Data Store | Data Store

Microservices

Feature A | Feature B | Feature C | Feature D

Runtime | Runtime | Runtime | Runtime

Data Store | Data Store | Data Store | Data Store

Lower Complexity, Easier to Run

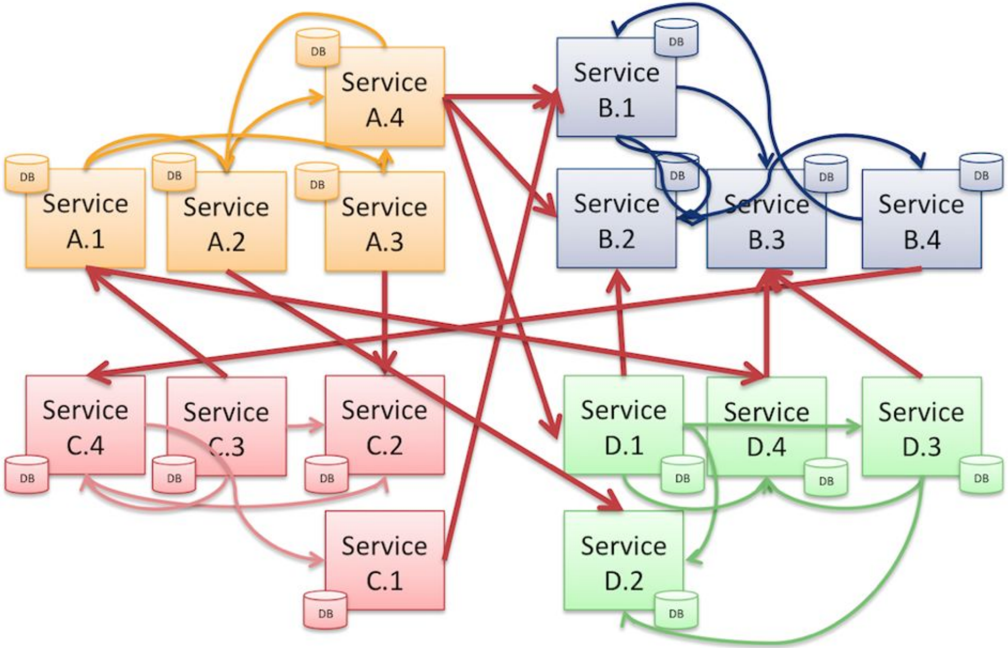
23

Gartner

- Independent
- Loosely coupled
- Multiple layers

Challenges

E2E
DevOps, CI & CD
Keeping tracks



Logging

By Tracing
Satellite

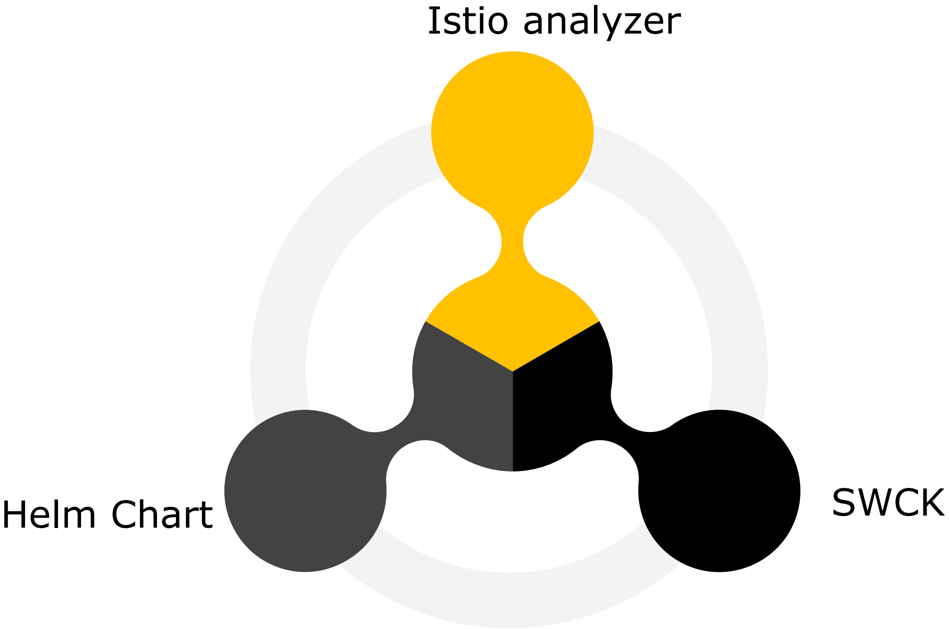
Metrics

Meter & MAL
system

Tracing

SW Native Agent
Nginx & Envoy

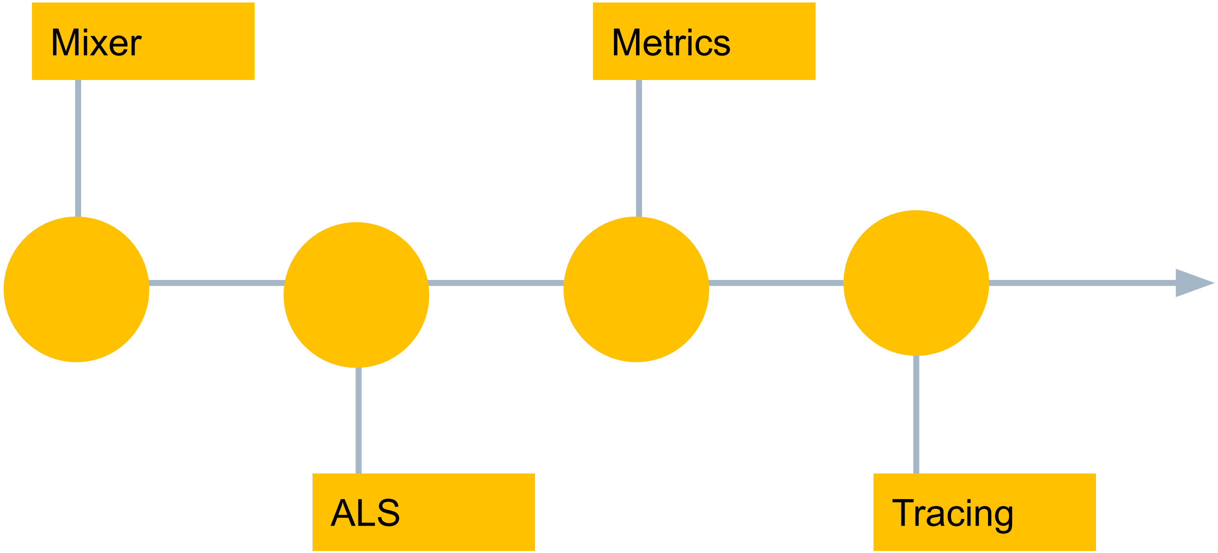
SW for CN



Observing Istio

02

Trust



Mixer

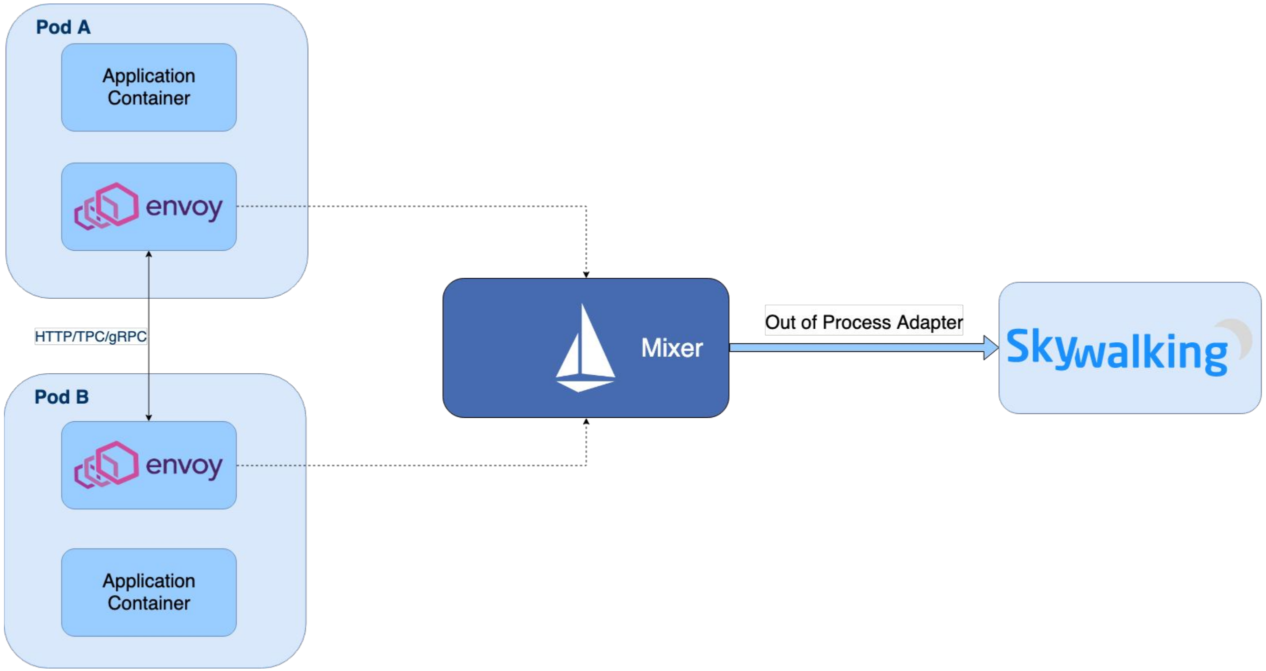
ALS

Metrics

Tracing

Mixer Adapter

Out of Process Adapter

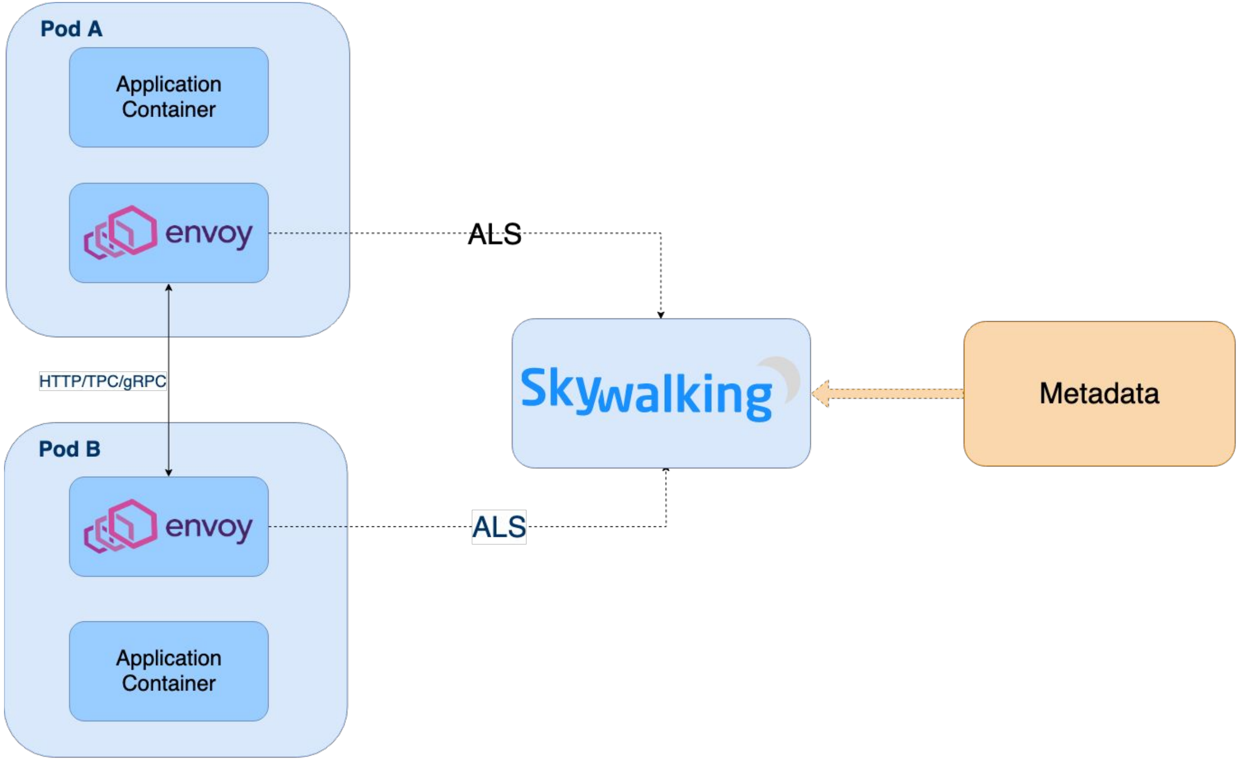


Access Log Service

gRPC service
pushes access log

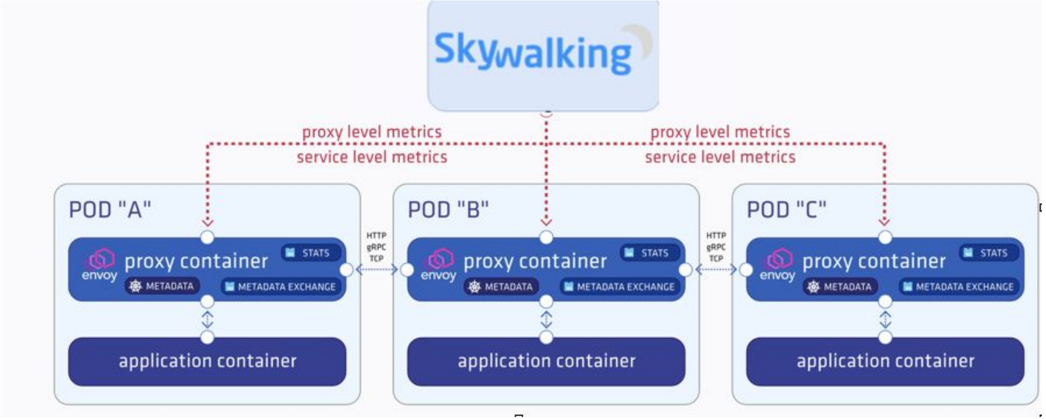
Metadata:

1. Kubernetes registry
2. Metadata Exchange



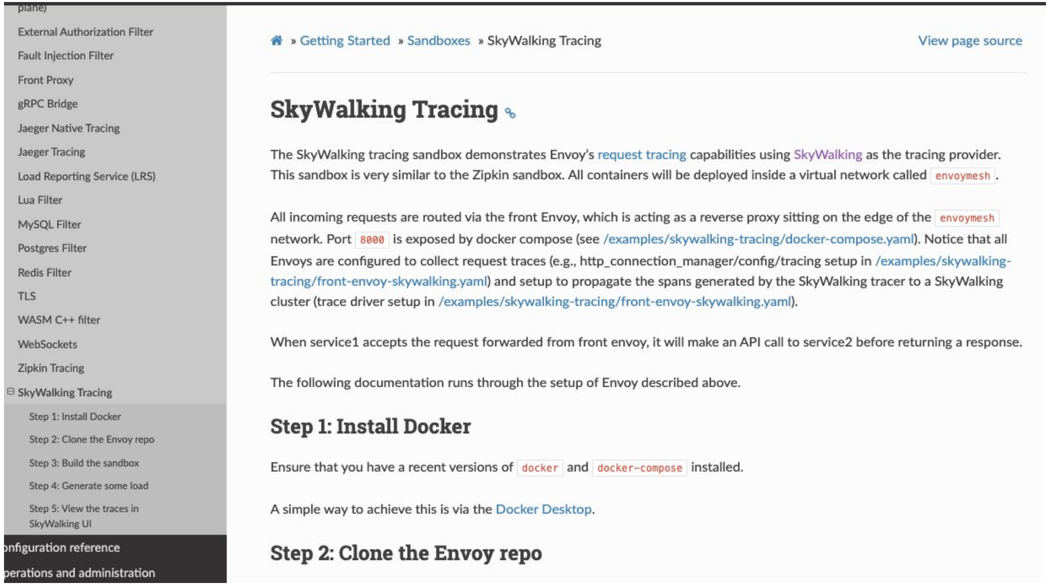
Metrics

- Data Plane
- Control Plane (Istiod)
- Topology(TODO)



Tracing

Envoy native support



plane)

External Authorization Filter

Fault Injection Filter

Front Proxy

gRPC Bridge

Jaeger Native Tracing

Jaeger Tracing

Load Reporting Service (LRS)

Lua Filter

MySQL Filter

Postgres Filter

Redis Filter

TLS

WASM C++ filter

WebSockets

Zipkin Tracing

⊟ SkyWalking Tracing

Step 1: Install Docker

Step 2: Clone the Envoy repo

Step 3: Build the sandbox

Step 4: Generate some load

Step 5: View the traces in SkyWalking UI

onfiguration reference

perations and administration

» Getting Started » Sandboxes » SkyWalking Tracing

View page source

SkyWalking Tracing

The SkyWalking tracing sandbox demonstrates Envoy's request tracing capabilities using SkyWalking as the tracing provider. This sandbox is very similar to the Zipkin sandbox. All containers will be deployed inside a virtual network called `envoymesh`.

All incoming requests are routed via the front Envoy, which is acting as a reverse proxy sitting on the edge of the `envoymesh` network. Port `8000` is exposed by docker compose (see /examples/skywalking-tracing/docker-compose.yaml). Notice that all Envoys are configured to collect request traces (e.g., http_connection_manager/config/tracing setup in /examples/skywalking-tracing/front-envoy-skywalking.yaml) and setup to propagate the spans generated by the SkyWalking tracer to a SkyWalking cluster (trace driver setup in /examples/skywalking-tracing/front-envoy-skywalking.yaml).

When service1 accepts the request forwarded from front envoy, it will make an API call to service2 before returning a response.

The following documentation runs through the setup of Envoy described above.

Step 1: Install Docker

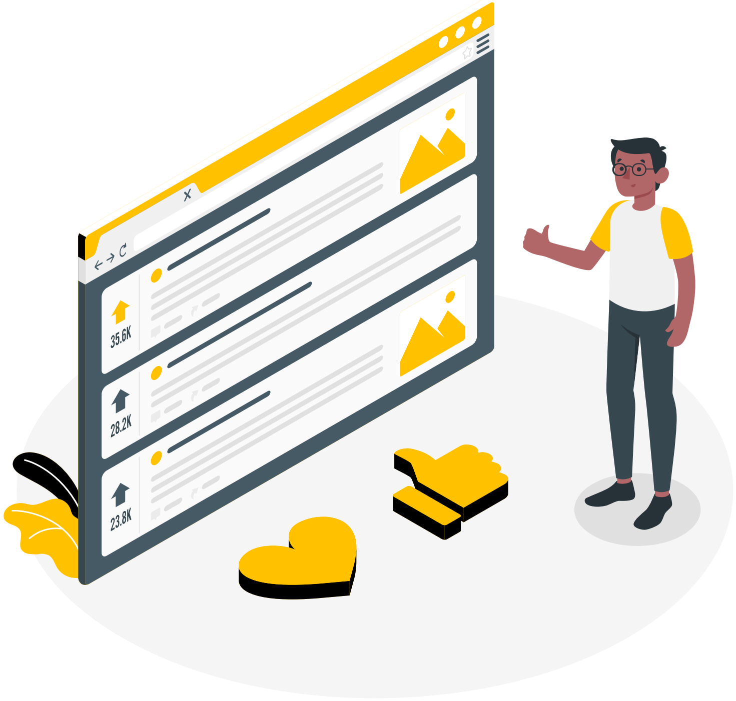
Ensure that you have a recent versions of `docker` and `docker-compose` installed.

A simple way to achieve this is via the Docker Desktop.

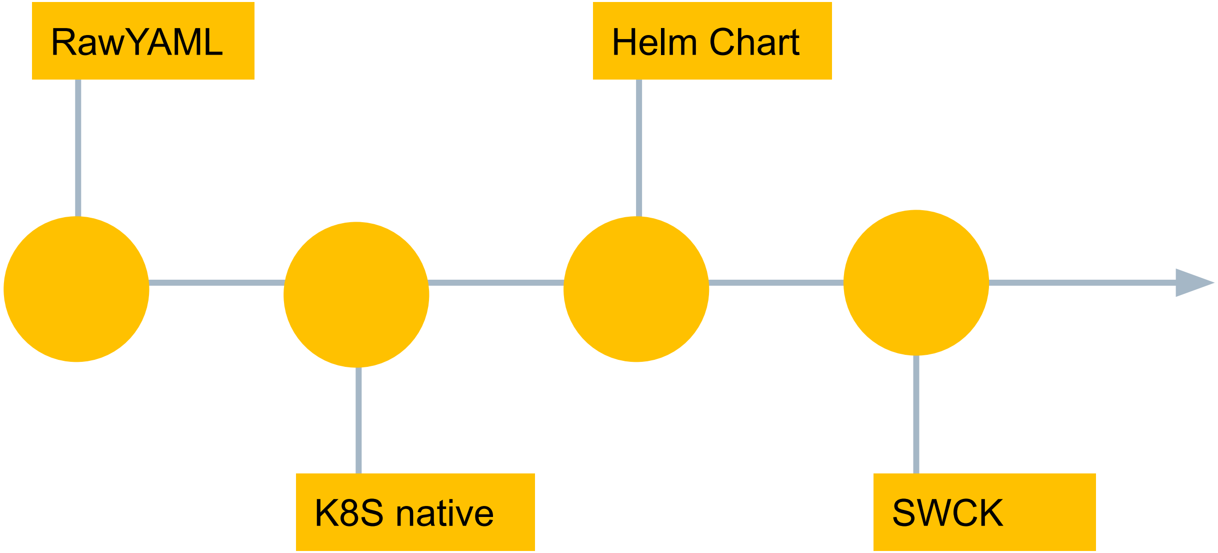
Step 2: Clone the Envoy repo

Running On K8S

03



Evolution

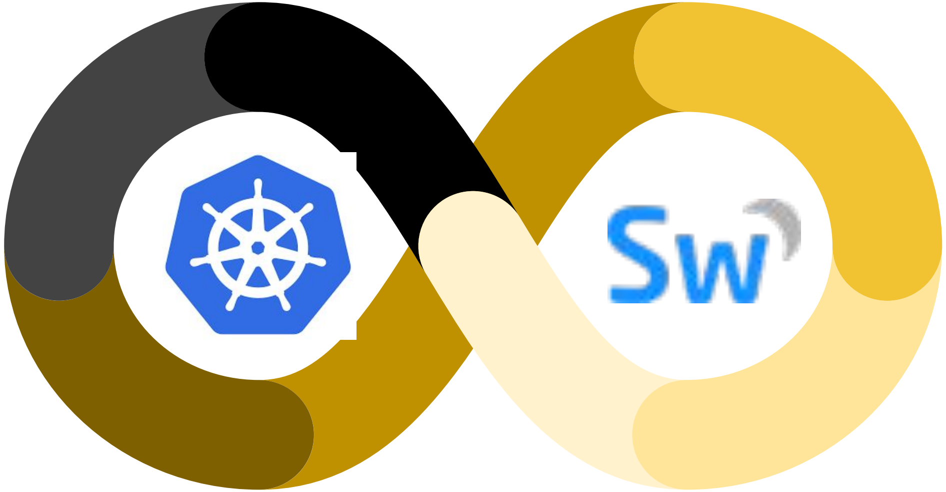


YAML by Hand

- All components
- Deployment, Service, Ingress, Statefulset and etc
- Verbose CofingMap

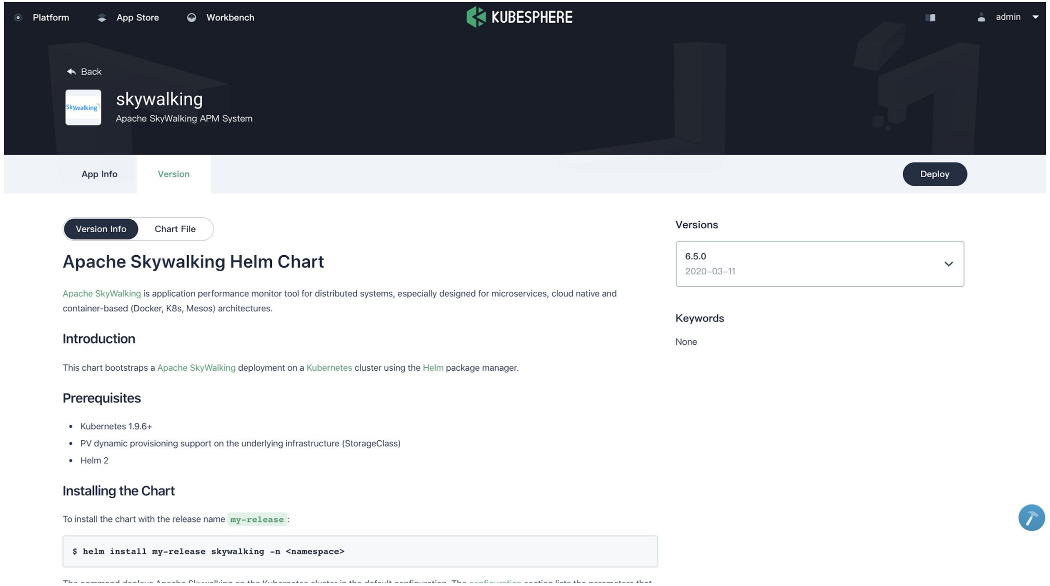
K8S Native

- Coordinator
- Env variable
- Health Check



Helm Chart

- Production Mode
- Elasticsearch Dependency
- Istio Integration



Apache Skywalking Helm Chart

Apache SkyWalking is application performance monitor tool for distributed systems, especially designed for microservices, cloud native and container-based (Docker, K8s, Mesos) architectures.

Introduction

This chart bootstraps a Apache SkyWalking deployment on a Kubernetes cluster using the Helm package manager.

Prerequisites

- Kubernetes 1.9.6+
- PV dynamic provisioning support on the underlying infrastructure (StorageClass)
- Helm 2

Installing the Chart

To install the chart with the release name `my-release`:

```
$ helm install my-release skywalking -n <namespace>
```

The command deploys Apache Skywalking on the Kubernetes cluster in the default configuration. The configuration section lists the parameters that

Versions

6.5.0
2020-03-11

Keywords

None

SkyWalking Cloud on K8S

Not just running, but living with Kubernetes

- Operator
- K8S O11Y
- HPA, VPA
- Istio
- User friendly
- More...

Future

04

Feats

SWCTL

Manifest

Operator init

CLI

UI

RocketBot

Istio Relevant
Panels
Topology based
on Metrics

More Popularity

- Artifact HUB
- Cloud provider
- More profiles
- More examples
- Meeting
- Blog & Video

Thanks

Do you have
any questions?

Wechat: putao209764
Twitter: @hanahmily
Mail: hanahmily@gmail.com