Speaker

- Iris Ding 丁少君
- IBM工程师，开源贡献者
  - Istio Maintainer
  - Apache CXF Committer
  - Github: irisdingbj

Agenda

- Install Operator API

- Custom Install via istioctl

- Custom install via Operator

- Canary Upgrade for Istio Control plane

Prepare for Installation

- Download the Istio release.

- Perform any necessary platform-specific setup.

- Check the Requirements for Pods and Services.

InstallOperator API(iop)

MeshConfig

```
message MeshConfig {
    // Address of the server that will be used by the proxies for policy
    // check calls. By using different names for mixerCheckServer and
    // mixerReportServer, it is possible to have one set of Mixer servers handle
    // policy check calls while another set of Mixer servers handle telemetry
    // calls.
    //
    // NOTE: Omitting mixerCheckServer while specifying mixerReportServer is
    // equivalent to setting disablePolicyChecks to true.
    string mixer_check_server = 1;

    // Address of the server that will be used by the proxies for policy report
    // calls.
    string mixer_report_server = 2;

    // Disable policy checks by the Mixer service. Default
    // is false, i.e. Mixer policy check is enabled by default.
    bool disable_policy_checks = 3;

    // $hide_from_docs
    // Disable telemetry reporting by the Mixer service for HTTP traffic.
    // Default is false (telemetry reporting via Mixer is enabled).
    // This option provides a transition path for Istio extensibility v2.
    bool disable_mixer_http_reports = 48;

    // Allow all traffic in cases when the Mixer policy service cannot be reached.
    // Default is false which means the traffic is denied when the client is unable
    // to connect to Mixer.
    bool policy_check_fail_open = 25;

    // Enable session affinity for Envoy Mixer reports so that calls from a proxy will
    // always target the same Mixer instance.
    bool sidecar_to_telemetry_session_affinity = 30;
```

Component configuration API

```
26  message IstioComponentSetSpec {
27      BaseComponentSpec base = 29;
28      ComponentSpec pilot = 30;
29      ComponentSpec policy = 33;
30      ComponentSpec telemetry = 34;
31      ComponentSpec cni = 38;
32      ComponentSpec istiod_remote = 39;
33
34      repeated GatewaySpec ingress_gateways = 40;
35      repeated GatewaySpec egress_gateways = 41;
36
37      reserved 31, 32, 35, 36, 37;
38      reserved "citadel", "galley", "node_agent", "proxy", "sidecar_injector";
39  }
40
41  // Configuration for base component.
42  message BaseComponentSpec {
43      // Selects whether this component is installed.
44      TypeBoolValueForPB enabled = 1;
45  }
46
47  // Configuration for internal components.
48  message ComponentSpec {
49      // Selects whether this component is installed.
50      TypeBoolValueForPB enabled = 1;
51      // Namespace for the component.
52      string namespace = 2;
53      // Hub for the component (overrides top level hub setting).
54      string hub = 10;
55      // Tag for the component (overrides top level tag setting).
56      TypeInterface tag = 11;
57
58      // Arbitrary install time configuration for the component.
59      TypeInterface spec = 30;
60
61      // Kubernetes resource spec.
62      KubernetesResourcesSpec k8s = 50;
63  }
64
```

5

InstallOperator API(iop)

- The legacy Helm installation API for backwards compatibility. operator/pkg/apis/istio/v1alpha1/values_types.proto
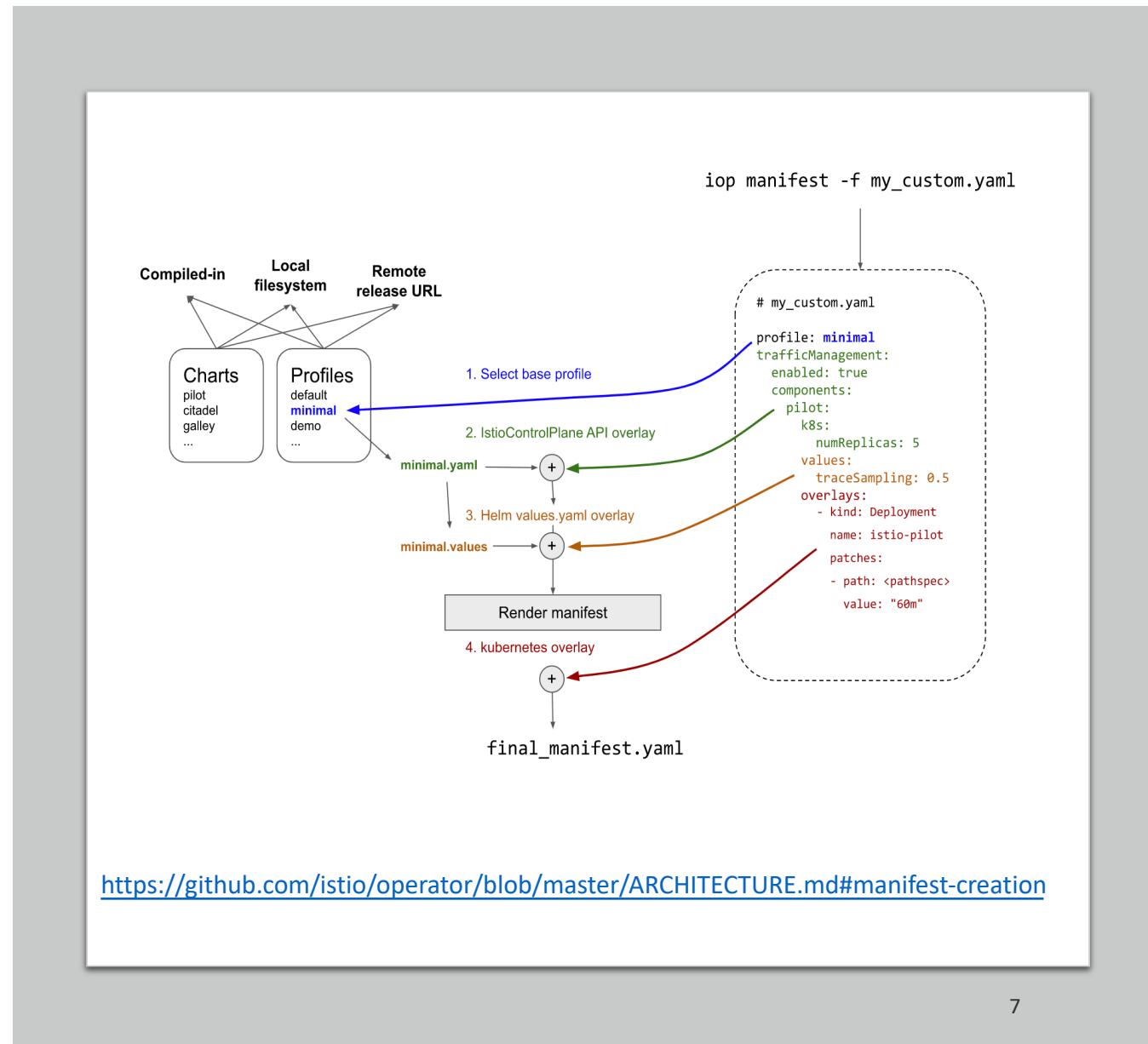
  https://github.com/istio/api/blob/master/operator/v1alpha1/operator.proto

Option 1: istioctl

- Istioctl install –f xxx.yaml
- Istioctl install --set
- Istioctl manifest generate
- Results:
  - Install Istio Control plane
  - Save Install State


```
iop manifest -f my_custom.yaml
```

Compiled-in Local filesystem Remote release URL

Charts
pilot
citadel
galley
...

Profiles
default
minimal
demo
...

1. Select base profile

2. IstioControlPlane API overlay

minimal.yaml

3. Helm values.yaml overlay

minimal.values

Render manifest

4. kubernetes overlay

final_manifest.yaml

```
# my_custom.yaml

profile: minimal
trafficManagement:
  enabled: true
  components:
    pilot:
      k8s:
        numReplicas: 5
      values:
        traceSampling: 0.5
      overlays:
      - kind: Deployment
        name: istio-pilot
        patches:
        - path: <pathspec>
          value: "60m"
```

https://github.com/istio/operator/blob/master/ARCHITECTURE.md#manifest-creation

Demo – installation customization

More support from istioctl

Istioctl profile list

1. default
2. Demo
3. minimal:
4. remote:
5. Empty
6. Preview

istioctl profile dump

|	default	demo	minimal	remote
Core components				
istio-egressgateway		X		
istio-ingressgateway	X	X		
istio-pilot	X	X	X	
Addons				
grafana		X		
istio-tracing		X		
kiali		X		
prometheus	X	X		X

More support from istioctl

istioctl profile dump
istioctl profile diff
istioctl manifest generate
istioctl manifest migrate
Istioctl manifest versions
…
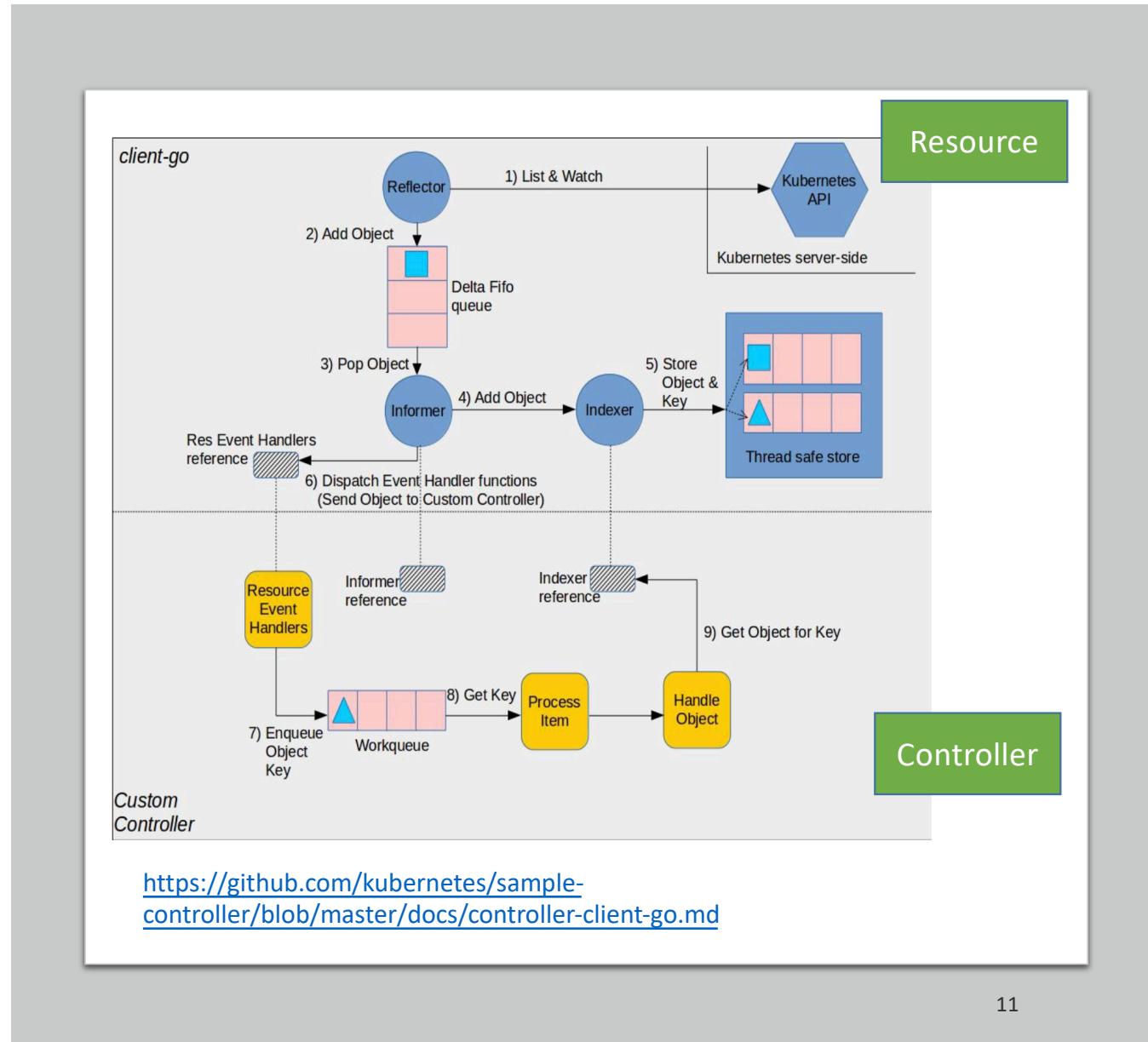
Option2: Istio Operator

- Based on K8s controller pattern

- An Operator represents human operational knowledge in software, to reliably manage an application

- Operator SDK
  - Resource spec: crd
  - Controller: actions taken when your resources changes

https://github.com/kubernetes/sample-controller/blob/master/docs/controller-client-go.md
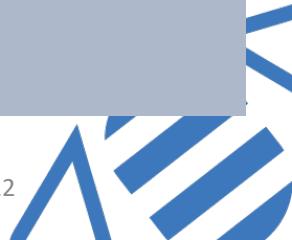
Option2: Istio Operator

```
---
apiVersion: apiextensions.k8s.io/v1beta1
kind: CustomResourceDefinition
metadata:
  name: istiooperators.install.istio.io
spec:
  group: install.istio.io
  versions:
   - name: v1alpha1
     served: true
     storage: true
  scope: Namespaced
  subresources:
   status: {}
  names:
   kind: IstioOperator
   listKind: IstioOperatorList
   plural: istiooperators
   singular: istiooperator
   shortNames:
   - iop
...
```

```
---
apiVersion: apps/v1
kind: Deployment
metadata:
  namespace: istio-operator
  name: istio-operator
spec:
  replicas: 1
  selector:
   matchLabels:
     name: istio-operator
  template:
   metadata:
     labels:
       name: istio-operator
   spec:
     serviceAccountName: istio-operator
     containers:
       - name: istio-operator
         image: gcr.io/istio-
testing/operator:1.5
         command:
         - operator
         - server
...
```

```
---
apiVersion: v1
kind: Service
metadata:
  namespace: istio-operator
  labels:
    name: istio-operator
  name: istio-operator-metrics
spec:
  ports:
  - name: http-metrics
    port: 8383
    targetPort: 8383
  selector:
    name: istio-operator
...
```

```
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  creationTimestamp: null
  name: istio-operator
rules:
# istio groups
- apiGroups:
  - authentication.istio.io
  resources:
  - '*'
  verbs:
  - '*'
- apiGroups:
  - config.istio.io
  resources:
  - '*'
  verbs:
  - '*'
- apiGroups:
  - install.istio.io
...
```

Option 2: Istio Operator

```
---
apiVersion: install.istio.io/v1alpha1
kind: IstioOperator
metadata:
  namespace: istio-system
  name: example-istiocontrolplane
spec:
  profile: demo
...
```

istioctl operator init

istioctl operator remove

Istio Upgrade

- In place upgrade
- [Canary upgrade](#)