

# How to Achieve the Perfect Union of GitOps and Observability with Argo, Istio, and SkyWalking

Jimmy Song (宋净超), Tetrade



#IstioCon

# Components

GitOps

Service Mesh

Observability

bound them together



- Argo CD
- Argo Rollouts
  - Canary Deployment



- Traffic management
  - Traffic Splitting



- Sink Metrics



# GitOps can not live without Kubernetes



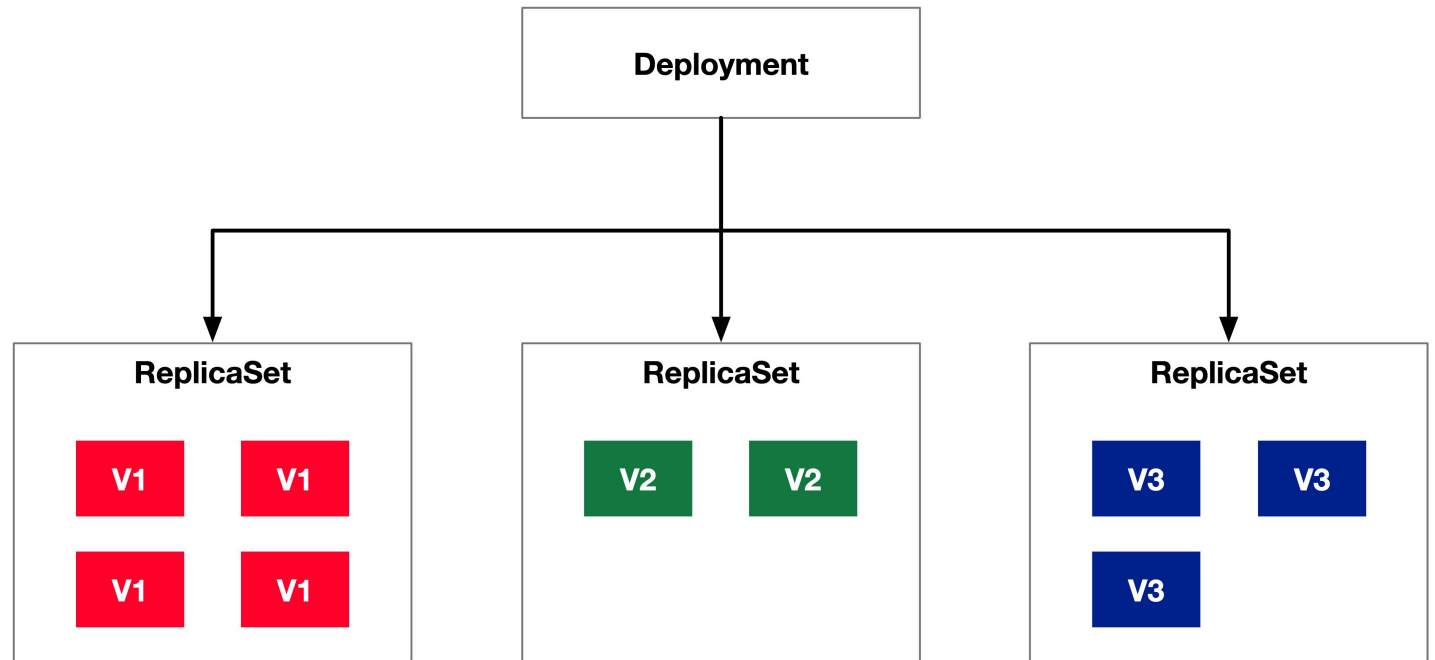
# kubernetes



# How to Rollout in Kubernetes

## Deployment

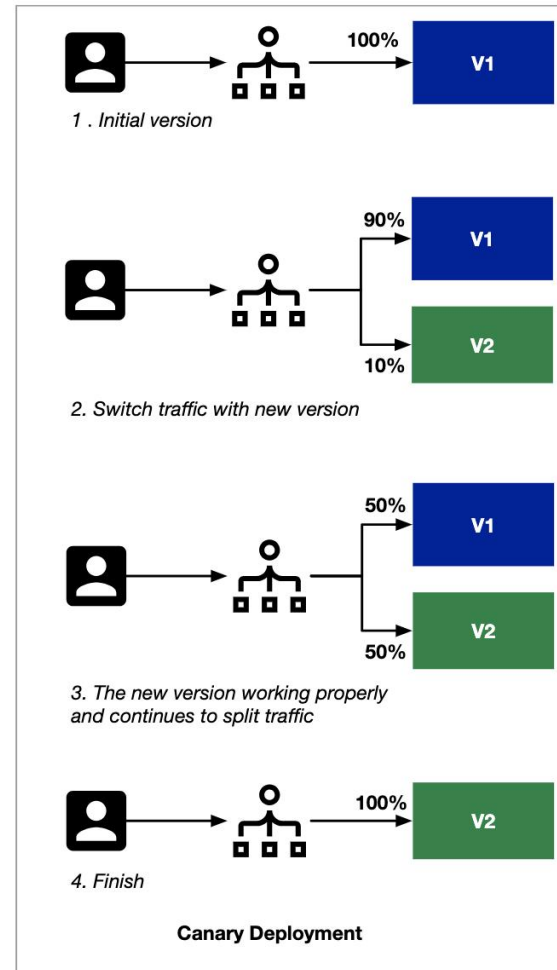
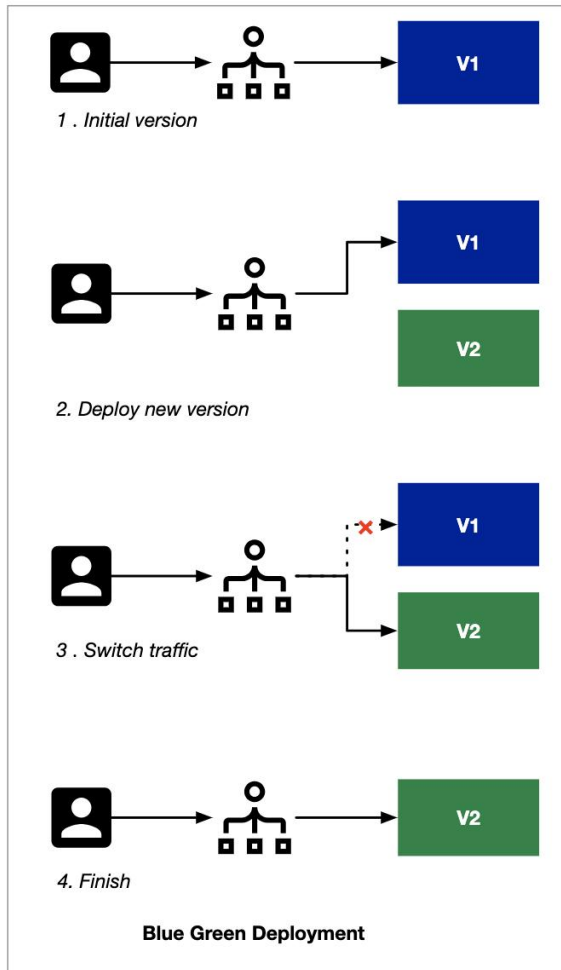
- Recreate
- RollingUpdate



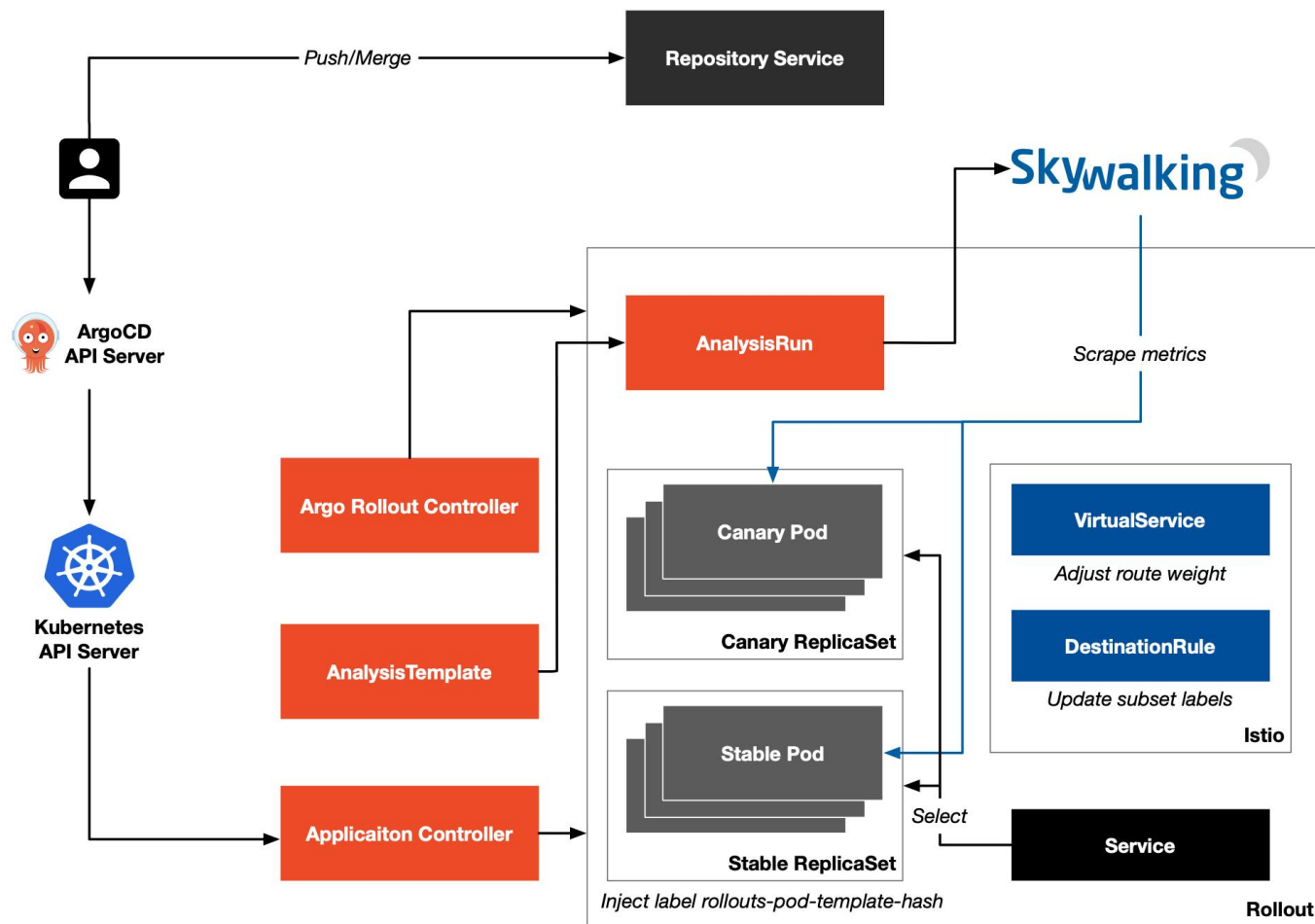
# GitOps with ArgoCD



# Argo Rollouts Zero Downtime Deployment



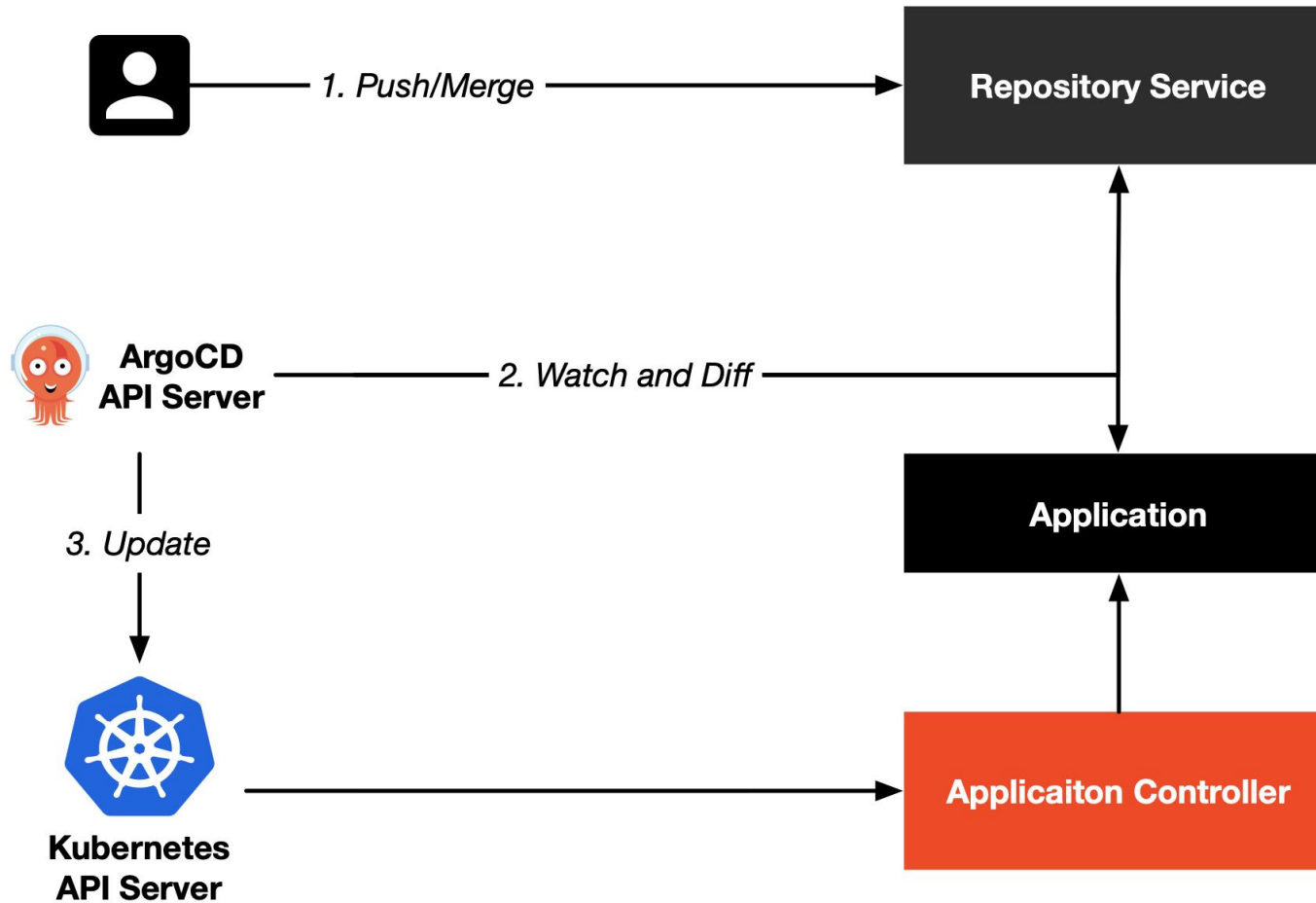
# Architecture



- Source Code: GitHub
- CD: Argo CD
- Canary Release: Argo Rollouts
- Traffic Splitting: Istio VS&DR
- Metrics Scraping: AnalysisTemplate
- APM: SkyWalking

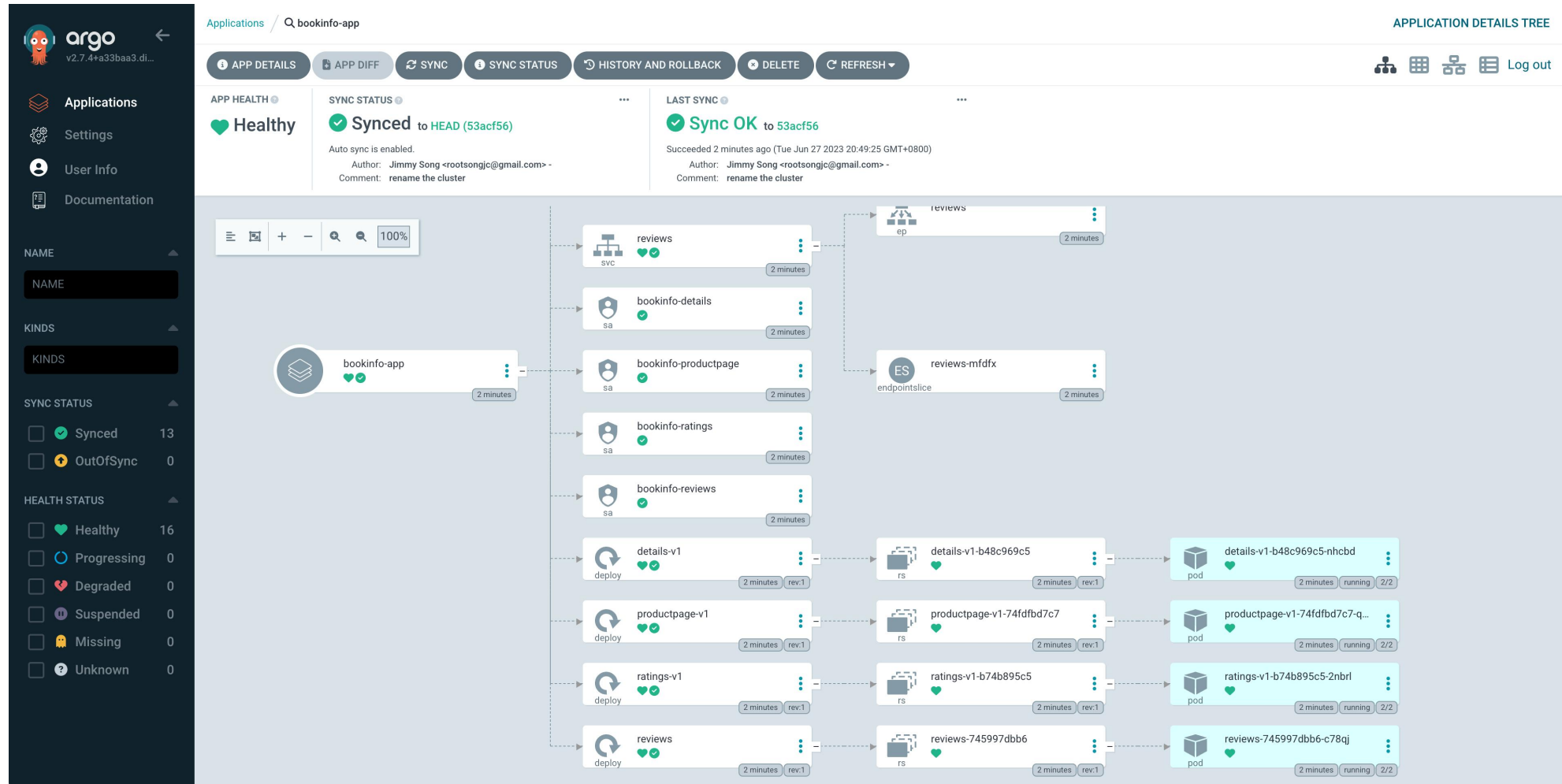


# Continuous Delivery





# ArgoCD Dashboard



The screenshot displays the ArgoCD dashboard for the 'bookinfo-app'. The interface is divided into a left sidebar, a top navigation bar, and a main content area.

**Left Sidebar:**

- Logo: argo v2.7.4+a33baa3.di...
- Menu items: Applications, Settings, User Info, Documentation.
- Filters: NAME, KINDS, SYNC STATUS, HEALTH STATUS.
- SYNC STATUS: Synced (13), OutOfSync (0).
- HEALTH STATUS: Healthy (16), Progressing (0), Degraded (0), Suspended (0), Missing (0), Unknown (0).

**Top Navigation Bar:**

- Applications / bookinfo-app
- Buttons: APP DETAILS, APP DIFF, SYNC, SYNC STATUS, HISTORY AND ROLLBACK, DELETE, REFRESH.
- APPLICATION DETAILS TREE (Log out)

**Main Content Area:**

- APP HEALTH:** Healthy
- SYNC STATUS:** Synced to HEAD (53acf56). Auto sync is enabled. Author: Jimmy Song <rootsongjc@gmail.com>. Comment: rename the cluster.
- LAST SYNC:** Sync OK to 53acf56. Succeeded 2 minutes ago (Tue Jun 27 2023 20:49:25 GMT+0800). Author: Jimmy Song <rootsongjc@gmail.com>. Comment: rename the cluster.
- Application Graph:** A hierarchical tree view showing the deployment of the 'bookinfo-app'. The root node is 'bookinfo-app' (2 minutes). It branches into several services: 'reviews' (2 minutes), 'bookinfo-details' (2 minutes), 'bookinfo-productpage' (2 minutes), 'bookinfo-ratings' (2 minutes), 'bookinfo-reviews' (2 minutes), 'details-v1' (2 minutes, rev:1), 'productpage-v1' (2 minutes, rev:1), 'ratings-v1' (2 minutes, rev:1), and 'reviews' (2 minutes, rev:1). Each service node further branches into its respective Kubernetes resources (e.g., 'details-v1' branches into 'details-v1-b48c969c5' and 'details-v1-b48c969c5-nhcbd'). The final nodes are pods, many of which are in a 'running' state (e.g., 'details-v1-b48c969c5-nhcbd' is running 2/2).

#IstioCon



# ArgoCD Rollout



# Argo Rollouts



# Migrating Deployment to Rollout

- Argo rollout

```
argocd app create reviews-rollout \  
--repo https://github.com/tetradeio/tse-gitops-demo.git \  
--path argo/rollout --dest-server https://kubernetes.default.svc \  
--dest-namespace bookinfo --sync-policy automated
```

- Sent traffics to reviews service, rollout will proceed to the next step when the success rate meets the AnalysisTemplate.

```
export GATEWAY_HOSTNAME=$(kubectl -n bookinfo get service \  
tsb-gateway-bookinfo -o jsonpath='{.status.loadBalancer.ingress[0].hostname}')  
while 1;  
do  
    curl -H "Host: bookinfo.tetrade.com" \  
    http://\$GATEWAY\_HOSTNAME/api/v1/products/1/reviews;  
    sleep 3;  
done
```



# ArgoCD Rollouts Dashboard

The dashboard displays the configuration and status of the 'reviews-rollout' in the 'bookinfo' namespace. At the top, the Argo Rollouts version is v1.2.0+eb26809. The rollout is currently in a paused state, indicated by a red pause icon.

**Actions:** Restart, Retry, Abort, Promote, PromoteFull

**Steps:**

- Set Weight: 10%
- Pause: 10m
- Set Weight: 20%
- Pause: 5m** (Current step)
- Set Weight: 40%
- Pause: 5m
- Set Weight: 60%
- Pause: 5m
- Set Weight: 80%
- Pause: 5m

**Summary:**

- Strategy: Canary
- Step: 3/10
- Set Weight: 20
- Actual Weight: 20

**Containers:**

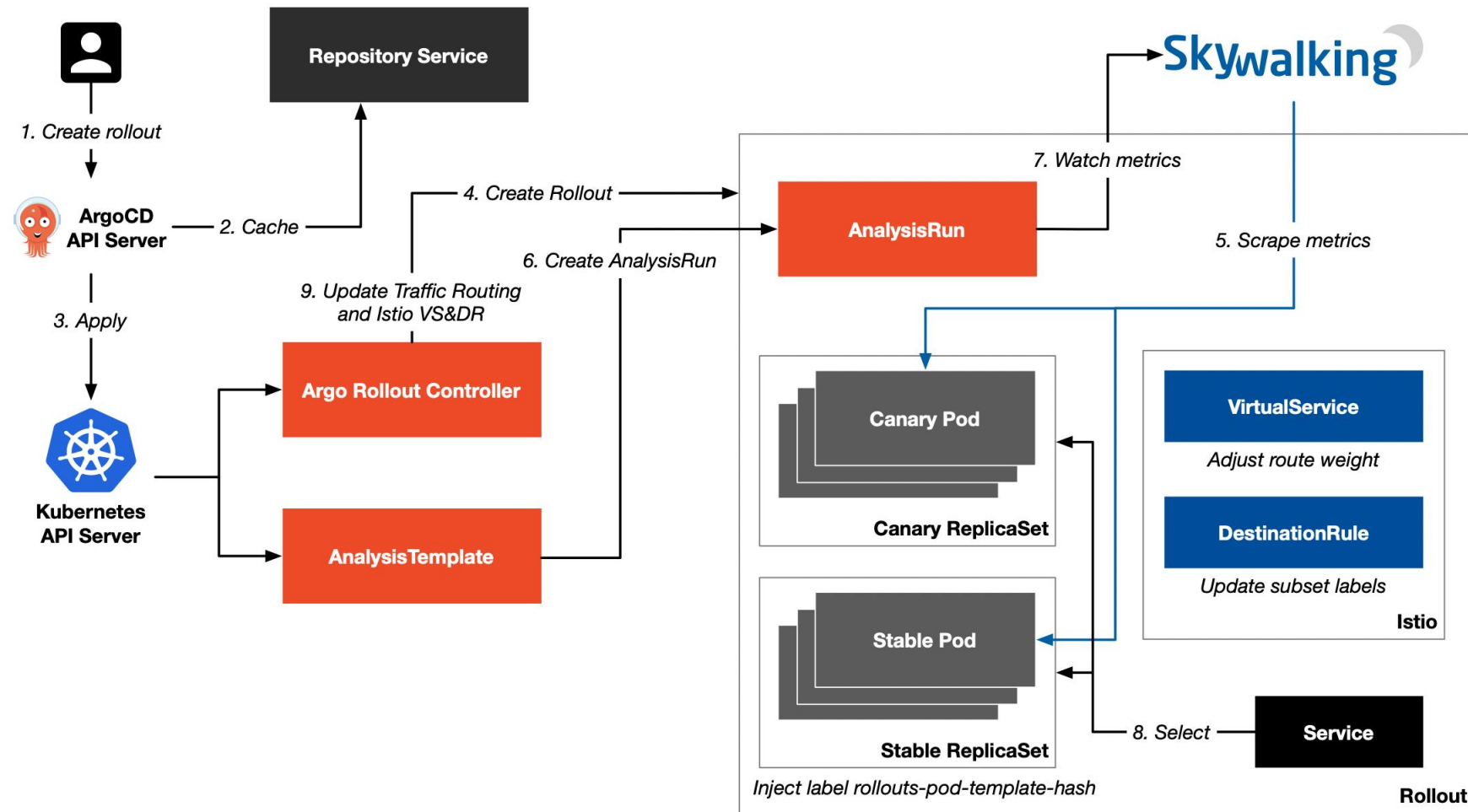
- reviews: docker.io/istio/examples-bookinfo-reviews-v2:1.16.4

**Revisions:**

- Revision 2:** docker.io/istio/examples-bookinfo-reviews-v2:1.16.4. Status: canary. Analysis Run: reviews-rollout-867b9c9bcb-2 (Success).
- Revision 1:** docker.io/istio/examples-bookinfo-reviews-v1:1.16.4. Status: stable. Analysis Run: reviews-rollout-5d9dc876c9 (Success).



# Argo Rollout





```

1  apiVersion: argoproj.io/v1alpha1
2  kind: Rollout
3  metadata:
4    name: reviews-rollout
5  spec:
6    replicas: 5
7    selector:
8      matchLabels:
9        app: reviews
10   workloadRef:
11     apiVersion: apps/v1
12     kind: Deployment
13     name: reviews
14   strategy:
15     canary:
16       analysis:
17         templates:
18           - templateName: apdex
19             startingStep: 2
20         args:
21           - name: service-name
22             value: canary|reviews|bookinfo|cluster-1|-
23       canaryMetadata:
24         annotations:
25           version: canary
26         labels:
27           version: canary
28           service.istio.io/canonical-revision: canary
29       stableMetadata:
30         annotations:
31           version: stable
32         labels:
33           version: stable
34           service.istio.io/canonical-revision: stable
35       trafficRouting:
36         istio:
37           virtualService:
38             name: reviews
39           destinationRule:
40             name: reviews
41             canarySubsetName: canary
42             stableSubsetName: stable
43       steps:
44         - setWeight: 10
45         - pause: {duration: 10m}
46         - setWeight: 20
47         - pause: {duration: 5m}
48         - setWeight: 40
49         - pause: {duration: 5m}
50         - setWeight: 60
51         - pause: {duration: 5m}
52         - setWeight: 80
53         - pause: {duration: 5m}

```

metric

Istio config

Rollout steps

```

- apiVersion: networking.istio.io/v1alpha3
  kind: DestinationRule
  metadata:
    name: reviews
    labels:
      istio.io/rev: tsb
    annotations:
      tsb.tetrate.io/organization: tse
      tsb.tetrate.io/tenant: tse
      tsb.tetrate.io/workspace: bookinfo-ws
      tsb.tetrate.io/trafficGroup: bookinfo-traffic
  spec:
    host: reviews
    subsets:
      - name: stable
        labels:
          app: reviews
          version: stable
      - name: canary
        labels:
          app: reviews
          version: canary
- apiVersion: networking.istio.io/v1alpha3
  kind: VirtualService
  metadata:
    name: reviews
    labels:
      istio.io/rev: tsb
    annotations:
      tsb.tetrate.io/organization: tse
      tsb.tetrate.io/tenant: tse
      tsb.tetrate.io/workspace: bookinfo-ws
      tsb.tetrate.io/trafficGroup: bookinfo-traffic
  spec:
    hosts:
      - reviews
    http:
      - route:
          - destination:
              host: reviews
              subset: stable
              weight: 100
          - destination:
              host: reviews
              subset: canary
              weight: 0

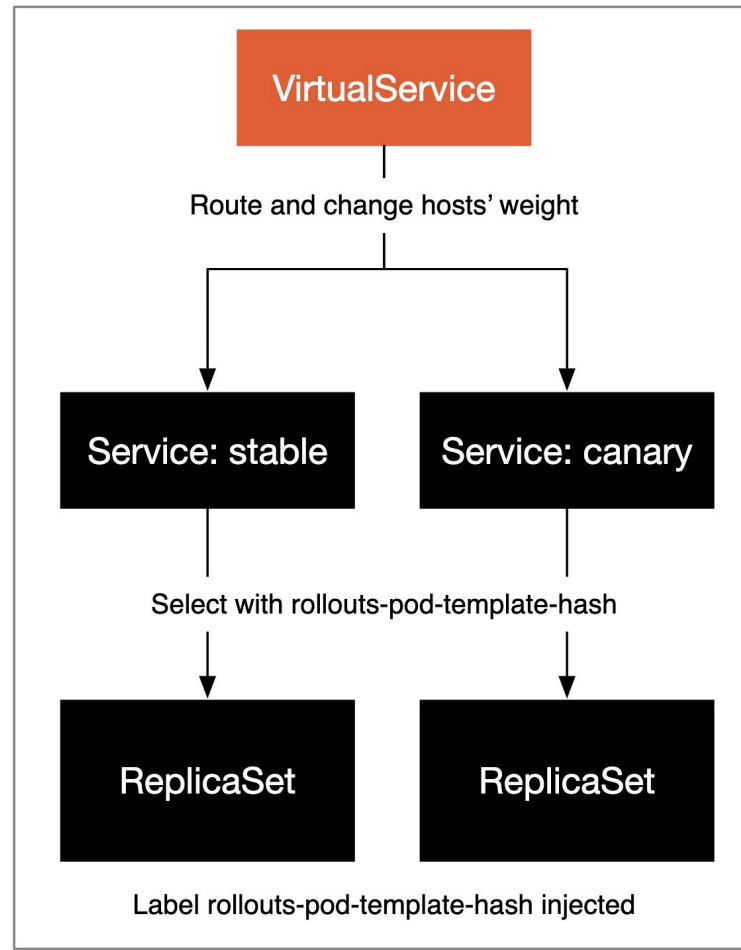
```

Update labels

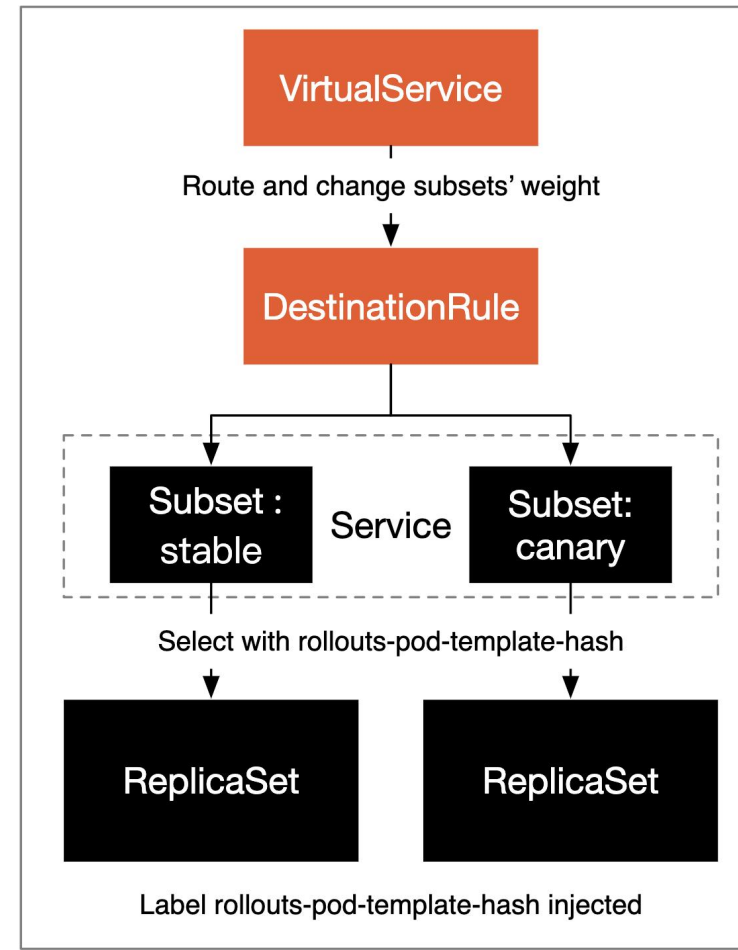
Update weight



# Argo Rollouts uses Istio for traffic splitting



**Host based traffic routing**



**Subset based traffic routing**



# AnaysisTemplate & SkyWalking

```
apiVersion: argoproj.io/v1alpha1
kind: AnalysisTemplate
metadata:
  name: apdex
spec:
  args:
    - name: service-name
  metrics:
    - name: apdex
      interval: 5m
      successCondition: "all(result.service_apdex.values.values, {asFloat(.value) >= 9900})"
      failureLimit: 3
      provider:
        skywalking:
          interval: 3m
          address: http://oap.istio-system:12800
          query: |
            query queryData($duration: Duration!) {
              service_apdex: readMetricsValues(
                condition: { name: "service_apdex", entity: { scope: Service, serviceName: "{{ args.service-name }}", normal: true } },
                duration: $duration) {
                label values { values { value } }
              }
            }
```





# GraphQL Querying to SkyWalking with Postman

The screenshot displays the Postman interface for a GraphQL query. The URL bar shows the endpoint: `https://a800f0b9d386e494992685b682a7b649-170393214.us-east-2.elb.amazonaws.com/graphql`. The query tab is active, showing the following query:

```
1 query ReadMetricsValues {
2   readMetricsValues(condition: {
3     name: "service_apdex", entity: {scope: Service, serviceName: "canary|reviews|bookinfo|cluster-1"},
4     normal: true}
5   }, duration: {
6     start: "2023-07-13 0812",
7     end: "2023-07-13 0813",
8     step: MINUTE
9   }) {
10     label
11     values {
12       id
13       value
14     }
15   }
16 }
17
```

The response tab shows the JSON body:

```
1 {
2   "data": {
3     "readMetricsValues": {
4       "label": null,
5       "values": {
6         "values": [
7           {
8             "id": "service_apdex_202307130812_Y2FuYXJ5fHJldmllZ3N8Ym9va2luZm98Y2x1c3Rlc0xfC0=.1",
9             "value": 10000
10          },
11          {
12             "id": "service_apdex_202307130812_Y2FuYXJ5fHJldmllZ3N8Ym9va2luZm98Y2x1c3Rlc0xfC0=.1",
13             "value": 10000
14          }
15        ]
16      }
17    }
18  }
19 }
```

The status bar at the bottom indicates: Status: 200 OK Time: 637.9 ms Size: 421 B.



# Summary

We can implement GitOps with ArgoCD and fully automated canary deployment with Argo Rollouts, Istio and SkyWalking.

You can find the demo code here:

<https://github.com/tetratelio/tse-gitops-demo>



# Thank you!

IstioCon China 2023

#IstioCon

