



TRACK 02 - MODERN APPS - CONTAINERS

Progressive Delivery using **Argo Rollouts**



Ansil H

Lead SRE - Armorblox





What is Progressive Delivery

- Progressive delivery is the **process of releasing updates of a product in a controlled and gradual manner**, thereby reducing the risk of the release, typically coupling automation and metric analysis to drive the automated promotion or rollback of the update.
- Progressive delivery is often described as **an evolution of continuous delivery**, extending the speed benefits made in CI/CD to the deployment process.





Deployment Strategies

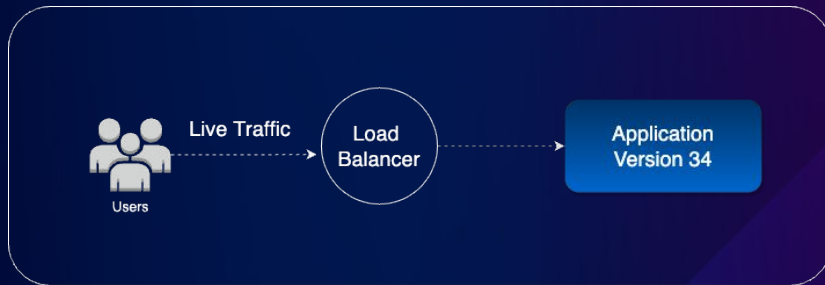
- A **Rolling Update** slowly replaces an old version with a new version.
- A **Recreate deployment** deletes the old version of the application before bring up the new version.
- A **Blue-Green deployment** has both the new and old version of the application deployed at the same time. During this time, only the old version of the application will receive production traffic.
- A **Canary deployment** exposes a subset of users to the new version of the application while serving the rest of the traffic to the old version.



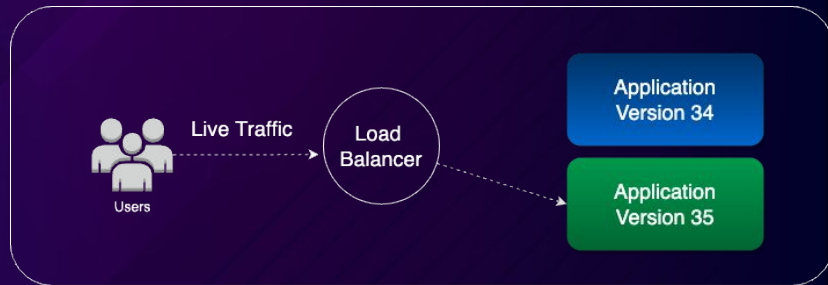


Deployment Strategies - Blue-Green

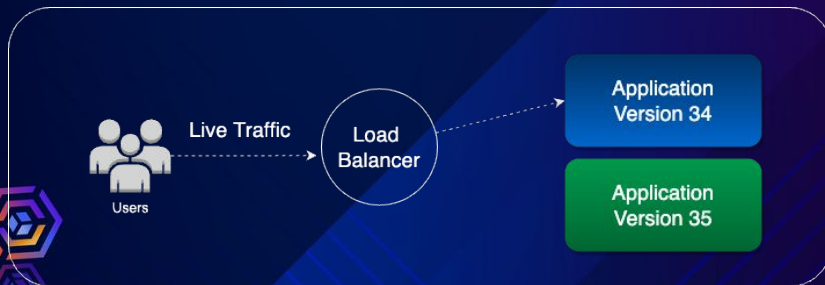
1. Initial Version



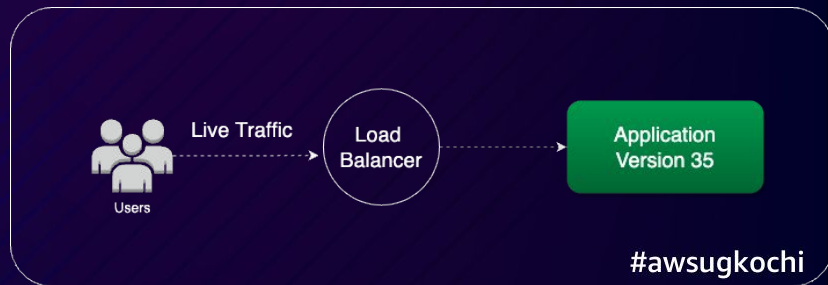
3. Switch Traffic



2. New Version Deployed



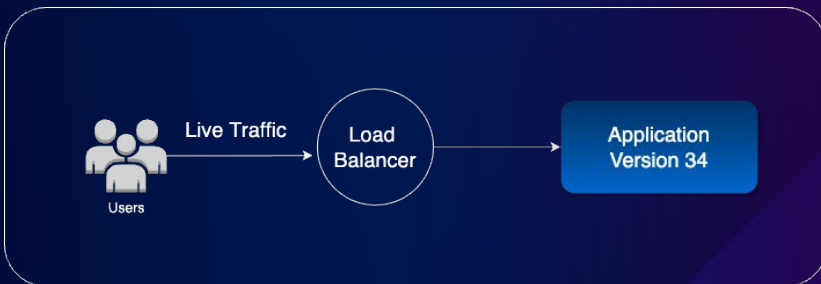
4. Finish



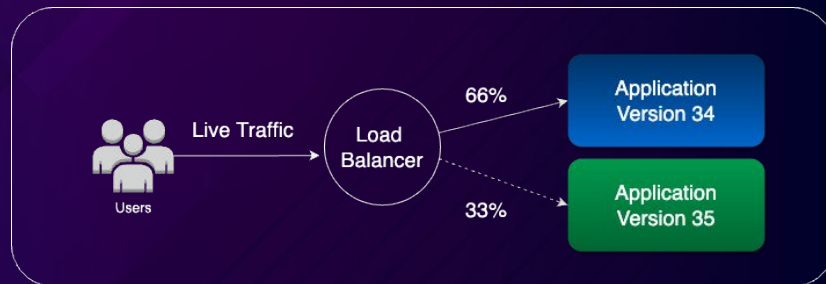


Deployment Strategies - Canary

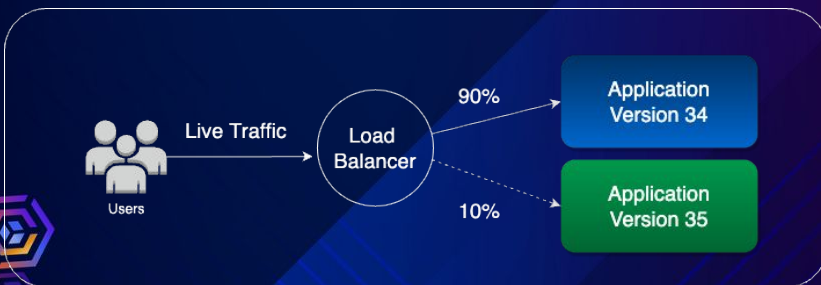
1. Initial Version



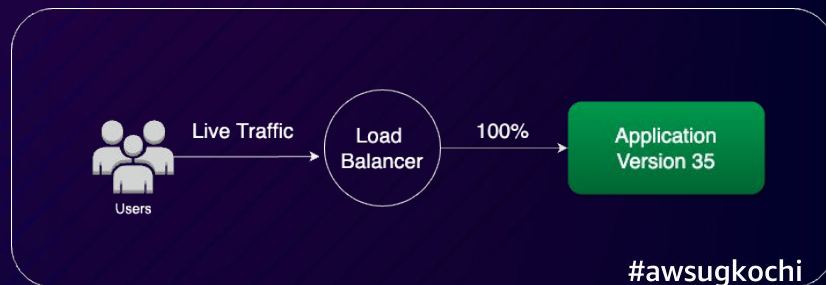
3. New Version used by 33% of users



2. New Version used by 10% of users



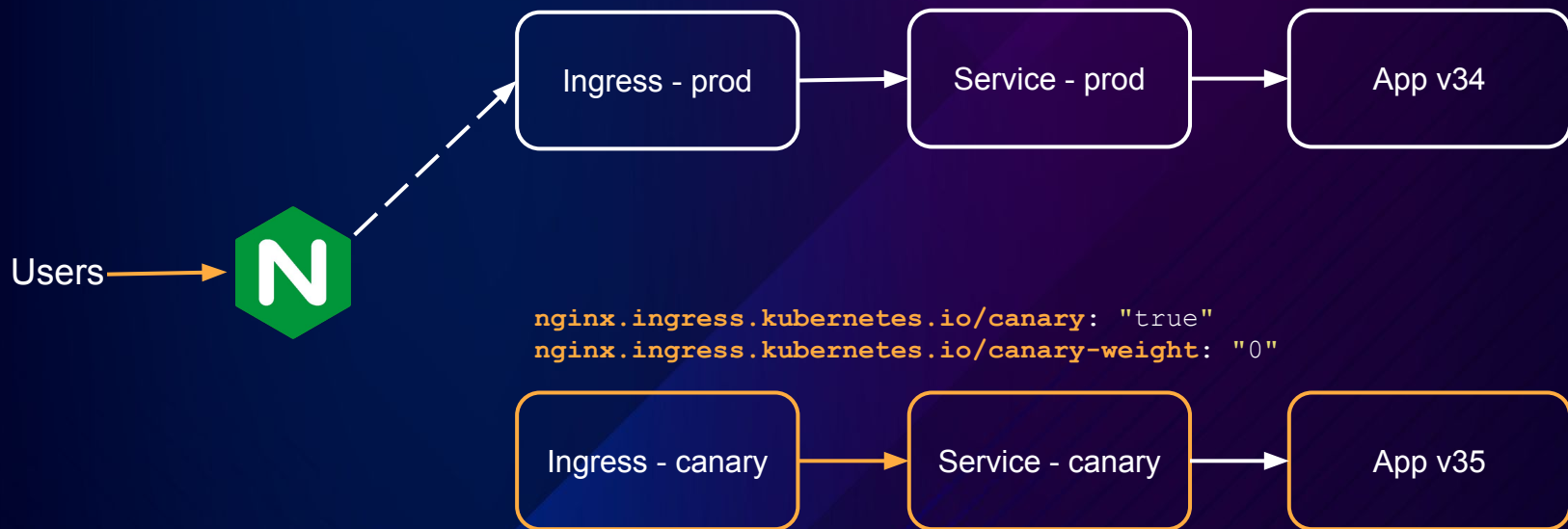
4. New Version used by 100% of users





Ingress Nginx - Canary

```
spec:  
  rules:  
  - host: app.io
```



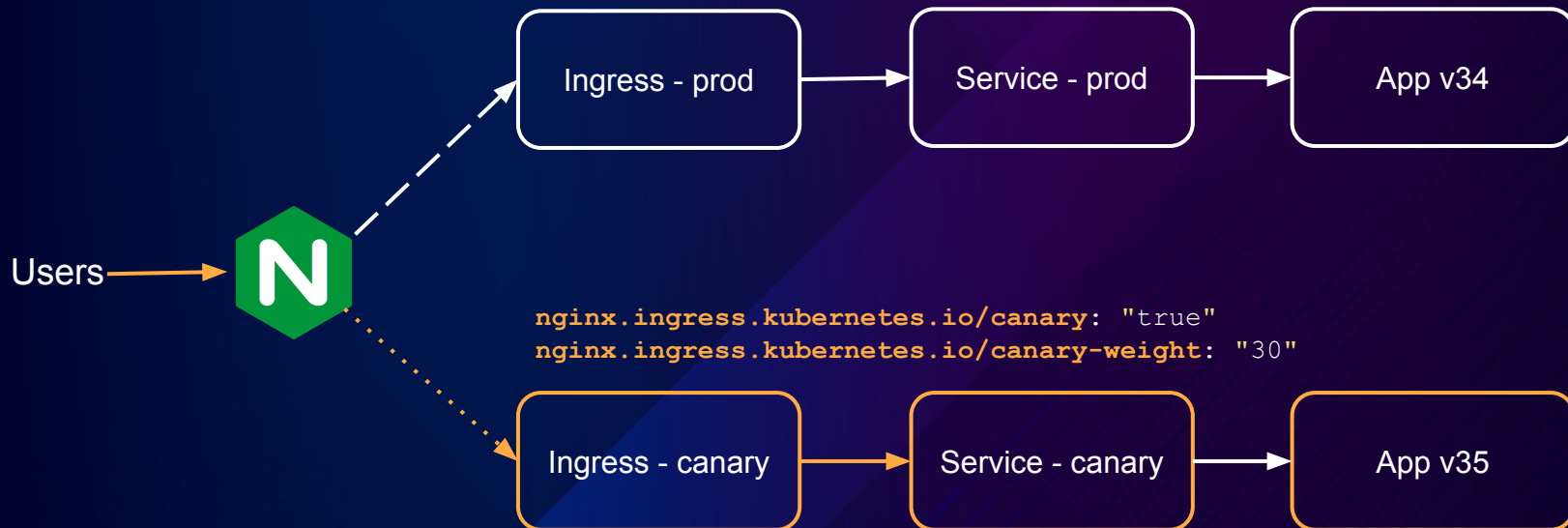
```
nginx.ingress.kubernetes.io/canary: "true"  
nginx.ingress.kubernetes.io/canary-weight: "0"
```

```
spec:  
  rules:  
  - host: app.io
```



Ingress Nginx - Canary

```
spec:
  rules:
  - host: app.io
```



```
spec:
  rules:
  - host: app.io
```



Argo Rollouts - Kubernetes Progressive Delivery Controller

- Argo Rollouts is a **Kubernetes controller and set of CRDs** which provide advanced deployment capabilities.
- Argo Rollouts **can integrates with ingress controllers and service meshes**, leveraging their traffic shaping abilities to gradually shift traffic to the new version during an update.

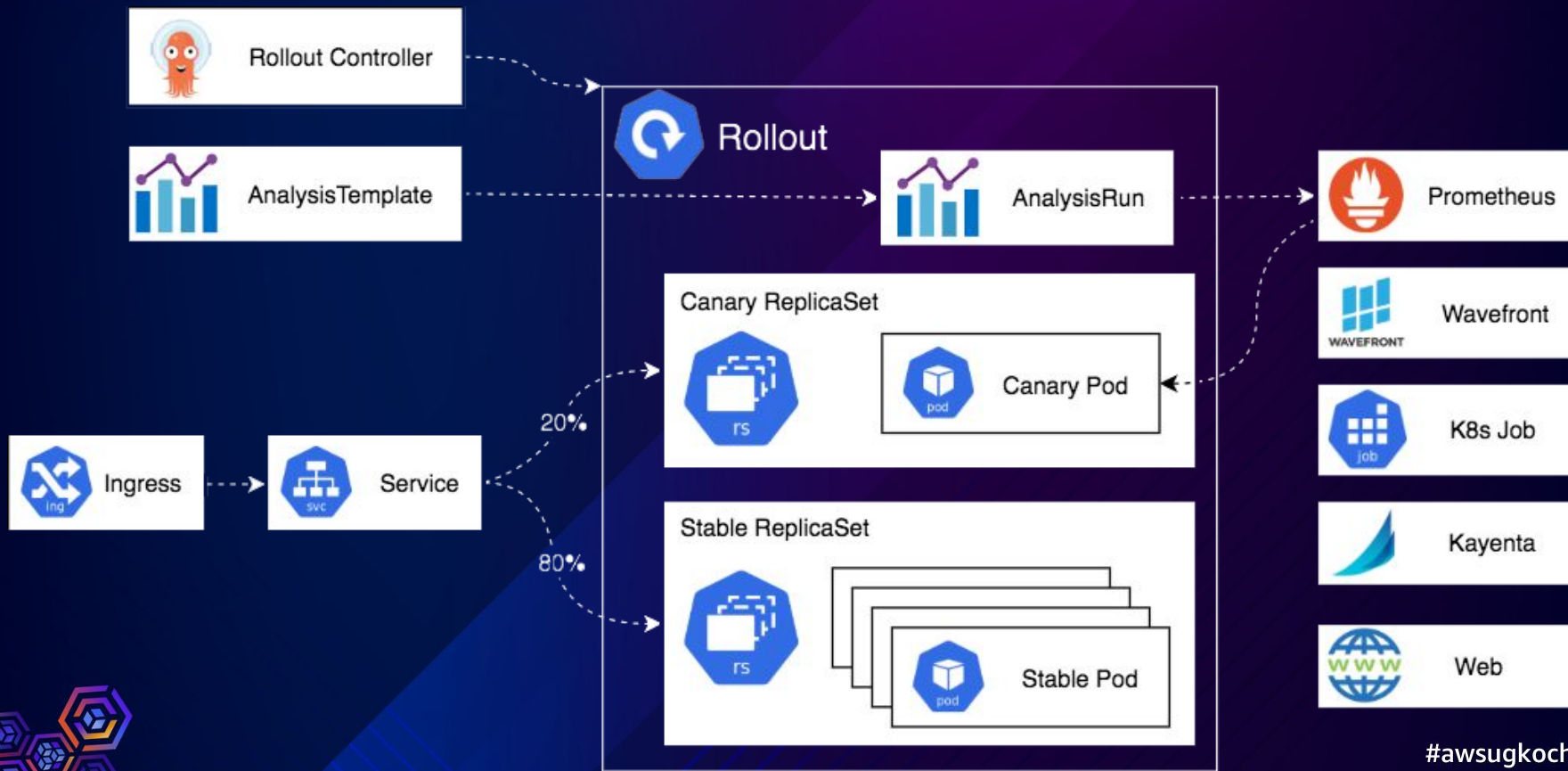


Deployment vs Argo Rollouts

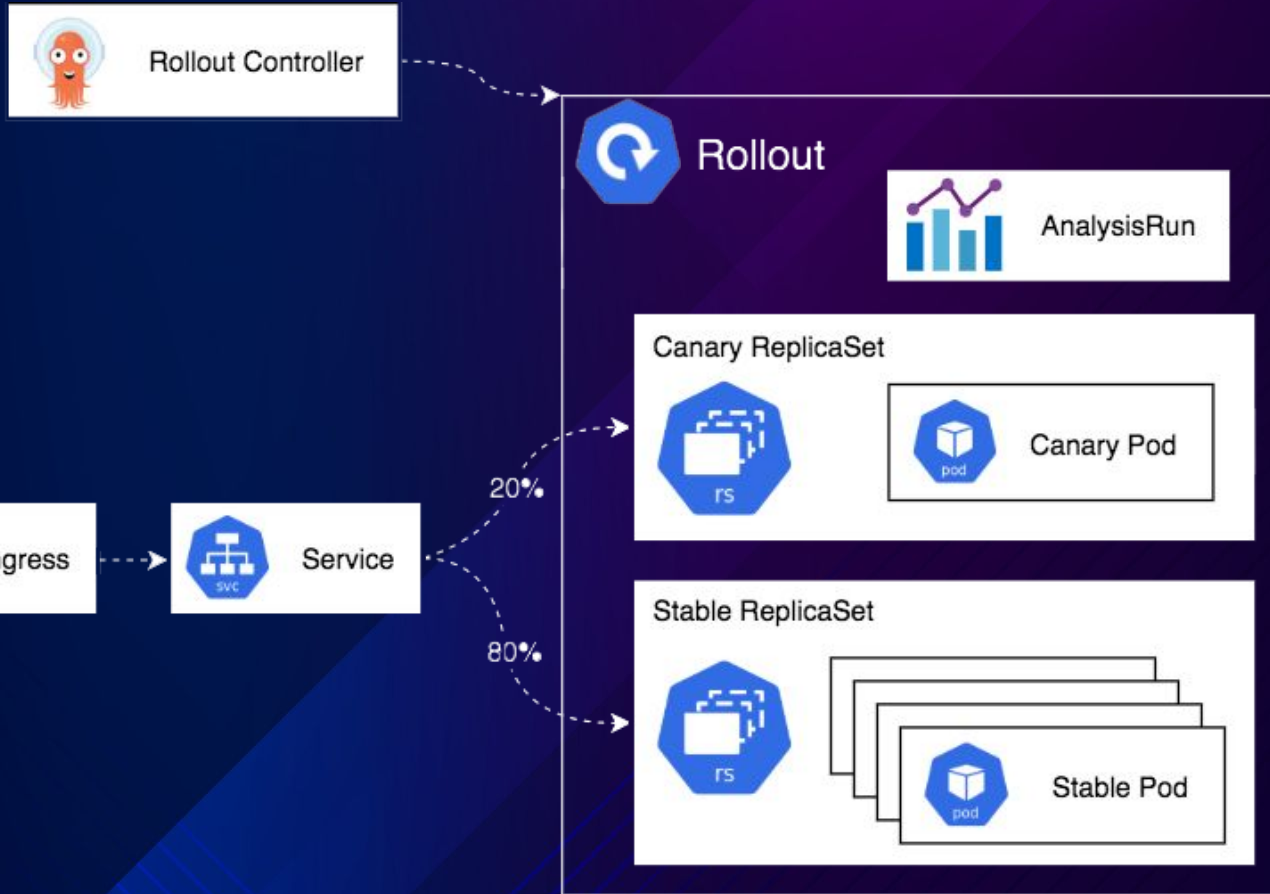
```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: rollouts-demo
spec:
  replicas: 5
  revisionHistoryLimit: 2
  selector:
    matchLabels:
      app: rollouts-demo
  template:
    metadata:
      labels:
        app: rollouts-demo
    spec:
      containers:
      - name: rollouts-demo
        image: registry.ansil.io/argoproj/rollouts-demo:blue
        ports:
        - name: http
          containerPort: 8080
          protocol: TCP
        resources:
          requests:
            memory: 32Mi
            cpu: 5m
      strategy:
        type: Recreate
```

```
apiVersion: argoproj.io/v1alpha1
kind: Rollout
metadata:
  name: rollouts-demo
spec:
  replicas: 5
  revisionHistoryLimit: 2
  selector:
    matchLabels:
      app: rollouts-demo
  template:
    metadata:
      labels:
        app: rollouts-demo
    spec:
      containers:
      - name: rollouts-demo
        image: registry.ansil.io/argoproj/rollouts-demo:blue
        ports:
        - name: http
          containerPort: 8080
          protocol: TCP
        resources:
          requests:
            memory: 32Mi
            cpu: 5m
      strategy:
        canary:
          steps:
          - setWeight: 20
          - pause: {}
          - setWeight: 40
          - pause: {duration: 10s}
          - setWeight: 60
          - pause: {duration: 10s}
          - setWeight: 80
          - pause: {duration: 10s}
```

Argo Rollouts - Architecture



Argo Rollouts - Demo Scope





Argo Rollouts - Demo

1. Argo Rollouts - Basic Operations
2. Argo Rollouts - Example - Canary Deployment
3. Argo Rollouts - Managing Rollouts using CLI and UI





Argo Rollouts - Demo





Argo Rollouts - Demo

Play Demo Video here





References - Additional Details

1. <https://argoproj.github.io/argo-rollouts>
2. <https://learning.codefresh.io>





Thank You

Request to share feedback and join AWS User Groups

