

ArgoCon

NORTH AMERICA



ArgoCon
NORTH AMERICA

Progressive Delivery with Cilium Gateway API and Argo Rollouts

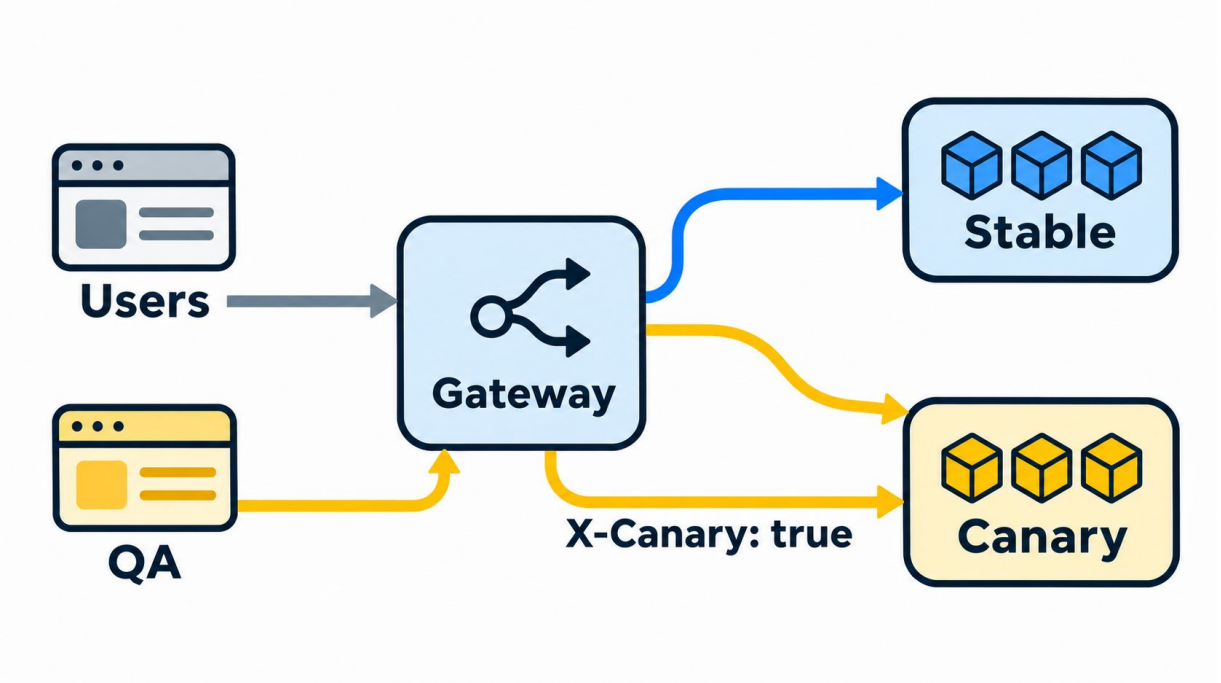
Christian Hernandez

Argo Rollouts With Gateway API

ArgoCon NORTH AMERICA

KnativeCon NORTH AMERICA

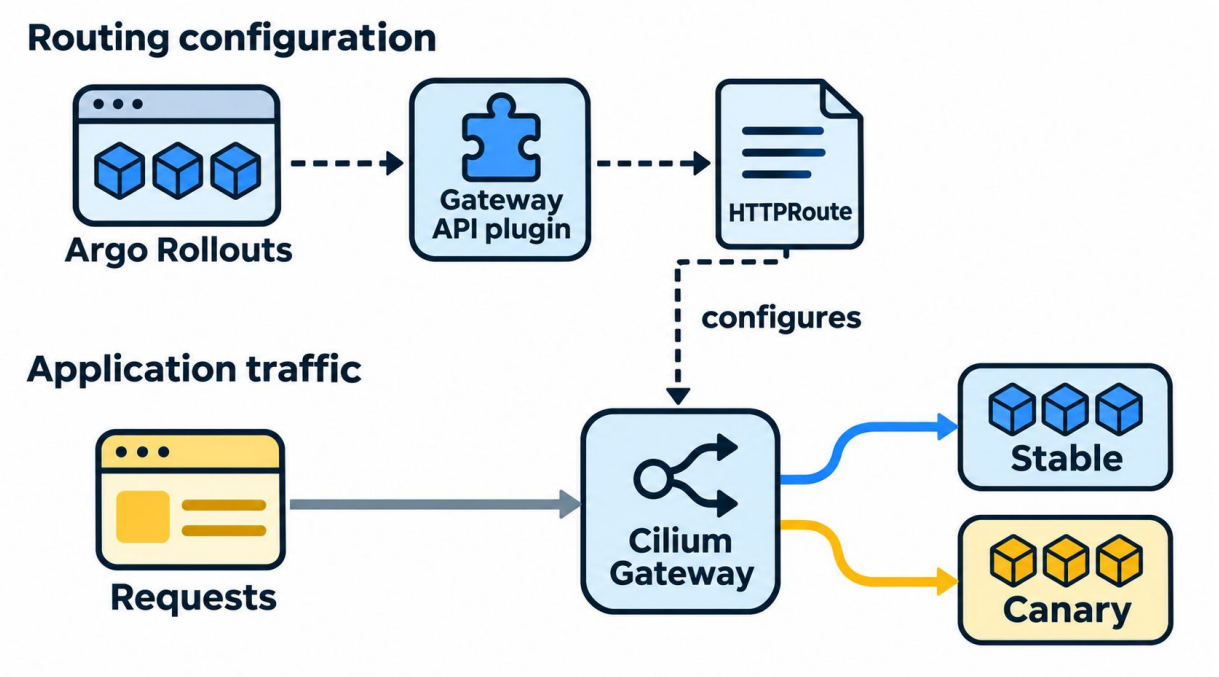
- Let QA consistently reach the new version
- Keep percentage-based traffic shifting alongside header routing
- Use the Gateway API plugin with Cilium



Give Rollouts a Common Traffic API

ArgoCon NORTH AMERICA
KnativeCon NORTH AMERICA

- The plugin updates `HTTPRoute` resources
- Your Gateway API implementation handles the traffic
- The same API can support different networking implementations
- Header-based canaries do not require adding a service mesh



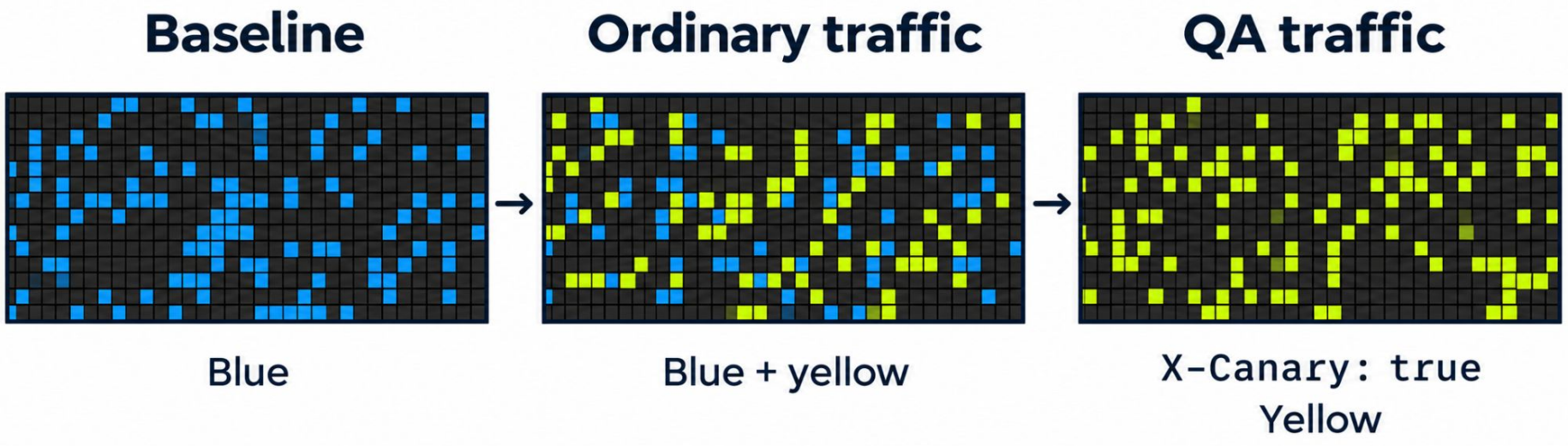
Add a Header Route to the Rollout

ArgoCon NORTH AMERICA

- Keep the weight step and add `setHeaderRoute`.
- Use `qa-team` in both `managedRoutes` and the header step

```yaml
trafficRouting:
  managedRoutes:
    - name: qa-team
steps:
  - setWeight: 50
  - setHeaderRoute:
      name: qa-team
      match:
        - headerName: X-Canary
          headerValue:
            exact: "true"
  - pause: {}
  - setWeight: 100
  - pause: {}
```

What Changes in the HTTPRoute

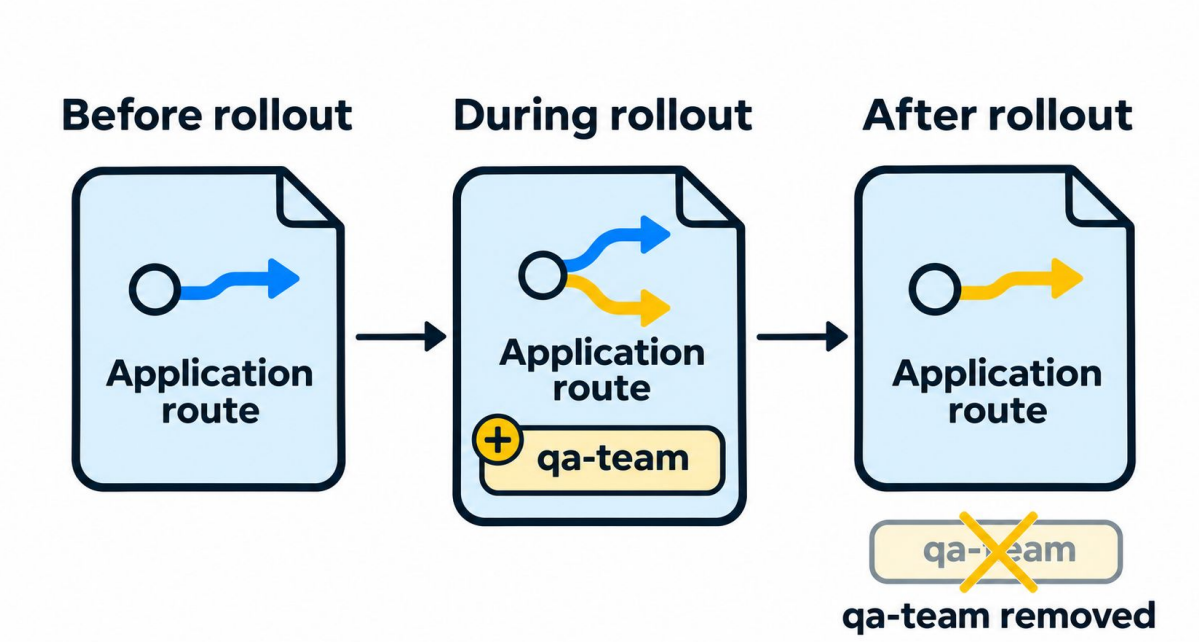


- The existing rule gets the stable and canary weights
- An added header rule points to the canary Service
- The application's path matches still apply

Let the Plugin Manage the Header Rule

ArgoCon NORTH AMERICA
KnativeCon NORTH AMERICA

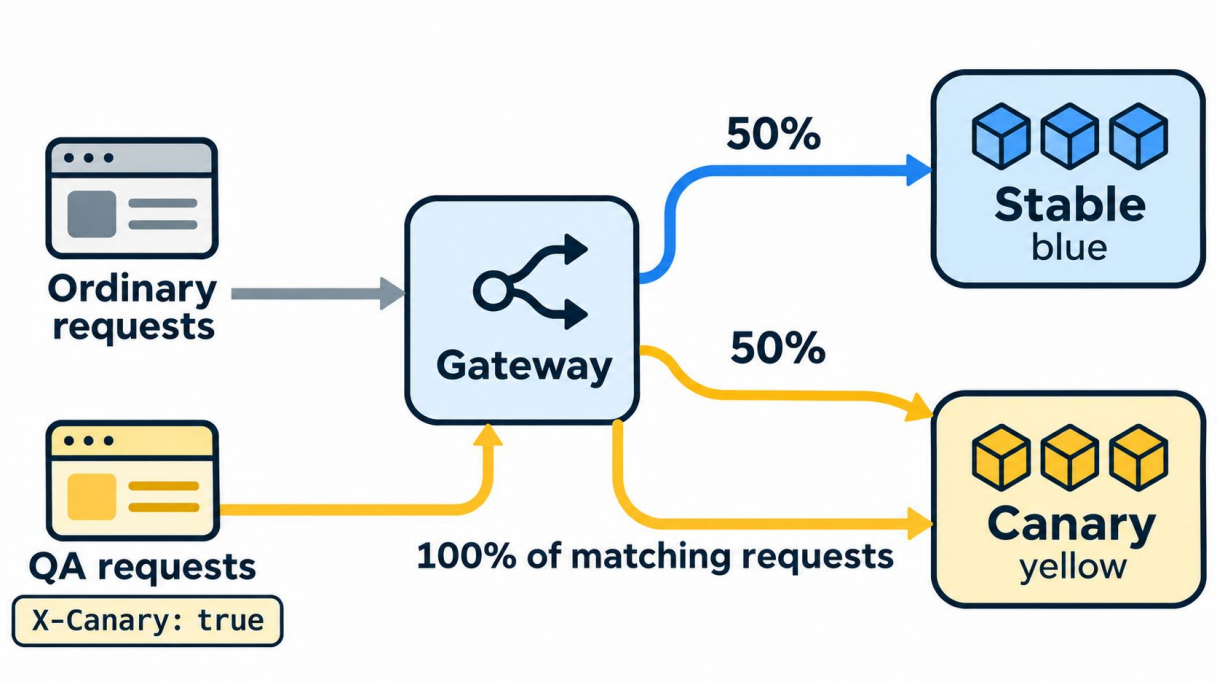
- `qa-team` names the header rule in this example
- Named rules let the plugin track what it added
- Header rules are created for the rollout
- The plugin removes them when the rollout finishes



QA Gets Canary, Everyone Else Gets the Split

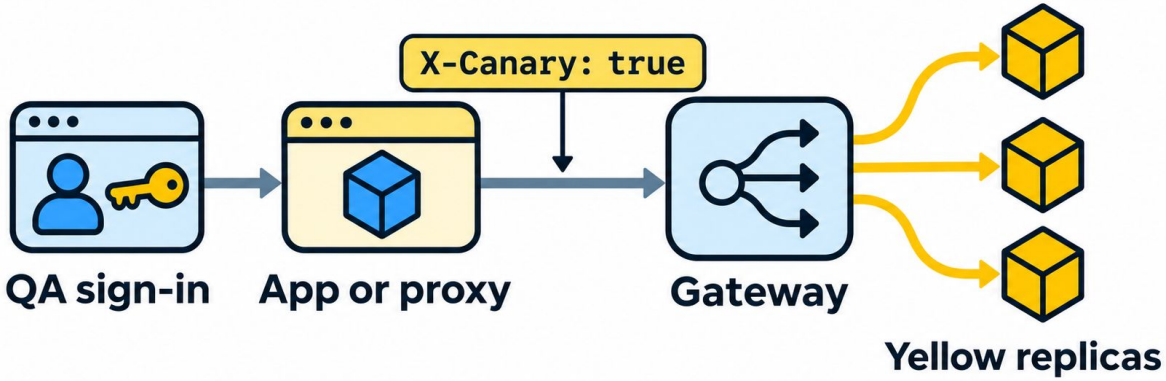
ArgoCon NORTH AMERICA
KnativeCon NORTH AMERICA

- Ordinary requests continue through the 50/50 split
- `X-Canary: true` requests go straight to yellow
- Yellow is available to ordinary traffic as well as QA
- Both routes contribute traffic to the canary



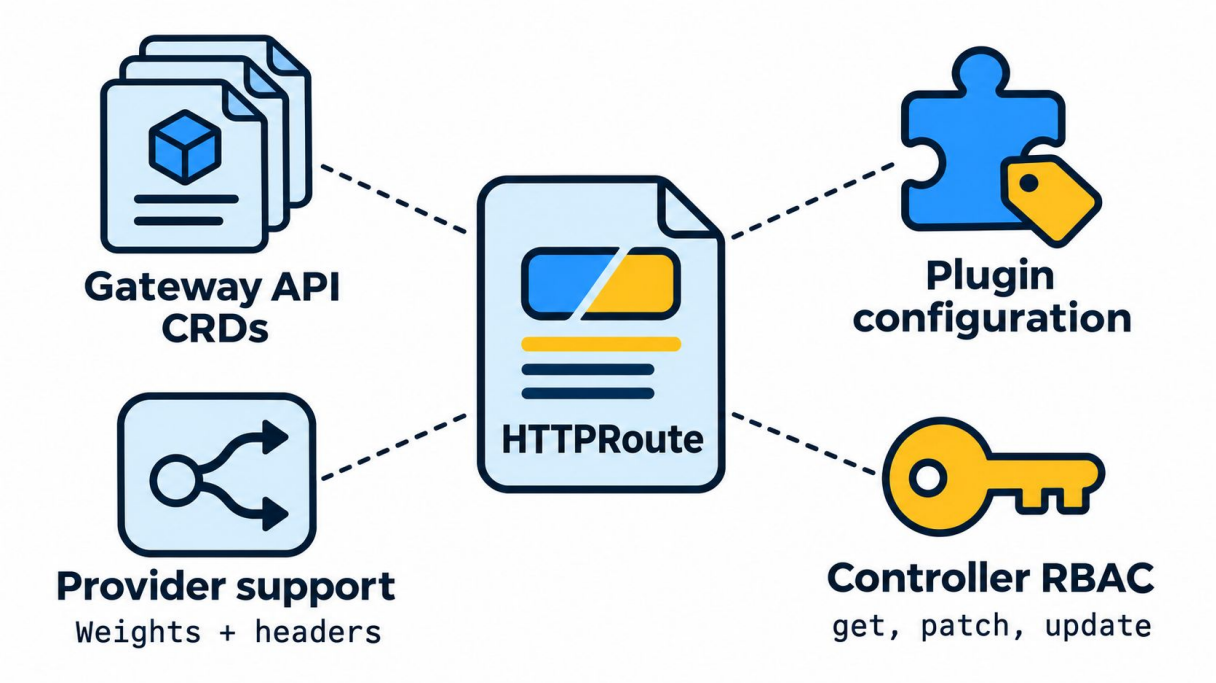
How Does QA Get the Header?

- An application or upstream proxy can add the header
- A login flow could identify QA users
- Relevant requests need to keep carrying the header
- Reaching the same version does not require the same Pod



What Needs to Be in Place?

- Gateway API CRDs with the fields the plugin needs
- A provider supporting weighted backends and header matching
- Plugin configuration matching the installed release
- Permission for the controller to update `HTTPRoute` resources





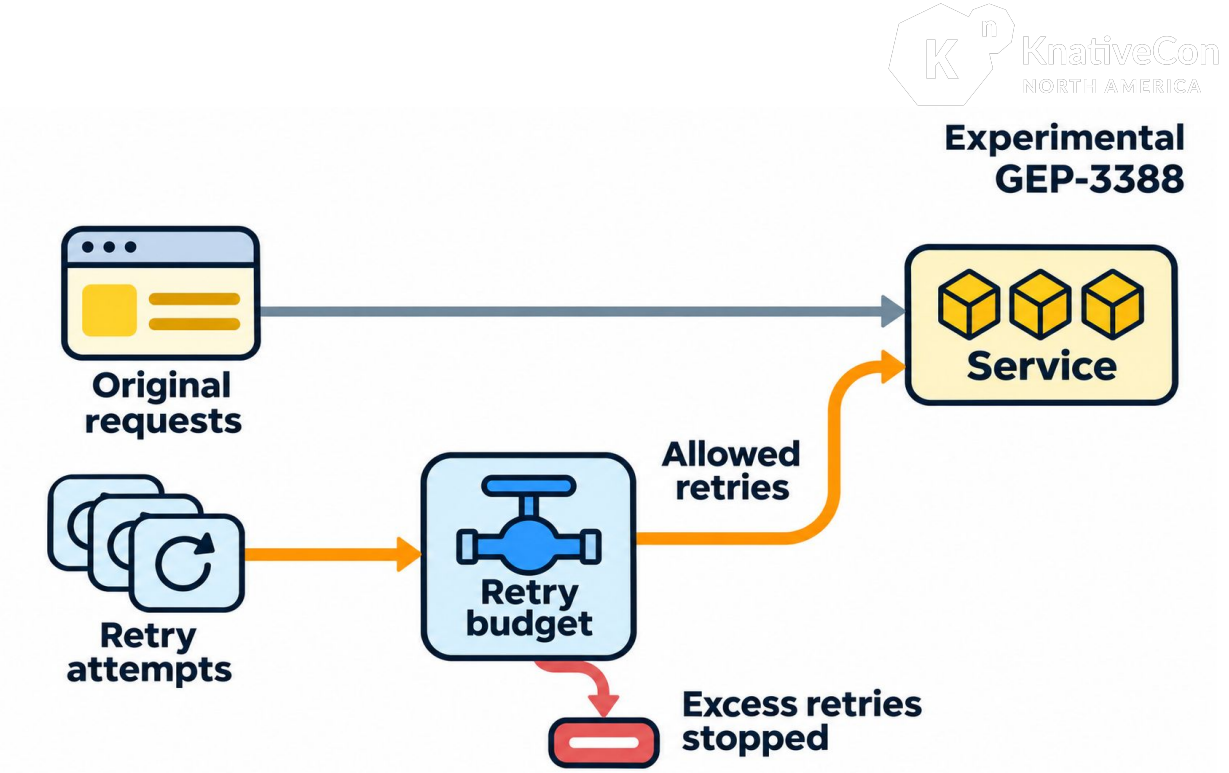
ArgoCon
NORTH AMERICA

Header-Based Canaries in Action

Live Demo 🤞

What's Next? Retry Budgets

- GEP-3388 is Experimental
- Keep retries from piling onto a failing service
- Allow useful retries without letting them dominate traffic
- A safeguard alongside your rollout checks

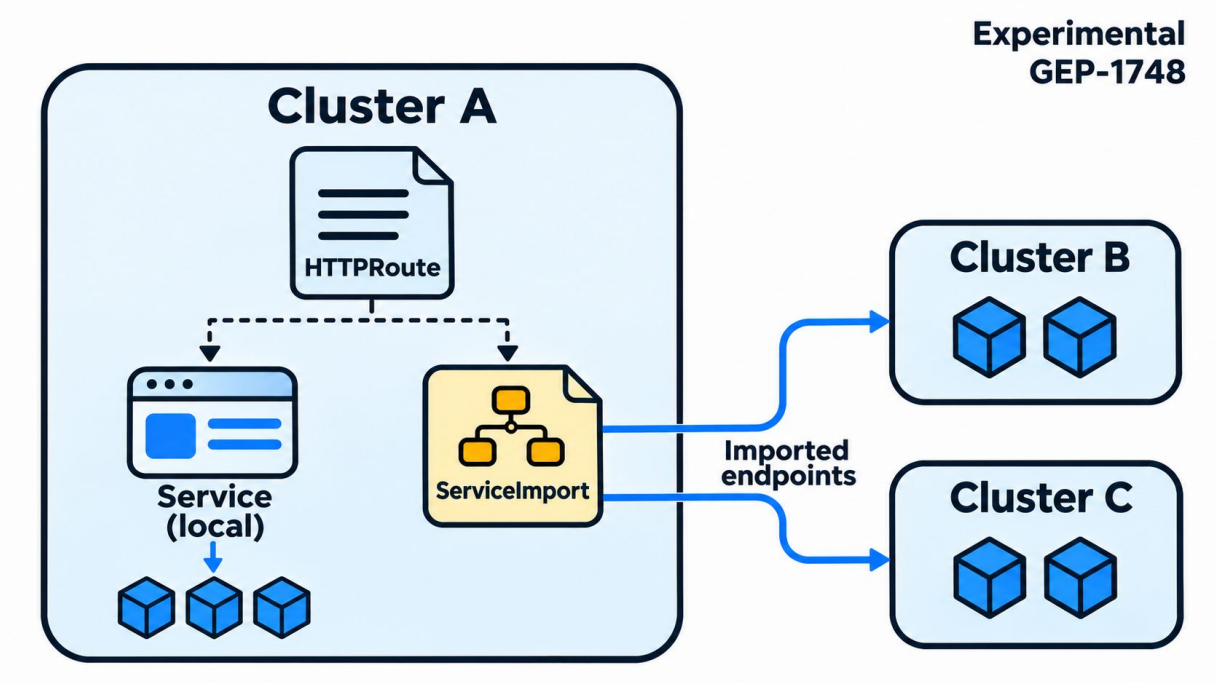


What's Next? Multi-Cluster Services

ArgoCon NORTH AMERICA

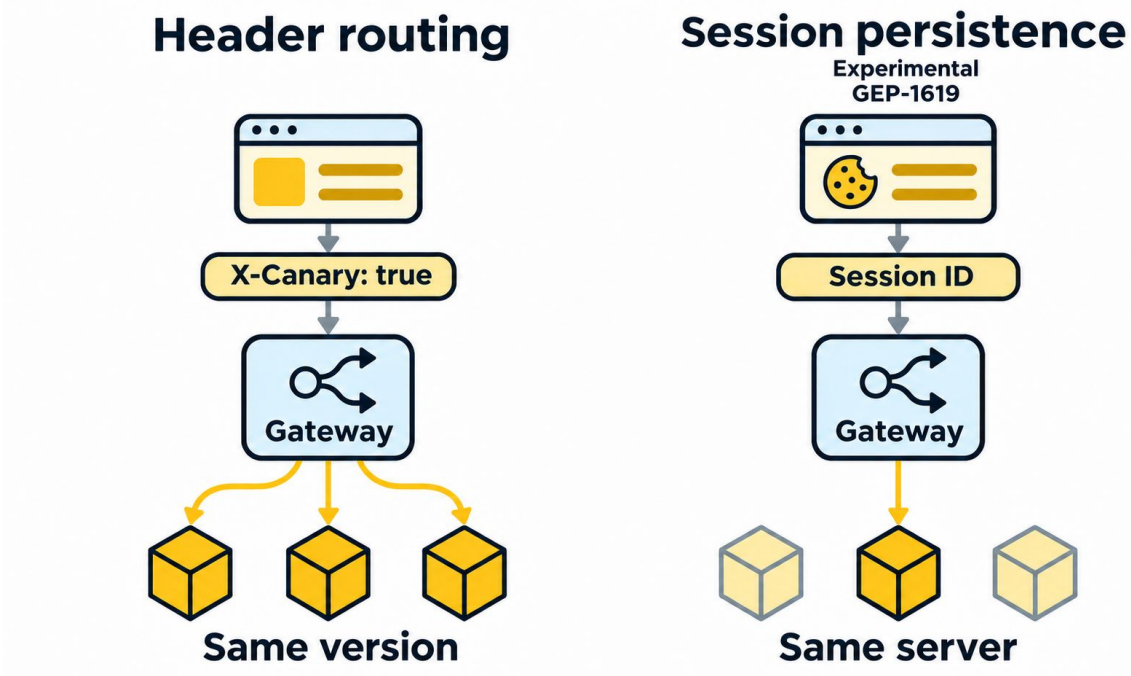
KnativeCon NORTH AMERICA

- GEP-1748 is Experimental
- Route to Services whose endpoints span clusters
- Combine a local Service with a `ServiceImport`
- Application releases across those clusters still need coordination



What's Next? Session Persistence

- GEP-1619 is Experimental
- Keep a session's requests going to the same backend server
- Cookie and header behavior are part of the ongoing work
- This goes beyond choosing stable or canary

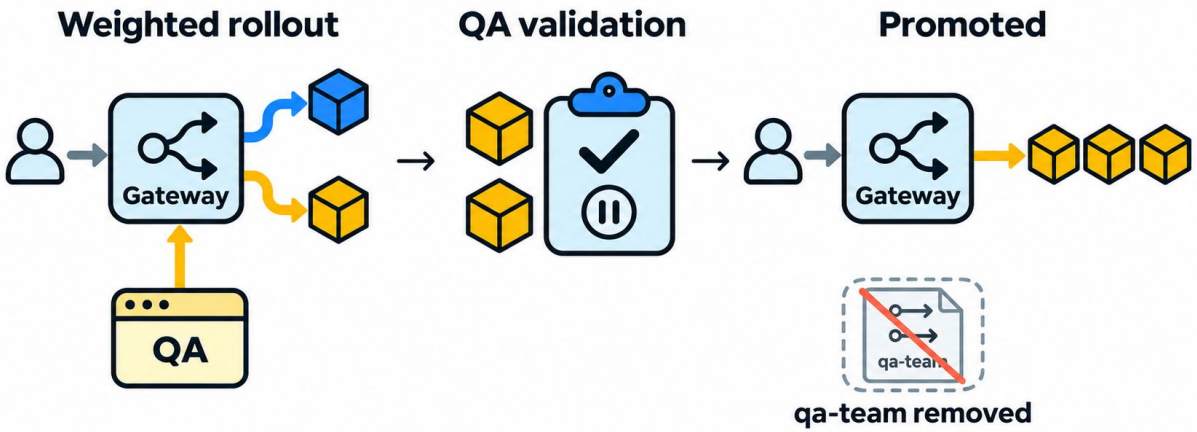


Bring Header Routing Into Your Rollouts

ArgoCon NORTH AMERICA

- Keep percentage-based rollout steps
- Give QA a consistent path to the canary
- Let the plugin manage the header rule through Gateway API
- Watch upstream features and the support needed to use them

KnativeCon NORTH AMERICA





ArgoCon
NORTH AMERICA

Thank You!

Feedback Appreciated

Progressive Delivery with Cilium Gateway API and Argo Rollouts