

Continuous Delivery Summit, May 2019

Project Showcase

Spinnaker



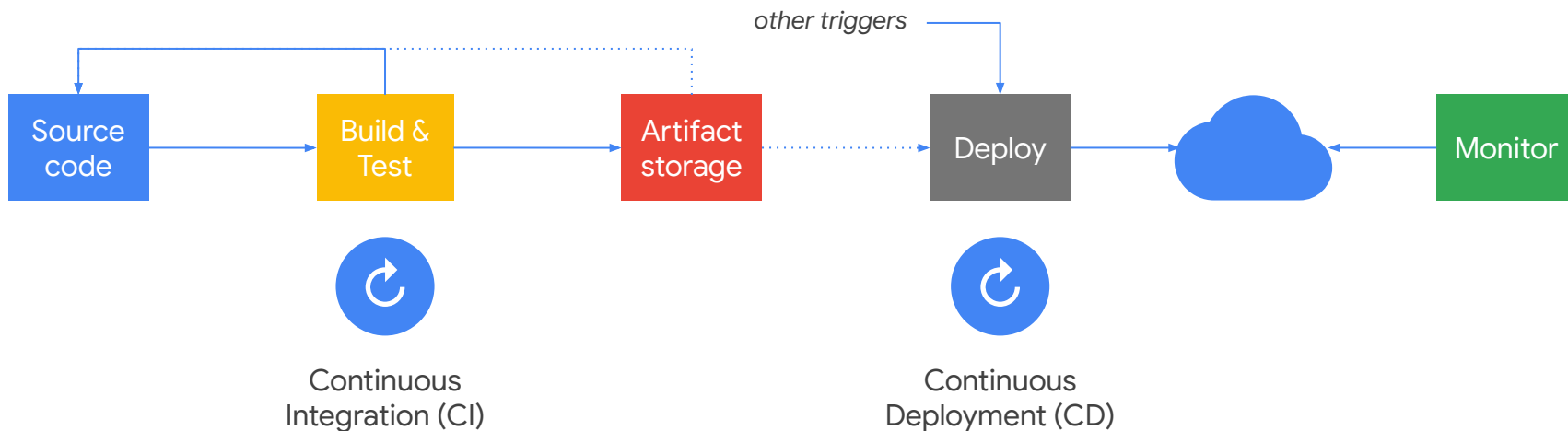
Spinnaker

Spinnaker is an open-source, multi-cloud, ***continuous delivery*** platform for releasing software changes with high velocity and confidence.

Spinnaker provides **application release** with **domain smarts** within the cloud.



Two cycles: developer feedback + application release



Different requirements!

CI & CD are both
“orchestrated
workflows”

Continuous Integration

Linear or combination of
parallel flows.

Self contained.

Failures typically indicate an
overall stop of the flow.

Generally short running:
seconds to hours.

Continuous Delivery

Complex, branching flows with
decision points & gates.

External forces.

Failures can invoke complex
paths: roll back, resize,
isolation/diagnostics.

Can be long running: hours,
days, longer....

Three core problem areas of CD



Developer experience: make it easy to get your apps running to move fast

Reliability: ensure apps can be rolled out safely to reduce risk

Control: manage and trace who can do what, when, to remain in compliance

Key components of application release



Application management

What is running where, and
how can I act on it safely?



Application deployment

How can I safely and
efficiently get release
candidates to prod?



Deployment **best practices**

How can I apply lessons
learned by experts
out-of-the-box?

Deployment best practices



Immutable
infrastructure



Release pipelines



Automated canary
analysis



Manual judgements



Blue/green deployments
and other strategies



Feature flag control



Rollbacks



Incorporate
monitoring signals

Deployment best practices



...and also needs to:

- Be simple for developers set up, and to (fully or in part) manage “as code”
- Allow for “just run my script” where necessary
- Support appropriate access control models
- Enable easy troubleshooting and traceability
- Bring as many “runtime smarts” as possible
- *and many more*

Spinnaker

Spinnaker is an open-source, multi-cloud, **continuous delivery** platform for releasing software changes with high velocity and confidence.

Spinnaker provides **application release** with **domain smarts** within the cloud.



Who's using Spinnaker?

box cerner cloudera gogo Google nest

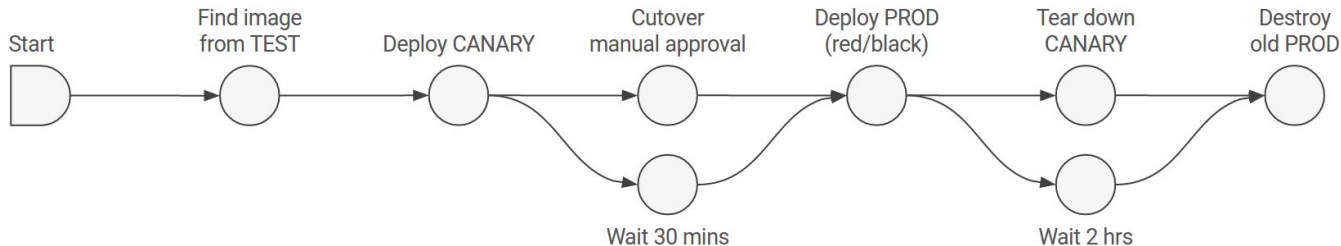
NETFLIX SCHIBSTED MEDIA GROUP TARGET waze verizon SCOPELY SEIZE THE PLAY CISCO Capital One

Optimizely UNDER ARMOUR BBM Sapiient Nike SAMSUNG SmartThings THOMSON REUTERS

Deployment actuation

Allows users to construct and manage continuous delivery workflows using **pipelines** and **stages**.

Pipelines consist of a sequence of actions, known as **stages**.



Deployment actuation

MY-KUBERNETES-ACCOUNT

Deploy to Production 1

Trigger: enabled

Configure

Start Manual Execution

DEPLOY TO STAGING
PIPELINE
[anonymous]
a minute ago

Status: **RUNNING**

Duration: 01:13

gs://spin-gcs-bucket-pkt7q2de...

gs://spin-gcs-bucket-pkt7q2de...

gcr.io/spinnaker-summit-demo...

Version sha256:9d35aa0cde2adb0e...

7 received artifacts were not consumed

Details

Bake production manifest

Diff generated manifests

Judge manifest diff

Deploy to production

Validation

Snapshot production

Cleanup

Cleanup diff

STAGE DETAILS: JUDGE MANIFEST DIFF
Duration: 00:48

Step	Started	Duration	Status
Judge manifest diff	2018-10-05 09:23:14 PDT	00:48	RUNNING

JUDGE MANIFEST DIFF

Manual Judgment

Task Status

Continue

Stop

Source | Permalink

Application management

Single pane of glass to manage global deployments across multiple clouds.

Surface health and status of running environments and metadata around deployments and individual instances.

Combines health monitoring with an actionable UI to resolve issues

The screenshot displays the Spinnaker application management interface. The top navigation bar includes 'SPINNAKER', 'Search', 'Projects', 'Applications', and a user profile 'stevenkim@spinnaker-test.net'. The main header shows 'mdservice' with tabs for 'PIPELINES', 'CLUSTERS', and 'TASKS'. On the right, there are links for 'LOAD BALANCERS', 'SECURITY GROUPS', and 'CONFIG'.

The left sidebar contains a search bar and several filter sections:

- SEARCH**: A search input field.
- ACCOUNT**: A dropdown menu with 'my-kubernetes-account' selected.
- REGION**: A dropdown menu with 'default' selected.
- STACK**: A dropdown menu with '(none)' selected.
- STATUS**: A list of status options: Healthy, Unhealthy, Disabled, Starting, Out of Service, and Unknown.
- AVAILABILITY ZONES**: A dropdown menu with 'default' selected.
- INSTANCE TYPES**: A dropdown menu.
- INSTANCE COUNT**: Input fields for 'Min:' and 'Max:'.

The main content area is titled 'Clusters' and shows two clusters under the 'MY-KUBERNETES-ACCOUNT':

- mdservice-prod**: 128 instances, 100% healthy. A 'V059' instance is highlighted with a green bar chart showing 100% health.
- mdservice-stage**: 2 instances, 100% healthy. A 'V073' instance is highlighted with a green bar chart showing 100% health.

Each cluster has a 'DEFAULT' section with a list of instances and their IDs. For example, the 'mdservice-prod' cluster has instances V059, V072, V071, and V070.

On the right, a 'Server Group Actions' menu is open for 'mdservice-prod-v059'. The menu includes options: Rollback, Resize, Disable, Destroy, and Clone. The 'Resize' option is highlighted, showing a target of '6 14:24:18 PDT' and a 'YES-ACCOUNT' button.

Below the actions menu, the 'Kube UI' for 'mdservice-prod-v059' is visible. It shows the following details:

- HEALTH**: Pods 128 a : 100%
- DEPLOYMENT**: No deployment found.
- REPLICAS**: Min 128, Max 128, Current 128.
- HORIZONTAL POD AUTOSCALING**: Desired 128, Target CPU 40%, Current CPU 0%, Latest Rescale a minute ago.

Reliable deployments

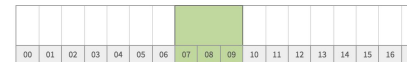
Execution Windows

☒ Restrict execution to specific time windows

Days of the Week (No days selected implies execution on any day if triggered)

Sun Mon **Tue** Wed Thu Fri Sat All None Week

Time of Day



This stage will only run within the following windows (all times in PDT):

From 07:00 to 10:00

Deploy (waiting for execution window)

Execution Windows Configuration

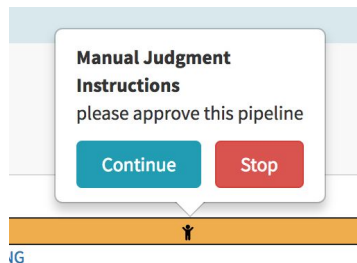
Stage execution can only run:

From 07:00 to 10:00 PDT

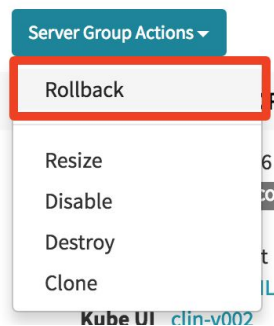
On Tue

Skip remaining window

Manual Judgements

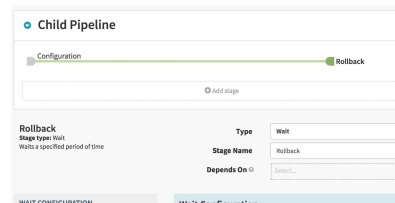


Manual Rollbacks



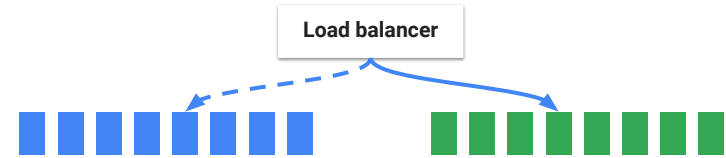
Automated Rollbacks

Trigger a pipeline that does a rollback on a failed deployment

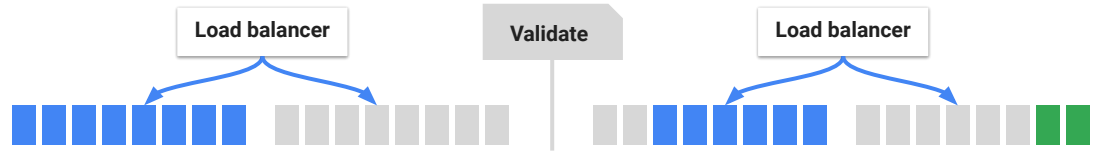


Built-in deployment strategies

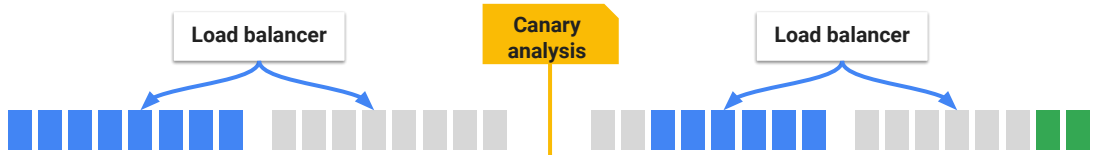
Blue/Green
(red/black)



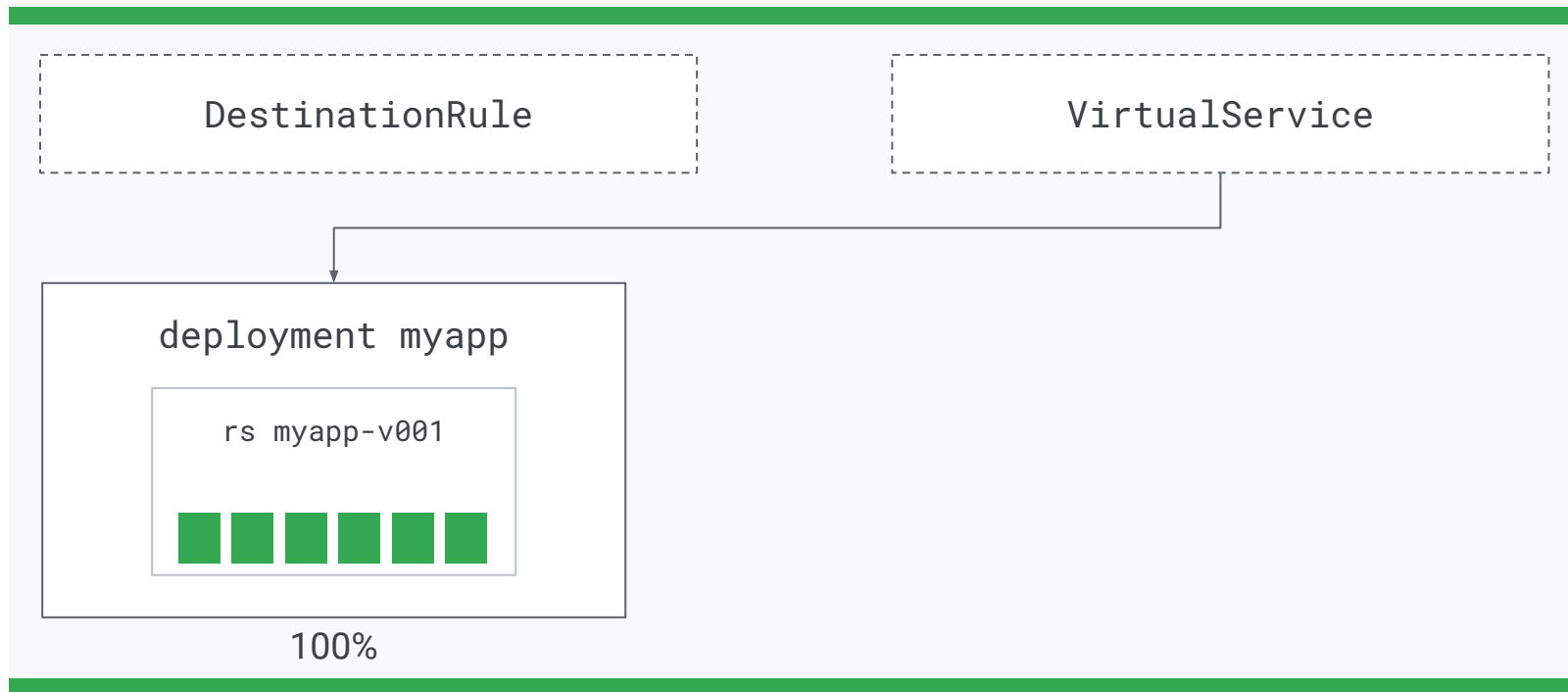
Traffic
Splitting



Automated
Canary Analysis



Domain Smarts example: Canary using Istio



Domain Smarts example: Canary using Istio

DestinationRule

subsets:

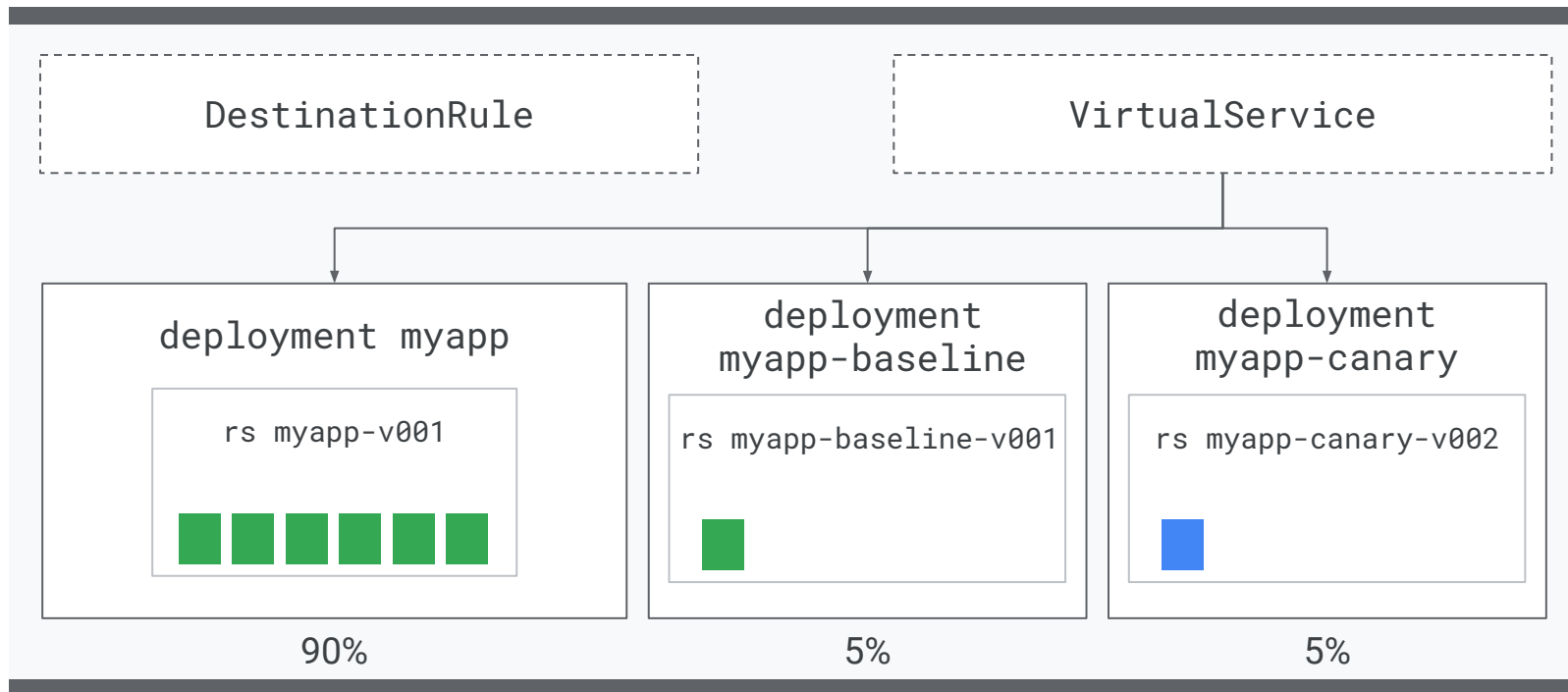
- name: myapp
labels:
app.kubernetes.io/name: myapp
app.kubernetes.io/component: server
- name: myapp-baseline
labels:
app.kubernetes.io/name: myapp
app.kubernetes.io/component: baseline
- name: myapp-canary
labels:
app.kubernetes.io/name: myapp
app.kubernetes.io/component: canary

VirtualService

http:

- route:
 - destination:
...
subset: myapp
weight: 90
 - destination:
...
subset: myapp-baseline
weight: 5
 - destination:
...
subset: myapp-canary
weight: 5

Domain Smarts example: Canary using Istio



Domain Smarts example: Canary using Istio

DestinationRule

subsets:

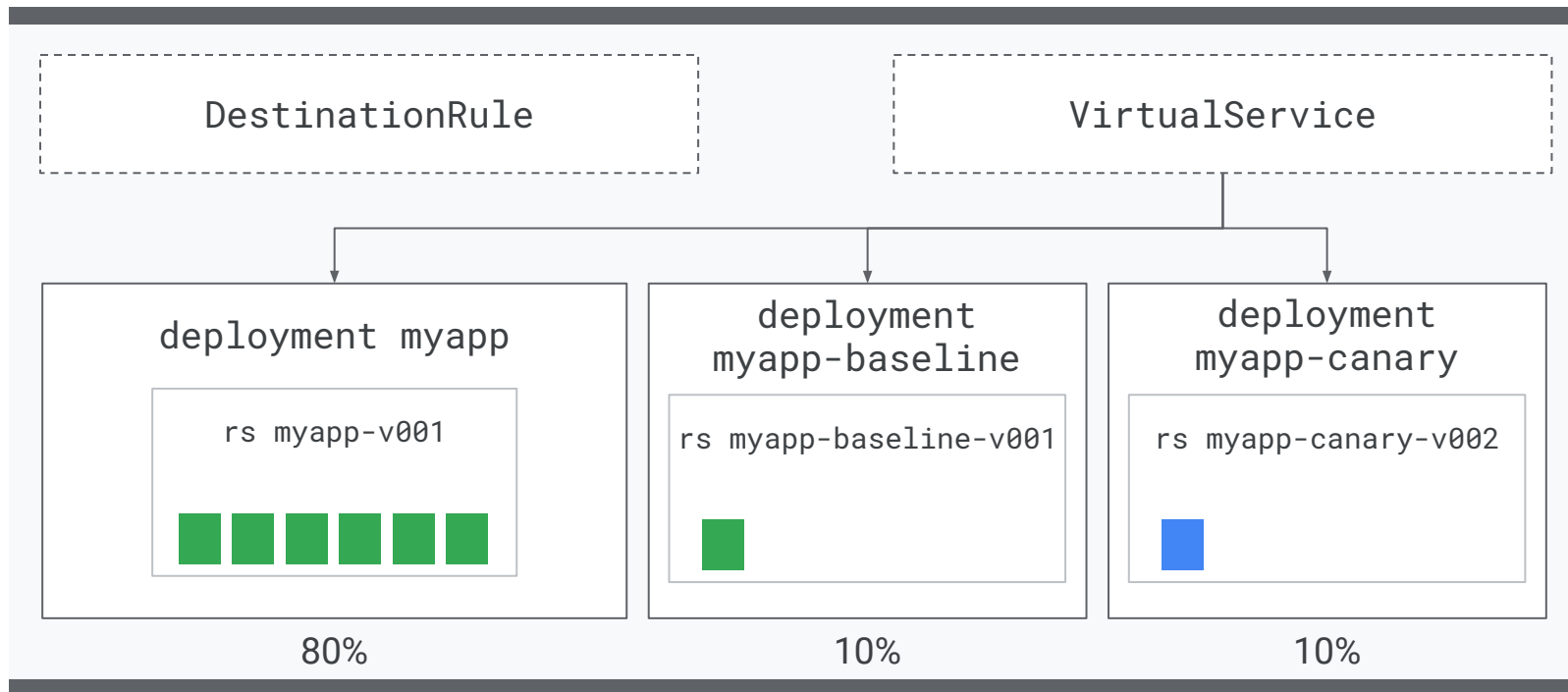
- name: myapp
labels:
 - app.kubernetes.io/name: myapp
 - app.kubernetes.io/component: server
- name: myapp-baseline
labels:
 - app.kubernetes.io/name: myapp
 - app.kubernetes.io/component: baseline
- name: myapp-canary
labels:
 - app.kubernetes.io/name: myapp
 - app.kubernetes.io/component: canary

VirtualService

http:

- route:
 - destination:
 - ...
 - subset: myapp
 - weight: 80
 - destination:
 - ...
 - subset: myapp-baseline
 - weight: 10
 - destination:
 - ...
 - subset: myapp-canary
 - weight: 10

Domain Smarts example: Canary using Istio



Domain Smarts example: Canary using Istio

DestinationRule

subsets:

- name: myapp

labels:

app.kubernetes.io/name: myapp

app.kubernetes.io/component: server

~~name: myapp baseline~~

~~labels:~~

~~app.kubernetes.io/name: myapp~~

~~app.kubernetes.io/component: baseline~~

~~name: myapp canary~~

~~labels:~~

~~app.kubernetes.io/name: myapp~~

~~app.kubernetes.io/component: canary~~

VirtualService

http:

- route:

- destination:

...

subset: myapp

~~weight: 80~~

~~destination:~~

~~...~~

~~subset: myapp baseline~~

~~weight: 10~~

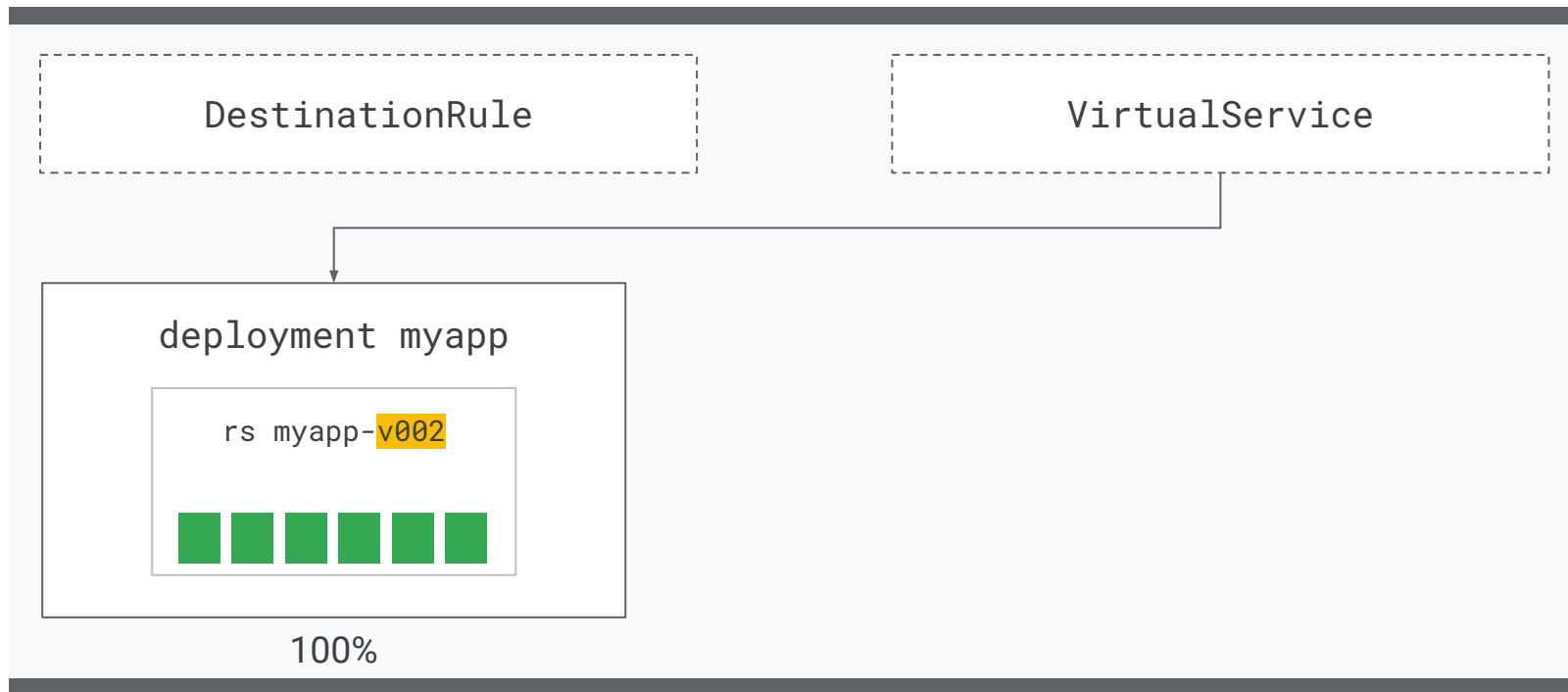
~~destination:~~

~~...~~

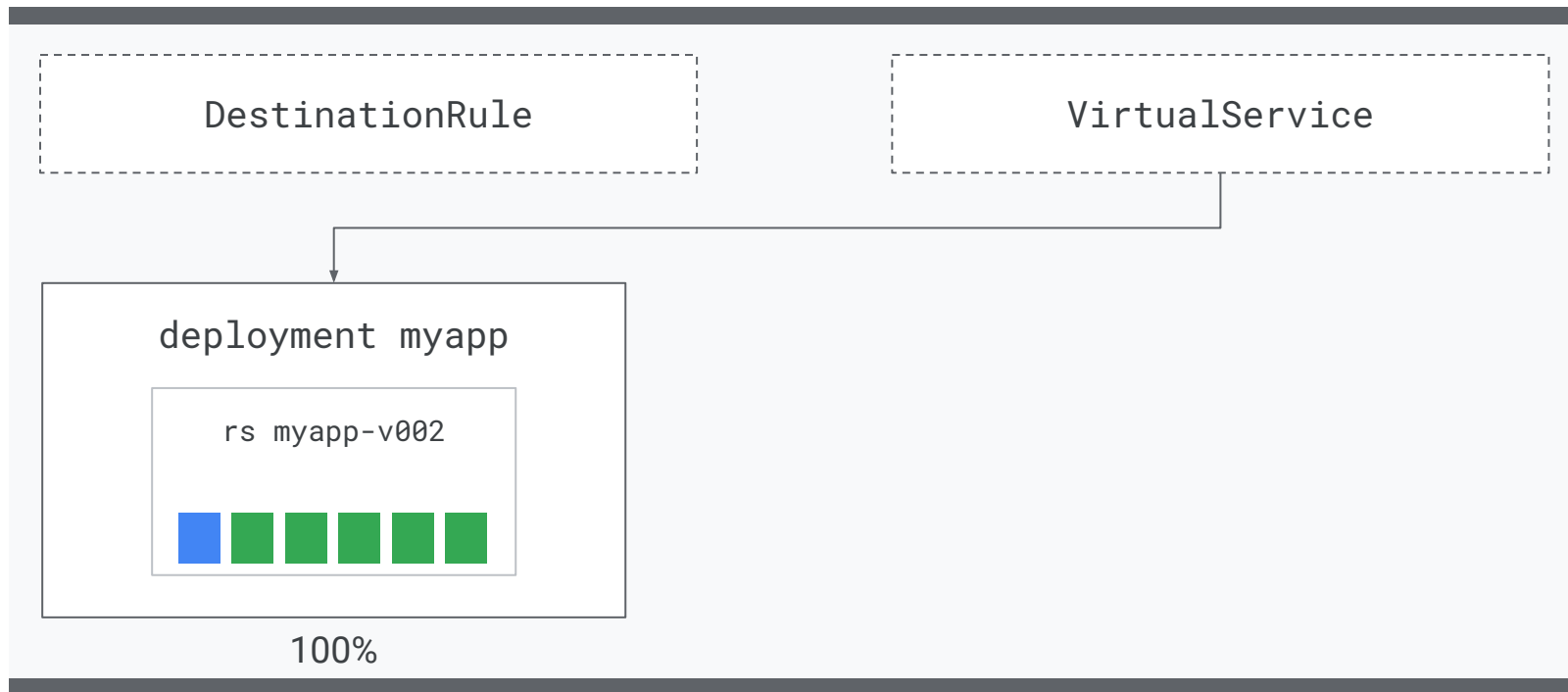
~~subset: myapp canary~~

~~weight: 10~~

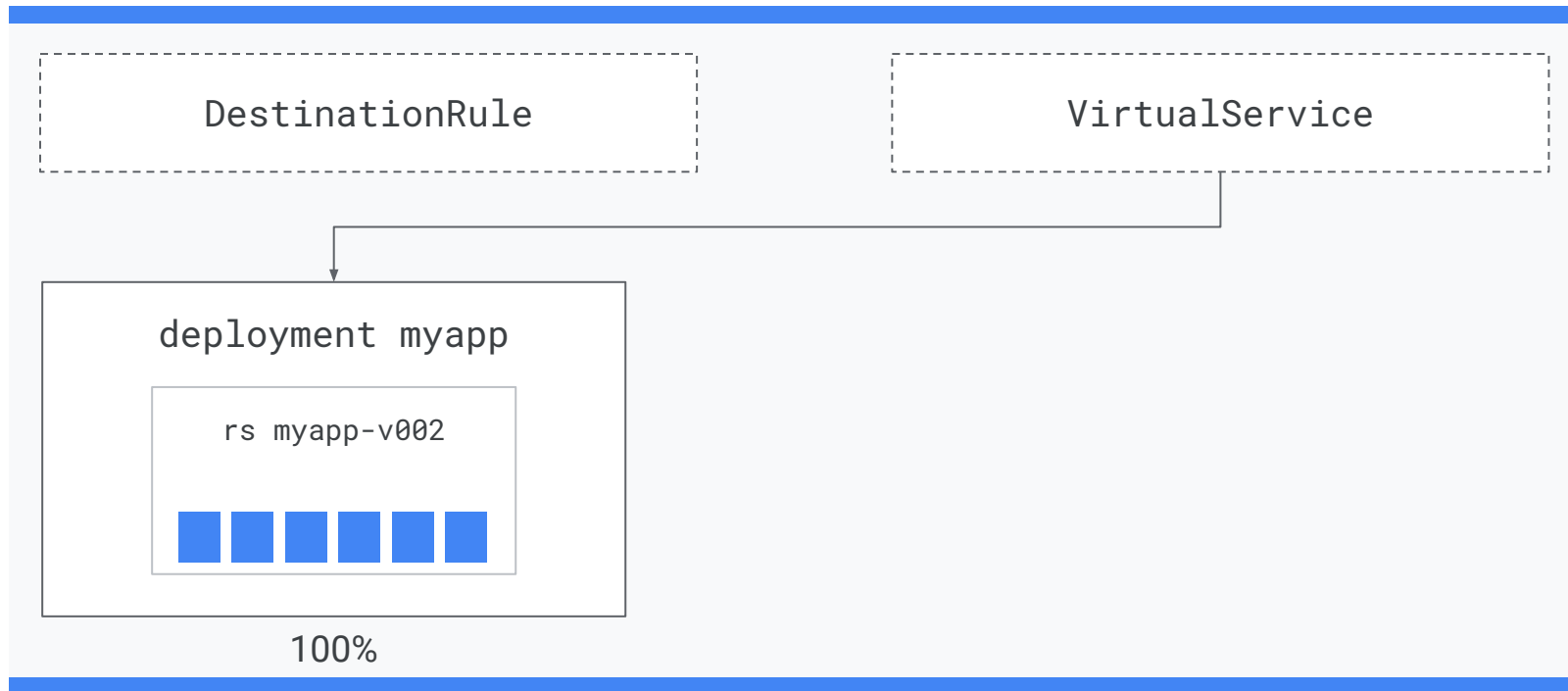
Domain Smarts example: Canary using Istio



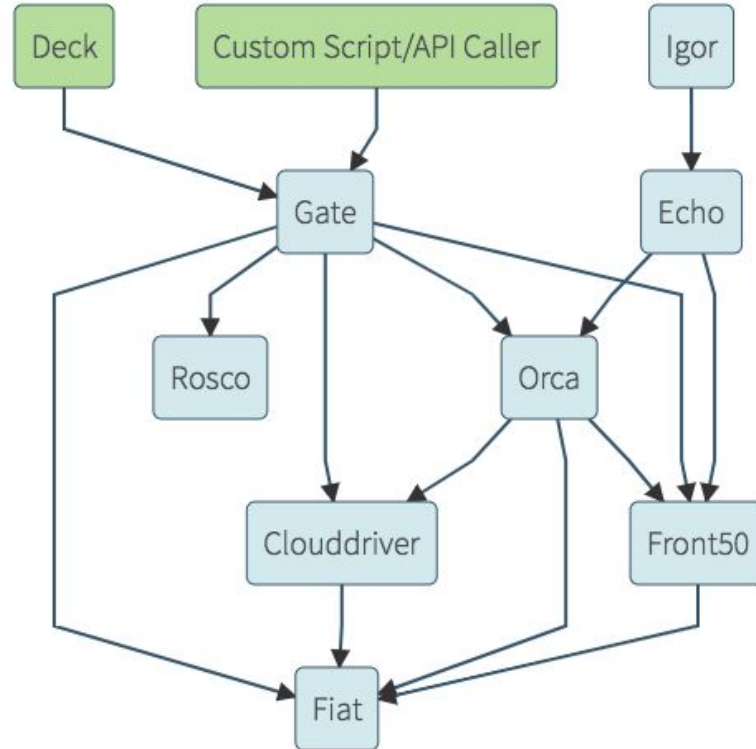
Domain Smarts example: Canary using Istio



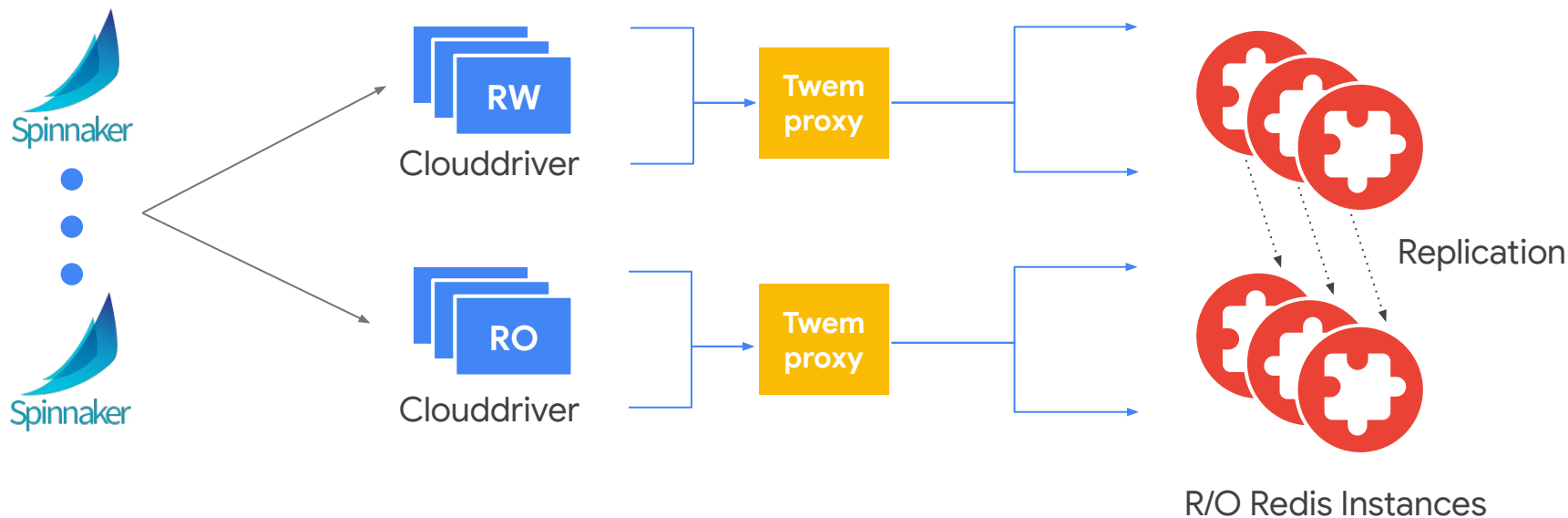
Domain Smarts example: Canary using Istio



Microservices: versatile components



Example: Spinnaker @ Target



Developer experience



Pipeline templates v2 “as code”

spin CLI for easier automation and bootstrapping

“Declarative Delivery”

Spinnaker

Spinnaker is an open-source, multi-cloud, **continuous declarative delivery** platform for releasing software changes with high velocity and confidence.

Spinnaker provides **simplified application release** with **domain smarts** within the cloud.



Links

- [Website, GitHub, Slack, forum](#)
- [Kubernetes codelab & one-line Kubernetes install](#)
- Presentation: [*From Datacenter to Cloud & 1800 Stores: Scaling Application Deployments Across Target's platforms*](#)
- Presentation: [*Large Scale Continuous Delivery at Netflix and Waze using Spinnaker*](#)
- Presentation: [*Canary Deployments With Istio and Kubernetes Using Spinnaker*](#)
- Webinar: [*Deploy like Netflix and Google with Automated Canary Analysis*](#)

