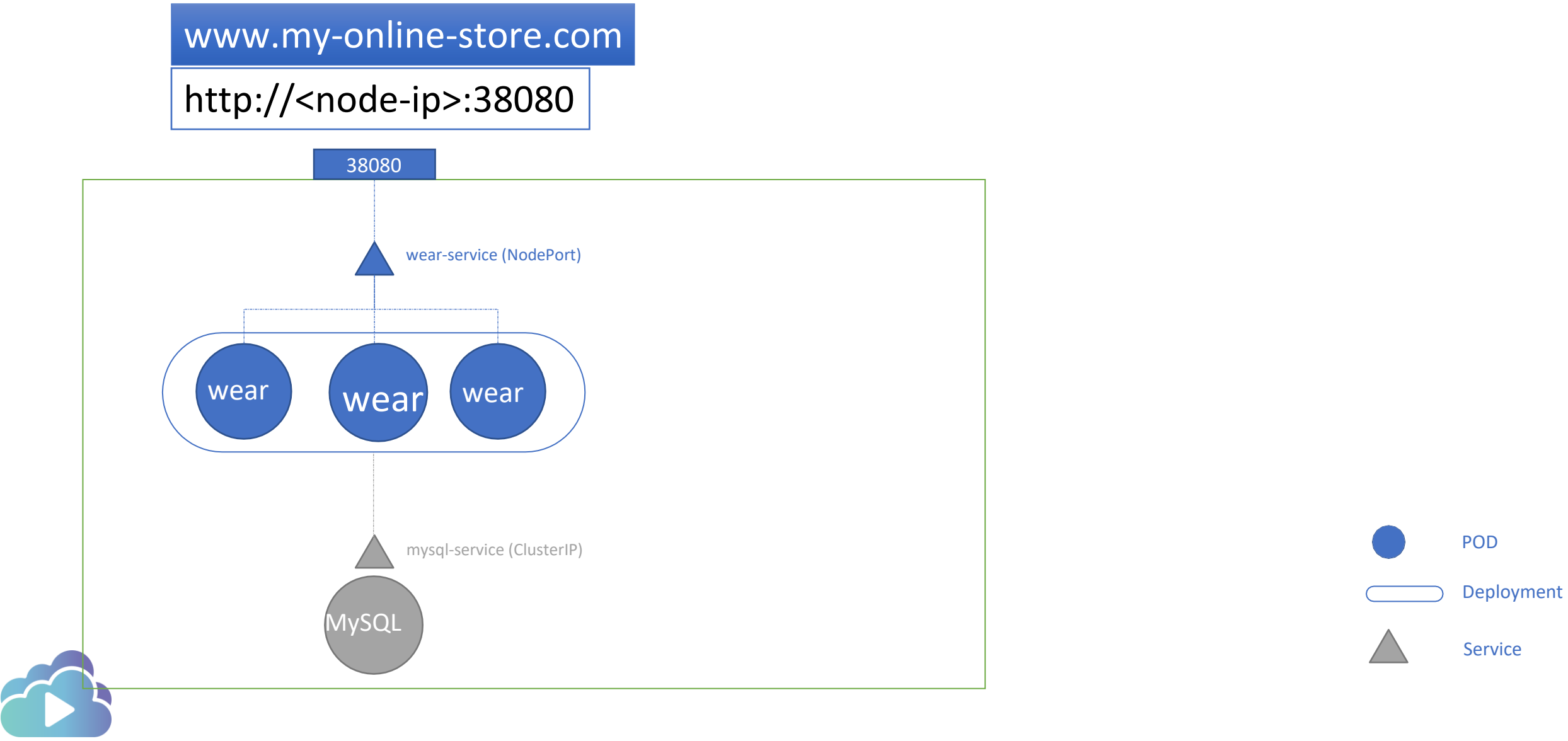
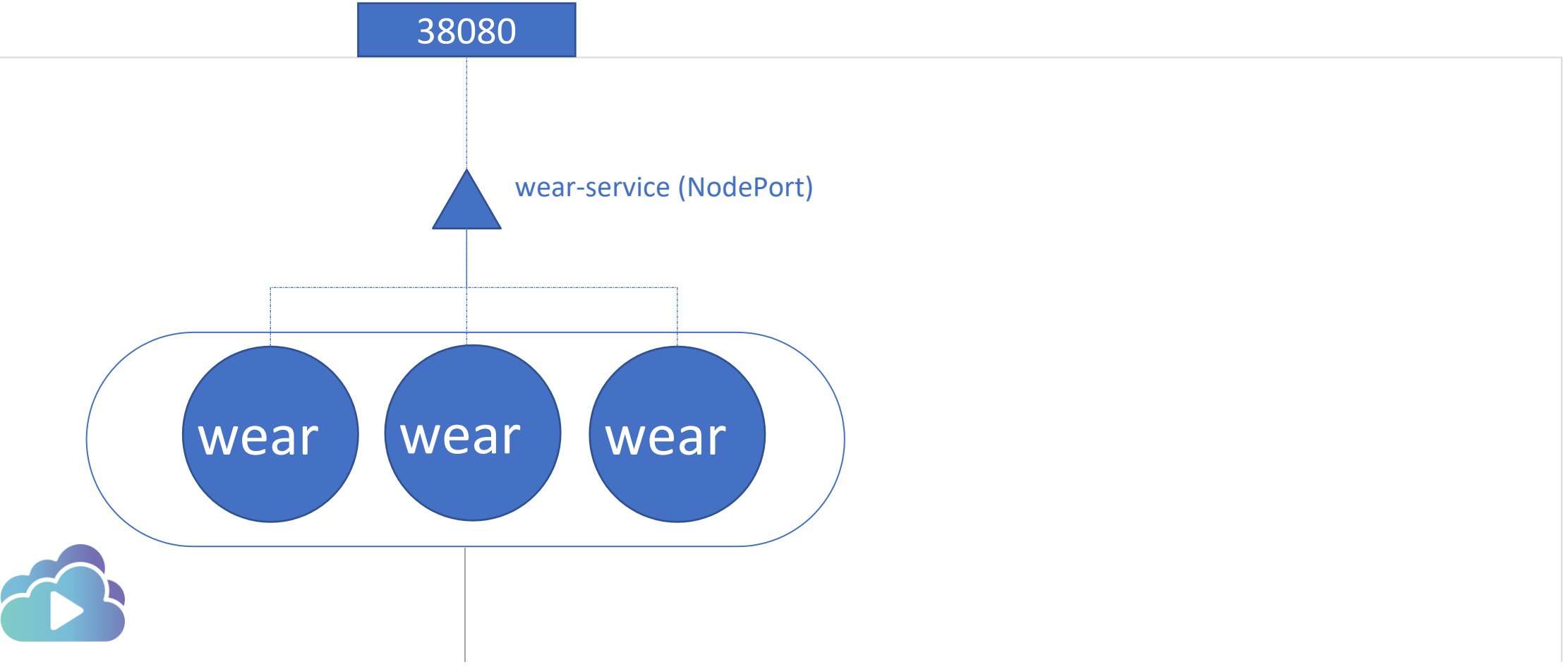
INGRESS

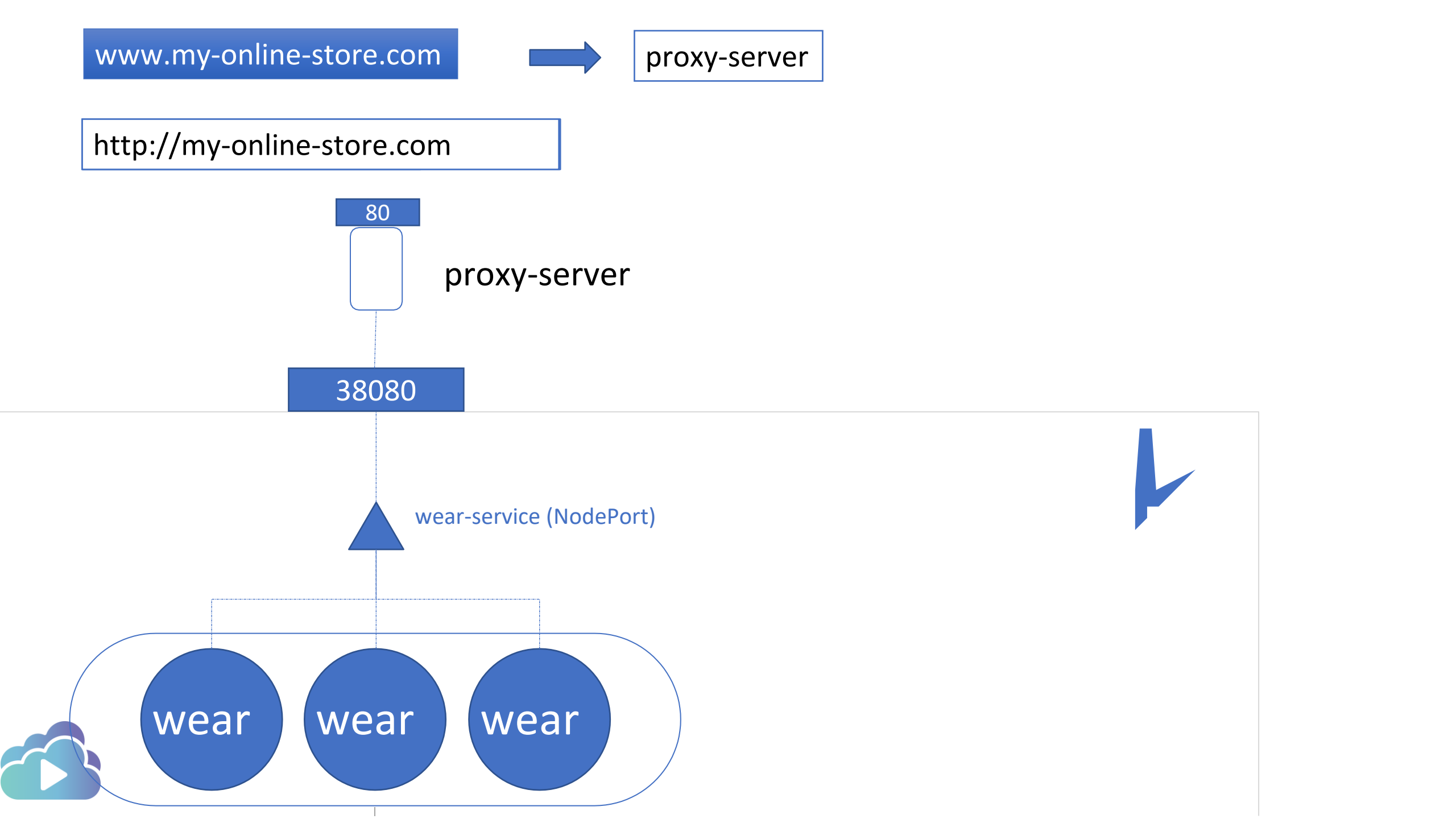
www.my-online-store.com


www.my-online-store.com
http://<node-ip>:38080
38080
wear-service (NodePort)
wear
wear
wear
mysql-service (ClusterIP)
MySQL
POD
Deployment
Service

www.my-online-store.com → <node-ip>

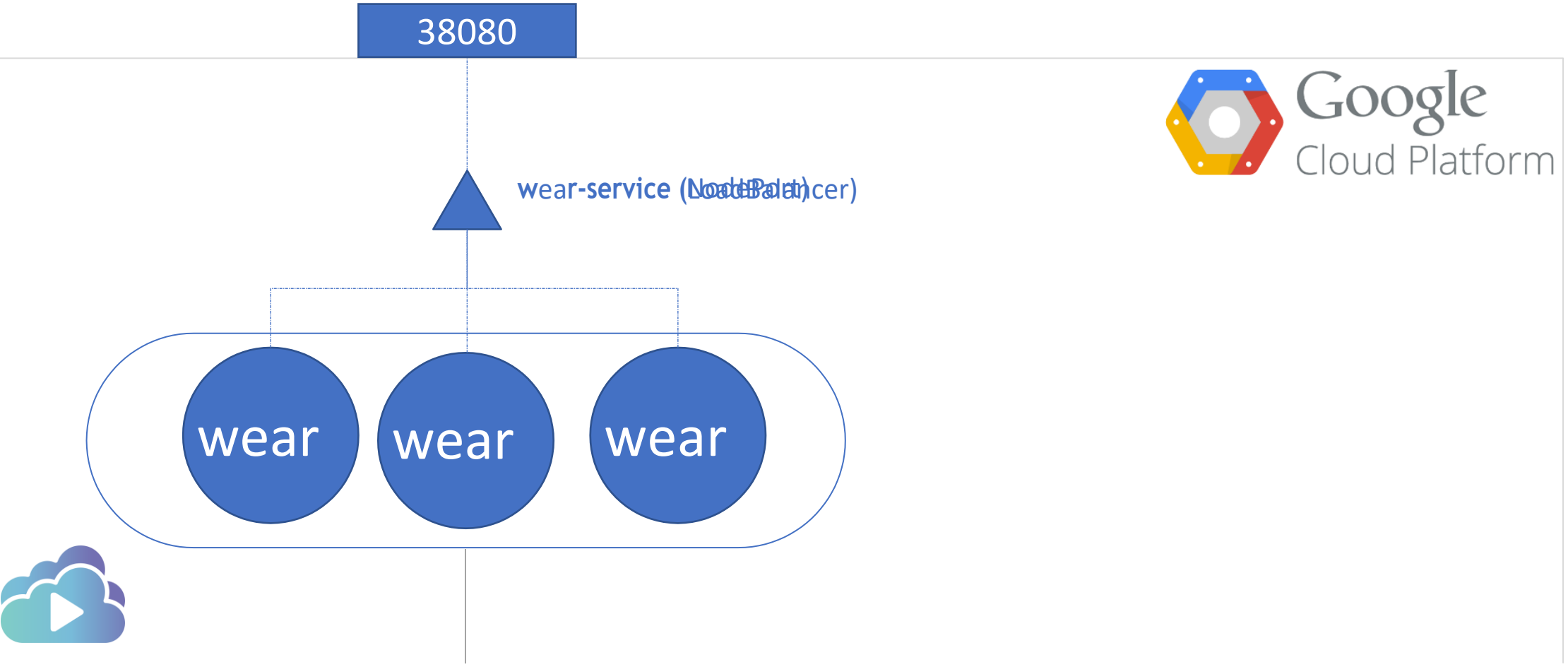
http://my-online-store.com:38080

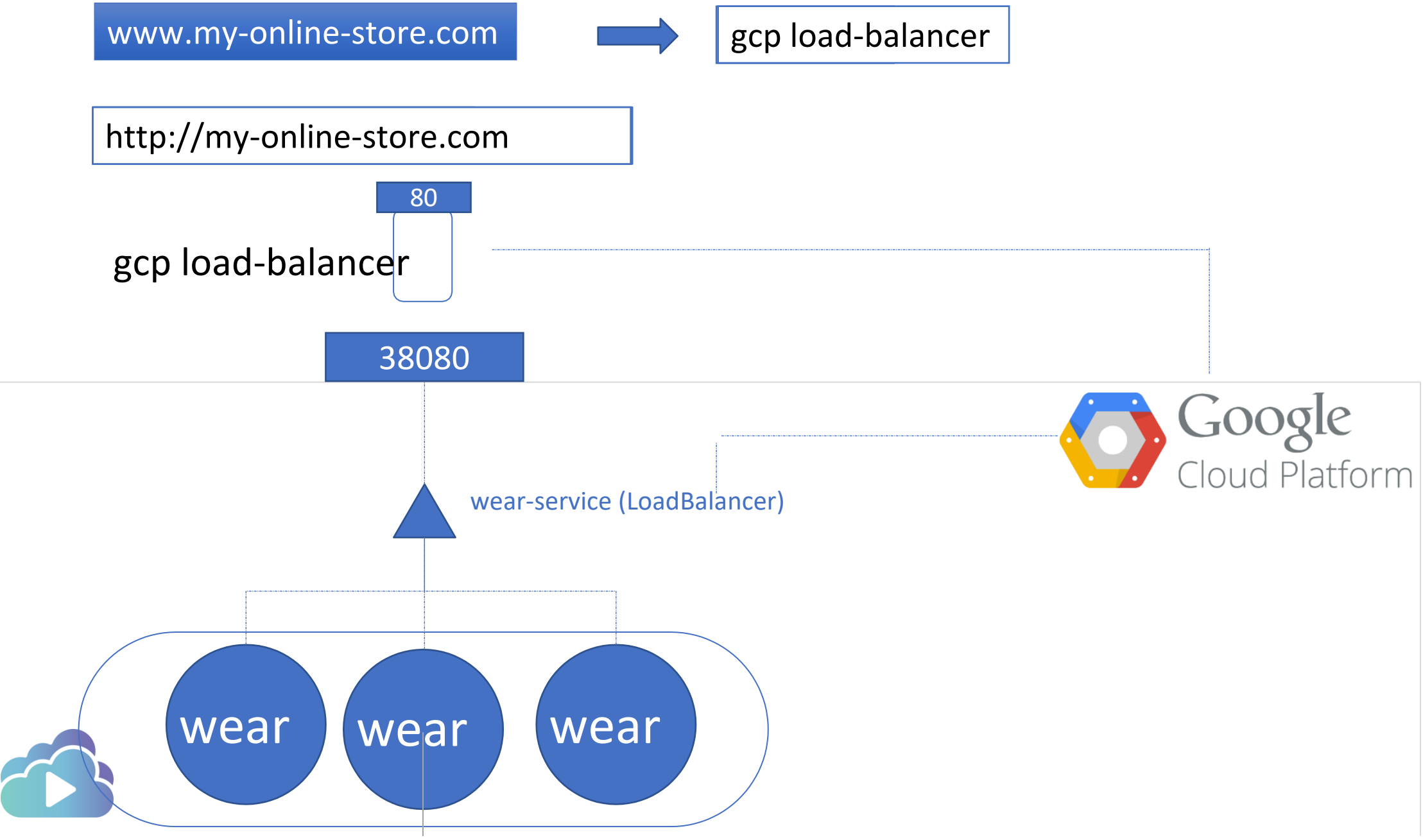



www.my-online-store.com
proxy-server
http://my-online-store.com
80
proxy-server
38080
wear-service (NodePort)
wear
wear
wear

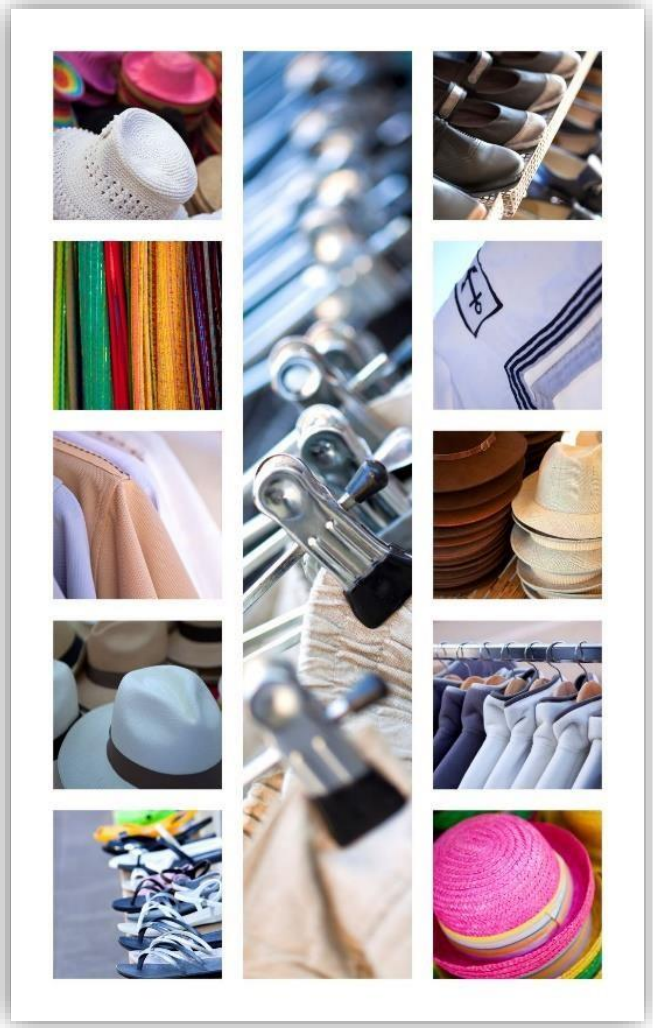
www.my-online-store.com → <node-ip>

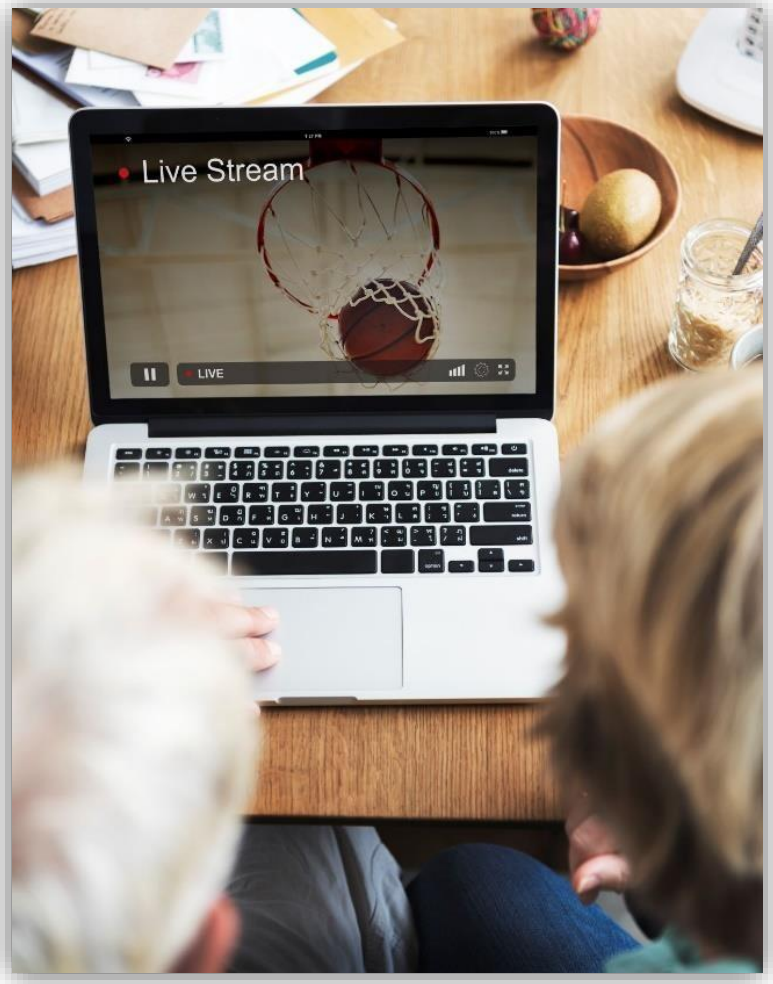
http://my-online-store.com:38080

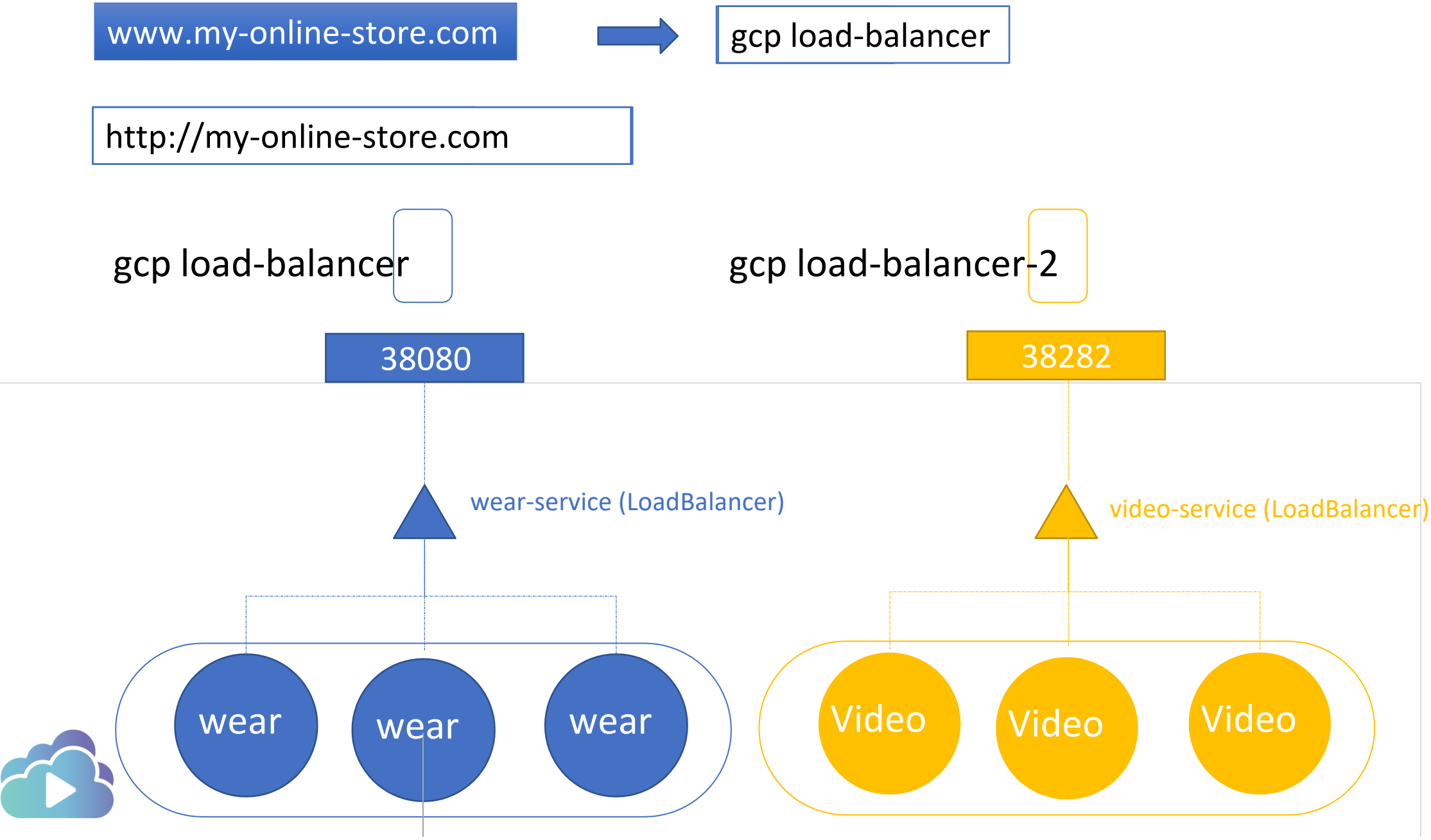


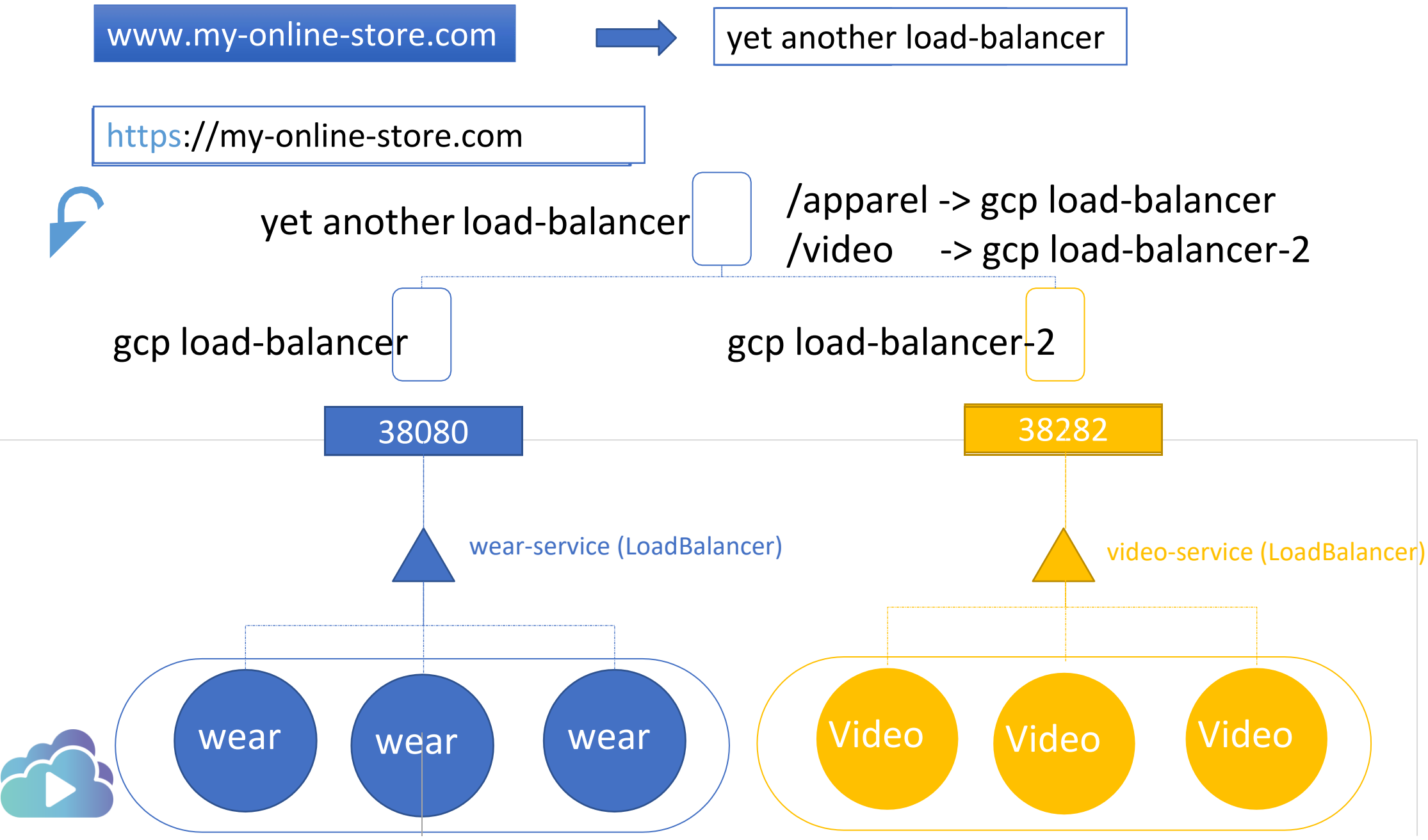

www.my-online-store.com
gcp load-balancer
http://my-online-store.com
80
gcp load-balancer
38080
Google
Cloud Platform
wear-service (LoadBalancer)
wear
wear
wear

www.my-online-store.com/wear www.my-online-store.com/watch





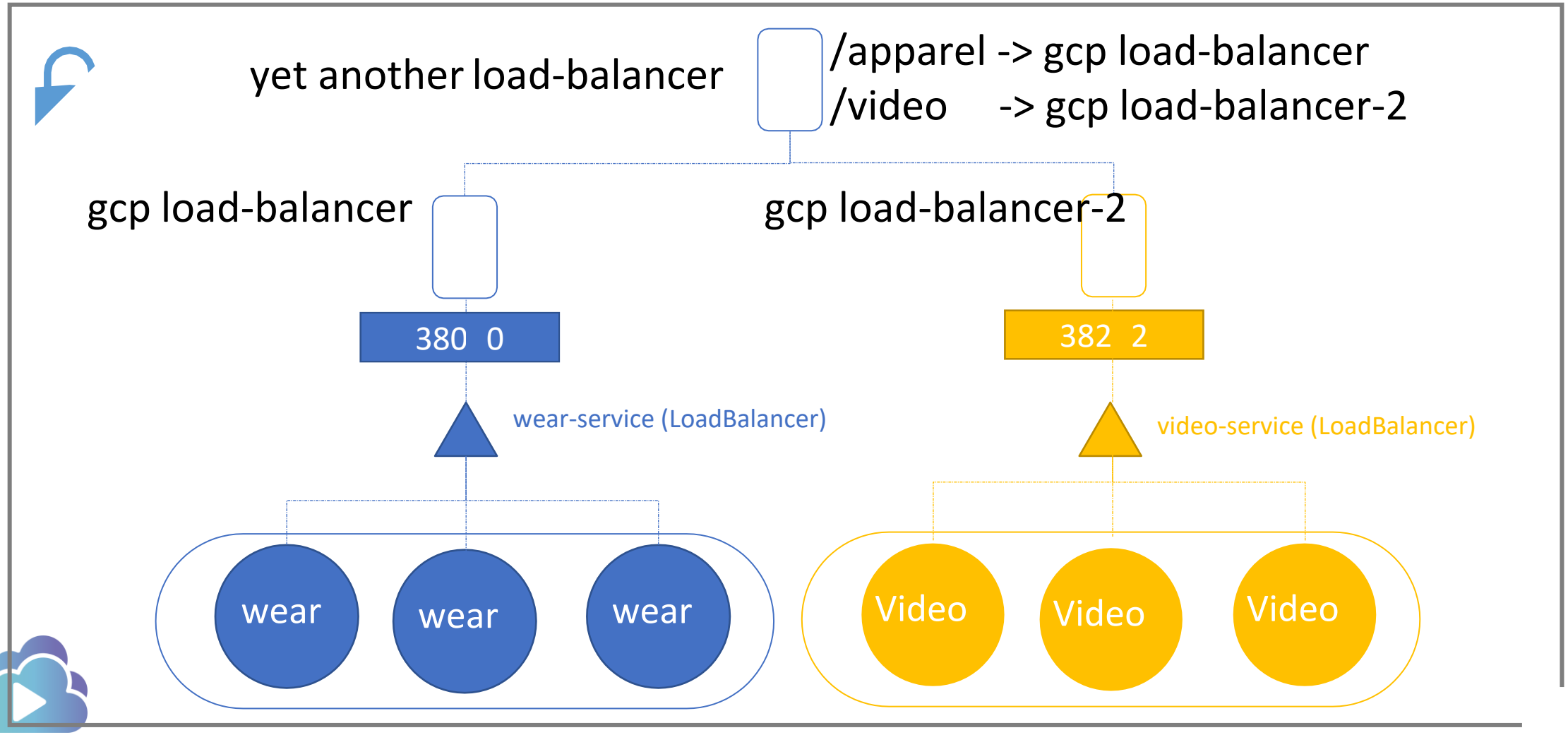

www.my-online-store.com
gcp load-balancer
http://my-online-store.com
gcp load-balancer
gcp load-balancer-2
38080
38282
wear-service (LoadBalancer)
video-service (LoadBalancer)
wear
wear
wear
Video
Video
Video


www.my-online-store.com
yet another load-balancer
https://my-online-store.com
yet another load-balancer
/apparel -> gcp load-balancer
/video -> gcp load-balancer-2
gcp load-balancer
gcp load-balancer-2
38080
38282
wear-service (LoadBalancer)
video-service (LoadBalancer)
wear
wear
wear
Video
Video
Video

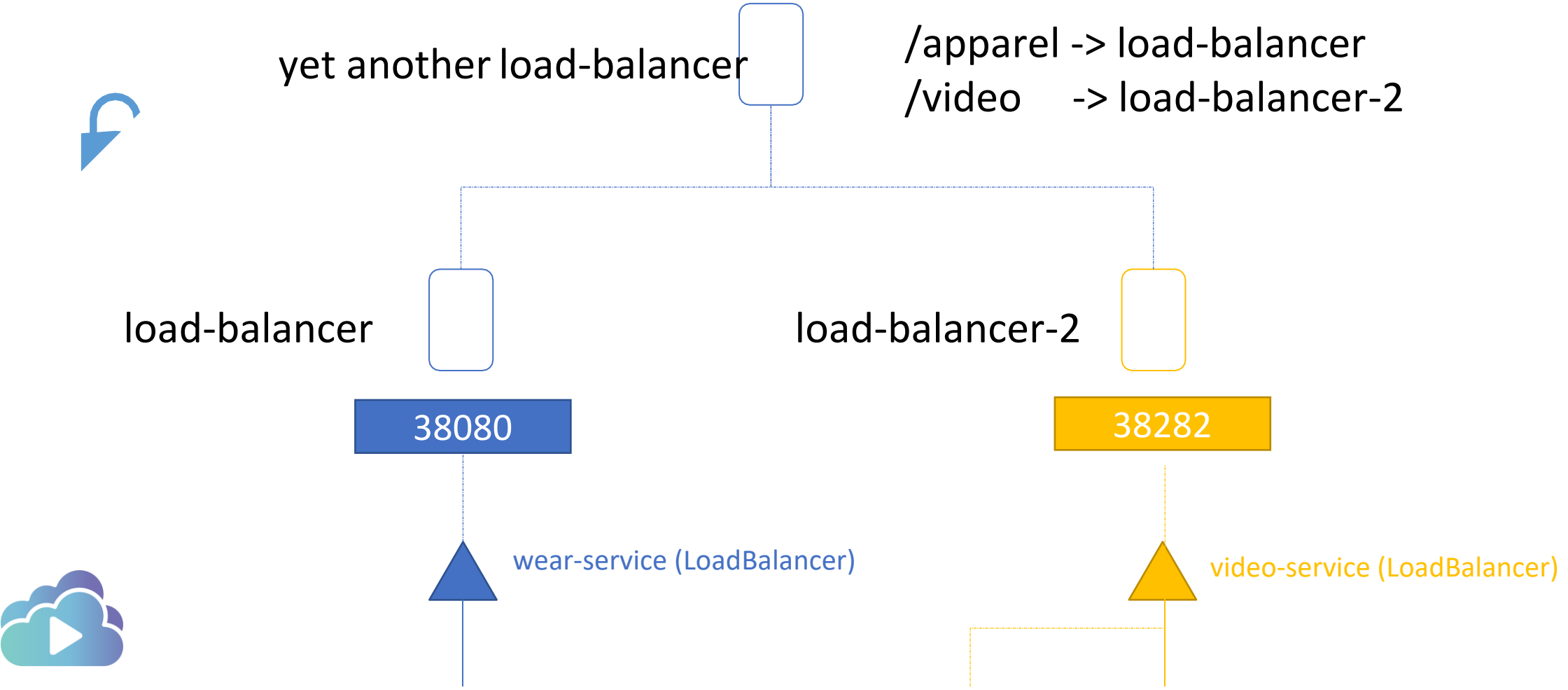
www.my-online-store.com

yet another load-balancer

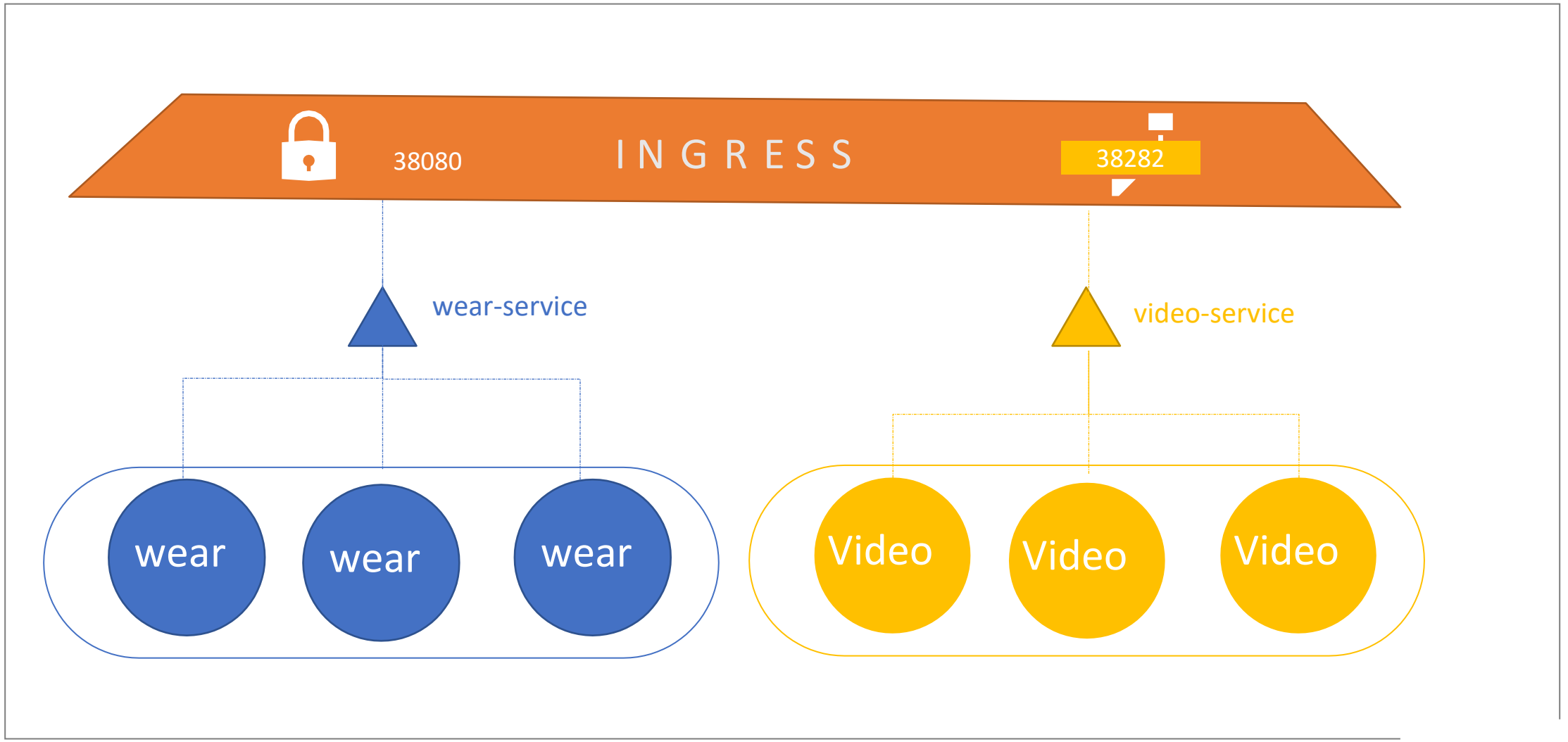
https://my-online-store.com



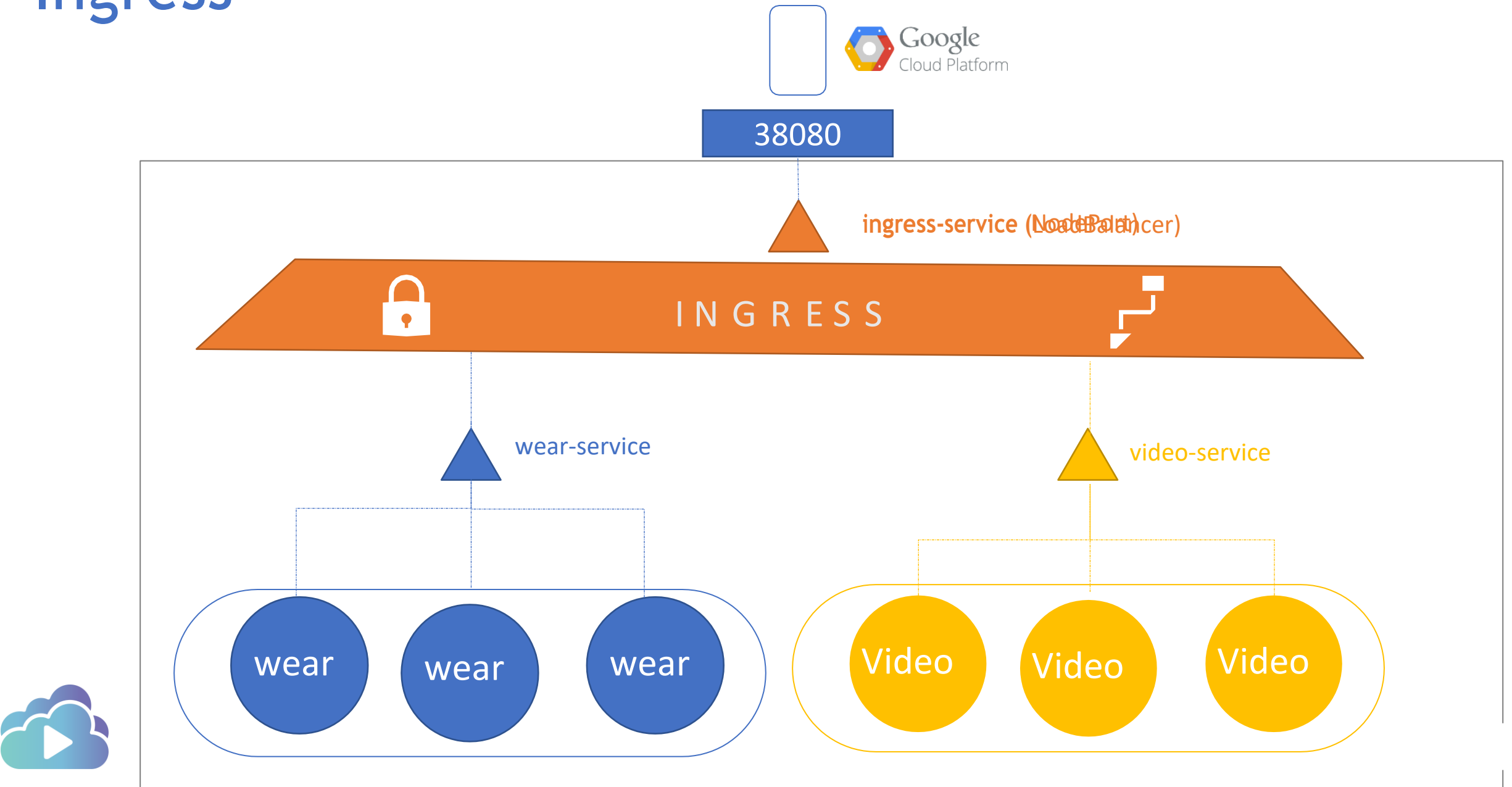
Ingress


yet another load-balancer
/apparel -> load-balancer
/video -> load-balancer-2
load-balancer
load-balancer-2
38080
38282
wear-service (LoadBalancer)
video-service (LoadBalancer)

Ingress


38080
INGRESS
38282
wear-service
video-service
wear
wear
wear
Video
Video
Video

Ingress


Google
Cloud Platform
38080
ingress-service (LoadBalancer)
INGRESS
wear-service
video-service
wear
wear
wear
Video
Video
Video

Ingress

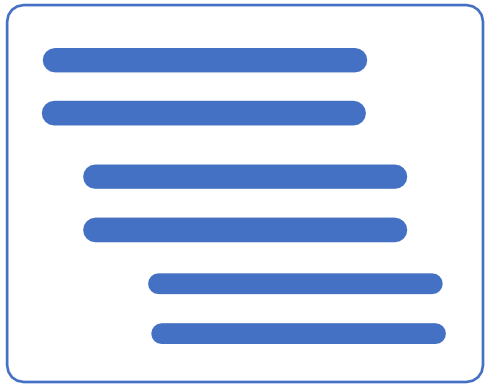
1. Deploy

INGRESS CONTROLLER

+

2. Configure

INGRESS RESOURCES



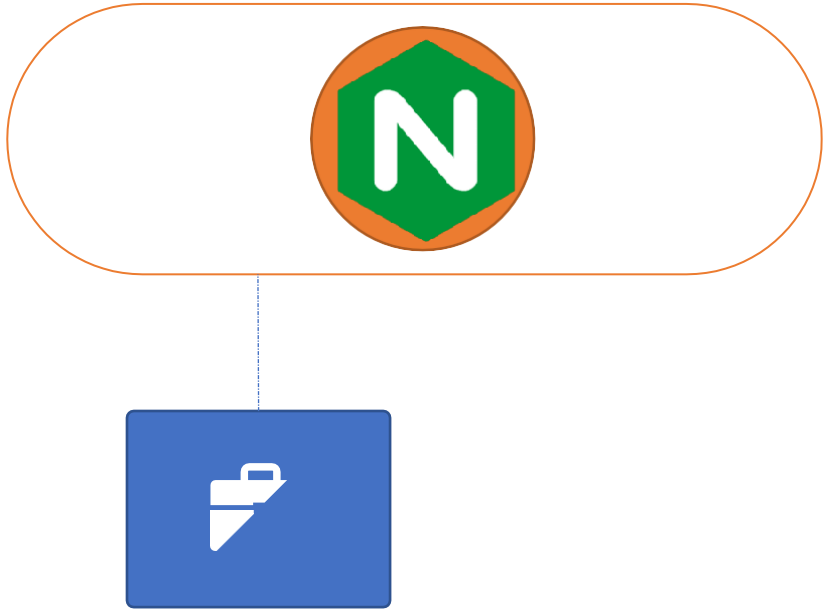
INGRESS CONTROLLER

GCP HTTP(S)
Load Balancer (GCE)

Contour

Istio

INGRESS CONTROLLER



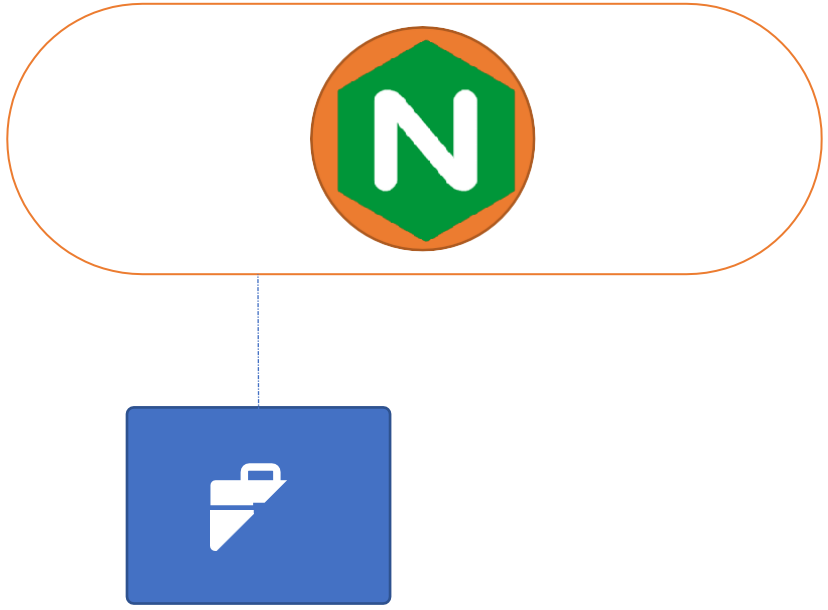
ConfigMap

nginx-configuration

```
kind: ConfigMap
apiVersion: v1
metadata:
  name: nginx-configuration
```

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: nginx-ingress-controller
spec:
  replicas: 1
  selector:
    matchLabels:
      name: nginx-ingress
  template:
    metadata:
      labels:
        name: nginx-ingress
    spec:
      containers:
        - name: nginx-ingress-controller
          image:quay.io/kubernetes-ingress-
                controller/nginx-ingress-controller:0.21.0

       args:
         - /nginx-ingress-controller
         - --configmap=$(POD_NAMESPACE)/nginx-configuration
```

INGRESS CONTROL



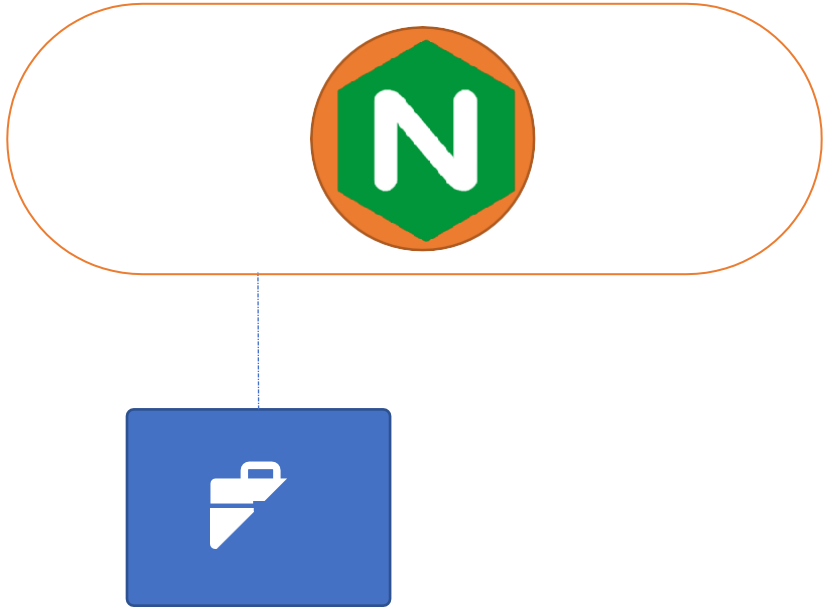
ConfigMap

nginx-configuration

```yaml
kind: ConfigMap
apiVersion: v1
metadata:
  name: nginx-configuration
```

```yaml
  name: nginx-ingress-controller
spec:
  replicas: 1
  selector:
    matchLabels:
      name: nginx-ingress
  template:
    metadata:
      labels:
        name: nginx-ingress
    spec:
      containers:
        - name: nginx-ingress-controller
          image: quay.io/kubernetes-ingress-
                 controller/nginx-ingress-controller:0.21.0

        args:
          - /nginx-ingress-controller
          - --configmap=$(POD_NAMESPACE)/nginx-configuration

        env:
          - name: POD_NAME
            valueFrom:
              fieldRef:
                fieldPath: metadata.name
          - name: POD_NAMESPACE
            valueFrom:
              fieldRef:
                fieldPath: metadata.namespace
```

INGRESS CONTROL

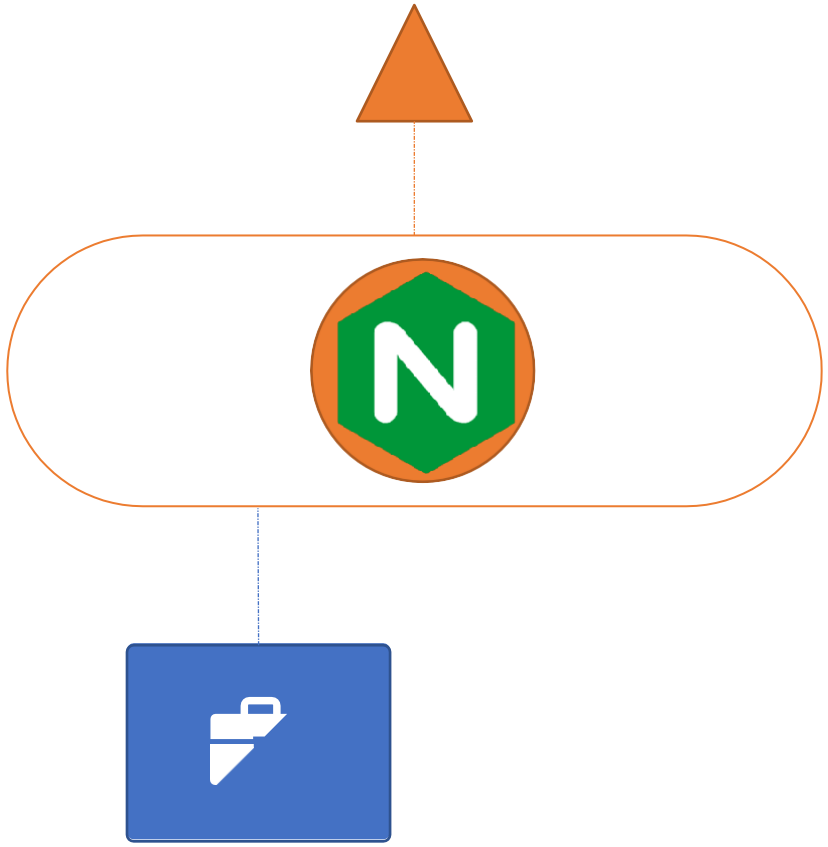


ConfigMap
nginx-configuration

```
kind: ConfigMap
apiVersion: v1
metadata:
  name: nginx-configuration
```

```
    - name: nginx-ingress-controller
      image: quay.io/kubernetes-ingress-
             controller/nginx-ingress-controller:0.21.0
  args:
    - /nginx-ingress-controller
    - --configmap=$(POD_NAMESPACE)/nginx-configuration
  env:
    - name: POD_NAME
      valueFrom:
        fieldRef:
          fieldPath: metadata.name
    - name: POD_NAMESPACE
      valueFrom:
        fieldRef:
          fieldPath: metadata.namespace

  ports:
    - name: http
      containerPort: 80
    - name: https
      containerPort: 443
```

INGRESS CONTROL



ConfigMap

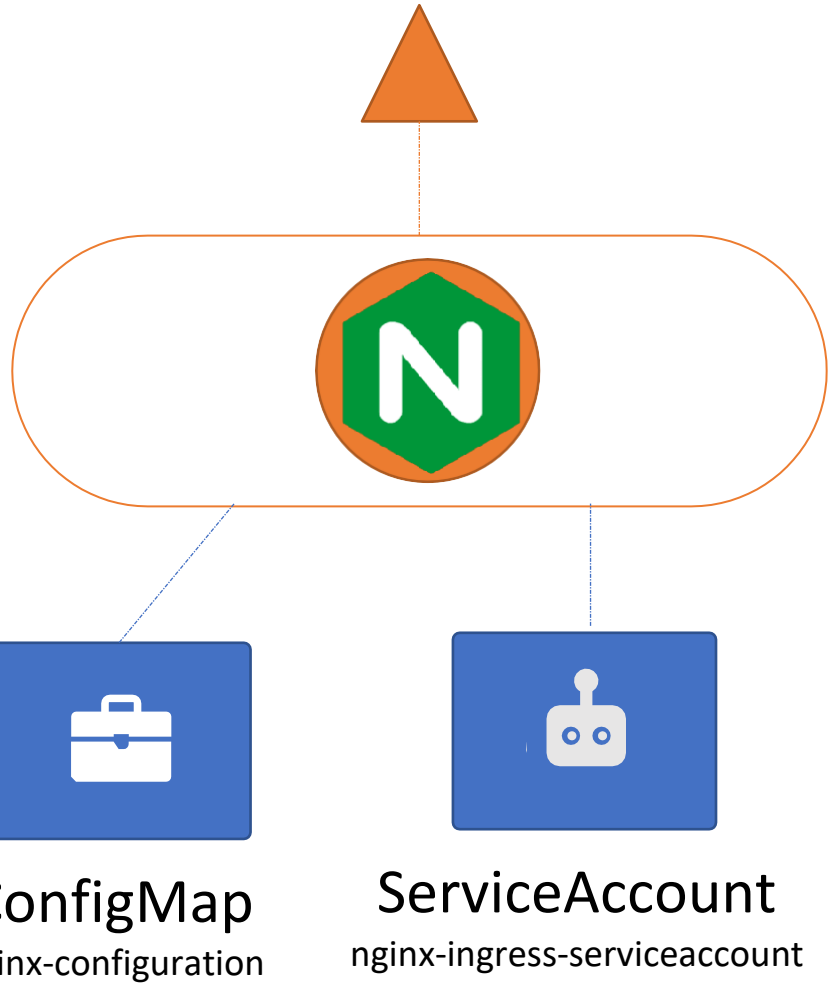
nginx-configuration

```
kind: ConfigMap
apiVersion: v1
metadata:
  name: nginx-configuration
```

```
            fieldPath: metadata.namespace
      ports:
        - name: http
          containerPort: 80
        - name: https
          containerPort: 443
```

```
apiVersion: v1
kind: Service
metadata:
  name: nginx-ingress
spec:
  type: NodePort
  ports:
  - port: 80
    targetPort: 80
    protocol: TCP
    name: http
  - port: 443
    targetPort: 443
    protocol: TCP
    name: https
  selector:
    name: nginx-ingress
```

INGRESS CONTRO



ConfigMap
nginx-configuration

ServiceAccount
nginx-ingress-serviceaccount

```
    protocol: TCP
    name: http
  - port: 443
    targetPort: 443
    protocol: TCP
    name: https
  selector:
    name: nginx-ingress
```

```
apiVersion:  v1
kind: ServiceAccount
metadata:
  name: nginx-ingress-serviceaccount
```

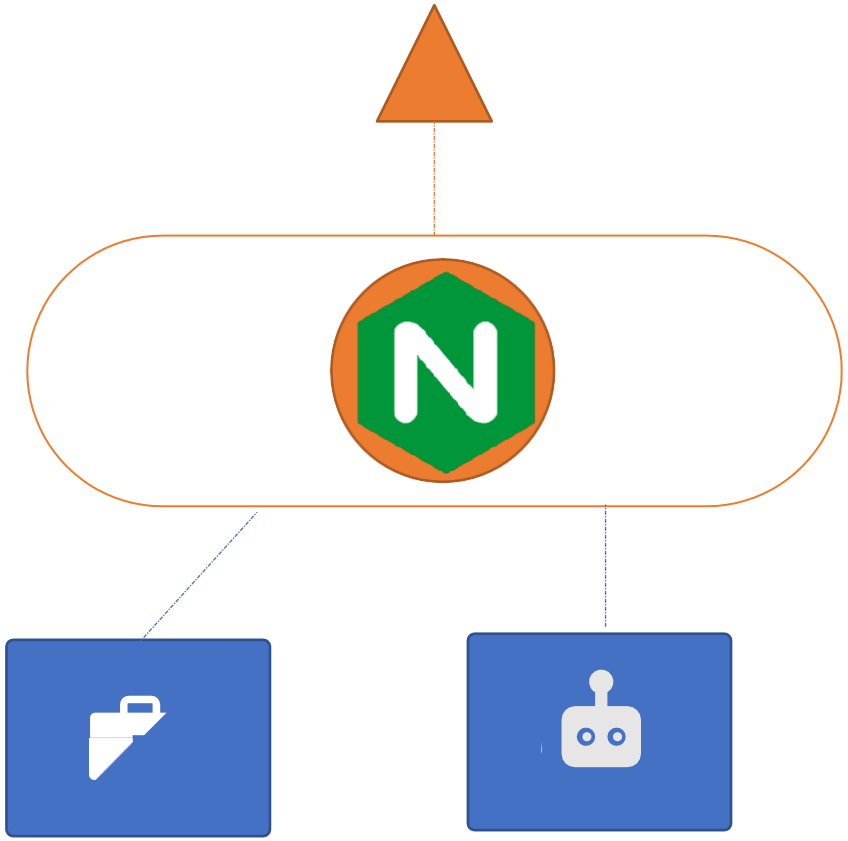
Roles

~~ClusterRol~~

~~RoleBinding~~

```
kind: ConfigMap
apiVersion: v1
metadata:
  name: nginx-configuration
```

INGRESS CONTROLLER



ConfigMap
nginx-configuration

ServiceAccount
nginx-ingress-serviceaccount

Deployment

```yaml
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: nginx-ingress-controller
spec:
  replicas: 1
  selector:
    matchLabels:
      name: nginx-ingress
  template:
    metadata:
      labels:
        name: nginx-ingress
    spec:
      containers:
        - name: nginx-ingress-controller
          image: quay.io/kubernetes-ingress-
                 controller/nginx-ingress-controller:0.21.0
        args:
          - /nginx-ingress-controller
          - --configmap=$(POD_NAMESPACE)/nginx-configuration

        env:
          - name: POD_NAME
            valueFrom:
              fieldRef:
                fieldPath: metadata.name
          - name: POD_NAMESPACE
            valueFrom:
              fieldRef:
                fieldPath: metadata.namespace

        ports:
          - name: http
            containerPort: 80
          - name: https
            containerPort: 443
```

Service

```yaml
apiVersion: v1
kind: Service
metadata:
  name: nginx-ingress
spec:
  type: NodePort
  ports:
  - port: 80
    targetPort: 80
    protocol: TCP
    name: http
  - port: 443
    targetPort: 443
    protocol: TCP
    name: https
  selector:
    name: nginx-ingress
```

ConfigMap

```yaml
kind: ConfigMap
apiVersion: v1
metadata:
  name: nginx-configuration
```

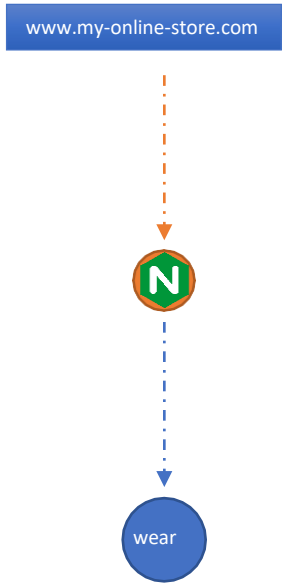
Auth

```yaml
apiVersion: v1
kind: ServiceAccount
metadata:
  name: nginx-ingress-serviceaccount
```

Roles | ClusterRoles | RoleBindings

INGRESS RESOURCE

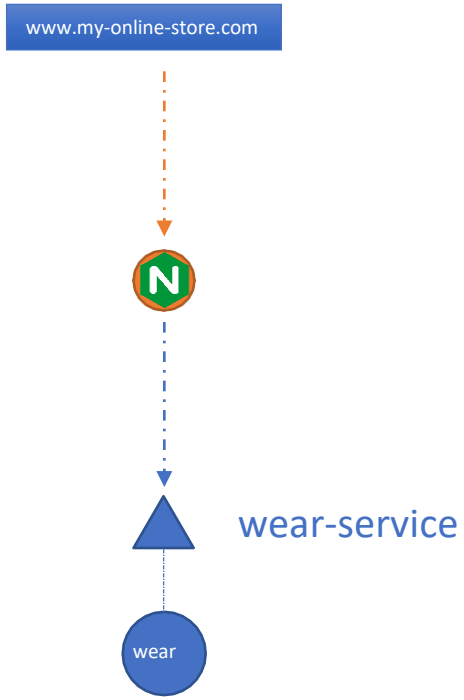
INGRESS RESOURCE



Ingress-wear.yaml

```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: ingress-wear
spec:
```

INGRESS RESOURCE



Ingress-wear.yaml

```yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: ingress-wear
spec:
 backend:
     serviceName: wear-service
     servicePort: 80
```

```
kubectl create -f Ingress-wear.yaml
ingress.extensions/ingress-wear created
```

```
kubectl get ingress
```

```
ingress-wear        *                    80        2s
```

INGRESS RESOURCE - RULES

www.my-online-store.com	www.wear.my-online-store.com	www.watch.my-online-store.com	Everything Else
Rule 1	Rule 2	Rule 3	Rule 4

INGRESS RESOURCE - RULES

DNS Name	Forward IP
www.my-online-store.com	10.123.23.12 (INGRESS SERVICE)
www.wear.my-online-store.com	10.123.23.12
www.watch.my-online.store.com	10.123.23.12
www.my-wear-store.com	10.123.23.12
www.my-watch-store.com	10.123.23.12

www.my-online-sto

ing Else

Rule 1

Rule 4

INGRESS RESOURCE - RULES

www.my-online-store.com

www.wear.my-online-store.com

www.watch.my-online-store.com

Everything Else

http://www.my-online-store.com/wear

http://www.my-online-store.com/watch

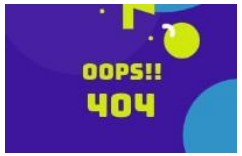
http://www.my-online-store.com/listen

Rule 1

Path /wear

Path /watch

Path /



Rule 2

Rule 3

Rule 4

INGRESS RESOURCE - RULES

www.my-online-store.com

www.wear.my-online-store.com

www.watch.my-online-store.com

Everything Else

tp://www.my-online-store.com/wear

http://www.wear.my-online-store.com/

tp://www.my-online-store.com/watch

http://www.wear.my-online-store.com/returns

tp://www.my-online-store.com/listen

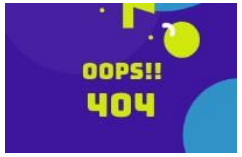
http://www.wear.my-online-store.com/support

Rule 1

Path /wear

Path /watch

Path /



Rule 2

Path /

Path /returns

Path /support

Rule 3

Rule 4

INGRESS RESOURCE - RULES

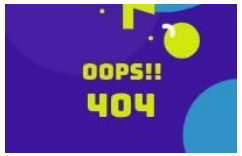
www.my-online-store.com	www.wear.my-online-store.com	www.watch.my-online-store.com	Everything Else
http://www.my-online-store.com/wear	http://www.wear.my-online-store.com/	http://www.watch.my-online-store.com/	
http://www.my-online-store.com/watch	http://www.wear.my-online-store.com/returns	http://www.watch.my-online-store.com/movies	
http://www.my-online-store.com/listen	http://www.wear.my-online-store.com/support	http://www.watch.my-online-store.com/tv	

Rule 1

Path /wear

Path /watch

Path /



Rule 2

Path /

Path /returns

Path /support

Rule 3

Path /

Path /movies

Path /tv

Rule 4

INGRESS RESOURCE - RULES

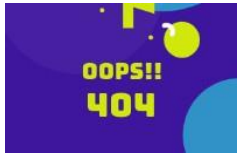
www.my-online-store.com	www.wear.my-online-store.com	www.watch.my-online-store.com	Everything Else
http://www.my-online-store.com/wear	http://www.wear.my-online-store.com/	http://www.watch.my-online-store.com/	http://www.listen.my-online-store.com/
http://www.my-online-store.com/watch	http://www.wear.my-online-store.com/returns	http://www.watch.my-online-store.com/movies	http://www.eat.my-online-store.com/
http://www.my-online-store.com/listen	http://www.wear.my-online-store.com/support	http://www.watch.my-online-store.com/tv	http://www.drink.my-online-store.com/tv

Rule 1

Path /wear

Path /watch

Path /



Rule 2

Path /

Path /returns

Path /support

Rule 3

Path /

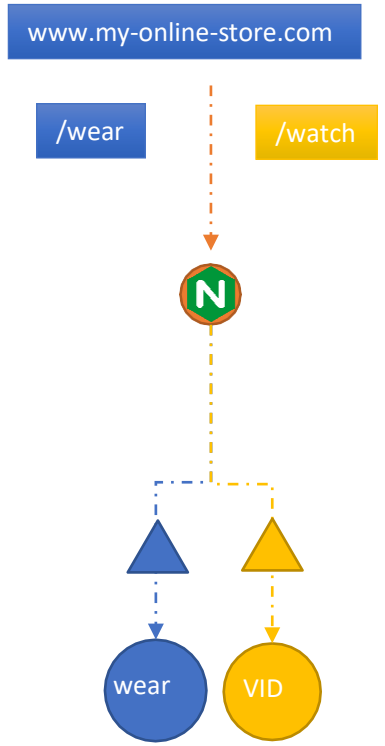
Path /movies

Path /tv

Rule 4

Path /

INGRESS RESOURCE



Ingress-wear.yaml

```yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: ingress-wear
spec:
  backend:
    serviceName: wear-service
    servicePort: 80
```

Ingress-wear-watch.yaml

```yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: ingress-wear-watch
spec:
  rules:
  - http:
      paths:
      - path: /wear



      - path: /watch
        backend:
          serviceName: watch-service
          servicePort: 80
```

INGRESS RESOURCE

```
kubectl describe ingress ingress-wear-watch
```

```
Name:             ingress-wear-watch
Namespace:        default
Address:
Default backend:  default-http-backend:80 (<none>)
Rules:
  Host  Path  Backends
  ----  ----  --------
  *
        /wear   wear-service:80 (<none>)
        /watch  watch-service:80 (<none>)
Annotations:
Events:
  Type    Reason  Age   From                      Message
  ----    ------  ----  ----                      -------
  Normal  CREATE  14s   nginx-ingress-controller  Ingress default/ingress-wear-watch
```

INGRESS RESOURCE

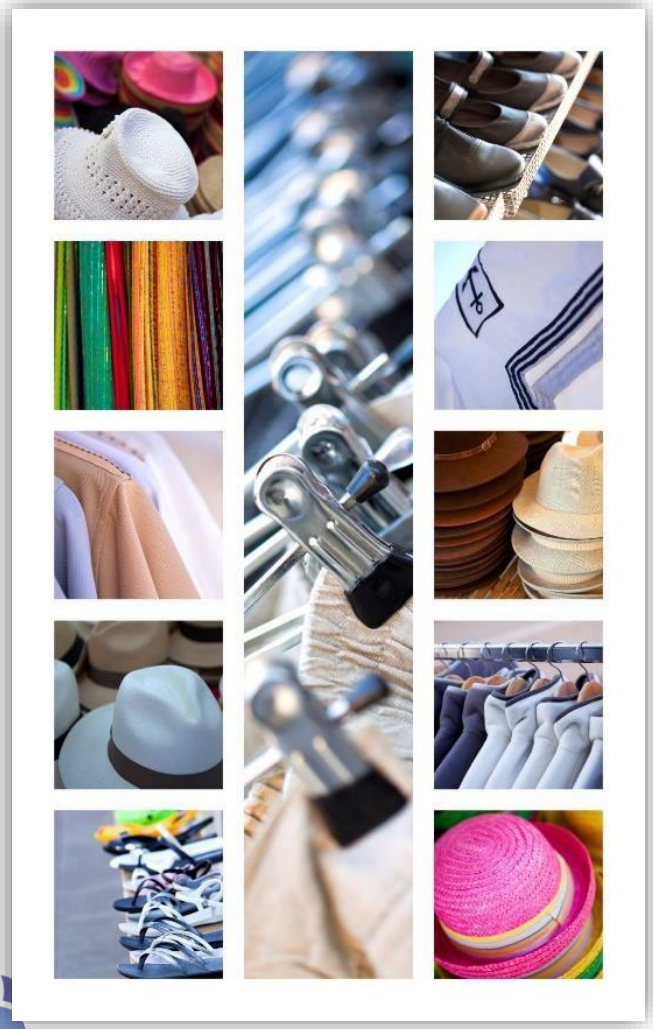
www.my-online-store.com/eat

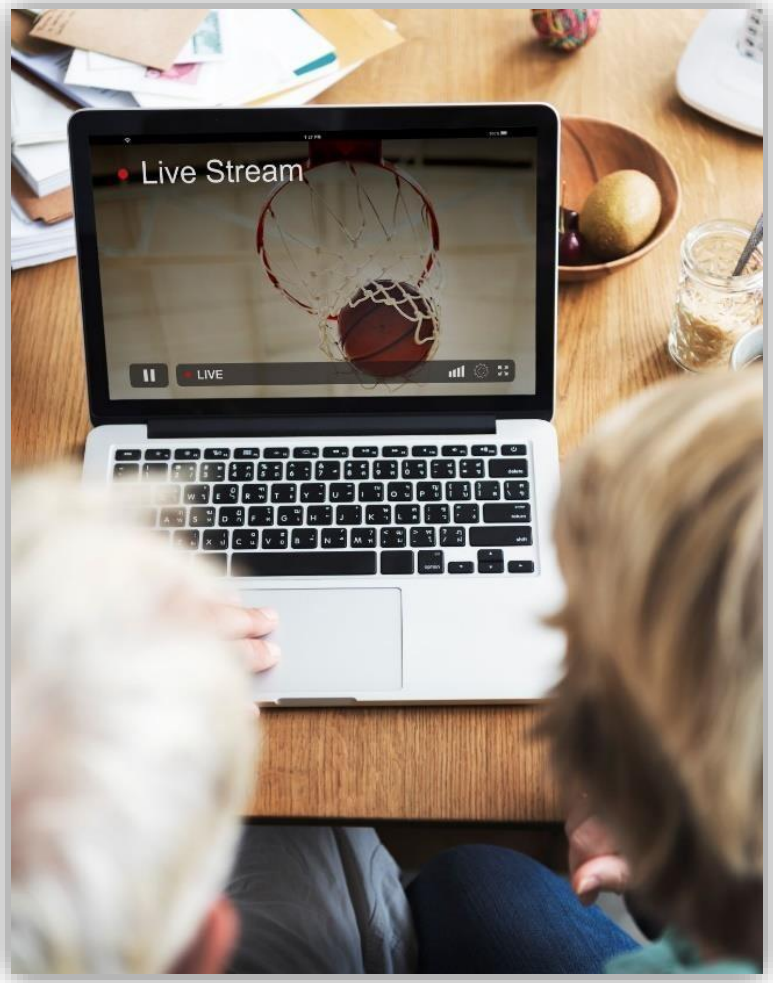
www.my-online-store.com/liste

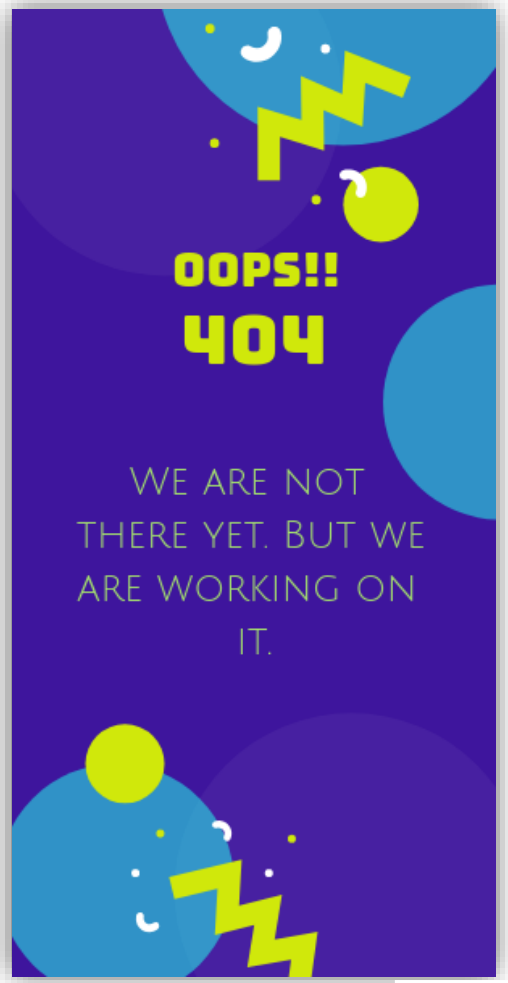
www.my-online-store.com/wear

www.my-online-store.com

www.my-online-store.com/watch







INGRESS RESOURCE

Ingress-wear-watch.yaml

```yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: ingress-wear-watch
spec:

  rules:
  - host: wear.my-online-store.com
    http:
      paths:
      - backend:
          serviceName: wear-service
          servicePort: 80
  - host: watch.my-online-store.com
    http:
      paths:
      - backend:
          serviceName: watch-service
          servicePort: 80
```

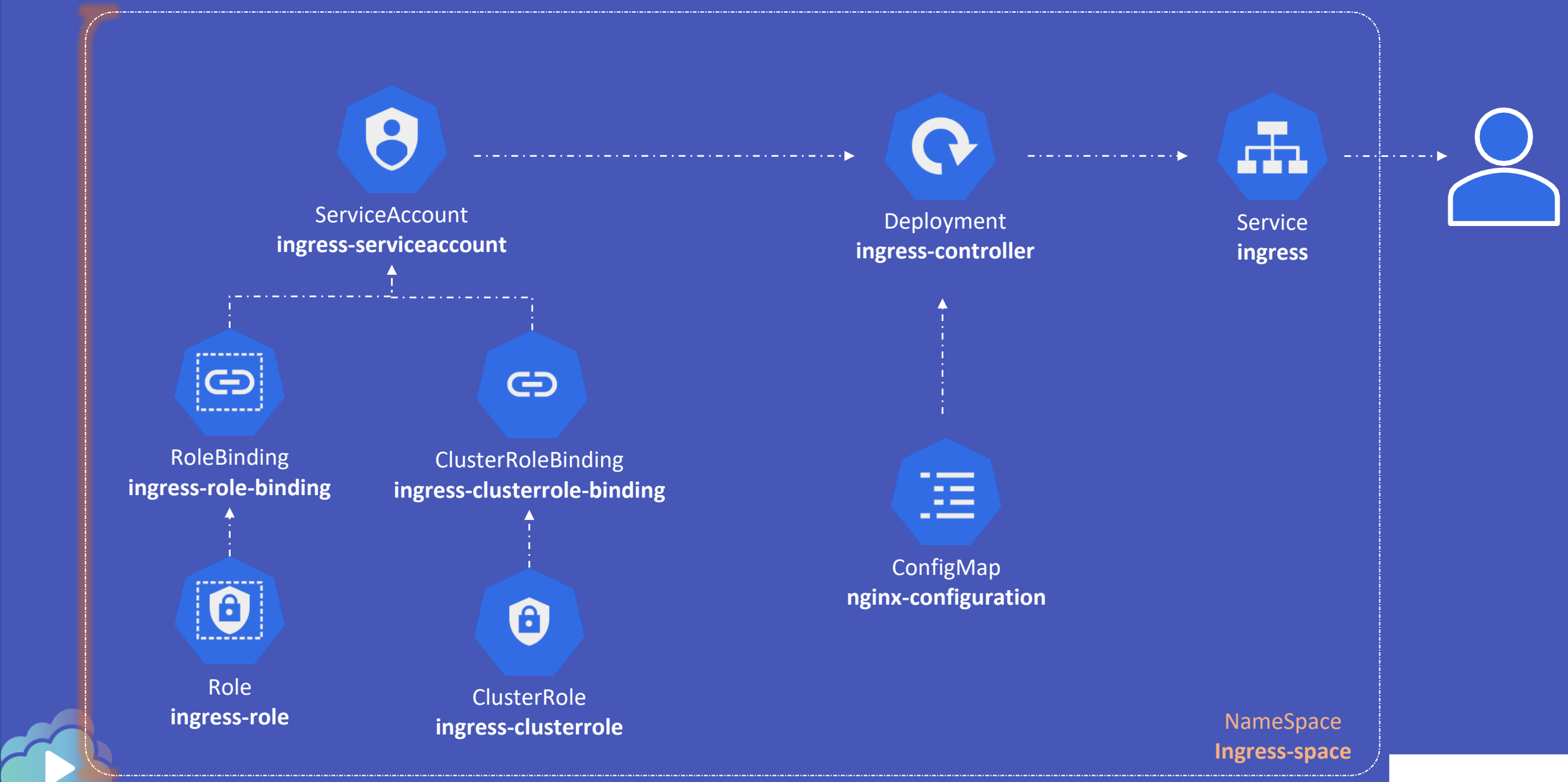
INGRESS RESOURCE

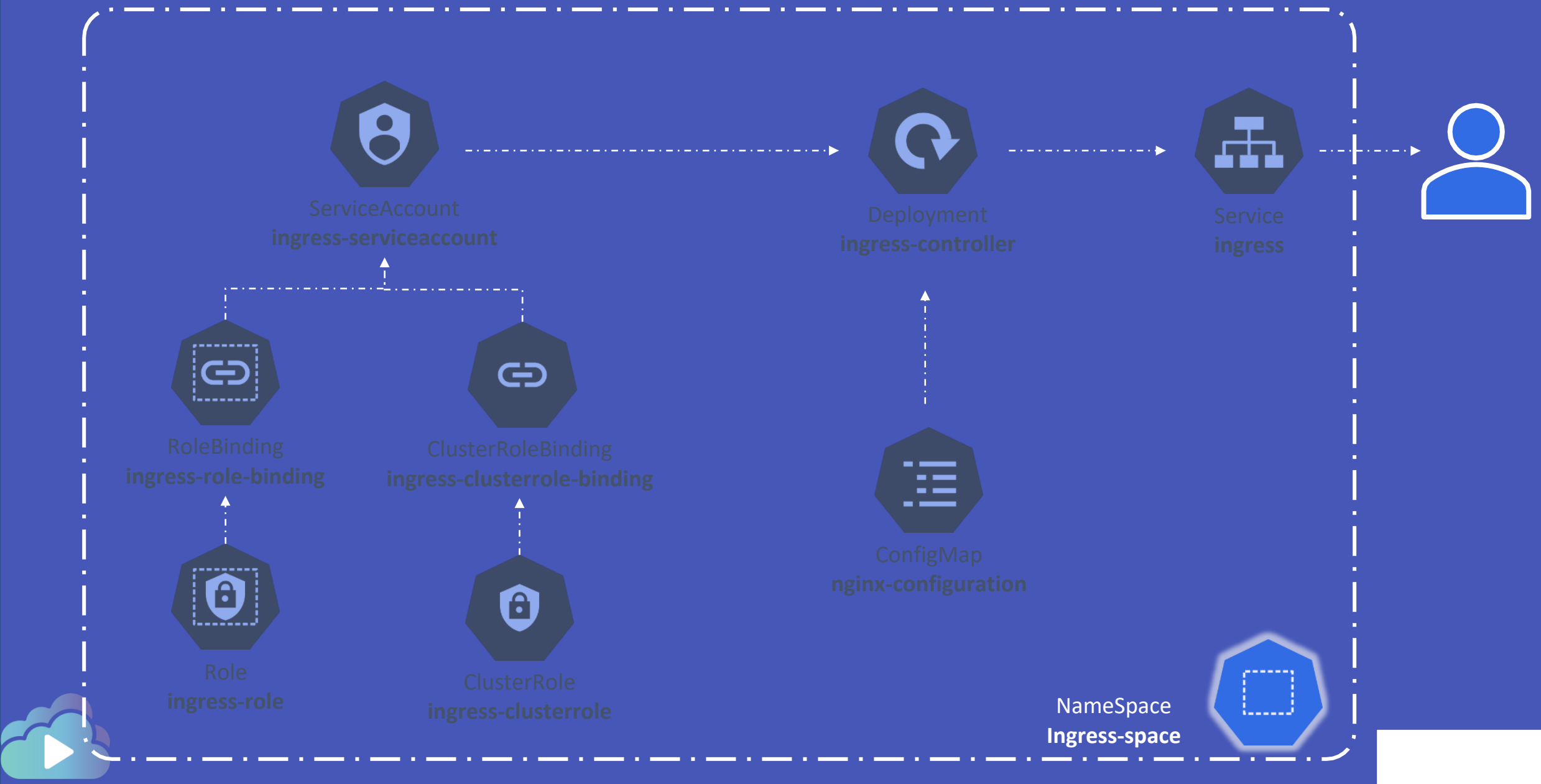
Ingress-wear-watch.yaml

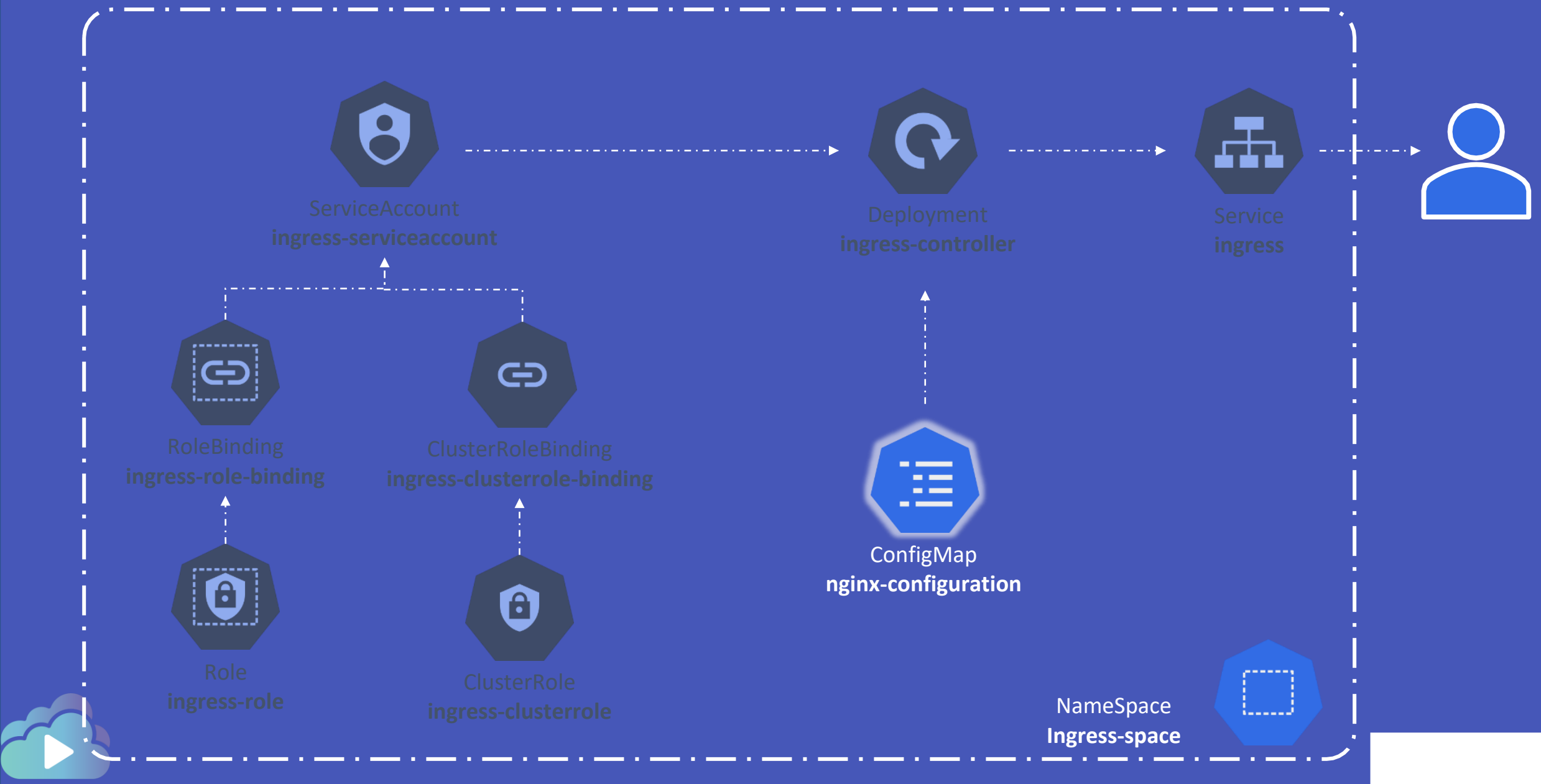
```yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: ingress-wear-watch
spec:
  rules:
  - http:
      paths:
      - path: /wear
        backend:
          serviceName: wear-service
          servicePort: 80
      - path: /watch
        backend:
          serviceName: watch-service
          servicePort: 80
```

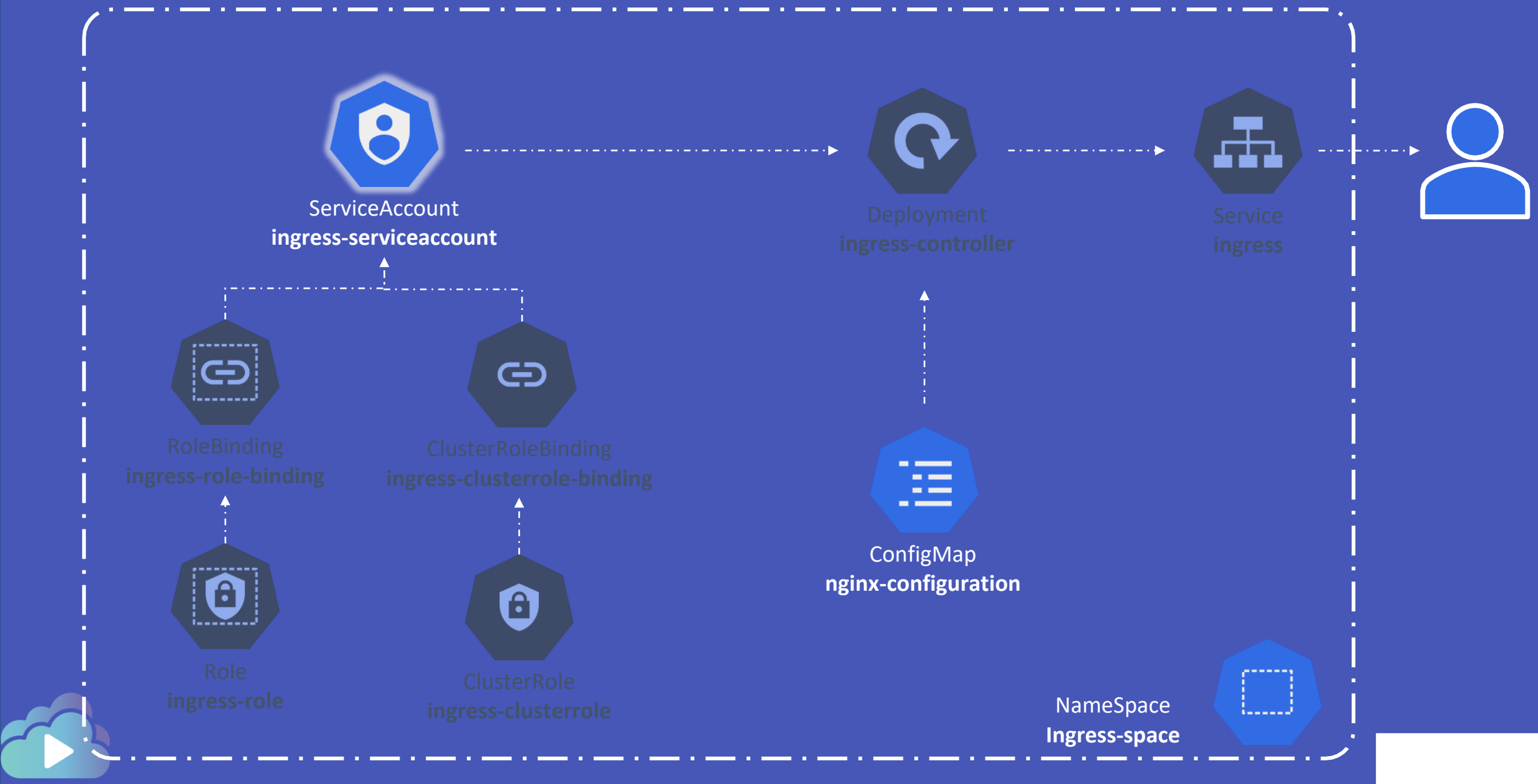
Ingress-wear-watch.yaml

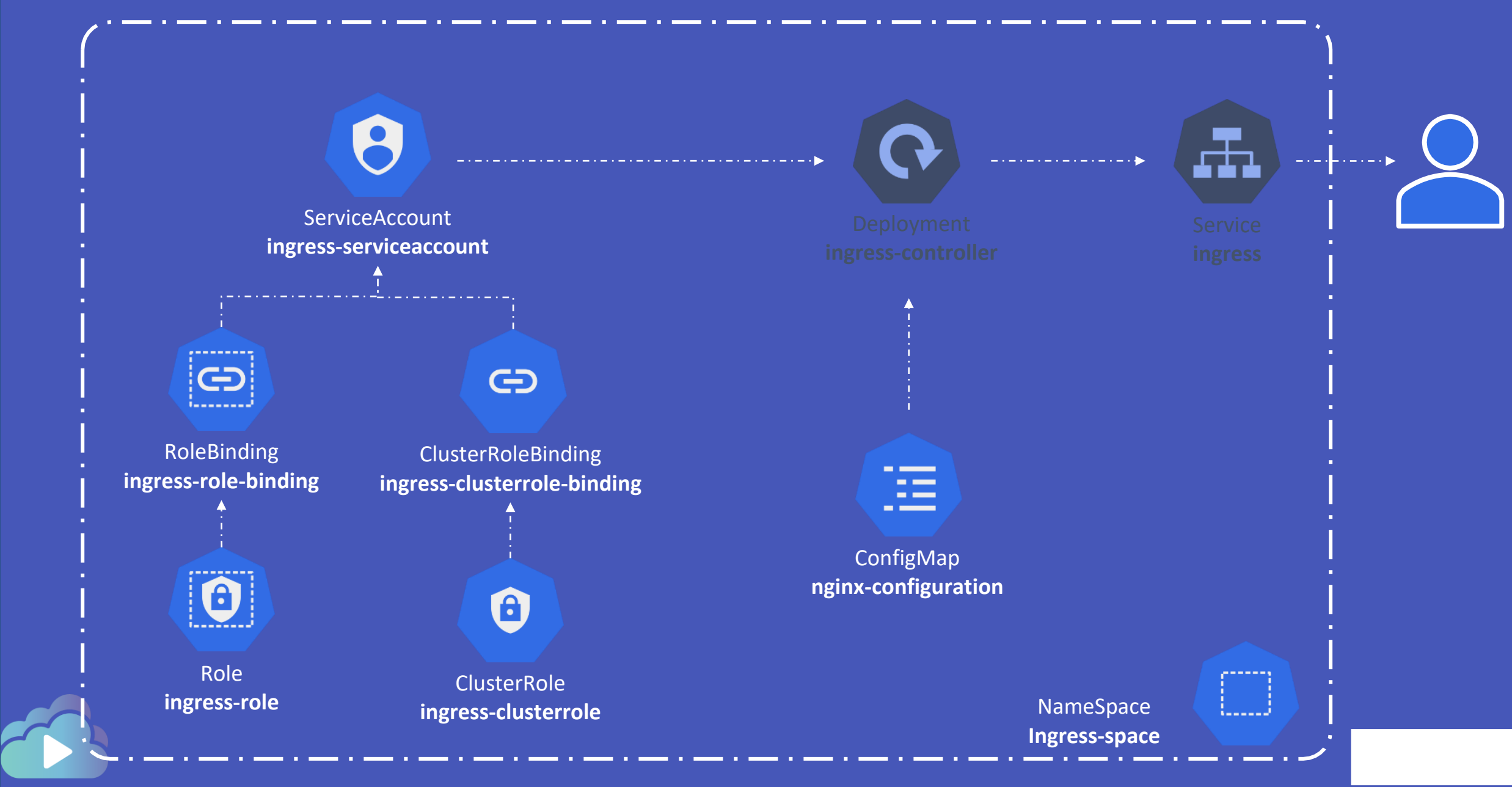
```yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: ingress-wear-watch
spec:

  rules:
  - host: wear.my-online-store.com
    http:
      paths:
      - backend:
          serviceName: wear-service
          servicePort: 80
  - host: watch.my-online-store.com
    http:
      paths:
      - backend:
          serviceName: watch-service
          servicePort: 80
```


ServiceAccount
ingress-serviceaccount
Deployment
ingress-controller
Service
ingress
RoleBinding
ingress-role-binding
ClusterRoleBinding
ingress-clusterrole-binding
ConfigMap
nginx-configuration
Role
ingress-role
ClusterRole
ingress-clusterrole
NameSpace
Ingress-space


ServiceAccount
ingress-serviceaccount
Deployment
ingress-controller
Service
ingress
RoleBinding
ingress-role-binding
ClusterRoleBinding
ingress-clusterrole-binding
ConfigMap
nginx-configuration
Role
ingress-role
ClusterRole
ingress-clusterrole
NameSpace
Ingress-space


ServiceAccount
ingress-serviceaccount
Deployment
ingress-controller
Service
ingress
RoleBinding
ingress-role-binding
ClusterRoleBinding
ingress-clusterrole-binding
ConfigMap
nginx-configuration
Role
ingress-role
ClusterRole
ingress-clusterrole
NameSpace
Ingress-space


ServiceAccount
ingress-serviceaccount
Deployment
ingress-controller
Service
ingress
RoleBinding
ingress-role-binding
ClusterRoleBinding
ingress-clusterrole-binding
ConfigMap
nginx-configuration
Role
ingress-role
ClusterRole
ingress-clusterrole
NameSpace
Ingress-space


ServiceAccount
ingress-serviceaccount
Deployment
ingress-controller
Service
ingress
RoleBinding
ingress-role-binding
ClusterRoleBinding
ingress-clusterrole-binding
ConfigMap
nginx-configuration
Role
ingress-role
ClusterRole
ingress-clusterrole
NameSpace
Ingress-space

ServiceAccount
ingress-serviceaccount
Deployment
ingress-controller
Service
ingress
RoleBinding
ingress-role-binding
ClusterRoleBinding
ingress-clusterrole-binding
ConfigMap
nginx-configuration
Role
ingress-role
ClusterRole
ingress-clusterrole
NameSpace
Ingress-space

ServiceAccount
ingress-serviceaccount
Deployment
ingress-controller
Service
ingress
RoleBinding
ingress-role-binding
ClusterRoleBinding
ingress-clusterrole-binding
ConfigMap
nginx-configuration
Role
ingress-role
ClusterRole
ingress-clusterrole
NameSpace
Ingress-space