

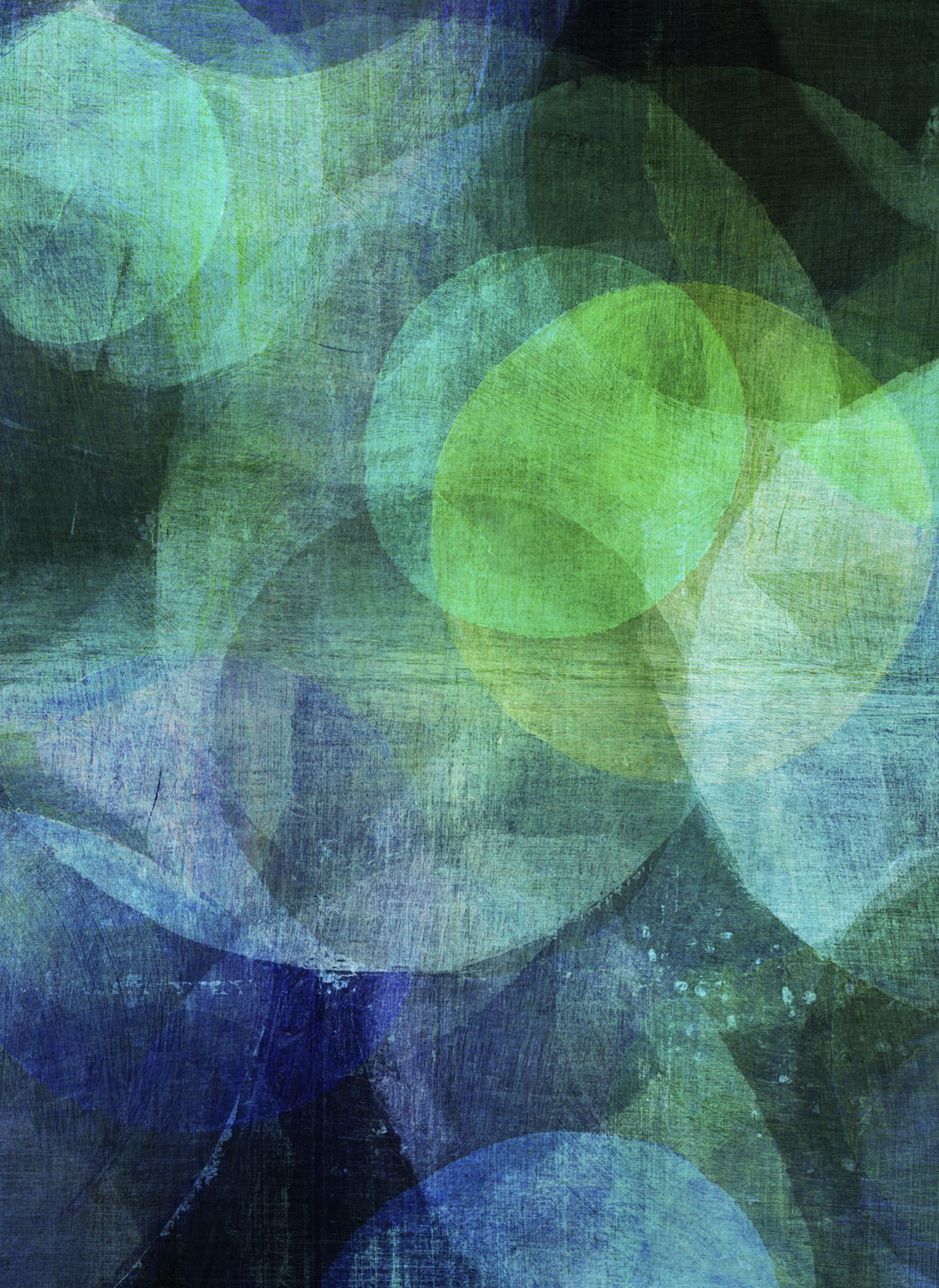
INTRODUCTION TO KUBERNETES



INTRODUCTION TO KUBERNETES

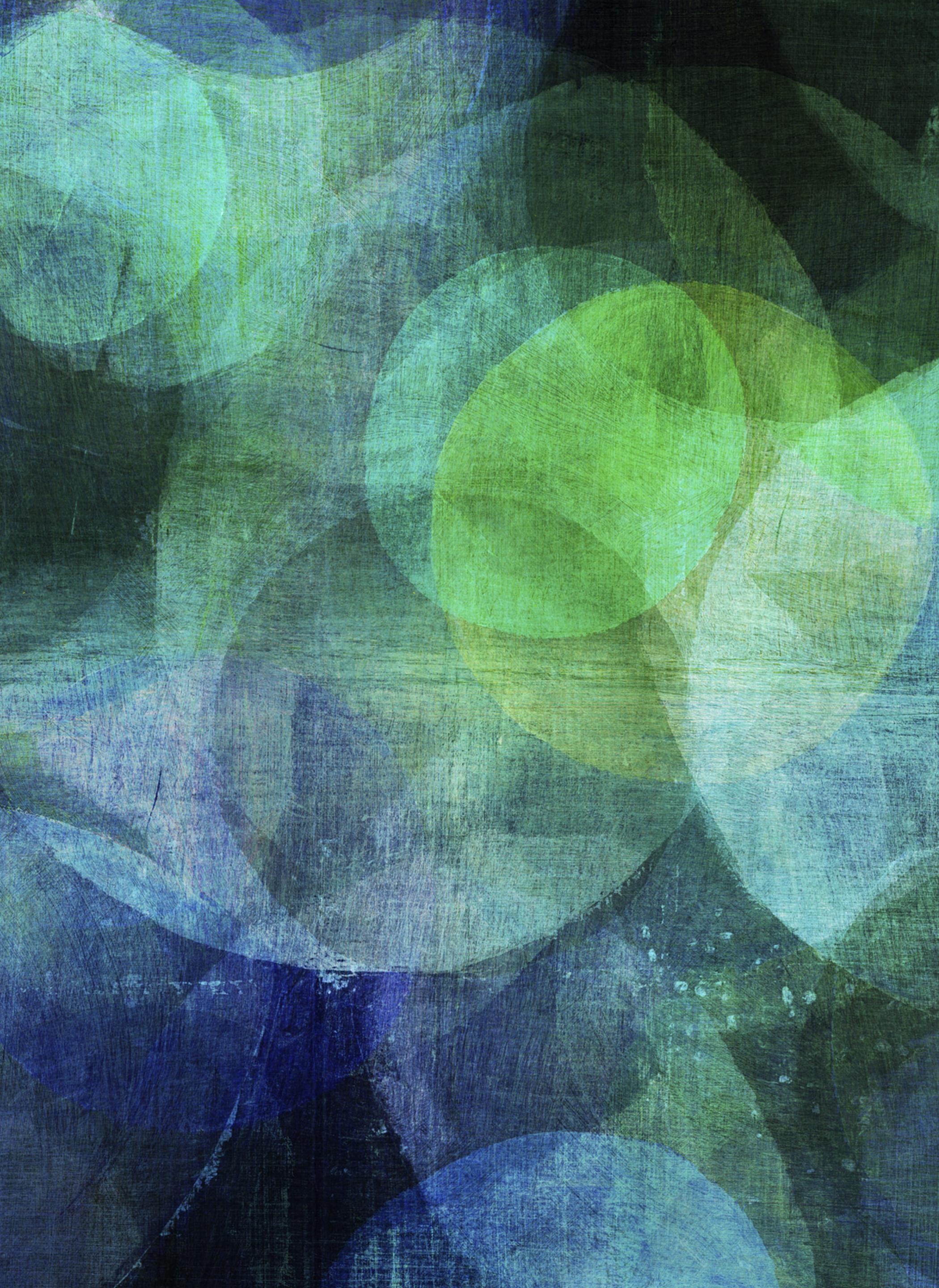
Kubernetes Configuration Files

AGENDA



AGENDA

- Configuration Format



AGENDA

- Configuration Format
- Deployment Config



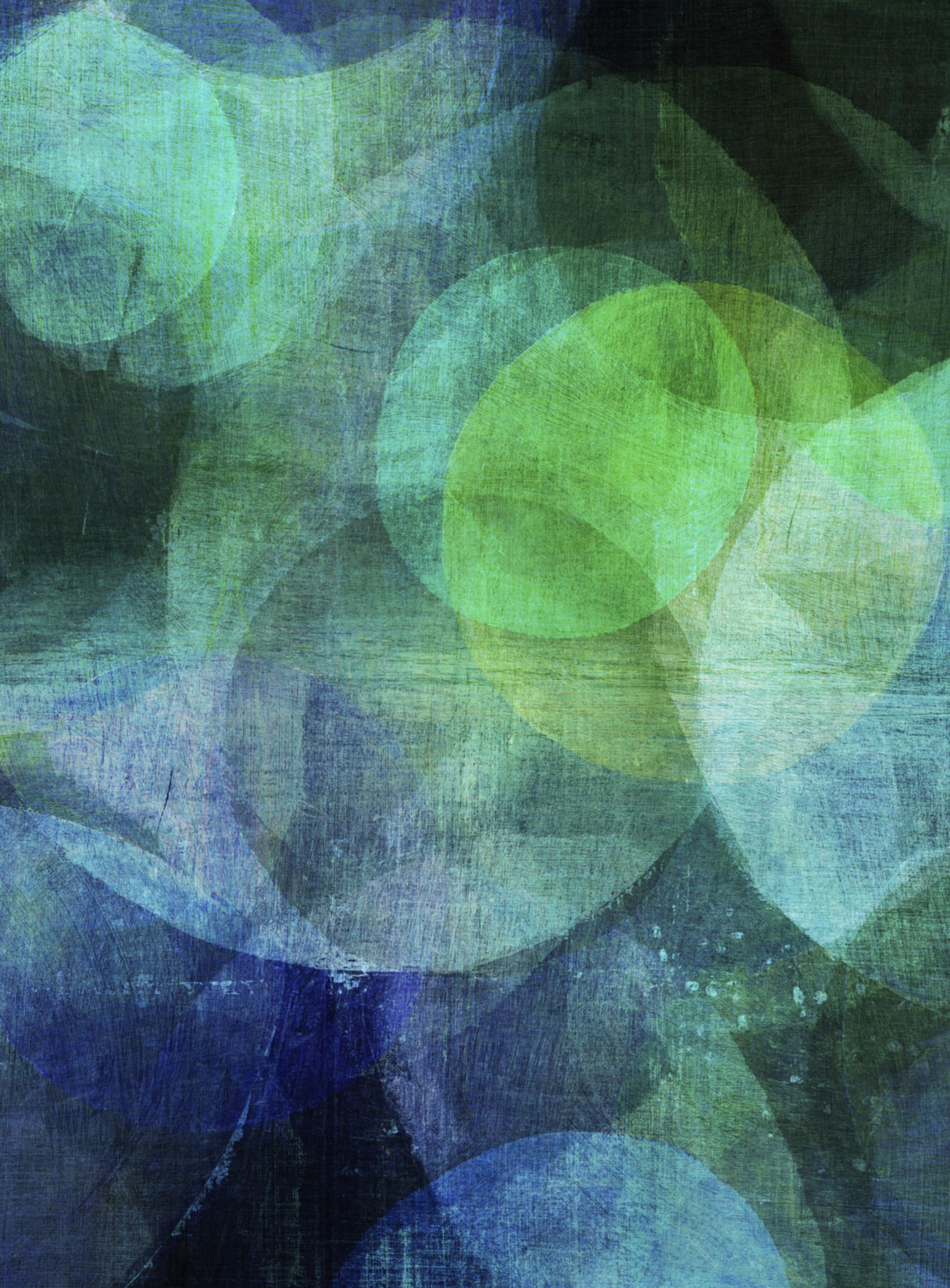
AGENDA

- Configuration Format
- Deployment Config
 - Service Config



AGENDA

- Configuration Format
- Deployment Config
 - Service Config
 - ConfigMap Config



CONFIGURATION FORMAT

YAML CONFIGURATION FILES

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
```

YAML: YAML Ain't Markup Language™

What It Is:

YAML is a human-friendly data serialization language for all programming languages.

Syntax:

Strict on **indentations**

Kubernetes Configuration File

- concept of infrastructure as a code
- Kept it with your Code

3 PART OF KUBERNETES CONFIG FILE

metadata :

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  namespace: my-namespace
  labels:
    app-type: api
spec:
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
```

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

3 PART OF KUBERNETES CONFIG FILE

spec:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  namespace: my-namespace
  labels:
    app-type: api
spec:
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
```

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

3 PART OF KUBERNETES CONFIG FILE

status:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  namespace: my-namespace
  labels:
    app-type: api
spec:
  selector:...
  template:...
status:
  conditions:
  - lastProbeTime: null
    lastTransitionTime: "2022-02-08T15:47:29Z"
    status: "True"
    type: Initialized
  - lastProbeTime: null
    lastTransitionTime: "2022-02-08T15:47:29Z"
```

3 PART OF KUBERNETES CONFIG FILE

status:

```
$ kubectl get [objects] [name] -o yaml
```

```
apiVersion: apps/v1
kind: Deployment
metadata: ...
  name: nginx-deployment
  namespace: my-namespace
  labels:
    app-type: api
spec: ...
  selector:...
  template:...
status: ...
  conditions:
  - lastProbeTime: null
    lastTransitionTime: "2022-02-08T15:47:29Z"
    status: "True"
    type: Initialized
  - lastProbeTime: null
    lastTransitionTime: "2022-02-08T15:47:29Z"
```

DEPLOYMENT CONFIGURATION

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.14.2
```

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
  template: [REDACTED]
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
```

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
```

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
```

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
template:
  metadata:
    labels:
      app: nginx
  spec:
    containers:
    - name: nginx
      image: nginx:1.14.2
```



Deployment manage ReplicaSet

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
```



Deployment manage ReplicaSet



ReplicaSet manage Pod

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
```



Deployment manage ReplicaSet



ReplicaSet manage Pod



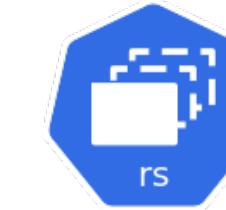
Pod is contain the containers

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
template:
  metadata:
    labels:
      app: nginx
  spec:
    containers:
    - name: nginx
      image: nginx:1.14.2
```



Deployment manage ReplicaSet



ReplicaSet manage Pod



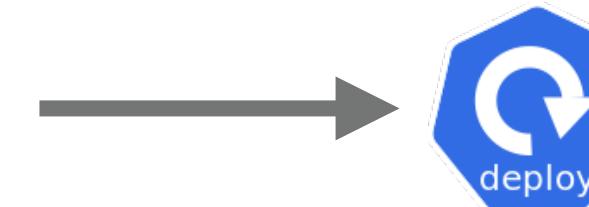
Pod is contain the containers



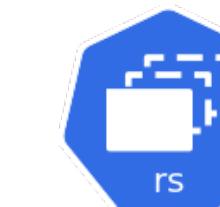
Container are executable

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
template:
  metadata:
    labels:
      app: nginx
  spec:
    containers:
    - name: nginx
      image: nginx:1.14.2
```



Deployment manage ReplicaSet



ReplicaSet manage Pod



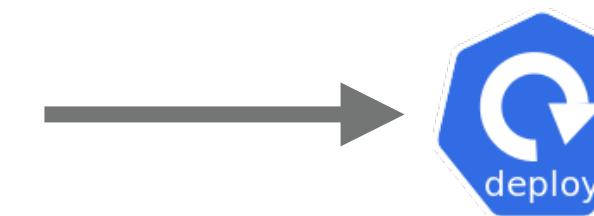
Pod is contain the containers



Container are executable

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
template:
  metadata:
    labels:
      app: nginx
  spec:
    containers:
    - name: nginx
      image: nginx:1.14.2
```



Deployment manage ReplicaSet



ReplicaSet manage Pod



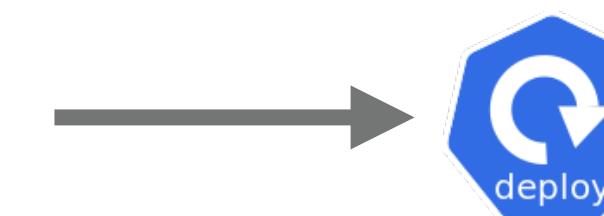
Pod is contain the containers



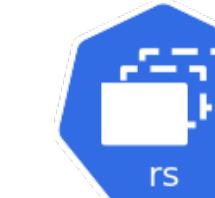
Container are executable

DEPLOYMENT - THE BLUEPRINT OF THE PODS (TEMPLATE)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
template:
  metadata:
    labels:
      app: nginx
  spec:
    containers:
    - name: nginx
      image: nginx:1.14.2
```



Deployment manage ReplicaSet



ReplicaSet manage Pod



Pod is contain the containers



Container are executable

SERVICE CONFIGURATION

SERVICE - LABEL AND SELECTOR

Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-web-deployment
  labels:
    app: my-web-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-web
  template:
    metadata:
      labels:
        app: my-web
    spec:
      containers:
        - image: nginx
          name: my-web-on-nginx
          ports:
            - containerPort: 80
```

Service

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

SERVICE - LABEL AND SELECTOR

Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-web-deployment
  labels:
    app: my-web-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-web
  template:
    metadata:
      labels:
        app: my-web
    spec:
      containers:
        - image: nginx
          name: my-web-on-nginx
          ports:
            - containerPort: 80
```

Service

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

SERVICE - LABEL AND SELECTOR

Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-web-deployment
  labels:
    app: my-web-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-web
  template:
    metadata:
      labels:
        app: my-web
  spec:
    containers:
      - image: nginx
        name: my-web-on-nginx
        ports:
          - containerPort: 80
```

Service

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

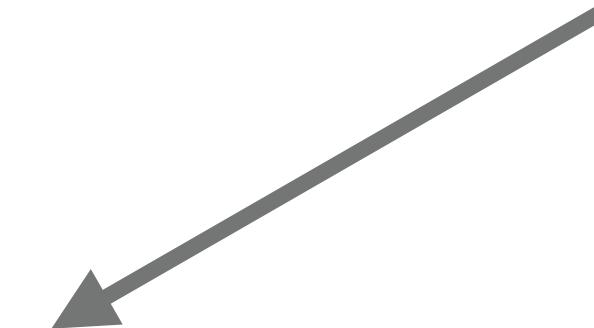
SERVICE - LABEL AND SELECTOR

Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-web-deployment
  labels:
    app: my-web-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-web
  template:
    metadata:
      labels:
        app: my-web
    spec:
      containers:
        - image: nginx
          name: my-web-on-nginx
        ports:
          - containerPort: 80
```

Service

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```



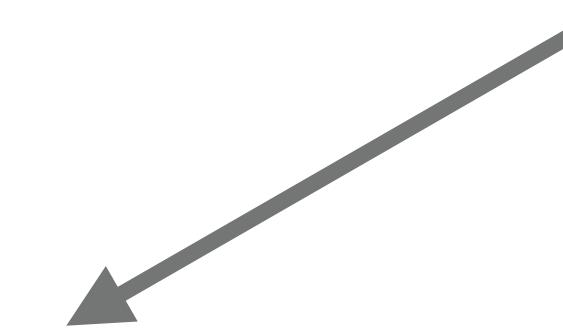
SERVICE - LABEL AND SELECTOR

Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-web-deployment
  labels:
    app: my-web-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-web
  template:
    metadata:
      labels:
        app: my-web
    spec:
      containers:
        - image: nginx
          name: my-web-on-nginx
        ports:
          - containerPort: 80
```

Service

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```



SERVICE - PORT

Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-web-deployment
  labels:
    app: my-web-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-web
  template:
    metadata:
      labels:
        app: my-web
    spec:
      containers:
        - image: nginx
          name: my-web-on-nginx
          ports:
            - containerPort: 80
```

Service

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

SERVICE - PORT

Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-web-deployment
  labels:
    app: my-web-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-web
  template:
    metadata:
      labels:
        app: my-web
    spec:
      containers:
        - image: nginx
          name: my-web-on-nginx
          ports:
            - containerPort: 80
```

Service

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

SERVICE - PORT

Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-web-deployment
  labels:
    app: my-web-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-web
  template:
    metadata:
      labels:
        app: my-web
    spec:
      containers:
        - image: nginx
          name: my-web-on-nginx
          ports:
            - containerPort: 80
```

Service

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

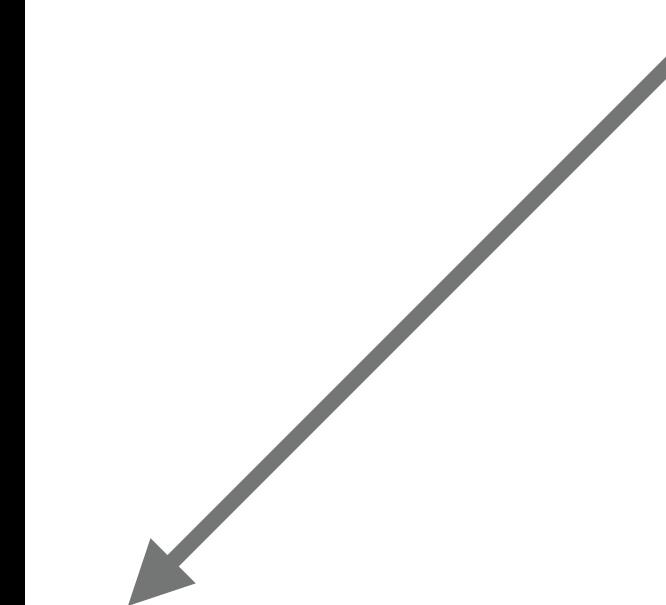
SERVICE - PORT

Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-web-deployment
  labels:
    app: my-web-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-web
  template:
    metadata:
      labels:
        app: my-web
    spec:
      containers:
        - image: nginx
          name: my-web-on-nginx
          ports:
            - containerPort: 80
```

Service

```
apiVersion: v1
kind: Service
metadata:
  name: my-web-service
spec:
  selector:
    app: my-web
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```



THANK YOU