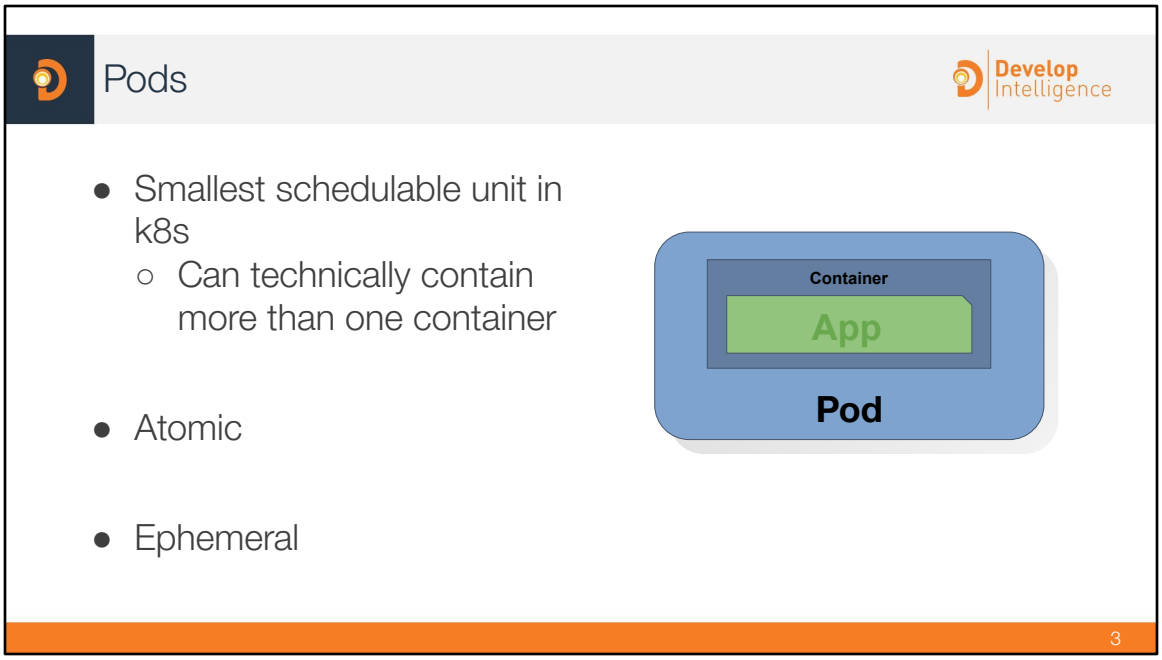
k8s: Basic Objects

Develop Intelligence

Module Outline

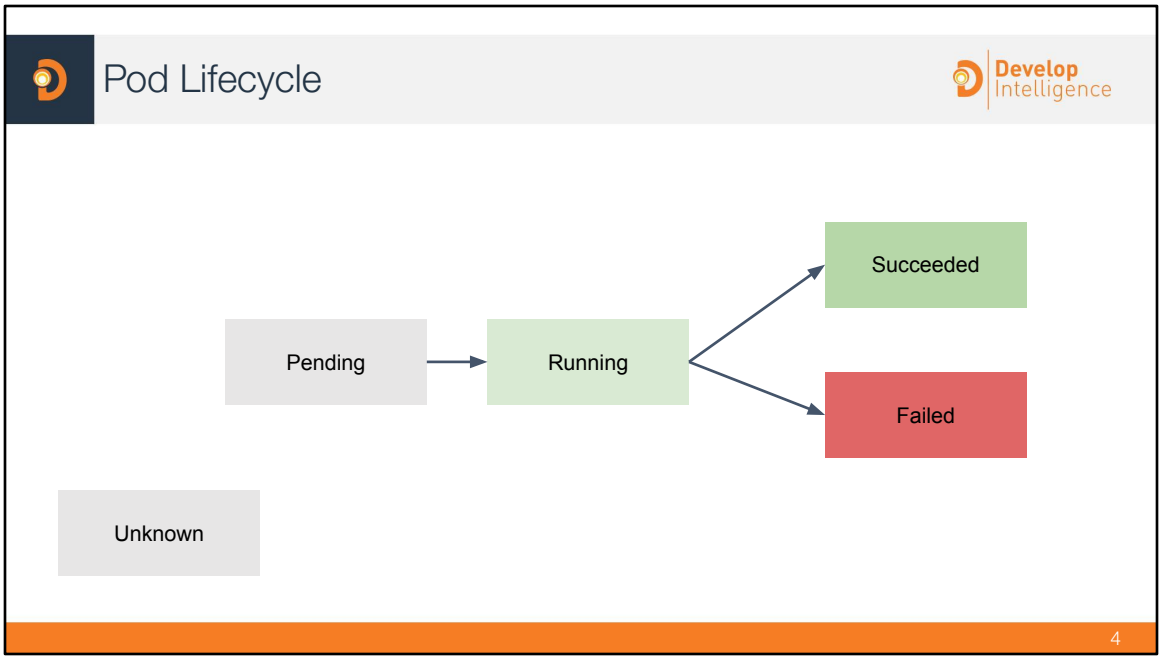
Develop Intelligence

- Pods
- Object Manifests
- Deployments
- Services
- Labels and Selectors
- Lab
- Review



Pods are atomic and mostly immutable. If you want to deploy a new version of your application, you would delete the existing pods and make new ones. You do not baby sit or treat Pods as special, if the application crashes or something goes wrong, you simply delete the pod and make a new one. They are disposable and Pods running the same application should be fungible units.

Pods also allow us to specify things such as RAM/CPU limits and application health checks.



From the docs:

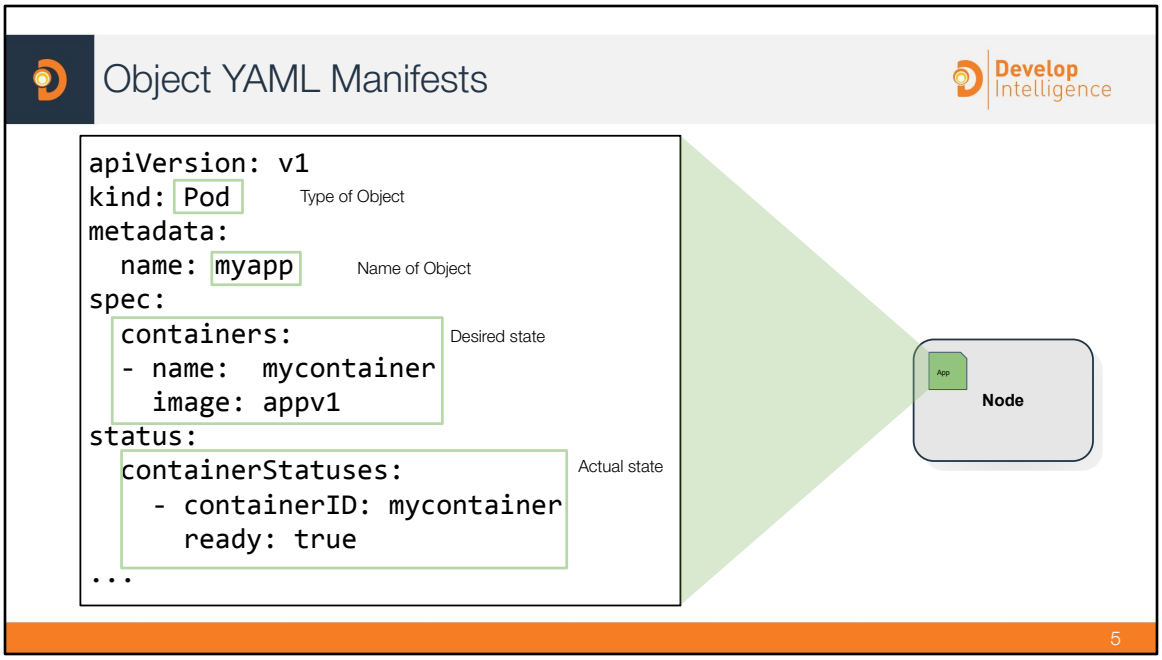
Pending The Pod has been accepted by the Kubernetes system, but one or more of the Container images has not been created. This includes time before being scheduled as well as time spent downloading images over the network, which could take a while.

Running The Pod has been bound to a node, and all of the Containers have been created. At least one Container is still running, or is in the process of starting or restarting.

Succeeded All Containers in the Pod have terminated in success, and will not be restarted.

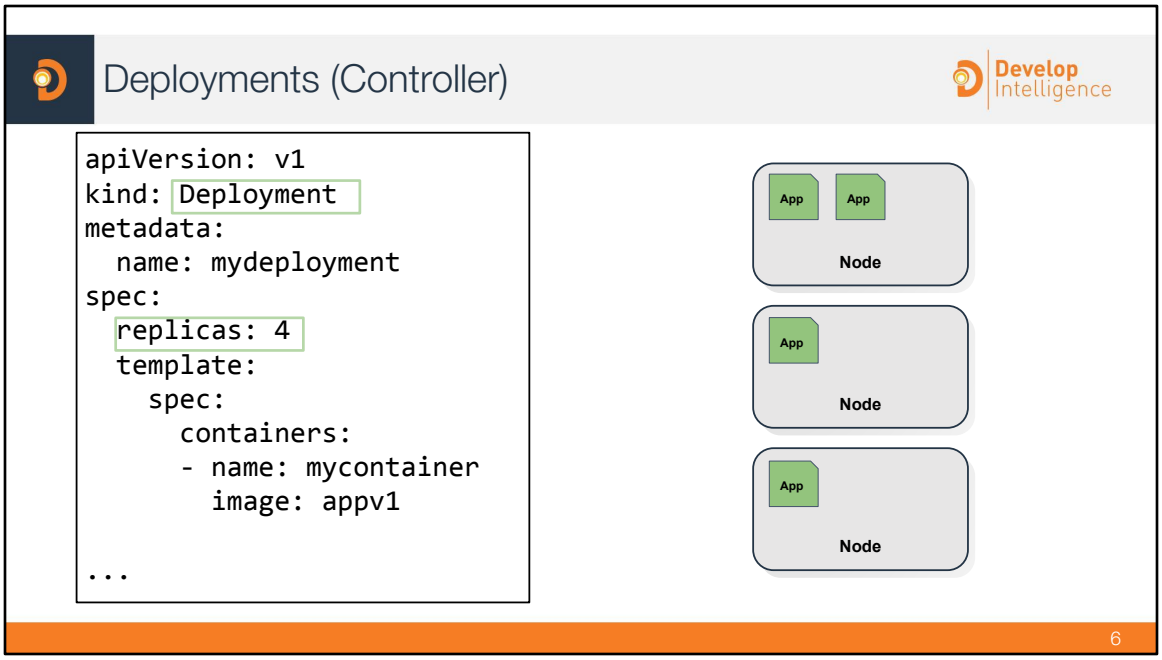
Failed All Containers in the Pod have terminated, and at least one Container has terminated in failure. That is, the Container either exited with non-zero status or was terminated by the system.

Unknown For some reason the state of the Pod could not be obtained, typically due to an error in communicating with the host of the Pod.



All Kubernetes resources can be described with YAML and follow the shown format. Each kubernetes object has its own set of attributes that can be specified. Information for what can be set on an object is defined in the docs or discovered with the "kubectl explain <type>" command.

https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.14/#pod-v1-core



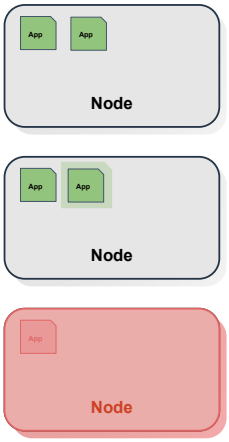
Remember, Pods are how we must run our application. We could define a single Pod with yaml but usually you allow other objects to make the pods for you. Deployments can handle the lifecycle of our application for us by creating some number of identical pods. Scaling, updates, and autohealing are all managed by the Deployment object. Deployments fall under the "Controllers" category of objects, this group is responsible for the creation of pods. Some other Controllers we will discuss later are DaemonSets, StatefulSets and Jobs.

Deployments - Autohealing

Develop Intelligence

```
apiVersion: v1
kind: Deployment
metadata:
  name: mydeployment
spec:
  replicas: 4
  template:
    spec:
      containers:
      - name: mycontainer
        image: appv1

...
```

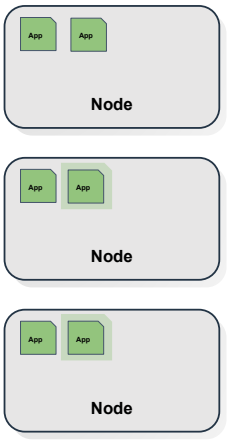


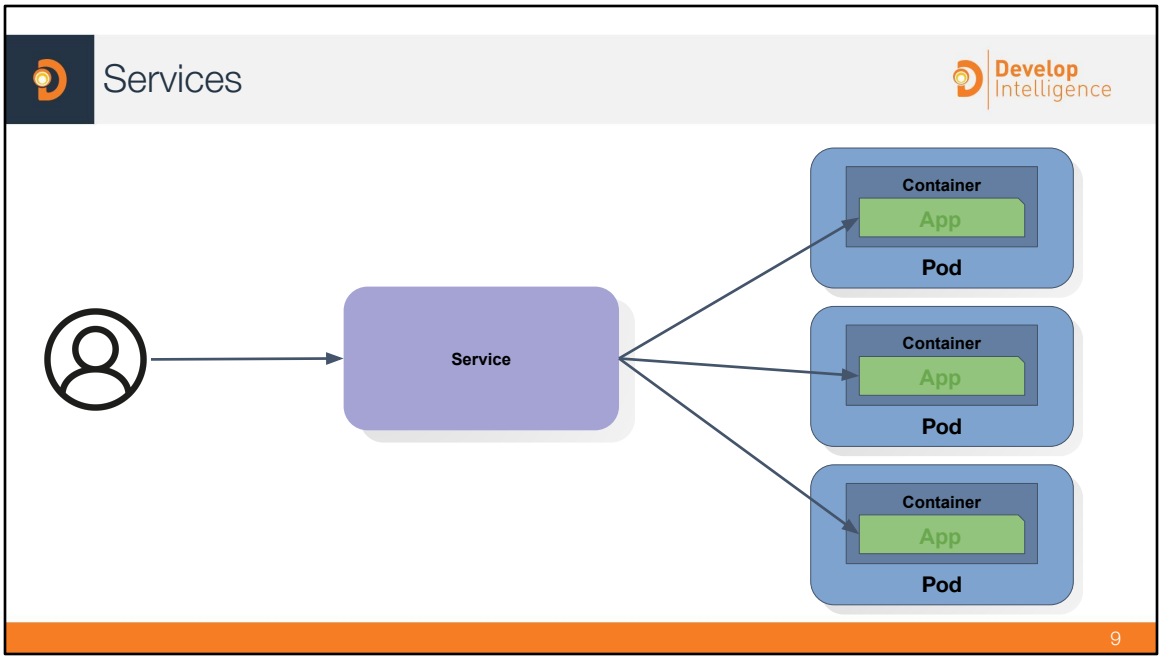
Deployments - Scaling

Develop Intelligence

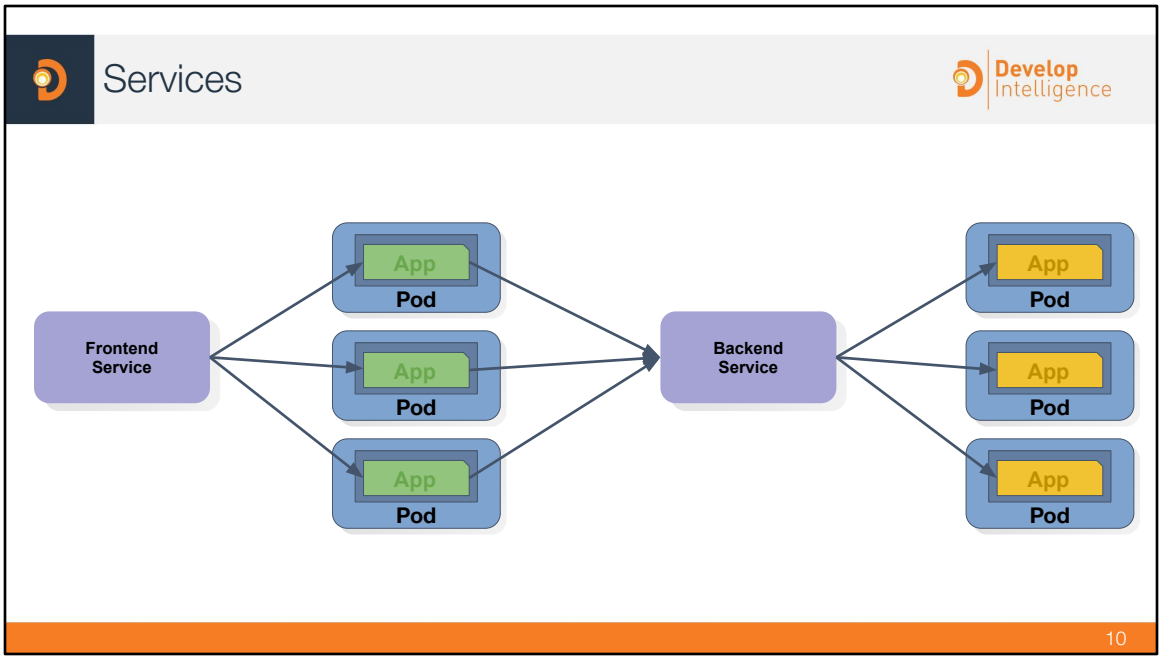
```
apiVersion: v1
kind: Deployment
metadata:
  name: mydeployment
spec:
  replicas: 6
  template:
    spec:
      containers:
      - name: mycontainer
        image: appv1

...
```

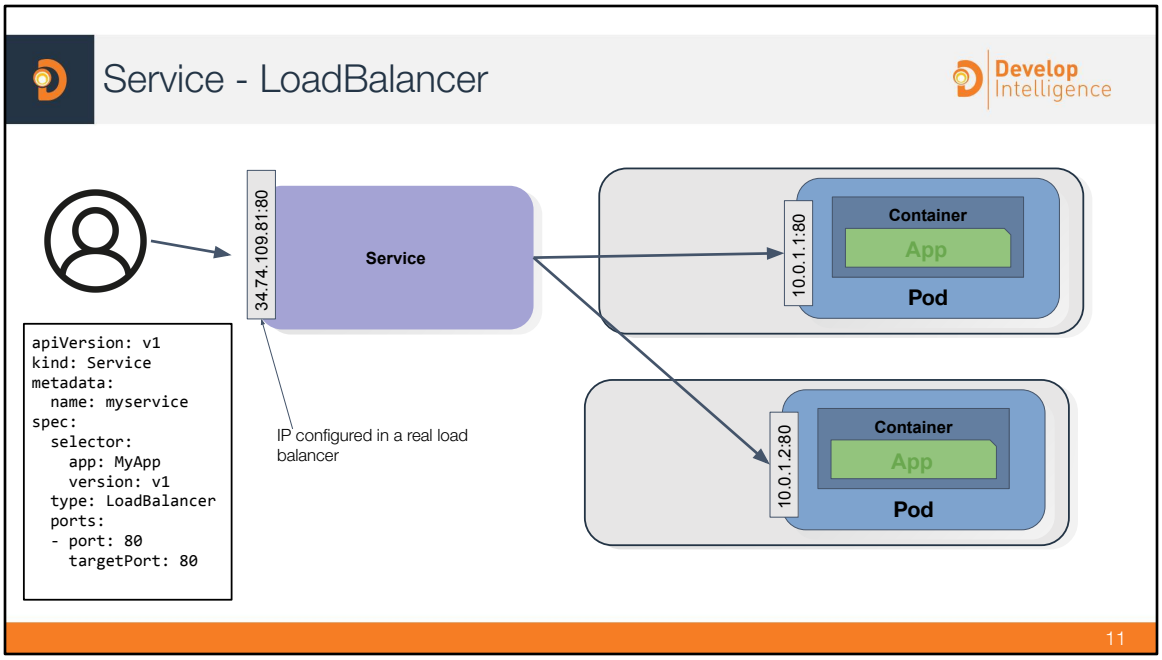




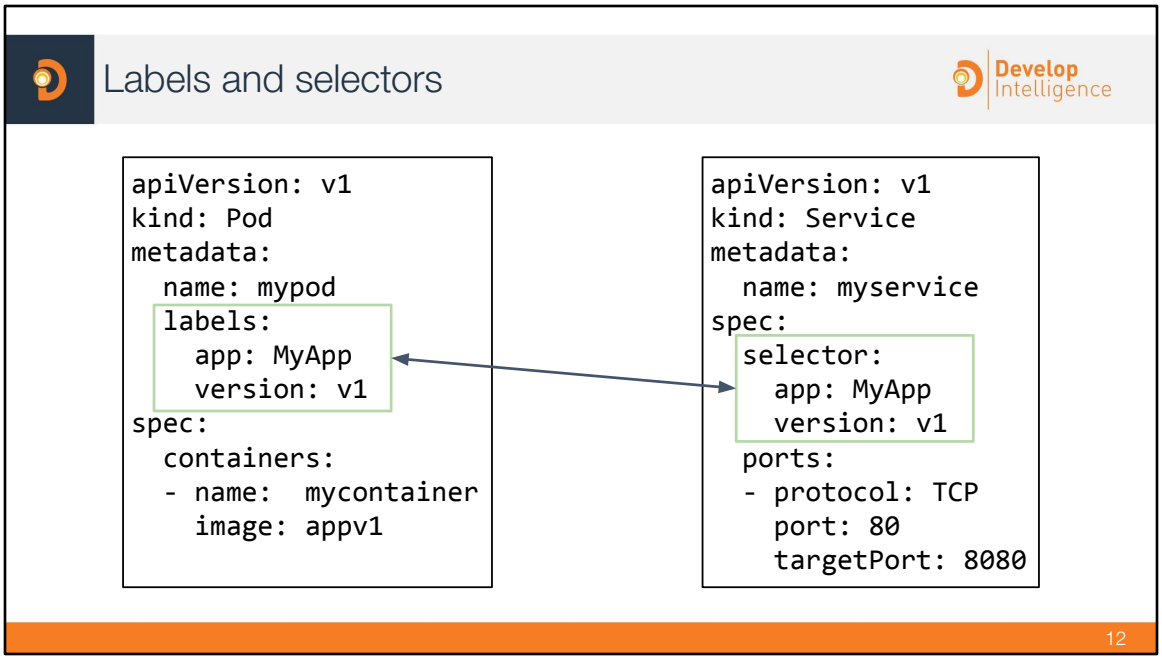
A Service is an abstraction that defines a logical set of pods and a way to access them over the network. Think of this as a load balancer.



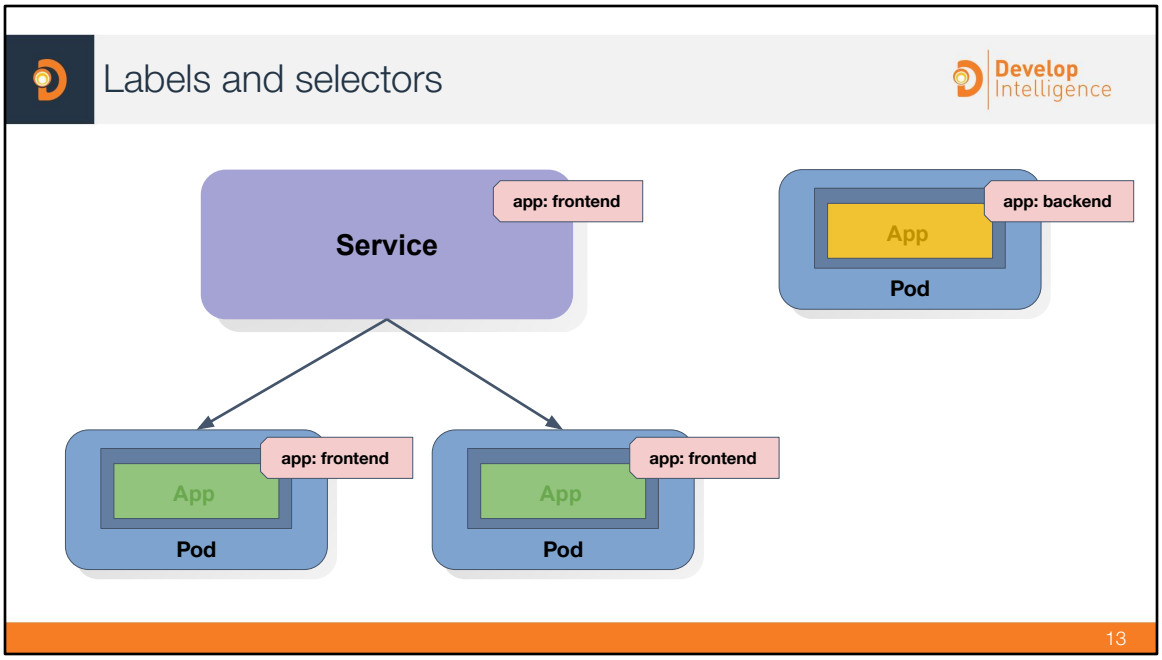
You may have a multi-tier application. In this case, the front end should talk to the backend through a service as well.



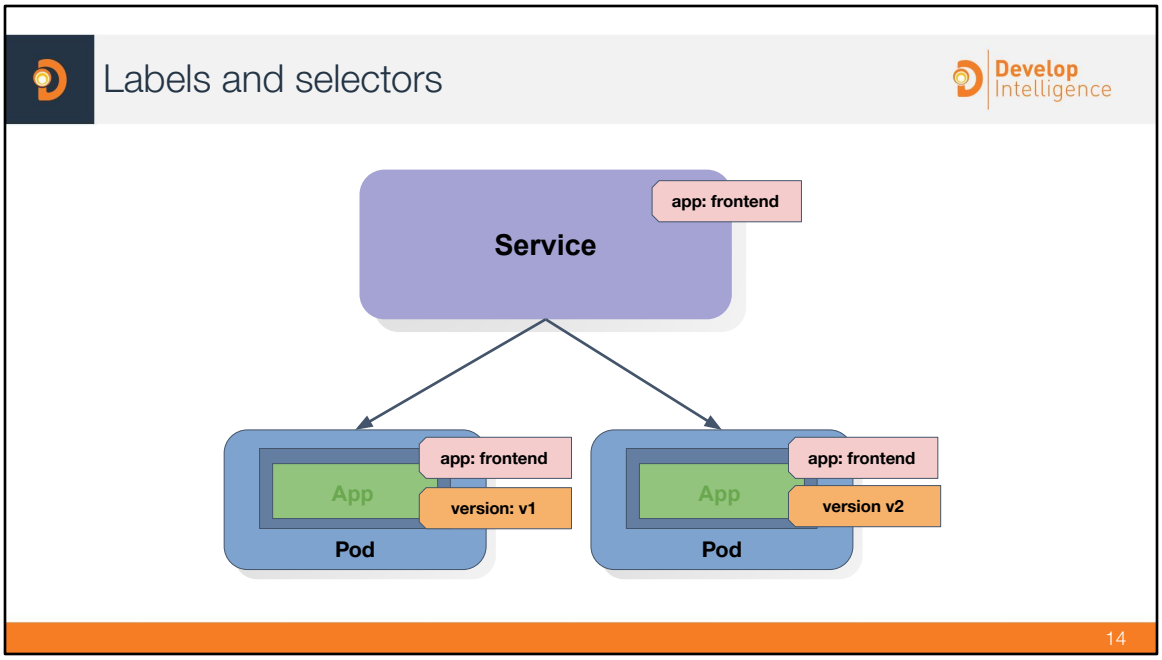
How a LoadBalancer type is deployed depends on the cluster. Clusters running in Google Cloud may be configured to use Google Load Balancers while an on-prem cluster may have an nginx server running (perhaps in a pod itself!).



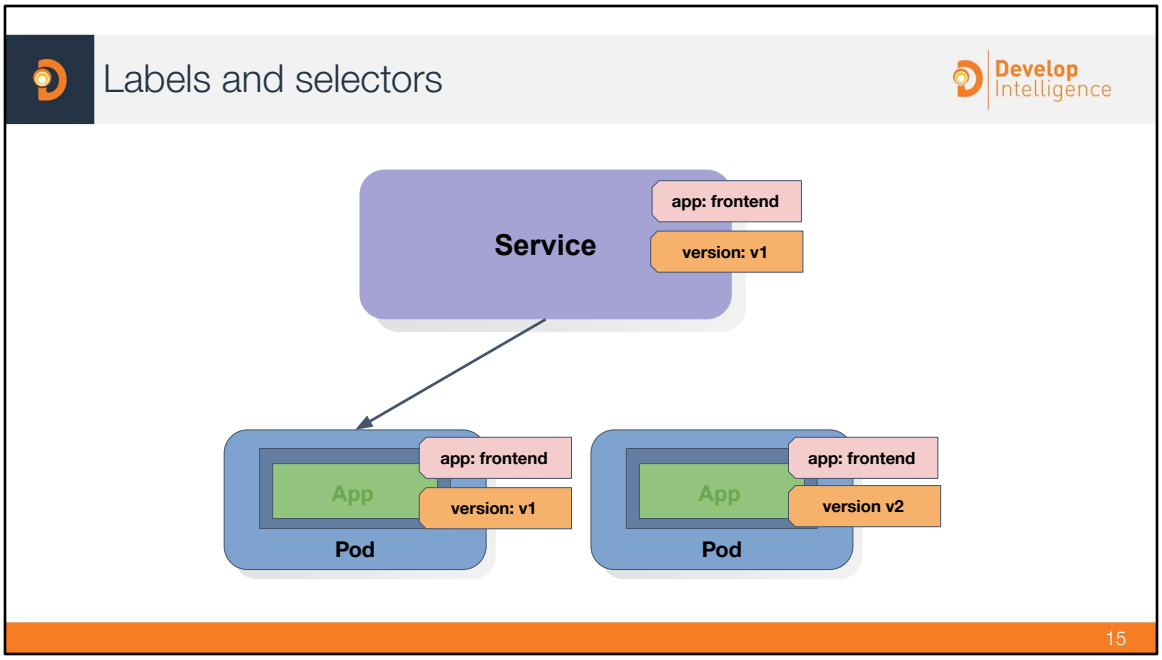
One-to-many relationships are typically denoted with labels and selectors. Any object can be given labels. Even nodes can have labels that we use to tell kubernetes where to schedule our pods. In the above example, the Service object has selectors that tell it to send traffic to pods that have both the *app: MyApp* and *version: v1* labels.



If we have different applications, we may decide to label the Pods to know what application it is running.

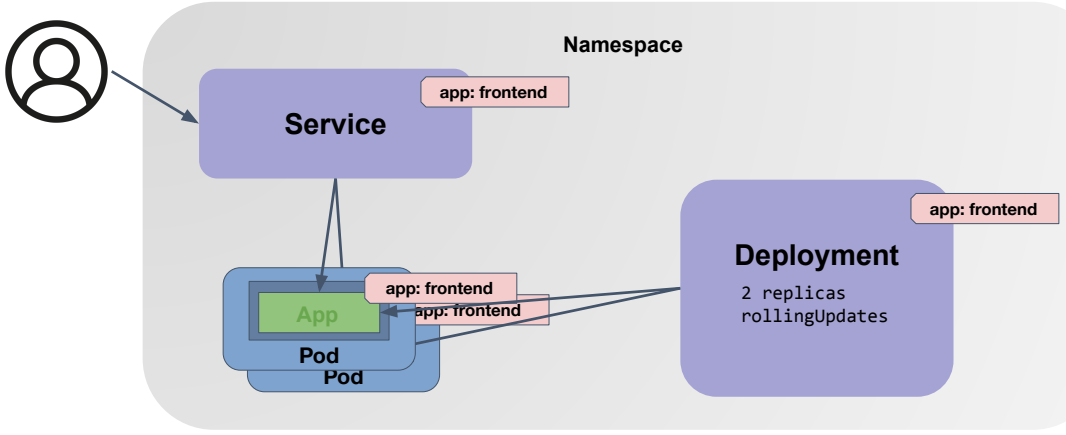


Services use selectors to know which pods to send traffic to.



Services use selectors to know which pods to send traffic to.

What we know so far




Lab
Develop
Intelligence
Declarative K8s
17

What is a pod?

Develop Intelligence

1. A minimal virtual machine containing running containers.

2. The smallest schedulable unit in k8s

3. A group of application nodes

How do we scale our application?

Develop Intelligence

1. Changing the replica count on our Deployment object

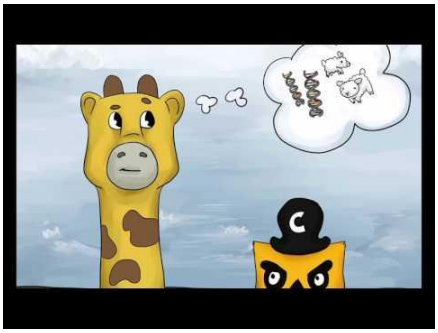
2. Adding new Deployment objects

3. Increasing/Decreasing the size of our Nodes (more RAM or CPU)

Video

Develop Intelligence

Children's Illustrated Guide to Kubernetes



20

https://www.youtube.com/watch?v=Q4W8Z-D-gcQ

Additional Resources

Develop Intelligence

- Pods - https://kubernetes.io/docs/concepts/workloads/pods/pod-overview/
- Deployments - https://kubernetes.io/docs/concepts/workloads/controllers/deployment/
- Services - https://kubernetes.io/docs/concepts/services-networking/service/
- Object Reference Docs - https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.14/