

# Debugging

# Useful commands

- Quick overview of everything on a cluster

```
kubectl get all
```

```
kubectl describe all
```

- Object state and events

```
kubectl describe <object> <object-name>
```

## Common pod errors

- `CrashLoopBackOff` : the pod runs a container that immediately exits.
- `ErrImagePull` : non-existing image.
  - In case of a rollout, you see it as "Pending" in the rollout status. If that "Pending" status is holding for long, look at the pods.

## Common service errors

- Default service type is `ClusterIP` which will only expose the service inside of the cluster.
- Shell into a pod in the cluster via `kubectl exec --it <pod> sh` and curl into the affected pod.
  - `kubectl get pods -o wide` gives you IPs of all pods in the cluster.

# Best practices

## Be declarative

- There is support for imperative commands ( `kubectl create` ) but abusing them is an antipattern.
- Use instead YAML files, filling up some templates:  

```
kubectl get <obj>/${MY-OBJ} -o yaml --dry-run >  
object.yaml
```
- `kubectl apply -f` works with URLs too.
- You can combine multiple YAML files with `---`.
- Remember `kubectl edit OBJ`.

## Be declarative (cont.)

- Use `--record` with `apply`
  - `kubectl apply -f auth.yaml --record`
  - `kubectl rollout history deploy auth`

# Pods

- One pod = one container (except for sidecars).
- Run one process per container
  - if you have 20 processes, how does K8s know if the container failed?
- Don't use `latest` or no tag.



# Availability

- Configure liveness and readiness probes.
- Provision at least 3 master nodes
  - `etcd` requires a majority to form a quorum and keep functioning.
- Distribute master and worker nodes across zones to prevent outages.
- Pods should be part of a deployment!
  - Naked pods are *not* rescheduled in case of node failure/shutdown.

# Resource Management

- Set up resource requests and limits.
- Use namespaces (but wisely).
- Configure resource quotas in namespaces to ensure that all resources get request and limit values.

## Use labels for all objects

Key	Description	Example
name	Application name	mysql
instance	Unique id	wordpress-abcd
version	Current version	5.7.2
component	Component within architecture	database
part-of	Parent application	wordpress
managed-by	Tool that manages its operation	helm

## Mapping external services

- Create a service without pod selectors and an `Endpoint` object with the external IP.
- This avoids hard-coding the IP into the application and makes updates easier.