VOLUMES AND DATA

Volumes and Data

Shared Volume Example

The following YAML file creates a pod, **exampleA**, with two containers, both with access to a shared volume:

```
....
   containers:
   - name: alphacont
     image: busybox
     volumeMounts:
     - mountPath: /alphadir
       name: sharevol
   - name: betacont
     image: busybox
     volumeMounts:
     - mountPath: /betadir
       name: sharevol
   volumes:
   - name: sharevol
     emptyDir: {}
```

Now, take a look at the following commands and outputs:

```
$ kubectl exec -ti exampleA -c betacont -- touch /betadir/foobar
$ kubectl exec -ti exampleA -c alphacont -- ls -l /alphadir

total 0
-rw-r--r-- 1 root root 0 Nov 19 16:26 foobar
```

You could use **emptyDir** or **hostPath** easily, since those types do not require any additional setup, and will work in your Kubernetes cluster.

Note that one container (**betacont**) wrote, and the other container (**alphacont**) had immediate access to the data. There is nothing to keep the containers from overwriting the other's data. Locking or versioning considerations must be part of the containerized application to avoid corruption.