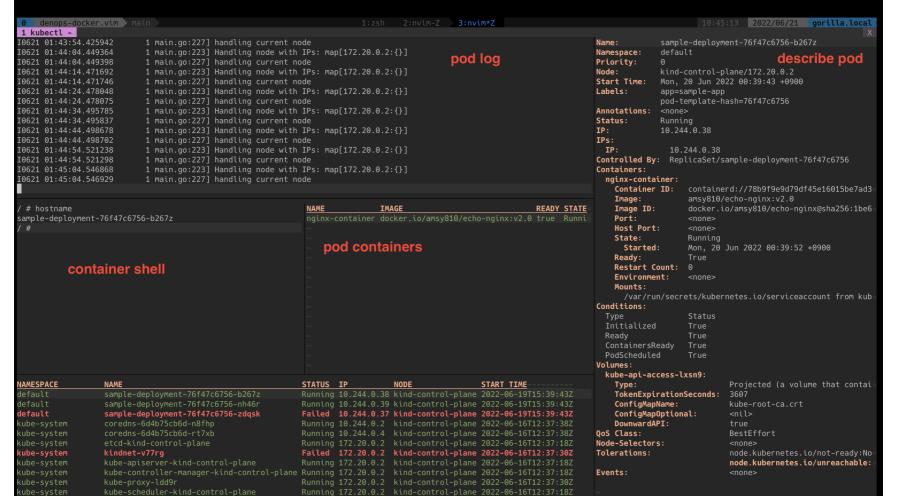


k8sk8s

k8s.vim

Vim k8s

- Pod List
- Pod Describe
- Attach Pods
- Service List
- Service Describe
- Events Watch
- . .



local-path-storage local-path-provisioner-9cd9bd544-tk676

Running 10.244.0.3 kind-control-plane 2022-06-16T12:37:38Z controller-7476b58756-w6p78 Running 10.244.0.54 kind-control-plane 2022-06-20T07:18:32Z

Running 172.20.0.2 kind-control-plane 2022-06-19T16:40:08Z

NORMAL term://~/dev/github.com/skanehira/denops-docker.vim//73029:/usr/local/bin/kubectl | unix | utf-8 | no ft 100% 10016:0

■ CLI (<u>k9s</u>

Vim

Pod

■ k8s

Vim

`kubectl` JSON

```
export async function getResourceAsText(
 resource: Resource,
): Promise<string> {
 const cmd = [
    "kubectl",
    "get",
   resource.type,
  ];
 if (resource.opts?.name) {
    cmd.push(resource.opts.name);
 if (resource.opts?.format) {
    cmd.push("-o", resource.opts.format);
 if (resource.opts?.fields) {
    cmd.push("--field-selector", resource.opts.fields);
 const output = await run(cmd);
 return output;
```

```
const opt = {
                                                           $ kubectl get pods -A -o json
  type: "pods",
  opts: {
   namespace: "all",
   format: "json"
await getResourceAsText(opt);
const opt = {
                                                           \ kubectl get rs -n default -o yaml \
  type: "rs",
                                                                 --field-selector metadata.name=nginx
  opts: {
   namespace: "default",
   format: "yaml",
   fields: "metadata.name=nginx"
 },
```

await getResourceAsText(opt);

```
Sk8s` JSON Vim
```

```
function! K8sDebugPod() abort
  let pod = b:k8s_pods[line(".")-2]
  call k8s#util#terminal#kubectl('debug', pod.metadata.name, '-it', '--image=busybox', '--', 'sh')
endfunction
command! K8sDebugPod :call K8sDebugPod()<CR>
```

Vim script

- `kubectl`

 `describe` `kubectl get event -A -w`

 `kubectl edit`
 - `k8s` openapi

\$ nerdctl ps
CONTAINER ID

```
kubectl get --raw /openapi/v2
```

IMAGE

COMMAND

CREATED

```
Worker `namespace`
```

PORTS

NAMES

STATUS