Helm Kubernetes package manager

CONCEPTS

A chart is the package, a release is one installation of it. Each upgrade creates a new numbered revision.

Repository chart index
↓ `helm pull / install`
Chart + values templates
↓ `helm template`
Manifests rendered YAML
↓ `helm install`
Release revisions in cluster

Helm uses your current kubectl context. Add -n <namespace> to every command, or releases land in the default namespace.

REPOSITORIES

`helm repo add <name> <url>`
Add a chart repository

`helm repo update`
Refresh local chart indexes

`helm search repo <keyword>`
Search added repositories

INSPECT CHARTS

`helm show values <chart>`
Print the default values

INSTALL

`helm install <release> <chart>`
Install a chart as a release

`helm install <release> <chart> -n <ns> --create-namespace`
Install into a new namespace

`helm install <release> <chart> -f <values.yaml>`
Install with a values file

`helm install <release> <chart> --set <key>=<val>`
Override a single value

INSTALL (cont.)

`helm install <release> <chart> --version <ver>`
Pin the chart version

`helm install <release> <chart> --wait --timeout 5m`
Wait until resources are ready

RELEASES

`helm list`
List releases in this namespace

`helm list -A`
List releases in all namespaces

`helm status <release>`
Show release status and notes

`helm get values <release>`
Values you supplied

`helm history <release>`
List revisions of a release

UPGRADE & ROLLBACK

`helm upgrade <release> <chart> -f <values.yaml>`
Upgrade with a values file

`helm upgrade --install <release> <chart>`
Install or upgrade in one command

`helm rollback <release>`
Roll back to the previous revision

`helm rollback <release> <revision>`
Roll back to a given revision

`helm uninstall <release>`
Remove a release

TEMPLATING & DEBUG

`helm template <release> <chart> -f <values.yaml>`
Render manifests locally

`helm lint <chart-dir>`
Check a chart for problems