

Bring Your Helm Chart to the Wonderful World

Mansun Kuo
2024-10-23



Outline

- Introduction
- Exercises
- 11 Tips of Helm
- References

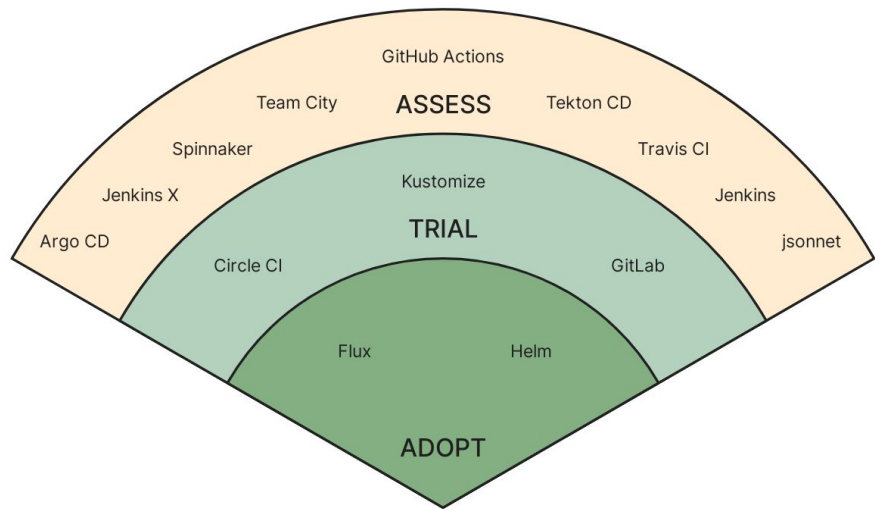
Introduction

What is Helm

- A handy tool that help you manage your applications on k8s
 - Declarative
 - Helm template language (based on Go template)
 - Variable, flow control, pipeline, ...
 - Manage complexity
 - Simple sharing
 - Rollback to any revision
- Adopted in CNCF technology radar, June 2020

CNCF Technology Radar

Continuous Delivery, June 2020



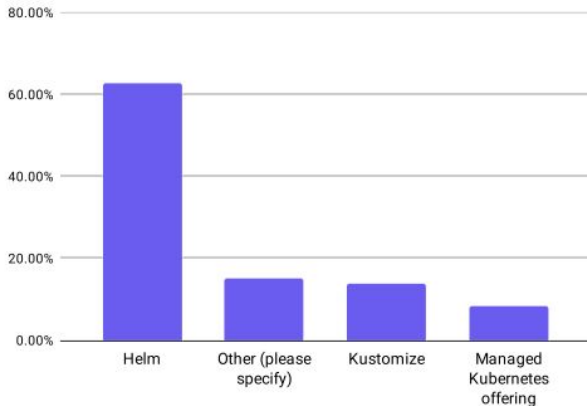
<https://radar.cncf.io/2020-06-continuous-delivery.png>

Alternative of Helm

- Helm is the most popular way to packaging applications on k8s
- Some other common approaches
 - Kustomize
 - Pure kubectl cli with envsubst
 - Custom scripts with template

Packaging Applications

What is your preferred method for packaging Kubernetes applications?



Helm is still the most popular tool for packaging Kubernetes applications and is used by 63% of respondents.

https://www.cncf.io/wp-content/uploads/2020/11/CNCF_Survey_Report_2020.pdf



<https://www.cncf.io/reports/cncf-annual-survey-2023/>

Three Key Concepts

Helm install **charts** into Kubernetes, creating a new **release** for each installation. You can search Helm chart **repositories** to find new charts.

- Chart
 - A helm package contains all of the resource definitions necessary to run an application, tool, or service inside of a Kubernetes cluster.
 - The Kubernetes equivalent of a Python package, apt, dpkg, or a Yum RPM file
- Repository
 - The place where charts can be collected and shared
 - Like Python's Pypi or the Fedora Package Database
- Release
 - An instance of a chart running in a Kubernetes cluster

Basic Commands

- `helm search [command]`
 - Find charts
- `helm repo [command]`
 - Add a repository
- `helm install [NAME] [CHART] [flags]`
 - Install a Helm chart
- `helm upgrade [RELEASE] [CHART] [flags]`
 - Upgrade (or install) a Helm chart
 - `helm upgrade --install`
- `helm list [flags]`
 - List all releases for a specific namespace
- `helm uninstall [RELEASE_NAME] [...] [flags]`
 - Uninstall a helm chart

Exercises

Exercise 1: Create Your First Helm Char

- Tutorial
 - [English](#)
 - [繁體中文](#)
- [GitHub](#)

Exercise 2: Modify It as an API

- Tutorial
 - [English](#)
 - [繁體中文](#)
- [GitHub](#)

Exercise 3: Why My Secret Is Not Updated

- Tutorial
 - [English](#)
 - [繁體中文](#)
- [GitHub](#)

Exercise 4: Add a Helm Dependency

- Tutorial
 - [English](#)
 - [繁體中文](#)
- [GitHub](#)

Exercise 5: Chart Releaser Action

- Tutorial
 - [English](#)
 - [繁體中文](#)
- [GitHub](#)
- Add your chart [here](#)

11 Tips of Helm

1. Get Help and Current Version

- `helm -h`
 - Get all available command
- `helm [command] -h`
 - Get help of a sub command
- `helm version`
 - Get version

2. Use Helm Template to Check Values

- Customizing values
 - `--values`: Specify a YAML file with overrides
 - `--set`: Specify overrides on the command line
- `helm template [NAME] [CHART] [flags]`
 - Render chart templates locally and display the output

3. Create your own chart with helm create

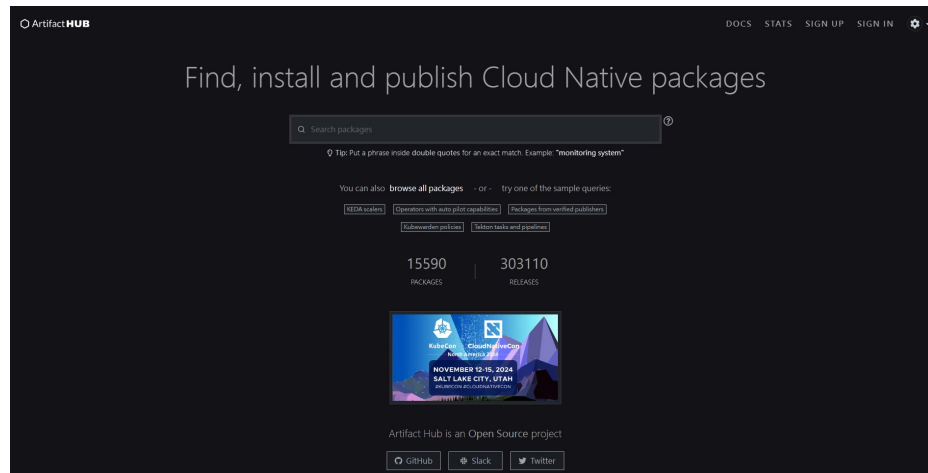
- helm create NAME [flags]
 - Create your own chart directory with the common files and directories used in a chart
- More about chart development
 - [Chart Development tip and Tricks](#)
 - [Built-in Objects](#)
 - [Template Functions and Pipelines](#)
 - [Template Function List](#)
 - [Flow Control](#)
 - [Variables](#)
 - [Named Templates](#)

4. Use “partial” and Template Includes

- In the templates/ directory, any file that begins with an underscore(_) is not expected to output a Kubernetes manifest file.
- By convention, helper templates and partials are placed in a __helpers.tpl file.

5. Find Packages on ArtifactHUB

- Artifact Hub
 - An open-source project to discover, install and publish packages and configurations



<https://artifacthub.io/>

6. Use Helm dependency

- You can add other packages with specific version into your chart with [helm dependency](#)
- You can also [overriding values from a parent chart](#).

```
dependencies:
- condition: redis.enabled
  name: redis
  repository: oci://registry-1.docker.io/bitnamicharts
  version: 20.x.x
- condition: postgresql.enabled
  name: postgresql
  repository: oci://registry-1.docker.io/bitnamicharts
  version: 16.x.x
- name: common
  repository: oci://registry-1.docker.io/bitnamicharts
  tags:
  - bitnami-common
  version: 2.x.x
```

<https://github.com/bitnami/charts/blob/main/bitnami/airflow/Chart.yaml#L22-L35>

7. Use common logic in bitnami/common

Bitnami Common Library Chart

A [Helm Library Chart](#) for grouping common logic between Bitnami charts.

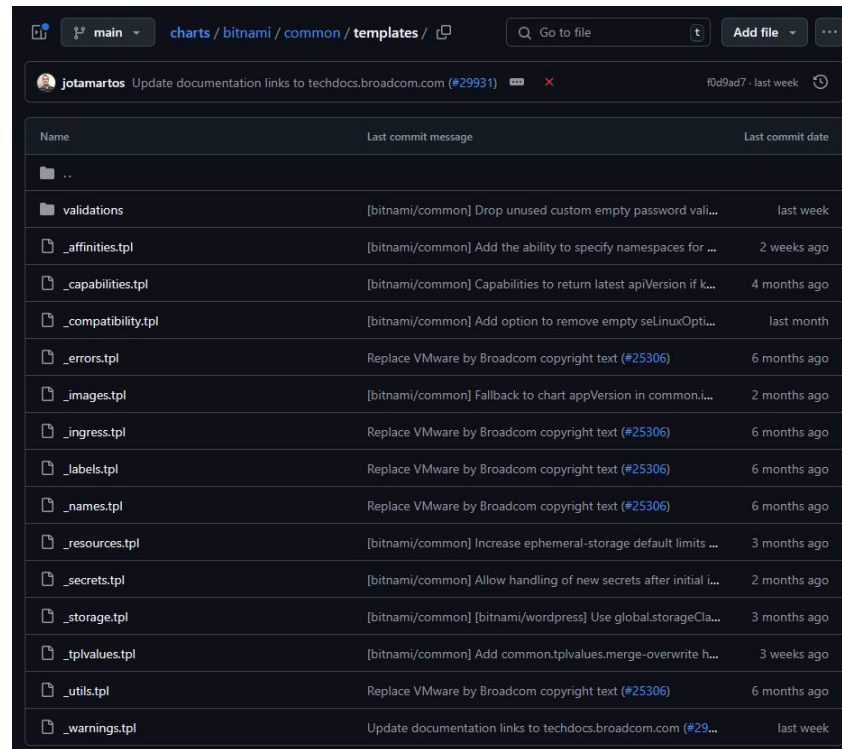
TL;DR

```
dependencies:
- name: common
  version: 2.x.x
  repository: oci://registry-1.docker.io/bitnamicharts
```

```
helm dependency update
```

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: {{ include "common.names.fullname" . }}
data:
  myvalue: "Hello World"
```

<https://github.com/bitnami/charts/tree/master/bitnami/common>



The screenshot shows the GitHub repository for the Bitnami Common Library Chart, specifically the `templates` directory. The repository is owned by `jotamartos` and has a commit history of 29931. The table below lists the files in the `templates` directory, their last commit messages, and their last commit dates.

Name	Last commit message	Last commit date
..		
validations	[bitnami/common] Drop unused custom empty password vali...	last week
_affinities.tpl	[bitnami/common] Add the ability to specify namespaces for ...	2 weeks ago
_capabilities.tpl	[bitnami/common] Capabilities to return latest apiVersion if k...	4 months ago
_compatibility.tpl	[bitnami/common] Add option to remove empty seLinuxOpti...	last month
_errors.tpl	Replace VMware by Broadcom copyright text (#25306)	6 months ago
_images.tpl	[bitnami/common] Fallback to chart appVersion in common.t...	2 months ago
_ingress.tpl	Replace VMware by Broadcom copyright text (#25306)	6 months ago
_labels.tpl	Replace VMware by Broadcom copyright text (#25306)	6 months ago
_names.tpl	Replace VMware by Broadcom copyright text (#25306)	6 months ago
_resources.tpl	[bitnami/common] Increase ephemeral-storage default limits ...	3 months ago
_secrets.tpl	[bitnami/common] Allow handling of new secrets after initial ...	2 months ago
_storage.tpl	[bitnami/common] [bitnami/wordpress] Use global.storageCla...	3 months ago
_tplvalues.tpl	[bitnami/common] Add common.tplvalues.merge-overwrite h...	3 weeks ago
_utils.tpl	Replace VMware by Broadcom copyright text (#25306)	6 months ago
_warnings.tpl	Update documentation links to techdocs.broadcom.com (#29...	last week

<https://github.com/bitnami/charts/tree/master/bitnami/common/templates>

8. Use Helm Plugin

- A Helm plugin is a tool that can be accessed through the helm CLI, but which is not part of the built-in Helm codebase.
- The core of a plugin is a simple YAML file named plugin.yaml
- helm-secrets
 - Decrypt secrets with Secret Backends
 - sops
 - vals
 - Hashicort Vault
 - AWS Secrets Manager
 - GCP Secrets Manager
 - Azure Key Vault
 - ...

9. Commit Your Charts into Git

- Treat git repository as the single source of truth of your infrastructure
- Never store secrets in plain text in git
- GltOps
 - A git repository with versioned CI/CD on top of declarative infrastructure
 - Deploy a new revision with updating the repository and the automated process handles everything else

10. Share Your Chart

- Install sources:
 - A chart repository
 - A local chart archive
 - An unpacked chart directory
 - A full URL
- Share
 - Git repository
 - Mount to other project with git submodule
 - Chart repository
 - An HTTP server that contains an index.yaml and optionally some packaged charts
 - Registry
 - Use an Docker registry with [Helm registries](#)

11. Have a better Developer Experience

- IDE
 - Visual Studio Code
 - [Kubernetes](#): Syntax highlight & UI
 - [Helm Intellisense](#): Intellisense for helm templates
- Autocomplete
 - [helm completion](#)

References

References

- [Kubernetes Summit 2024 - Workshop](#)
- [Bring Your Helm Chart to the Wonderful World](#)

Thanks
