

# What The Helm?!

Panoramica su Helm

Luca Andrea Fusè

28 Ottobre 2023

- Membro onorario dell'associazione [P.O.u.L.](#)
- Socio di [I.L.S.](#) nella sezione di Milano
- Membro di [Red Hat Accelerators](#)
- Consulente per [Kyndryl](#)
  - TL;DR Faccio cose con OpenShift/Kubernetes/Cloud
- Sono ~~1~~0 giorni che non parlo di Kubernetes e Linux

# Di cosa parleremo?

- 1 Cosa fa e quali problemi risolve?
- 2 Un minimo di terminologia...
- 3 Comandi per iniziare
- 4 Anatomia di un Helm Chart (**molto** concentrata)
- 5 Quando Helm ci fa dire WTH?!

Cosa fa e quali problemi risolve?

- Helm è un tool che permette di configurare e deployare applicazioni su Kubernetes
- Semplifica il deploy sfruttando l'uso di **variabili** e **template**
- In poche parole, Helm è "il gestore di pacchetti" per Kubernetes

# Quali problemi risolve?

- Voglio avere la certezza che i deploy che faccio siano *sempre uguali*
- Voglio avere la possibilità di riutilizzare e condividere il ~~template~~ Helm Chart che ho creato
- Voglio fare un deploy di più applicazioni con dipendenze tra di loro
- Voglio che sia anche possibile gestire un update/rollback di quanto deployato

Un minimo di terminologia...

# Un minimo di terminologia...

## Helm

Tool da cli che ci permette di creare e gestire gli **Helm Chart**



# Un minimo di terminologia...

## Helm Chart

Progetto che contiene informazioni, template di risorse ed eventuali valori di default che vengono processati quando viene fatto un deploy.

# Un minimo di terminologia...

## Template di una risorsa

Template (Go template) che, in base alle logiche con cui è stato scritto e alle variabili passata, va ad essere convertito nella definizione della risorsa

# Un minimo di terminologia...

## Helm repository

Sorgente dalla quale vengono scaricati gli Helm Chart

## Comandi per iniziare

## Helm repo

Con il comando `helm repo add` possiamo aggiungere un repository da cui pescare gli Helm Chart

## Esempio

```
$ helm repo add \
  bitnami https://charts.bitnami.com/bitnami
```

## Helm search repo

Il comando `helm search repo` ci permette di cercare nei repository aggiunti un determinato termine

## Esempio

```
$ helm search repo mysql
```

NAME	CHART VERSION	APP VERSION
bitnami/mysql	9.14.1	8.0.35
bitnami/phpmyadmin	13.0.0	5.2.1
bitnami/mariadb	14.1.0	11.1.2



## Helm Install

Il comando `helm install` permette di deployare l'applicazione/le applicazioni definite nel nostro chart

## Esempio

```
$ helm install \  
  -f myvalues.yaml --set mariadb.auth.database=userdb \  
  my_wordpress bitnami/wordpress
```

# Anatomia di un Helm Chart (**molto** concentrata)

# Anatomia di un Helm Chart

Abbiamo già accennato a che cosa sia un Helm Chart, ma:

- Come faccio a crearne uno?
- Da cosa è composto?
- Come sono fatti i template al loro interno?

# Anatomia di un Helm Chart

## Creazione di un Helm Chart

Per creare un Helm Chart posso facendolo a mano oppure con il comando

```
helm create
```

# Anatomia di un Helm Chart

## Esempio

```
$ helm create mychart
$ ls mychart
.helmignore
Chart.yaml
values.yaml
charts/
templates/
```

## Da cosa è composto?

Un helm chart è composto da diverse file, di cui i principali sono:

- `Chart.yaml`
- `values.yaml`
- `templates/`

# Anatomia di un Helm Chart

## Chart.yaml

```
apiVersion: v2
name: wordpress
description: WordPress is the world...
[...]
dependencies:
- name: mariadb
  repository: [...]
  version: 14.x.x
[...]
version: 18.0.10
appVersion: 6.3.2
```



# Anatomia di un Helm Chart

## values.yaml

```
[...]
image:
  registry: docker.io
  repository: bitnami/wordpress
  tag: 6.3.2-debian-11-r3
  pullPolicy: IfNotPresent
  pullSecrets: []
  debug: false
[...]
livenessProbe:
  enabled: true
  httpGet:
    path: /wp-admin/install.php
    port: '{{_.Values.wordpressScheme}}'
    scheme: '{{_.Values.wordpressScheme|upper}}'
    httpHeaders: []
[...]
```

# Anatomia di un Helm Chart

## template/deployment.yaml

```
apiVersion: {{ include "common.capabilities.deployment.apiVersion" . }}
kind: Deployment
metadata:
  name: {{ include "common.names.fullname" . }}
  namespace: {{ .Release.Namespace | quote }}
[...]
containers:
- name: wordpress
  image: {{ include "wordpress.image" . }}
  imagePullPolicy: {{ .Values.image.pullPolicy | quote }}
  {{- if .Values.diagnosticMode.enabled }}
  command: >
    {{- include "common.tplvalues.render"
      (dict "value" .Values.diagnosticMode.command "context" $) | nindent 12 }}
  {{- else if .Values.command }}
  command: >
    {{- include "common.tplvalues.render"
      (dict "value" .Values.command "context" $) | nindent 12 }}
  {{- end }}
[...]
```

Quando Helm ci fa dire WTH?!

## Cosa può andare storto?

Per quanto Helm sia una ottima soluzione, a volte può farci tirare alcuni ~~WTF?! WTH?!~~.

## Template di Helm

Il motore che è stato scelto per i template è quello di Go.  
I suoi costrutti, assieme alla sintassi yaml delle risorse k8s, rendono la scrittura e la lettura dei template di Helm non facilissima.

## Dipendenze

Con la sezione `dependencies` di `Chart.yaml` posso andare a dichiarare gli Helm Chart da cui il nostro dipende.

Questo porta ovviamente a possibili bug/incompatibilità in base ai Chart che sto utilizzando e su cui non ho il controllo.

## Complessità

Anche se abbiamo un martello, non vuol dire che tutto debba essere un chiodo.

Per quanto Helm sia un ottimo strumento, non è detto che per il nostro deploy sia uno strumento sensato.

## Complessità 2

Poter far in modo che il nostro Helm Chart dichiari gli altri Chart di cui necessita è **molto** comodo.

Però bisogna far attenzione a non lasciarsi prendere (troppo) la mano!



## Sicurezza

Quando si utilizzano Helm Chart provenienti dalla community, bisogna fare attenzione ad utilizzare Chart che si reputano affidabili.

Questo si risolve facilmente utilizzando Chart ufficiali oppure scrivendone di nostri custom.

*Domande?*

- [Documentazione di Helm](#)
- [Helm Chart](#) da cui sono stati tratti gli snippet
- [Artifactor Hub](#) da cui si possono recuperare diversi Helm Chart (oltre a diversi altri tipi di oggetti)
- [Repo di queste slides](#)

**Grazie per l'attenzione!**

Queste slides sono rilasciate sotto [CC-BY-SA 4.0](#)

