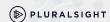
Helm Intro



Intro to Helm

- "Package manager" for Kubernetes
- https://helm.sh
- Common public repos: S3, GitHub, Docker Hub
- Installation prereqs: Docker, Kubernetes, kubectl



Demo: Install, Inspect, and Deploy with Helm

Lab 01: Deploy Prefab Chart



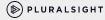
Chart Building



Building Helm Charts

Relies fundamentally on Kubernetes manifests under the hood

- The "magic" comes with Go templating (more on this later)
- Added benefit of semantic versioning, upgrades, and rollbacks like a "proper" package manager



Demo: Deploy Custom Chart

Lab 02: Build App and Deploy via Helm

Chart Dependencies



Managing Dependencies

Related to but distinct from "subcharts" (covered later)

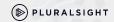
- Conceptually the same as regular program dependencies
- Defined in chart.yaml or separate requirements.yaml
- Third parties and avoiding "dependency hell"



Demo: Manage Dependencies

Lab 03: Manage Dependencies

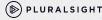
Helm Templating



Helm templates

- Relies on Go template syntax (e.g. {{ .Values.var }})
- Good to learn from the default templates created by helm create

- Different scopes of applied values (hardcoded, values.yaml, CLI)
- Highly flexible logic familiar to programmers, e.g. iterating through lists



Demo: Helm Templates



Lab 04: Helm Templates



Demo: Spring App with Built-In Helm Functions

Lab 05: Templates and Built-In Functions

Demo: Custom Functions



Lab 06: Write Custom Functions

Helm Plugins



Helm plugins

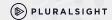
 External modules/programs used by helm (generally called by **helm** wrapper)

- Can be written in any programming language
- Some <u>curated lists</u> out there

 Be sure to perform due diligence as with any third party software



Demo: Write and Manage Plugins



Lab 07: Write and Manage Plugins



Subcharts



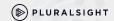
Subcharts

 At its simplest, a chart directory within an existing chart directory

- Not required, but best to declare as explicit dependency
- Deploy multiple charts in a single release
- Can propagate values down to subcharts



Demo: Write and Deploy Subcharts



Lab 08: Write and Deploy Subcharts



Final Q&A



Thank you!

If you have any additional questions, please ask!

