adaptTo()

APACHE SLING & FRIENDS TECH MEETUP

2 - 4 SEPTEMBER 2019

Current State of Peregrine-CMS

A Content Management System built on VueJS and Sling

Who we Are

Gaston Gonzalez

Senior Architect, headwire.com, Inc

Ruben Reusser

CTO, headwire.com, Inc

DEMO

Deploying Peregrine to k8s

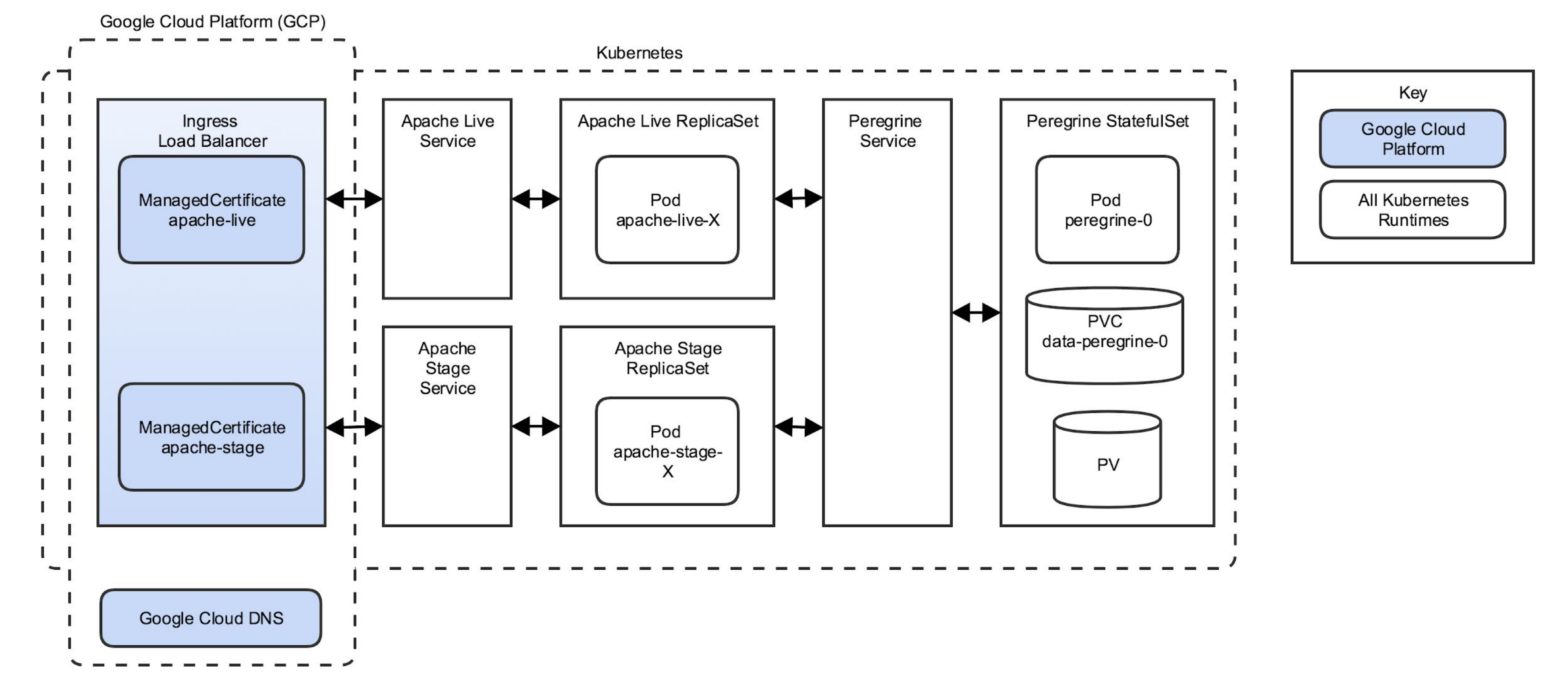
1. Add Peregrine to Helm repository

```
$ helm repo add peregrine https://peregrine-cms.github.io/helmcharts/
$ helm repo update
```

2. Deploy Peregrine to Kubernetes

```
$ helm install --name demo peregrine/peregrine
```

GKE & Peregrine



adaptTo()

LET’S AUTHOR A PAGE

Current State of Peregrine

- Sites are near 100 points in Chrome Audit
- Page transitions on hyperlinks are wicked fast
- The Admin UI is almost as fast :-)
- Almost ready for its 1.0.0 release
- Could use some more documentation

Agenda

- Quick demo
- Why are we developing Peregrine?
- What are our basic principals?
- Who is using Peregrine today?
- Where will Peregrine be in one year?

Agenda

- Quick demo
- **How did we get to that speed?**
- Why are we developing Peregrine?
- What are our basic principals?
- Who is using Peregrine today?
- Where will Peregrine be in one year?

adaptTo()

Page Speed

Page Speed

Can You get There?

Create a sample with the basic technology you'd like to use and evaluate it regarding to speed

Page Speed

Use Available Tools to Measure!

- Chrome Audit
- Lighthouse
- Axe
- Web.Dev
- PageSpeed Insights

Page Speed

- Easy Optimizations
  - Combine & Minimize CSS/JS
  - Inline (critical) CSS/JS
  - Compression
  - Modern Image Formats
  - SSL Session Sharing

Page Speed

Mod PageSpeed

https://developers.google.com/speed/pagespeed/module/

APACHE INCUBATOR

Page Speed

Bootstrap vs Utility based CSS Approach

- Bootstrap does not reach 100 points out of the box
- Utility First CSS approach delivers smaller CSS out of the box and desired page speed
- Utility First CSS is harder to get right
- How much JavaScript is still necessary today?

Page Speed

Page Transition Speed

- Use SPA Concepts
  - Less Data to Transmit
  - Browser has to evaluate less CSS

adaptTo()

Why Peregrine

Why Peregrine

- More influence on all aspects of the CMS we work with
- Enable outside site developers to contribute
- Determine how 'we' should write websites for the future
- Be able to produce an extendable site right out of the gate
- Show how awesome Apache Sling and Jackrabbit Oak are
- Bring Apache Sling to a wider audience

Our Principals

Our Principles

- Be fast
- Be open
- Be inclusive to all
- Build for today, try to keep the future in mind
- Bring back the fun!

Who is Using Peregrine

Who uses Peregrine?

cyber:con

meta:proc

head wire

HOTWAX COMMERCE

weedguide™

Outlook

Outlook

- Continue building the Community
- Focus on Speed, Accessibility, best practices
- Richer, better documentation
- Contribute back to what we use
- UX Improvements
- Make the authoring experience authorable
- Work with the Community on the future of Peregrine

adaptTo()

Thanks To

Thanks To

- Apache Sling
- Apache Jackrabbit Oak
- Composum
- VueJS
- Trumbowyg
- Vue Form Generator
- Swagger (OpenAPI)
- Apache Maven
- WCM.io
- NodeJS
- …

Thanks To

Our Contributors:

Andreas Schaefer
Andreas Hauser
Ben Kahn
Byron Ponce
Devin Tuffy
Dimitri Plotnikov
DJ Pelland
Erin Dailey
Fabian Haupt
Felix Pütz
Gaston Gonzalez
Henry Saginor
Jak Ratiwanich
Li Qin
Muzaffar Nurmukhammadov
Nicholaus Perez
Niklas Fonseca
Luis Plotnikov
Ruben Reusser
Swapnil M Mane
William McKeehan
Xan Nick

adaptTo()

Summary

Summary

- Page Speed is important! Make it a focus of your next project
- Get in the game, start building smaller sites on Apache Sling
- Join the Peregrine Community!

Start on your Local Computer:

Docker

```
$ docker run -it -p 8080:8080 peregrinecms/peregrine-cms:sling11
```

NPM

```
$ npm install percli -g
$ percli server install
```

Resources & Questions?

peregrine-cms: https://www.peregrine-cms.com

peregrine slack: https://peregrine-cms.slack.com

peregrine git repo: https://github.com/headwirecom/peregrine-cms

mod_pagespeed: https://developers.google.com/speed/pagespeed/module/

web.dev: https://web.dev

pagespeed insights: https://developers.google.com/speed/pagespeed/insights/

axe: https://www.deque.com/axe/

lighthouse: https://www.npmjs.com/package/lighthouse

chrome audit: ctrl-shift-i > audits

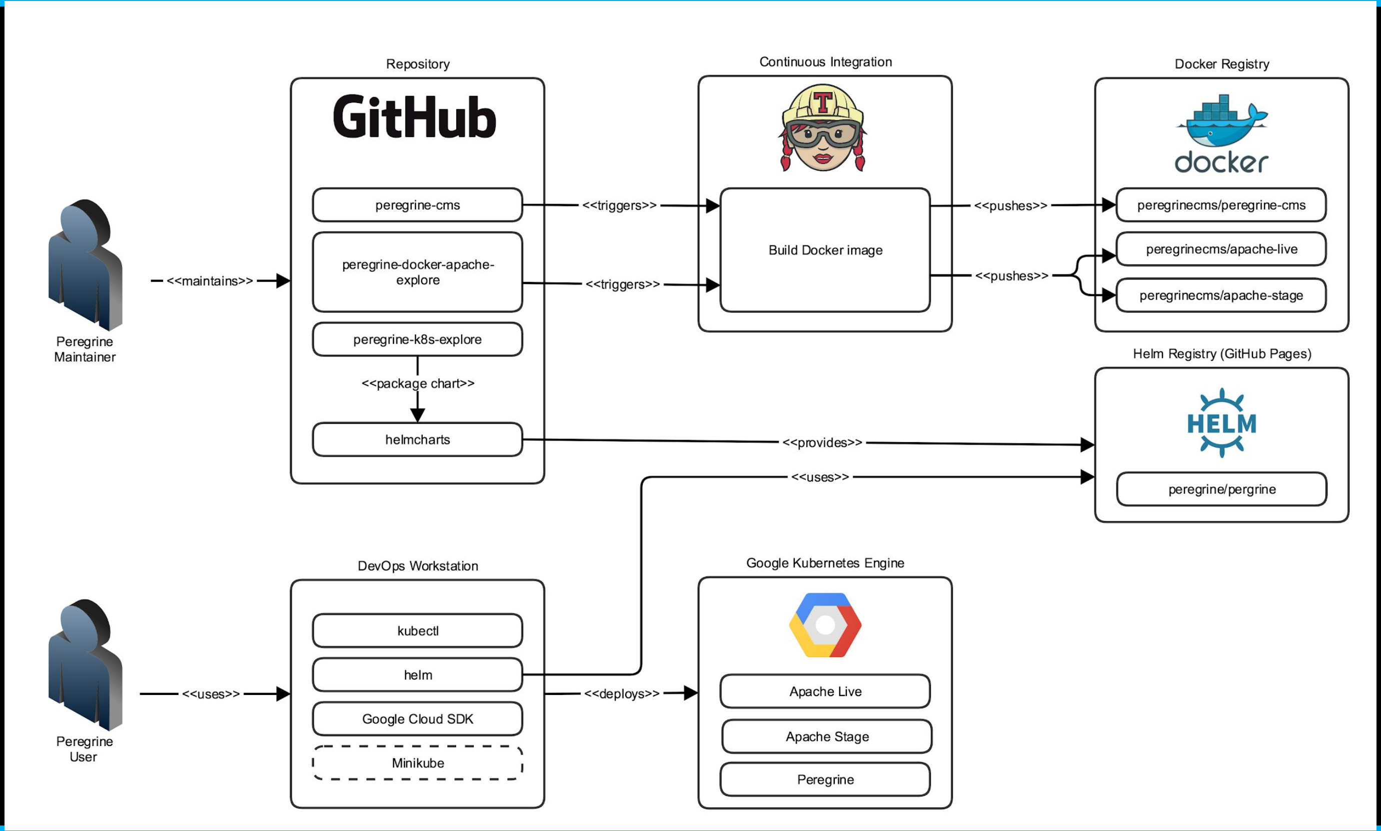
adaptTo()

Thank You!

Appendix

Docker and Kubernetes (k8s)

Container Workflows



Repository

GitHub

peregrine-cms

peregrine-docker-apache-explore

peregrine-k8s-explore

<<package chart>>

helmcharts

Continuous Integration

Build Docker image

Docker Registry

docker

peregrinecms/peregrine-cms

peregrinecms/apache-live

peregrinecms/apache-stage

Helm Registry (GitHub Pages)

HELM

peregrine/pergrine

Peregrine Maintainer

<<maintains>>

<<triggers>>

<<triggers>>

<<pushes>>

<<pushes>>

<<provides>>

<<uses>>

Peregrine User

<<uses>>

DevOps Workstation

kubectl

helm

Google Cloud SDK

Minikube

<<deploys>>

Google Kubernetes Engine

Apache Live

Apache Stage

Peregrine

Kubernetes: State-of-the-State

- Supported (tested) Runtimes
  - Minikube
  - Kubeadm (a.k.a bare metal)
  - Google Kubernetes (GKE)
- Future Runtimes
  - Azure Kubernetes Service (AKS)
- Upcoming Changes
  - Segment NodeStore