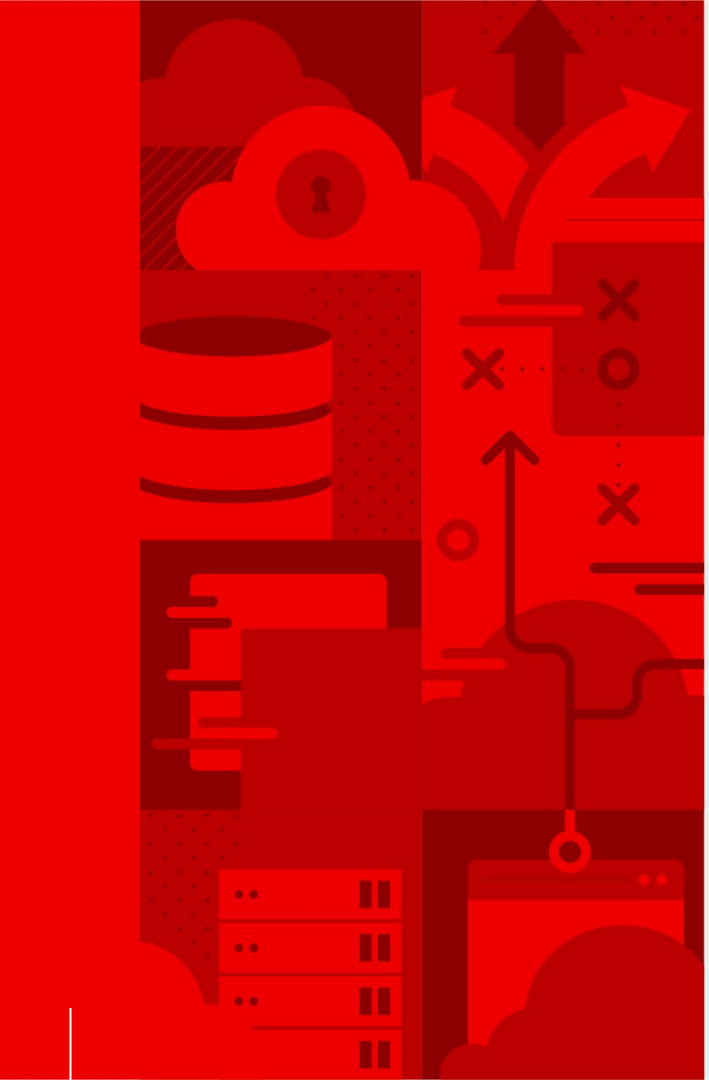


OpenShift Developer

Technical Training

Automating with Tekton & ArgoCD

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat

Self introduction

Name: Wanja Pernath

Email: wpernath@redhat.com

Base: Germany (very close to the Alps)

Role: EMEA Technical Partner Development Manager - OpenShift and MW

Experience: Years of Consulting, Training, PreSales at Red Hat and before

Twitter: https://twitter.com/wpernath

LinkedIn:https://www.linkedin.com/in/wanjapernath/

GitHub: https://github.com/wpernath

First book just published

Getting GitOps

A technical blueprint for developing with Kubernetes and OpenShift based on a REST microservice example written with Quarkus

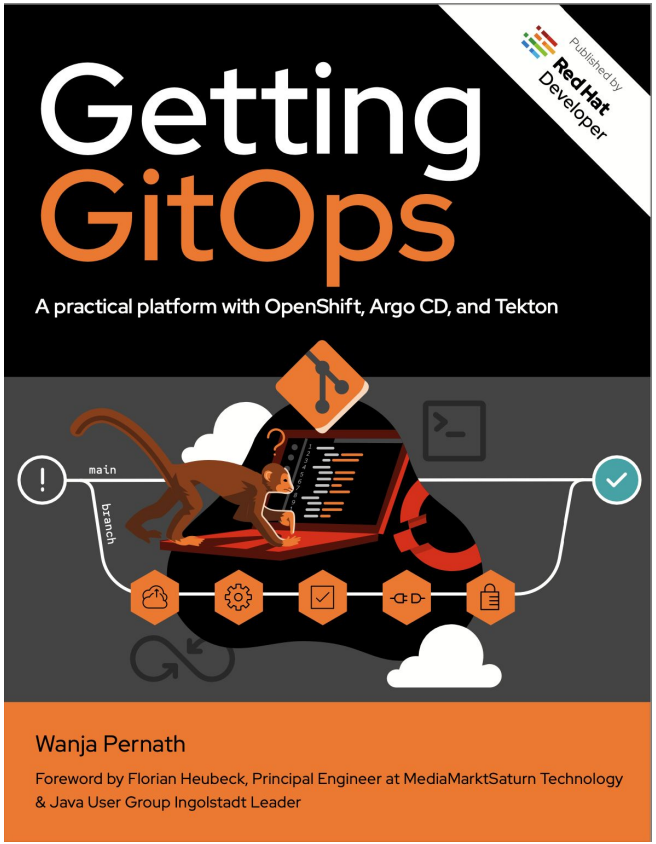
Technologies discussed:
Quarkus, Helm Charts, Kustomize, Tekton Pipelines, Kubernetes Operators, OpenShift Templates, ArgoCD, CI/CD, GitOps….

Download for free at:

https://developers.redhat.com/e-books/getting-gitops-practical-platform-openshift-argo-cd-and-tekton

Interview with full GitOps Demo:

https://www.youtube.com/watch?v=znMfVqAlRzY&ab_channel=OpenShift



Agenda

Agenda

- What is this all about
- Application Packaging with Kubernetes
- Tekton
- ArgoCD & GitOps

What is this all about?

Learning Goals

- Getting an overview of the whole OpenShift ecosystem for developers
- Learn about the benefits of OpenShift prior plain Kubernetes
- Learn how to effectively and efficiently demo OpenShift for developers
- Learn to use OpenShift and its ecosystem
- Learn to start coding with OpenShift in your preferred language
- Learn to do proper release management with OpenShift; understand the basics and how to include it into your release management process but also learn how to benefit from OpenShift Pipelines and when to use what
- Understanding and using Operators for releases
- Understand and make use of advanced OpenShift features like Istio and Serverless

The Use Case

Source Code for this workshop:
(Fork it, clone it, use it. It's open source!)

https://github.com/wpernath/quarkus-grumpycat

The Use Case

- In order to understand Quarkus and how great it integrates into all the necessary technologies, let's write a game. Let's call it **Quarkus GrumpyCat**
- It's a multi player game for up to 4 concurrent players
- Goal is to be last dog standing

The Use Case

The server is stateless and maintains the communication between the 4 players in a game

The Use Case


Energy:
100 %
00001280
25
50
50
50
50

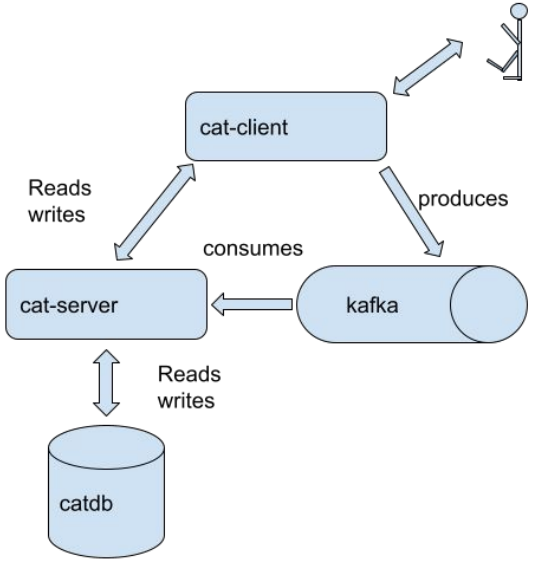

Energy:
100 %
00001600
25
46
50
49
50

The Use Case

- The client is fully written in JavaScript with a simple game framework called MelonJS (github.com/melonJS/melonJS)
- The client runs in any recent browser (Chrome, Safari, Firefox have been tested)
- The client communicates with the server via
  - REST API and
  - WebSocket
- But, this is a Quarkus workshop (or: a Kubernetes Native Workshop, so we try to ignore the client here, but trust me: Writing a JavaScript game is - pheww).

Architecture of the Game

- The user talks via Browser to the cat-client.
- The client talks to the cat-server
- The server has integrations for the database to store state and high scores
- The server also talks to Kafka



Getting GitOps

- We use the GitOps paradigma for this project, so we will also implement Tekton Pipelines and will use ArgoCD on OpenShift to do CI/CD
- At the end of this workshop you're going to have a blueprint for your own projects.