

DEVOXX
MOROCCO



Un opérateur Kubernetes en Java avec Quarkus

Horacio Gonzalez

2020-06-22



@Wildagsx



@LostInBrittany

Qui sommes-nous ?

Et qui est OVHcloud ?

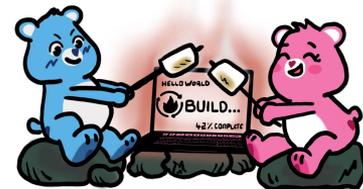


Horacio Gonzalez



@LostInBrittany

Espagnol perdu en Bretagne



Stéphane Philippart

- 🛒 Junior DevRel@OVHCloud 🥑 🦄 🎉
- 🛒 Co-créateur de TADx (meetups Agile, Dev, DevOps)

🐦 @wildagsx

📝 <https://philippart-s.github.io/blog>

🐙 <https://github.com/philippart-s/>

🌐 <https://www.linkedin.com/in/philippartstephane/>



TAD 



Web Cloud & Telecom



Private Cloud



Public Cloud



Storage



Network & Security



30 Data Centers
in 12 locations



34 Points of Presence
on a 20 TBPS Bandwidth Network



2200 Employees
worldwide



115K Private Cloud
VMS running



300K Public Cloud
instances running



380K Physical Servers
running in our data centers



1 Million+ Servers
produced since 1999



1.5 Million Customers
across 132 countries



3.8 Million Websites
hosting



1.5 Billion Euros Invested
since 2016



P.U.E. 1.09
Energy efficiency indicator



20+ Years in Business
Disrupting since 1999

- présentation des opérateurs ~ 15 mins / Stef (+all)
 - présentation générale (CRD / API / Controllers)
 - zoom sur les SDK / FWK
 - Ressources existantes
 - [Slides](#) Horacio
 - [Slides](#) Stef
 - Slides Sebi ?

Disclaimer



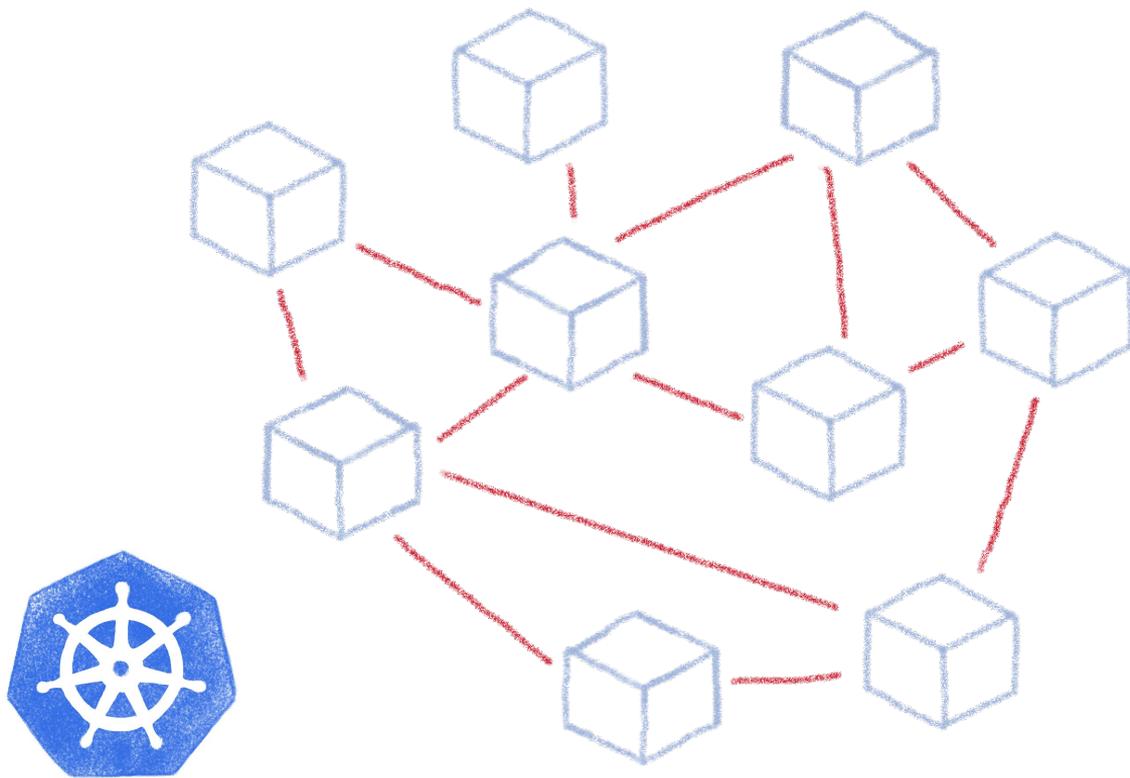
- Explications simplifiées pour que l'on parle le même langage
- Pas des Ops / Expert Kubernetes
- Développeurs

Mais c'est quoi un opérateur Kubernetes ?

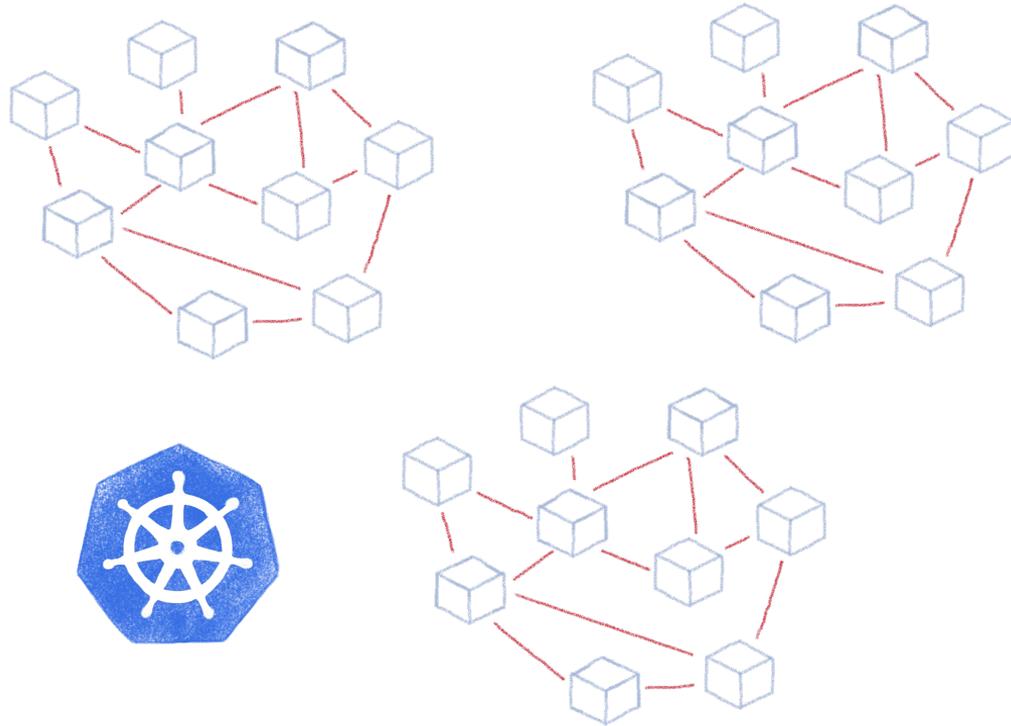
Comment ils aident à maîtriser
la complexité de Kubernetes



K8s aide à dompter les microservices



Mais on fait quoi des cas complexes ?



Ingress

Services

Deployments

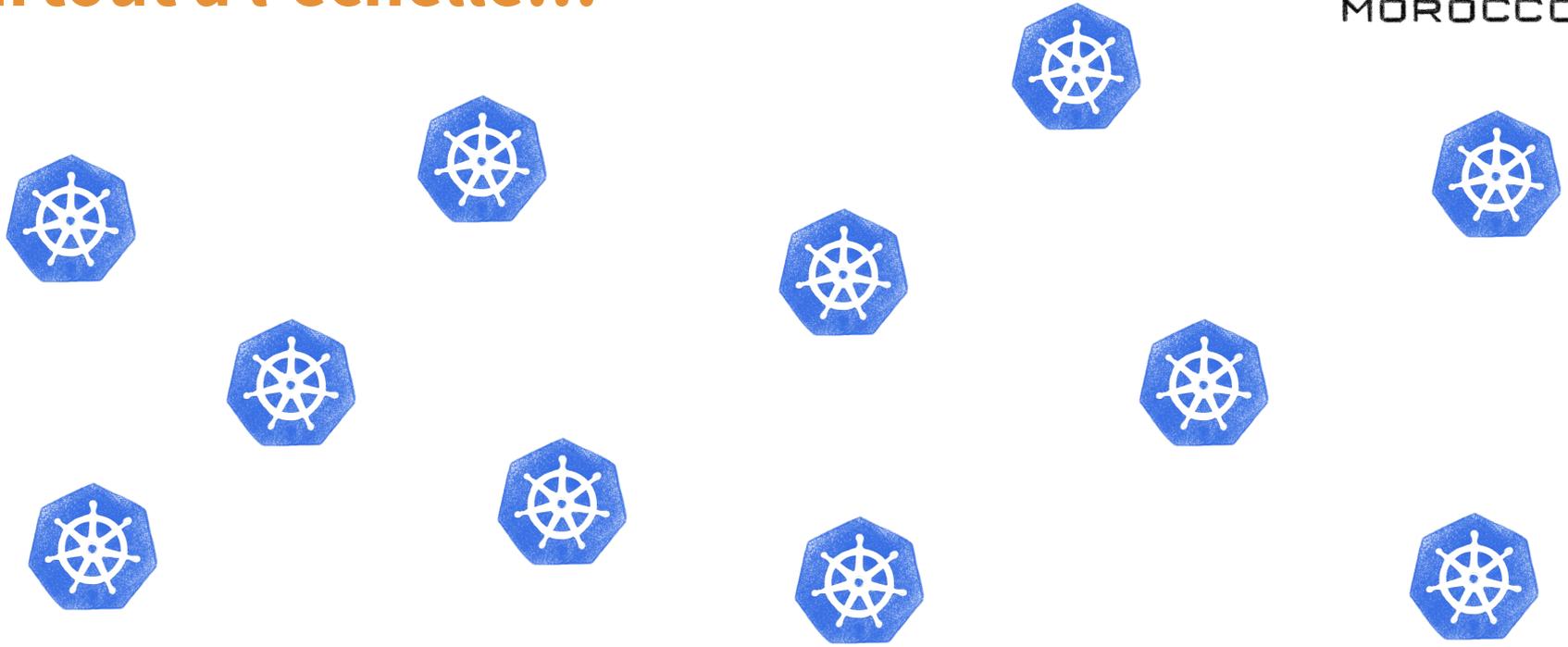
Pods

Sidecars

Replica Sets

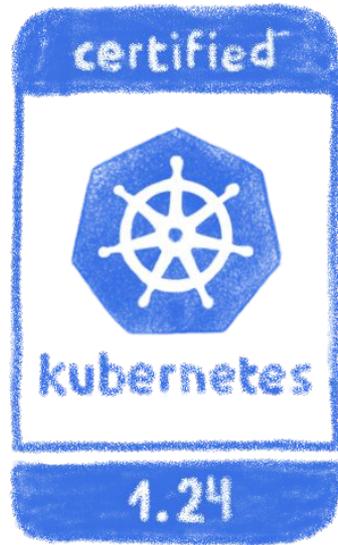
Stateful Sets

Surtout à l'échelle...



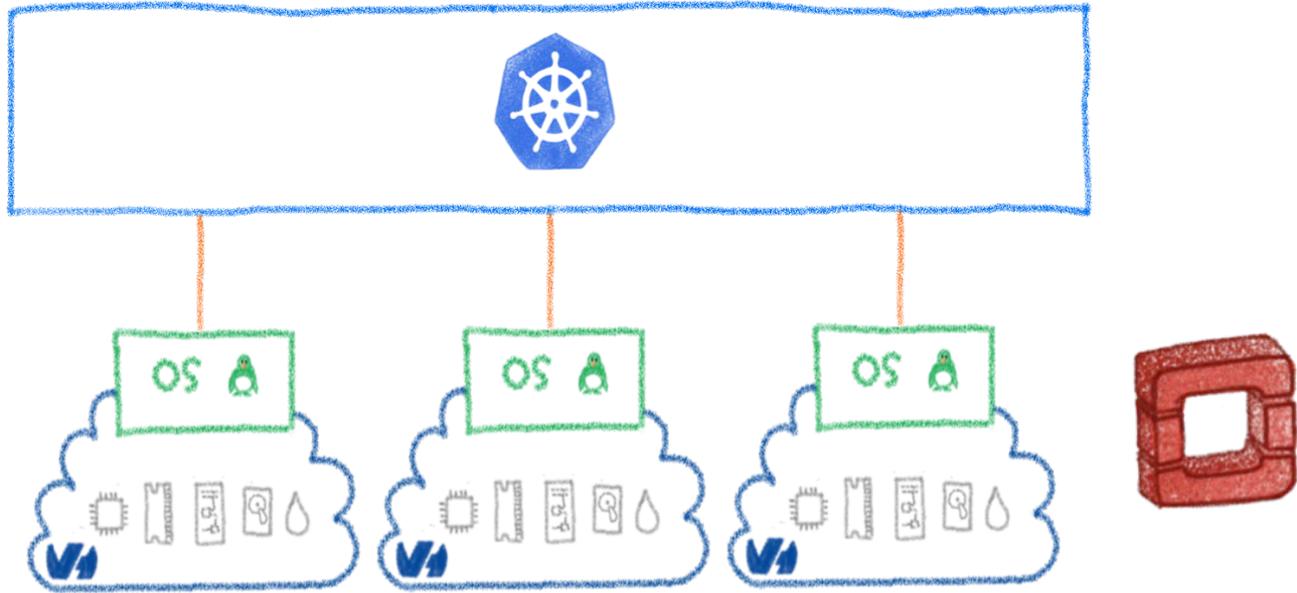
Lots of clusters with lots and lots of deployments

Et cela était juste notre cas

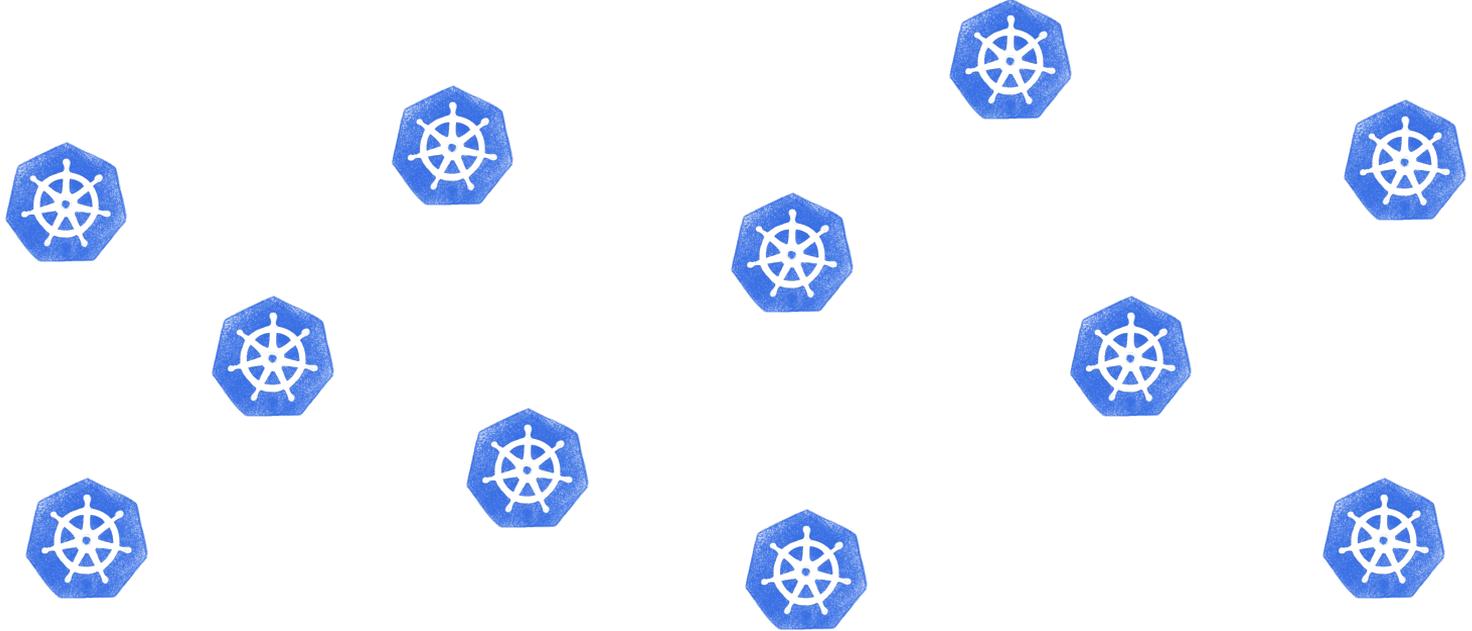


Nous utilisons Kubernetes mais nous opérons aussi
une plateforme Kubernetes managée

Bâtie sur notre Public Cloud basé sur Openstack

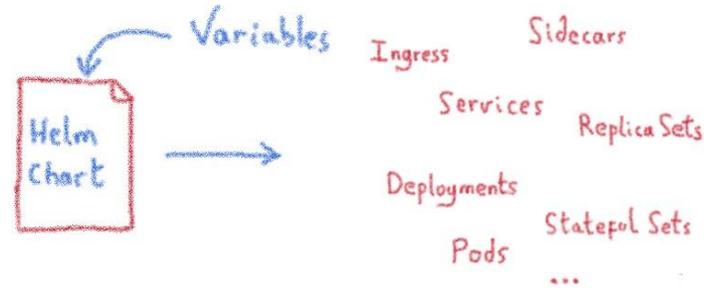


Nous avons besoin de dompter la complexité



Helm était une première réponse

A package manager for Kubernetes



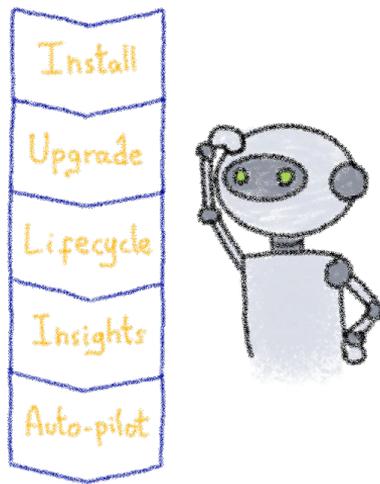
- Manage complexity 
- Easy upgrades 
- Simple sharing 
- Easy rollbacks 

Mais les chartes Helm ne font que la configuration



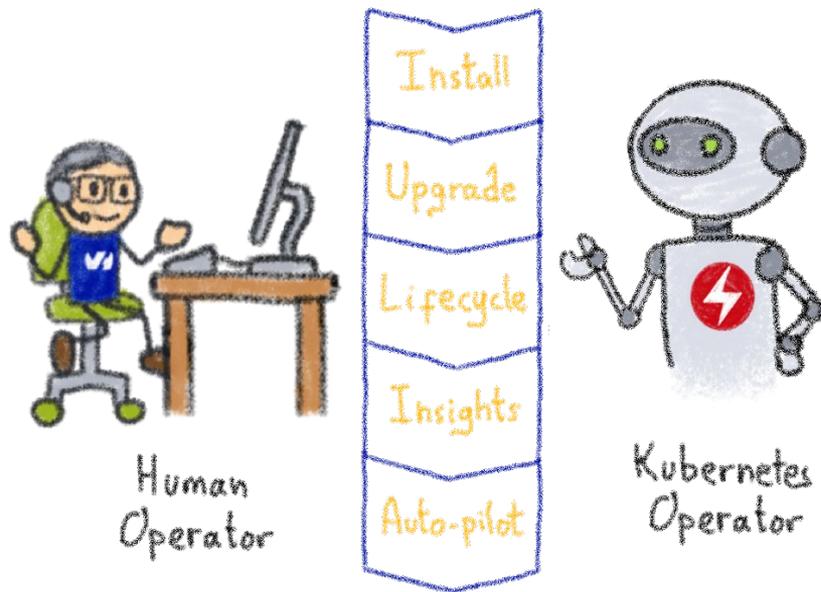
Opérer une plateforme c'est plus que
gérer les installs et les mises à jour

Kubernetes c'est surtout de l'automatisation



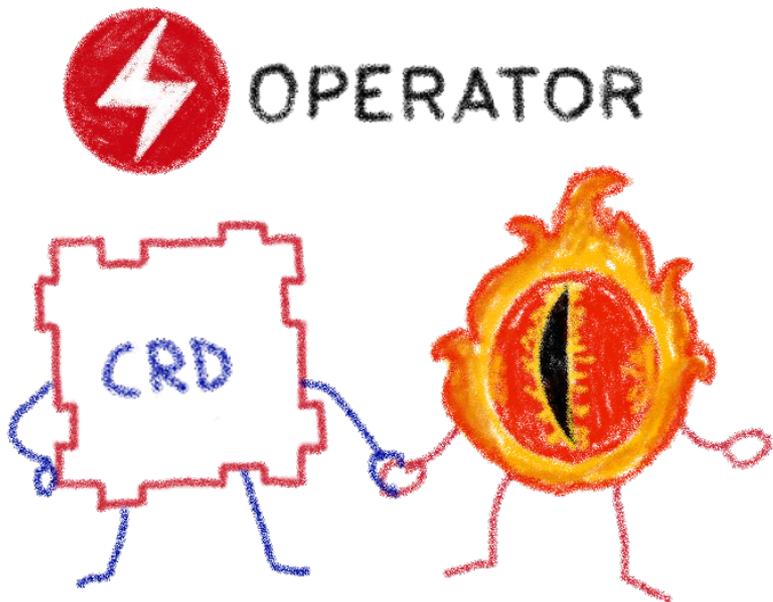
Comment automatise-t-on les opérateurs humains ?

Opérateurs Kubernetes



Une version Kubernetes de l'opérateur humain

Comment construit-on des opérateurs



Éléments basiques Kubernetes :
Contrôleurs & Ressources Customisées

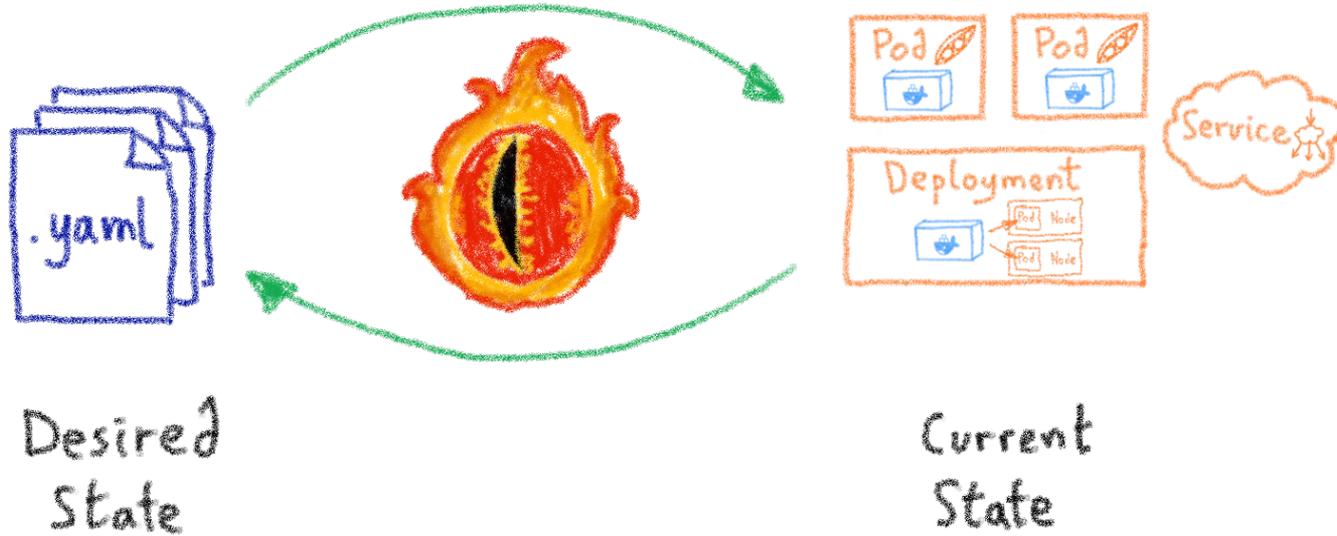
Kubernetes Controllers

Toujours un oeil sur les ressources



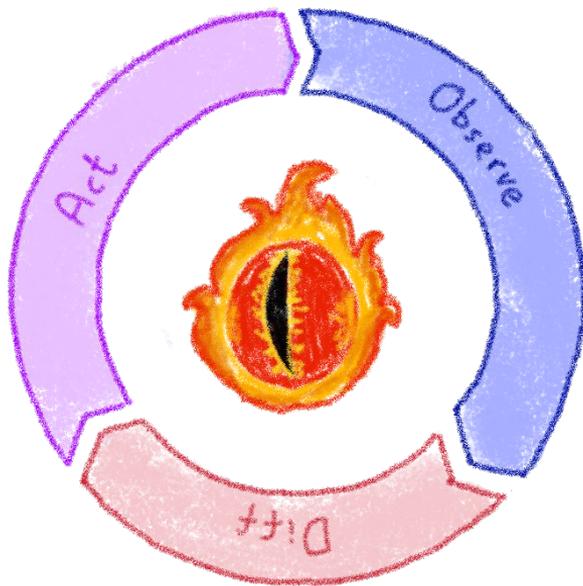
* Contrôleurs Kubernetes

Une boucle de contrôle



Surveille l'état du cluster et fait les changements pour le rapprocher de l'état souhaité

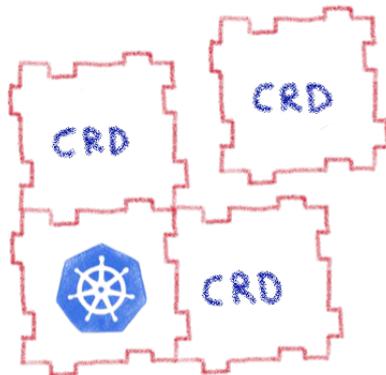
Une boucle de réconciliation



Cherche à réconcilier l'état présent avec l'état souhaité

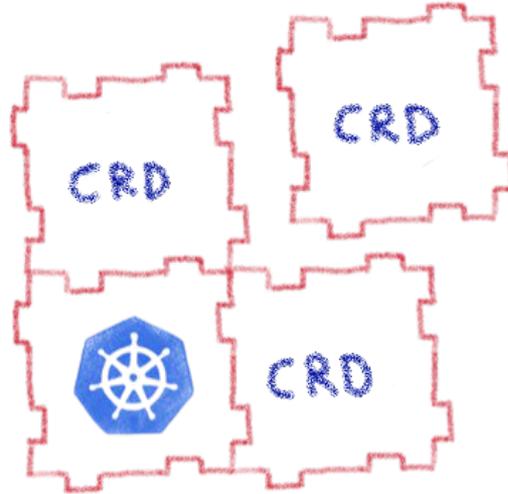
Custom Resource Definitions*

On étend l'API Kubernetes



* Définitions de ressources customisées

On étend l'API Kubernetes



En définissant des nouveaux types de ressources

Custom Resource Definition et Custom Resource

→ Custom Resource Definition = CRD

→ “Sorte” de schéma / classe pour les objets Kubernetes

→ Custom Resource : CR

→ Sorte “d’objet instanciant la classe (CRD)”

```
1 apiVersion: apiextensions.k8s.io/v1
2 kind: CustomResourceDefinition
3 metadata:
4   name: helloworldcustomresources.fr.wilda
5 spec:
6   group: fr.wilda
7   names:
8     kind: HelloWorldCustomResource
9     plural: helloworldcustomresources
10    shortNames:
11      - hw
12    singular: helloworldcustomresource
13  scope: Namespaced
14  versions:
15    - name: v1
16      schema:
17        openAPIV3Schema:
18          properties:
19            spec:
20              properties:
21                → name:
22                  type: string
23                type: object
24              status:
25                type: object
26              type: object
27  served: true
28  storage: true
```

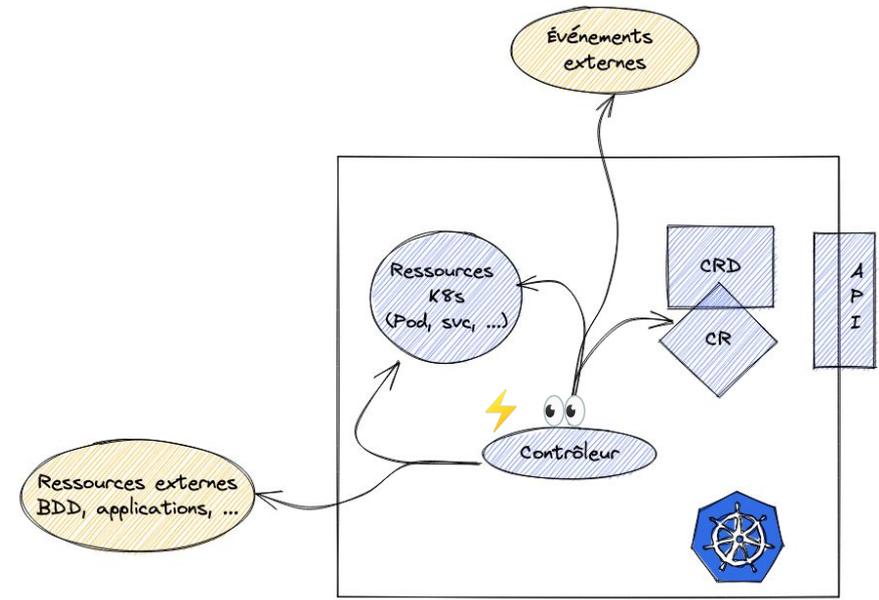
```
1 apiVersion: "fr.wilda/v1"
2 kind: HelloWorldCustomResource
3 metadata:
4   name: hello-world
5   namespace: test-hw-crd
6 spec:
7   name: stef
```

Opérateur Kubernetes

Automatisation des opérations



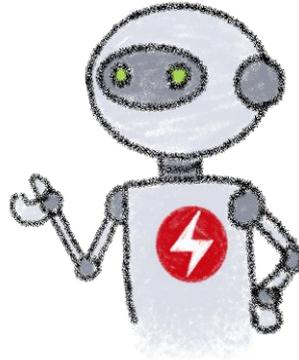
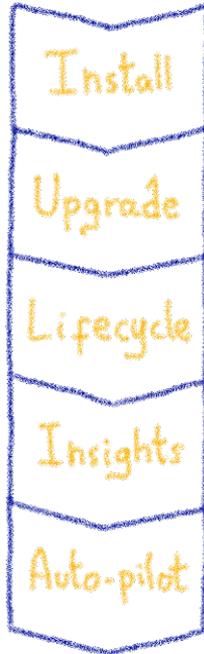
- API + CRD (et CR) + Contrôleur
- Gère une et une seule application (déployée ou non dans Kubernetes)
- Essaie de maintenir l'état de l'application avec ce qui est déclaré dans la CR



C'est quoi donc un opérateur Kubernetes ?



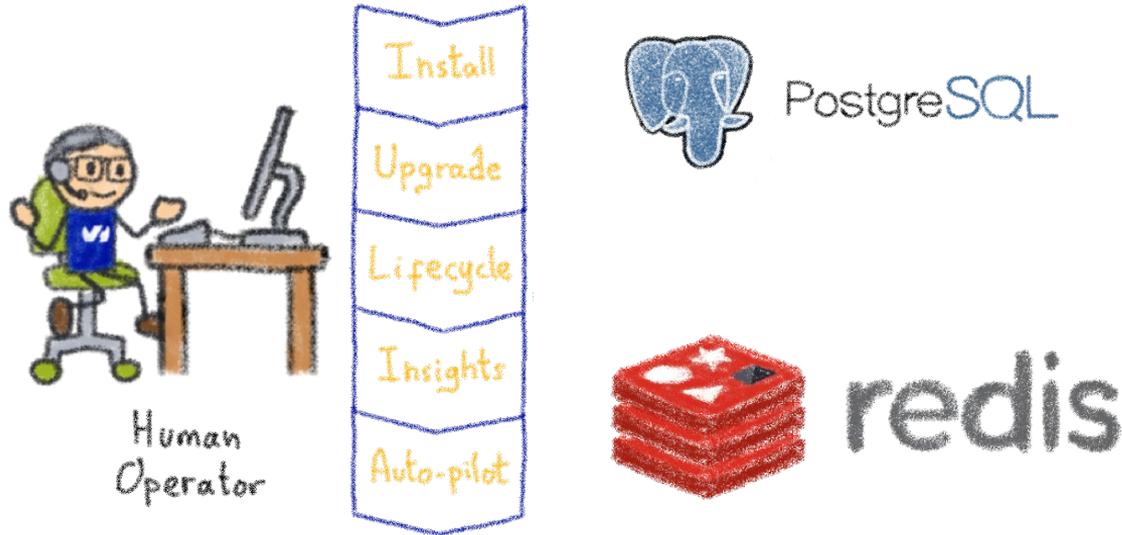
Human
Operator



Kubernetes
Operator

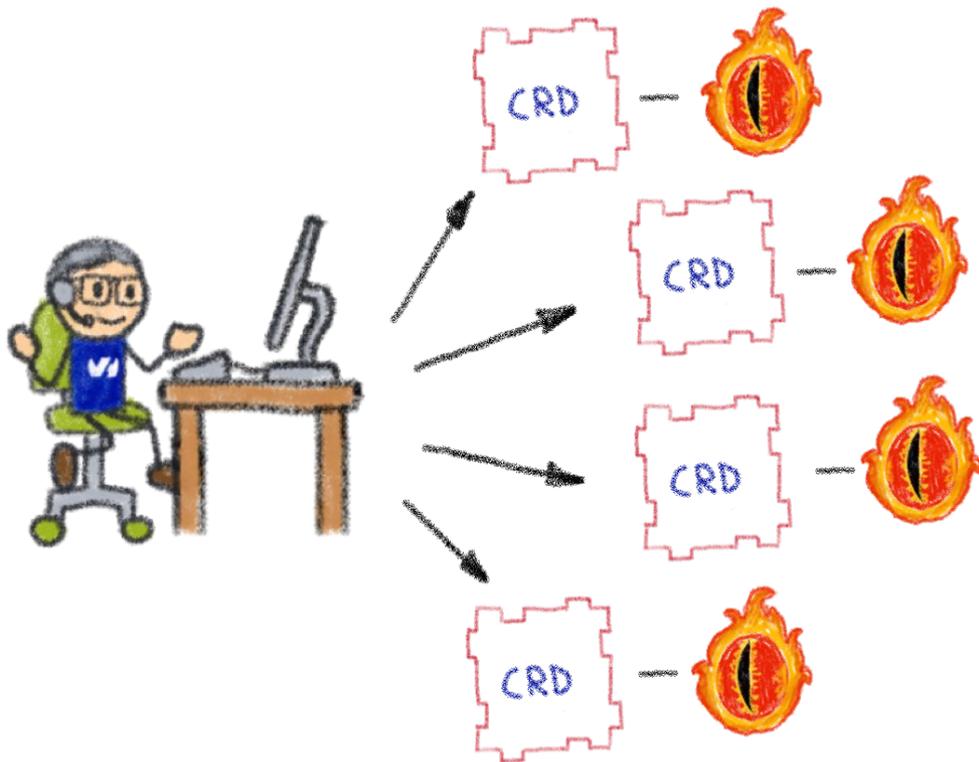
An Operator represents human operational knowledge in software to reliably manage an application

Exemple : des bases de données



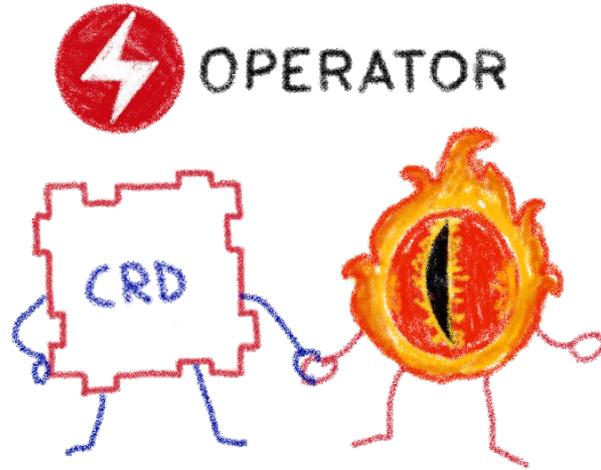
Des choses comme ajouter une instance à un pool,
faire des backups, du sharding...

Ils encodent la connaissance dans des CRDs et des contrôleurs



Encapsulating
business logic in
CRDs & Controllers

Des contrôleurs customisés pour des ressources customisées

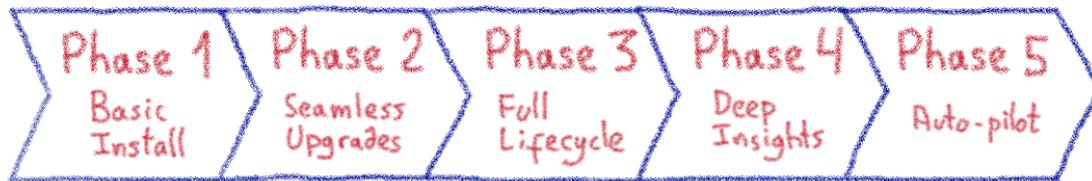
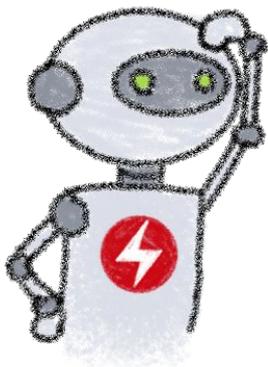


Ils implémentent et gèrent des Custom Resources avec une logique de réconciliation customisée

Le modèle de capacités des opérateurs



OPERATOR
CAPABILITY MODEL

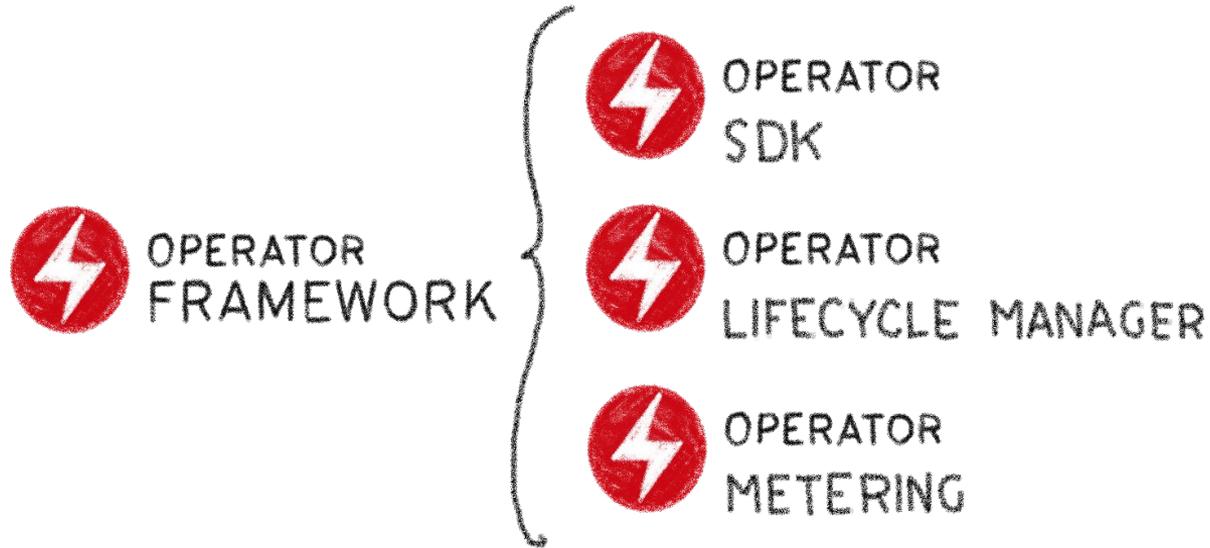


Mesurer la maturité de l'opérateur

👉 One more thing !

- ➔ Un opérateur n'est qu'un Pod qui exécute une application
- ➔ Il peut donc être développé avec n'importe quel langage
- ➔ Il suffit d'appeler les API Kubernetes ou d'utiliser un client compatible

Le Framework Opérateur



Framework open source pour accélérer le développement d'un opérateur

L'Operator SDK

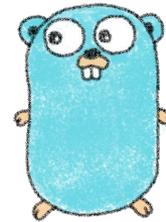


OPERATOR
SDK

BUILD
TEST
ITERATE



ANSIBLE

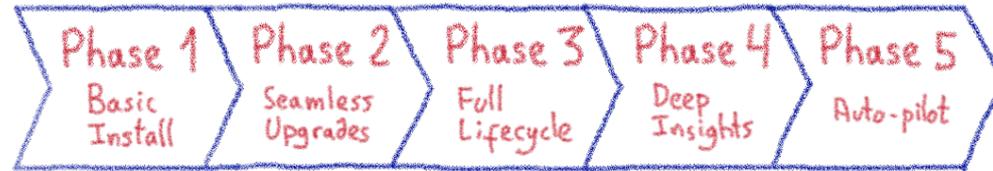
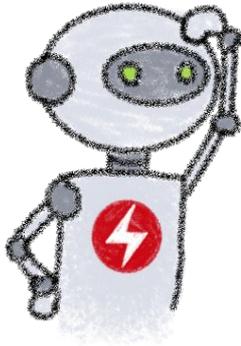


Trois façons différentes de construire des opérateurs

L'Operator SDK et le Modèle de Capacité

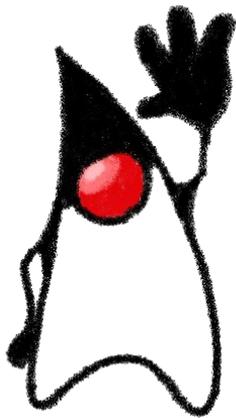


OPERATOR
CAPABILITY MODEL



Mais je suis un dev Java!

Puis-je coder un opérateur ? Facilement ?





Alors en java ?



The screenshot shows the homepage of the Java Operator SDK. The header features a navigation menu with links for HOME, DOCS, CODE OF CONDUCT, and RELEASES, along with social media icons for GitHub and Discord. The main content area has a large orange background with the Java logo and the text "[JAVA OPERATOR SDK]". Below this are three buttons: "GET STARTED", "CONTRIBUTE", and "DISCORD CHANNEL". The bottom section is titled "Sponsored by:" and features logos for Container Solutions and Red Hat.



Java Operator SDK

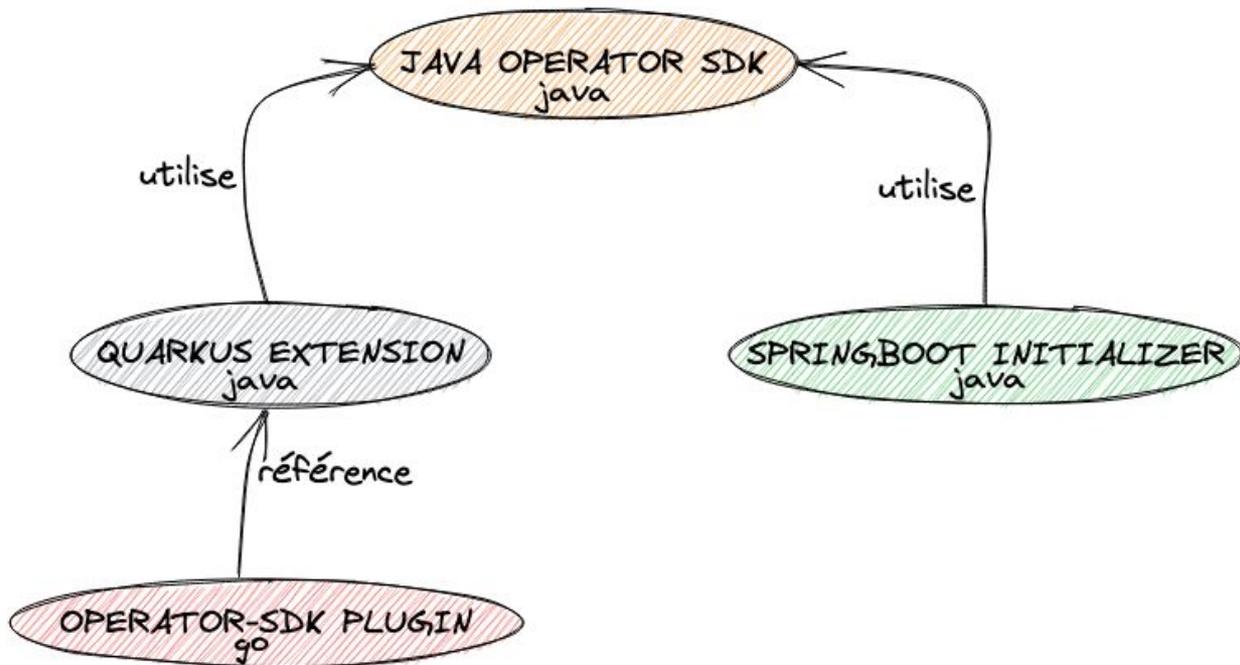


✨ Les principales fonctionnalités du Framework

- Client Kubernetes Fabric 8 pour manipuler les ressources Kubernetes
- Création (scaffolding) du squelette applicatif pour le mode Quarkus
- Facilitation de la génération des CRD et du contrôleur (annotations)
- Gestion de la boucle de réconciliation
- Exécution locale avec debugger possible
- Intégration avec Spring Boot et Quarkus
- Packaging de l'opérateur simplifié



Les différents composants



Quarkus

**SUPERSONIC /
SUBATOMIC /
JAVA**



Cloud Computing



MICROPROFILE™

JakartaEE



Rapide au démarrage



Empreinte mémoire raisonnable

DEVOX
MAROCCO



Build first





Faciliter le développement





JVM & GraalVM



Extensions



Command Line Interface

```
root@ubuntu: ~$ |
```

Pourquoi ajouter Gitpod ?

Parce que nous voulons vous faciliter la vie !



Laissez-moi vous raconter une histoire

Ton prochain projet

From: Boss <boss@mycompany.com>

To: Dev <dev@mycompany.com>

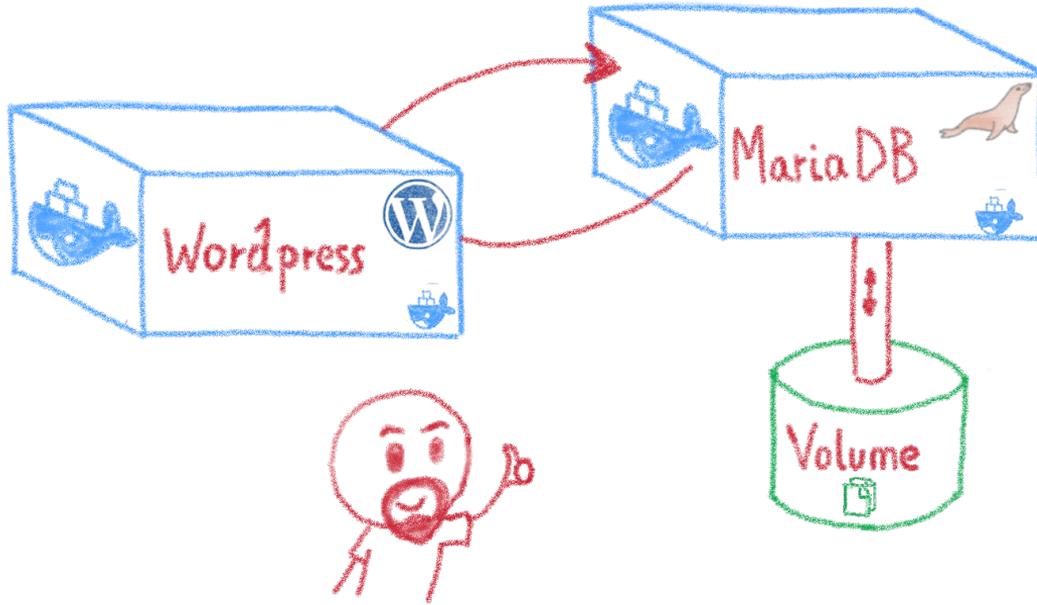
Coucou Dev !

Lundi prochain, vous commencez un nouveau projet avec Big Customer. Vous serez dans une équipe construisant un système avec une architecture distribuée, utilisant le stockage Redis & MariaDB, des applications en Go, Python et Java, un bus événementiel RabbitMQ, ...

Et voilà votre nouvel ordinateur portable, un bon cette fois, il dispose de 8 Go de RAM !

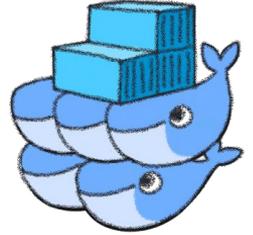
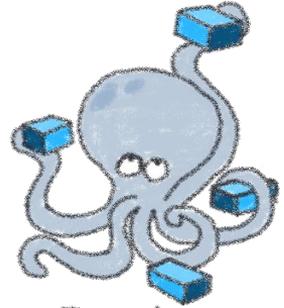
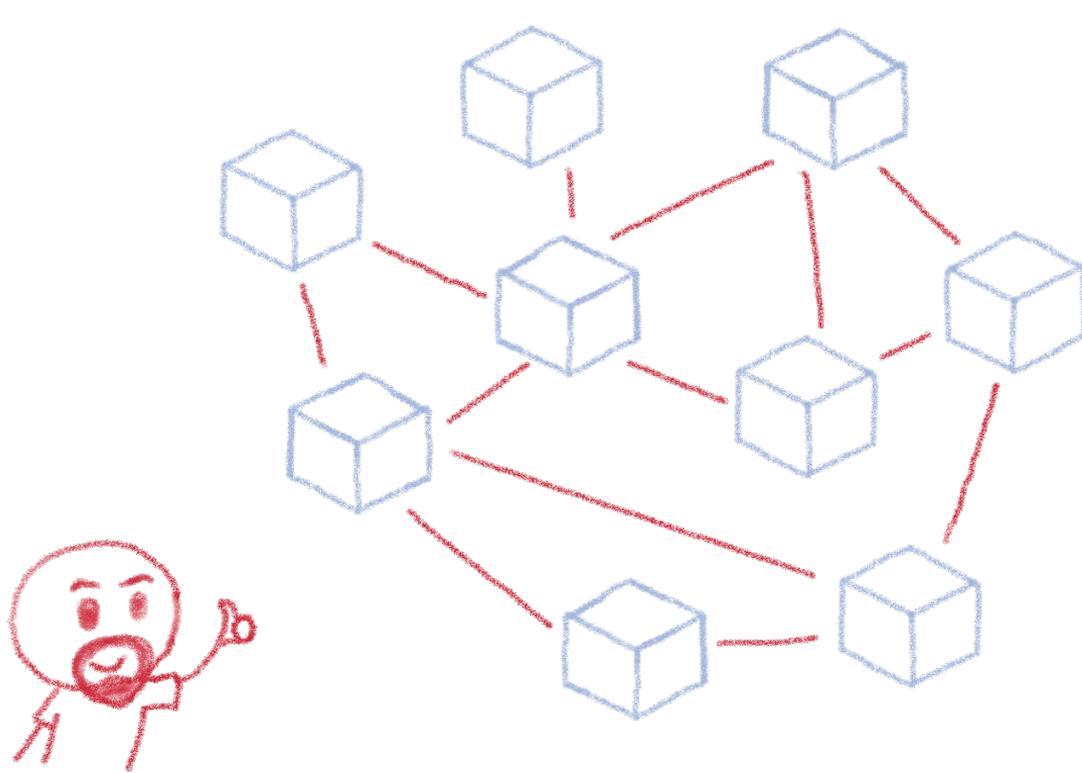


Les conteneurs ont changé la donne

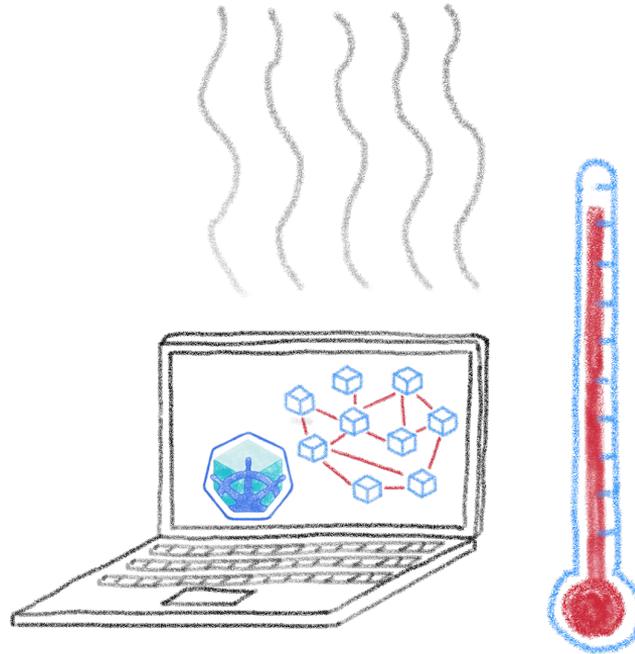


Utiliser la même architecture sur le laptop de dev et en prod

Même pour des architectures complexes

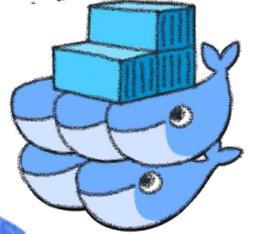
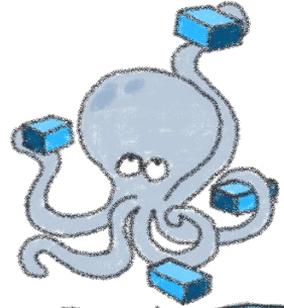
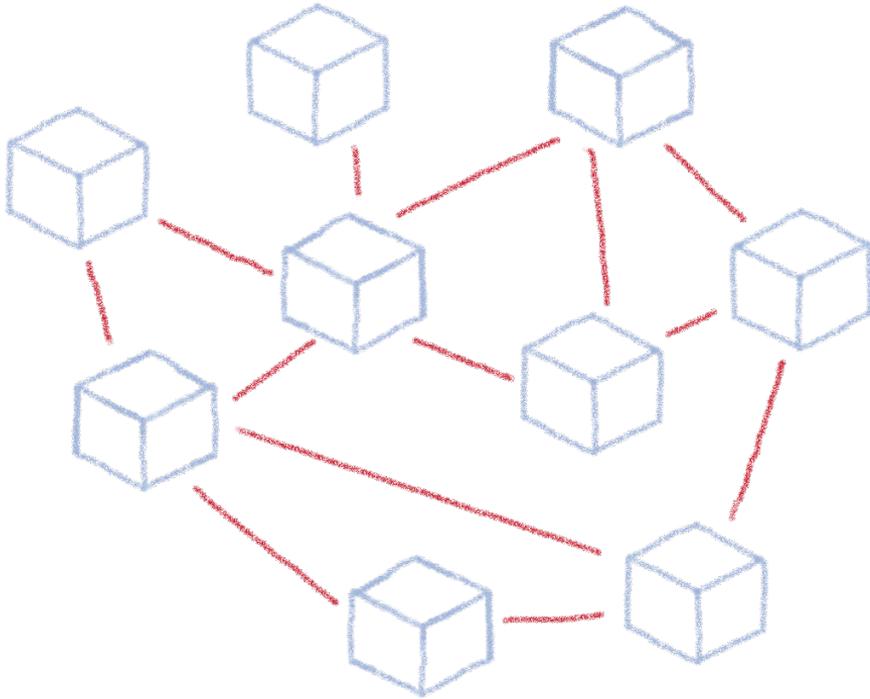


Mais un laptop reste un laptop



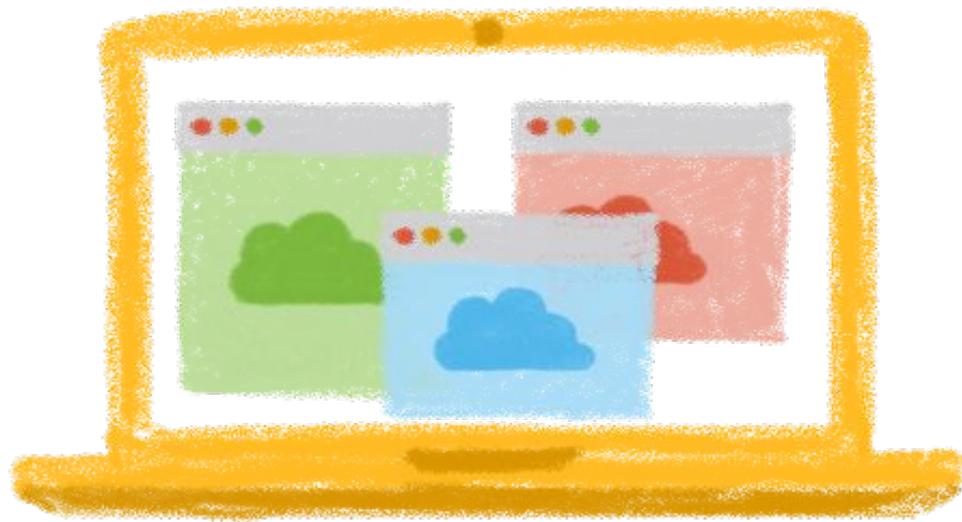
Ces containers demandent beaucoup de RAM et CPU

Sans parler du temps de mise en route



L'installation et la configuration ne sont pas triviales...

Dans l'idéal, je voudrais un IDEaaS



Integrated Development Environment as a Service

Gitpod est un IDEaaS open source

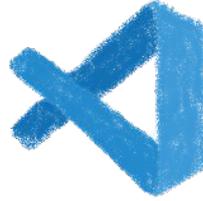


Environnements de dev automatisés et éphémères sur le Web

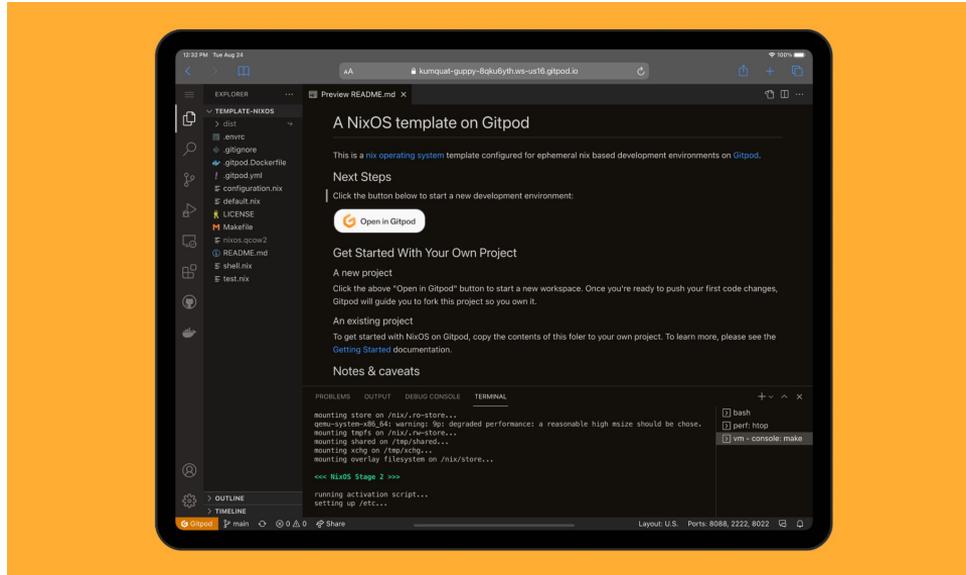
Un IDE complet dans le navigateur

Un IDE sur le Web 

- Eclipse Theia → VS Code
- Remote
- Workspaces



Il est plus qu'un IDE

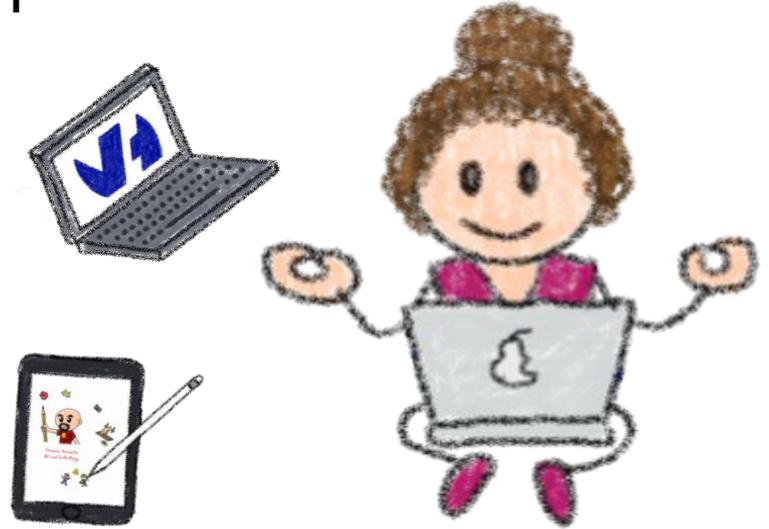


Un IDE n'est qu'un élément constitutif de
un environnement de dev basé sur le cloud

Sans avoir besoin d'un laptop surpuissant



- Low cost ultraportable laptop
- Chromebook
- iPad / Tablette Android*



* L'expérience est encore un peu décevante dans certains cas...



Les projets Gitpod se basent sur Git

Il peut se connecter sur :

- N'importe quelle version de Gitpod:
 - GitLab
 - GitHub
 - BitBucket
- Versions on-premises ou open source
 - Votre propre dépôt Git privé



Gitpod

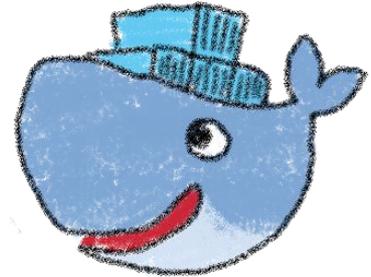
Gitpod - comment y accéder ?

- Gitpod.io ou Gitpod self hosted
- À partir d'un dépôt Git
 - Ajout de `https://gitpod.io/#`
- Extension navigateur
- Intégration plus poussée sur Gitlab



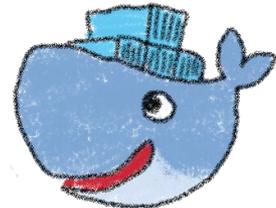
Anatomie d'un projet Gitpod

- Une image  Docker par défaut :
`gitpod/workspace-full`
- Deux fichiers  de configuration :
`.gitpod.dockerfile`
`.gitpod.yml`



Projects & Workspaces

- Un workspace est un environnement  Docker complet
- Des workspaces multiples pour chaque projet
 - Par exemple pour des branches ou des commits spécifiques
- Les workspaces sont peuvent être partagés
- Communication possible entre workspaces



Combien ça coûte ?

- SaaS version



Basic Free	Personal €8 Per User/Month	Professional €23 Per User/Month	Unleashed €35 Per User/Month
<ul style="list-style-type: none">✓ 50 hours/month✓ Private & Public Repos✓ Unlimited Prebuilds✓ Shared Workspaces✓ Snapshots	<ul style="list-style-type: none">✓ 100 hours/month✓ 4 Parallel Workspaces✓ Unlimited Prebuilds✓ Shared Workspaces✓ Snapshots✓ 30min Timeout	<ul style="list-style-type: none">✓ All in Personal✓ 8 Parallel Workspaces✓ Unlimited Hours✓ Teams	<ul style="list-style-type: none">✓ All in Professional✓ 16 Parallel Workspaces✓ 1hr Timeout✓ 3hr Timeout boost
Try Now	Buy Now	Buy Now	Buy Now

- Free Professional Open Source plan: no limit (public projects)
- Self hosted (K8S) Open Source (Free) et Pro (29€)



Gitpod - Codespace ?

- Open Source
- Ressources plus généreuses pour Gitpod
- Possibilité de l'héberger soi-même
- Gitpod est aussi utilisable avec Gitlab et Bitbucket
- Les prebuild n'existent pas pour CodeSpace

Démo time !

D'autres use-cases

Des autres exemples avec Gitpod



La nouvelle équipe arrive lundi

De : Patron <boss@mycompany.com>

À : Chef de projet <pm@mycompany.com>



Salut chef de projet !

Votre équipe Shiny Project arrivera lundi prochain. Vous aurez 15 développeurs dans l'équipe, et ils doivent être productifs dès le premier jour. Un lot de 15 ordinateurs portables et 3 serveurs a été envoyé par DHL à votre bureau, veuillez vous assurer qu'ils sont installés et fonctionnent d'ici lundi matin.

Passé un bon weekend!

Le client X veut tester la fonctionnalité

De : Patron <boss@mycompany.com>

À : Chef de projet <pm@mycompany.com>



Salut chef de projet !

Le client X a demandé de tester la nouvelle fonctionnalité sur laquelle travaille votre équipe. J'ai pointé vers le dépôt Git, mais ils ne sont pas intéressants pour lire du code, ils veulent le tester !

Je leur ai dit que vous leur enverriez un moyen de tester d'ici lundi.

Passé un bon weekend!

Cours sur OVHcloud Managed Kubernetes

De : Partenaire <training@partner.com>

À : Horacio Gonzalez <horacio.gonzalez@ovhcloud.com>



Salut Horacio !

Comme expliqué par téléphone, nous aimerions que vous suiviez une formation OVHcloud Managed Kubernetes pour notre organisation. Nous devons être entièrement basés sur le Web, les participants ne pourront rien installer sur leurs ordinateurs (même pas kubectl, helm ou tout autre outil).

Les participants n'auront que leurs navigateurs et leurs identifiants API OVHcloud.

Une plateforme prête à utiliser pour K8s

☰ README.md ✎

Gitpod for K8s

A working Gitpod setup for managing Kubernetes clusters. The workspace includes Kubectl, Helm & Kustomize.

To open the workspace, simply click on the *Open in Gitpod* button, or use [this link](#).

 Open in Gitpod

Configuring Kubeconfig

To administrate your Kubernetes cluster from Gitpod, you need to add your Kubeconfig to your Gitpod workspace. There are two ways to do it:

1- Copying the into your workspace

Copy your kubeconfig into a `~/.kube/config` file in your Gitpod workspace.

Releases

No releases published
[Create a new release](#)

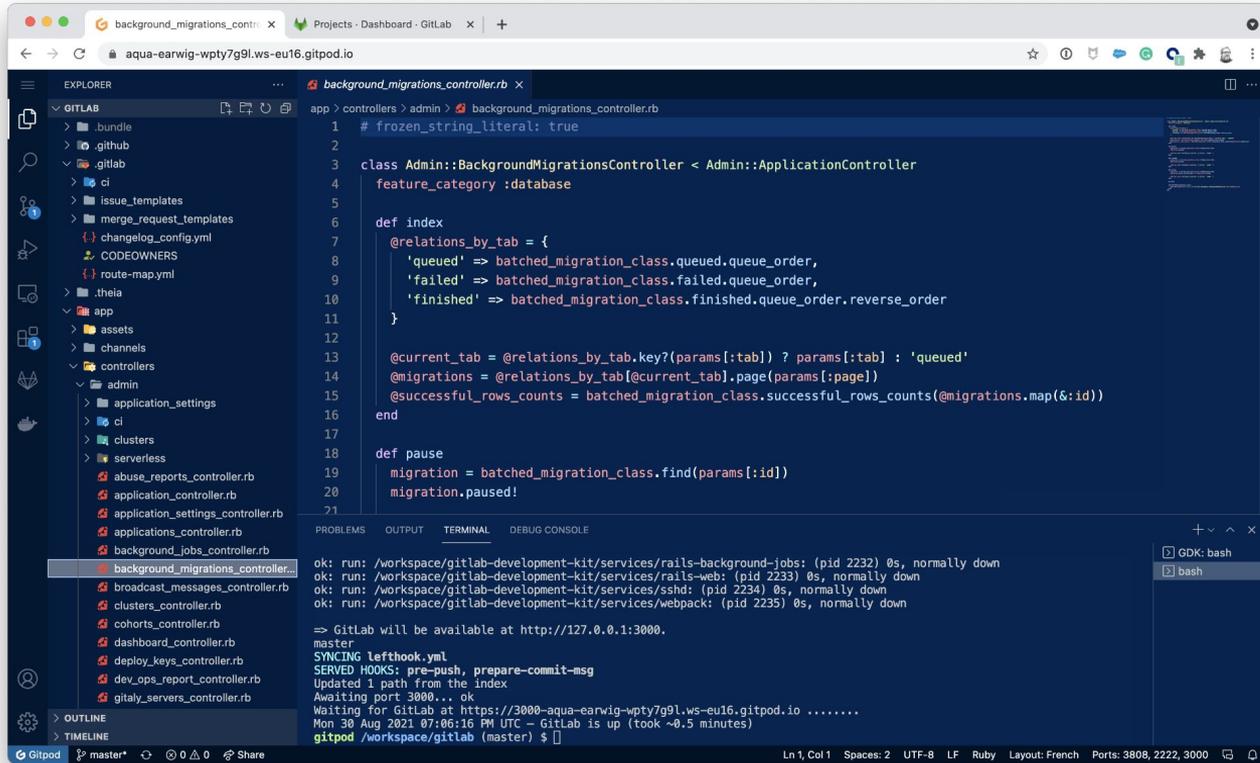
Packages

No packages published
[Publish your first package](#)

[gitpod4k8s](#)



Codez pour les nuages, dans les nuages



```
# frozen_string_literal: true

class Admin::BackgroundMigrationsController < Admin::ApplicationController
  feature_category :database

  def index
    @relations_by_tab = {
      'queued' => batched_migration_class.queued.queue_order,
      'failed' => batched_migration_class.failed.queue_order,
      'finished' => batched_migration_class.finished.queue_order.reverse_order
    }

    @current_tab = @relations_by_tab.key?(params[:tab]) ? params[:tab] : 'queued'
    @migrations = @relations_by_tab[@current_tab].page(params[:page])
    @successful_rows_counts = batched_migration_class.successful_rows_counts(@migrations.map(&:id))
  end

  def pause
    migration = batched_migration_class.find(params[:id])
    migration.paused!
  end
end
```

ok: run: /workspace/gitlab-development-kit/services/rails-background-jobs: (pid 2232) 0s, normally down
ok: run: /workspace/gitlab-development-kit/services/rails-web: (pid 2233) 0s, normally down
ok: run: /workspace/gitlab-development-kit/services/ssh: (pid 2234) 0s, normally down
ok: run: /workspace/gitlab-development-kit/services/webpack: (pid 2235) 0s, normally down

=> GitLab will be available at http://127.0.0.1:3000.
master
SYNCING lefthook.yml
SERVED HOOKS: pre-push, prepare-commit-msg
Updated 1 path from the index
Awaiting port 3000... ok
Waiting for GitLab at https://3000-aqua-earwig-wpty7g9l.ws-eu16.gitpod.io
Mon 30 Aug 2021 07:06:16 PM UTC - GitLab is up (took ~0.5 minutes)
gitpod /workspace/gitlab (master) \$ []

Vous avez un workshop à préparer...

Félicitations Horacio

De : Big Conference <organization@bigconference.com>

À : Horacio Gonzalez <horacio.gonzalez@gmail.com>



Salut Horacio !

Nous sommes heureux de vous annoncer que votre atelier "Toi aussi apprends à développer un opérateur Kubernetes en Java avec Quarkus" a été accepté à la Big Conference 2022.

Vous aurez 20-50 participants physiquement dans la salle. L'intégration est essentielle dans cette configuration, assurez-vous que les participants puissent facilement créer un environnement de travail en quelques minutes, même avec un mauvais réseau ou des ordinateurs pas tout récents.



WORKSHOP



- partie pratique ~ 1h20 / all
 - pré-requis
 - evt gitpod ok
 - accès à un kube ok → est-ce que on (OVHcloud) on fournit un evt sandbox le temps du workshop ou autre solution ?
 - bootstrap du projet
 - CLI Quarkus (en utilisant le FWK core) vs sdk-operator CLI ?
 - hello, world !
 - prise en main de FWK
 - développement simple d'un exemple :
 - affiche hello, World <name> lors de la création d'une CR
 - affiche Goodbye <name> lors de la suppression de la CR
 - permet de voir
 - création de CRD / Controlleur
 - maj auto de la CRD
 - mode développeur Quarkus + mode debug GitPod (VSCode)
 - packaging / déploiement dans Kube
 - cas plus réel : déploiement / suppression d'une application Quarkus sur release dans GitHub
 - pré-requi :
 - une application Quarkus (<https://github.com/philippart-s/hello-world-from-quarkus>)
 - la mettre à dispo dans une registry accessible (OVHcloud / Docker Hub)
 - gestion de la détection d'evt : release GH

Préparons le poste de travail

Mettons en oeuvre Gitpod





Quelques configurations utiles



```
% wget http://51.210.145.31:8080/kubeconfig/<XXX>  
% mv <xxx> ./kubeconfig.yml  
% export KUBECONFIG=./kubeconfig.yml  
% kubectl cluster-info
```

Kubeconfig: `azzzsq<XXX>`

Gitpod

Idées de ce qu'il faut dans le workspace

- CLI : Quarkus, kubectl, kubectx, Operator-SDK
- Softs : Java 11, Maven, k9s, Kubernetes ? (à voir si on peut pas se créer une instance chez OVHcloud pour le workshop)
- plugins : java redhat + quarkus
-

-> Gros cluster kube chez OVHcloud avec un NS par participant avec un générateur de kubeconfig par participant

Hello, World ! With Quarkus

Un projet simple avec Quarkus

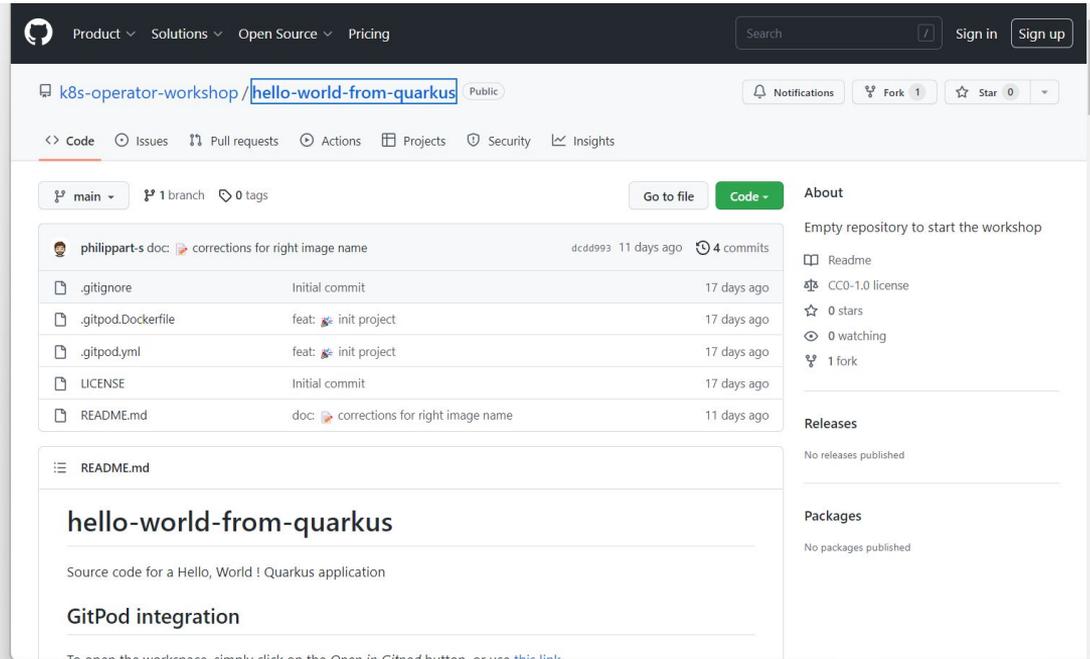


👋 Hello, World ! 🌍 With Quarkus

- 👉 Prise en main de la CLI Quarkus pour initialiser le projet
- 👉 Création d'un service "Hello, World !"
- 👉 Création du test unitaire
- 👉 Packaging
- 👉 Release GitHub et ajout de l'archive (JAR)

 <https://github.com/k8s-operator-workshop/hello-world-from-quarkus>

Hello, World ! With Quarkus



The screenshot shows the GitHub interface for the repository 'k8s-operator-workshop/hello-world-from-quarkus'. The repository is public and has 1 fork and 0 stars. The main branch is selected. The file list includes:

File Name	Commit Message	Commit Date
philippart-s doc	corrections for right image name	11 days ago
.gitignore	Initial commit	17 days ago
.gitpod.Dockerfile	feat: init project	17 days ago
.gitpod.yml	feat: init project	17 days ago
LICENSE	Initial commit	17 days ago
README.md	doc: corrections for right image name	11 days ago

The README.md file content is visible below the file list:

```
hello-world-from-quarkus

Source code for a Hello, World! Quarkus application

GitPod integration

To open the workspace, simply click on the Open in Gitpod button, or use this link:
```

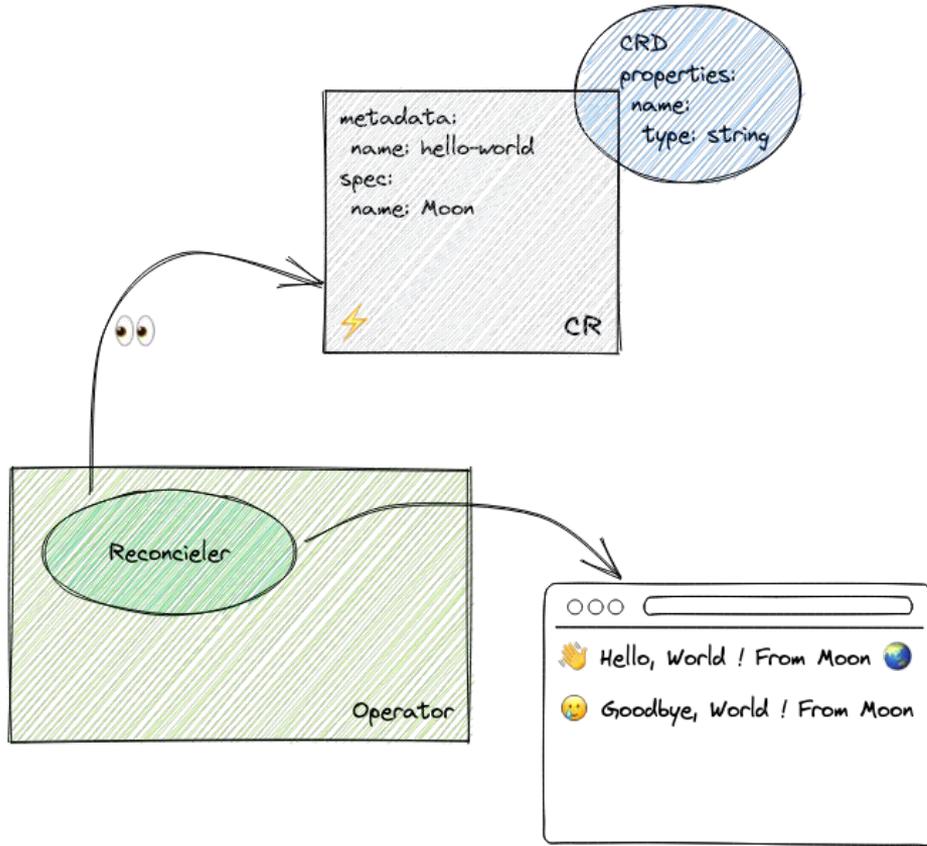
 <https://github.com/k8s-operator-workshop/hello-world-from-quarkus>

Operator Hello World !

Notre premier opérateur Kubernetes



👋 Operator Hello World ! 🌍



👋 Operator Hello World ! 🌍

- 👉 Prise en main de la CLI pour initialiser le projet
- 👉 Utilisation de la CLI pour créer l'API / CRD
- 👉 Modification de la CRD (classe Java)
- 👉 Modification du *reconcilier* pour prise en compte des évènements sur la CR
- 👉 Tests avec création / suppression d'une CR

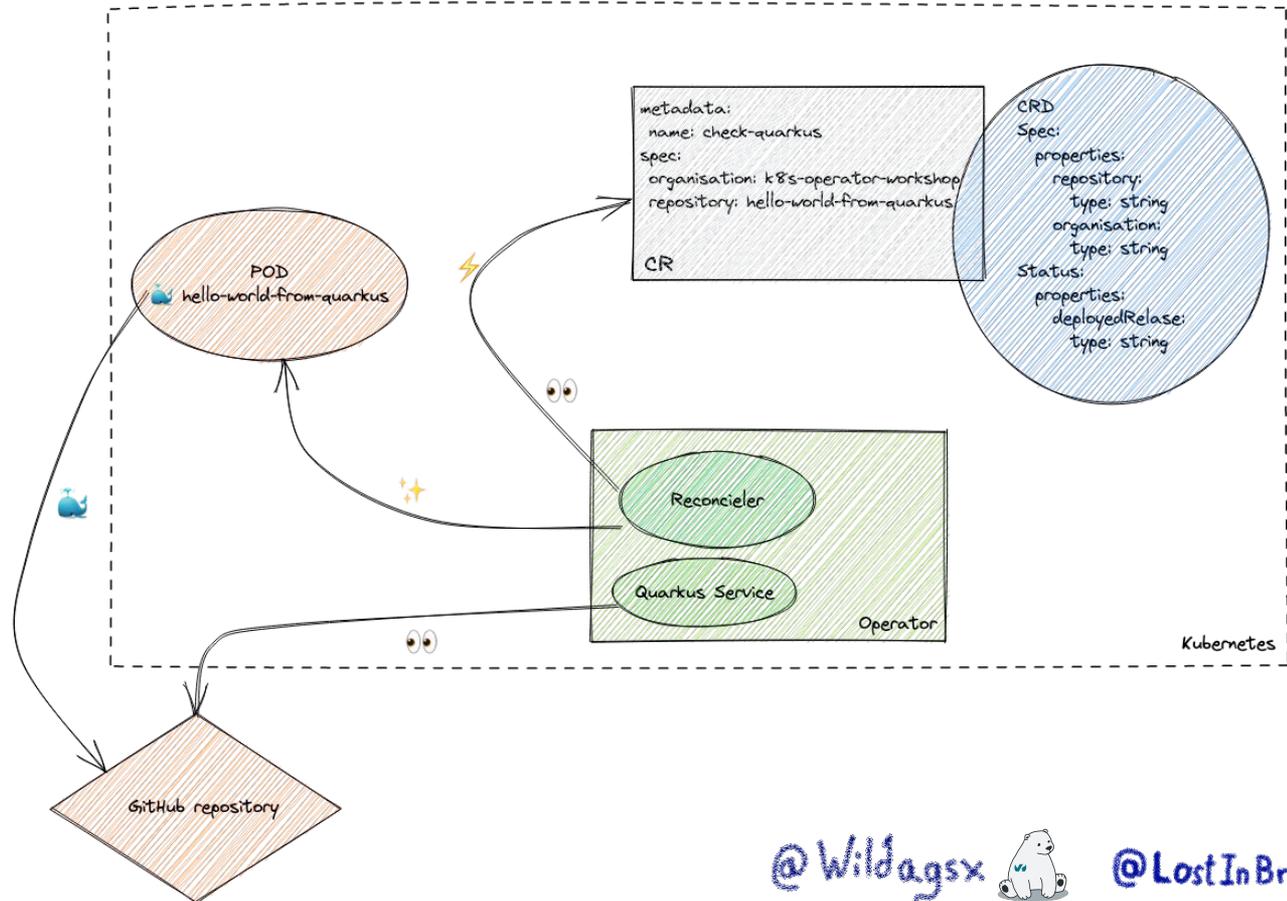


 <https://github.com/k8s-operator-workshop/workshop-operator-hello-world>

GitHub release watcher

Un opérateur qui fait vraiment quelque chose





GitHub release watcher

-  Création du projet
-  Définition de la CR (Spec et Status)
-  Création d'un service Quarkus
-  Gestion d'un événement externe (méthode pooling)
-  Packaging et déploiement dans Kubernetes
-  Gestion de la sécurité



<https://github.com/k8s-operator-workshop/workshop-operator-release-detector/tree/old-way-resources-and-pooling>

More exercises / examples

Questions

**Kahoot -> livre Aurélie + d'autres pour les
3 premiers + vouchers**

- <https://github.com/k8s-operator-workshop>
- <https://gitpod.io/>
- <https://javaoperatorsdk.io/>
- <https://sdk.operatorframework.io/>
- <https://kubernetes.io/docs/concepts/extend-kubernetes/api-extension/custom-resources/>
- <https://kubernetes.io/docs/concepts/architecture/controller/>
- <https://quarkiverse.github.io/quarkiverse-docs/quarkus-operator-sdk/dev/index.html>