Jug Summer Camp

-enjoy it-

From Silos to DevOps to Platform Engineering

Embracing GitOps and going behind the hype

Speaker : Horacio Gonzalez - @LostInBrittany

clever cloud

SERLI MAIF Couchbase elastic GitHub clever cloud Cloud APIM DARVA gedivote.

Horacio Gonzalez

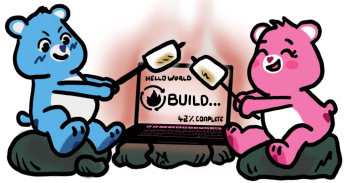
@LostInBrittany

Spaniard Lost in Brittany

Head of DevRel

clever cloud

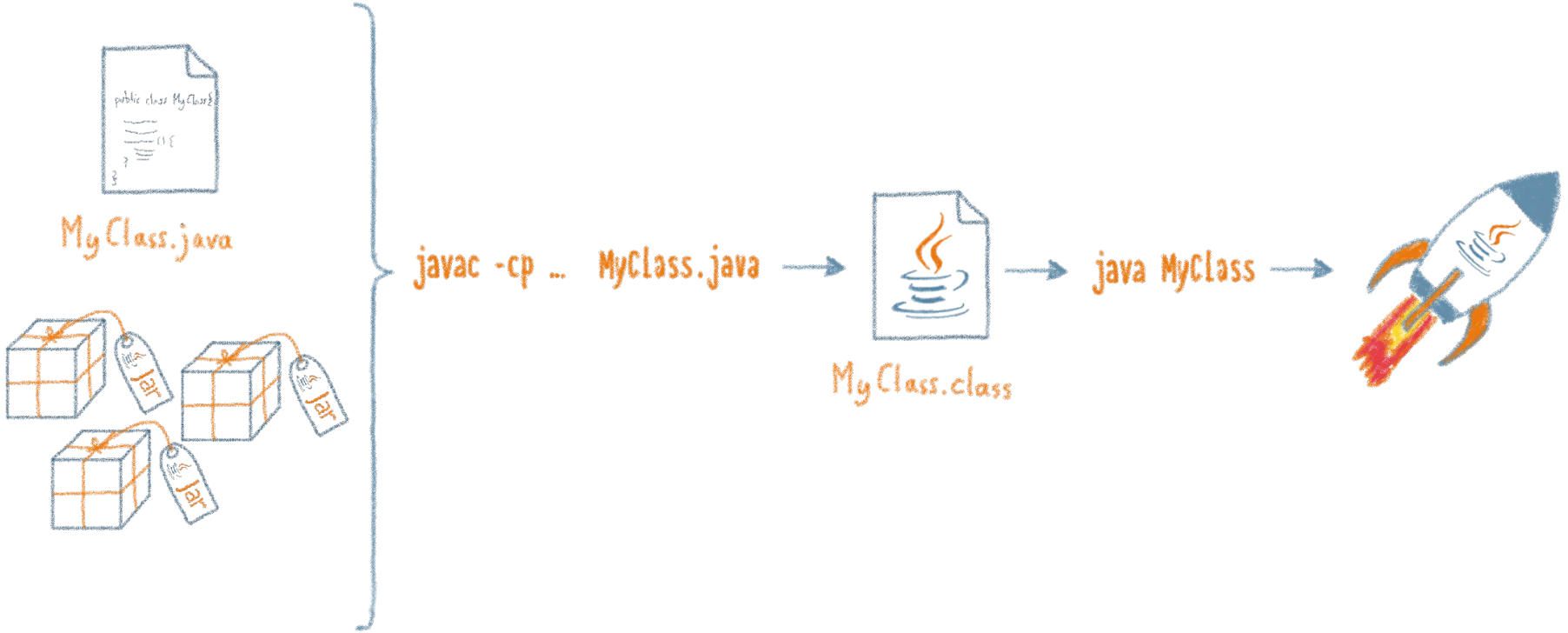
Finist Devs



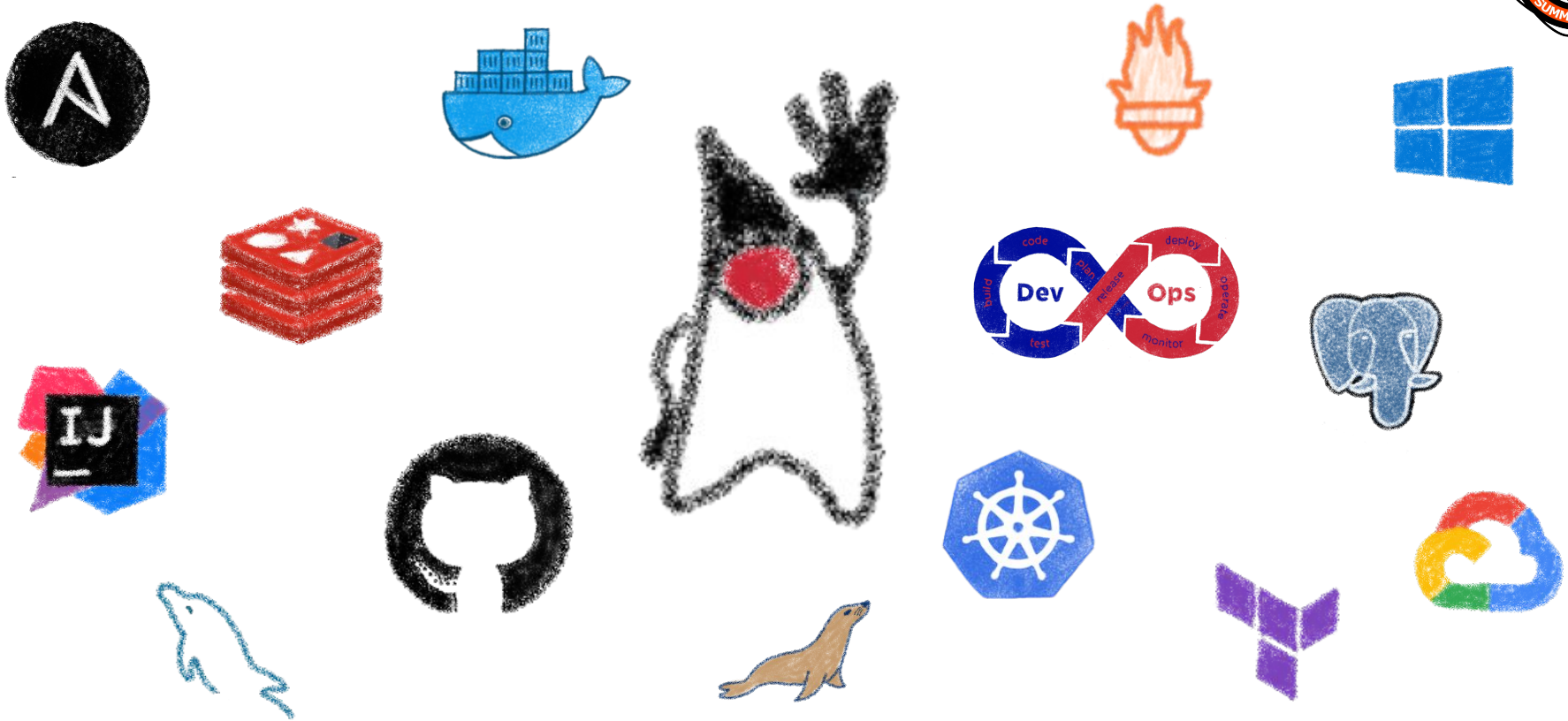
Adding layers of shiny complexity

Last 30 years in software development

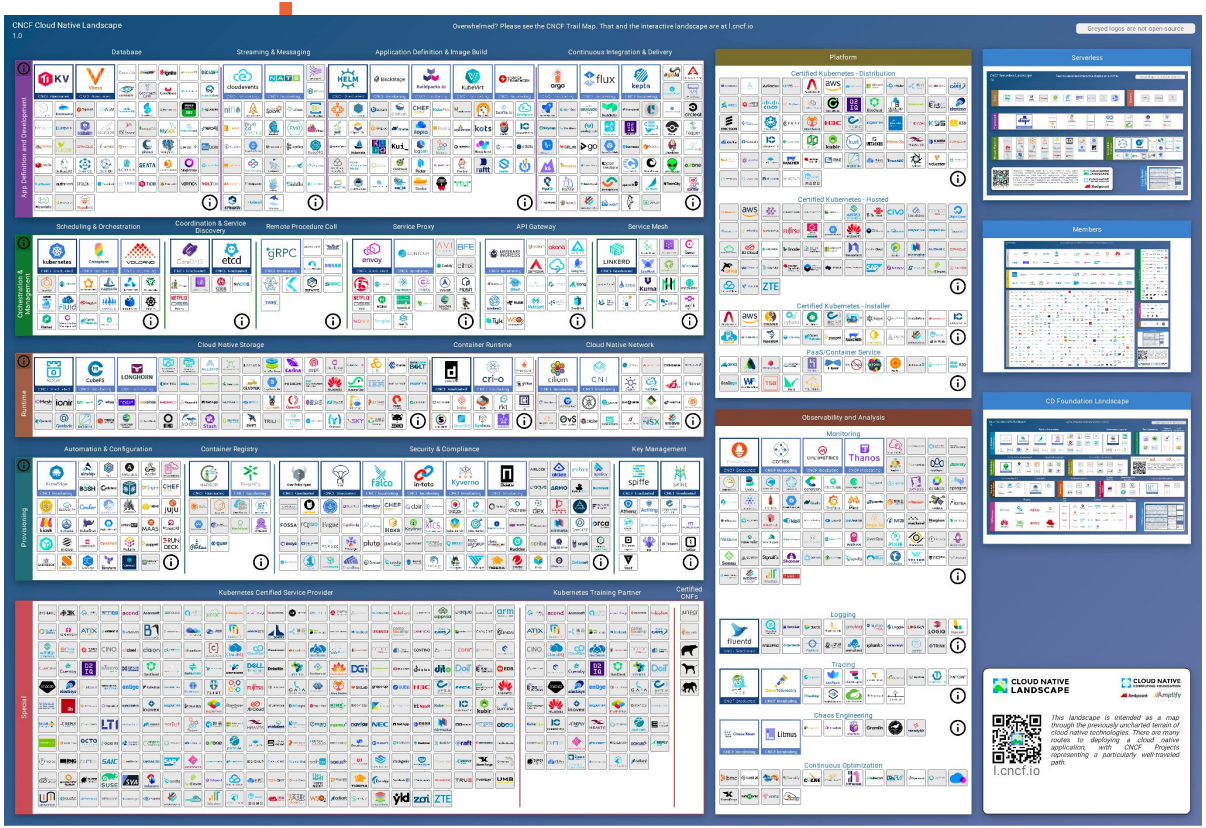
From write-compile-exec to Kubernetes



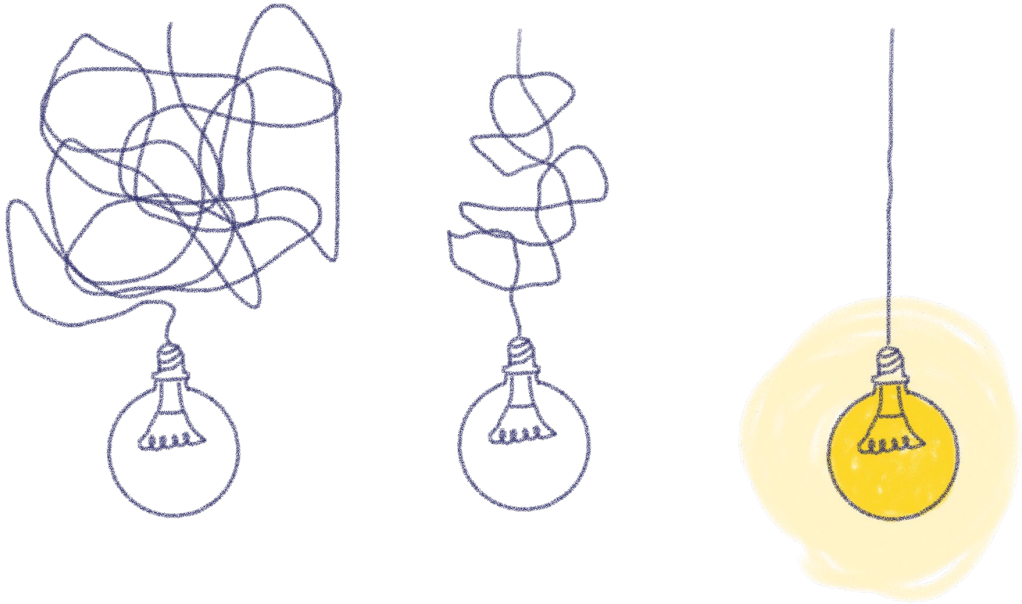
From write-compile-exec to Kubernetes



If I were a student now, I would feel



Platform Engineering to the rescue



Empowering developers while reducing complexity

IT in the 90s

Once upon a time...

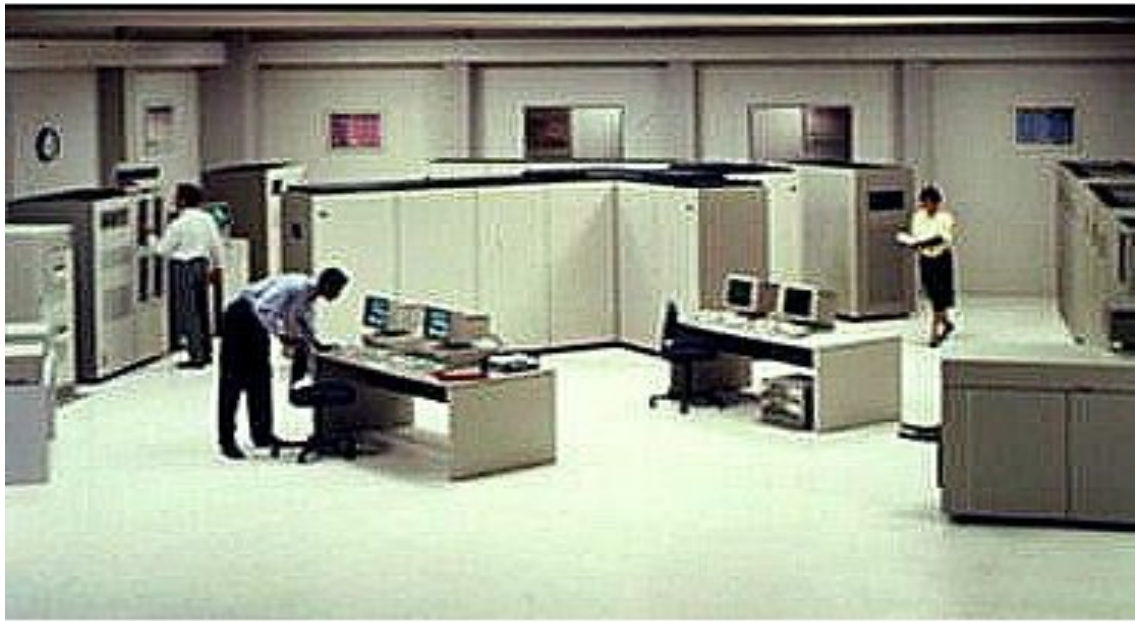
In a time almost forgotten

When even internet was young...

When Windows 95 was the cutting edge

And a 100 Mb disk was huge...

Big companies still used mainframes



Bigger, fancier, but still the same old IBM

Bare-metal based IT reigned

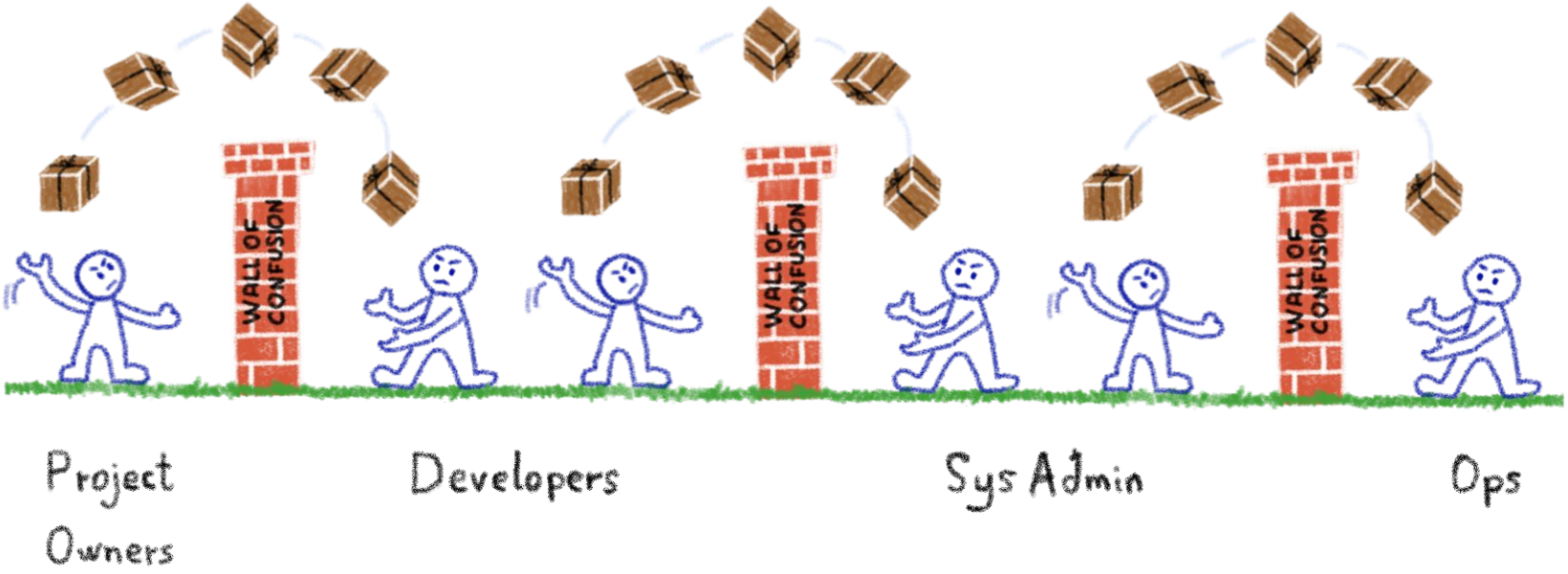
Control, reliability, security...

But cost, rigidity, logistics...

Applying the industrial model

Trying to shoehorn IT into a model where it doesn't fit

Walls & Silos



And procedures, and hierarchy, and corporate politics

Why are we managing IT like factories?

Because we didn't know otherwise?

Rémi Verchère ❄

@rverchere

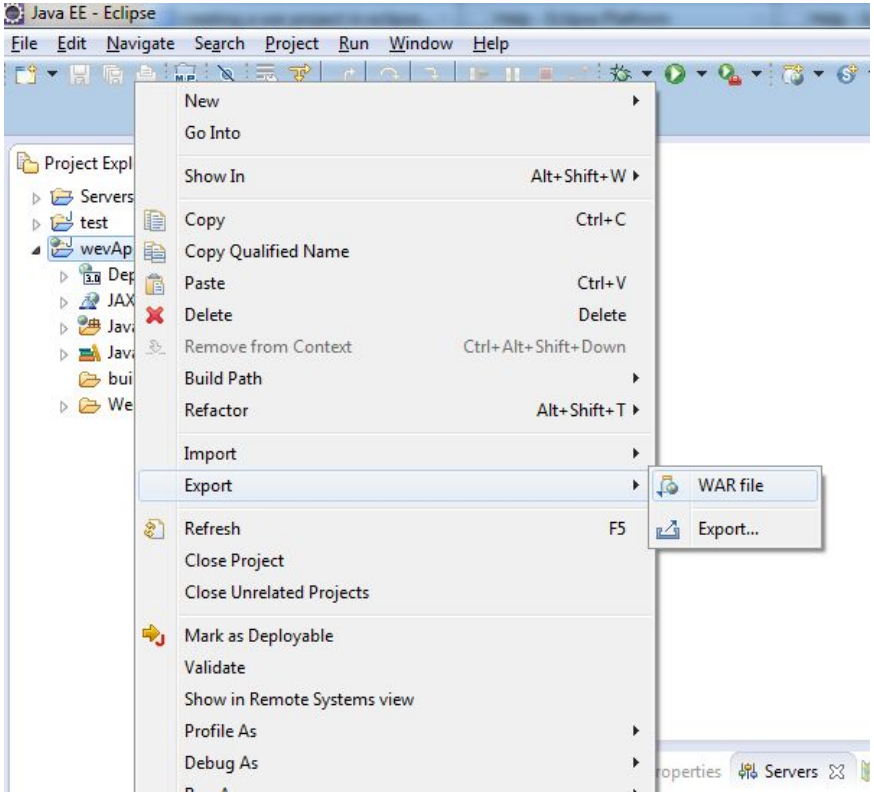
bash will still be used

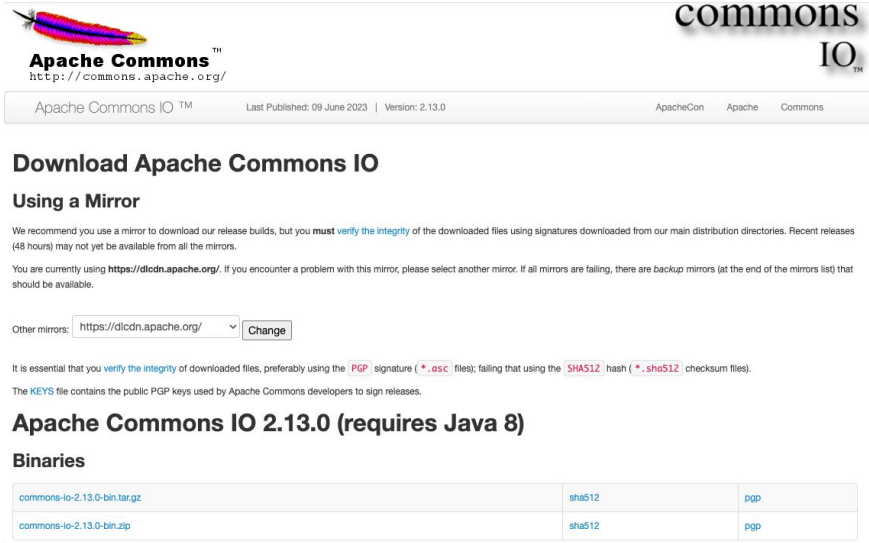
7:46 PM · Aug 8, 2023 · **290** Views

Tooling evolves

CVS, Ant and (Leeeroy) Jenkins

Old school procedures





Tooling empowering changes

Theory existed since 1999
But without the right tooling...

Source control tools

ClearCase

CVS

Better than copying and renaming folders...

Dependency management & build

The agile dependency manager

Better than grabbing each dependency in their website and running `javac` by hand...

Unit testing and continuous integration

Hudson CI

continuum™

JUnit

Jenkins

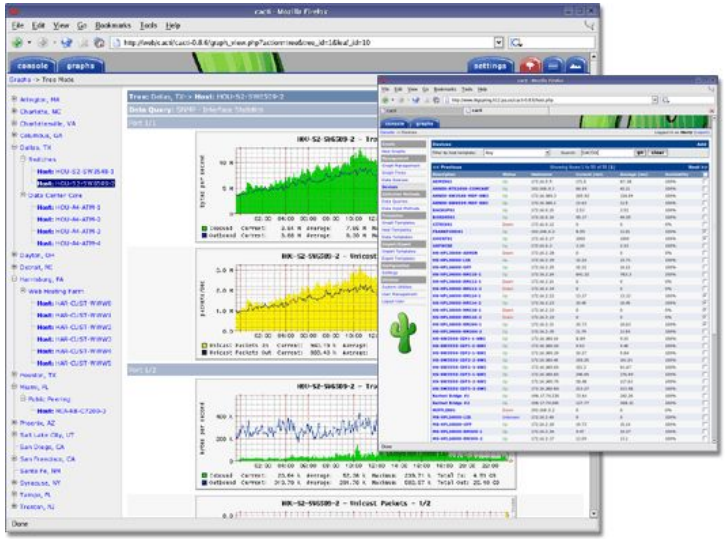
If *Testing is Doubting*, let's doubt automatically

Monitoring tools

cacti

Nagios

ZABBIX



No more spending nights looking at a status screen

So many more possibilities...

So much more complexity!

Rémi Verchère ❄

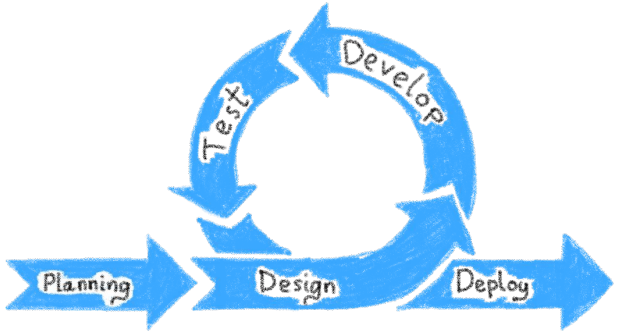
@rverchere

bash will still be used

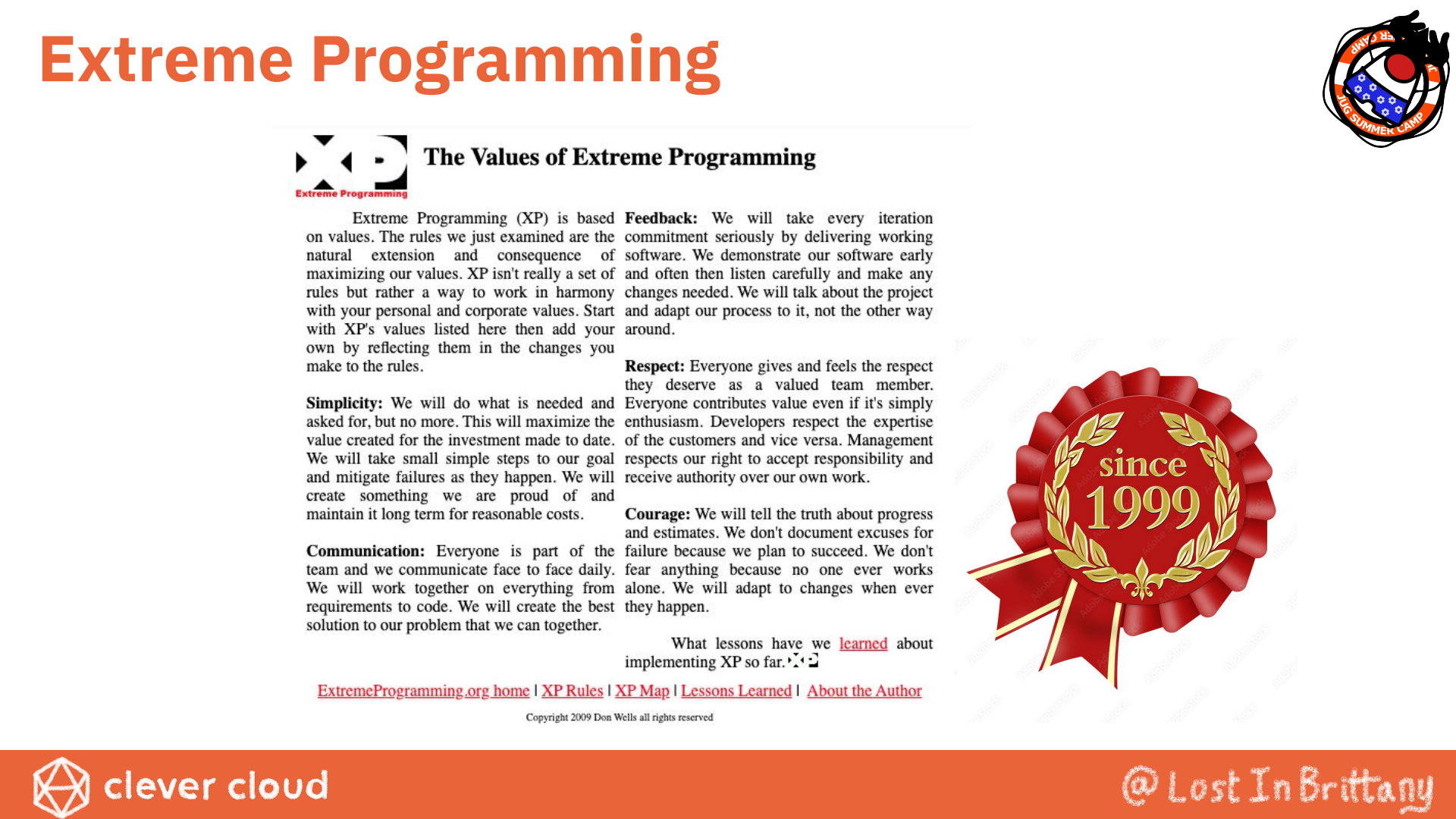
7:46 PM · Aug 8, 2023 · **290** Views

XP, agility and DevOps

Buzzwords that changed the IT



Extreme Programming



The Values of Extreme Programming

Extreme Programming (XP) is based on values. The rules we just examined are the natural extension and consequence of maximizing our values. XP isn't really a set of rules but rather a way to work in harmony with your personal and corporate values. Start with XP's values listed here then add your own by reflecting them in the changes you make to the rules.

Simplicity: We will do what is needed and asked for, but no more. This will maximize the value created for the investment made to date. We will take small simple steps to our goal and mitigate failures as they happen. We will create something we are proud of and maintain it long term for reasonable costs.

Communication: Everyone is part of the team and we communicate face to face daily. We will work together on everything from requirements to code. We will create the best solution to our problem that we can together.

Feedback: We will take every iteration commitment seriously by delivering working software. We demonstrate our software early and often then listen carefully and make any changes needed. We will talk about the project and adapt our process to it, not the other way around.

Respect: Everyone gives and feels the respect they deserve as a valued team member. Everyone contributes value even if it's simply enthusiasm. Developers respect the expertise of the customers and vice versa. Management respects our right to accept responsibility and receive authority over our own work.

Courage: We will tell the truth about progress and estimates. We don't document excuses for failure because we plan to succeed. We don't fear anything because no one ever works alone. We will adapt to changes when ever they happen.

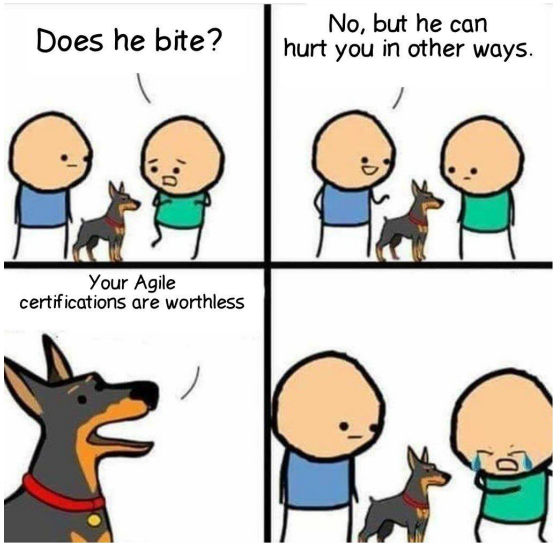
What lessons have we learned about implementing XP so far.

ExtremeProgramming.org home | XP Rules | XP Map | Lessons Learned | About the Author

Manifesto for Agile Software Development

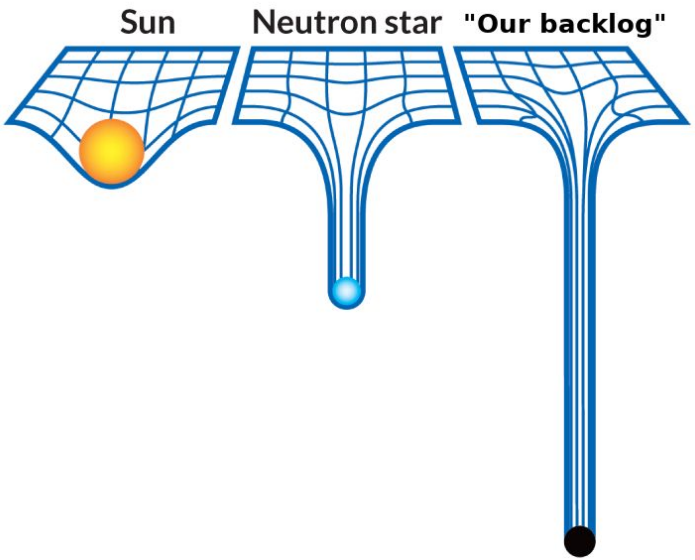
Breaching walks, breaking down silos

The business of Agility

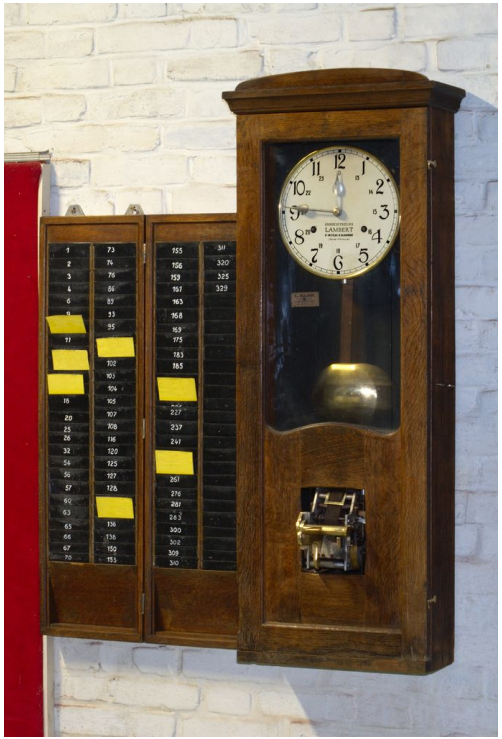


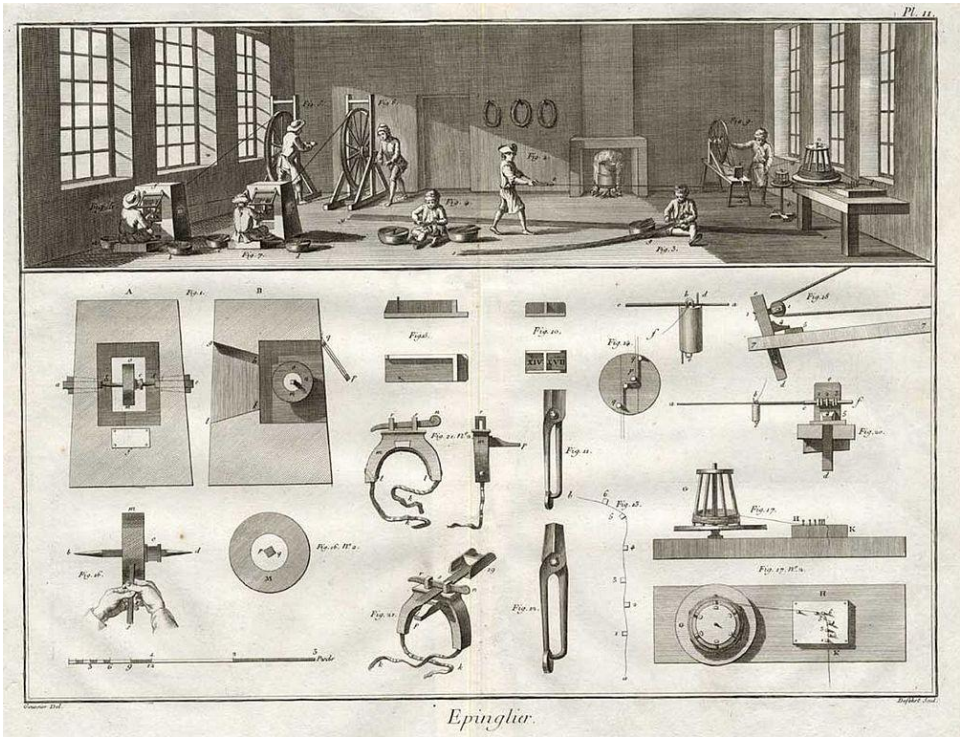
The Dark Side rises

Agile Tooling



Back to industrial practices ?



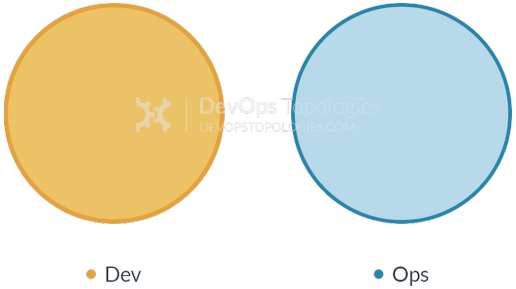


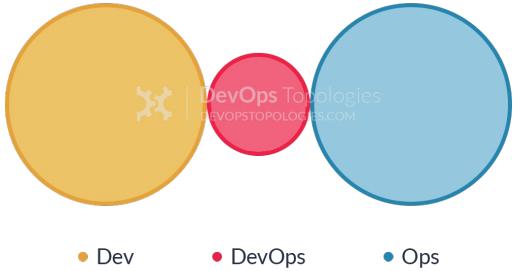
DevOps: breaking Dev and Ops Silos

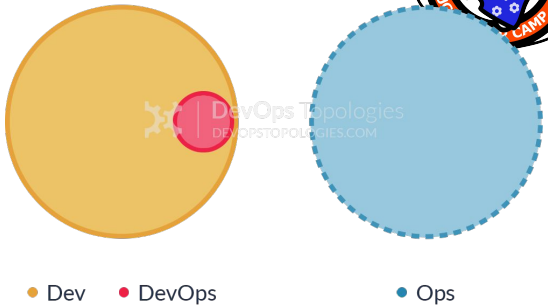
DevOps is a reaction to the wall of confusion

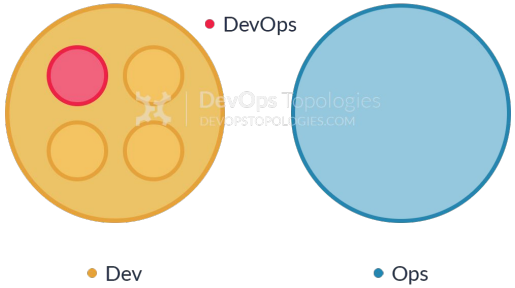
Making the different stakeholders
to work together in sync

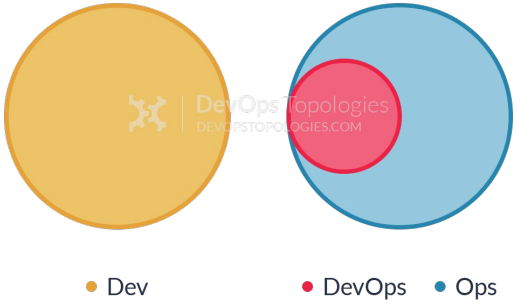
DevOps anti patterns











More in [DevOps Topologies](#) site

DevOps also has another Dark Side

> A DevOps engineer is an IT generalist who should have a wide-ranging knowledge of both development and operations, including coding, infrastructure management, system administration, and DevOps toolchains.

WTF is a DevOps Engineer? And a DevSecOps?
A DevMlOps? A DevAiDataSecOps? A Dev*Ops?

So we have Cults of Agility and Dev*Ops

And so much more complexity!

Rémi Verchère ❄

@rverchere

bash will still be used

7:46 PM · Aug 8, 2023 · **290** Views

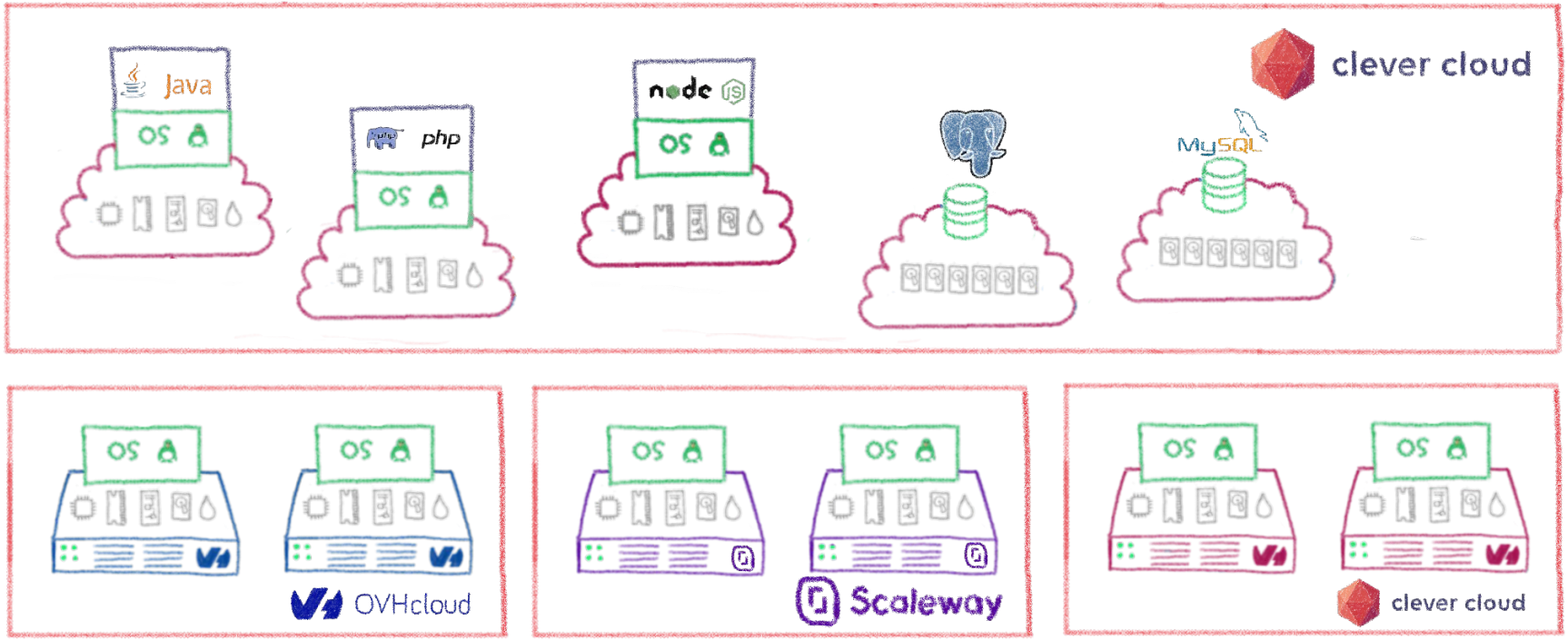
Enter the Cloud

Renting server time in other's people infra

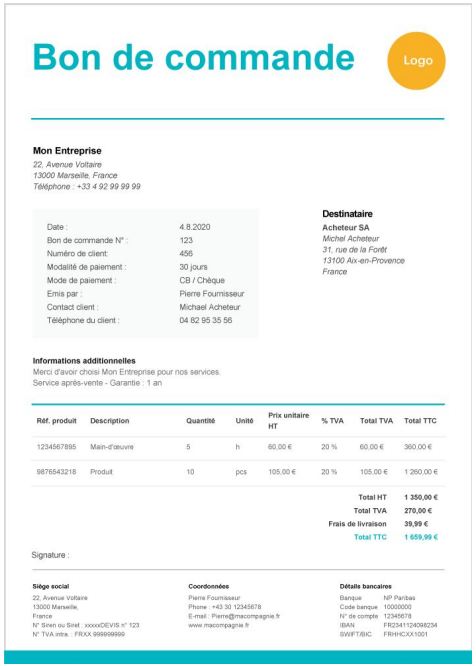
From virtualisation to the cloud

How to use the infrastructure at its full capacity

Cloud demands automation



It changes the way how IT works



Bon de commande

Logo

Mon Entreprise
22, Avenue Voltaire
13000 Marseille, France
Téléphone : +33 4 92 99 99 99

Date :	4.8.2020
Bon de commande N° :	123
Numéro de client :	456
Modalité de paiement :	30 jours
Mode de paiement :	CB / Chèque
Emis par :	Pierre Fournisseur
Contact client :	Michael Acheteur
Téléphone du client :	04 82 95 35 56

Destinataire
Acheteur SA
Michel Acheteur
31, rue de la Forêt
13100 Aix-en-Provence
France

Informations additionnelles
Merci d'avoir choisi Mon Entreprise pour nos services.
Service après-vente - Garantie : 1 an

Réf. produit	Description	Quantité	Unité	Prix unitaire HT	% TVA	Total TVA	Total TTC
1234567895	Main-d'oeuvre	5	h	60,00 €	20 %	60,00 €	360,00 €
9876543218	Produit	10	pcs	105,00 €	20 %	105,00 €	1 260,00 €

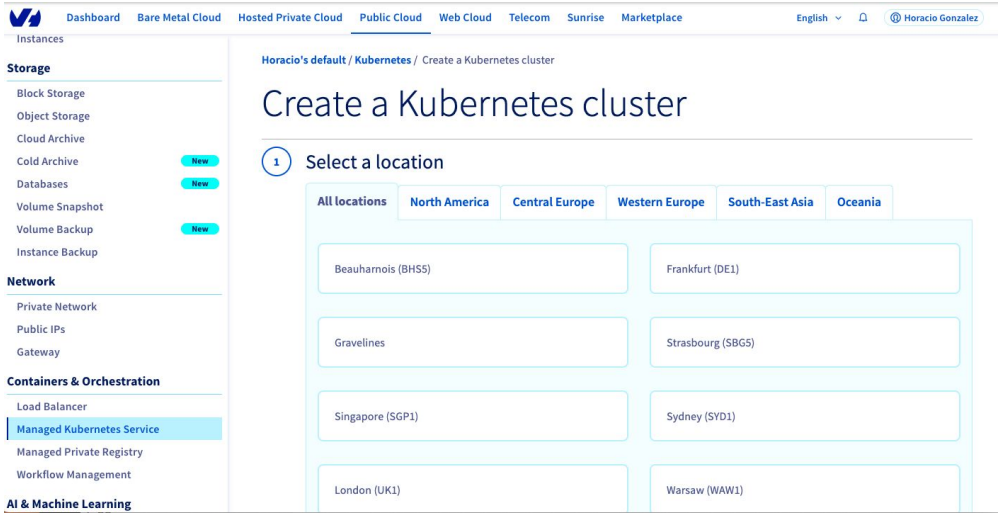
Total HT	1 350,00 €
Total TVA	270,00 €
Frais de livraison	39,99 €
Total TTC	1 659,99 €

Signature :

Siège social
22, Avenue Voltaire
13000 Marseille
France
N° Siren ou Siret : xxxxxDEVIS n° 123
N° TVA intra. : FRXX 999999999

Coordonnées
Pierre Fournisseur
Phone : +43 30 12345678
E-mail : Pierre@macompagnie.fr
www.macompagnie.fr

Détails bancaires
Banque : NP Paribas
Code banque : 10003000
N° de compte : 12345678
IBAN : FR2341124098234
SWIFT/BIC : FRHHCXX1001

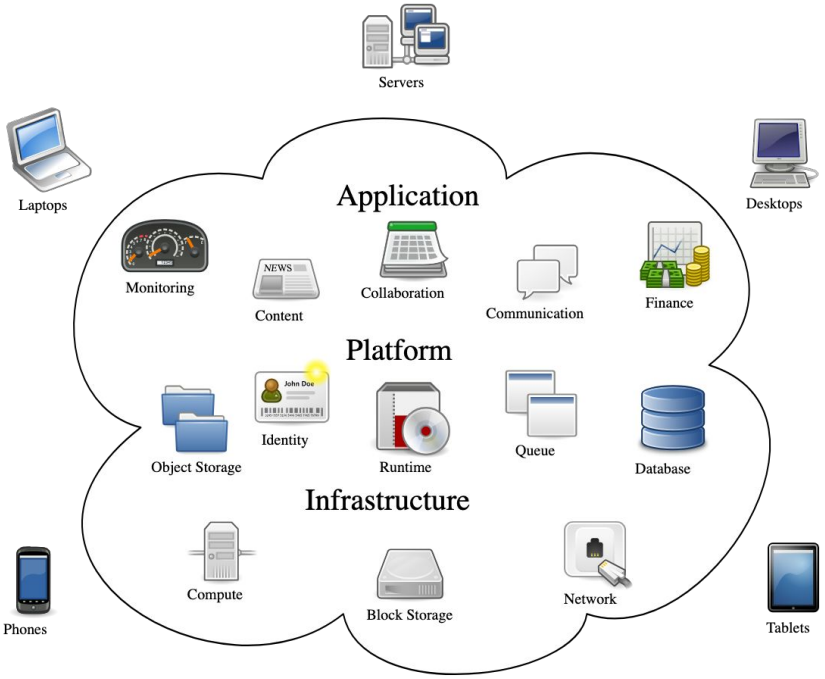


And it demands a mentality change

Empowering developers

Infrastructure is only a click away

Distributed is the new black



Cloud Native architectures and services

Sysadmins who code

Creating tools: automation, monitoring, observability...

New roles appear: SRE

WTF is a System Reliability Engineer?

250 cloud products only in this provider...

And so much more complexity!

Rémi Verchère ❄

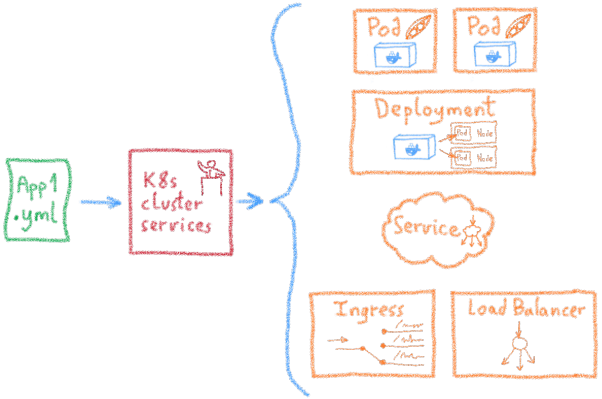
@rverchere

bash will still be used

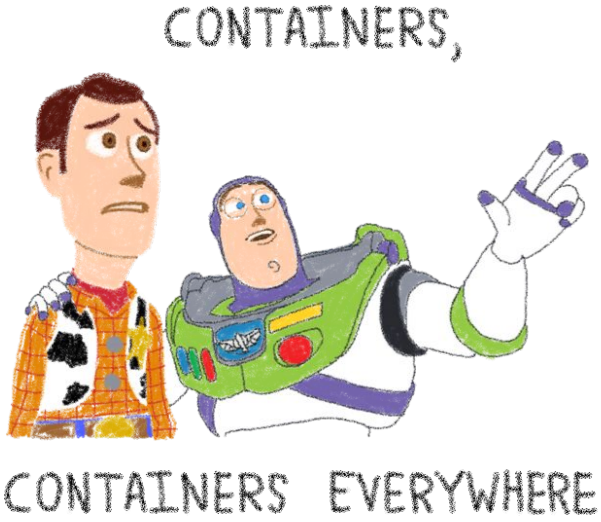
7:46 PM · Aug 8, 2023 · **290** Views

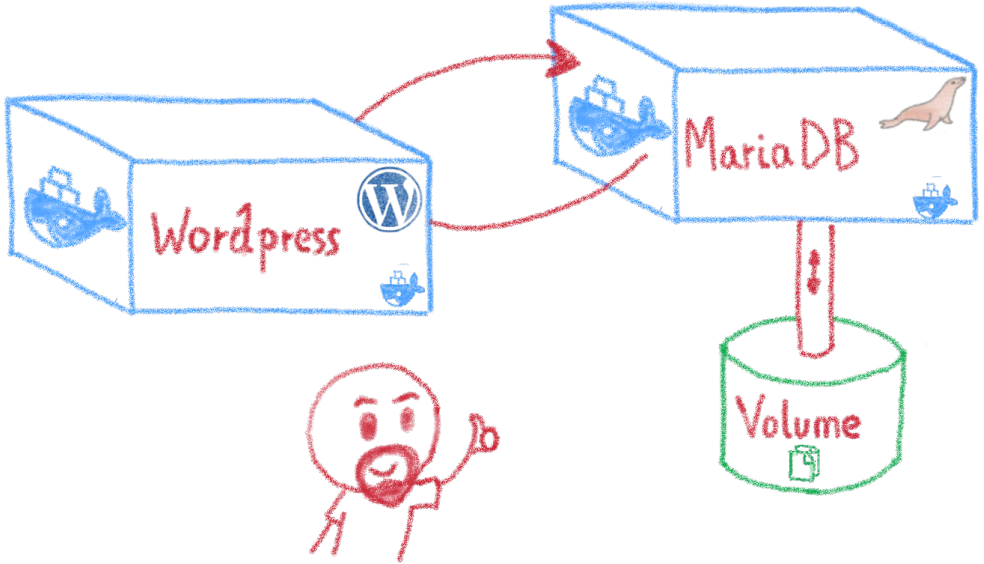
Declarative Infrastructure

The intern metaphor

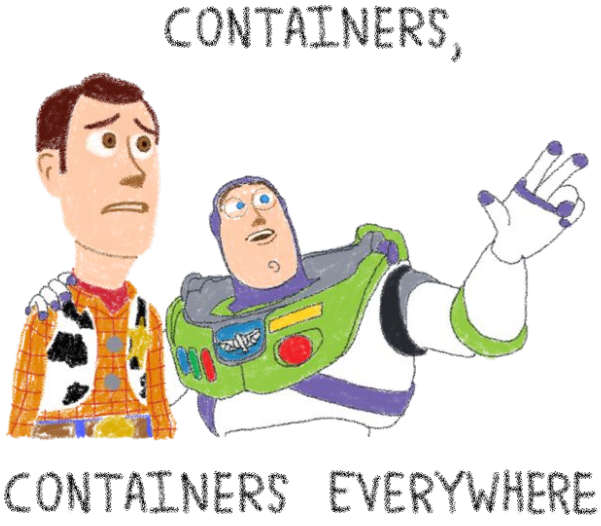


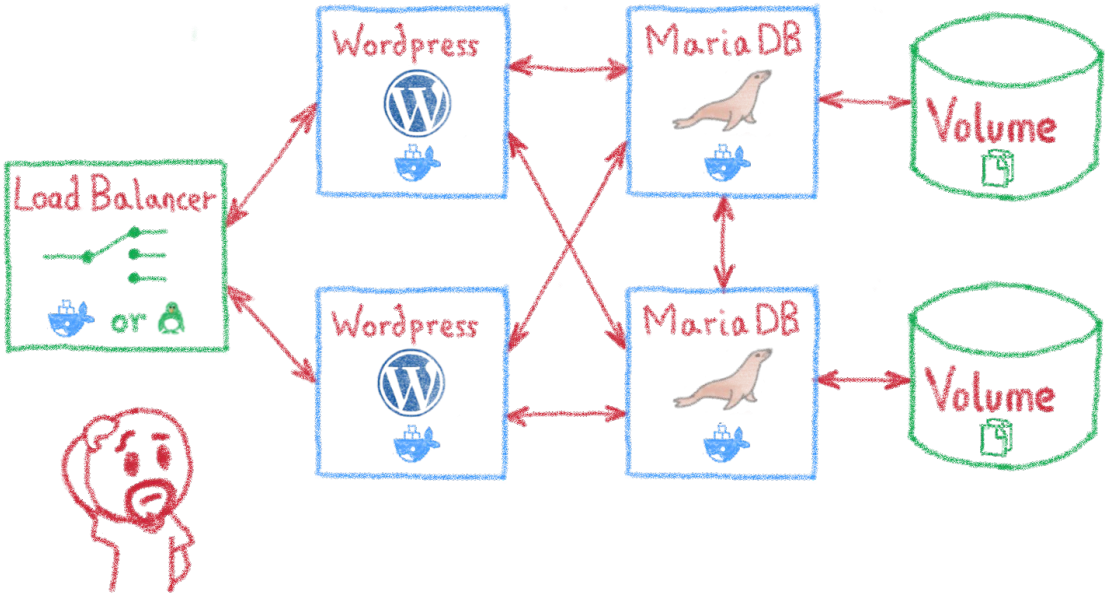
Containers make dev life easier





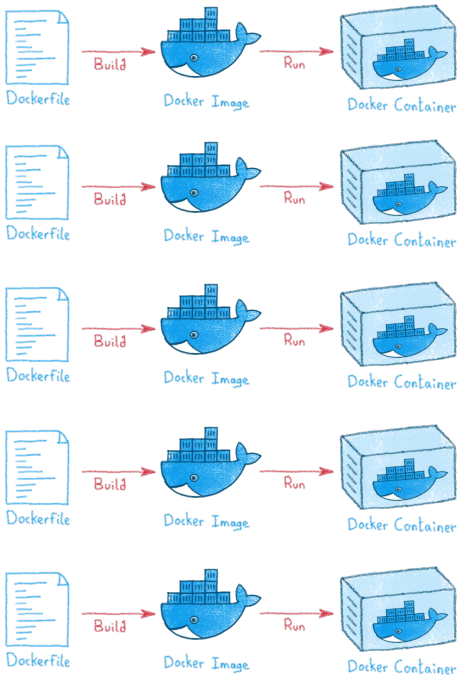
Less simple if you must operate them





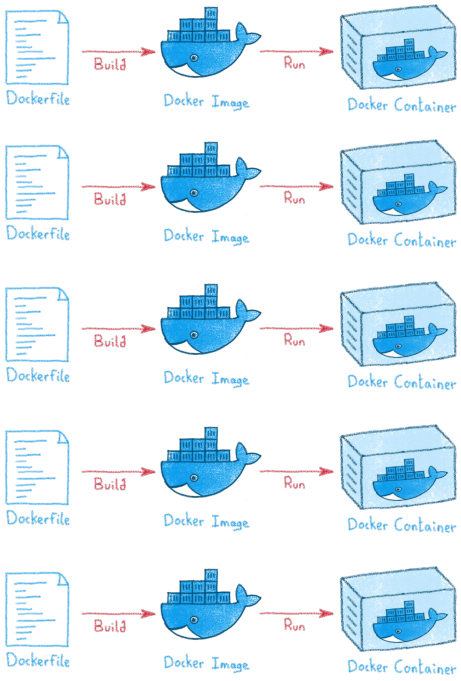
Like in a production context

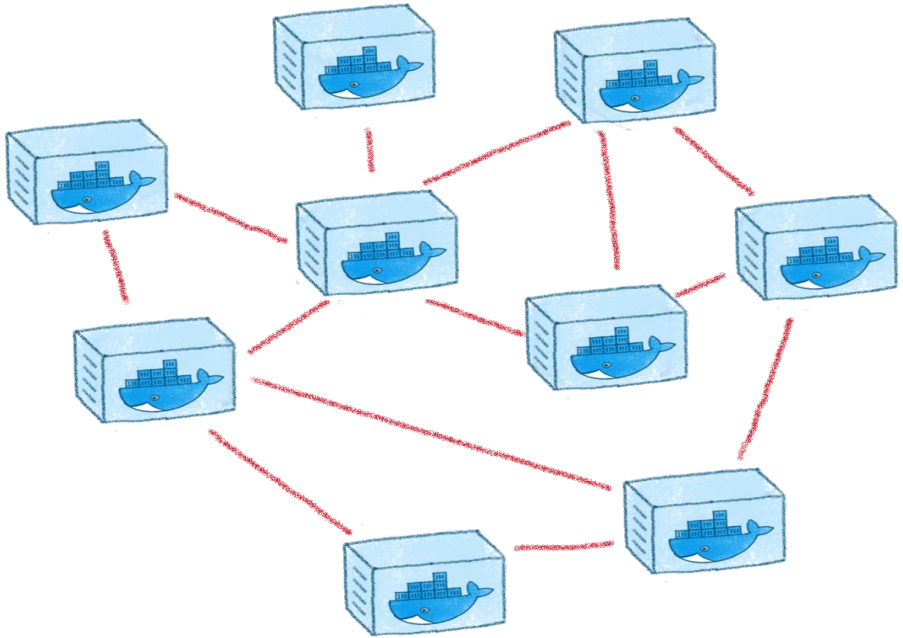
And what about microservices?



Are you sure you want to operate them by hand?

And what about microservices?



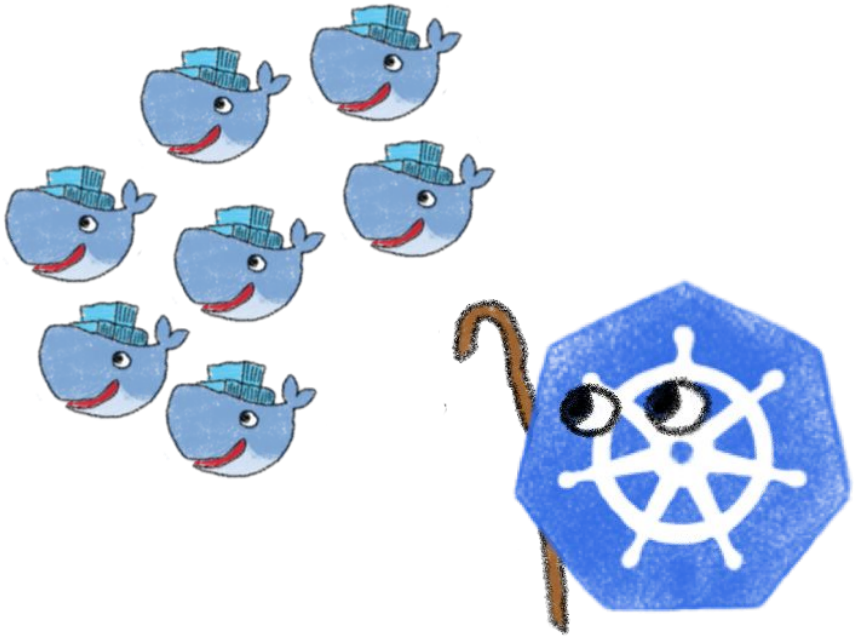


Are you sure you want to operate them by hand?

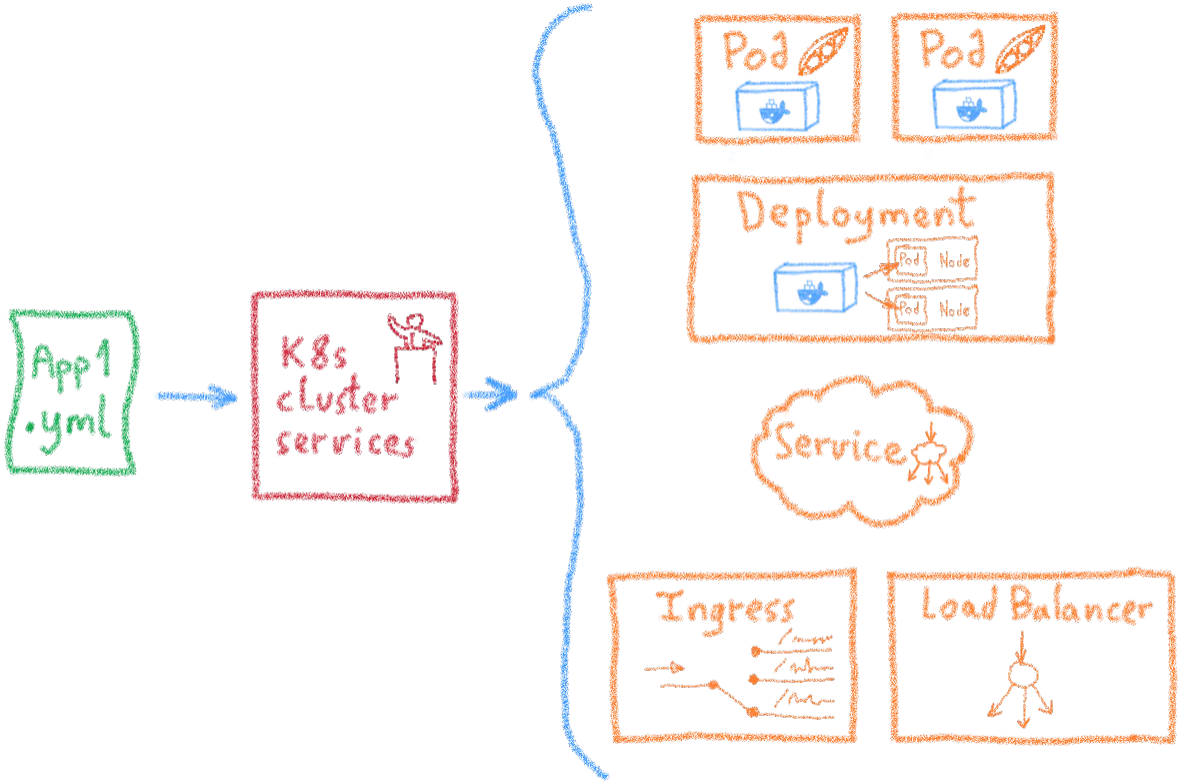
Kubernetes: a full orchestrator

Takes care of:

- Deployment
- Scaling
- Monitoring
- Repairing
- Securing
- ...



Kubernetes - Desired State Management



Ingress

Services

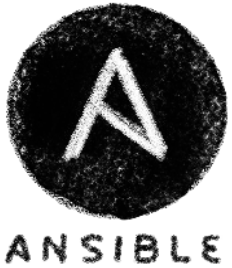
Deployments

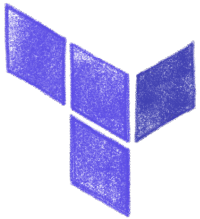
Pods

Sidecars

Replica Sets

Infrastructure as Code





HashiCorp
Terraform

- Build 
- Modify 
- Version 

your infrastucture

Containers? Pods? Ansible? Terraform?

So much more complexity!

Rémi Verchère ❄

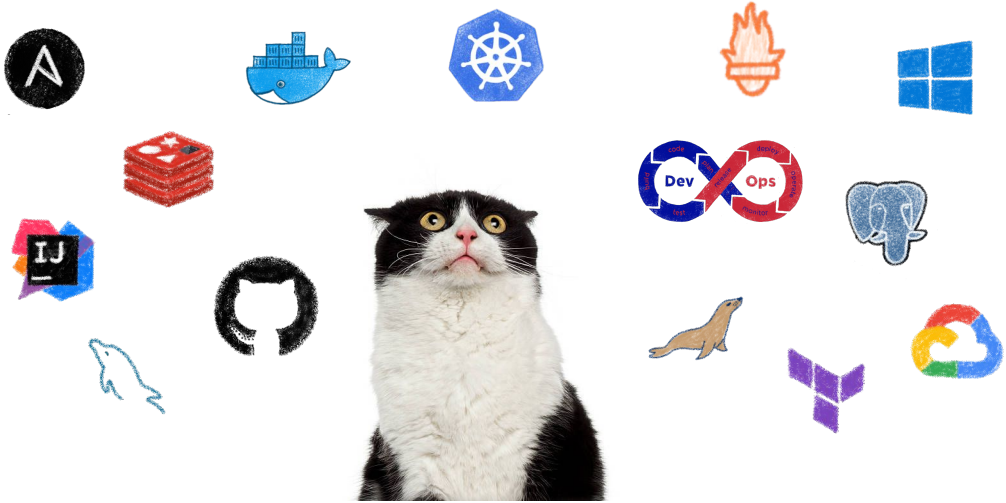
@rverchere

bash will still be used

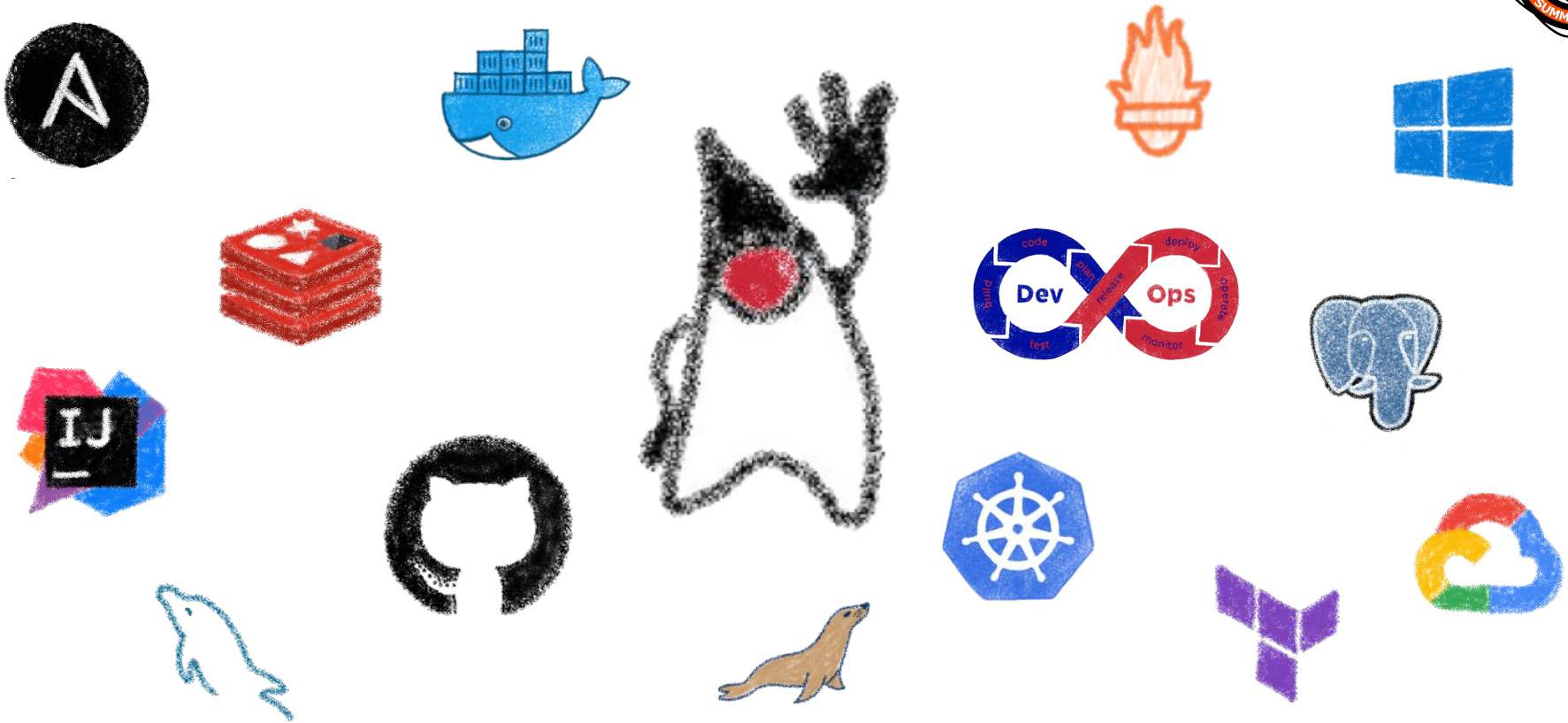
7:46 PM · Aug 8, 2023 · **290** Views

Becoming a developer in 2024

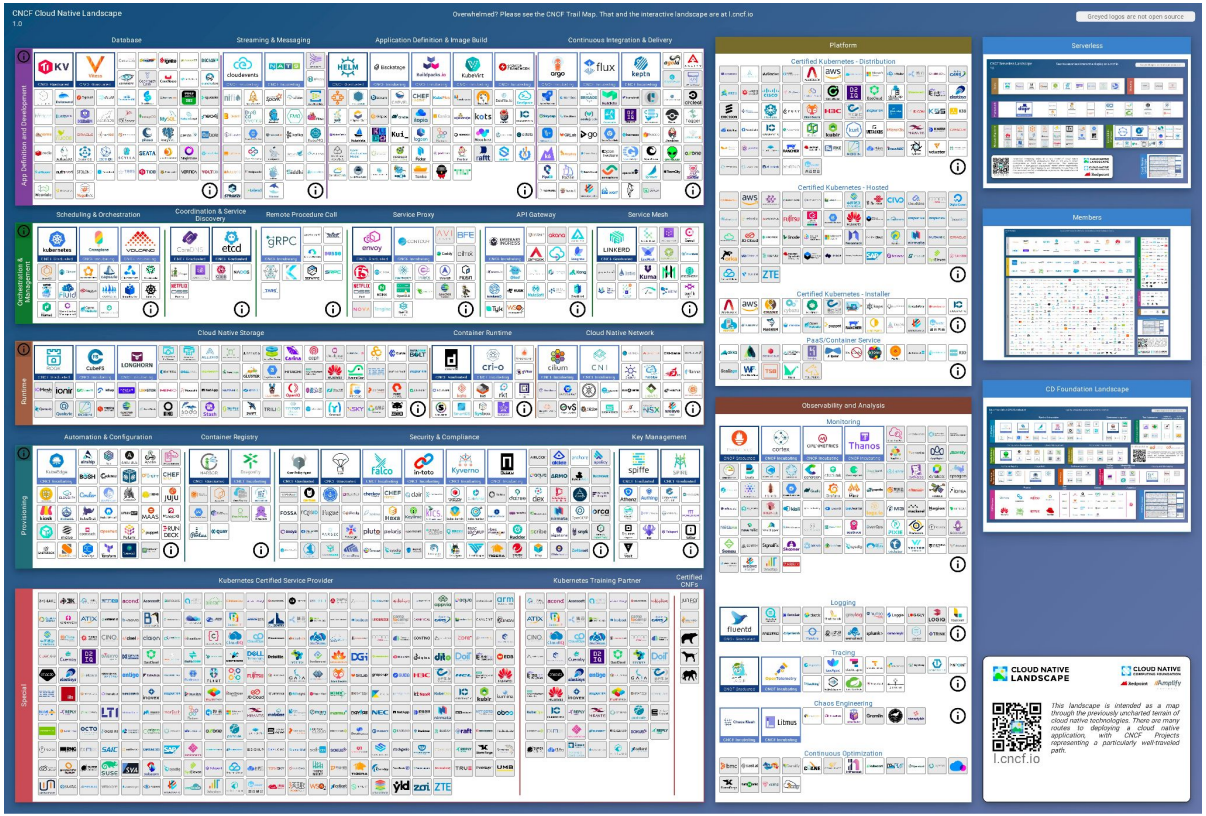
Do I need to know all that?



So many things to learn...

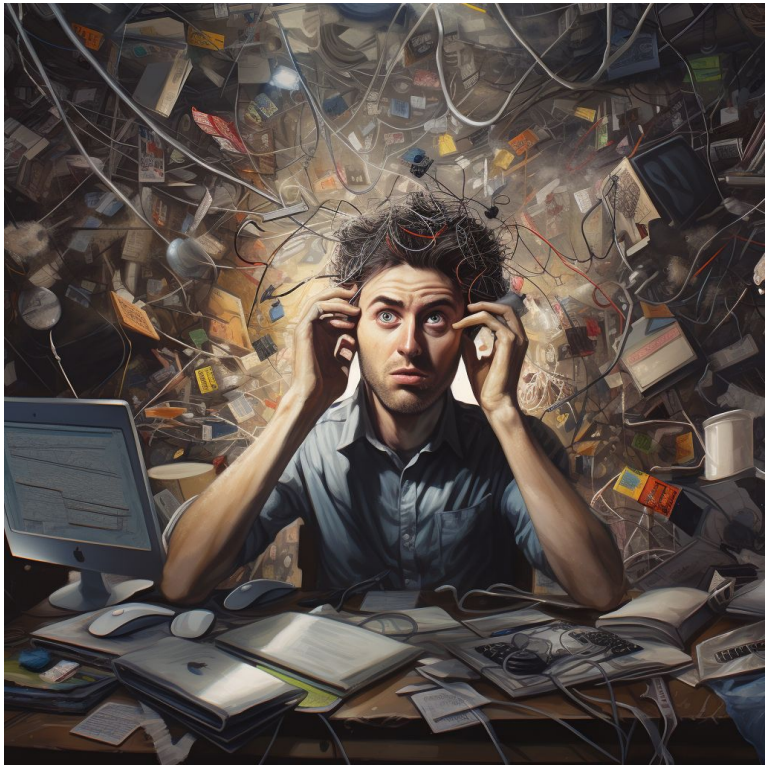


So many options ...



Managing the self-service commodity

Shift left and Cognitive Load



Platform Engineering to the rescue

Without adding more complexity?

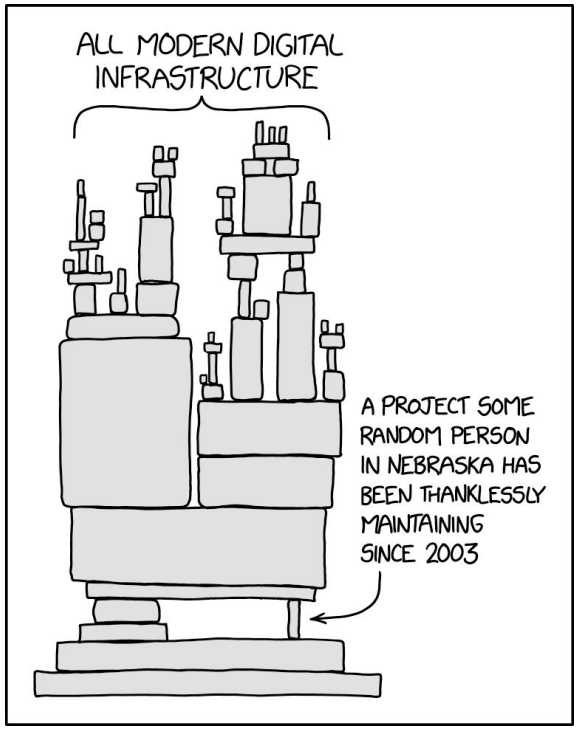
What's Platform Engineering?

Platform engineering is the discipline of designing and building toolchains and workflows that enable self-service capabilities for software engineering organizations in the cloud-native era.

Platform engineers provide an integrated product most often referred to as an “Internal Developer Platform” covering the operational necessities of the entire lifecycle of an application.

Lucca Galante

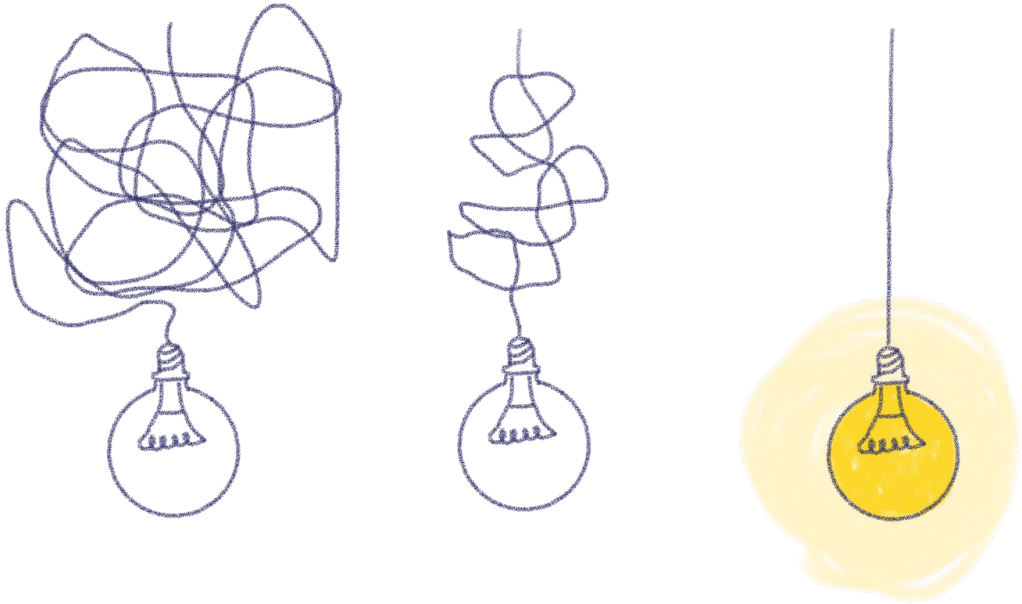
A fancy name for something already there



Most companies already have some kind of platform

Often homemade…

The purpose of Platform Engineering



Empowering developers while reducing complexity

But how can we create them?

GitOps

What is GitOps?

Git is the single source of truth

Treat everything as code

Operations through Git workflows

Benefits of GitOps

- Collaboration
- Visibility & auditing
- Security & reliability
- Provisioning & deployment

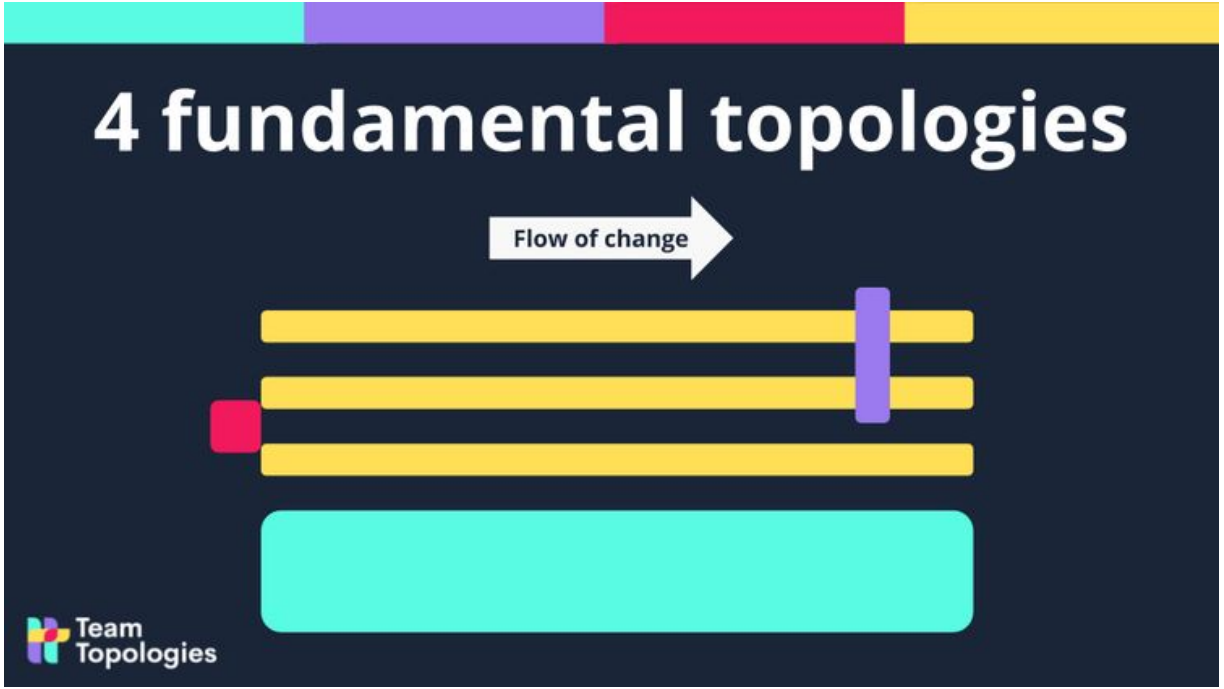
Team Topologies

It's not only a tool, but an organisation

Team Topologies

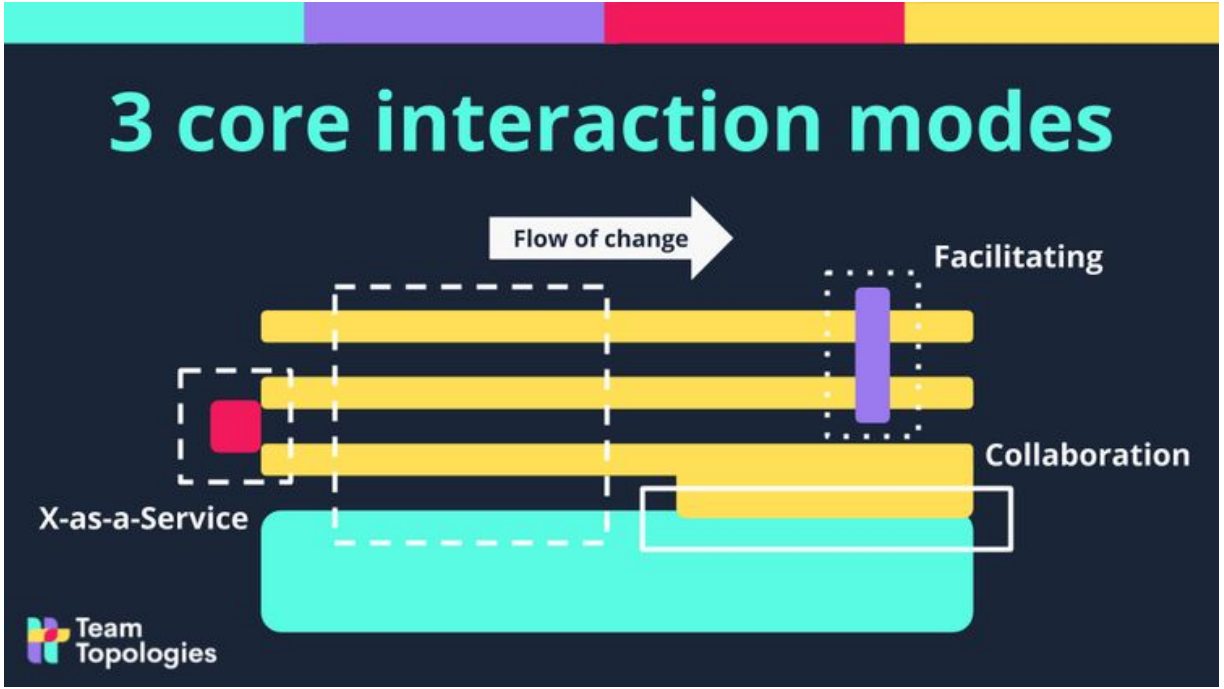
https://teamtopologies.com/

Team Topologies



https://teamtopologies.com/

Team Topologies



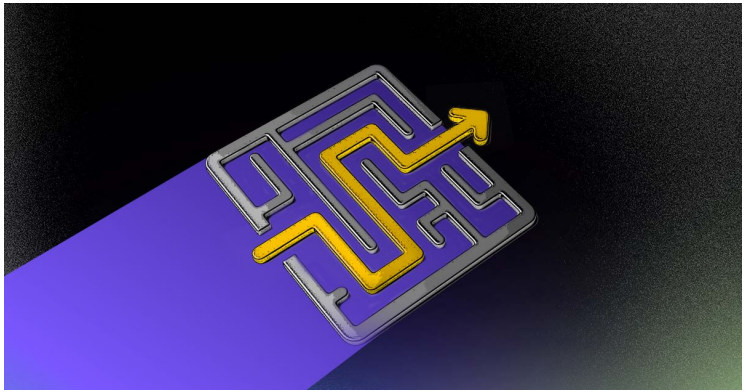
https://teamtopologies.com/

Principles of Platform Engineering

To make it work

Paving golden paths

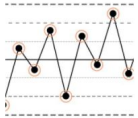
Platform engineering is about binding process and tools into a paved road. Rather than letting everybody operate everything and having to understand the entire toolchain to do so, platform engineers provide the glue to bind everything into a consistent self-service experience.



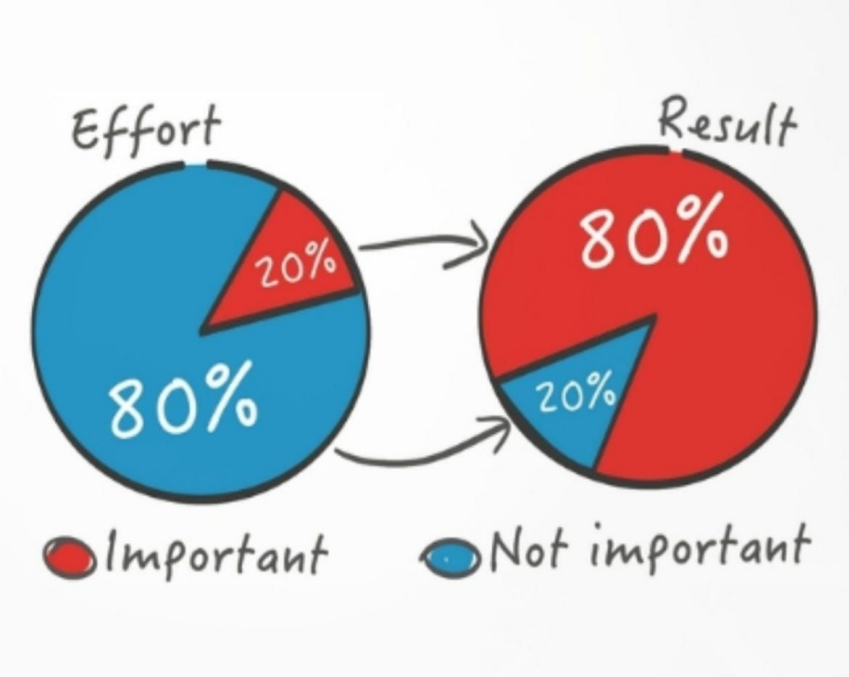
Clear mission and role

Platform as a Product

Work Types

Type	Output	Planning	Goal	Approach
Manufacturing			• Maximise efficiency • Commodification	• Mass production • Precise Specifications • Control Variation
Construction			• Compromise between cost, time, scope and quality.	• Plan & Execute • Comprehensive Specifications • Control Change
Product Development			• Maximise quality and features • Discard low value work	• Iterative experimentation • Respond to feedback

Focus on common problems



Glue is valuable

Don't reinvent the wheel

Rémi Verchère ❄

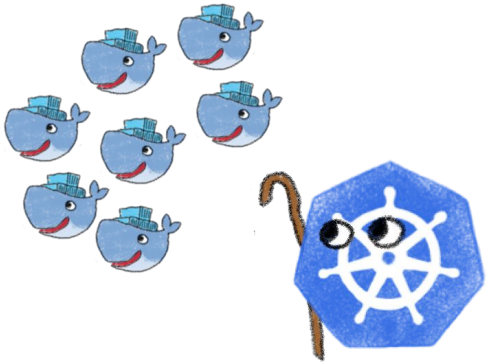
@rverchere

bash will still be used

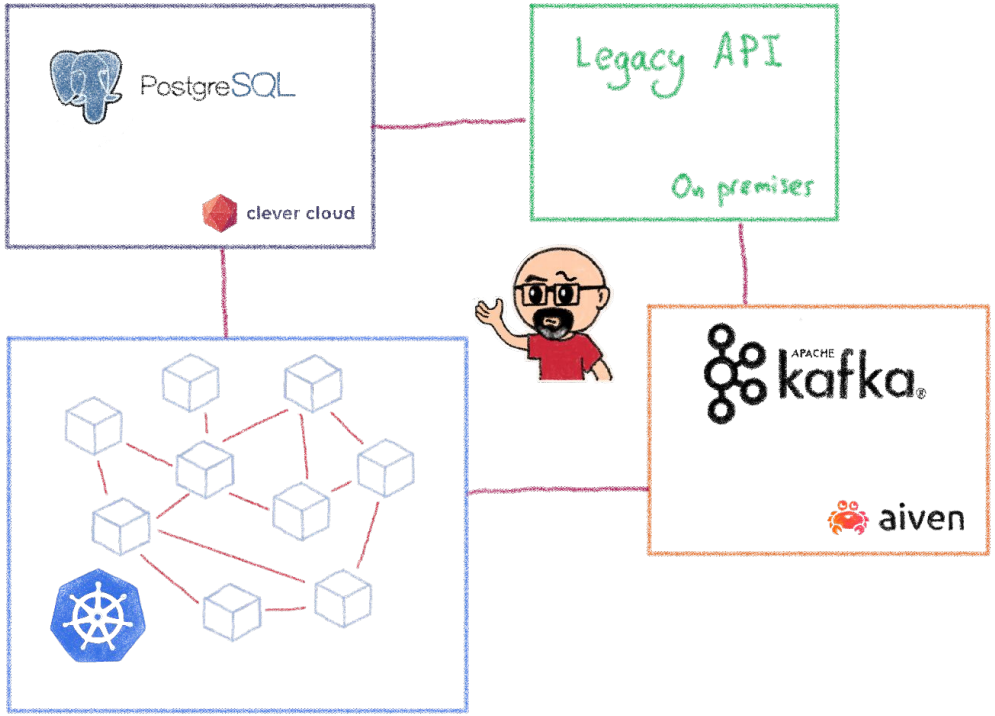
7:46 PM · Aug 8, 2023 · **290** Views

An IDP over Kubernetes

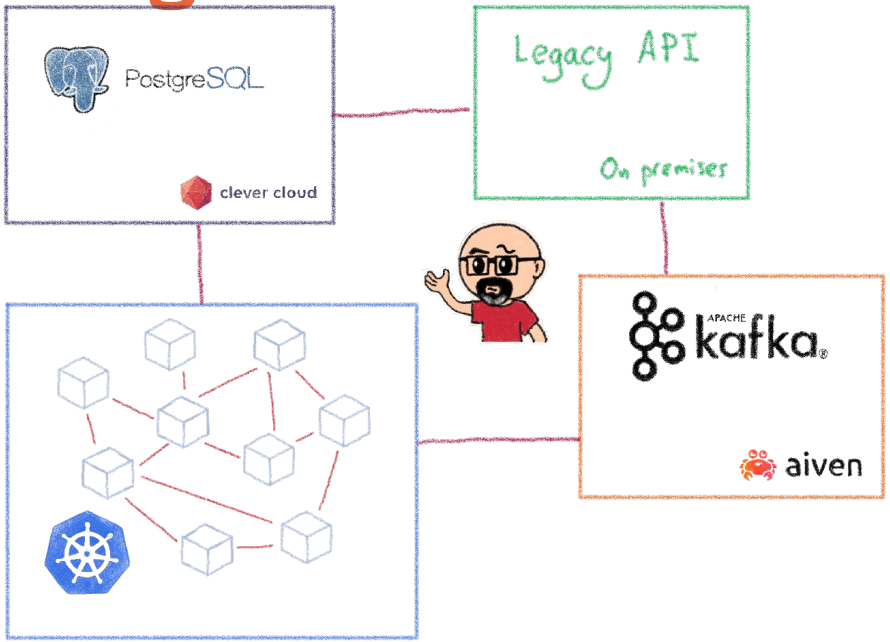
What could I use?

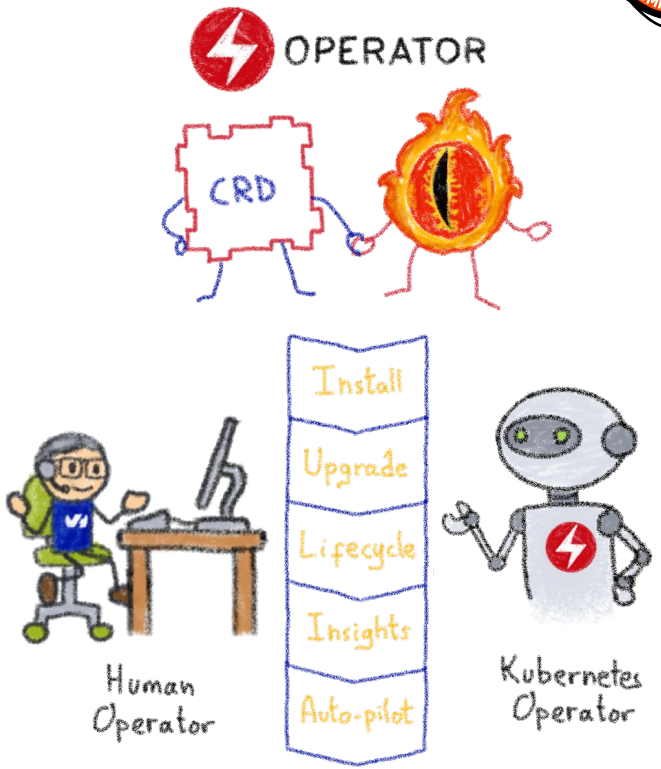


Don't put everything into Kubernetes

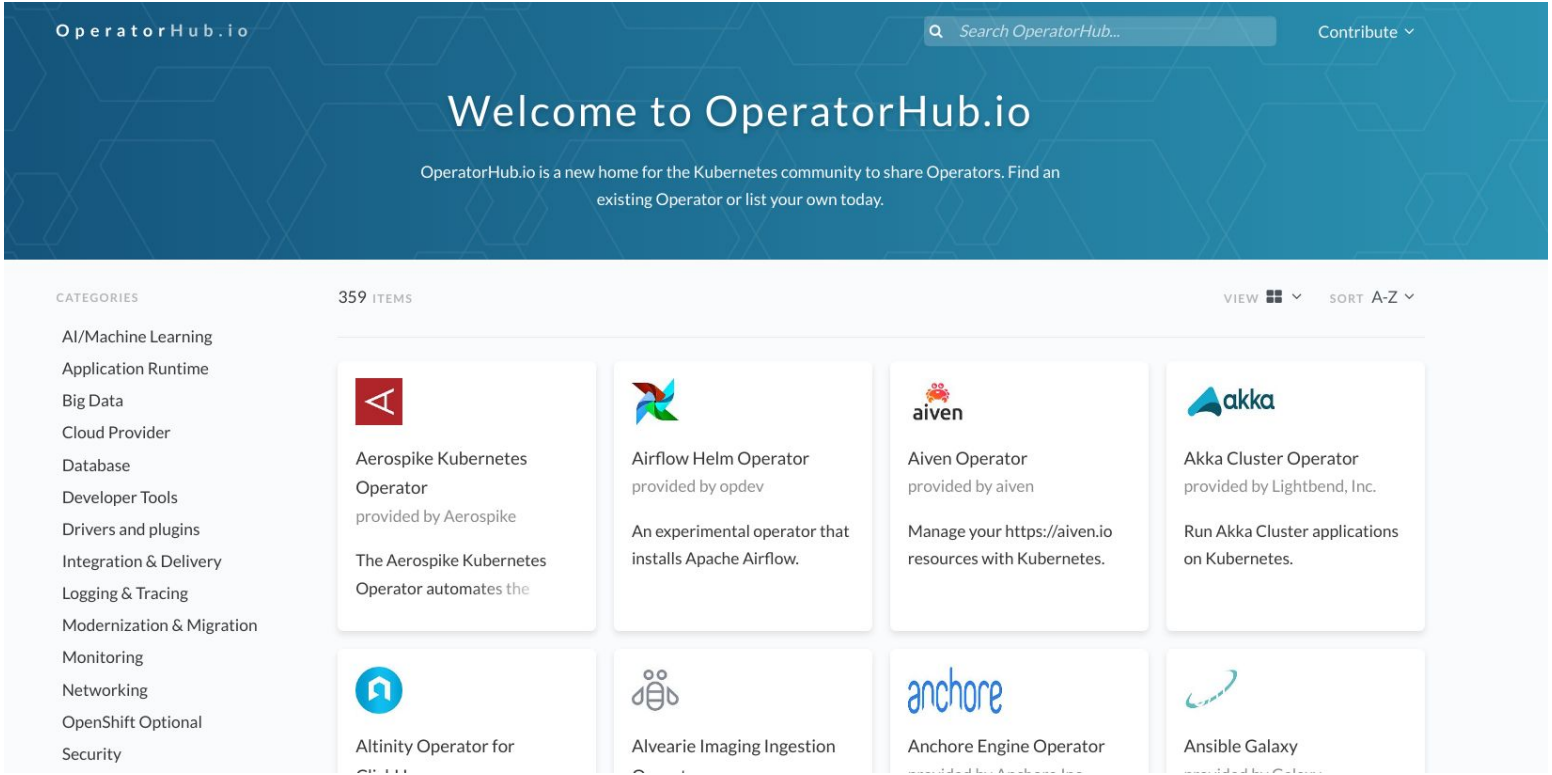


Operators simplify Kubernetes integration



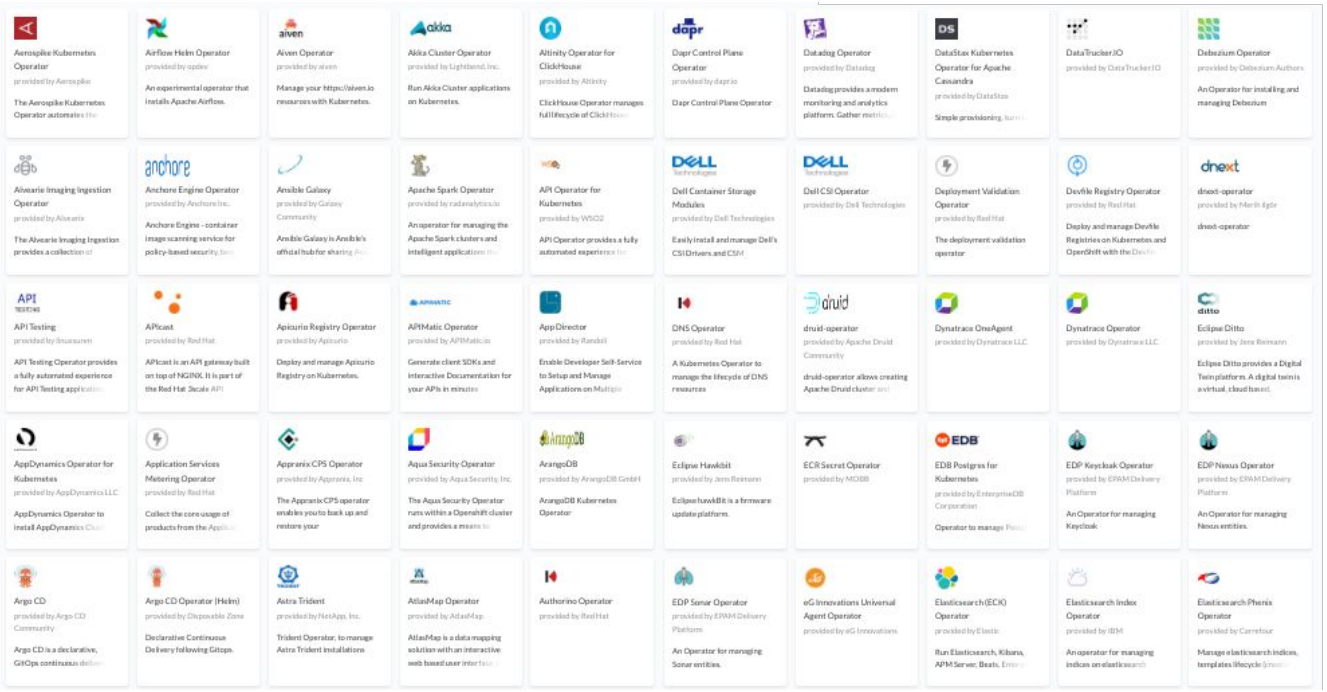


Lots of available operators



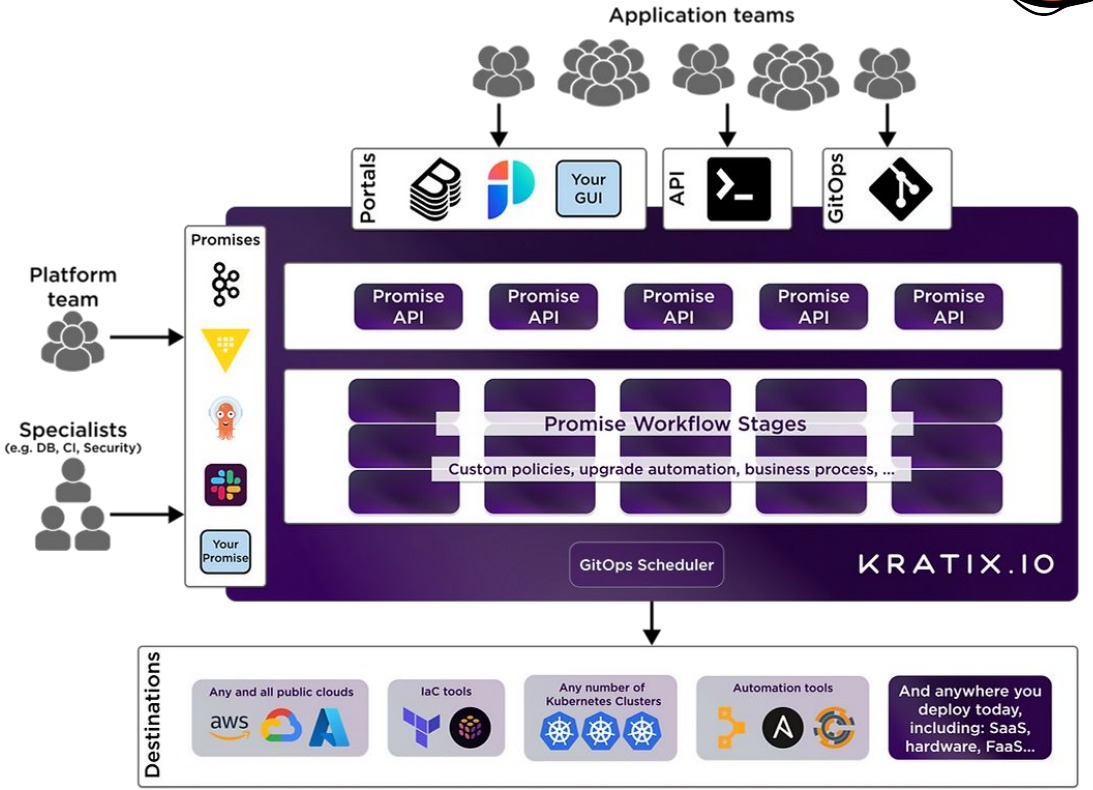
Hey JUG Summer Camp, operators ❤️ Java

So you have a bunch of operators...

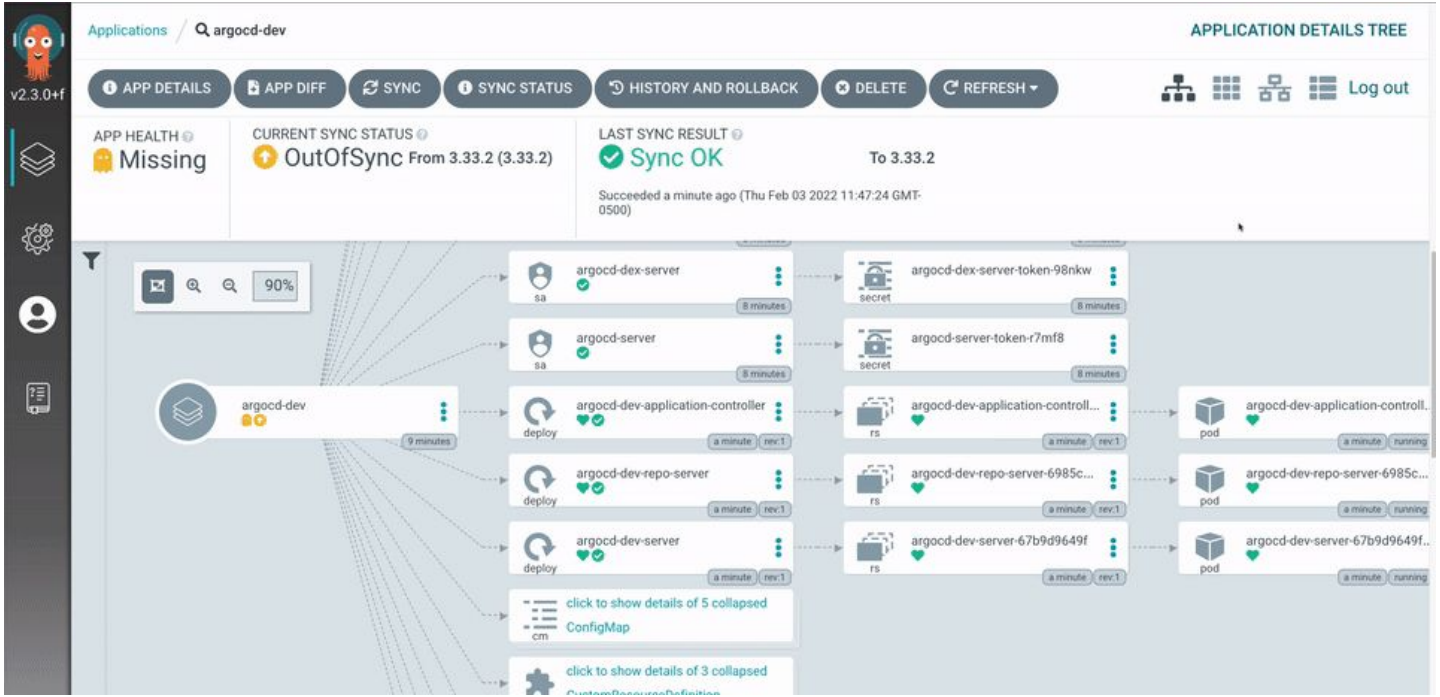


Too many options again...

Use a Platform Engineering framework

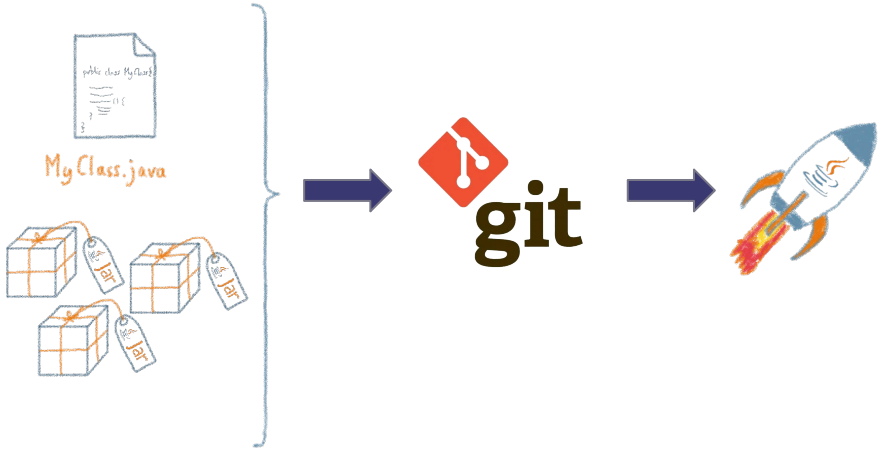


Set-up a robust CI/CD/GitOps pipeline

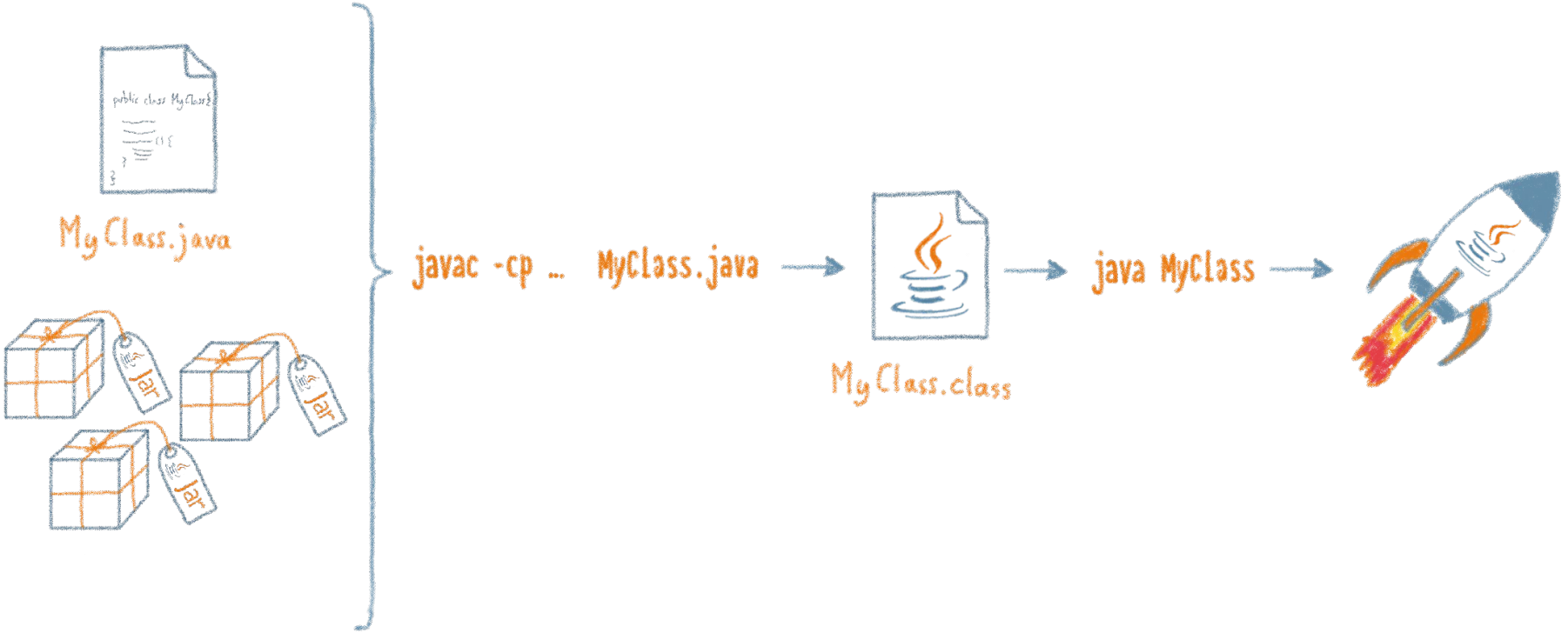


An IDP without Kubernetes

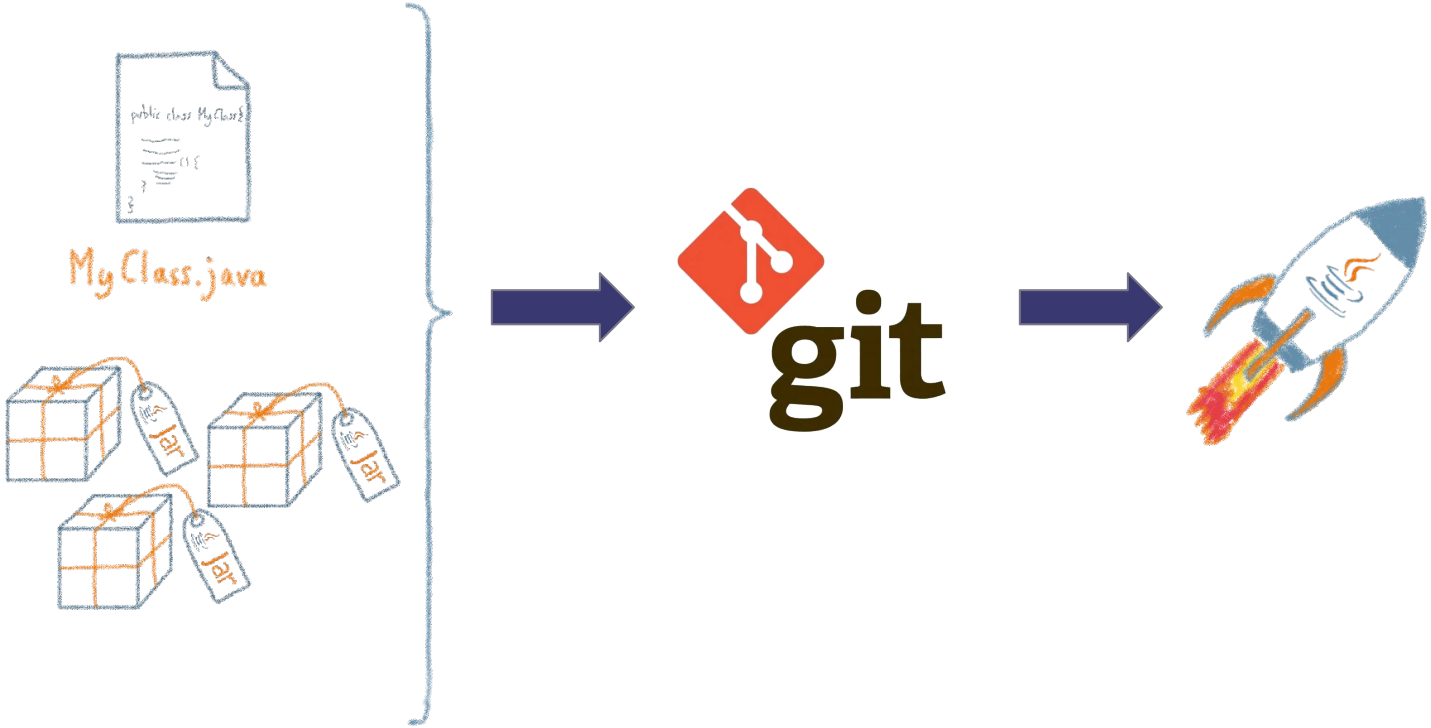
There is a life outside K8s...



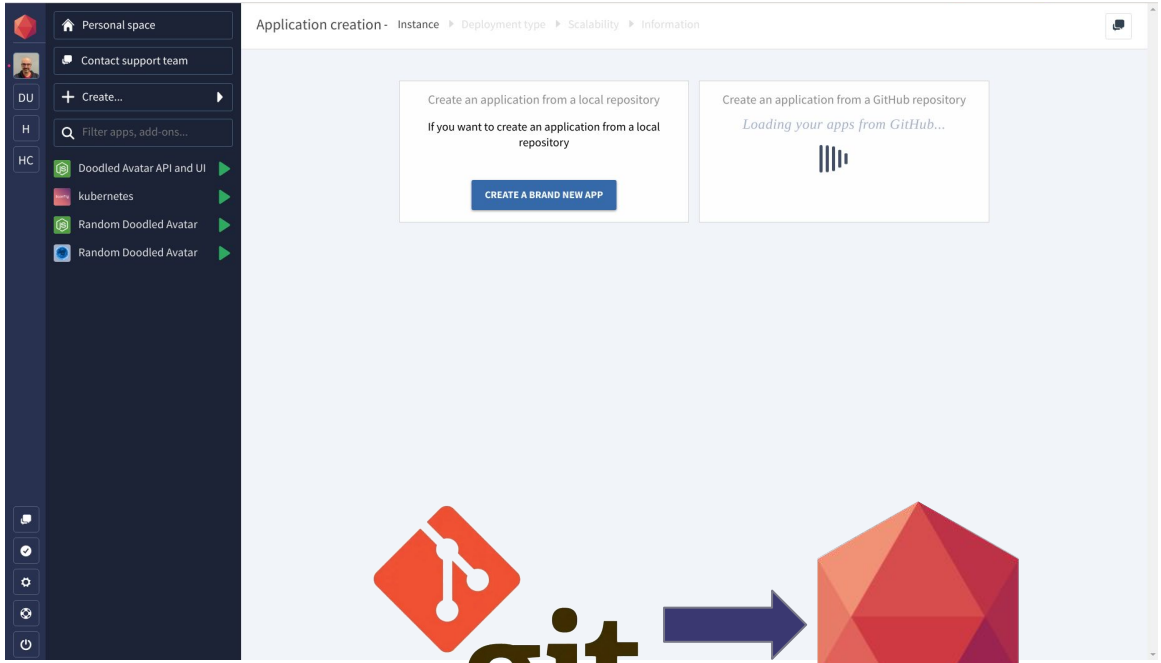
What if developers only developed?



PaaS as cornerstone of a dev-centered IDP



Clever Cloud



clever cloud

Some Links to go further

- There Is No Such Thing as a DevOps Engineer
- DevOps Topologies
- What is Platform Engineering
- Team Topologies
- Kratix
- Aiven
- Clever Cloud