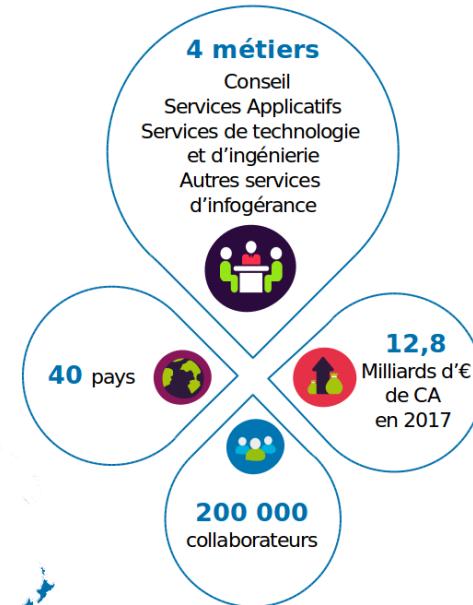
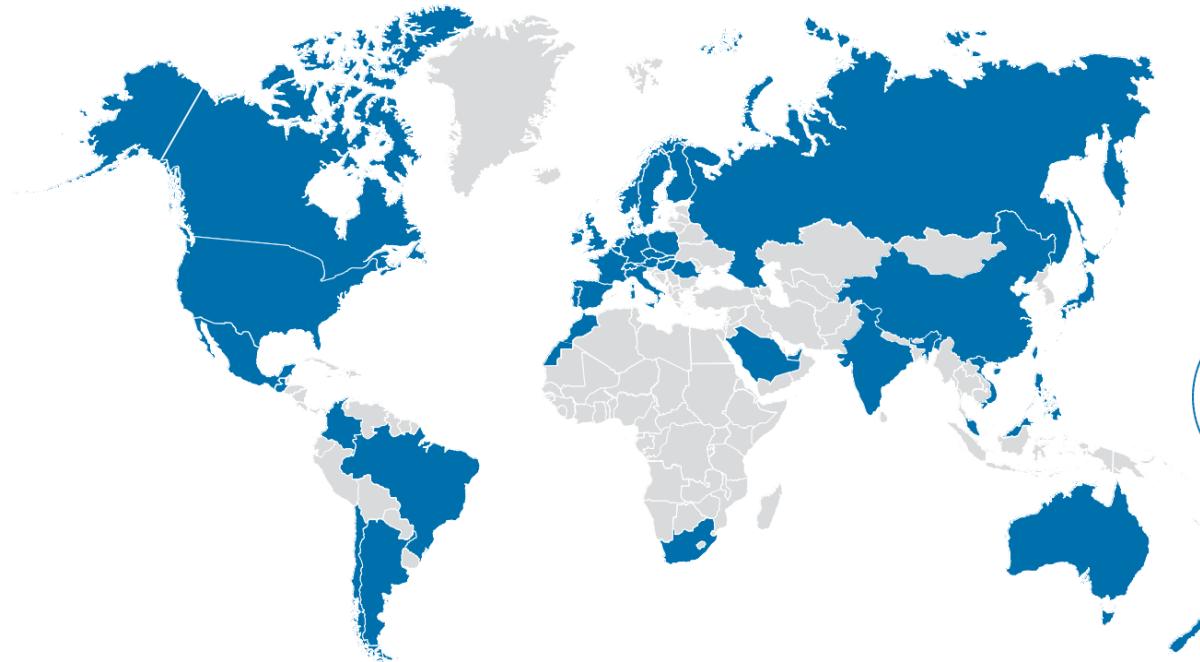




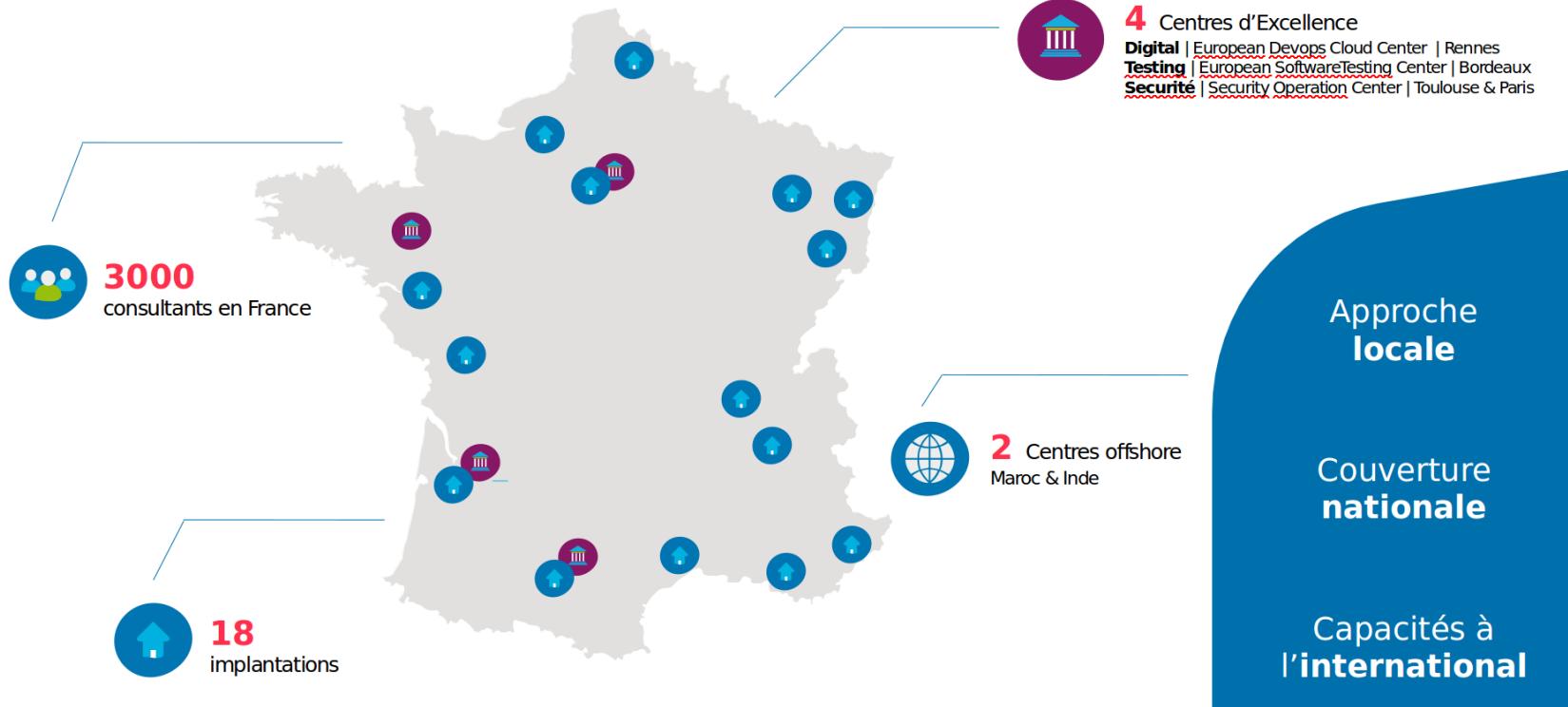


SOGETI, FILIALE DU GROUPE CAPGEMINI



*Septembre 2017

SOGETI EN FRANCE



INITIATIVES



Défi



- **Combattre la pollution de l'eau partout en France**
- **Affirmer notre dimension technologique et d'innovation**
- **Innover pour l'insertion des personnes en situation de handicap**

SOGETI LILLE

ATS (Application Testing Sécurité) : 100 consultants



Plus de 1000 collaborateurs réunis dans notre nouveau bâtiment à Lille

Jenkins Simple, facile, basique

- Afterwork
- Soirées d'agence
- Career Manager
- Communautés (Agile, Technique, Test, Career Manager, Féminine)

Stages 2018 - 2019

POLE INNOVATION



clarifai

- **Stage Microsoft « SmartBuilding »**

Environnement technologique :
C#, .Net, Unity 3D, Azure, Agile SCRUM, Azure DevOps

- **Stage chatbot « SoYouMan »**

Environnement technologique :
Python, Agile SCRUM, Tensor flow, Watson, Dialogflow

- **Stage Java « SoBee »**

Environnement technologique :
Java Spring, MySQL, Web responsive, Watson, Clarifai

- **Stage Java & Python « EDS »**

Environnement technologique :
Java, Tensor flow, Python, PHP, Agile SCRUM, Docker, GIT

Jenkinsfile, simple, basique



Jenkinsfile, simple, basique

- Théorie
- L'écosystème
- Pratique en TP autour de Jenkins

Jenkins ~~Difficile, complexe, difficile à apprendre~~, simple, basique

**Envoyer ses bugs
automatiquement et
plus vite en prod**

- un troll

Jenkins ~~Difficile, complexe, basique~~
Facile, simple, basique

Jenkins HISTORIQUE, simple, basique

1993 : L'expression "intégration continue" est employée pour recommander une intégration assez fréquente mais "programmée" dans le cadre d'un processus incrémental

Jenkins HISTORIQUE simple, basique

1996 : Technique du "Daily Build and Smoke Test" par Microsoft pour Windows NT 3.0

Jenkins HISTORIQUE, simple, basique

2000 : Description des principes pour l'intégration continue (Martin Fowler)

- utilisation d'un référentiel de gestion de versions du code source
- automatisation du processus de "build"
- présence de tests unitaires et fonctionnels automatisés (en lieu et place d'un "smoke test" manuel)
- exécution au minimum quotidienne de l'ensemble (build+test)
- gestion manuelle du processus d'intégration

Jenkins HISTORIQUE simple, basique

2001 : Cruise Control

Jenkins HISTORIQUE simple, basique

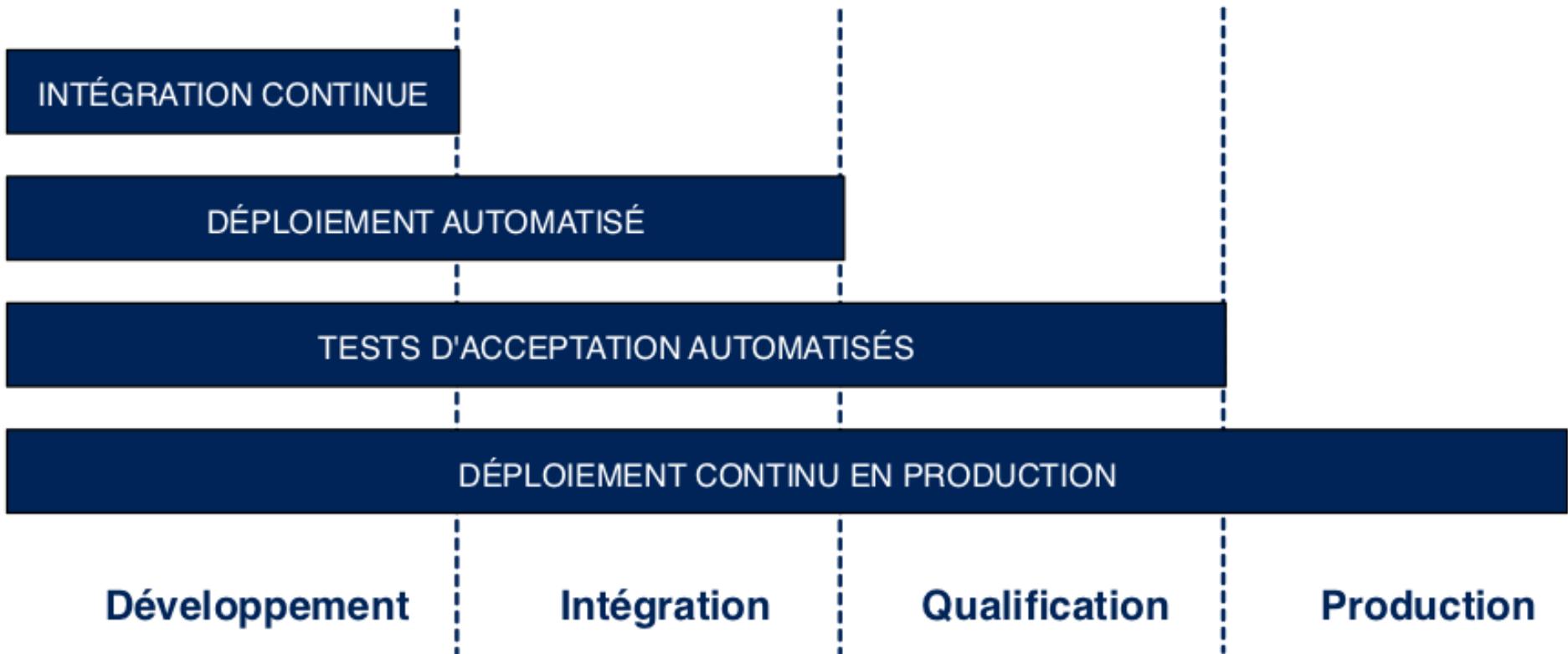
2007 : Déploiement continu

- <http://referentiel.institut-agile.fr/ci.html>
- <https://www.martinfowler.com/articles/originalContinuousIntegration.html>

Jenkins VOCABULAIRE
safe, simple, basique

Différence Intégration continue /
Déploiement continu

VOCABULAIRE



Jenkins PRINCIPES AGILES

- **Augmenter la fréquence de déploiement**
- **Fail fast, meilleure réactivité**
- **Pas de coupure de service**
- **Sprint 0**

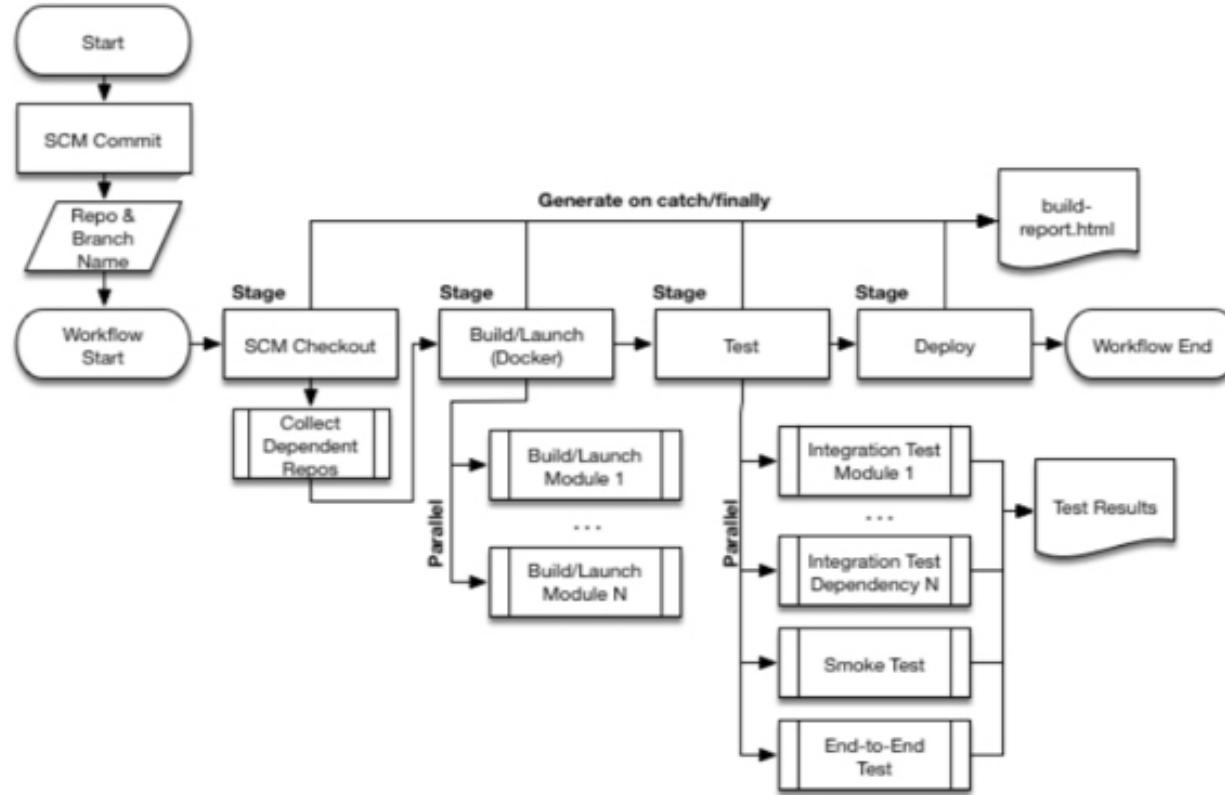
VERSIONING (SIX RETROUVEZ)

- **Patterns**
- **Tags**
- **Snapshot vs release**

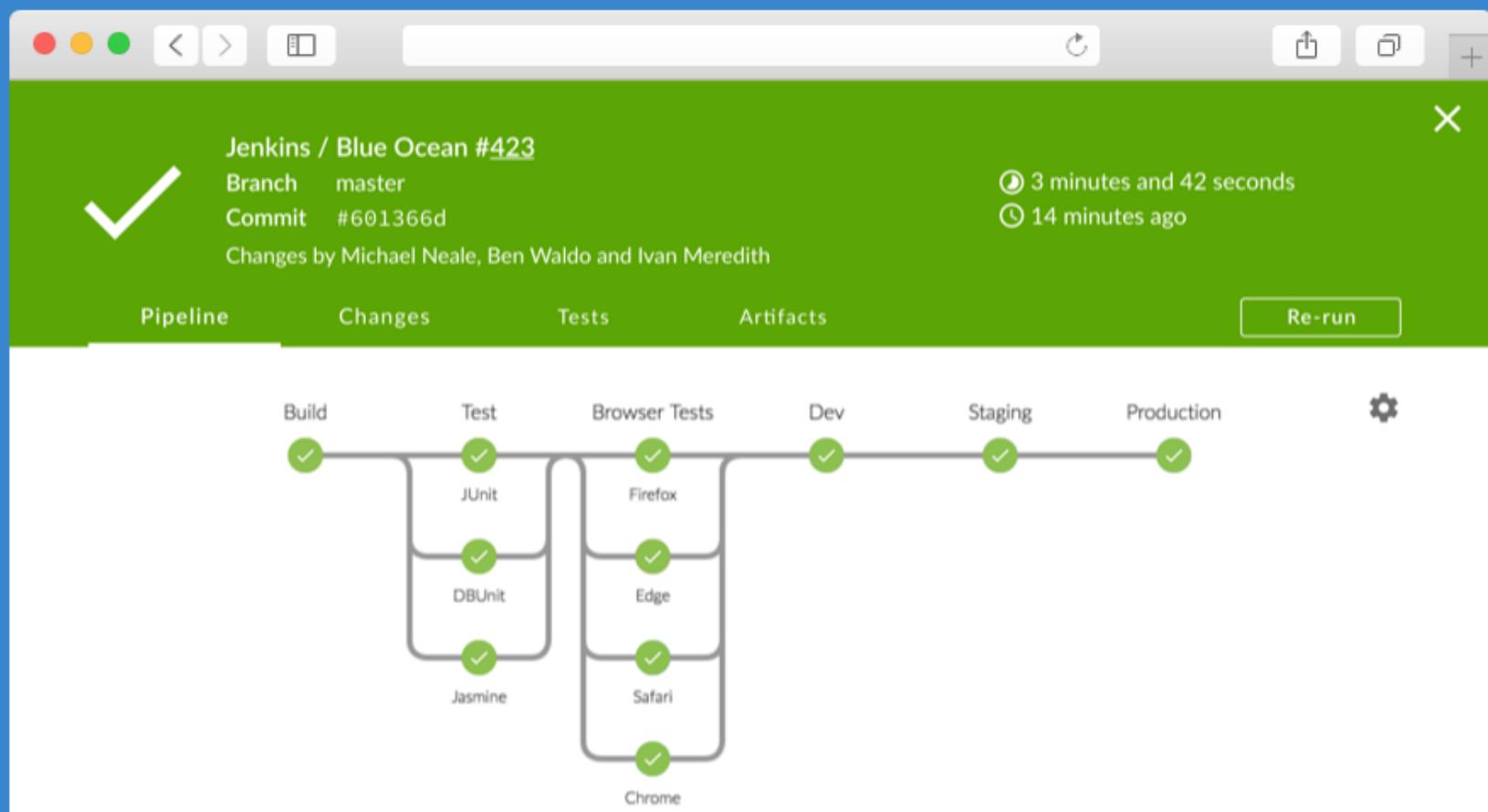
CE QU'ON PEUT AUTOMATISER

- **Compilation**
- **Tests unitaires**
- **Inspection du code (métrique de qualité)**
- **Construction des livrables**
- **Mise à disposition des livrables**
- **Déploiement sur environnement de test**
- **Tests d'intégration**
- **Tests UI sur différents navigateurs**
- **Test de charge**
- **Déploiement**
- **Publication de rapport**
- **Notification des résultats**
- **Génération de doc**
- **Release note**
- ...

CE QU'ON PEUT AUTOMATISER



Jenkinsfile, simple, basique



Jenkinsfile, simple, basique

The screenshot shows the Jenkins Blue Ocean interface for a pipeline named "Jenkins / Blue Ocean #423". The pipeline consists of several stages: Build, Test, Browser Tests, Dev, Staging, and Production. The Build stage has a green checkmark. The Test stage contains JUnit, DBUnit, and Jasmine sub-steps, all of which have green checkmarks. The Browser Tests stage contains Firefox, Edge, Safari, and Chrome sub-steps, with Firefox having a green checkmark and the others having red X's. The Dev, Staging, and Production stages are currently empty circles. A "Re-run" button is located in the top right corner of the pipeline summary.

Pipeline Summary:

- Branch: master
- Commit: #601366d
- Changes by: Michael Neale, Ben Waldo and Ivan Meredith
- Duration: 3 minutes and 42 seconds (14 minutes ago)

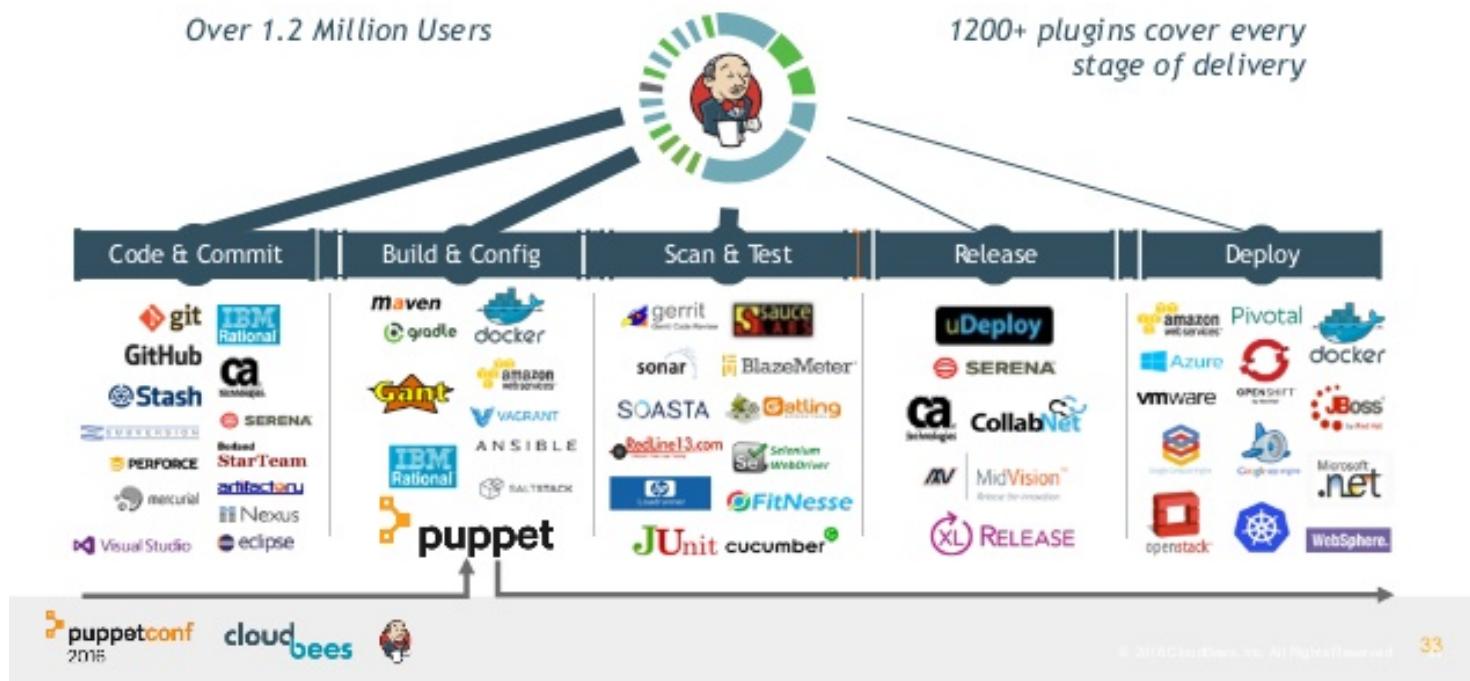
Build Log - Edge:

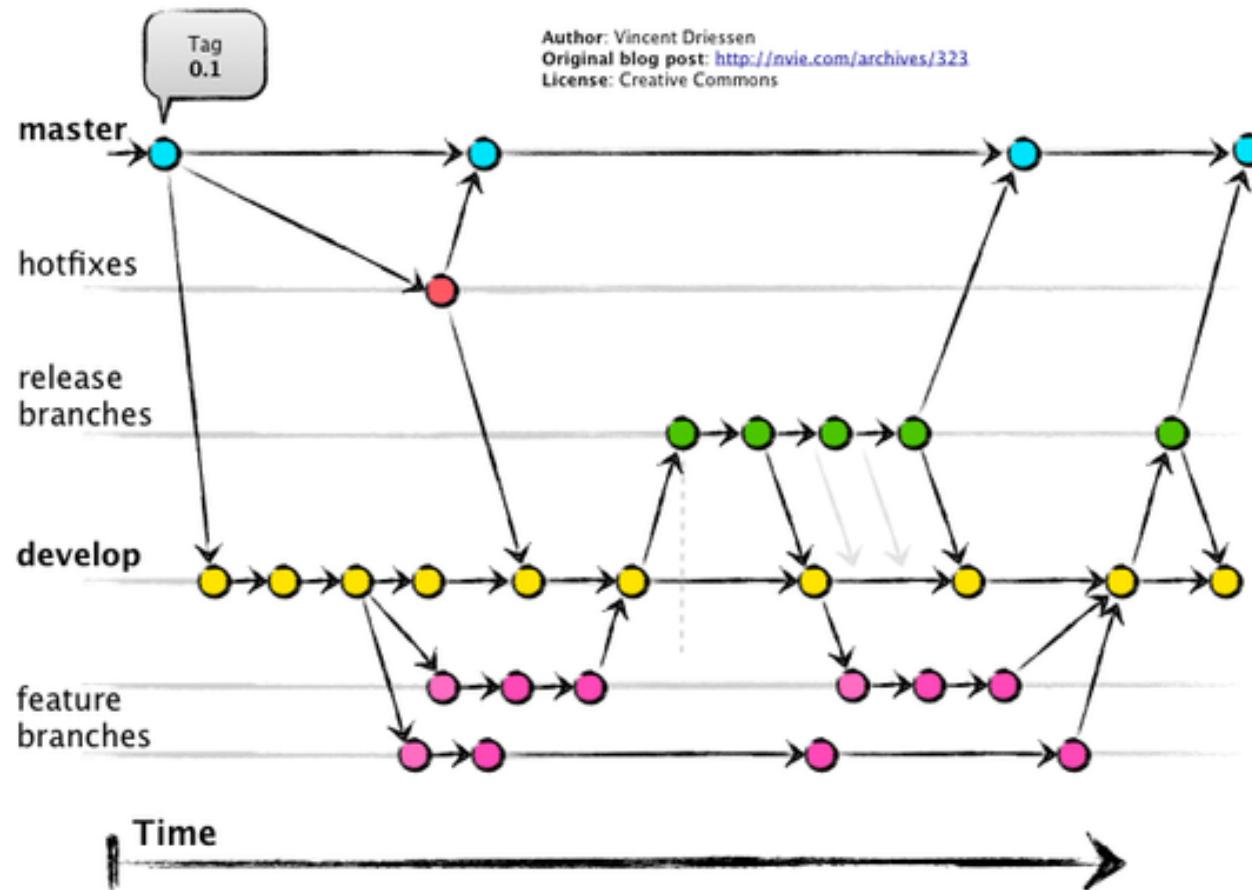
✓ > Start Docker container	3 minutes and 42 seconds
✓ > Warm maven caches	5 seconds
✓ > Install Java Tools	8 seconds
✗ > Maven	6 minutes and 12 seconds

Jenkinsfile, simple, basique

L'ÉCOSYSTÈME

Jenkins is the Hub of Continuous Delivery







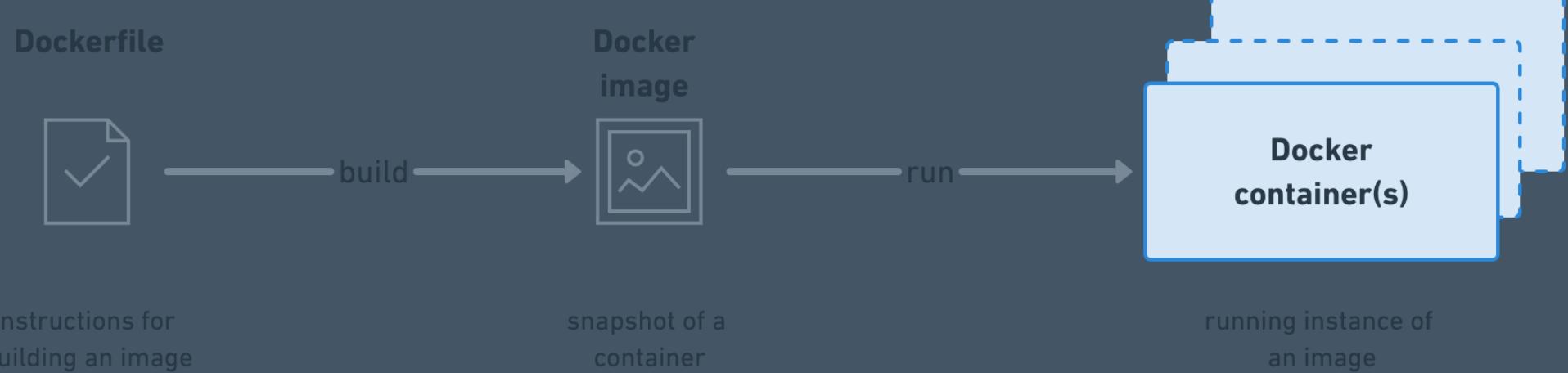
AVEC DOCKER Jenkins Re, simple, basique FAIRE



AVEC DOCKER Jenkins Re, simple, basique TOUX TOUT FAIRE

- Portabilité
- Modularité
- Sécurité
- Scalabilité

AVEC DOCKER Jenkins tuto REFLUX tout FAIRE



LA PRATIQUE AVEC JENKINS



<http://localhost:7777>

LA RÉVOLUTION
Jenkinsfile, simple, basique

Avant / après jenkinsfile

**LA RÉPONSE
Jenkinsfile, simple, basique**

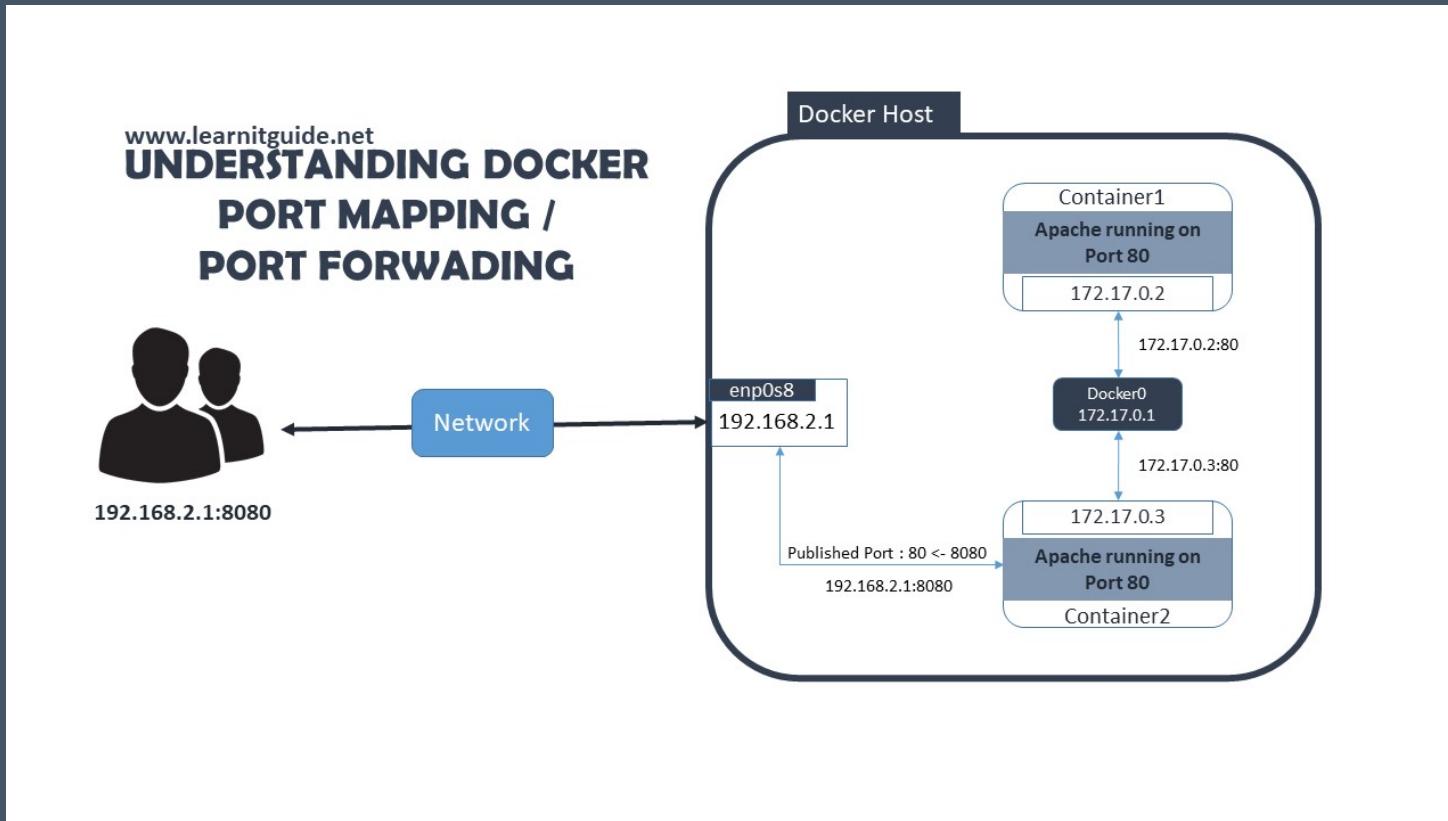
Ecriture du Jenkinsfile (Declarative > Scripted)

Jenkinsfile, simple, basique

DOCKER - COMMUNIQUER AVEC L'EXTÉRIEUR

- Port mapping
- Variables d'environnement
- Volumes

DOCKER - PORT MAPPING



DOCKER - VARIABLES D'ENVIRONNEMENT

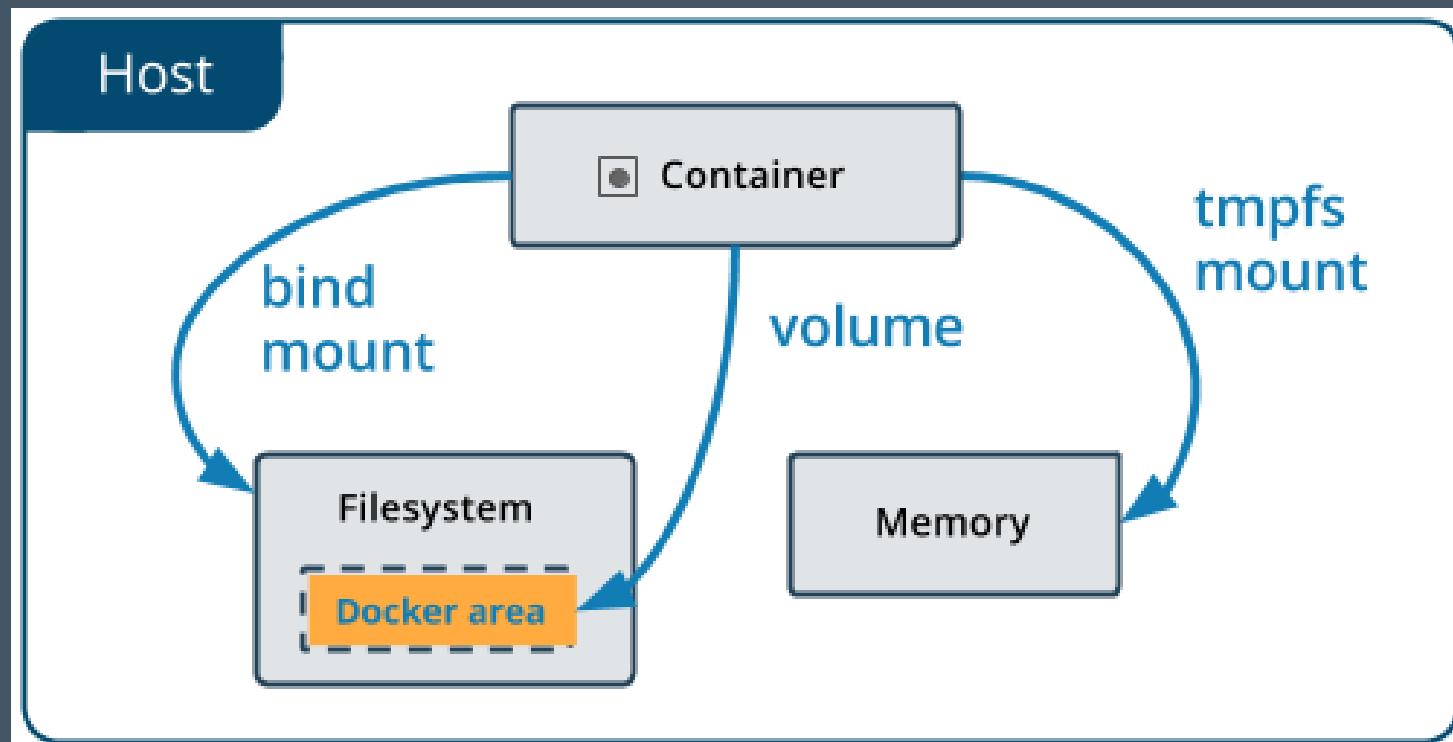
ENV vars

```
host> docker run -it -e MY_KEY=213323  
192.168.200.100:5000/redis:latest bash
```

```
container>echo $MY_KEY
```

DOCKER - VOLUMES

Quand un container est stoppé, les données sont supprimées.



DOCKER - VOLUMES

- **docker volume create my-vol**

DOCKER - VOLUMES

- **docker volume create my-vol**
- **docker volume ls**

DOCKER - VOLUMES

- **docker volume create my-vol**
- **docker volume ls**
- **docker volume inspect my-vol**

DOCKER - VOLUMES

- **docker volume create my-vol**
- **docker volume ls**
- **docker volume inspect my-vol**
- **docker run -d --name devtest -v my-vol:/app nginx:latest**

DOCKER - VOLUMES

- **docker volume create my-vol**
- **docker volume ls**
- **docker volume inspect my-vol**
- **docker run -d --name devtest -v my-vol:/app**
nginx:latest
- **docker exec devtest touch /app/toto**

DOCKER - VOLUMES

- **docker volume create my-vol**
- **docker volume ls**
- **docker volume inspect my-vol**
- **docker run -d --name devtest -v my-vol:/app**
nginx:latest
- **docker exec devtest touch /app/toto**
- **ls dans le point de montage (cf inspect)**

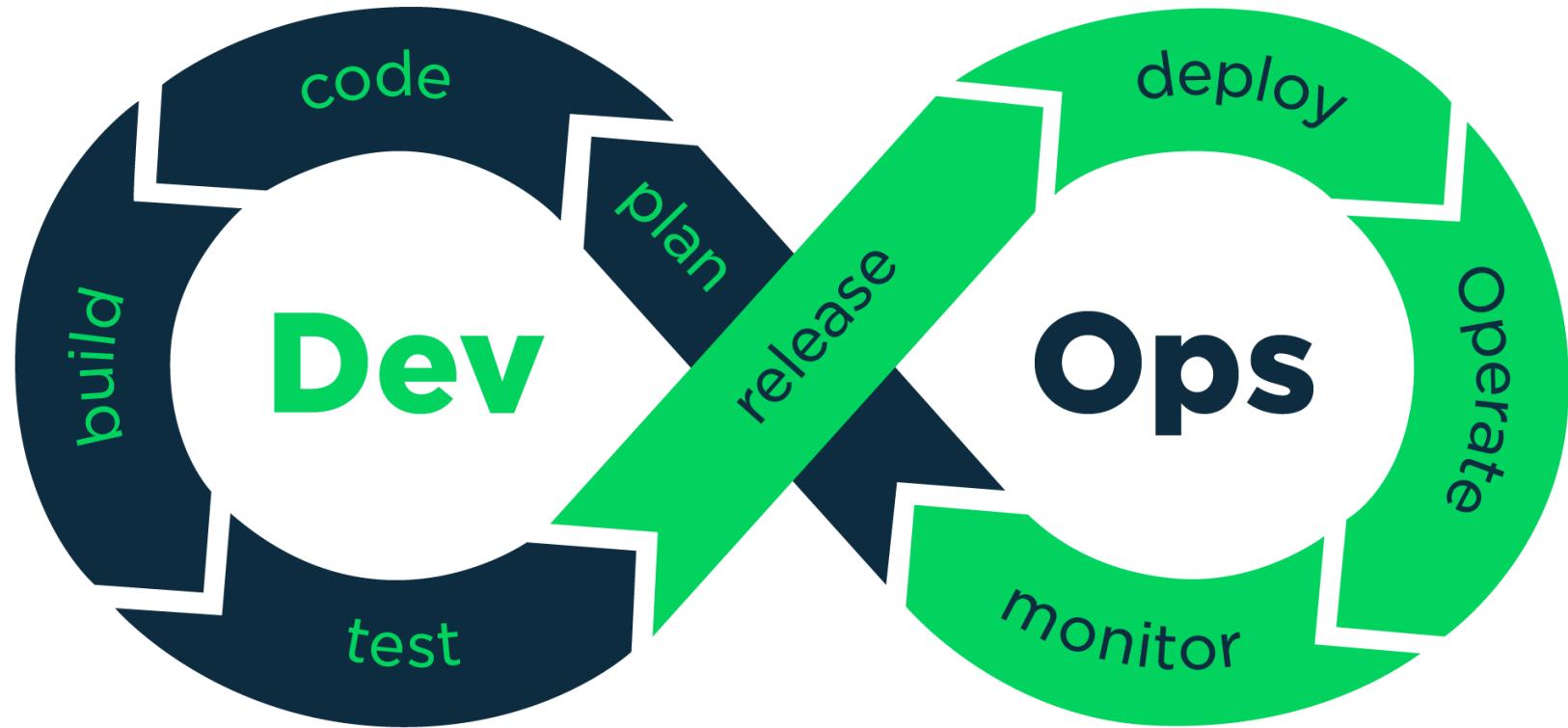
DEVOPS - DÉFINITION

Un ensemble de pratiques qui visent à réduire le Time to Market et à améliorer la qualité des produits logiciels, en réinventant la coopération entre DEV et OPS.

DevOps, c'est un modèle d'organisation, une culture, un assemblage de processus, d'outils et de patterns d'architecture.

- *octo.com*





CLOUD - DÉFINITION

CLOUD - DÉFINITION

- Maintenance des serveurs (Gestion de la performance déplacée, cf TP Jenkins)
- Modèle économique

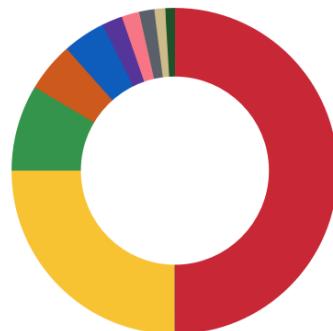
CLOUD - SCM



CLOUD - BUILD

Top 10 CI systems used with GitHub.com

(based on most used commit status contexts)



- Travis CI
- Circle CI
- Jenkins
- AppVeyor
- CodeShip
- Drone
- Semaphore CI
- Buildkite
- Wercker
- TeamCity



CI CD

CLOUD PROVIDERS



Google Cloud Platform



heroku



OPENSHIFT



CLOUD PROVIDERS

Public Cloud Leadership by Region – Q1 2018

Rank	Worldwide	North America	EMEA Region	APAC Region	Latin America
Leader	AWS	AWS	AWS	AWS	AWS
#2	Microsoft	Microsoft	Microsoft	Alibaba	Microsoft
#3	Google	Google	Google	Microsoft	Google
#4	Alibaba	IBM	IBM	Google	Salesforce
#5	IBM	Salesforce	Salesforce	Tencent	IBM

Source: Synergy Research Group

TP CLOUD

https://github.com/Arnotjevleesch/IGCI/blob/master/tp_cloud.ipynb