

# INTÉGRATION CONTINUE

GAGNEZ EN QUALITÉ SANS RISQUE !



GitLab

Clément BOURREAU

Outils, bonnes pratiques, qualité, retours d'expérience...

# Sommaire

---

## 1. Présentation

- 1. Définition, avantages, prérequis, outils

## 2. Qualité

- 1. IDE, commit, workflow de travail, revues de code

## 3. Pipeline CI

- 1. Configuration, images docker, MR, variables, services...

## 4. Pipeline CD

- 1. SSH, Rancher



# Intégration continue

Présentation

# Intégration continue - Définition



## **Intégration continue :**

L'intégration continue est un ensemble de pratiques utilisées en génie logiciel consistant à vérifier à chaque modification de code source que le résultat des modifications ne produit pas de régression dans l'application développée.



## **Livraison continue :**

La livraison continue est une approche d'ingénierie logicielle dans laquelle les équipes produisent des logiciels dans des cycles courts, ce qui permet de le mettre à disposition à n'importe quel moment. Le but est de construire, tester et diffuser un logiciel plus rapidement.

# Intégration continue - Avantages

- ❑ Réduction des risques
- ❑ Amélioration de la qualité
- ❑ Meilleure communication
- ❑ Gain de temps
- ❑ Disponibilité des livrables



# Intégration continue - Prérequis

- Contrôle de version
- Commits réguliers
- Tests automatisés
- Commits réguliers

In case of fire



1. git commit



2. git push



3. leave building

Programmers protocol

# Intégration continue - Outils



Team Foundation Server



CODESHIP



GitLab



Travis CI



Jenkins



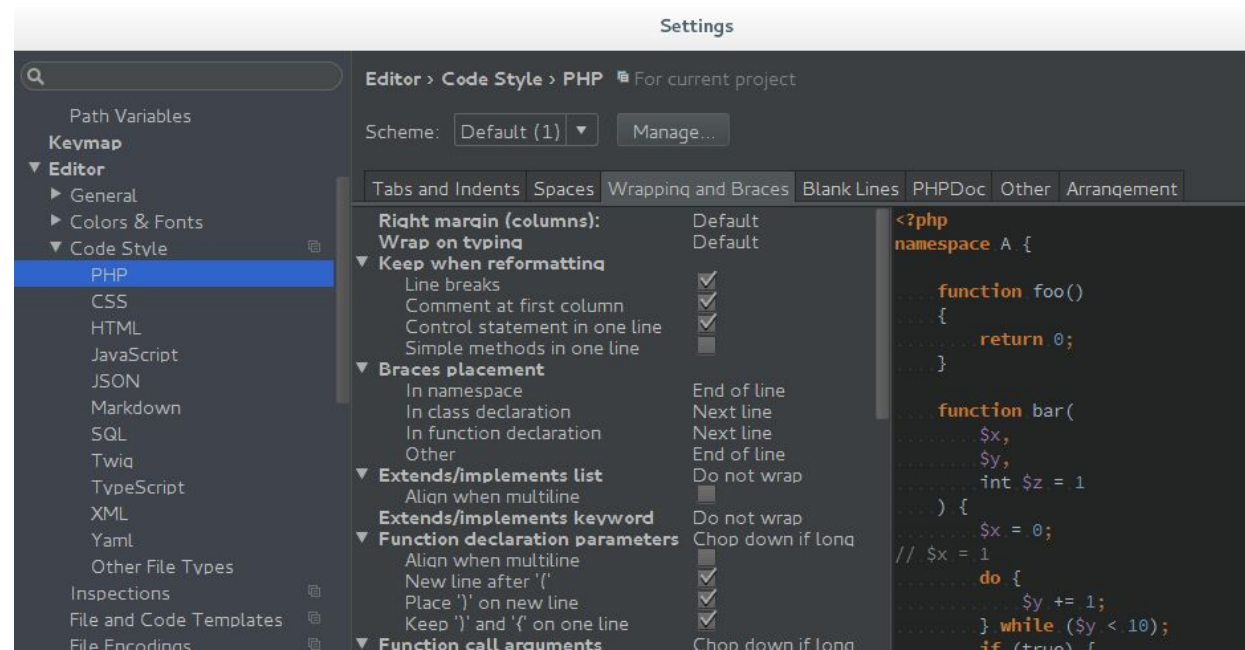
# Intégration continue

Qualité



# Qualité - IDE

- Un code style unique pour tout le monde
- Réarranger son code avant commit Ctrl+Shift+Alt
- Vérifier ses modifications avant de commit



# Qualité - Commit

- Indiquer la référence du ticket si possible
- Uniformiser les messages (verbe, langue...)
- Utiliser 50 caractères max. pour l'en-tête

<Subject>  
[<BLANK LINE>]  
[<Message Body>]  
[<BLANK LINE>]  
[<Message Footer>]

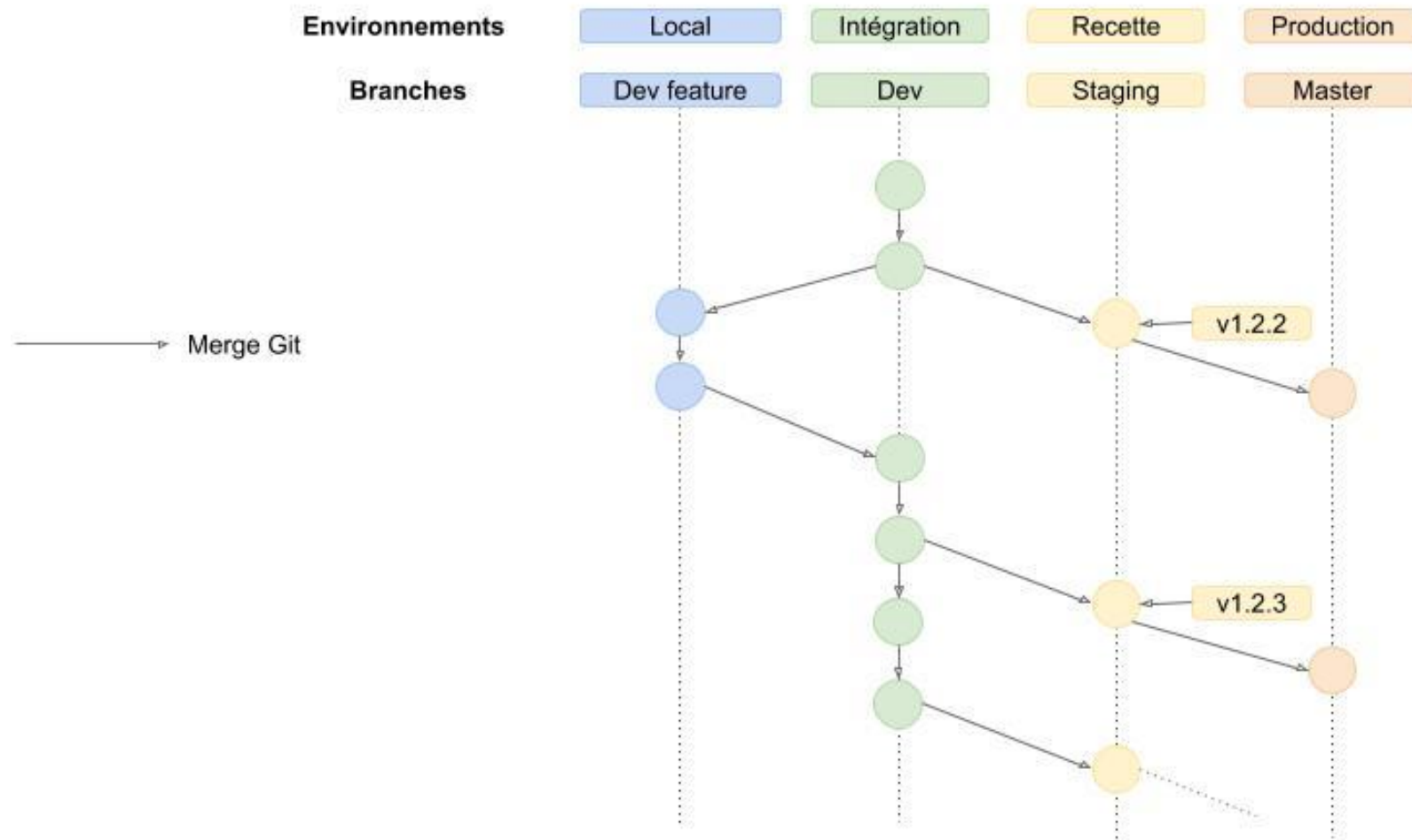


	COMMENT	DATE
○	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
○	ENABLED CONFIG FILE PARSING	9 HOURS AGO
○	MISC BUGFIXES	5 HOURS AGO
○	CODE ADDITIONS/EDITS	4 HOURS AGO
○	MORE CODE	4 HOURS AGO
○	HERE HAVE CODE	4 HOURS AGO
○	AAAAAAA	3 HOURS AGO
○	ADKFJSLKDFJSDKLFJ	3 HOURS AGO
○	MY HANDS ARE TYPING WORDS	2 HOURS AGO
○	HAAAAAAAANDS	2 HOURS AGO

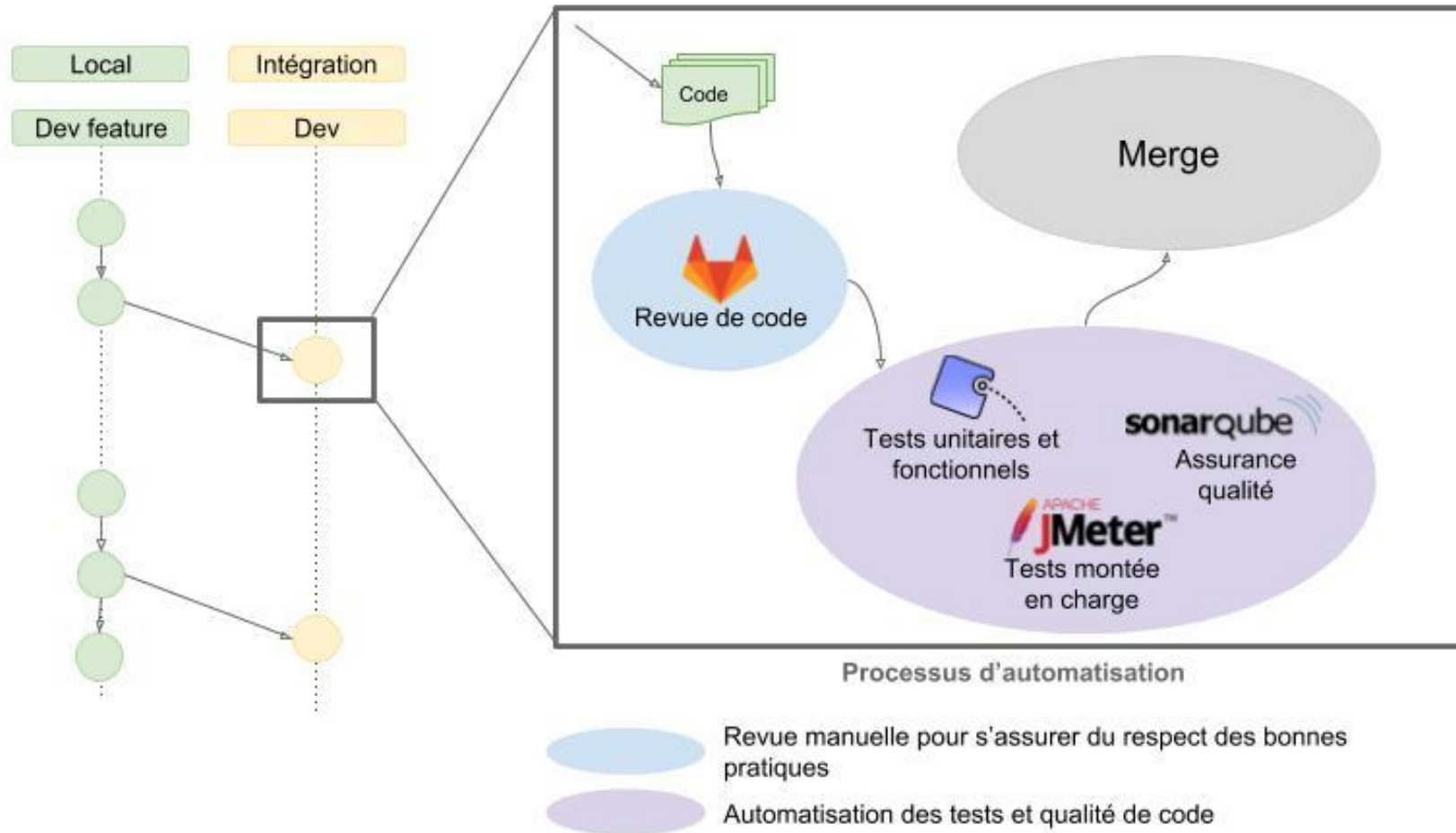
AS A PROJECT DRAGS ON, MY GIT COMMIT  
MESSAGES GET LESS AND LESS INFORMATIVE.

- Exemple : commit message convention AngularJS :  
<https://gist.github.com/stephenparish/9941e89d80e2bc58a153>

# Qualité – Workflow de travail



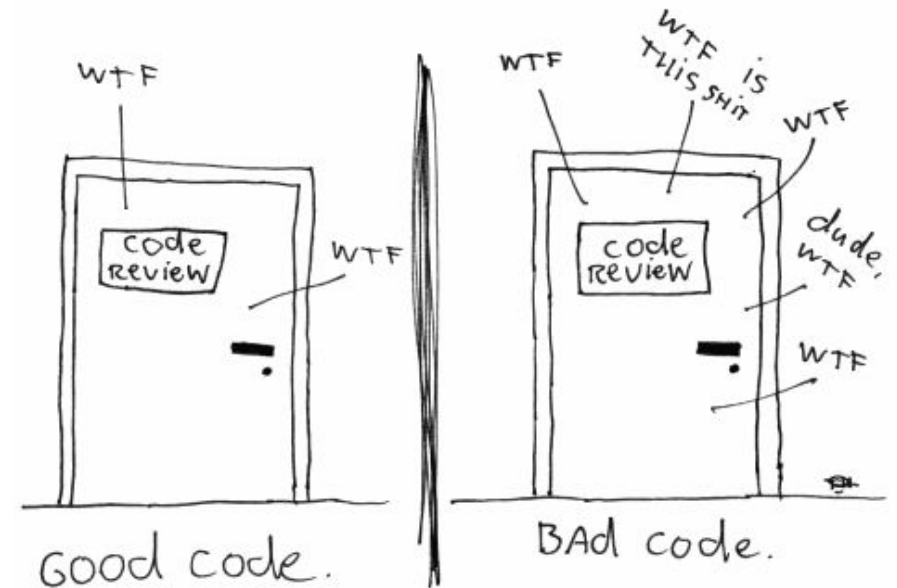
# Qualité – Industrialiser les développements



# Qualité – Revue de code

- Améliore la qualité et la maintenabilité du code
- Permet une meilleure vue d'ensemble du projet
- Améliore le partage de connaissance

The ONLY valid measurement  
of code quality: WTFs/minute

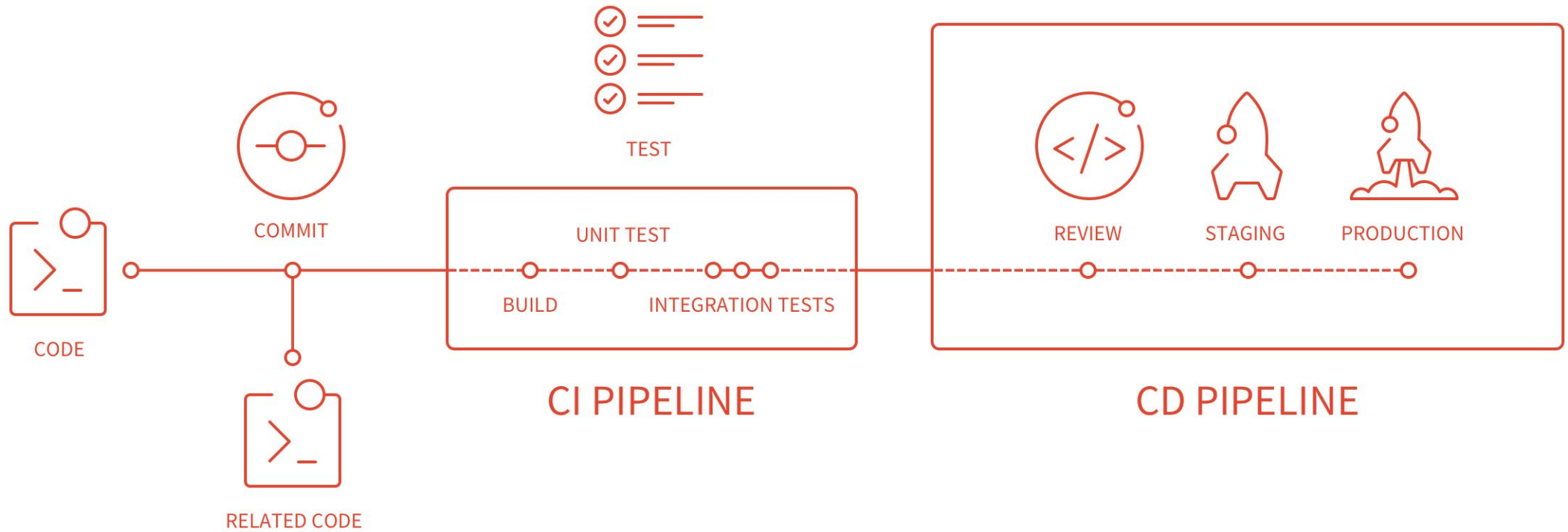




# Intégration continue

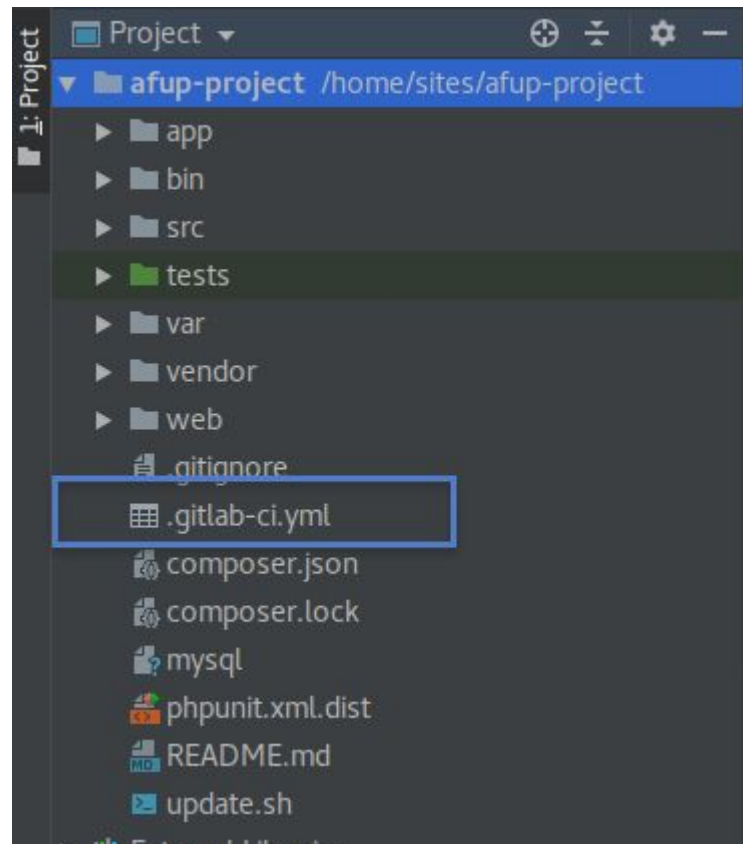
Pipeline CI

# Pipeline – késako ?



# Pipeline – Configuration

- Ajout d'un fichier `.gitlab-ci.yml` à la racine





# Pipeline – Image docker

- Utilisation d'images docker : [https://hub.docker.com/\\_/php/](https://hub.docker.com/_/php/)

OFFICIAL REPOSITORY

php ☆

Last pushed: 3 days ago

---

Repo Info Tags

Short Description

While designed for web development, the PHP scripting language also provides general-purpose use.

Docker Pull Command

```
docker pull php
```

Full Description

**Note:** the description for this image is longer than the Hub length limit of 25000, so has been trimmed. The full description can be found at <https://github.com/docker-library/docs/tree/master/php/README.md>. See [docker/hub-beta-feedback#238](https://github.com/docker-library/docs/tree/master/php/README.md) for more information.

**Supported tags and respective Dockerfile links**

- 7.3.0RC5-cli-stretch, 7.3-rc-cli-stretch, rc-cli-stretch, 7.3.0RC5-stretch, 7.3-rc-stretch, rc-stretch, 7.3.0RC5-cli, 7.3-rc-cli, rc-cli, 7.3.0RC5, 7.3-rc, rc ([7.3-rc/stretch/cli/Dockerfile](#))
- 7.3.0RC5-apache-stretch, 7.3-rc-apache-stretch, rc-apache-stretch, 7.3.0RC5-apache, 7.3-rc-apache, rc-apache ([7.3-rc/stretch/apache/Dockerfile](#))
- 7.3.0RC5-fpm-stretch, 7.3-rc-fpm-stretch, rc-fpm-stretch, 7.3.0RC5-fpm, 7.3-rc-fpm, rc-fpm ([7.3-rc/stretch/fpm/Dockerfile](#))
- 7.3.0RC5-zts-stretch, 7.3-rc-zts-stretch, rc-zts-stretch, 7.3.0RC5-zts, 7.3-rc-zts, rc-zts ([7.3-rc/stretch/zts/Dockerfile](#))
- 7.3.0RC5-cli-alpine3.8, 7.3-rc-cli-alpine3.8, rc-cli-alpine3.8, 7.3.0RC5-alpine3.8, 7.3-rc-alpine3.8, rc-alpine3.8, 7.3.0RC5-alpine, 7.3-rc-alpine, rc-alpine, 7.3.0RC5, 7.3-rc, rc

# Pipeline – Job

- Job de test simple avec image docker personnalisée

```
test:  
  image: registry.gitlab.com/tcs-poitiers-projects/docker-php:latest  
  before_script:  
    - composer install  
  script:  
    - ./vendor/bin/simple-phpunit
```

# Pipeline – Execution sur un runner

Récupération du  
service


























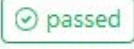











Récupération de  
l'image du job

Clone du repo  
par défaut

```
Running with gitlab-runner 11.5.0-rc1 (e900028d)
  on docker-auto-scale fa6cab46
Using Docker executor with image php ...
Starting service mysql:latest ...
Pulling docker image mysql:latest ...
Using docker image sha256:2dd01afbe8df1fe326f6609c56b08beefc6bf254d28993263da188b8fbf1254d for mysql:latest
Waiting for services to be up and running...
Pulling docker image php ...
Using docker image sha256:690589dc27e9bd6e73d0a00676a6618b80d38caba57877276d982e673162202e for php ...
Running on runner-fa6cab46-project-9380240-concurrent-0 via runner-fa6cab46-srm-1542135725-8260a000...
Cloning repository...
Cloning into '/builds/cboureau/afup-project'...
Checking out 0133afd3 as dev-demo...
Skipping Git submodules setup
Checking cache for default...
```



Identification  
du runner


# Pipeline – Suivi des builds



Status	Pipeline	Commit	Stages	
 failed	#36484979 by  latest	 dev-demo  67091e33  add error that no one will noti...	  	 00:00:42  in 1 minute   
 passed	#36484140 by 	 dev-demo  b5a02274  test d'erreur + sample all stages	  	 00:00:45  7 minutes ago  
 passed	#36483140 by 	 dev-demo  de79690f  Add environments	  	 00:00:58  19 minutes ago  


- ❑ Echoue en cas d'exit code



# Pipeline – Merge request

 **Request to merge** `dev-demo`  **into** `dev`  
The source branch is 3 commits behind the target branch

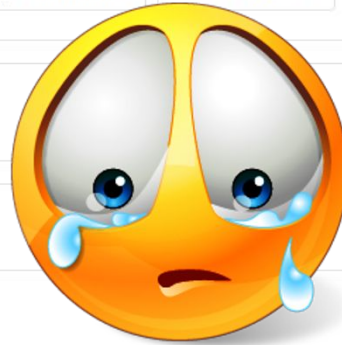
[Open in Web IDE](#) [Check out branch](#) 



 **Pipeline #36578990 failed for e4beb6ec on dev-demo** 


 No Approval required



 **Merge** ☐ Remove source branch ☐ Squash commits  [Modify commit message](#)


You can merge this merge request manually using the [command line](#)





 **Request to merge** `dev-demo`  **into** `dev`  
The source branch is 3 commits behind the target branch

[Open in Web IDE](#) [Check out branch](#) 

 **Pipeline #36579637 passed for a0e28273 on dev-demo** 

 No Approval required

 **Merge** ☐ Remove source branch ☐ Squash commits  [Modify commit message](#)

You can merge this merge request manually using the [command line](#)



# Pipeline – Services

- Se base sur les images docker
- Supporte MySQL, Redis, PostgreSQL...
- Structure très similaire avec docker-compose

```
services:
  - mysql:latest
variables:
  MYSQL_DATABASE: $MYSQL_DATABASE
  MYSQL_ROOT_PASSWORD: $MYSQL_ROOT_PASSWORD
```

# Pipeline – Variables CI

- S'appuie sur des variables CI pour le stockage des données sensibles

## Variables ?

Collapse

Variables are applied to environments via the runner. They can be protected by only exposing them to protected branches or tags. You can use variables for passwords, secret keys, or whatever you want.

MYSQL_DATABASE	*****	Protected	<input type="checkbox"/>	⊖
MYSQL_ROOT_PASSWORD	*****	Protected	<input checked="" type="checkbox"/>	⊖
Input variable key	Input variable value	Protected	<input type="checkbox"/>	

Save variables

Reveal values

- S'utilise sous le format : `$MYSQL_DATABASE`

# Pipeline – Optimisation

- Cache : permet de réduire le temps d'exécution des jobs

```
cache:  
  paths:  
    - vendor/  
    - .composer/cache
```

- Images dockers adaptés sur le Registry Gitlab

## Container Registry

With the Docker Container Registry integrated into GitLab, every project can have its own space to store its Docker images.

Learn more about [Container Registry](#).

[tcs-poitiers-projects/docker-php](#)

Tag	Tag ID	Size	Created	
1eb331c51b9c086c2265fba2d39f3ab36ac46a4a-dev	d736c894a	184.48 MiB	1 month ago	
1eb331c51b9c086c2265fba2d39f3ab36ac46a4a	d589ce1f6	170.18 MiB	1 month ago	
latest-dev	d7e60acdb	184.71 MiB	1 month ago	
latest	c593b9bbe	170.03 MiB	2 weeks ago	



# Pipeline – Restrictions

- Selon la branche, le tag, le mode de déclenchement (web, schedule...)

`only` and `except`

- Selon le résultat / le type de déclenchement

`when`

- Selon l'étape de la pipeline

`stage`




# Intégration continue

Pipeline CD

# Pipeline – Déploiement par SSH

Nécessaire pour  
permettre la  
connexion SSH



```
deploy-prod:
  stage: deploy
  only:
    - master
  when: manual
  before_script:
    - 'which ssh-agent || ( apt-get update -y && apt-get install openssh-client -y )'
    - eval $(ssh-agent -s)
    - echo "$HOST_PRIVATE_KEY" | tr -d '\r' | ssh-add - > /dev/null
    - mkdir -p ~/.ssh
    - chmod 700 ~/.ssh
    - echo -e "Host *\n\tStrictHostKeyChecking no\n\n" > ~/.ssh/config
  variables:
    APP_PATH: '/home/sites/afup-project/'
  script:
    # Checkout & pull
    - ssh -p22 $SSH_USER@$HOST "cd $APP_PATH && git checkout master && git reset --hard origin/master && git pull"
    # Update app
    - ssh -p22 $SSH_USER@$HOST "cd $APP_PATH && bash update.sh"
  environment:
    name: production
    url: https://afup-poitiers.fr
```

# Pipeline – Upgrade stack Rancher

```
upgrade-image:  
  image: cdrx/rancher-gitlab-deploy  
  stage: upgrade  
  only:  
    - master  
  when: on_success  
  script:  
    - upgrade --stack Teamweek-Rancher --service php --environment Main --finish-upgrade
```

# Merci

---

