

Écrire et débbugger un fichier `.gitlab-ci.yml`



LA FORGE

des Communs Numériques Éducatifs



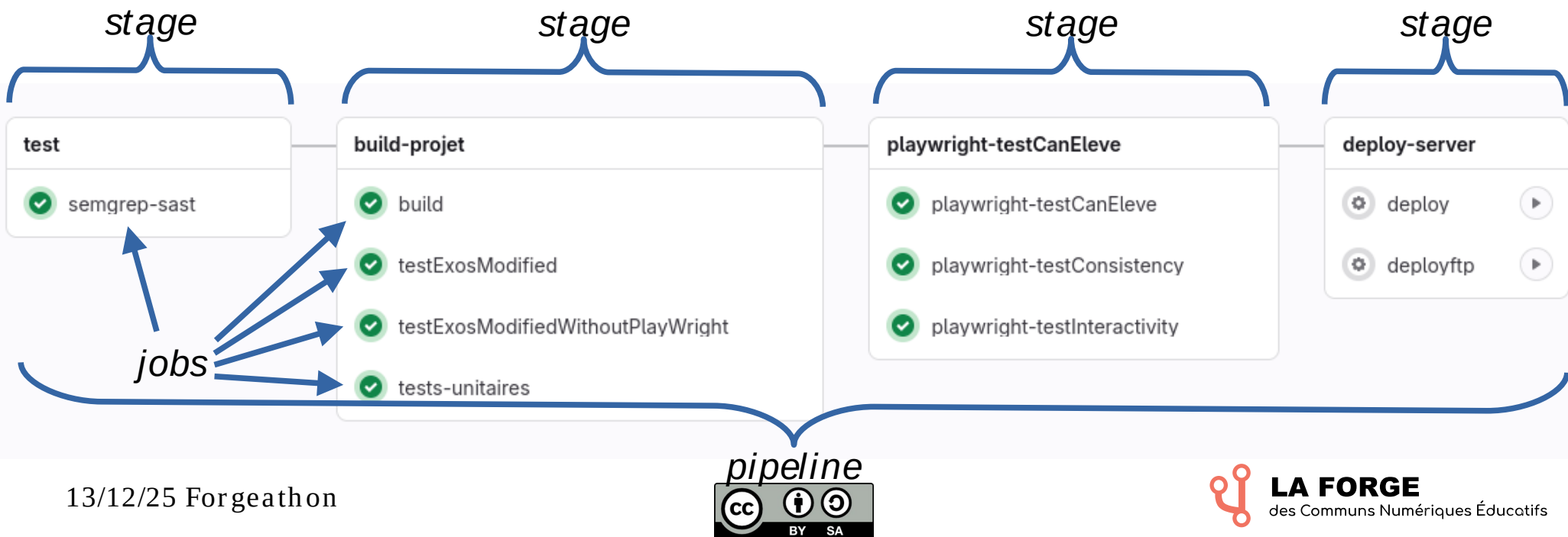
GitLab

Louis Paternault — 13/12/25



CI/CD

Exécution de code sur le serveur de la forge à chaque *commit* (validation) ou *push*.



Un premier fichier .gitlab-ci.yml

```
image: alpine
pages:
  script:
    - mkdocs build
  pages: true
```

Étapes (*stages*)

- Trois étapes par défaut : build/test/deploy
- Configurable avec le mot-clef stages.

```
stages:  
  - pretest  
  - build  
  - test  
  - pages
```

https://forge.apps.education.fr/komit/recettes/-/blob/main/.gitlab-ci.yml?ref_type=heads

D'autres commandes

artifact : ensemble de fichiers qui sont conservés après exécution du *job*, et conservés pour les autres *jobs* du *stage*.

```
artifacts:  
  paths:  
    - dist/
```

<https://docs.gitlab.com/ci/yaml/>

Variables

- Ont une influence sur l'exécution
- Sont utilisées dans les scripts
- Permettent d'adapter les scripts à l'environnement d'exécution
- Permettent de cacher des secrets

https://docs.gitlab.com/ci/variables/predefined_variables/

Quand exécuter les *pipelines* et les *jobs* ?

- *workflow* pour contrôler l'exécution des *pipelines*
- *rules* pour contrôler l'exécution des *jobs*.

Débugger

https://forge.apps.education.fr/outils/typst-dockerfile/-/commits/main/.gitlab-ci.yml?ref_type=heads

```
docker run -i
  --volume $(pwd) : /test
  -t node
  /bin/bash
```

Au travail !

<https://forge.apps.education.fr/bacasable-lp/gitlab-ci>

Merci pour votre attention !

