

Jobs pipeline

Deux jobs jenkins par projet : un job projectname et un job projectname-wip.
Les deux jobs utilisent le même Jenkinsfile, placé à la racine du dépôt git du projet projectname.

Le Jenkinsfile :

- utilise tox pour exécuter les tests (qui lui même utilise pytest et pylint)
- build les paquets debian (si il est appelé par le job projectname)
- publie les rapports de tests, coverage, pylint
- notifie les résultats de build par mail (par <https://wiki.jenkins.io/display/JENKINS/Mailer>)

Une librairie est utilisé pour éviter la redondance entre les Jenkinsfile des différents projets (<http://git.entrouvert.org/jenkins-lib.git/>).

Documentation jenkins pipeline : <https://jenkins.io/doc/book/pipeline/>

Documentation jenkins shared library : <https://jenkins.io/doc/book/pipeline/shared-libraries/>

Job projectname

- job de type pipeline
- ne s'exécute que sur la branche master (déclenchement automatique par scrutation de changements sur le dépôt git toutes les 5 minutes, et à défaut de changements une fois par jour)
- exécute les tests, ne build pas les paquets debian
- notifie les résultats de build par mail à une liste de diffusion : admin+jenkins-projectname@entrouvert.com

Job projectname-wip

- job de type pipeline multibranch
- s'exécute que sur les branches wip/* (déclenchement automatique par scrutation de changements sur le dépôt git toutes les 5 minutes)
- exécute les tests et build les paquets debian dans la foulée
- notifie les résultats de build par mail à l'auteur du dernier commit de la branche

Jobs classiques (déprécié)

- créer un jenkins.sh dans chaque projet
- utiliser uniquement tox pour organiser les tests, pylint, génération de la doc, etc..
- si plusieurs suites de tests utiliser merge-junit.sh et merge-coverage.sh du projet authentic2
- ne pas activer l'option "envoyer des emails aux personnes qui ont cassé le build" pour les projets tiers
- configurer un build par scrutation du git + un build périodique (@daily, utile pour détecter au plus tôt des problèmes avec des mises à jour de dépendance)
- ne pas configurer les jobs "-deb" avec une scrutation de gir, les configurer pour constuire après le build du projet
- pour tester les branches de développement il suffit déclarer un "branch specifier" avec un wildcard, ex.: wip/*, attention si cela affecte le statut je job quelque soit la branche, il vaut donc mieux le faire sur un job spécifique (nommé par exemple <project>-wip-branches)