# Le serveur d’Intégration Continue

## Ses caractéristiques

* Cross plates-formes (Windows, Linux)
* Cross langages (.NET, JAVA/J2EE, Javascript)
* Cross référentiel de code source
* Open Source

## Jenkins leader mais

Jenkins est mal adapté au pipeline de déploiement. Il n’a pas été modélisé pour être « first class ». Il existe des plugins pour palier en partie au manque de cette fonctionnalité mais ils ne semblent jamais fonctionner correctement. Bien entendu nous pouvons définir des « jobs » qui s’exécuteront avant ou après tel autre « job » mais cela est sujet aux erreurs lors de séquences complexes.

Jenkins est un serveur d’Intégration Continue orienté technique. De nombreuses configurations se font par de petits scripts shell coller dans des petites zones de texte et par une profusion de plugins. Son expérience utilisateur (UX), très peu développée n’offre pas de vision claire aux non-initiés. Par exemple, pour afficher le journal de sortie d’une build échoué dans Jenkins peut demander jusqu’à trois clics à partir de la page d’accueil. Cette orientation technique dessert plutôt mal un des principes fondamental du pipeline de déploiement qu’est la collaboration entre les différentes entités impliquées dans le projet. Dans une optique agile et DevOps, où le développeur tente de se rapprocher du métier et des opérationnels en exposant l’avancé de ses développements au jour le jour Jenkins se pose en frein.