



USINE LOGICIELLE



- Connaissances préalables :
 - Java (les fondamentaux)
 - Programmation POO
 - Notions de Shell

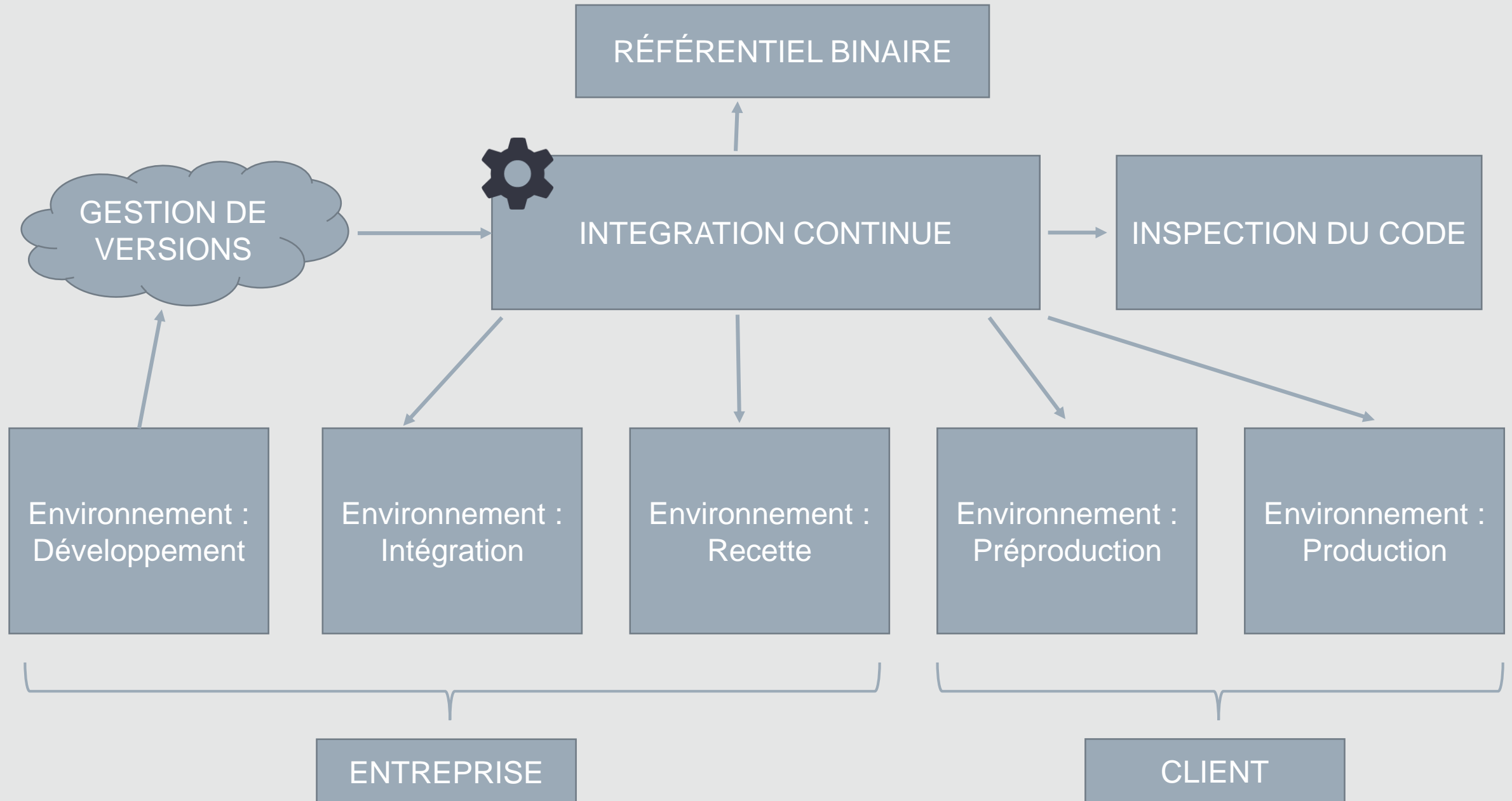
MODULE 1 : Les concepts de l'usine logicielle

MODULE 2 : Les gestionnaires de version

MODULE 3 : Les moteurs de production

MODULE 4 : Les tests unitaires

MODULE 5 : L'intégration continue



Les différents environnements



| ENVIRONNEMENT | RÔLE | ACTEURS |
|---------------|--|---|
| Développement | Développement de l'application | Développeurs |
| Intégration | Vérifier que l'application soit viable techniquement. Généralement l'environnement est ISO prod. | Développeurs, intégrateurs, architectes |
| Recette | Vérifier que l'application réponde à la demande du client, Généralement l'environnement est ISO prod. | Testeurs fonctionnels |
| Préproduction | Environnement pour vérifier l'installation de l'application, vérifier une dernière fois la stabilité du logiciel et des procédures d'installation, Généralement l'environnement est ISO prod. | Client, hébergeur |
| Production | Environnement final de l'application. | Client, visiteurs... |

Le gestionnaire de version



| RÔLE |
|--|
| Versionner le code des développeurs |
| Mutualiser le code entre développeurs |
| Organiser le développement des fonctionnalités |
| Réaliser un suivi du code |
| Gérer les différentes versions du code source |
| Réaliser un « backup » des codes sources |

Le gestionnaire de version



INTEGRATION CONTINUE

| RÔLE |
|-----------------------------------|
| Récupérer les dépendances |
| Compiler l'application |
| Exécuter les tests |
| Vérifier la qualité du code |
| Packaging de l'application |
| Réaliser les déploiements |
| Mise en forme de la documentation |
| Notifier les différents acteurs |

Le gestionnaire de version

INSPECTION DU CODE

| RÔLE |
|--|
| Analyser les tests réalisés sur l'application |
| Réaliser des rapports sur la qualité du code |
| Conseiller les développeurs pour améliorer le code |
| Suivi de la qualité du code dans le temps |

Le gestionnaire de version

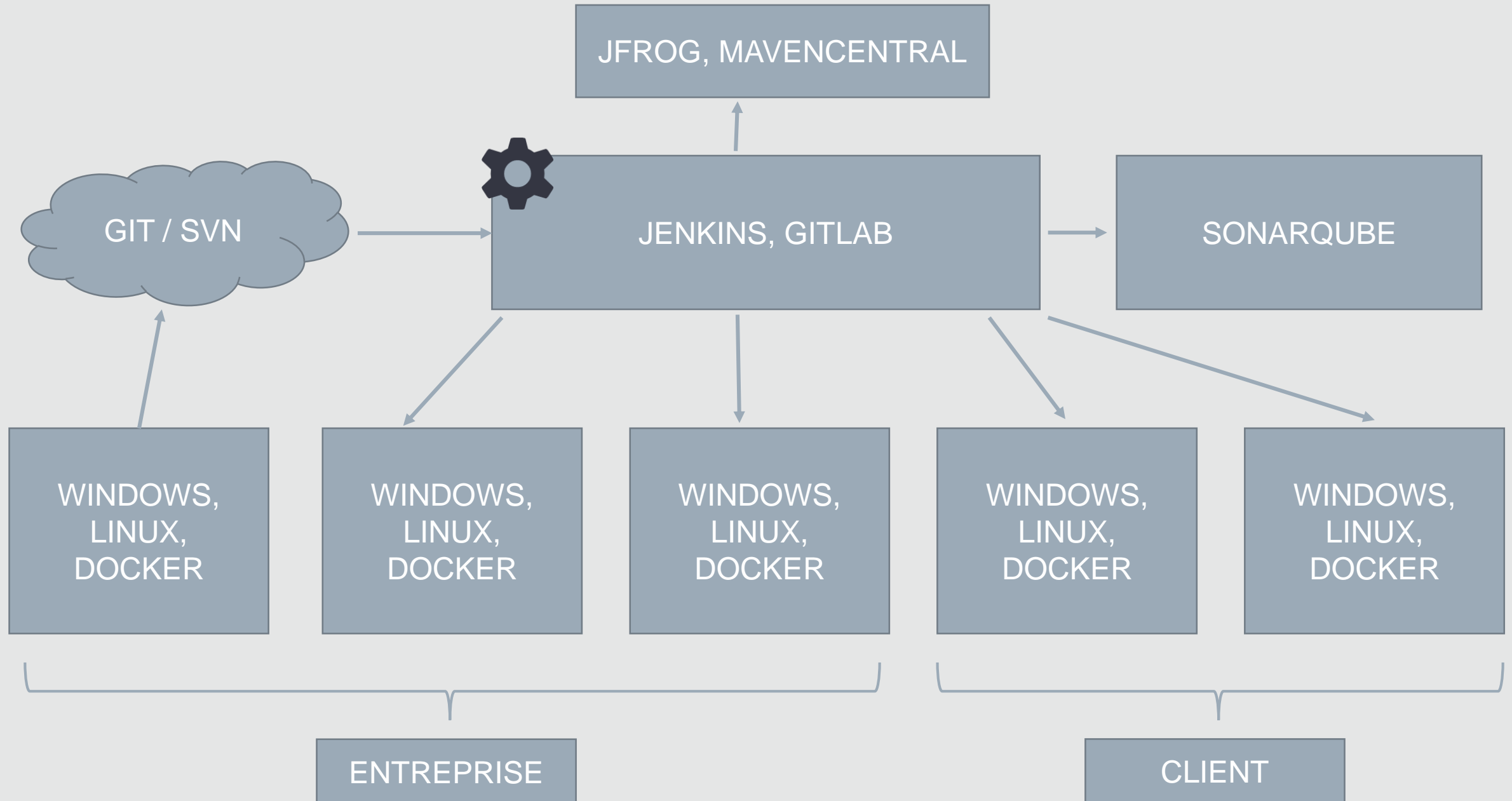
RÉFÉRENTIEL BINAIRE

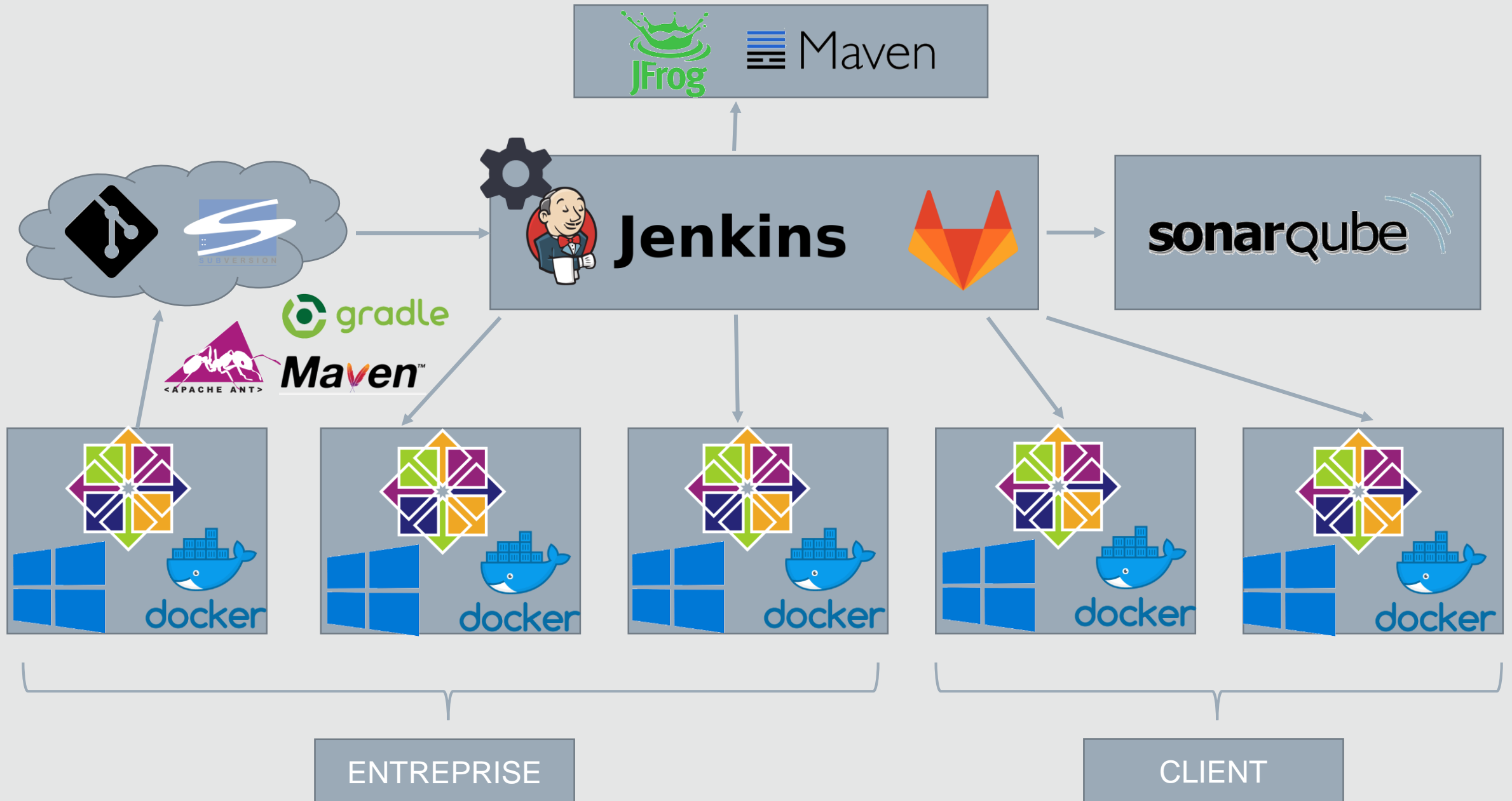
RÔLE

« Graver dans le marbre » les différentes versions du logiciel

Réaliser une rétention des packagings réalisés

Mutualiser le code via la rétention de librairies (*.jar)





La gestion du code source, des versions, des rapports, de la documentation etc. est très lourde !

Il est nécessaire d'automatiser un maximum afin de:

- gagner du temps
- consolider les processus
- simplifier la vie des développeurs...

L'usine logicielle est là pour ça !



FIN DU MODULE

