

Développement durable au



Constat

- Beaucoup de code développé
- Peu de capitalisation / peu de code réutilisé
- Peu de continuité dans les développements
- Peu de code diffusé

Objectifs

- Accompagner les équipes de recherche sur les aspects développement logiciel, diffusion et valorisation
 - Favoriser l'adoption de bonnes pratiques de développement
 - Industrialiser la production logicielle du laboratoire
 - Assurer les déploiements des applications sur différents environnements (systèmes d'exploitation, version de langages...)
 - Valoriser le patrimoine logiciel du laboratoire
 - Assurer la reproductibilité des résultats

Contraintes

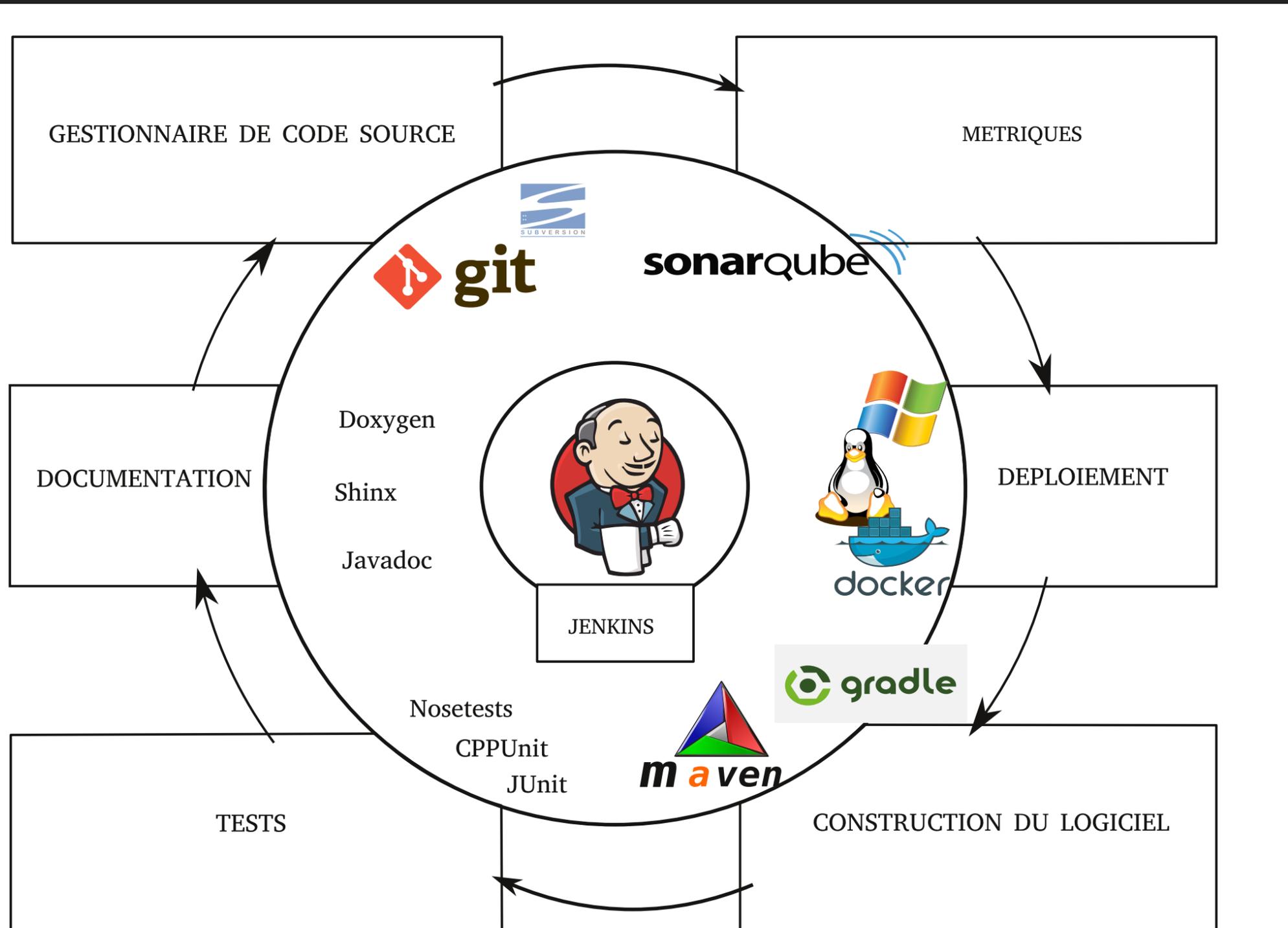
- Diversité des contributeurs : titulaires, étudiants en thèse, en stage, développeurs sur contrats courts
- Gestion des plateformes d'exécution (version des OS, des langages, des bibliothèques, des outils)

Moyen

- Mise en oeuvre d'une plateforme d'intégration continue



Outils mis en oeuvre



Gestionnaires de code source



- travailler en autonomie
- travailler en équipe
- maintenir plusieurs versions d'un logiciel (diffusion de plusieurs versions, mises à jour, correctifs, ajouts de fonctionnalité)

Métriques

- Analyse de code source
- Respect des conventions de codage
- Détection de duplication de code
- Mesure de complexité



Main Dashboard

Time changes... ▾

Lines Of Code 27,993 ↗	Files 396	Functions 1,997 ↗
c++	Directories 13	Classes 283
	Lines 52,742 ↗	Statements 18,531 ↗

SQALE Rating A	Technical Debt Ratio 0.1%
Debt 1d 5h ↘	Issues 353 ↗
<ul style="list-style-type: none"> 🚫 Blocker 0 🔴 Critical 112 ↗ 🔴 Major 104 ↗ 🟢 Minor 137 ↘ 🟢 Info 0 	

Duplications
7.5%

Lines 3,949 ↗	Blocks 161 ↗	Files 45
-------------------------	------------------------	--------------------

Directory Tangle Index 1.1%	Dependencies To Cut
	Between Directories 1
	Between Files 3
Cycles > 1	

Complexity
5,586 ↗

/Function 2.8	/Class 17.9	/File 14.1
-------------------------	-----------------------	----------------------

● Functions ○ Files

Unit Tests Coverage 0.0%	Unit Test Success 86.8% ↗
Line Coverage 0.0%	Failures 14
	Errors 0
	Tests 106
	Execution Time 17:46 min ↗

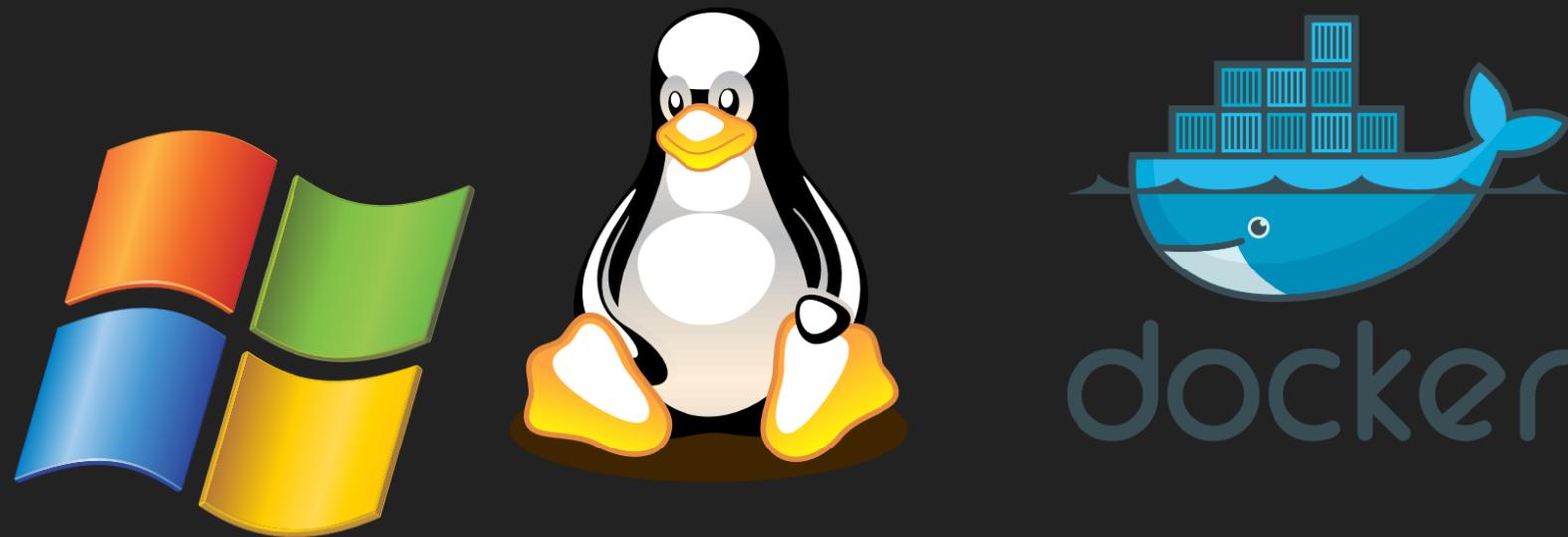
Events **All** ▾

No event

PilgrimRelational PilgrimRelational.15.05

Profiles: [Sonar way](#) (c++)

Déploiement sur cibles hétérogènes



Construction du logiciel

- Compilation
- Gestion des dépendances
- Création de packages



Outils de Tests

- Tester le logiciel
 - tests unitaires
 - tests d'intégration
 - tests fonctionnels
 - tests de non régression
 - tests de performance
 - tests de déploiement / d'installation
- Analyse dynamique (Profiler, Valgrind)

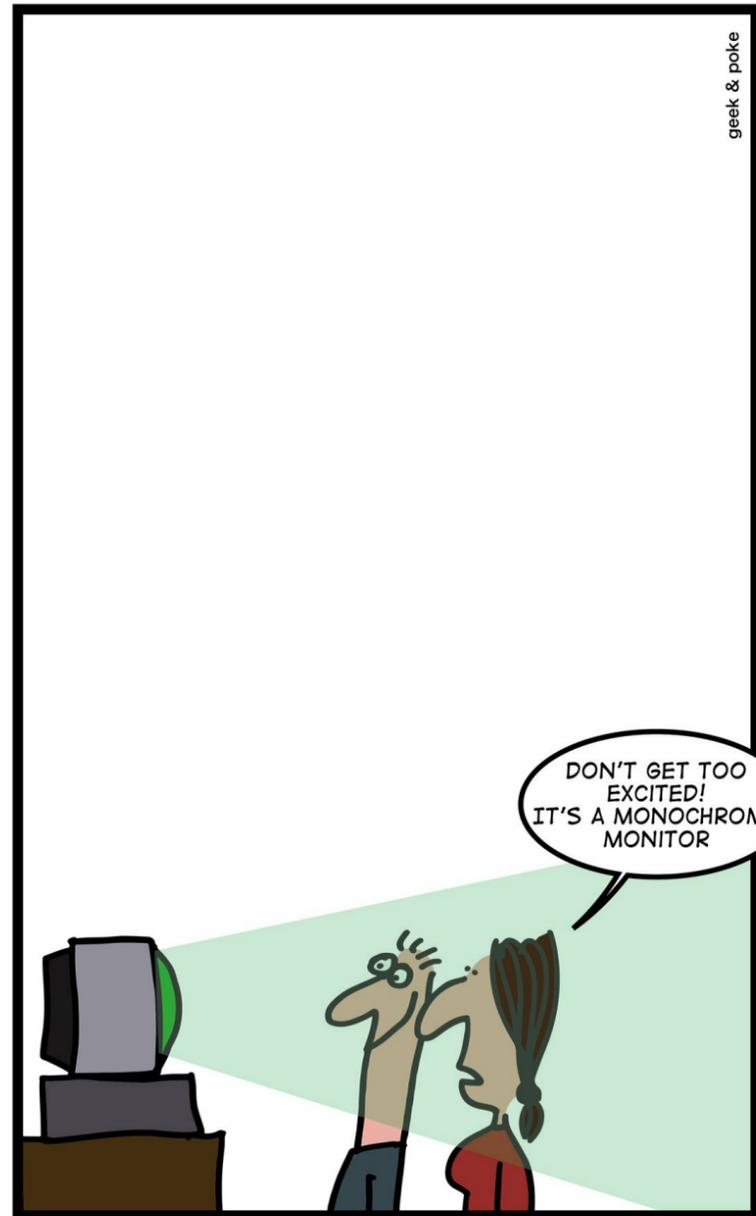


geek & poke

50 SHADES OF GREEN

Idea from Jens Wolfgramm

THE HISTORY OF IT

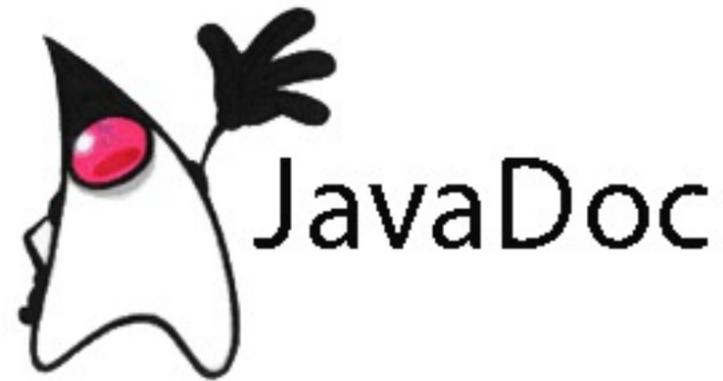
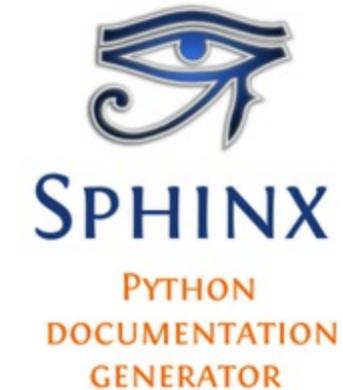


geek & poke

JENKINS ON AN IBM PC AT

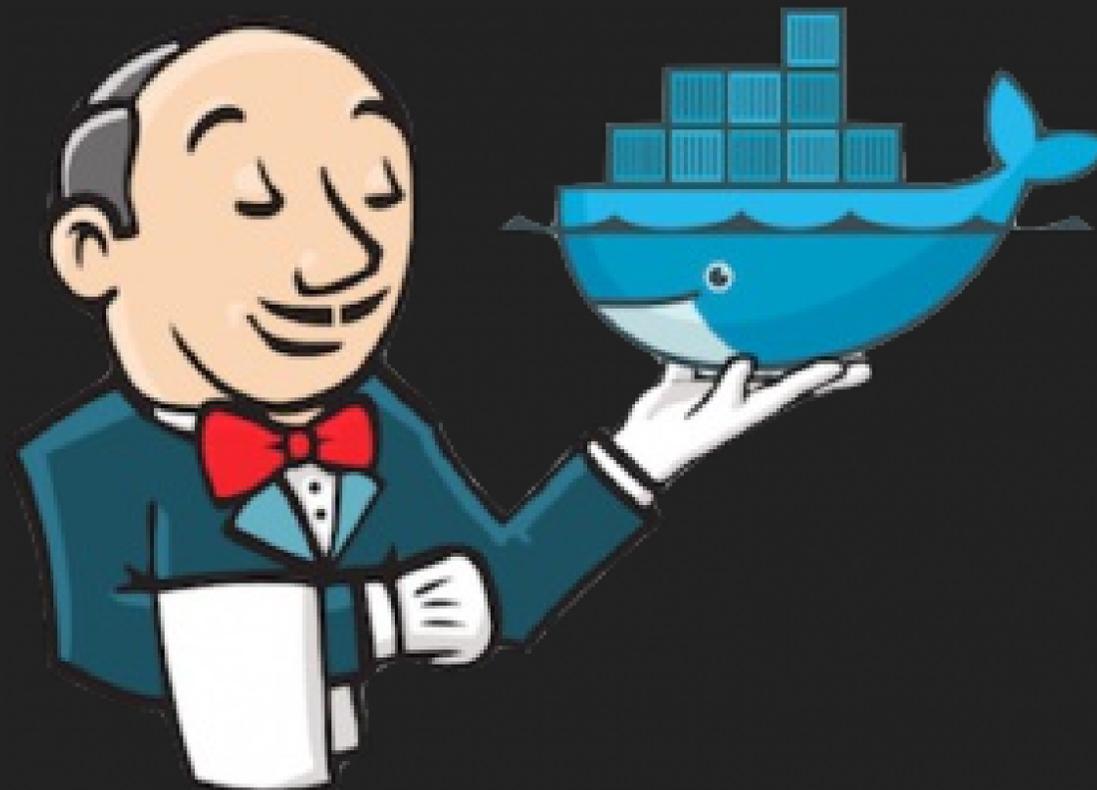
Documentation

- Génération automatique de la documentation à partir des fichiers source

The logo for Doxygen, featuring the word "Doxygen" in a blue, sans-serif font with a blue underline, set against a white background.The logo for JavaDoc, featuring a stylized cartoon character with a red eye and a black hand, next to the text "JavaDoc" in a black, sans-serif font.The logo for Sphinx, featuring a stylized blue eye icon above the text "SPHINX" in a blue, sans-serif font, with "PYTHON DOCUMENTATION GENERATOR" in a smaller, orange, sans-serif font below it.

Conclusion

- Adoption de bonnes pratiques de développement
- Personnalisation les cibles d'exécution (choix de l'OS, versions des outils et librairies)
- Gestion simplifiée les prérequis
- Facilité de déploiement des logiciels développés
- Garantie de la reproductibilité des résultats





**KEEP
CALM
AND
USE
JENKINS**