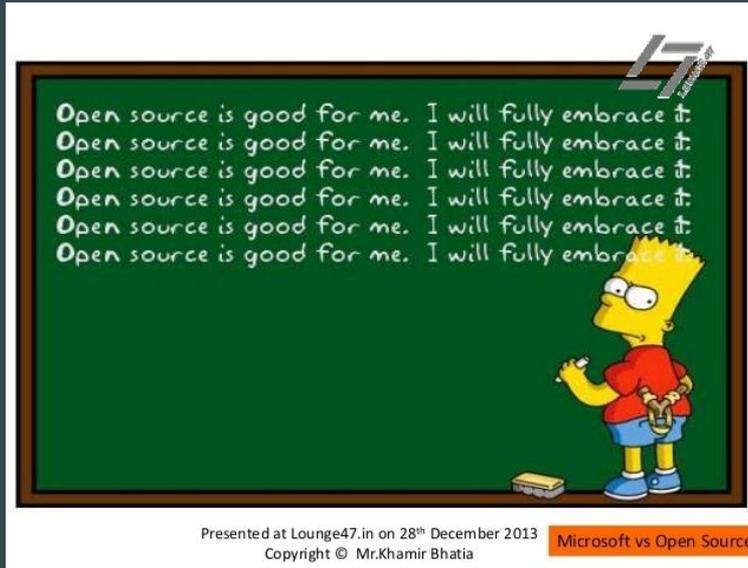


L'Open Source au secours du développeur (et de l'architecte) ?



THANK YOU

Faisons connaissance

👋 Qui utilise un ou plusieurs projets Open Source ?

👋 Qui a déjà participé une fois à un projet Open Source ?

👋 Qui participe régulièrement à un projet Open Source ?

👋 Qui le fait de manière professionnelle (cad payé pour ça 😊) ?

Question bonus : 👋 qui (ou sa société) a déjà payé / sponsorisé un projet Open Source ?

(hors achat de support)

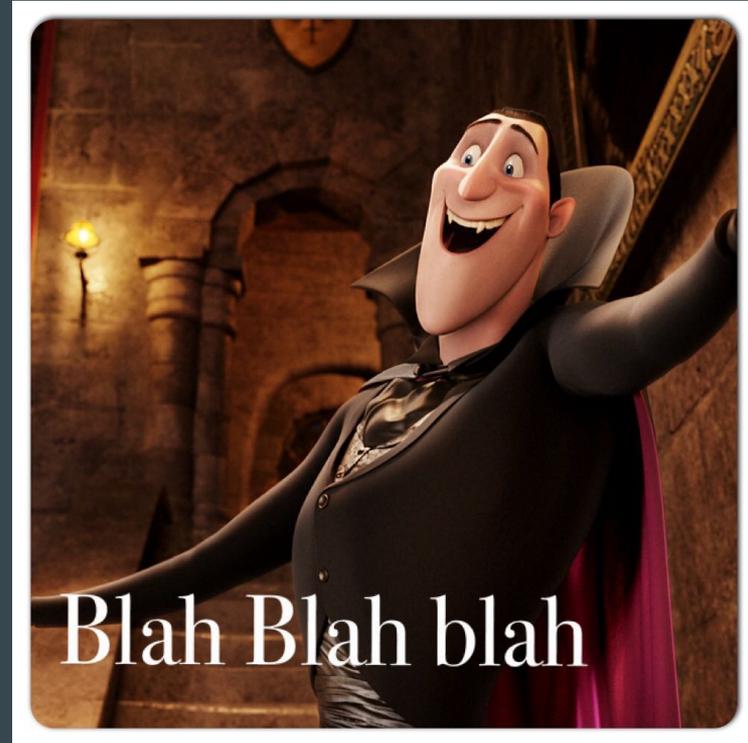


Éléments de langages

- ❌ Open source <> Freeware
- ❌ Open source <> Gratuit
- ❌ Open source <> Pas de licence

- ✅ Open source == Accès au code source
- ✅ Open source =~ Ouvert à la participation
- ✅ Open source == Licence attachée au source (MIT, BSD, Mozilla, ...)

[source](#)



Un peu de contexte

- Développeur un jour ... développeur toujours
 - Débuts à l'âge de 9 ans
 - Passion avant un métier

- De développeur à architecte
 - 13 ans de développement
 - 7 ans d'architecture / développement

- D'intellij à powerpoint

```
var scrollHeight =  
element.clientHeight + 0.02;  
window.scroll(0, scrollHeight);  
}  
  
(function() {  
function a_setFocus() {  
var form = null;  
if (document.getElementById) {  
fo
```



Et vous vous en êtes où ?

👉 Qui estime qu'il ne code pas assez ?

👉 Au niveau professionnel ?

👉 Au niveau personnel ?

👉 Qui n'est pas développeur ?



Pourquoi se lancer dans l'Open Source ?



Continuer à développer

🕒 20% du temps restant pour développer (au mieux)

📦 Side project

🎯 Trouver un objectif motivant

Les sides projects



Continuer à progresser

👨‍🎓 Permet de faire de la R&D et continuer à apprendre

💬 Se confronter aux autres

🔍 Découvrir le code des autres

👁️ Exposer son code au regard des autres



source

Partage et communauté

🤝 Le développement est un travail d'équipe

❤️ Participer à quelque chose d'utile

📥 Re-verser un peu à la communauté



Les mauvaises raisons pour faire aller vers l'open source

👑 Gloire et beauté

📷 Célébrité et renommé

💶 Richesse



Par où commencer ?

 Internet ...

 [How to contribute to open source project as beginner \(Catalin Pit\)](#)

 [Conférence](#) donnée par Helen Scott (Jet Brains)



Trouver le bon projet

🔍 Utiliser des “indexeurs” de projets ouverts à participation

➔ <https://up-for-grabs.net>

➔ <https://www.codetriage.com>

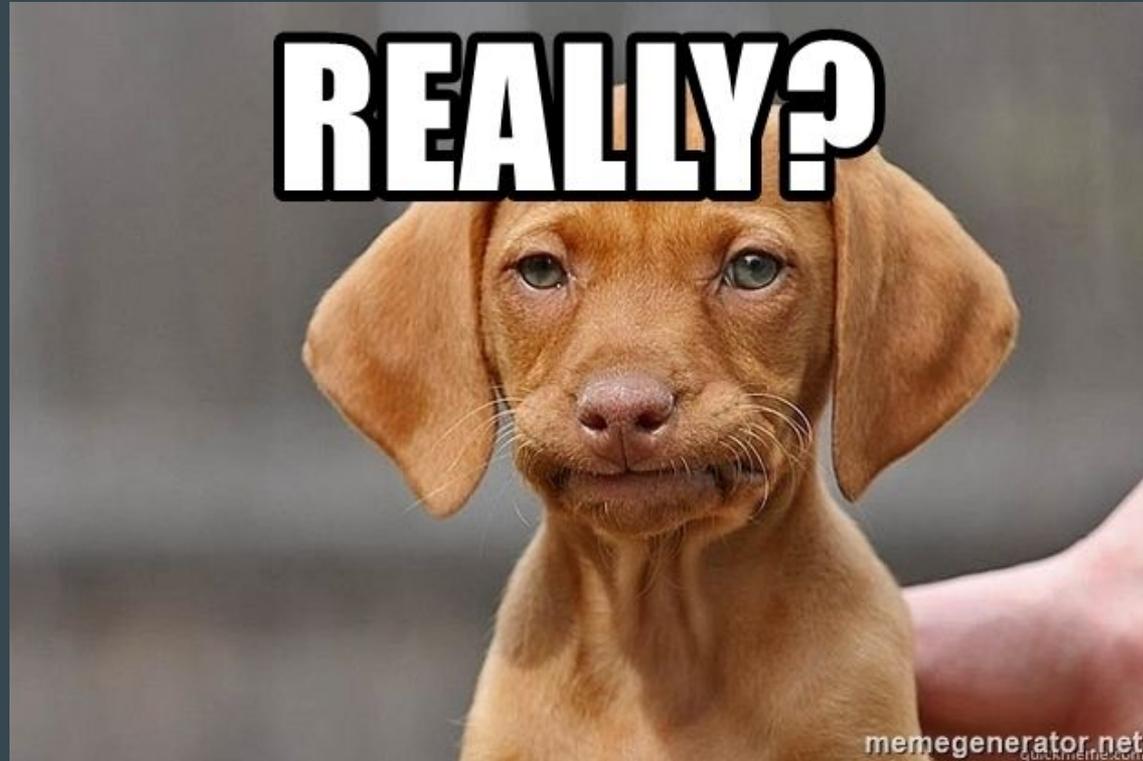
➔ <https://www.firsttimersonly.com/>

🔍 Topics dans GitHub / GitLab

➔ good first issue, hacktoberfest, help wanted, ...



Un projet connu et utilisé régulièrement



Un projet connu et presque utilisé régulièrement

🧠 Syndrome de l'imposteur ou marche trop haute ?

→ [SpringBoot](#)

→ [OpenJDK](#)

→ [Jenkins-core](#)

🏭 JenkinsPipelineUnit ... Trop professionnel

→ Groovy

→ Candidat idéal pour une démarche Open Source “professionnelle”





- 💡 Ne pas viser tout de suite la lune, accepter de commencer modestement
- 💡 Essayer de trouver un projet utilisé par l'entreprise pour y participer sur le temps de travail

Un projet connu fonctionnellement et utilisé régulièrement

👍 Jenkins reste un bon choix

⚙️ Fonctionnement connu : utilisation quotidienne

❤️ Plugins plus “abordables” : moins complexe que le core

👤 Java 8 (pas encore 11)



<https://jenkins.io>



💡 A défaut de participer à un projet utilisé dans ses développements, participer à un projet qui est une application utilisée tous les jours peut aider à “rentrer” dans le projet



© 2010 ESA CNES ARIANE SPACEDOPPOQUE VIDEO D'ILLUSTRATION

<https://www.ariane.group/fr/photos-videos/>

Comprendre (comment fonctionne) le projet



Les documentations projet sont inégales



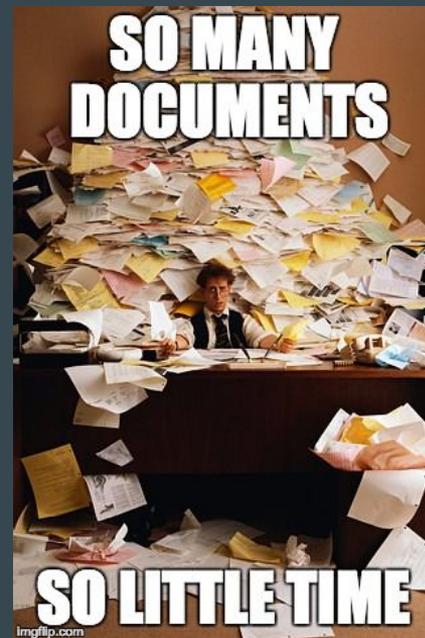
Peu de documentation de conception / d'architecture



Privilégier les projets dont les issues sont sur GitHub / GitLab



Avis personnel (préférence par rapport à JIRA)





💡 Regarder les tests unitaires (en règle générale les projets Open Source ont pas mal de tests) aide à comprendre comment le projet doit être utilisé

La première contribution

→ [Ansi-color-plugin](#)

Lire les différents éléments inhérents à la contribution
(CONTRIBUTING.MD)

- règles de développement (checkstyle, nom des branches, ...)
- code de conduite

“Il n’y a pas de petites contributions”

- Correction d’un lien du README vers le CONTRIBUTING



www.birdsdessines.fr



Fix broken link in README.md #230

Merged `dblock` merged 1 commit into `jenkinsci:master` from `philippart-s:fix-broken-link-contributing` on Jun 22

- Verrou débloqué
- Envie d'aller plus loin !



💡 Lire les documentations pour la contribution et l'organisation de la communauté facilite l'intégration et la prise en compte des futures PR

💡 Ne pas hésiter à commencer par des PR simplistes ou de documentation

💡 Demander avant de travailler sur une issue, c'est plus apprécié et permet de nouer un premier contact

Le début de l'aventure



La recherche du Graal

→ [conventional-commits-plugin](#)

- ✓ Géré dans GitHub
- ✓ Pas trop gros mais actif
- ✓ Communauté active
- ✓ Peu de code (facilité de prise en main)
- ✓ Un peu de chance ... (GSoC)





💡 Essayer de trouver un projet d'une taille pas trop grosse facilite sa prise en main

💡 Une communauté réduite est simple à contacter et identifier (qui fait quoi, notamment les maintainers)

💡 Une communauté ayant un canal pour discuter librement (ou du moins trouver le canal 😊)

La première issue / Pull Request (PR)

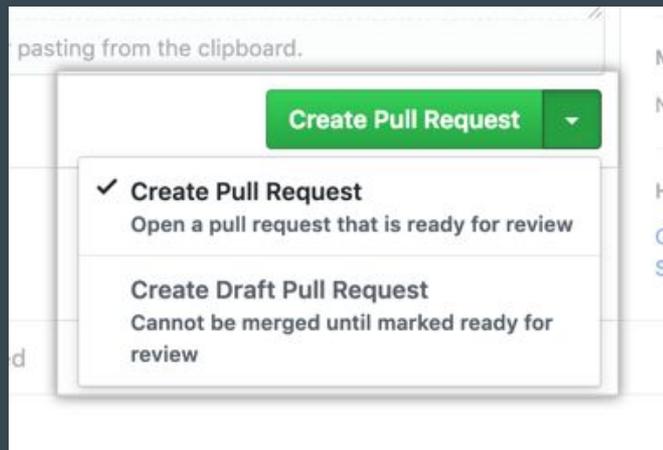
🔗 Fork du projet (et non clone) : penser à mettre à jour l'upstream pour faciliter les merges

🕒 PR au plus tôt (dès la modification de la documentation par exemple)

❤️ Coder pour le plaisir

- retrouver la raison première d'être développeur
- pas de biais d'entreprises (budgets, rétro planning, managers, ...)

 A ce jour une petite quinzaine de PR



<https://docs.github.com>



💡 Créer la PR au plus tôt en draft

💡 Expliquer ce que l'on compte faire

💡 Pusher le code au fil de l'eau

Rythme de travail

😱 Ne pas reproduire le stress du monde professionnel

🕒 Prendre le temps pour développer feature après feature

📅 July 17 Une feature : 1 à 2 semaines

📅 July 17 Rythme de review : 3 à 5 jours

➡ Un projet Open Source est géré par des anonymes qui font cela sur leur temps personnel comme vous et moi !



À grand pouvoir, grandes responsabilités.

Spiderman (Ben Parker)

www.citation-celebre.com

⚠ Avant de se lancer dans la participation d'un projet Open Source il faut s'assurer que l'on a le temps pour cela une fois déduites toutes les activités habituelles (travail, sommeil, famille, ...).

⚠ S'engager sur des implémentations de features et ne pas le faire par manque de temps pénalise le projet mais aussi les personnes qui auraient pu travailler sur ces features.

Organisation

 Principalement le WE et pause déjeuner

 Accepter le rythme de travail des autres lors de reviews de PR

 En profiter pour faire une pause / review des autres PR



 3 à 4 heures par semaine



- 💡 Ne pas cloner son projet sur son ordinateur pro pour ne pas être “tenté”
- 💡 Un seul projet pour débuter dans le l’Open Source et pour ne pas avoir à switcher et avoir des moments de pause
- 💡 Ne pas s’engager si on n’a pas vérifié notre capacité à faire

Bonus apporté par l'Open Source

🇬🇧 Améliorer mon niveau d'anglais

🎓 Apprendre de nouvelles façons de développer / de s'organiser

🆘 Aider d'autres développeurs

😁 Petite fierté personnelle : au bout de quelques semaines je suis devenu core commiter



Pari gagné ?

- ✓ Reprendre un rythme de développement en Java
- ✓ Continuer à apprendre ... même en Java 8
- ✓ Vaincre (un peu) le syndrome de l'imposteur
- ✓ Sentiment d'être utile



The screenshot shows the GitHub Actions release page for version 0.7.3. It includes a 'Latest release' badge, the version number '0.7.3', a commit hash 'b2356e1', and a 'Verified' badge. A 'Compare' dropdown menu is visible. The release was made by 'github-actions' 3 days ago. The 'New features and improvements' section lists two features: 'feat: Write next version gradle (#105) @philippart-s' and 'feat: Write next version helm (#100) @philippart-s'.

Latest release

0.7.3

b2356e1

Verified

Compare ▾

github-actions released this 3 days ago

 **New features and improvements**

- feat: Write next version gradle (#105) @philippart-s
- feat: Write next version helm (#100) @philippart-s

Et la suite ?

📊 Projet en légère perte de vitesse : réalité pour un projet Open Source, il peut être abandonné

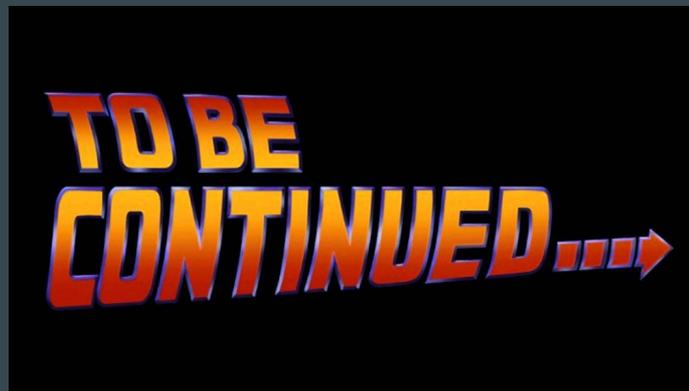
✓ ~~essayer d'être un~~ Maintainer principal

✓ Hacktoberfest

✨ Se lancer dans d'autres projets

➔ Projet Java 11 / 17

➔ Les nouveaux challenges apportés par mon intégration à OVH Cloud



Dans tous les cas continuer cette aventure !

Alors et maintenant ?

👉 Qui a envie de se lancer dans l'aventure ?



Stéphane Philippart

- 👛 Baby DevRel@OVHCloud (depuis 2 jours) 🥑
- 👛 Co-créateur de [TADx](#) (meetups Agile, Dev, DevOps)
- 👛 Commiteur Open Source

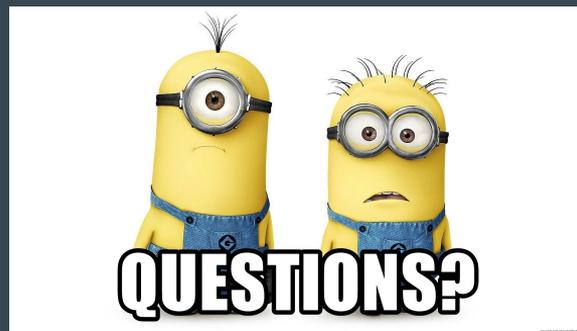
🐦 [@wildagsx](#)

🌐 <https://philippart-s.github.io/blog>

🔗 <https://github.com/philippart-s/>

🌐 <https://www.linkedin.com/in/philippartstephane/>

“Il n’y a pas de petites contributions”



<https://roti.express/r/sc22-040>

Liens

- <https://opensource.com/>
- <https://screenrant.com/>
- <https://daily.dev/blog/how-to-contribute-to-open-source-projects-as-a-beginner>
- <https://blog.jetbrains.com/idea/2020/08/jetbrains-technology-day-for-java-how-i-started-contributing-to-open-source-and-why-you-should-too>
- <https://github.com/spring-projects/spring-boot/blob/main/CONTRIBUTING.adoc>
- https://openjdk.java.net/contribute/?utm_source=pocket_mylist
- <https://www.jenkins.io/participate/>
- <https://www.jenkins.io/>
- <https://www.conventionalcommits.org/en/v1.0.0/>

