

Des routes et des ponts (9) – L'argent et l'open source

Nadia Eghbal a déjà évoqué plusieurs fois les liens entre l'argent et l'open source (si vous avez manqué des épisodes). Elle y revient dans ce chapitre, en insistant sur les questions fondamentales que pose l'argent aux communautés open source ainsi qu'à leurs membres.

Question de nature quasi-philosophique : l'open source peut-il perdre son âme à cause de l'argent ? Question de gouvernance : qui va décider de l'utilisation des fonds ? Et pour finir question éthique et politique : jusqu'à où peut-on, doit-on accepter les requêtes des financeurs ?

La relation compliquée de l'open source avec l'argent

Traduction Framalang : goudron, Penguin, serici, goofy, Rozmador, xi, Lumibd, teromene, xi, Diane, et 3 anonymes



L'argent est un sujet tabou dans les projets *open source*, et ce depuis les premiers jours du mouvement du logiciel libre qui émergea en réponse directe aux pratiques commerciales des logiciels propriétaires. Dans le contexte du mouvement du logiciel libre, l'aversion pour l'argent est tout à fait compréhensible. L'argent est ce qui permettait de commercialiser les logiciels dans les années 1980 et il a

fallu des décennies pour revenir sur cet état d'esprit et promouvoir les avantages liés à l'élaboration de logiciels qui soient libres d'utilisation, de distribution et de modification. Même si de nos jours, nous prenons les logiciels libres pour acquis, dans les années 1980, c'était une véritable contre-culture, un état d'esprit révolutionnaire.

Au sein même des communautés *open source*, il existe une croyance répandue selon laquelle l'argent est de nature à corrompre l'*open source*. Et en effet, le nombre de projets nés d'un « travail-passion » est assez incroyable. Aujourd'hui, le développement de logiciel est considéré comme un domaine lucratif, dont les écoles de programmation appâtent leurs futurs étudiants avec des promesses de premiers salaires en dollars à six chiffres. Par contraste, il y a quelque chose de pur et d'admirable dans le fait de créer un logiciel simplement pour le plaisir.

D'un point de vue plus pratique, les projets *open source* se créent traditionnellement autour d'un besoin réel et identifiable. Quelqu'un estime qu'un projet pourrait être mieux fait, décide de le *forker*, effectue des améliorations, puis les diffuse pour qu'on en fasse usage. Le pragmatisme est au cœur de la culture *open source*, comme le prouve sa scission stratégique avec le mouvement du logiciel libre à la fin des années 1990. Certains contributeurs *open source* craignent, peut-être avec raison, que l'argent n'introduise un développement « artificiel » du système, avec des développeurs qui lancent de nouveaux projets simplement pour acquérir des financements, plutôt que pour répondre à un besoin réel.

David Heinemeier Hansson (aussi connu sous le pseudo de DHH), qui a créé le *framework* populaire *Ruby on Rails*, mettait en garde en 2013 contre les mélanges entre *open source* et argent :

Si l'open source est une incroyable force pour la qualité et pour la communauté, c'est précisément parce qu'elle n'a pas

été définie en termes de marché. Dans le cadre du marché, la plupart des projets open source n'auraient jamais eu leur chance.

Prenez Ruby on Rails. [...] C'est une réalisation colossale pour l'humanité ! Des milliers de gens, collaborant pendant une décennie entière pour produire une structure et un écosystème incroyablement aboutis, disponibles pour tous gratuitement. Prenez une seconde pour méditer sur l'ampleur de cette réussite. Pas seulement pour Rails, évidemment, mais pour de nombreux autres projets open source, encore plus grands, avec une filiation plus longue et encore plus de succès.

C'est en considérant ce fantastique succès, dû aux règles de vie d'une communauté, que nous devrions être extraordinairement prudents avant de laisser les lois du marché corrompre l'écosystème.

Structurellement, le meilleur atout de l'*open source* : son penchant pour la démocratie, est aussi sa faiblesse. Beaucoup de projets *open source* ne sont rien de plus qu'un dépôt numérique public où est stocké du code auquel un groupe de gens contribue régulièrement : l'équivalent d'une association officieuse sur un campus universitaire. Il n'y a pas de structure légale et il n'y a pas de propriétaire ou de chef clairement défini. Les « mainteneurs » ou les contributeurs principaux émergent souvent *de facto*, en fonction de qui a créé le projet, ou de qui y a investi beaucoup de temps ou d'efforts. Cependant, même dans ces cas-là, dans certains projets on répugne à introduire une hiérarchie favorisant clairement un contributeur par rapport à un autre.

En avril 2008, Jeff Atwood, un développeur .NET bien connu et dont nous avons déjà parlé, a annoncé qu'il donnait 5 000 \$ au projet *open source* : *ScrewTurn Wiki*. *ScrewTurn Wiki* est un projet de wiki développé par Dario Solara, un autre

développeur .NET, et maintenu par des volontaires. Atwood a dit à Dario que le don était « sans condition » : Solara pouvait utiliser l'argent de la manière qu'il jugerait la plus utile au projet.

Plusieurs mois plus tard, Atwood demanda à Solara comment il avait décidé de dépenser l'argent. Solara lui répondit que l'argent de la donation était « *encore intact. Ce n'est pas facile de l'utiliser... Que suggères-tu ?* » Atwood a écrit que cette réponse l'avait « terriblement déçu ».

La nature décentralisée du monde *open source* en a fait ce qu'il est : des logiciels produits de façon participative, que n'importe qui peut élaborer, partager, et améliorer. Mais quand vient le moment de discuter des besoins organisationnels, ou de la viabilité, il peut être difficile de prendre des décisions faisant autorité.

Ces transitions vers une viabilité à long terme peuvent être interminables et douloureuses. Un des exemples les plus connus est le *noyau Linux*, un projet *open source* utilisé dans de nombreux systèmes d'exploitation à travers le monde, parmi lesquels Android et Chrome OS. Il a été créé en 1991 par Linus Torvalds, un étudiant en informatique .

Au fur et à mesure que le noyau Linux gagnait en popularité, Linus rechignait à discuter de l'organisation du développement du projet, préférant tout gérer tout seul. L'inquiétude et aussi la colère à l'égard de Torvalds grandirent chez les développeurs du projet, déclenchant de « vraies grosses disputes » selon Torvalds. Le conflit a atteint son apogée en 2002, on évoqua même un possible schisme.

Torvalds attribua ces conflits internes à un manque d'organisation, plutôt qu'à un quelconque problème technique :

Nous avons eu de vraies grosses disputes aux alentours de 2002, quand j'appliquais des correctifs à droite à gauche, et que les choses ne fonctionnaient vraiment pas. C'était très douloureux pour tout le monde, et également beaucoup pour

moi. Personne n'aime vraiment les critiques, et il y avait beaucoup de critiques virulentes, et comme ce n'était pas un problème strictement technique, on ne pouvait pas juste montrer un correctif et dire : « Hé, regardez, ce patch améliore les performances de 15% » ou quoique ce soit de ce genre. Il n'y avait pas de solution technique. La solution a été d'utiliser de meilleurs outils, et d'avoir une organisation du travail qui nous permette de mieux distribuer les tâches.

La Fondation Linux a été créée en 2007 pour aider à protéger et à maintenir Linux et ses projets associés. Torvalds ne pilote pas la Fondation Linux lui-même, il a préféré recevoir un salaire régulier en tant que « Compagnon Linux », et travailler sur ses projets en tant qu'ingénieur.

Malgré le fait que le logiciel *open source* soit admirablement ancré dans une culture du volontariat et de la collaboration relativement peu touchée par des motivations extérieures, la réalité est que notre économie et notre société, depuis les sociétés multimillionnaires jusqu'aux sites web gouvernementaux, dépendent de l'*open source*.

Dans l'ensemble, c'est probablement une évolution positive pour la société. Cela signifie que les logiciels ne sont plus limités à un développement privé et propriétaire, comme cela a été le cas pendant des dizaines d'années. Le fait que le gouvernement des États-Unis, ou un réseau social possédant des milliards d'utilisateurs, intègrent des logiciels construits par une communauté, annonce un futur optimiste pour la démocratie.

De plus, de nombreux projets fonctionnent très bien de manière communautaire lorsqu'ils sont d'une des deux tailles extrêmes possibles, c'est-à-dire soit des petits projets qui ne demandent pas de maintenance significative (comme dans l'exemple de Arash Payan et Appirater), soit de très gros

projets qui reçoivent un soutien important de la part d'entreprises (comme Linux).

Cependant, beaucoup de projets sont coincés quelque part entre les deux : assez grands pour avoir besoin d'une maintenance significative, mais pas d'une taille suffisante pour que des entreprises déclarent leur offrir un soutien. Ces projets sont ceux dont l'histoire passe inaperçue, ceux dont on ne parle pas. Des deux côtés, on dit aux développeurs de ces projets « moyens » qu'ils sont le problème : du côté des « petits projets », on pense qu'ils devraient simplement mieux s'organiser et du côté des « gros projets », on pense que si leur projet était « assez bon », il aurait déjà reçu l'attention des soutiens institutionnels.

Il existe aussi des intérêts politiques autour de la question du soutien financier qui rendent encore plus difficile la prospection d'une source de financement fiable. On peut imaginer qu'une entreprise seule ne souhaite pas sponsoriser le développement d'un travail qui pourrait également bénéficier à son concurrent, qui lui n'aurait rien payé. Un mécène privé peut exiger des privilèges spécifiques qui menacent la neutralité d'un projet. Par exemple, dans les projets en lien avec la sécurité, le fait d'exiger d'être le seul à qui sont révélées les potentielles failles (c'est-à-dire payer pour être le seul à connaître les failles de sécurité plutôt que de les rendre publiques) est un type de requête controversé. Des gouvernements peuvent également avoir des raisons politiques pour financer le développement d'un projet en particulier, ou pour demander des faveurs spéciales comme une « *backdoor* » (une porte dérobée, c'est-à-dire un accès secret qui permet d'outrepasser les authentifications de sécurité), même si le projet est utilisé dans le monde entier.

Les récents démêlés légaux entre le FBI et Apple sont un bon révélateur des tensions qui existent entre technologie et gouvernement, au-delà même des projets *open source*.

Le FBI a, de manière répétée, et à l'aide d'assignations en

justice, demandé l'aide d'Apple pour déverrouiller des téléphones afin d'aider à résoudre des enquêtes criminelles. Apple a toujours refusé ces requêtes. En février 2016, le FBI a demandé l'aide d'Apple pour déverrouiller le téléphone d'un des tireurs d'une attaque terroriste récente à San Bernardino, en Californie. Apple a également refusé de les aider, et a publié une lettre sur son site, déclarant :

Tout en croyant que les intentions du FBI sont bonnes, nous pensons qu'il serait mauvais pour le gouvernement de nous forcer à ajouter une « backdoor » dans nos produits. Et finalement, nous avons peur que cette demande mette en danger les libertés que notre gouvernement est censé protéger.

En mars 2016, le FBI a trouvé une tierce partie pour l'aider à déverrouiller l'iPhone et a laissé tomber l'affaire.

Une des plus grandes forces de l'*open source* est que le code est considéré comme un bien public, et beaucoup de projets prennent la gestion de ces projets au sérieux. Il est important à titre personnel, pour beaucoup de développeurs de projets, que personne ne puisse prendre seul le contrôle d'une chose que le public utilise et dont il bénéficie. Toutefois, cet engagement à rester neutre a un prix, puisque beaucoup de ressources disponibles pour les développeurs de nos jours (comme les capitaux-risques ou les donations d'entreprises) attendent en contrepartie d'influer sur le projet ou des retours sur investissement.

Le logiciel *open source* est créé et utilisé de nos jours à une vitesse jamais vue auparavant. Beaucoup de projets *open source* sont en train d'expérimenter la difficile transition d'une création désintéressée à une infrastructure publique essentielle.

Ces dépendances toujours plus nombreuses signifient que nous avons pour responsabilité partagée de garantir à ces projets

le soutien dont ils ont besoin.



Crédits pour les 2 images Eelke (CC BY 2.0)