



Open Source

L'enjeu Open Source : co-opérer et tirer partie d'un écosystème différent



Introduction

Les chiffres publiés à la suite de l'Open Source Summit (Paris, les 10 et 11 décembre 2019) attestent de la place croissante du logiciel libre. Selon l'étude annuelle du cabinet Tecknowlogy Group, en effet :

- l'édition des logiciels Open source et les services informatiques associés pèsent plus de 5,2 milliards d'euros en France, soit 10,3% du marché global IT (logiciels et services IT) et plus de 25 milliards en Europe,
- le marché Open source français devrait croître de plus de 8,5% par an, chaque année jusqu'en 2023, ce qui est supérieur à la croissance moyenne du marché de l'IT (près de 4% en 2019-2020),
- ce marché correspond à plus de 60.000 emplois en 2020, en France (chiffre à rapprocher des 474 000 salariés employés par le secteur IT),
- près de 80% des entreprises anticipent que l'utilisation de l'Open source va augmenter dans les deux ans à venir.

L'écosystème Open source, non négligeable au vu des chiffres précédents, diffère de l'écosystème éditeurs avec lequel nos entreprises ont l'habitude d'interagir sous bien des aspects : support et expertise, achats et relations fournisseurs, compétences, vie du produit (pérennité garantie par les éditeurs versus des communautés sans engagement contractuel).

Comment tirer partie et co-opérer avec un écosystème différent, auquel toutes les entreprises ne sont pas habituées ?

L'objectif de cette Fiche Pratique est de partager des expériences, projets, organisations mises en œuvre, recommandations etc. issus de la vraie vie des Cripiens avec l'écosystème Open source.



Pourquoi s'intéresser à l'Open source ?



Les entreprises invoquent le classique argument d'économie budgétaire mais pas seulement. Dans les motivations mises en avant, on trouve :

- la notion de souveraineté et d'indépendance
- la transparence apportée par un code source en accès libre
- la maintenabilité
- et l'innovation, nombre de solutions innovantes se jouant en Open source.

En quoi l'écosystème du libre diffère-t-il du monde propriétaire ?

On n'achète pas du logiciel libre. Ce que l'on achète, c'est du service. Ceci provient des caractéristiques des licences couvrant ces codes source mis à disposition de tous et téléchargeables, même si les binaires ne sont pas toujours mis à disposition de manière libre. Parmi les services vendus par des sociétés, il peut d'ailleurs y avoir la vente du binaire. Mais généralement on trouve les codes source, les binaires dans une forme prête à être déployée. Disposer du logiciel ne requiert pas une action d'achat.

Le monde du libre offre **un certain niveau de support de la part des Communautés**, mais sans garantie de service, ni responsabilité ou engagement contractuel.

Ceci fonctionne grâce à la bonne volonté de chacun avec le risque d'erreur. En revanche, pour avoir du support et de l'expertise avec engagement de délai, qualité, fiabilité, il faut acheter des prestations comme pour un logiciel propriétaire.

Pour les logiciels libres, il n'y a ni marketing, prospection ou présence commerciale chez les clients. Les fournisseurs « classiques » viennent voir leurs clients, leur affectent des gestionnaires de comptes, alors que des sociétés qui éditent

des solution Open source sont présentes dans les forums mais ne sollicitent pas les clients, ni ne connaissent leurs besoins. C'est donc à **chacun de faire sa propre veille technologique pour repérer des solutions Open source** en réponse aux projets de sa société, de fréquenter les communautés sans attendre d'être démarché. On se doit d'être proactif, investiguer et aller à la rencontre des sociétés de l'écosystème Open source. Certains éditeurs de logiciels libres facturent même leurs journées d'avant-vente.

Cette approche trouve probablement son origine dans la démarche « geek » (passionné de technologie) où les gens s'impliquent, s'auto-forment en cherchant des connaissances sur internet. Dans le monde Open source, en investissant du temps, on trouve -de manière autonome- toutes les informations nécessaires. A contrario, pour les solutions propriétaires, ce sont les informations de nature marketing, commerciale ou un résumé sans détail technique qui sont mises à la disposition de tous. Difficile ainsi de se faire soi-même une idée exacte de ces solutions propriétaires sans passer par des services commerciaux ou se connecter à un espace client.

Ces différences impliquent -pour l'Open source- une **approche spécifique** sur différents items tels que le **Support, les Achats, la Gestion des compétences ou encore la Vie du produit.**



Support et expertise Open source



Au-delà de quelques déploiements sur des sujets de niche avec une forte expertise en interne, une société - dès qu'elle utilise plus de 30 ou 40 logiciels libres - ne peut pas déployer une solution sans se doter de support. Pour produire et fournir un service, garantir la Production IT aux Métiers, il faut pouvoir assurer le support en achetant du service à l'extérieur ou bien attribuer des ressources internes.

Pour obtenir un niveau de service similaire aux solutions propriétaires il faut acheter :

- ✓ de l'expertise pour le Build
- ✓ du support pour le Run
- ✓ des **distributions** pouvant être considérées comme un service de packaging (par exemple, la distribution Ubuntu de Linux, la distribution Cloudera de Hadoop ou encore la distribution OpenStack de Red Hat). Les technologies OpenStack ou Hadoop embarquent un grand nombre de logiciels (dont la plupart sont sous des licences permissives) mais elles sont compliquées à mettre en œuvre, d'où l'intérêt présenté par les distributions car elles offrent un assemblage garantissant un bon fonctionnement dans la durée. Ces distributions épargnent le travail d'intégration, tests et vérification de bon fonctionnement aux entreprises.

À la question :

«Pensez-vous que l'organisation du support soit un frein au développement de l'Open source dans votre entreprise et pourquoi ?»



La réponse de plusieurs sociétés est affirmative, mais dans quelle mesure ?

Le support est un point clé qui fait débat. Contrairement à un logiciel commercial, il pose problème aux Métiers, aux juristes, à la Direction Générale. Certaines DSI refusent d'implémenter des solutions informatiques Open source sur des sites considérés comme sensibles, une usine de production par exemple, par peur du risque encouru.

Mettre de l'Open source sur des applications Métiers suscite plus d'exigences que pour des applications destinées à la DSI. Pour une application Métier, il faut parfois confirmer que l'on a souscrit un contrat de support auprès d'un éditeur. L'équipe IT se sent davantage responsable d'une solution Open source. Ainsi, le premier filtre « de précaution » vient de l'IT qui se montre souvent très (trop ?) prudente. Certains citent, par exemple, des réticences à migrer des applications sur une base de données PostgreSQL, alors que la réalité montre que c'est une base de données très fiable.

À-t-on vraiment besoin d'acheter du support ?

Une grande banque souhaite implémenter la version communautaire de CentOS pour remplacer du Linux Red Hat avec souscription payante sur 70% des environnements. Les clients internes insistent pour avoir du support payant CentOS alors que très peu d'incidents ont été constatés dans l'année avec la version Red Hat. A chacun donc d'apprécier son risque.

Autre stratégie : si on ne peut pas choisir CentOS, il peut s'avérer judicieux de déployer partout du Red Hat pour homogénéiser les environnements, car il n'y aura pas forcément d'économie de souscription. Tout dépend des modalités de licensing du support définies dans le contrat.

Un Etablissement public de taille moyenne explique avoir **choisi un support mixte** avec, d'une part, trois agents en charge du support interne et, d'autre part, un contrat avec une ESN spécialisée sur PostgreSQL. En cas de problème, l'Etablissement questionne la Communauté et, en l'absence de réponse, il s'adresse à ce prestataire pour obtenir le patch.

Un grand Organisme du secteur public a **scindé son approche du Support** pour des solutions Open source en trois catégories :

- pour les solutions Métiers non critiques : le support est assuré en interne avec la Communauté concernée et l'Organisme complète avec de l'achat d'expertise auprès d'un prestataire spécialisé. Mais les ressources « pointues » peuvent s'avérer difficiles à trouver.
- pour des solutions critiques, cet Organisme fait systématiquement des appels d'offre pour se ré-assurer auprès d'un prestataire externe... reproduisant ainsi un schéma éditeur classique.

Une banque dit utiliser a minima un contrat de maintenance pour les solutions Open source qui couvrent des applications critiques et également sur des distributions enrichies fonctionnellement par des tiers. Si possible, elle s'efforce de ne pas activer de maintenance sur les environnements de développement et/ou non critiques.

Un grand acteur du secteur de l'énergie a souscrit un **contrat support auprès d'une société qui propose un catalogue de 150 solutions Open source**. Le ticket d'entrée pour un mode « support global » est assez élevé et il convient d'être précis à propos des services fournis derrière l'appellation « support ». La grille d'intervention peut être assez large : du support au Conseil, tests d'intégration... Cette société a classifié les niveaux de services ainsi que les critères permettant de savoir quand activer ou pas la prestation de support. Elle achète ponctuellement de l'expertise pointue et chère mais jugée rentable. Il faut anticiper : ce type de compétences ne s'acquiert pas en mode urgence.

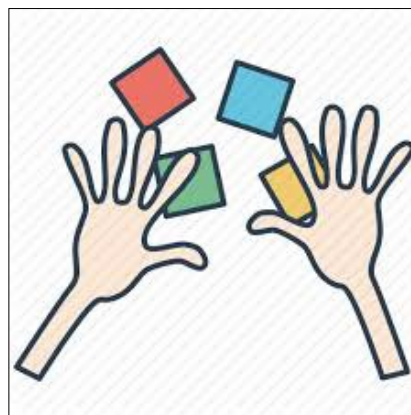
Une grande ville de France utilise des solutions Open source depuis 15 ans et **se charge du support en interne** - exemple : des milliers de serveurs Tomcat fonctionnent sans support Linux. Le support souscrit auprès de Red Hat n'était pas utilisé, il a été décidé d'interrompre le contrat support. Par ailleurs cette Ville achète de l'expertise pour pallier d'éventuelles défaillances de compétences internes.

Un grand organisme paritaire n'a pas hésité à bâtir depuis 2014 une filière sur l'écosystème x86/Linux conçue pour **supporter de véritables applications de gestion critiques** (référentiels Métiers, applications

et flux Métiers) avec des exigences de performance, disponibilité, sécurité etc. de même niveau que celles de la filière traditionnelle (propriétaire).

Cette filière réserve une place importante aux solutions Open source (Red Hat, PostgreSQL, Hibernate, JBoss, Apache, Spring Batch, Nagios Centreon, Prometheus, Grafana etc.).

En tous cas, c'est un monde où il faut davantage s'impliquer dans l'appropriation de l'outil.



Avoir un contrat de support constitue parfois un passage obligé particulièrement pour les logiciels sensibles et critiques, mais comment trouver de bons prestataires qui offrent un support analogue à celui offert par un grand éditeur ? Au dire des Crippiens, avoir davantage de sociétés sur le marché, qui fournissent de l'expertise sur des technologies Open source pour le Développement et l'architecture de solutions, serait un atout pour son adoption.

Autre point de vigilance : « prendre les outils pour quoi ils sont faits »

Un opérateur télécom utilise l'image suivante : « on peut peindre avec un balai mais c'est mieux de le faire avec un pinceau ». Utiliser les solutions Open source pour leur usage prévu simplifie le support. Il faut être attentif aux évolutions de leurs fonctionnalités.

Cet opérateur télécom insiste sur la nécessité :

- d'intégrer -dès le début du projet- les multi-compétences nécessaires en termes de support, sécurité etc. autour de la solution Open source
- de tenir compte du coût de la prise en main de ces solutions
- et de son interconnection aux outils existants.

Pour favoriser l'adoption des solutions Open source, il recommande d'expliquer aux Opérations ce que l'équipe va gagner en utilisant ces technologies, de clarifier leur rôle, notamment en ce qui concerne le support.

La question du risque est une notion variable :



✓ elle dépend de la **criticité d'une application telle que la perçoit la DSI**. Par exemple, un constructeur automobile estime que l'usage de l'Open source est problématique pour une usine alors qu'un site web grand public est largement pourvu de solutions comme Tomcat, sans que cela ne pose problème, car la perception de la criticité est différente.

✓ **Si un éditeur ne supporte pas son progiciel sur une base de données Open source**, la Société n'a pas d'autre choix que de se passer de support (mais difficile si cela concerne une application jugée critique) ou bien d'utiliser la solution sur une base propriétaire. **Mener des actions auprès des éditeurs** de progiciels afin qu'ils supportent leur produit sur une base de données ou un serveur d'application Open source peut aider à développer son adoption par les entreprises.

Un Ministère explique qu'il met cela comme critère dans ses Cahiers des Charges et indique clairement sa préférence pour une

base PostgreSQL. Ce qui est permis puisque PostgreSQL n'est pas une technologie exclusive d'un éditeur : n'importe qui peut proposer du service ou l'intégrer.

✓ En cas d'incident, le service IT ne peut pas le résoudre sans l'aide de l'éditeur, qui y consacre ou non des ressources en fonction de ses propres priorités. Ceci, contrairement à l'Open source, où **les entreprises peuvent faire du debugging** si elles sont pressées. De ce fait, on peut considérer que le risque est moindre.

✓ Concernant la pérennité d'un éditeur propriétaire, la **Direction des Achats dispose de métriques sur le risque** lié à la taille de l'entreprise etc. A contrario, elle ne dispose pas de métriques équivalentes permettant d'évaluer le risque économique de solutions Open source.

✓ Dans un contexte **Cloud et réversibilité du choix** de l'opérateur, l'indépendance est d'autant plus assurée que l'on utilise des technologies Open source. C'est bien l'argument mis en avant par Google qui dit soutenir les acteurs de l'Open source comme MongoDB pour qu'ils puissent vivre. D'autres opérateurs proposent des technologies propriétaires, ce qui compromet les chances de réversibilité de l'entreprise.

Pour conclure sur la question du risque :

• Dans certains domaines Métiers, il vaut mieux être pragmatique et ne pas forcer l'utilisation de l'Open source. Il est préférable de délimiter les champs « super-sensibles » et de **se concentrer sur l'adoption du libre dans d'autres domaines qui serviront** peut-être de **preuves de succès**. C'est davantage une problématique DSI que Direction Générale.

• Certaines entreprises affirment que, lorsque l'on a utilisé le modèle Open source sur des cas concrets (logiciel, base de données...), **on apprend à mieux interagir et à ne plus avoir peur de ce modèle**.

Achat Open source et relations fournisseurs



Ainsi que l'indique un Responsable Achats d'un grand ministère : « **on traite un Projet Open source comme un Projet normal** » c'est-à-dire :

- Connaître ses besoins actuels et futurs
- Imaginer le meilleur scénario et le pire puis valoriser leur impact, probabilité, les manières de les cadrer
- Mettre en concurrence les réponses aux besoins, les challenger, sélectionner celle répondant au mieux à l'ensemble des besoins de la Société dans le temps.

Achat Open source : de quoi parle-t-on ?

Rappelons qu'il s'agit **d'achat de prestations de services** (adaptation, installation, maintenance corrective et évolutive, formation) et non d'acquisition ou de droit d'utilisation de logiciels.

Le type de licence (Achat ou Usage) n'est pas important à moins que le but ne soit de distribuer des logiciels « construits » à partir de logiciels Open source.

Toutes les licences garantissent : la liberté d'étudier le fonctionnement du logiciel, de redistribuer des copies du logiciel, d'améliorer le logiciel et de publier ses améliorations. Il existe deux grandes familles de licence de logiciel libre : avec Copyleft et sans Copyleft (permissive).

Le rôle de l'acheteur Projet

L'acheteur aide l'équipe projet à construire le « **puzzle** » de la meilleure manière possible en fonction des contraintes imposées. Il peut jouer un rôle significatif dans l'introduction d'une solution Open source en assistant notamment l'équipe Projet pour les questions suivantes :

- Comment le Client interne envisage-t-il de traiter le sujet ?
- Est-ce que le fournisseur est déjà plus ou moins choisi ou choix contraint ?
- Est-ce que le sujet lui est familier ? Ou est-il dans le portefeuille de l'un de ses collègues Acheteurs ?
- Peut-on réduire les besoins exprimés par les Projets de façon à ce que l'expression des besoins ouvre la porte aux logiciels libres ?

Pourquoi le modèle propriétaire est-il souvent apprécié par les Acheteurs ?

Car c'est un modèle « facile ». L'éditeur propriétaire a normalement une stratégie commerciale, produit... Pour survivre, ces modèles doivent être pérennes dans le temps. L'organisation Cliente a donc peu de questions à se poser.

La manière de travailler avec les acteurs économiques autour de ces logiciels libres dépend de l'entité qui est propriétaire du code source : Editeurs, Communautés ou Fondations ?



- Plus la **Communauté est fermée**, plus l'action d'achat est proche de celle menée avec un éditeur propriétaire. L'argument « souveraineté » est moins vrai. Et plus on se rapproche du modèle éditeur, moindre sont les gains constatés sur l'achat de la solution Open source.

Par exemple : **un seul acteur économique détient tous les droits de propriété** (éditeur Open source) - Alfresco Software détient tous les droits pour le logiciel Alfresco.

- Une **fondation** (ensemble d'acteurs structuré) **détient les droits de propriété.**

Par exemple : Apache Software Foundation détient les droits de propriété sur Tomcat, The Document Foundation détient les droits de propriété sur LibreOffice.

- Une **communauté informelle de développeurs**, plusieurs dizaines de milliers, détient des droits sur le code du noyau Linux.

Critères pour mesurer l'ouverture d'une communauté

- Gestionnaire de code source public
- Mise à jour de sécurité publique
- Publication de versions binaires
- Gestionnaire de ticket public pour remonter des anomalies
- Forum d'échange ou d'entraide public
- Documentation complète publique
- Pas de modèle Freemium ou Open Core (fonctionnalités complémentaires sous licence propriétaire)
- Absence d'un réseau de partenaires agréés
- Développeurs individuels ou de multiples entreprises

L'ouverture de la communauté est déterminante pour la stratégie Achat de logiciel Open source.

Quand il s'agit d'une Communauté ouverte, pour sourcer des fournisseurs, il faut mener une action spécifique : recenser les besoins pour un projet donné et évaluer les fonctionnalités de la solution. Il n'est pas facile de connaître l'offre de service et de trouver des interlocuteurs auprès des communautés.

A l'inverse, pour une Communauté fermée, l'offre de service est facile à connaître mais la concurrence est inexistante. Pour le secteur public, cela nécessite un marché négocié sans pouvoir le justifier en vertu d'une exclusivité sur le code source.

L'acheteur accompagne l'équipe projet pour « challenger », mettre en question, la maîtrise d'ouvrage et ses choix fonctionnels.

Par exemple, un support 7 jours sur 7 de 8 à 19 heures est-il vraiment indispensable ? L'acheteur met en balance les coûts, le respect de la forme du marché etc.

Dans le cas où l'entreprise doit respecter le process des marchés publics, les administrations notamment, il faut **mettre en concurrence plusieurs sociétés**. Pour des entreprises de droit privé, c'est également recommandé, sauf en cas de certificat d'exclusivité, lequel ne peut pas être fourni par un éditeur de logiciels libres.

Cependant, certaines sociétés démontrent une meilleure expertise sur telle ou telle solution. La Jurisprudence établit que, dans la mesure où le code est libre, chaque prestataire peut se former et donc se positionner sur la fourniture du service demandé. Ainsi l'Administration a le droit de **stipuler dans un marché qu'elle souhaite acheter du service sur un produit libre spécifique** (par exemple Libre Office) et exclure -de fait- telle suite bureautique propriétaire.

Un Institut de Recherche scientifique inscrit des contraintes Open source dans toutes ses clauses d'achat logiciel, mais sans grand succès pour le moment. Le succès viendra aussi du nombre de demandes effectuées auprès des éditeurs sur la technologie Open source. Tout dépend du monopole, ou pas, de l'éditeur sur son domaine. C'est tout l'art de mettre ces critères dans un Cahier des Clauses Techniques Particulières (CCTP) pour obtenir l'exigence technique Open source sans s'enfermer dans un choix à solution unique.

L'utilisation des plateformes de publication des marchés publics, de par sa complexité, exclut les petits éditeurs tout comme les acteurs de l'Open source, qui ne sont pas structurés pour faire de la veille commerciale sur ces plateformes. Parfois se crée un consortium de petites sociétés pour répondre à un grand marché lancé par l'Administration.

D'où l'**importance de faire de la veille technologique** pour les organisations qui souhaitent privilégier les solutions Open source.



Enfin, les Organisations du secteur public comme les entreprises du secteur privé peuvent avoir intérêt à passer plusieurs marchés ou contrats :

- un contrat support auprès d'une ESN généraliste concernant le support de multiples produits
- et d'autres contrats relatifs à l'achat d'expertise pointue -mais plus coûteuse- portant sur une technologie particulière, avec un objectif Architecture ou Développement, auprès de sociétés spécialisées.

En effet, un petit éditeur ne pourrait pas apporter une réponse concurrentielle pour un contrat limité à quelques produits.

Toutefois, cette approche est **conditionnée par l'état de l'offre** : il existe des domaines où -en France- il n'existe pas d'offre structurée d'expertise, seulement une offre communautaire.

On assiste à une certaine **évolution marchande** ou «**Marchandisation**» de l'**Open source** dénoncée parfois comme une dérive. Un opérateur télécom explique choisir des solutions « full Open source » (telles que Docker, Elasticsearch/ELK...). De façon générale, il ne souhaite pas payer pour des fonctionnalités de produits Open source. Il accepte

de se priver de **certaines fonctionnalités payantes** et l'assume. Lors des déploiements, les Projets n'ont d'ailleurs pas remonté de besoin justifiant l'achat de ces fonctionnalités. L'approche se fait au cas par cas.

En tous cas, les outils et les services pour les exploiter, le support, l'expertise pour les intégrer au Système d'Information existant, certaines fonctionnalités etc. tout cela n'est pas gratuit et se rapproche du modèle éditeur.

Le remède à cette dérive marchande -selon certains Crippiens- existe : **il faut que les Sociétés contribuent davantage pour faire évoluer les outils Open source.**



Un grand opérateur du secteur énergie insiste sur la nécessité de contribuer avec du code, de la documentation etc. A défaut ce sont les éditeurs qui vont piloter les évolutions produits et il sera difficile de maintenir la gratuité. Par exemple, la solution ELK est issue du travail de (environ) 500 ingénieurs dans le monde, qui coûtent relativement cher ce qui explique que les nouvelles fonctionnalités deviennent payantes.

Gestion des compétences Open source : attirer, former, retenir



Si l'on va massivement vers l'Open source, il faut disposer d'un socle de compétences minimum, même sans vouloir devenir totalement autonome.

Or, le monde du libre affiche une pénurie de compétences. Contrairement aux éditeurs, les acteurs de l'écosystème Open source n'ont pas développé des relations de longue date avec les universités. D'où la nécessité **d'acquérir des compétences en agissant de plusieurs manières** :

- ✓ trouver les talents et les attirer
- ✓ les former, dans le cas de recrutement de jeunes diplômés ou de collaborateurs IT embauchés sans compétence sur le sujet Open source
- ✓ les transformer, dans le cas de collaborateurs internes
- ✓ les retenir.

Un grand ministère -qui a des difficultés à recruter des informaticiens par concours- fait appel à des ESN et essaie de mettre en place des contrats de longue durée avec des prestataires externes ayant de solides compétences Open source.

Un constructeur automobile dit recruter « au compte-goutte » sur des fonctions critiques (Sécurité, Télécom, Data) mais sans politique de recrutement Open source.

Sur ce sujet, il privilégie l'option formation interne par les équipes qui ont conçu telle ou telle solution : socle technique etc.

Pour renforcer les compétences Open source dans le Groupe, une grande banque a lancé en 2016 un programme articulé sur trois piliers :

- Attirer
- Innover
- Contribuer

Le constat de départ était que l'outil utilisé pour la gestion des compétences ne permettait pas d'identifier les compétences Open source de manière suffisamment ciblée et précise, notamment d'évaluer si tel collaborateur connaissait et pratiquait l'Open source ou bien était simplement un utilisateur. Aussi, en 2018, la banque a décidé d'approfondir l'identification des compétences en se focalisant sur l'aspect contribution : quel collaborateur utilise ou contribue à l'Open source ?

Un [programme de sensibilisation a été lancé autour de l'Open source](#) en commençant par des « talks ». Ensuite, lors d'un événement interne visant à promouvoir les diverses réalisations IT de la banque, un flux Open source a été créé ayant pour but de créer des échanges autour du sujet. D'autres initiatives ont été mises en œuvre comme un blog technique, des rencontres etc. pour augmenter l'attractivité et la visibilité du sujet.

Pour former et gérer les compétences, deux Massive Open Online Courses (MOOCs - publications) ont été faits, avec un **questionnaire portant sur le processus de contribution, sur les licences...** (par exemple : Comment je contribue au nom de ma banque sur le repository GitHub ?).

Ont été ciblés 6 à 8.000 développeurs dans le monde.

Ces questionnaires (MOOC licence et MOOC contribution) sont utilisés lors des recrutements pour évaluer le candidat.

Notons que cela ne reflète pas la situation générale dans les sociétés et que cette banque y consacre beaucoup de moyens.



Pour fidéliser et [conserver la motivation des collaborateurs](#) défenseurs de l'Open source des actions sont menées par certaines entreprises. Citons :

- Mettre en place un programme Open source avec un **sponsor Direction Générale** (CIO Groupe) et une structure dédiée de quelques (4 à 5) personnes.
- **Le passage de postes de travail sous Linux** – Les outils utilisés au quotidien sont sous Windows. Mais des tests sont en-cours pour passer en production sur un poste de travail sous Linux, d'ici quelques mois, l'objectif étant de retrouver les mêmes outils que sur leur poste de travail actuel.
- Encourager la contribution Open source en **mettant en place un programme d'expertise**, avec des degrés Expert, principal engineer etc. et en accordant **une journée de contribution par mois** pour contribuer, corriger des bugs...Les meilleurs contributeurs sont valorisés et présentent leurs réalisations. La banque cite comme exemple de contribution : 3.000 lignes de code sur la dernière release d'OpenStack. Autre façon de valoriser les contributeurs : leur offrir un voyage sur un salon OpenStack pour montrer leurs réalisations.
- **Adapter l'organisation du travail** en s'inspirant de celle des communautés Open source - mettre en Open source les projets de la banque, **développer l'Inner Sourcing** avec 30.000 projets sur le GitHub d'entreprise, les mettre en public pour favoriser le partage, encourager le partage de code en interne ou la correction de bugs des autres solutions. Le souhait étant que l'ouverture du code peut aider à la « fertilisation transverse » des développeurs de la DSI. Toutes ces initiatives sont faites en espérant qu'il ne s'agisse pas seulement de réutiliser le code de son collègue mais également de le corriger...
- Stocker sur une forge externe n'est pas suffisant pour créer une Communauté. Il faut trouver d'autres utilisateurs intéressés qui contribuent, il faut **animer une Communauté**, répondre aux questions etc. D'ailleurs, les dates de dernière mise à jour, le nombre de questions en un an etc. sont autant de métriques pour évaluer le dynamisme de la Communauté.

Une autre grande banque parle d'« [Internal Open source](#) » : les créations logiciels sont diffusées sur des GitHub internes mais il n'y a pas de cadre pour sortir des développements internes de la banque vers l'extérieur. Les aspects de Propriété Intellectuelle -au sein du Groupe et encore plus en externe- ne sont pas réglés. En conséquence, des développeurs ont envie de contribuer mais n'en ont pas encore le droit.

Lors de l'embauche de développeurs, cette banque vérifie, malgré tout, la volonté du candidat de contribuer.

Un opérateur télécom fait état d'une forte volonté chez ses développeurs de fournir des solutions en Open source et de contribuer. Mais là aussi, les **aspects juridiques** constituent un frein.

Pour acquérir un socle de compétences Open source, un grand organisme paritaire a choisi d'élaborer un [programme de formation interne](#) destiné à ses ressources internes, sachant qu'il n'avait pas la possibilité d'embaucher.

Il a utilisé son Centre de Formation pour délivrer une formation spécifique à son contexte, impliquant les acteurs qui avaient créé la plateforme Open source et son outillage.

Un grand ministère a adopté une autre organisation : pas de ressource dédiée à l'Open source mais plusieurs [collaborateurs qui ont des missions Open source clairement identifiées dans leurs activités](#), telles que :

- Participer au Comité d'architecture organisé pour chaque projet en préparation, avec comme but d'analyser les technologies, l'architecture retenue, les niveaux de sécurité d'un projet, les logiciels utilisés pour mener à bien le projet, « challenger » les choix techniques et favoriser l'Open source. De plus, le Comité d'architecture évalue si tel logiciel est indispensable en regard des besoins du projet, il conseille aux membres du projet de regarder telle solution libre.

- Faire en sorte que le périmètre du support évolue en cohérence avec les choix de version effectués pour les projets.
- Animation du marché de support des logiciels libres.
- Essayer de converger entre les diverses Directions pour un même besoin : maintenir une cohérence au niveau global de la DSI, essayer de faire déployer davantage de logiciels libres à travers les Ministères.
- Déclencher des études de veille, mesurer la taille et la qualité de la Communauté autour d'une solution, développer une expertise sur de nouvelles briques que le Ministère voudrait intégrer.
- Mener une étude de veille par mois – Ne pas attendre qu'un commercial vienne leur vendre telle solution Open source, sachant que ce commercial n'existe pas.

Dans les entreprises qui n'ont pas identifié de ressources dédiées, ces tâches sont assumées par des architectes techniques, selon un grand constructeur automobile.

Une entreprise qui veut vraiment **privilégier l'Open source met en place une gouvernance affirmée avec les moyens** de la déployer.

Nous avons rencontré peu d'entreprises qui mettent des ressources dédiées. Celles qui le font sont assez matures sur la stratégie Open source et se dotent des moyens de la piloter. C'est un marqueur qu'il existe une politique claire, connue et dotée de moyens de vérifier son respect.

Travailler avec des partenaires



Lorsqu'une société **travaille beaucoup avec des partenaires de type ESN** (Entreprise de Services du Numérique) pour le Développement IT, la question est de savoir comment conduire sa démarche Open source avec eux. Comment animer et contribuer à la communauté lorsque la société a peu de ressources développeurs en interne ?

Le choix d'un grand constructeur automobile a été **d'établir une co-gouvernance technique** Partenaire et Société :

- Les choix techniques sont faits d'un commun accord, chaque fois que c'est possible. La raison est double : avoir un cadre technique qui garantit la productivité des partenaires et convienne au donneur d'ordre. Les séances de pilotage et de travail sont faites ensemble.
- La préférence Open source est affichée et partagée avec les partenaires. Au niveau opérationnel, l'équipe se compose d'un Chef de projet Société et de collaborateurs du Partenaire.
- L'objectif pour la Société est de conserver le contrôle des choix techniques et veiller à l'intérêt commun : l'Open source fait partie des sujets de co-gouvernance avec les partenaires.

Ce constructeur automobile a pris conscience de la nécessité de faire un minimum de customisation sur les technologies standard Open source, pour faciliter leur adoption par les prestataires externes. Mais tout dépend du degré de maturité de la technologie Open source adoptée. Si elle n'est pas mature mais que l'on veut l'utiliser, on a tendance à vouloir la personnaliser. Dans ce cas, ce constructeur recommande de :

- Remettre cette customisation dans la Communauté.
- Développer la customisation sous forme de plug-in sans toucher au coeur du logiciel.

Un Institut de Recherche scientifique explique que ses partenaires sont payés pour développer des solutions avec la condition de les mettre en Open source et d'en assurer le support sinon l'Institut ne signe pas le marché. Dans certains cas, si le partenaire estime que la solution à développer est trop spécifique et refuse d'en assurer le support, alors l'Institut ne retient pas ce partenaire.

Confier le support, ou des travaux à façon, à un prestataire éloigne des communautés. Le mieux est d'anticiper et de prévoir dans le cahier des charge et contrat passé avec la société que celle-ci reverse le code à la Communauté pour que cette dernière aide à faire le support.

Concernant la Propriété Intellectuelle, un grand ministère a l'approche suivante :

- Si le correctif est produit par la société prestataire, la Propriété Intellectuelle lui appartient.
- S'il s'agit de travaux plus complexes, il décide au cas par cas. Le ministère inscrit dans les marchés que la Propriété Intellectuelle lui revient. Le code est reversé aux Communautés au nom du ministère par la société prestataire.



Contribuer n'est pas quelque chose de naturel pour un développeur. Un Institut National explique qu'une proportion minime de son équipe Développement pratique la contribution Open source. La plupart consomme et les contributions relèvent d'initiatives individuelles et ponctuelles.

Certains disent **mesurer les contributions** Open source faites par les développeurs sur leur temps libre, ce qui n'est pas évident.

Une banque a défini cinq projets majeurs pour inciter les collaborateurs à contribuer de préférence sur ces domaines.

Dans le cadre d'un recrutement, le nombre de contribution peut être aussi un critère pour mesurer le « talent » d'un collaborateur.

À noter que l'Etat produit et publie beaucoup de code en Open source.

La conclusion tirée par plusieurs est que :

- L'animation d'une communauté nécessite un certain travail, surtout au début (pour attirer des gens, la faire vivre...), ensuite le travail est davantage de modérer.
- Mieux vaut inclure les actions attendues dans la fiche de poste du collaborateur.
- Chez un grand constructeur automobile, la filière d'expertise correspond à un domaine sur lequel l'entreprise veut avoir un niveau élevé d'expertise en interne. Les « experts » auront comme mission de construire des programmes de formation interne, de « rayonner » à l'extérieur, diffuser leur savoir, contribuer à des Communautés Open source etc.

Pour favoriser le développement des compétences Open source internes, il faudrait :

- Evaluer les Compétences à « haut potentiel » Open source – « Non seulement j'utilise mais je sais aussi contribuer » - Créer un programme d'évaluation pour recruter des prestataires et des candidats, le partager pour que chacun puisse évaluer son niveau.
- Donner du temps dédié pour que le collaborateur contribue.
- Reconnaître -valoriser et récompenser- les personnes qui vont contribuer.
- Former et communiquer largement sur les technologies Open source, même si l'approche peut être de donner des formations assez légères sur les technologies et encourager la montée en compétences de façon autonome sur le terrain.

Pérennité de la solution Open source



Ce produit a-t-il un futur ?

La garantie de la durée de vie du produit fait partie des critères de choix. Or, la pérennité d'une solution éditeur -versus un produit issu de Communautés peu stables- offre davantage de garantie. Comment peut-on quand même évaluer la pérennité d'une solution Open source et choisir de l'implémenter ?

Le **degré d'ouverture des Communautés**, le fait que le **code soit porté par** une seule Société ou par **plusieurs acteurs fait varier le risque**. Un changement de politique de l'unique Société qui supporte le code Open source peut faire sortir ce dernier du périmètre des logiciels libres. De même, une Communauté fermée est un facteur de fragilité pour le produit qu'elle supporte. Citons l'exemple du langage Java développé par Sun puis racheté par Oracle.

Certaines fondations sont pilotées par plusieurs sociétés mercantiles, mais le nombre d'organisations qui agissent autour contrebalance ce pouvoir.

Une communauté de grands acteurs commerciaux garantit un certain équilibre. Par exemple, plusieurs acteurs proposent des services autour d'OpenStack ce qui crée une certaine stabilité autour du produit.

Moins connu que le Gartner, il existe néanmoins des **référentiels permettant de s'informer** :

- La société Smile publie un annuaire, avec mise à jour annuelle, des solutions Open source disponibles par type d'usage, domaine technique et fonctionnel. Ce guide présente 355 solutions Open source professionnelles. Il est disponible en ligne sur son site et peut être téléchargé gratuitement en français et en anglais - Liens : <https://www.smile.eu/fr/livres-blancs/livres-blancs/guide-lopen-source>
- Le **Socle Interministériel de Logiciels Libres (SILL) est le catalogue de référence des logiciels libres recommandés par l'Etat**. Il est construit de façon collaborative par des communautés d'agents informaticiens de la fonction publique d'État et de la fonction publique hospitalière. Les logiciels sont classés par domaine et fonction avec un statut « recommandé » ou « en observation » sur des sujets émergents. Ce catalogue présente l'avis du ministère sur ce produit - Liens : <https://www.mim-libre.fr/> et <https://disic.github.io/sill/>

À noter que la question de la pérennité se pose également pour les logiciels propriétaires. L'avantage du libre étant que, **même en cas d'abandon** par l'unique Société ou Communauté, le **code source reste accessible** et on peut toujours le faire évoluer.

Quelle stratégie Open source adopter ?



Il y a des avantages indéniables à avoir une politique Open source : coût, indépendance etc.

Pour en tirer vraiment partie, encore faut-il mettre les moyens en face de ses choix.

On peut classer ces politiques en 4 grandes catégories :

1) Stratégie « Opportunisme »

Elle est caractérisée par :

- Absence de politique affichée, chaque projet fait ses choix.
- On choisit l'Open source quand on y voit un intérêt économique.

Conseil : Penser à la cohérence des choix technologiques pour éviter de multiplier les solutions, partager le retour d'expérience projet avec les autres membres de l'équipe, préconiser en interne la solution Open source digne d'intérêt. Penser qu'il n'y a pas de service Achats qui va généraliser la solution.

Rationaliser -après coup- le choix effectué et tirer de son expérience une démarche plus systématique.

2) Stratégie « Préférence » Open source

Sur certains domaines, on identifie et promeut quelques solutions.

Il n'y a pas d'organisation dédiée, ce sont les architectes qui préconisent ces solutions.

3) Stratégie « Open source first »

Les choix non Open source deviennent de plus en plus rares.

Des ressources sont identifiées pour encadrer la politique Open source et travaillent dans une organisation structurée.

Citons comme exemple le Socle interministériel des logiciels libres.

Il s'agit d'une politique publiée, contrôlée et le défenseur d'un choix non Open source doit justifier les raisons de sa demande.

4) Stratégie « Contribution »

Une telle stratégie implique de participer aux projets au-delà de son entreprise et de contribuer aux communautés.

Autant dire que de telles Organisations sont rares...



En conclusion

Au fil du temps, chaque société a mis en place sa propre stratégie concernant les logiciels Open source : opportuniste, pour le choix de nouvelles solutions si on peut démontrer un ROI supérieur à celui d'une solution propriétaire, « Open source first » etc.

Le Groupe de Travail open source du CRiP recommande une vraie politique « Open source first » qui, parmi les différentes approches possibles du logiciel libre, est celle qui sert le mieux les grands enjeux, maîtrise des coûts et souveraineté notamment.

Mais, si en termes de coût, de fonctionnalités ou d'indépendance envers les fournisseurs, la solution Open source s'avère moins intéressante, alors on se doit de comparer objectivement les solutions libres et propriétaires.



Lorsqu'une entreprise est convaincue que l'Open source est LA bonne solution, il faut adopter et communiquer clairement la position « Open source first », ce qui aide à dépasser les débats internes. En effet, l'Open source nécessite de quitter sa zone de confort, ce qui est plus facile lorsque le management cautionne clairement cette position, en mettant en place une gouvernance adéquate avec un pilotage et des relais, des collaborateurs qui portent ces sujets. La politique Open source doit être déployée sous tous ses aspects, techniques, compétences, organisation etc.

Au-delà d'afficher une préférence Open source, la Direction devra **l'accompagner par une politique de formation et d'achat** pour apprendre aux collaborateurs à surmonter les difficultés rencontrées.

Les résultats en termes de gains économiques et acquisition de compétences s'obtiennent par les Organisations qui ont une politique claire « Open source first » et **outillée par une gouvernance**.

Citons un grand Ministère qui mène des expérimentations sur un poste full Linux depuis quelques années et a travaillé à supprimer toutes les adhérences entre les applications et le poste de travail. Sont concernés les collaborateurs IT et les Métiers qui présentent les caractéristiques appropriées pour le déploiement de ce type de poste libre : navigateur, bureautique, email sont libres. Des centaines de collaborateurs IT utilisent un poste entièrement sous Linux. Le poste de l'agent sous Windows10 est déjà orienté libre avec un maximum d'applications sans adhérence avec la bureautique.

Une approche volontaire à mener sur du long terme ...



Remerciements

Nous remercions très sincèrement :

- les contributeurs -et leur Société- qui ont partagé leur retour d'expérience avec le Groupe de Travail Open source
- les participants au Groupe de Travail qui ont permis d'enrichir les échanges
- et tout particulièrement, pour leurs idées et leur implication précieuse pour la réalisation de cette Fiche Pratique :

Thierry Aimé – Ministère de l'Economie et des Finances, Responsable des Logiciels Libres

Noël Cavaliere – Groupe PSA, Chief IT Strategy & Technical Architecture

Eric Gillon – Agirc-Arrco, Responsable Normes et Conformité Logiciels

Animation du Groupe de Travail et assistance éditoriale : **Monique Castruccio - CRiP**



Groupe de Travail Open Source

Pilote :



Noël CAVALIERE
Groupe PSA
Chief IT Strategy & Technical Architecture

Animation et coordination : Monique Castruccio (CRiP)

CRiP
Technology & Digital

Club des Responsables Infrastructures, Technologies et Production IT
24 rue Erlanger 75016 Paris - contact@crip-asso.fr



www.crip-asso.fr

En application de la loi du 11 mars 1957, il est interdit de reproduire ; sous forme de copie, photocopie, reproduction, traduction ou conversion, le présent ouvrage que ce soit mécanique ou électronique, intégralement ou partiellement, sur quelque support que ce soit, sans autorisation du CRiP.