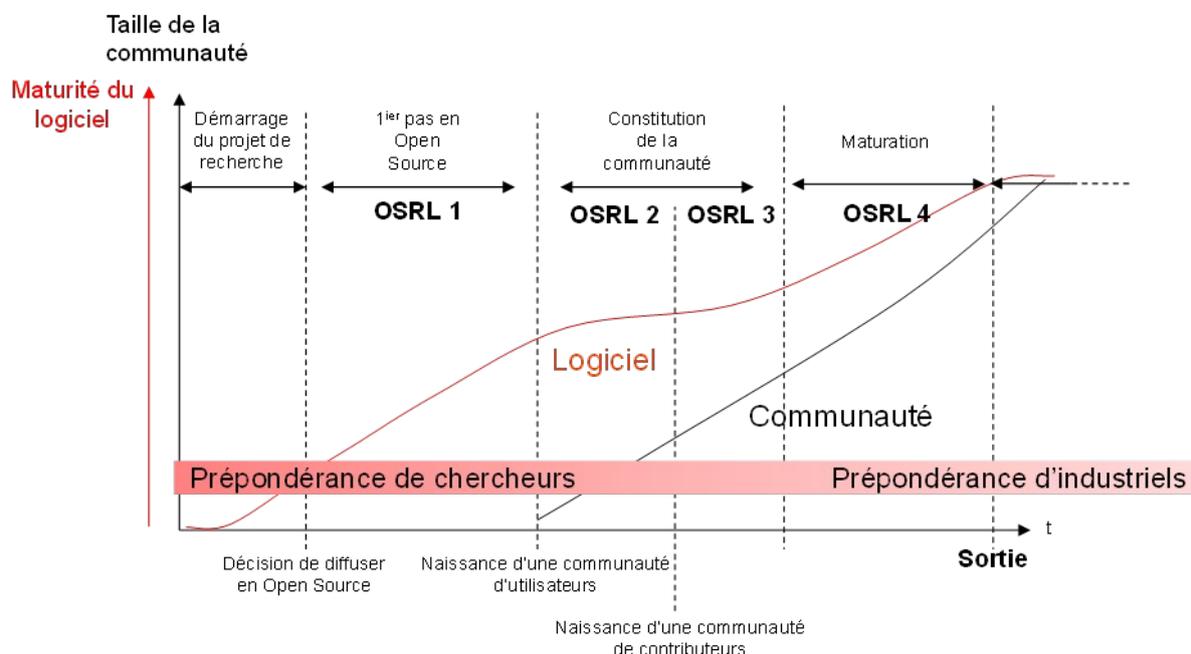


Le cycle de vie idéal des logiciels libre / open source

Patrick Moreau

le 13/02/2024

Publié sous licence CC By-SA 4.0



Par « cycle de vie idéal », on entend un cycle de vie à-même de permettre de capitaliser sur la valeur ajoutée de la communauté pour participer au développement, pour accroître l'impact et la « transférabilité » du logiciel.

Nous considérons dans la suite de ce document qu'une communauté est constituée :

- d'utilisateurs passifs : utilisent simplement le logiciel,
- d'utilisateurs actifs : utilisent le logiciel, remontent des bugs, suggèrent des améliorations, expriment des besoins, améliorent éventuellement la documentation (ces utilisateurs sont considérés comme des contributeurs),
- de contributeurs développeurs : adaptent le logiciel, contribuent directement sur le code, le portent sur d'autres systèmes d'exploitation, etc.

A l'instar de l'échelle TRL (*Technology Readiness Level*) utilisée pour les systèmes industriels, l'échelle OSRL (*Open Source Readiness Level*) est proposée :

- OSRL 1 : juste après la diffusion sous licence open source, absence à ce stade d'utilisateurs externes,
- OSRL 2 : une communauté d'utilisateurs passifs commence à voir le jour,
- OSRL 3 : la communauté de contributeurs « utilisateurs actifs », et éventuellement de « développeurs », s'étoffe,
- OSRL 4 : la communauté de contributeurs actifs se consolide et le processus de maturation de l'édition du logiciel se met en place, en vue de la pérenisation de l'édition du logiciel.

Selon les logiciels, il est constaté une durée 5 à 15 ans pour la totalité du cycle.

Le cycle de vie proposé n'a pas vocation à être présenté comme étant le seul modèle pertinent mais comme celui qui, de notre point de vue, peut-être repris par un grand nombre de projets open source de la recherche publique.

Les quatre phases de ce cycle font l'objet d'une description plus détaillée ci-après.

Décision de diffuser en open source

Une fois qu'un logiciel développé au sein d'une équipe de recherche a atteint une certaine maturité, sa diffusion sous licence libre peut être alors considérée. Le choix de la licence est un élément majeur impactant la typologie de la communauté à venir.

OSRL 1 : les premiers pas : L'ANCRAGE

Le développement est réalisé par des chercheurs, ingénieurs et doctorants ou stagiaires dépendant d'une équipe de recherche en interne ou éventuellement, dans le cadre d'un partenariat avec un tiers. Le développement n'est pas encore ouvert ou rendu visible. Le logiciel est alors généralement tout juste une preuve de concept. La documentation utilisateur, idéalement agrégée d'exemples, et des outils (site web, mailing-list, bug tracker, forum...) se mettent alors en place pour attirer puis accueillir les premiers utilisateurs.

OSRL 2 : constitution d'une communauté d'utilisateurs : L'IMPACT

Cette phase débute au moment où les projets commencent à identifier des utilisateurs externes à l'équipe de recherche. Ces utilisateurs sont au départ passifs. Au fur et à mesure que le nombre d'utilisateurs grandit, une communauté se forme. Ces utilisateurs sont dans un premier temps des chercheurs, voire des ingénieurs, dans la recherche publique ou dans les services R&D de l'industrie. Ces utilisateurs téléchargent le logiciel et peuvent s'inscrire aux mailing lists.

Dès cette phase, le logiciel peut avoir de l'impact sur l'écosystème (éditeurs, intégrateurs et industries) en étant réutilisé par ce dernier (par bout de code ou réutilisation des algorithmes, le plus souvent à ce stade). Cet impact est difficile à estimer du fait qu'il n'y a pas de retours formalisés.

OSRL 3 : constitution d'une communauté de contributeurs : L'IMPACT

Cette phase débute au moment où les projets admettent des contributeurs externes à l'équipe de recherche. On considère qu'une communauté de contributeurs (utilisateurs actifs ou développeurs) existe dès l'instant où l'on constate de l'activité sur les mailing lists ou/et le bug tracker et le forum. La communauté peut commencer à influencer sur le développement du logiciel.

OSRL 4 : maturation :

Le logiciel ne répond plus seulement à des problématiques de recherche mais de plus en plus à des problématiques d'applications industrielles. La communauté s'intègre de plus en plus à la stratégie produits, et aussi outils, des industriels. En conséquence, le profil de la communauté peut alors être amené à évoluer.