
Les logiciels libres comme publication scientifique d'un nouveau type

Bernard Mourrain

Notre activité de recherche, de manière simplifiée, est une succession d'étapes d'approfondissement, d'innovation et d'explication. Un résultat (mathématique) n'apparaît dans la communauté que s'il est présenté, expliqué et justifié auprès des pairs. Cela se fait par le biais de publications, évaluées par ces pairs, accessibles dans les bibliothèques ou de manière électronique, par internet. Cela passe aussi souvent par des présentations dans des séminaires, des conférences, avec ou sans actes. Il est clair que cette activité de présentations et de publications des travaux est un moteur essentiel des avancées du domaine. Elle permet de faire le point des connaissances existantes et d'explorer de manière plus pertinente ce qui n'est pas connu, sans devoir reprendre systématiquement les mêmes chemins et franchir les mêmes obstacles.

Dans certains domaines comme, par exemple, la géométrie algébrique effective ou le calcul formel, dans lequel je travaille, la recherche de résultats d'ordre théorique peut être associée à des motivations algorithmiques, voire expérimentales. Le développement de logiciels pour valider des idées nouvelles, pour expérimenter des méthodes qui semblent prometteuses, ou pour mieux comprendre certains phénomènes, joue un rôle important. Cette étape d'expérimentation et d'implantation permet de valider des réponses à certaines questions tout en en posant de nouvelles. C'est également un moteur des avancées du domaine.

L'écriture de logiciels participe donc au progrès des connaissances. Elle peut être perçue comme une forme de description d'un certain savoir-faire ou d'une certaine expertise. Si, de plus, le logiciel est distribué, cette activité pourrait également être considérée comme une publication scientifique à part entière. Or, c'est loin d'être le cas. L'affichage de développements logiciels dans un dossier de candidature à un poste de chercheur, par exemple, n'a pas du tout le même poids que la publication d'articles, même si cette activité logicielle se révèle plus chronophage et parfois plus exigeante. Pourquoi en est-il ainsi ?

Pour pouvoir jouer un rôle équivalent à celui d'une publication classique, il faudrait d'abord pouvoir bâtir sur l'existant. Autant dans un article, il est facile (et recommandé) de citer des publications connexes, pour mieux les étendre ou les enrichir. Autant quand il s'agit de développement logiciel, ceci devient un véritable casse-tête.

Un obstacle important est certainement l'accessibilité. Ce mécanisme de référence d'un logiciel à un autre est souvent bloqué par les licences qui leurs sont associées. L'utilisation de licences dites propriétaires empêche un tiers de s'appuyer simplement sur un logiciel pour l'étendre. Une conséquence directe est que des codes ayant les mêmes fonctionnalités sont réécrits à plusieurs endroits, sans vraiment utiliser l'expertise sous-jacente et donc sans vraiment progresser. Il s'agit d'une vision propriétaire des connaissances. En mathématiques, peut-on dire que la commutativité de la multiplication des nombres rationnels ou le théorème

de Pappus sont la propriété de quelqu'un ? Au niveau logiciel, il devrait en être de même. C'est, pourtant, cette règle de bon sens que les brevets logiciels tentent de forcer.

Les logiciels dits libres, au contraire, garantissent l'accessibilité tout en favorisant la possibilité de réutilisation externe. Ce type de licence permet en effet l'utilisation et l'extension d'un logiciel existant dans la mesure où le nouveau logiciel (étendu) est distribué avec la même propriété de copie et d'extension. Cela permet effectivement de considérer la diffusion d'un logiciel comme une publication scientifique. L'enjeu est du même ordre que pour les publications classiques : faire progresser les connaissances et mieux comprendre ce qu'on ne connaît pas.

Pour aller encore plus loin dans cette logique, pourquoi ne pas envisager d'évaluer ces travaux logiciels comme sont évalués les articles, en mettant en place un mécanisme de référé adapté à ce nouveau type de publication. Ceci permettrait d'analyser la pertinence des travaux et de valider leur qualité, tout en reconnaissant le travail des auteurs à une plus juste valeur. Cela permettrait également de construire de nouveaux développements sur des bases plus solides, sans devoir réinventer la roue. Enfin, cela fournirait un suivi historique permettant sans doute, *a posteriori*, de mieux comprendre l'évolution des connaissances dans le domaine.

Bien sûr ce modèle basé sur l'échange de savoirs est en opposition avec un modèle propriétaire et fermé, qui voit les logiciels comme des produits. On pourrait être tenté de penser que ce dernier modèle est dicté par des raisons économiques et que tout autre fonctionnement n'est pas possible. Du point de vue d'un utilisateur, vaut-il mieux devoir payer un nouveau produit chaque fois qu'une version améliorée apparaît, ou plutôt contribuer (financièrement ou directement) à l'évolution du logiciel, suivant ses besoins ? C'est ce dernier type de solution « plus économique » que favorisent les logiciels libres. Dans un modèle propriétaire, le prix de vente est directement lié à l'effort de maintenance et d'évolution du logiciel. Dans un modèle libre, cette évolution est ouverte à la communauté des utilisateurs et seul un service basé sur ce logiciel peut être payé. Les brevets et licences non-libres sont donc, d'une certaine manière, des freins à l'évolution d'un logiciel.

Personnellement, j'ai fait le choix d'une licence libre (de type GPL), pour le développement d'une bibliothèque logicielle synaps (voir <http://www-sop.inria.fr/galaad/software/synaps/>), dédiée au calcul symbolique-numérique et à la résolution d'équations polynomiales. Des contributions externes venant de différentes universités françaises (Limoges, Paris 6) ou étrangères (Athènes, Santander, Buenos Aires) enrichissent régulièrement ses fonctionnalités. Avec plusieurs groupes de la communauté de calcul formel impliqués dans des activités logicielles, nous proposons à travers le projet roxane (voir <http://www-sop.inria.fr/galaad/software/roxane/>), de coordonner nos développements et de rendre nos logiciels interopérables. Deux écoles d'été, l'une à Giens du 15 au 20 septembre 2002, l'autre à Sophia-Antipolis du 4 au 9 septembre 2005, présentant des tutoriels de logiciels libres en Algèbre et Géométrie, ont également été organisées dans le but de promouvoir l'utilisation de ces outils.