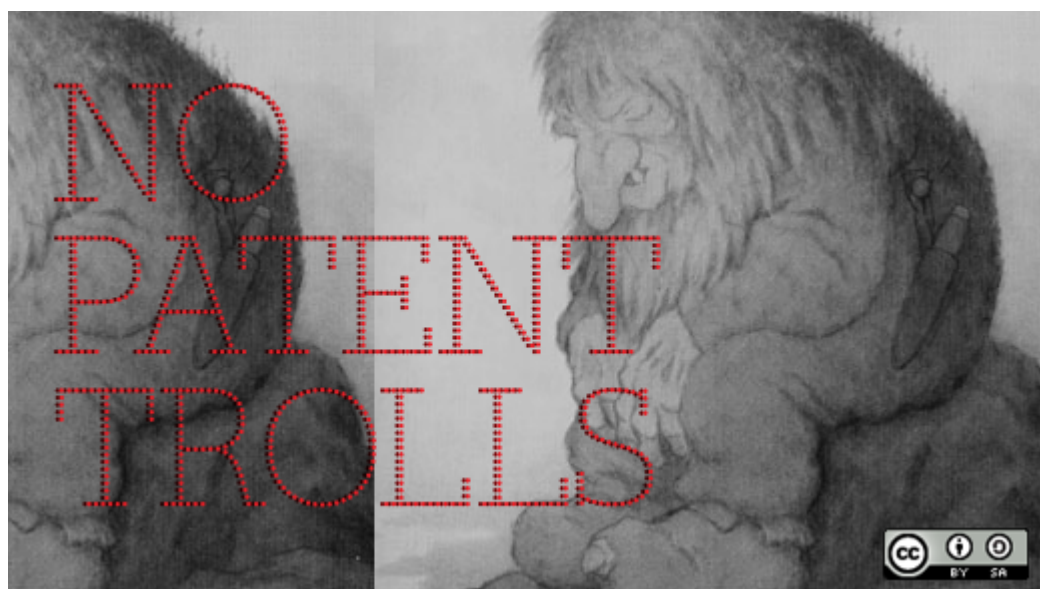


Protéger le secteur du logiciel des brevets, par Richard Stallman

En novembre dernier, Richard Stallman faisait paraître dans le magazine Wired un article important sur l'épineuse et dangereuse question des brevets logiciels (ou plutôt « brevets sur des idées informatiques » comme nous le verrons ci-après).

Un article que nous nous sommes empressés de traduire via notre circuit, désormais classique, compte Twitter + Framapad, et qui a été relu et corrigé par la liste « trad-gnu » de l'April.



Protéger le secteur du logiciel des brevets

Giving the Software Field Protection from Patents

Richard Stallman - version du 02 février 2013 - Gnu.org (CC BY-ND)

(Traduction Framalang : satbadkd, Thérèse, DarthMickey, geecko, Marc, igor, EEva, greygjhart)

Une première version de cet article a été publiée sur Wired en novembre 2012.

Les brevets menacent chaque concepteur de logiciel, et les guerres de brevet que nous avons longtemps craintes ont éclaté. Les développeurs et les utilisateurs - soit, dans notre société, la plupart des gens - ont besoin de logiciels libres de tout

brevet.

Les brevets qui nous menacent sont souvent appelés « brevets logiciels », mais ce terme est trompeur. Ces brevets ne concernent aucun programme en particulier. En fait, chaque brevet décrit une idée applicable en pratique, et affirme que quiconque utilise cette idée peut être poursuivi en justice. Il est donc plus clair de les appeler « brevets sur des idées informatiques », ou « brevets sur des algorithmes ».

Le système de brevets américain ne différencie pas les « brevets logiciels » des autres. Seuls les développeurs font la distinction entre les brevets qui nous menacent - ceux qui concernent des idées pouvant être implémentées dans des logiciels - et les autres. Par exemple : si l'idée brevetée est la forme d'une structure physique ou une réaction chimique, aucun programme ne peut implémenter cette idée ; ce brevet ne menace pas le secteur du logiciel. Si par contre l'idée qui est brevetée est un algorithme, alors le canon de ce brevet est braqué sur les développeurs et les utilisateurs.

Cela ne veut pas dire que les brevets couvrant des algorithmes concernent seulement les logiciels. Ces idées peuvent être aussi implémentées dans du matériel... et beaucoup d'entre elles l'ont été. Chaque brevet couvre typiquement les implémentations matérielles *et* logicielles de l'idée.

Le problème particulier du logiciel

Toujours est-il que c'est dans le domaine du logiciel que les brevets sur des algorithmes posent un problème particulier. Il est facile de combiner des milliers d'idées dans un seul programme. Si 10% de ces idées sont brevetées, cela signifie que des centaines de brevets le menacent.

Quand Dan Ravicher, de la Public Patent Foundation (Fondation publique des brevets) a étudié en 2004 un programme de taille importante (Linux, qui est le noyau du système d'exploitation GNU/Linux), il a trouvé 283 brevets américains qui semblaient couvrir des algorithmes implémentés dans son code source. Cette année-là, on estimait la part de Linux dans le système GNU/Linux complet à 0,25%. En multipliant 300 par 400, on peut estimer que le nombre de brevets qui menacent le système dans son ensemble est de l'ordre de 100 000.

Si la moitié de ces brevets était supprimée pour cause de « mauvaise qualité » -

c'est-à-dire pour cause de ratés du système de brevets – cela ne changerait pas grand chose. Que ce soit 100 000 ou 50 000 brevets, la catastrophe est la même. C'est pourquoi c'est une erreur de limiter nos critiques des brevets logiciels aux seuls *patent trolls* ou aux brevets de « mauvaise qualité ». En ce sens Apple, qui n'est pas un « troll » selon la définition habituelle, est actuellement l'entreprise la plus dangereuse quand elle se sert de ses brevets pour attaquer les autres. Je ne sais pas si les brevets d'Apple sont de « bonne qualité », mais plus la « qualité » du brevet est élevée, plus la menace est grande.

Nous devons corriger l'ensemble du problème, pas seulement une partie.

Pour corriger le problème sur le plan législatif, on suggère habituellement de changer les critères d'octroi des brevets – par exemple, d'interdire la délivrance de brevets sur les pratiques algorithmiques et les systèmes nécessaires à leur mise en œuvre. Mais cette approche a deux inconvénients.

Premièrement, les avocats reformulent les brevets de manière astucieuse pour qu'ils correspondent à toute règle applicable ; ils transforment toute tentative de limiter un brevet sur le fond en une simple exigence de forme. Par exemple, de nombreux brevets américains sur des algorithmes décrivent un système qui comprend une unité de traitement arithmétique, un séquenceur d'instruction, une mémoire ainsi que des contrôles pour mener à bien un calcul précis. C'est une manière assez particulière de décrire un programme tournant sur un ordinateur pour effectuer un certain calcul ; elle a été élaborée pour que la demande de brevet se conforme aux critères que, pendant quelques temps, l'on a cru être ceux du système américain de brevets.

Deuxièmement, les États-Unis ont déjà plusieurs milliers de brevets sur des algorithmes, et changer les critères pour empêcher d'en créer d'autres ne permettrait pas de se débarrasser de ceux qui existent. Il faudrait attendre pratiquement 20 ans avant que le problème ne soit entièrement résolu du fait de l'expiration des brevets. Et abolir les brevets existants par la loi est probablement anticonstitutionnel (de manière assez perverse, la Cour suprême a insisté pour que le Congrès puisse étendre les privilèges privés au détriment des droits du public mais ne puisse pas aller dans l'autre direction).

Une approche différente : limiter l'effet des brevets, pas la

brevetabilité

Ma proposition est de changer l'*effet* des brevets. Il faut inscrire dans la loi que développer, distribuer ou exécuter un programme sur des systèmes informatiques polyvalents ne constitue pas une violation de brevet. Cette approche a plusieurs avantages :

- elle n'impose pas de classer les brevets selon qu'ils sont logiciels ou non ;
- elle apporte aux développeurs ainsi qu'aux utilisateurs une protection contre les brevets sur des algorithmes, existants ou futurs ;
- les avocats spécialistes des brevets ne peuvent plus trouver d'échappatoire en changeant la formulation de leurs demandes.

Cette approche n'invalide pas entièrement les brevets existants sur des algorithmes, parce qu'ils continueront à s'appliquer aux implémentations utilisant du matériel dédié. C'est un avantage dans le sens que cela supprime un argument mettant en question la validité de cette proposition du point de vue législatif. Les États-Unis ont légiféré il y a quelques années afin d'immuniser les chirurgiens contre les procès en contrefaçon de brevet, de sorte que même si des procédures chirurgicales sont brevetées, les chirurgiens sont protégés. Cela fournit un précédent pour ce type de solution.

Les développeurs et les utilisateurs de logiciels ont besoin de protection contre les brevets. Cette proposition est la seule solution législative qui apporte une protection totale à tous. Nous pourrions ensuite retourner à notre monde de concurrence ou de coopération... sans craindre qu'un inconnu ne vienne balayer notre travail.

Voir également : Une réforme des brevets n'est pas suffisante

Crédit photo : OpenSourceWay (Creative Commons By-Sa)