

Les logiciels sont brevetables par nature

Livre blanc de l'Observatoire Economique du Droit (UVSQ)
sous la direction de Philippe Simonnot

Le conseil de l'Union européenne du 7 mars 2005 a fait la proposition suivante:

Les Etats membres seront obligés de faire en sorte que au regard de leur droit national, les inventions mises en œuvre par ordinateur soient considérées comme appartenant à un domaine technologique. Pour être brevetable, une invention mise en œuvre par ordinateur doit être nouvelle, susceptible d'application industrielle et impliquer une activité inventive (c'est-à-dire apporter une contribution à l'état de la technique ne portant pas uniquement sur un objet non brevetable) ;

Conformément à la Convention européenne des brevets, un programme d'ordinateur en tant que tel ne peut constituer une invention brevetable. Ne sont pas brevetables les inventions consistant en des programmes d'ordinateur, qu'ils soient exprimés en code source, en code objet ou sous tout autre forme, qui mettent en œuvre des méthodes pour l'exercice d'activités économique, des méthodes mathématiques ou d'autres méthodes et ne produisent pas d'effets techniques au-delà des interactions physiques normales entre un programme et l'ordinateur, le réseau ou un autre appareil programmable sur lequel celui-ci est exécuté ;

Le Conseil a néanmoins introduit une nouvelle disposition afin de préciser que, dans certaines circonstances et à des conditions très strictes, un brevet peut correspondre à une revendication pour un programme d'ordinateur, seul ou sur support. Le Conseil estime que cette disposition alignerait la directive sur ce qui est actuellement pratique courante, tant à l'OEB que dans les Etats membres.

Depuis plusieurs années, la question de la brevetabilité des logiciels agite les esprits et provoque des discussions passionnées. Comme à chaque fois que la technique ouvre un champ nouveau à l'économie. Pour contribuer au débat en deuxième lecture du Parlement européen sur la directive proposée par le conseil des ministres de l'Union Européenne le 7 mars 2005, l'Observatoire Economique du Droit (Université de Versailles Saint Quentin) publie ce livre blanc : se situant au-delà de la polémique, l'Observatoire établit ce que la logique économique indique sur ce difficile sujet.

Les conclusions sont les suivantes.

- 1) Le raisonnement qui est tenu par les différentes parties au débat est le plus souvent conséquentialiste. Il cherche à prévoir les conséquences de la brevetabilité ou de la non brevetabilité des logiciels sur les innovations, leur rythme, leur nature, et par conséquent aussi sur la croissance économique, non seulement au niveau des pays mais aussi au niveau mondial, avec une

- interrogation sur les effets pour les pays en voie de développement. Le constat de l'Observatoire Economique du Droit est qu'aucune conclusion objective pour ou contre la brevetabilité des logiciels ne peut être tirée d'une telle analyse .
- 2) En fait, le débat actuel sur la brevetabilité des logiciels retrouve les termes classiques du débat sur le brevet lui-même depuis qu'il a été institué à la fin du 18^e siècle. Le brevet a tout de suite été considéré comme un compromis fragile entre la nécessité d'encourager l'innovation et la crainte de constituer des monopoles indus. La théorie économique classique confirme l'impasse conséquentialiste dans laquelle se trouve une telle conception du brevet.
 - 3) Les textes fondateurs européens (convention de Munich) concernant la brevetabilité des logiciels reflètent cet embarras, auquel s'ajoute la confusion née de l'assimilation du logiciel au domaine des idées. D'où la fameuse formule : le programme d'ordinateur «en tant que tel » n'est pas brevetable. Formule qui a été reprise par le conseil de l'Union européenne du 7 mars 2005.
 - 4) La pratique s'est écartée des textes : des dizaines de milliers de logiciels ont été brevetés en Europe comme dans d'autres parties du monde. C'est dire que les textes n'étaient pas conformes à la nature même du logiciel.
 - 5) Car, par nature, le logiciel est brevetable, ce que fait apparaître la théorie économique de la propriété. Aussi bien le Conseil du 7 mars 2005 a-t-il bien fait d'ajouter une nouvelle disposition aux textes fondateurs.

1) Le raisonnement qui est tenu par les différentes parties au débat est le plus souvent conséquentialiste. Il cherche à prévoir les conséquences de la brevetabilité ou de la non brevetabilité des logiciels sur les innovations, leur rythme, leur nature, et par conséquent aussi sur la croissance économique, non seulement au niveau des pays mais aussi au niveau mondial, avec une interrogation sur les effets sur les pays en voie de développement. Le constat du livre blanc est qu'aucune conclusion objective ne peut être tirée d'une telle analyse pour ou contre la brevetabilité des logiciels.

1.a. Neutralité axiologique.

Avant de se lancer dans l'analyse des conséquences, il est important de préciser ce point : l'Observatoire Economique du Droit s'efforce d'être axiologiquement neutre. Il n'entend pas prendre position sur les fins que poursuivent les gouvernements, les sociétés ou les individus. Dans le sujet qui nous occupe ici, l'argumentation porte

principalement sur les effets des brevets en matière d'innovation. Nous devons affirmer que l'innovation ne présente pour nous une valeur ni positive ni négative. Pour toute une littérature, du reste, l'invention est connotée péjorativement. Au 12^e siècle, par exemple, le terme invention désigne tout aussi bien une trouvaille, une merveille qu'une ruse ou un mensonge. Jeanne d'Arc, faut-il le rappeler, était désignée comme « menteuse, pernicieuse, inventeresse de révélations et apparitions ».

Beaucoup d'économistes qui se sont penchés sur le problème du brevet et des logiciels ne manquent pas eux aussi d'afficher leur neutralité sur le plan des valeurs. Mais chassées par la porte, les valeurs reviennent par la fenêtre d'autant plus habilement et subrepticement. Ainsi, Jean Tirole, l'un des plus grands économistes français contemporains, dans le rapport au Conseil d'analyse économique du premier ministre français sur la propriété intellectuelle¹, écrit en toutes lettres : « Le rapport évitera l'utilisation de notions floues telles que l'éthique et la morale pour ne s'intéresser qu'à l'efficacité et la distribution, notions plus concrètes ... Concrètement, la raison pour laquelle des formules abstraites comme celle de la théorie de la relativité doivent être exclues du champ du brevetable n'a rien à voir avec le fait que ce sont des formules sur les « lois de la nature », mais est liée aux conséquences que de tels brevets auraient sur les utilisateurs de ces lois (ainsi qu'au fait que ces recherches fondamentales sont le plus souvent financées par la finance publique). La brevetabilité ... doit être analysée en termes d'efficacité ... et d'impact redistributif, et non déterminée par un tabou quelconque quant au champ du brevetable. Toute protection de la propriété intellectuelle, quelle qu'elle soit, implique la privatisation d'une connaissance faisant partie du patrimoine commun de l'humanité. Elle est donc fondamentalement pragmatique et doit être basée sur une analyse des coûts et bénéfices sociaux ». Si l'efficacité et la distribution sont peut-être des notions plus « concrètes » que l'éthique et la morale, elles n'en sont pas plus précises : quelle est la mesure de cette efficacité ? au profit de qui doit se faire la (re)distribution ? De plus, un tabou réapparaît bien vite, puisque dans la même page Jean Tirole écrit : « Certaines innovations (disons, relatives à la fabrication de drogues ou à certaines formes de manipulations génétiques) peuvent être jugées néfastes à un moment donné par la société ». A un moment donné... ! On est en plein relativisme. Mais pourquoi, par exemple, les innovations en matière d'armement, qui sont aussi dangereuses que celles concernant les drogues ou les gènes, seraient-elles brevetables ?

La doctrine qui est ici exposée par Jean Tirole, représentant de toute une génération d'économistes, est explicitement conséquentialiste. C'est dire qu'elle juge de la brevetabilité en fonction

¹ *Propriété intellectuelle*, Rapports Jean Tirole, Claude Henry, Michel Trommetter et Laurence Tubiana, Bernard Caillaux, Commentaires, Daniel Cohen, Lionel Fontagné.

des conséquences en termes d'efficacité et de redistribution, et ce au plan national comme au plan mondial. Tirole le dit lui-même : « On peut à partir d'un raisonnement étayé estimer que la propriété privée et non contrôlée d'une connaissance n'est pas souhaitable soit parce qu'elle est inefficace (elle empêche par trop la diffusion de la connaissance ou bloque le processus d'innovations futures), soit parce qu'elle a des conséquences redistributives néfastes (vis-à-vis des plus démunis ou des pays en voie de développement) » Si les mots ont un sens, on devrait déduire de ces dernières lignes qu'une propriété « privée et non contrôlée » (que vaut une propriété privée si elle est contrôlée ?) pourrait être arrachée à son propriétaire s'il est estimé (par quel tribunal ?) que par son existence même, elle entrave l'innovation et la redistribution. Il ne s'agit pas d'une vue de l'esprit. Des licences obligatoires sont parfois imposées aux détenteurs de brevets, au mépris du droit de propriété - à noter que cette disposition existe en France depuis 1968, mais qu'elle n'a pratiquement pas été utilisée.

D'entrée de jeu, on peut donc constater que la position adoptée par l'éminent économiste français est tout à fait éthique et morale, avec en plus des tabous non avoués ou non assumés. En outre elle est stratégique : Qui ne voit que celui qui s'y opposerait passerait immédiatement pour un ennemi du genre humain et un exploiteur des prolétaires, voire des « nations prolétaires », comme on a parfois appelé les pays en voie de développement ? On a insisté sur le texte de Jean Tirole car il est symptomatique d'une certaine pensée dominante dans ce domaine, que l'on n'aurait pas de difficulté à retrouver dans les propos de maints hommes politiques intéressés par notre sujet.

1.b. Récompenser l'innovation.

Dans cette perspective, le problème est relativement simple : la seule justification du brevet est de récompenser l'innovation. Mais, à partir du moment où la récompense est accordée par la société, les choses se compliquent immédiatement. Car la « société » va avoir son mot à dire pour définir quelles inventions doivent être encouragées et dans quelle mesure elles doivent l'être ? Il va donc falloir définir ce qu'est une invention brevetable (selon les termes de l'ADPIC, on le sait, il faut « qu'elle soit nouvelle, qu'elle implique une activité inventive, et qu'elle soit susceptible d'application industrielle ») et combien on est prêt à la payer. La politique du brevet doit trouver un juste milieu entre une diffusion insuffisante de l'innovation à cause du pouvoir de monopole que l'on accorde à l'inventeur et une innovation insuffisante due à de trop faibles perspectives de profit pour l'inventeur.

Rappelons que le logiciel est ensemble de programmes qui interagissent. Chaque programme est une suite d'instructions écrites dans un langage bien défini, qui codent des opérations, des procédés ou des algorithmes s'appliquant à des données au sens large. Le programme est d'abord écrit dans un langage de programmation

compréhensible par l'homme de métier. Il se présente alors sous la forme d'un « code source », qui permet le travail d'élaboration et de production du produit logiciel. Une fois écrite, cette suite d'instructions peut être compilée, c'est-à-dire traduite et transformée en une suite binaire (de 1 et de 0) pour la présence ou non d'une impulsion électrique, compréhensible par une machine mais désormais opaque pour un esprit humain : il est alors sous forme d'un « code objet » ou code binaire.

L'invention logicielle va entrer dans la problématique générale de la brevetabilité : encourager l'inventeur mais pas trop. Ni trop ni trop peu : il s'agit de trouver le *fine tuning*.

Deux positions ici se font face à l'aune du critère de l'innovation. Aux partisans du brevet logiciel s'opposent les militants de la cause du logiciel libre, basé sur la libre communication et utilisation des codes sources. Au contraire, les tenants du logiciel propriétaire ne divulguent pas leur code source. Pour les tenants du *free software*, le brevet logiciel n'est pas indispensable à l'innovation, il peut même lui être un handicap. Pour les avocats du logiciel propriétaire, la position est symétriquement inverse : pas d'innovation logicielle sans brevet.

Les positions sont souvent tranchées. Ainsi une instance aussi respectable en France que le Conseil général des mines n'hésite pas à proclamer que le brevet logiciel aurait un impact négatif sur l'innovation et la concurrence. Le Comité de coordination des sciences et des techniques de l'information et de la communication (CCSTIC) – un département du CNRS – est du même avis².

1.c. *Le copyright suffit-il ?*

Le brevet n'est pas indispensable pour les partisans du logiciel libre, car la protection accordée par le *copyright* suffit. Soit dit en passant, aucun des deux camps ne remet en cause le *copyright* et nous expliquerons pourquoi. Le créateur d'un logiciel libre ne renonce pas à ses droits d'auteur sur le logiciel qu'il a écrit. Il faut rappeler que les droits exclusifs dont jouit le titulaire des droits d'auteur sont beaucoup plus larges que ceux accordés au propriétaire d'un brevet. Le titulaire des droits d'auteur peut s'opposer à toute exploitation de son logiciel, quelle qu'elle soit, y compris son exploitation à des fins de recherche, alors que le propriétaire du brevet ne peut que s'opposer à son exploitation à des fins commerciales. La protection du brevet est beaucoup plus courte (20 ans) et limitée aux pays pour lesquels un brevet existe alors que le droit d'auteur est universel et dure 70 ans. Mais encore une fois, le droit d'auteur protège le texte, pas l'invention, l'expression, mais non la fonctionnalité, puisque les fonctions peuvent être codées de multiples façons.

Sans doute, la copie d'un code binaire est très facile et son coût est infime. Il n'est pas nécessaire pour l'imitateur de repasser par les mêmes étapes d'organisation du programme, d'écriture et de

² *Le Monde* du 17 octobre 2001

compilation, ni d'avoir accès au code source ou à la description de l'ensemble des opérations effectuées par le logiciel. Mais le *copyright* est censé dissuader les copieurs d'accomplir leur forfait. Ou bien on admet par avance un certain nombre de copies illégales, de la même façon qu'un hyper-marché intègre dans ses coûts un certain pourcentage de vol à l'étalage. On peut même considérer que de très nombreux logiciels ne seraient tout simplement pas vendus aussi bien s'ils n'étaient pas copiables ! Des solutions techniques existent pour empêcher les copies trop faciles. Si elles ne sont pas utilisées, c'est qu'à la limite un logiciel protégé a moins de valeur qu'un logiciel librement copiable à cause des « effets de réseau » qu'il génère, effets que nous examinerons ci-après. Du reste, à un prix suffisamment faible, il est plus intéressant d'acheter que de copier.

Il n'en va pas ainsi pour les partisans du brevet logiciel. Car l'inventeur court un autre risque que la copie servile, à savoir que soit développée par un autre programmeur une innovation proche de l'original sans violer le *copyright*. Ce qui est volé ici c'est non pas le texte, mais l'invention elle-même, avec une autre écriture, un autre code. Si l'invention originelle n'est pas protégée par un brevet, il va en résulter une concurrence destructrice de profits et décourageante pour les inventeurs. Un brevet aura justement pour mission de définir plus ou moins explicitement un périmètre de « proximité » en deçà duquel une innovation concurrente est réputée enfreindre la protection.

1.d. *Volontariat.*

Pas du tout rétorquent les anti-brevets. Les programmeurs passionnés d'informatique, les chercheurs, les universitaires peuvent très bien se passer de récompenses. L'honneur, la reconnaissance de leurs pairs, les perspectives de carrière suffisent à les motiver. Les anti-brevet tirent argument de ce que l'industrie du logiciel est essentiellement une industrie de main d'œuvre, nécessitant peu de capital. Mieux encore : on estime que 60 à 80% des coûts globaux de R&D pour un logiciel sophistiqué résident dans la mise au point et le debuggage du programme, et la réécriture progressive du code. Cette activité de développement serait efficacement effectuée par des réseaux de programmeurs, qui apportent des visions critiques et indépendantes dans le cadre d'un projet. Cette méthode d'approche constitue la base de la dite « communauté du logiciel libre » : le logiciel est distribué avec son code source, ce qui permet à des développeurs de repérer et de corriger les dysfonctionnements d'un logiciel en cours d'élaboration. Les perspectives de carrière du programmeur évoquées plus haut sont favorisées, puisque ses capacités sont évaluées publiquement grâce à la divulgation du code source et du travail qui est fait dessus. La « communauté du logiciel libre » prétend être particulièrement efficace pour la mise au point de logiciels sophistiqués. L'ouverture des codes sources permettrait leur examen par de nombreux experts indépendants, la détection plus efficace des erreurs, et la correction plus rapide de ces

erreurs. La « communauté du logiciel libre » a un aspect viral ou contaminant : toute personne qui redistribue le logiciel, avec ou sans modification, doit aussi transmettre la possibilité de le copier ou de le transformer, sans jamais pour autant cesser de respecter le *copyright*.

Comment peut-on croire qu'une activité qui génère chaque année des milliards de dollars de chiffres d'affaire puisse reposer sur du volontariat ? demandent les partisans du brevet logiciel. De toutes façons, remarque-t-il, si le client ne paie pas les redevances d'un brevet, il paie les services et il peut en résulter un Total Cost of Ownership, comme on dit dans la littérature américaine, beaucoup plus élevé. Rien n'est gratuit en ce bas monde, et le logiciel libre est peut-être *free*, mais il n'est pas gratuit, ne se confond pas avec un *freeware*. Un logiciel libre est un logiciel protégé par le droit d'auteur comme on vient de le dire, soumis à un contrat de licence particulier (en général GNU GPL). Les entreprises qui le produisent attachent à sa distribution, par des pratiques d'achats liés, la fourniture de services personnalisés de formation, de support, de maintenance, d'extension ou d'adaptation à des clients spécifiques.

Quant à cet argent qui afflue dans l'industrie du logiciel, il faut bien qu'il aille quelque part. De deux choses l'une : ou bien le programmeur travaille pour une société, et alors toute la plus-value due à son travail inventif est captée par son employeur, ou bien il travaille à son compte, et c'est alors le client qui s'approprie l'aubaine éventuelle. Or, ce client peut être tout aussi bien un autre programmeur, une société, le secteur public, un gouvernement étranger. Il n'est pas sûr que les militants du logiciel libre soient toujours conscients qu'ils travaillent pour le roi de Prusse, comme on disait autrefois, qui pourrait être aujourd'hui le gouvernement d'une dictature. A la limite, la communauté du logiciel libre pourrait fonctionner indéfiniment si tous les citoyens du monde étaient programmeurs et également inventifs. Comme ce n'est pas le cas, il y a des fuites dans le système, et les non programmeurs ou les programmeurs non inventifs profitent des programmeurs inventifs, et à long terme, le système n'est pas « soutenable ». Si les partisans du logiciel libre font tant de pression sur les gouvernements pour qu'ils adoptent des législations qui leur soient favorables, c'est peut-être qu'ils ont le pressentiment que sans aide gouvernementale au moins sur le plan législatif, mais aussi si possible, au niveau des commandes publiques, la communauté est condamnée à dépérir³. De fait, la plupart des projets de logiciel libre proviendraient plus ou moins directement du clonage de logiciels propriétaires. Par conséquent, le logiciel libre ne devrait pas devenir un mode dominant sous peine de voir l'innovation propriétaire se tarir. En vérité, il y a une sorte d'asymétrie entre ces deux modes opposés : par définition, le code source des logiciel libre est plus facilement accessible que celui

³ DeLong James V., *The Enigma of Open Source software*, The Progress Freedom Foundation, , March 2004.

des logiciels propriétaires, et de même est plus aisée la détection d'une violation de protection.

1.e. *Le titulaire le plus efficient.*

Pour ses partisans, le brevet a un autre mérite, c'est de permettre de transférer la situation de monopole qu'il confère à un agent plus efficace. Or c'est bien l'une des vertus des lois du marché, si bien définies par Ronald Coase, prix Nobel d'économie, dans son célèbre théorème : le titulaire final d'un titre de propriétaire, quel qu'il soit et quelle que soit la situation de départ, est celui qui peut le valoriser le mieux⁴.

1.f. *La question du contournement.*

De plus, l'existence même du brevet permet d'agir sur les anticipations des acteurs. Un entrant potentiel peut être totalement dissuadé de développer une imitation ou une innovation trop proche techniquement et qui serait invalidée par un brevet existant. Le brevet oblige donc les candidats à l'entrée sur le marché à développer de vraies innovations – encore faut-il que l'on puisse définir ce que sont de « vraies innovations ». A quoi, les anti-brevet répondent que ces stratégies de « contournement » peuvent être inutilement coûteuses pour la société. Dans les industries classiques, l'effort de contournement peut constituer une source fondamentale de progrès, obligeant à de nouvelles inventions. Philippe Aigrain, lorsqu'il était chef de secteur pour les Technologies du logiciel, à la Direction Générale Société de l'information de la Commission Européenne, avant de diriger Sopinspace, soutenait que dans l'industrie du logiciel l'effort de contournement ne menait nulle part. Beaucoup d'argent était dépensé dans l'espoir de redevances qui n'étaient jamais à la hauteur. Le système des brevets logiciels était pour lui un jeu perdant. On finançait de manière répétitive les mêmes travaux.

1.g. *L'effet « réseau ».*

L'« effet réseau » sert aux anti-brevets à renforcer leur argumentation. Sous ce nom barbare se cache un phénomène qui n'est pas nouveau, mais prend une extension gigantesque en informatique, à savoir que la valeur qu'un consommateur accorde à un bien dépend du nombre d'utilisateurs de ce bien. De même qu'une voie ferrée isolée ne peut fonctionner lorsqu'elle n'est pas connectée au réseau, de même ceux dont les ordinateurs ou les logiciels ne sont plus compatibles avec la majorité des autres utilisateurs voient leur investissement dévalorisé. Les externalités de réseau sont bien connues des industries du transport et des communications où la concurrence entre les entreprises se fait par l'extension des réseaux et où la valeur d'un réseau s'accroît considérablement s'il peut se connecter à d'autres réseaux. Mais dans ces

⁴ Rappelons que Ronald Coase, a obtenu le Nobel en 1991 pour seulement deux articles, l'un, *La nature de la firme*, écrit en 1937, à l'origine même du concept des coûts de transaction, de l'autre *Le problème du coût social*, publié en 1960.

domaines, le réseau a vocation à s'étendre à l'ensemble de la planète. L'effet réseau n'est autre que l'économie d'échelle appliquée à la demande.

Dans la grande industrie classique, les économies d'échelle liées à la production sont tôt ou tard contrebalancées par des « déséconomies d'échelle » - autrement dit des économies d'échelle négatives causées par les difficultés qu'implique la gestion d'énormes organisations. Ainsi explique-t-on qu'aucun géant de l'automobile n'ait jamais conquis l'ensemble du marché mondial. Grâce au progrès de la gestion informatique ces limites peuvent être repoussées, mais non éliminées. Dans l'économie de l'information les économies d'échelle liées à la production sont relayées par les économies d'échelle liées à la demande (l'effet de réseau), lesquelles ne s'arrêtent jamais de croître puisqu'elles augmentent avec la taille du marché. Si tout le monde utilise Microsoft Word, c'est une raison de plus de l'utiliser pour les nouveaux venus sur le marché.

L'effet de réseau a même été mis en équation sous le nom de loi Metcalfe, du nom de son inventeur : la valeur d'un réseau augmente en raison du carré du nombre d'utilisateurs. Si par exemple la valeur d'un réseau pour chaque utilisateur est égale à 1 E, et qu'il y a 1 million d'utilisateurs, la valeur totale du réseau est de 1 million de E. S'il y a dix millions d'utilisateurs, la valeur totale du réseau tend vers 100 millions de E. Et ainsi de suite. Cette loi qui, évidemment, n'a rien de scientifique, même si elle est observable sur certains segments de certains marchés d'information, n'en donne pas moins une image parlante de l'effet réseau.

L'« effet réseau » protégerait mieux les inventeurs que des procédures juridiques lourdes, opaques, et terriblement coûteuses. A lui tout seul, l'effet réseau aboutirait à la concentration sur quelques logiciels. Un logiciel peut ainsi monopoliser certain segment de marché uniquement parce qu'il a été introduit en premier, et cette monopolisation peut être très rapide. Si c'est le nombre d'utilisateurs qui fait la valeur d'un logiciel standard, le fait de récompenser par un brevet un objet dont la valeur ne provient finalement que des usagers, et non pas de son action propre, ne présente aucun intérêt économique. Cela revient à affirmer que le développement technique relèverait de la loterie.

La concurrence sur ce type de marché prend alors une forme extrême, le *tipping* dans la littérature américaine (la *tipping line* ressemble à une ligne de partage des eaux : on est sur un versant ou sur un autre). Le logiciel qui est entré le premier est tarifé à un prix très faible de manière à empêcher l'arrivée de nouveaux entrants, et grâce aux effets de réseau il occupe l'essentiel du marché. Des rentes de monopole sont ainsi assurées. Dans un tel contexte, un logiciel concurrent tente d'entrer massivement sur le marché en tant que logiciel libre : ainsi Linux face à MS Windows, et Netscape face à MS Explorer essaient d'imposer un nouveau standard. Ici, le logiciel libre s'avoue comme élément d'une stratégie commerciale analogue au dumping. Ce qui apporte de l'eau au moulin des partisans du brevet logiciel, qui arguent qu'il n'y a pas de monopole dans la nature, que

tout monopole existant s'appuie d'une manière ou d'une autre sur l'Etat (et c'est bien le cas du brevet), que ces positions dites dominantes sont fragiles et peuvent être renversées du jour au lendemain si elles ne sont pas protégées par des brevets. Un tel basculement serait « socialement coûteux », car la victoire du nouveau logiciel serait due non pas à ses qualités intrinsèques, mais simplement parce qu'il a su mieux utiliser que son concurrent l'effet de réseau en jouant sur les anticipations des agents, la mode, etc. Là encore on entre sur une terre bien peu ferme où un « juge de paix » (lequel et selon quels critères) pourrait déterminer à l'avance quelles sont les innovations « socialement coûteuses » et celles qui sont « socialement bénéfiques ».

1.h. *Durée du brevet et technique.*

Quant à la durée du brevet (20 ans) elle ne serait pas en concordance avec l'évolution technique qui ramène la durée effective d'un logiciel à quelques années. La protection qu'apporte le brevet serait donc illusoire. Une autre pierre dans le jardin des partisans du brevet logiciel. Lesquels ne manquent pas de répondre : si cette protection est tellement illusoire, pourquoi vous en formalisez-vous ? En fait, la durée moyenne de vie d'un brevet ne dépasse pas sept à huit. Au-delà de ce délai, rares sont les titulaires d'un brevet qui continuent à payer à l'Etat les annuités qui permettent de garder le brevet en vigueur (en France, 300 e par an et par Etat couvert), tout simplement parce que cela n'en vaut plus la peine.

1.i. *Périmètre et clôture.*

Les tenants du logiciel libre mettent alors en avant le fait que toute innovation logicielle n'est qu'un maillon dans une chaîne, et que le programmeur qui invente un nouveau logiciel pénètre en toute bonne foi et sans s'en rendre compte sur le périmètre plusieurs innovations existantes. Comme un promeneur s'aventure sur le terrain mal délimité d'un voisin. L'argument est à double tranchant, car il peut être utilisé pour justifier une définition plus précise des droits de propriété et leur affichage sur la « clôture » du périmètre de l'invention brevetée. A partir de quoi les partisans du logiciel libre pourraient remarquer que si à chaque fois qu'un programmeur pénètre sur le territoire du « voisin » il doit négocier un accord, on aboutirait très vite, en théorie, à des « coûts de transaction » pharamineux. Mais il faut bien que ces coûts de transaction soient supportés par le client. Autrement dit, l'augmentation des coûts de transaction est freinée par les forces du marché.

1.j. *Le problème des PME.*

Du reste, du point de vue de la concurrence, la protection accordée par un brevet logiciel n'est pas forcément plus monopolistique que celle dont dispose l'entreprise qui a su acquérir une large position avec les seules armes du commerce. Car la « start up », si elle dispose d'une invention, accède à la protection du brevet plus

facilement qu'à la « protection commerciale ». Ainsi prétendre que le brevet protège les positions acquises des grandes entreprises est souvent contraire à la réalité. Au contraire, sans brevet, les « jeunes pousses » qui ne disposent par définition d'aucun réseau commercial, ont toutes les chances de ne pas survivre et de se voir prendre leur éventuelle innovation. Comme le reconnaît Jean Tirole dans le rapport déjà cité, « le brevet n'est pas forcément au désavantage des PME⁵ ». En réalité ce pourrait être tout le contraire. Une PME, qu'elle soit ou non une « start-up » a besoin de s'aménager un espace plus ou moins à l'abri de ses concurrents. Si elle cherche à vendre son logiciel à une grande entreprise, son pouvoir de négociation sera certainement plus élevé si elle dispose d'un brevet en bonne et due forme.

La littérature oppose ici deux types de brevets. Le brevet sous sa forme défensive est destiné à protéger un produit nouveau, alors que le brevet offensif donne à son titulaire le contrôle d'un secteur technique nouveau. Les banques demandent le plus souvent que le produit sur lequel se lance une PME soit protégé par un brevet. Certes, le dépôt, l'entretien et la défense de brevets sont coûteux. Mais on exagère beaucoup ces frais : la préparation et le dépôt d'un brevet coûte environ 6 000 à 10 000 E, auxquels il faut ajouter 300 E pour les frais de procédure ; au bout d'un an, le candidat se voit proposer un « PCT » (Patent Cooperation Treaty), c'est-à-dire un dépôt international pour une couverture mondiale, qui lui coûtera 5 000 E. Ensuite, il peut opter pour une protection de l'Office Européen des Brevets qui le couvrira sur 30 pays. Il lui en coûtera 100 000 E. Très rares ceux qui vont jusque là.

Dans leur dialogue avec les établissements de prêt, le brevet permet aux PME de donner du crédit à leur projet. Les bailleurs de fonds n'ont pas forcément les connaissances nécessaires pour évaluer le bien fondé technique du dossier qu'on leur présente. Le brevet vient donc valider le projet de la PME au moins au point de vue technique. On a tellement dit, en France notamment, que la protection du logiciel relevait uniquement du droit d'auteur que beaucoup de patrons de PME découvrent maintenant que des brevets logiciel existent, et qu'elles n'ont pas su protéger à temps leurs innovations par des dépôts de brevets. Comme on l'a dit plus haut, il est pratiquement impossible de ne pas pénétrer sur des périmètres de brevets existants. Ce risque n'est pas spécifique au domaine du logiciel. Entre les grands groupes règne un « équilibre de la terreur » qui, comme les stratèges le savent bien, est un équilibre fondé sur la raison, et qui se termine souvent par des accords à l'amiable et à des transactions contre espèces sonnantes et trébuchantes. Une PME dépourvue de tout brevet ne peut entrer dans ce jeu de la dissuasion. Autrement dit, loin de fausser la concurrence en concentrant toutes les cartes dans les mêmes mains, le brevet logiciel permet au contraire un jeu moins inégal. Pour filer la métaphore de la

⁵ Jean Tirole, op. cit. p. 21.

stratégie nucléaire jusqu'au bout, on dira que le brevet permet éventuellement une dissuasion du faible au fort. De même qu'il y a eu égalisation par l'atome, il pourrait y avoir une égalisation par le brevet. Du reste, beaucoup de PME l'ont compris, puisque sur 16 000 brevets déposés en France, 13 000 sont d'origine française, 5 000 sont détenus par 100 grandes entreprises et 5000 par des PME, selon les chiffres donnés par Philippe Cadre (Institut National de la Propriété Industrielle) lors d'un débat sur la protection des logiciels le 19 mars 2002.

En aucun cas, donc, contrairement à ce qui est souvent proclamé sur la place publique, le brevet logiciel n'est l'apanage des grandes entreprises. En revanche – mais c'est moins connu - de très grandes entreprises de l'informatique utilisent le logiciel libre comme arme commerciale selon la stratégie indiquée plus haut.

Cela n'empêche pas les partisans du logiciel libre d'utiliser l'argument PME. Quand nous développons une application innovante, disent-ils, 1% vient de nous et 99% du pot commun de ceux qui nous ont précédés. Si le pot commun cesse d'être commun, que l'on permet son appropriation par des brevets, il n'y a simplement plus d'innovation possible sauf pour les grands acteurs capables de passer des accords de licence croisée. C'est oublier que le « pot commun » du passé restera commun, puisqu'on ne brevète pas le passé.

1.k. *Le paradoxe du logiciel, bien durable.*

Le logiciel est un bien durable. On peut même dire qu'il est un bien durable parfait : il ne s'use pas même quand on s'en sert. Sa seule obsolescence vient de l'évolution technique. Comme Ronald Coase, l'économiste américain déjà cité, l'a montré, le monopole sur un bien durable se fait concurrence à lui-même. En effet, après avoir équipé les usagers les plus pressés d'acquérir un logiciel, le monopole est conduit à baisser ses tarifs de manière à toucher une clientèle moins enthousiaste ; les acheteurs de la première heure peuvent donc retarder leur achat s'ils anticipent cette baisse de tarifs⁶. La liberté d'adapter ses tarifs de manière à toucher une population de plus en plus large se retourne contre le monopole et fait fondre sa rente, prisonnier qu'il est des anticipations baissières des acheteurs. Aucun brevet ne pourra empêcher un tel processus. Soit, répondront les partisans du brevet logiciel, si le brevet ne protège pas vraiment, laissez-nous breveter en paix ! En réalité, le paradoxe de Coase ne s'applique pas aussi nettement qu'on pourrait le croire au cas des logiciels, car, comme le fait remarquer l'économiste français Bernard Caillaux dans sa

⁶ v Coase R, (1972) « Durability and Monopoly », *Journal of Law and Economics*, n° 15, p. 143-149, qui n'est plus une conjecture : Bulow J. (1982), « Durable-Goods Monopolists », *Journal of Political Economy*, n° 90, p. 314-332, Fudenberg D., Levine D. et Tirole J. : « Infinite Horizon Models of Bargaining with One-Sided Incomplete Information », in *Game Theoretic Models of Bargaining*, Roth (ed.), Cambridge University Press, Cambridge, p. 73-98 ; Gul F., Sonnenschein H. et Wilson R. (1986) : « Foundation of Dynamic Monopoly and the Coase Conjecture », *Journal of Economic Theory*, n° 39, p. 55-190.

participation au rapport déjà cité au premier ministre français, « les entreprises de logiciels peuvent apporter des améliorations marginales à leurs produits, une plus grande fiabilité et une plus grande convivialité, qui permettent de vendre régulièrement des produits dérivés (*upgrading*) ». Peuvent aussi être adoptées des stratégies d'obsolescence programmée⁷.

1.l. L'incidence sur la concentration.

Le jeu se complique du fait que le choix à la disposition des inventeurs de logiciel n'est pas seulement entre breveter et ne pas breveter. Ils peuvent très bien recourir au secret commercial. Si la brevetabilité des logiciels était entravée, c'est un tel choix qui s'imposerait à la plupart. Même avec la prolifération des brevets logiciels, le choix du secret est celui qui s'impose encore le plus pour des raisons que nous expliquerons. Du point de vue de la concentration dans l'industrie des logiciels, il a été soutenu que le secret comme le brevet y conduisent. Donc, la brevetabilité n'ajouterait rien dans ce domaine. Au contraire, comme on l'a dit plus haut, on peut s'attendre à une déconcentration du fait de l'utilisation que les PME peuvent faire du brevet. Précisons que l'Observatoire Economique du Droit n'accorde aucune valeur *a priori* ni à la concentration ni à la déconcentration (encore une fois, il n'y a pas de monopole dans la nature, et même une entreprise seule à produire tel bien ou tel service serait en situation de concurrence, car ce bien ou ce service seraient en concurrence avec tous les autres).

1.m. L'incidence sur la divulgation des connaissances.

Par rapport au secret commercial, le brevet et le logiciel libre favorisent la divulgation des connaissances. Si cette divulgation est l'objectif recherché (et cet objectif repose sur un certain système de valeurs), reste à savoir laquelle des deux branches de l'alternative est la meilleure. Si la brevetabilité encourage ceux qui avaient recours au secret commercial à prendre des brevets, alors l'objectif sera atteint puisque de toutes façons ils n'auraient pas eu recours au logiciel libre. Il est vrai que même en cas de brevet une grande partie de l'innovation incorporée dans le code source peut rester inaccessible. Or, l'information qui est la plus utile se trouve dans le code source. C'est l'accès à ce code qui permet d'améliorer en permanence le logiciel. Même les partisans du logiciel libre hésitent à publier le code source. A partir du moment où aucun programme complexe ne peut être écrit sans enfreindre des brevets, on essaie de garder secret le code source. L'argument se retourne dans l'autre sens. Si l'on ne connaît pas le code source d'un logiciel libre, alors ce n'est plus un logiciel vraiment libre.

1.n. Juridicisation.

L'un des grands reproches que l'on fait aux brevets logiciels, c'est la juridicisation du secteur, l'expérience des Etats-Unis étant

⁷ Bernard Caillaux, op. cit. p. 144.

montrée comme l'exemple à ne pas suivre. Il est vrai que des entreprises se constituent que l'on peut considérer comme de véritables entreprises juridiques. Leur fonction est d'acheter et de gérer des brevets dans le but de rechercher systématiquement des occasions de conflits avec des détenteurs de logiciels, qu'ils soient libres ou brevetés. Mais on ne voit pas pourquoi de telles entreprises favoriseraient systématiquement les grands groupes. Les PME peuvent tout aussi bien profiter de cette dérive juridique, voire adopter des stratégies offensives. Quant aux coûts supposés très élevés de cette juridicisation, on peut leur appliquer la même remarque qu'aux coûts de transaction, dont, du reste, ils font partie. Ils sont régulés par le marché.

1.o. *L'intérêt du grand public.*

Si l'on se place maintenant du point de vue du grand public, on constate que ce dernier est mieux servi par le logiciel propriétaire que par le logiciel libre, qui est trop sophistiqué et demande trop de service après vente, toujours fort coûteux. La brevetabilité pourrait donc être un moyen de réorienter la R&D vers des innovations « socialement plus souhaitables », si cette expression a un sens.

1.p. *L'intérêt de l'Europe.*

Il est parfois soutenu que la R&D en logiciel se déplace vers les Etats-Unis et ralentit en Europe, et que pour contrer ce mouvement il faudrait que l'Union Européenne emboîte le pas aux Etats-Unis en matière de brevets logiciels. Ce dernier point de vue est contesté par le raisonnement suivant : Si le brevet du logiciel se généralise, alors il y aurait en Europe un coup d'arrêt au développement du logiciel libre, car l'insécurité juridique deviendrait trop grande pour lui. En revanche, si l'Union Européenne reste sur ses positions restrictives, alors les brevets déposés aux Etats-Unis seraient sans effet en Europe, l'innovation des logiciels y serait facilitée et plus rapide, et le Vieux Monde aurait une nouvelle chance dans la bataille du logiciel. C'est oublier que le marché le plus important se trouve aux Etats-Unis et que c'est sur ce marché qu'une invention logicielle européenne doit être validée d'une manière ou d'une autre.

1.q. *Conclusion.*

La conclusion de Bernard Caillaux, déjà cité, est intéressante, car elle confirme que l'analyse conséquentialiste ne peut aboutir à aucun résultat. « En l'absence d'une analyse empirique de l'amplitude de tous ces effets, écrit-il, il est difficile d'adopter une position tranchée sur l'extension de la brevetabilité des logiciels. Les innovations majeures ont vu le jour avant que la protection par brevet ne soit adoptée, voire même envisagée, pour ces inventions...les études empiriques sont encore insuffisantes...Les mesures d'innovations sont contestables ».

Et d'ajouter : « A notre connaissance, il n'est à ce jour aucune étude économétrique capable de répondre sans ambiguïté à la question de la brevetabilité »⁸.

L'analyse conséquentialiste conduit à une impasse.

2) En fait, le débat actuel sur la brevetabilité des logiciels retrouve les termes classiques du débat sur le brevet lui-même depuis qu'il a été institué à la fin du 18^e siècle. Le brevet a tout de suite été considéré comme un compromis fragile entre la nécessité d'encourager l'innovation et la crainte de constituer des monopoles indus. La théorie économique classique confirme l'impasse conséquentialiste dans la quelle se trouve une telle conception du brevet.

2.a. Privilège.

Historiquement, le brevet est un privilège accordé par la puissance régaliennne dans l'intérêt général. Pour des économistes qui croient aux vertus du libre échange dans tous les domaines, cette intervention de la puissance publique au nom de l'intérêt général ne peut qu'être gênante.

Aussi bien, à un Lincoln qui énonçait, « The Patent System added the fuel of interest to the fire of genius », Michel Chevalier lui répondait de ce côté-ci de l'Atlantique que le brevet était « un outrage à la liberté et à l'industrie ». La solution proposée par les libéraux était simple : le succès industriel est le garant de la validité d'une invention et non sa protection par l'Etat. Un autre auteur de la même époque, F. Malapert, écrivait dans des termes que pourrait reprendre la communauté des logiciel libres dans ses manifestes : « Il résulte de l'étude des faits que rien ne justifie l'énorme privilège qui permet à un homme d'empêcher les autres de travailler à leur guise ... Il faut savoir que tout monopole est une cause de démoralisation en même temps qu'un obstacle au progrès des arts et de l'industrie, en un mot à la marche de la civilisation ».

2.b. La théorie économique classique.

Si l'on se réfère maintenant à la manière dont la théorie économique classique rend compte du dilemme du brevet, on comprend mieux les impasses dans lesquelles s'est fourvoyé le débat sur le brevet logiciel.

On va présenter ici cette théorie dans les termes les plus simples, pour n'en retenir que ce qui est utile à notre propos.

⁸ Bernard Caillaux, op cit 128.

On suppose que l'industrie du logiciel travaille à coût constant. Sur un système de coordonnées où les quantités sont portées en abscisses et les prix en ordonnées, la courbe d'offre sera horizontale du fait de la constance des coûts. Soit C_1 cette courbe (cf. graphique). Elle va rencontrer la courbe de demande, qui comme toutes les courbes de demande descend vers la droite (les quantités achetées par les consommateurs varient en fonction inverse des prix), en un point E_1 , appelé point d'équilibre. La théorie économique classique indique qu'en ce point le « surplus » du consommateur de logiciel est au maximum (triangle I sur le graphique). C'est-à-dire qu'en ce point, le bien-être de la société considérée est au plus haut possible. Les coûts étant constants, et la concurrence étant supposée fonctionner sans entrave, en ce même point E_1 , les profits du producteur de logiciel sont nuls (on suppose que la rémunération de l'entrepreneur est comprise dans les coûts).

Ce même entrepreneur, qui a investi dans la R&D en matière de logiciel, découvre une invention qui fait descendre le coût des services associés au logiciel. En l'absence de tout brevet, et donc de toute protection pour cette invention, la courbe d'offre va descendre de C_1 en C_2 . Le jeu de l'offre et de la demande, qui, par hypothèse, n'est ni entravé ni faussé, va déplacer le point d'équilibre, lequel descend en E_2 . La quantité vendue de services associés au logiciel sera augmentée alors même que leur prix diminue. Le surplus du consommateur sera augmenté d'autant et donc le bien-être de la société. Mais le profit de l'entrepreneur est toujours aussi nul. Dans ces conditions, l'entrepreneur, qui n'est pas un philanthrope, ne fera pas les investissements de R&D nécessaires à l'invention puisqu'il anticipe qu'il n'en tirera aucun profit, et l'innovation ne verra pas le jour. On en restera au point E_1 , c'est-à-dire à un niveau de bien-être inférieur à celui qu'on aurait pu atteindre si l'innovation avait été faite.

Pour sortir de cette situation, le brevet est institué. Il accorde à l'inventeur une certaine rente pendant un certain temps. Le résultat est le suivant. Pendant la durée du brevet, l'entrepreneur peut garder pour lui l'invention (on suppose qu'il l'exploite lui-même). La courbe d'offre reste donc la même, en C_1 . Mais comme ses coûts sont descendus en C_2 , l'entrepreneur jouit d'un profit par unité vendue égal à la différence entre ces deux coûts. Le profit total s'obtient en multipliant ce profit unitaire par les quantités vendues (rectangle II sur le graphique). S'il cède son brevet à une autre entreprise, le résultat est le même, car le coût de

production des services associés au logiciel en question sera toujours égal à C_1 puisqu'au coût C_2 cette nouvelle entreprise devra ajouter la rente réclamée par l'inventeur.

Pendant toute la durée du brevet, le surplus du consommateur reste au niveau qu'il avait atteint avant l'invention. Dès que l'invention entre dans le domaine public, alors la courbe d'offre descend réellement en C_2 .

Le surplus du consommateur augmente d'autant, avec un profit nul pour toutes les entreprises. Sur le graphique, ce surplus est égal alors à la somme des surfaces I, II, et III. Les consommateurs sont maintenant les seuls bénéficiaires de l'innovation. Le bien-être social a finalement augmenté, ce qui justifie ex post l'institution du brevet.

Le problème est que pendant toute la durée du brevet il y a une perte de bien-être mesurée exactement par le triangle III du graphique. Les connaisseurs de la théorie économique classique reconnaîtront dans ce triangle la perte sèche de bien-être inhérente à toute intervention de l'Etat, quelle se fasse sous la forme d'impôt ou de monopole. On dit perte sèche, parce que tout le monde y perd.

En matière de brevet, cette perte de bien-être provient du fait que pendant toute la durée de la protection, au surplus du consommateur resté au niveau antérieur à l'invention (triangle I) ne vient s'ajouter que la rente du détenteur de brevet (rectangle II). La somme de ses deux satisfactions (si tant est qu'on puisse les additionner) est inférieure au surplus du consommateur une fois que la durée du brevet est achevée. La différence, c'est justement la perte sèche (triangle III) dont on vient de parler.

Toute la question est maintenant de savoir si cette perte de bien-être est justifiée par le surplus de bien-être qui suivra. D'où un arbitrage qui n'est pas sans analogie avec la démarche d'un investisseur qui compare des profits futurs à des sacrifices présents. Le problème vient de ce que cet arbitrage est rendu non par un individu seul responsable de ses actes et de ses choix et qui peut comparer la valeur pour lui d'un bien futur à un bien présent, mais par la « société » qui ne peut être en aucune façon être ramenée à une seule entité individuelle douée de raison, sauf à pécher par anthropomorphisme social.

Cette tendance affleure par exemple dans la lettre souvent citée de Jefferson à McPherson : « Les inventions ne peuvent être par nature un objet de propriété. La société peut concéder un droit exclusif sur les fruits qui en résultent, pour encourager les hommes dans la recherche d'idées pouvant être de quelque utilité, mais que la société concède ou non un

tel droit dépend de son bon vouloir, sans que quiconque puisse revendiquer ou s'en plaindre ».

A partir de là, il devient évident que la « société » va avoir son mot à dire sur les innovations qui méritent d'être brevetées.

2.c. Tautologies.

En ce qui concerne la définition des inventions brevetables, il en résulte ces formules qui paraissent si bizarres au profane par l'impression de tautologie qu'elles donnent. L'invention doit être utile (useful aux Etats-Unis, ayant une application industrielle en France), ne doit pas être évidente pour un spécialiste (démontrant une activité inventive), et doit faire preuve de nouveauté.

Pour résumé, une invention doit être inventive et une nouveauté nouvelle. En plus, elle doit être « utile », mais selon les normes en vigueur. Signe évident que l'on n'est pas sorti de la théorie économique classique. Ce n'est pas le lieu de faire ici une critique complète de cette théorie⁹. Le problème du brevet n'est qu'un aspect de ses défaillances.

En ce qui concerne les logiciels, le contrôle de la société a conduit à aller plus loin encore dans la tautologie. Un programme d'ordinateur n'est pas brevetable « en tant que tel ». Il faut, du moins en Europe, qu'il ait un effet technique.

On n'est sorti de la tautologie (l'invention est inventive) que pour entrer dans une autre (la technique est technique) Cette nouvelle impasse est fort gênante et nourrit en grande partie la controverse actuelle au Parlement européen. Et pour cause : les flux électroniques induits dans les circuits de l'ordinateur par un programme constituent un effet technique. Et dans ces conditions tout programmes d'ordinateur en tant que tel est brevetables. D'où la nécessité de définir un critère plus restrictif. Ce à quoi, on le sait l'Office Européen des Brevets s'est employé : il faut pour qu'une invention soit brevetable, énonce-t-il, qu'elle ait des effets techniques « au-delà des interactions physiques normales entre un programme et l'ordinateur, le réseau ou un autre appareil programmable sur lequel celui-ci est exécuté », formule reprise par le conseil de l'Union Européenne. La brevetabilité suppose qu'il ait dans l'invention en question une nouvelle contribution technique ou un apport technique.

Aucune justification sur le fond n'est avancée par l'Office de l'exclusion du champ de la brevetabilité pour absence d'effet technique. Aucune explication n'est donnée permettant de comprendre pourquoi l'existence ou l'absence

⁹ Se reporter ici à Economie du droit, tomes I et II (Belles Lettres), et à L'Erreur économique (Denoël)

de caractère technique sépare les innovations en celles qui devraient être brevetables et celles qui ne le seraient pas. On en est réduit à supputer que les principales motivations de l'Office Européen des Brevets est d'éviter un surinvestissement dans les domaines où la R&D est déjà propulsée par d'autres moteurs que la rente apportée par un brevet. Mais la dénomination « non technique » agrège de fait des activités de R&D très différentes, certaines présentant des spécificités susceptibles de motiver une protection par brevets », comme le dit excellemment Bernard Caillaux, déjà cité¹⁰

Michel Rocard, député européen, qui n'est pas le dernier à regretter la tautologie des définitions européennes, propose d'entendre par domaine technique un « domaine industriel d'application nécessitant l'utilisation de forces contrôlables de la nature pour obtenir des résultats prévisibles dans le monde physique ». Il rejoint d'une certaine manière la définition japonaise de la brevetabilité : « toute création hautement développée d'idées techniques par l'application d'une loi de la nature », qui est considéré comme plutôt laxiste...

3) Le texte fondateur de l'Office Européen des Brevets, à savoir la convention de Munich, en ce qui concerne la brevetabilité des logiciels, reflète cet embarras, auquel s'ajoute la confusion née de l'assimilation du logiciel au domaine des idées. D'où la fameuse formule : le programmes d'ordinateur en tant que tel n'est pas brevetable. Formule qui a été reprise par le conseil de l'Union européenne du 7 mars 2005.

3.a. L'article 52.

A la lumière des analyses précédentes, il peut être intéressant de relire l'article 52 de la convention de Munich qui sert encore de référence dans les discussions sur la brevetabilité des logiciels

1) *Les brevets européens sont délivrés pour les inventions nouvelles impliquant une activité inventive et susceptibles d'application industrielle.*

2) *Ne sont pas considérées comme des inventions au sens du paragraphe 1 notamment :*

a) *les découvertes ainsi que les théories scientifiques et les méthodes mathématiques ;*

b) *les créations esthétiques ;*

c) *les plans, principes et méthodes dans l'exercice d'activités intellectuelles, en matière de jeu ou dans le domaine*

¹⁰ Bernard Caillaux, op. cit. p. 159.

des activités économiques, ainsi que les programmes d'ordinateurs ;

d) les présentations d'informations.

3) Les dispositions du paragraphe 2 n'excluent la brevetabilité des éléments énumérés aux dites dispositions que dans la mesure où la demande de brevet européen ou le brevet européen ne concerne que l'un de ces éléments, considéré en tant que tel.

4) Ne sont pas considérées comme des inventions susceptibles d'application industrielle au sens du paragraphe 1, les méthodes de traitement chirurgical ou thérapeutique du corps humain ou animal et les méthodes de diagnostic appliquées au corps humain ou animal. Cette disposition ne s'applique pas aux produits, notamment aux substances ou compositions, pour la mise en œuvre d'une de ces méthodes.

3.b. Prendre la théorie économique de la propriété comme guide. Si l'on essaie de trouver une logique dans cet inventaire à la Prévert, au demeurant assez peu poétique, on prendra comme guide non pas la théorie économique classique, mais la théorie de la propriété issue de l'analyse économique du droit. Résumons de quoi il s'agit en commençant par la lettre déjà citée, si intéressante, de Jefferson à McPherson sur le contrôle de la société sur les inventions. Dans cette même lettre, l'on peut lire des formules appelées à la célébrité, qui vont littéralement polluer le débat sur la brevetabilité : « celui qui reçoit de moi une idée, reçoit un savoir sans affaiblir le mien ; comme celui qui allume sa chandelle à la mienne reçoit la lumière sans atténuer la sienne ». La dite communauté du logiciel libre pourrait aisément s'en inspirer.

En termes contemporains, les économistes traduiraient la lettre de Jefferson de la manière suivante : une idée est un bien à la fois non excluable et non rival :

non excluable parce qu'il est impossible d'exclure de l'usage un utilisateur même si ce dernier ne contribue pas au financement du bien ;

non rival, parce que la « consommation » d'une idée par un individu ne diminue pas la quantité qui reste disponible pour les autres.

Pour les économistes contemporains, ces deux qualités de non exclusivité et de non rivalité définissent ce qu'ils appellent un bien collectif ou un bien public, et beaucoup en déduisent que ce type de bien ne peut être financé que par la collectivité, alors même que l'on peut facilement montrer qu'il existe dans la réalité des biens collectifs privés.

Et l'on arrive facilement à cette formulation, qui est, elle aussi, devenue un lieu commun : « La connaissance est un bien

publique mondial »¹¹ . Ou encore : « Toute protection de la propriété intellectuelle, quelle qu'elle soit, implique la privatisation d'une connaissance faisant partie du patrimoine commun de l'humanité. ¹²»

La non rivalité n'a jamais empêché la commercialisation d'un produit. Le fait d'écouter de la musique dans une salle de concert n'enlève rien au plaisir que tire un autre spectateur de cette même musique au même moment. Il existe pourtant des salles de concert privées où se produisent des orchestres privés constitués de musiciens indépendants.

Le point crucial est l'excluabilité. La théorie économique de la propriété invite à remplacer ce terme par appropriabilité. Un bien est appropriable si le bénéfice que je tire de cette appropriation est supérieur au coût de cette même appropriation.

Ce qui revient à dire que la propriété ne va pas de soi. Elle émerge quand la balance cost/benefit que l'on vient de décrire est positive.

Exemple : je plante des roses dans mon jardin. Ce jardin est à moi, ce sont mes roses. Suis-je pourtant propriétaire de leur parfum, de leur beauté ? Si par propriété j'entends la capacité d'exclure autrui de la consommation de mon bien, alors je ne pourrais être propriétaire de cette qualité de mes roses qu'en érigeant un mur tel qu'il empêche les passants d'en profiter. Le « coût » d'une telle construction – et pas seulement le coût d'érection du mur – est évidemment prohibitif. Donc je livre et ne puis faire autrement que de livrer au domaine public cette qualité-là de mes roses. Sans doute en suis-je le propriétaire légal ? Mais je n'en suis pas le propriétaire économique, le possesseur.

La première condition pour qu'un objet quel qu'il soit, puisse être brevetable est qu'il soit appropriable. Comment pourrais-je breveter quelque chose qui ne m'appartient pas ? Le défaut du terme classique est qu'il éloigne la réflexion de la question des droits de propriété qui doit être replacée au centre de la discussion sur la brevetabilité des logiciels.

Ainsi les idées ne sont pas brevetables, parce qu'elles ne sont pas appropriables, et elles ne sont pas appropriables parce que la balance cost/benefit de leur appropriation serait négative. Il n'est pas besoin pour arriver à ce résultat de recourir à de grandes notions comme le patrimoine de l'humanité ou le bien public mondial, qui relèvent plutôt du *flatus voci*.

3.c. la logique de l'article 52.

Disposant de cet outil, relisons l'article 52 de la convention de Munich.

¹¹ Jean Tirole, op. cit. p. 14.

¹² Id. p. 15.

Les exclusions du paragraphe 2, alinéa a, concernent « les découvertes ainsi que les théories scientifiques et les méthodes mathématiques », dont on peut concevoir qu'elles relèvent du monde des idées, donc non appropriables, et par conséquent non brevetables.

L'exclusion de l'alinéa b relève d'un autre dispositif de protection, celui des « modèles »

Les exclusions de l'alinéa c, à savoir « les plans, principes et méthodes dans l'exercice d'activités intellectuelles, en matière de jeu ou dans le domaine des activités économiques, ainsi que les programmes d'ordinateurs », relèvent du même principe que celles de l'alinéa a. Il s'agit encore d'un monde idéal, non appropriable et donc non brevetable. Les programmes d'ordinateur sont en quelque sorte inclus dans cette liste par raccroc. Ils y figurent bien, et sont donc rangés dans le monde des objets non appropriables, et donc non brevetables.

L'alinéa d obéit à la même logique : quoi de moins appropriable qu'une information ?

L'inventaire à la Prévert tout d'un coup prend une certaine logique.

Cependant, le paragraphe 3 est plus difficile à interpréter. Il vient qualifier les exclusions du paragraphe 2. Ce qui est désormais exclu de la brevetabilité, c'est seulement un élément « considéré en tant que tel ». On le voit, l'expression ne s'adresse pas seulement aux programmes d'ordinateur. Et il serait intéressant de vérifier si elle a été utilisée pour d'autres domaines que les logiciels. Il semble que cela a été le cas dans la présentation d'informations et dans les jeux.

Néanmoins, on peut comprendre que le paragraphe 3 est inspiré du souci de ne pas accorder la brevetabilité à un élément qui pris en tant que tel appartiendrait à l'ordre des idées non appropriables.

Le paragraphe 4 relève d'une tout autre logique qui est celle du respect de la vie en tant que telle, qu'elle soit humaine ou animale. Elle est hors de notre sujet. Elle repose sur la logique de l'exception ou de l'urgence, qui peut se justifier économiquement. Si des considérations de brevets empêchent de sauver une vie, la perte est irrémédiable.

3.d. Toilettage.

On conviendra que la rédaction de l'article 52 est plutôt maladroite et qu'un toilettage s'impose. Les seuls petits mots « en tant que tel » sont la source de confusion, d'incertitude et d'insécurité juridiques. Déjà en décembre 1997, une table ronde organisée par l'Union des Praticiens Européens en Propriété Industrielle se terminait par cette conclusion :

1) the exclusion of « programs for computers » in Art. 52 (2) European Patent Convention should be deleted or at least clarified and that

2) great efforts should be made to inform private inventors as well as small and medium-size firms in Europe that patents are available for computer software.

4) La pratique s'est écartée des textes : des dizaines de milliers de logiciels ont été brevetés en Europe comme dans d'autres parties du monde.

4.a. Prolifération des brevets logiciels.

Que la législation actuelle sur la brevetabilité des logiciels soit contournée, on en trouve facilement la preuve dans le nombre de brevets qui existent déjà sur les logiciels en Europe. Pour ne rien dire de la multiplication des brevets logiciels à laquelle on assiste aux Etats Unis et au Japon, où, il est vrai, la législation est moins restrictive. On a donc assisté à une extension de l'espace du brevetable logiciel en toute légalité...

4.b. Evolution de la jurisprudence.

La position a évolué en Europe à partir de la décision Schlumberger Cour d'Appel de Paris du 15 juin 1981, qui renversait la jurisprudence Mobil Oil (Cour d'Appel de Paris, 22 mai 1973). La jurisprudence de 1981 établissait que l'objet de la revendication en question ne se réduisait pas à un traitement d'informations par ordinateur ; elle comportait également des mesures concrètes relevées dans les forages. Un procédé ne pouvait être privé de brevetabilité pour le seul motif qu'une ou plusieurs de ses étapes étaient réalisées par ordinateur. Une telle solution aboutirait, en effet, à exclure du domaine de la brevetabilité des inventions importantes qui nécessitent l'intervention d'un programme d'ordinateur. Pourquoi le hardware serait brevetable, et pas le software ?

A la suite de quoi les spécialistes des brevets oeuvrant dans le secteur informatique ont vite compris que l'exception de brevetabilité prévue pour les programmes d'ordinateur par la convention de Munich n'avait guère de sens. D'autant que depuis le début des années 1970, les appareils industriels étaient passés d'une logique câblée à une logique programmée.

L'Office Européen des Brevets a suivi le même chemin jusqu'aux décisions IBM T935/97 et T1173/97 qui ont accepté les revendications de programme-produit. A partir de là, l'OEB a mis fin à toute ambiguïté, car il était, clair qu'en acceptant une revendication de programme-produit, le programme seul était visé *de facto*.

Aux Etats-Unis, l'évolution des pratiques de l'USPTO a précédé, et de manière encore plus radicale, l'évolution européenne. Il est vrai qu'outre-Atlantique, le contexte avait radicalement changé depuis l'adoption en 1980 du *Bay-Dole Act 5(1980)* qui encourage les prises de brevet pour les inventions sponsorisées par l'Etat qu'elles soient faites par les agences, les universités ou les entreprises privées. Les chercheurs universitaires ont été incités à sortir de leur tour d'ivoire en brevetant leurs innovations et en créant des entreprises - avec en général un petit intéressement monétaire pour l'université. Du coup le nombre de brevets par an a été multiplié par trois. Au début des années 1990, on a assisté aux Etats-Unis à un débat tout à fait semblable à celui se déroule aujourd'hui dans l'Union européenne. Le software n'était brevetable que s'il était combiné à du hardware. A la suite d'une série de décisions de justice, le logiciel est devenu brevetable (cf. USPTO's « Examination Guidelines for Computer-Related Inventions », de 1995). Très vite le software est devenu indépendant du hardware. En 1962, pour la première fois, un seul software a atteint 1 million de dollars de vente. Aujourd'hui 150 milliards de dollars de software sont vendus chaque année aux Etats-Unis.

En Europe, avec l'infléchissement de la position de l'Office Européen des Brevets, quelques dizaines de milliers de brevets logiciels ont été délivrés et la protection par brevets des logiciel est donc un fait bien établi dans l'Union européenne.

L'évolution récente dans les décisions d'attribution et dans la résolution des conflits semblent donc entériner la brevetabilité des logiciels, y compris des logiciel en tant que tels. L'OEB a même accordé des brevets même pour des méthodes commerciales. En 2000, la décision TY0931/95 a rejeté une méthode de gestion des fonds de pension ; mais dans le même temps, des brevets portant sur un procédé interactif de sélection d'un choix dans un menu (EP 756731) ou de format d'écriture unique pour toutes les tâches de gestion d'une entreprise (EP 02099907) ont été accordés. En somme, le brevet de logiciel existe en Europe sans grandes restrictions¹³.

Bref, l'OEB a fait évoluer sa procédure d'examen vers une plus grande acceptabilité des programmes logiciels, faisant fi de sa tutelle gouvernementale, sous la pression des industries traditionnelles qui voyaient avec angoisse une partie de leur chaîne de valeur ajoutée passer dans un domaine qui ne relevait pas des modes de protection qu'ils maîtrisaient.

Peut-on revenir en arrière ? Mais que faire de tous les brevets logiciels déjà octroyés ? On n'imagine pas de lois

¹³ Bernard Caillaux, op. cit. p. 155.

rétroactives. La proposition de directive de la commission européenne est venue de la nécessité de sortir de ce dilemme...

5) Par nature, le logiciel est brevetable, ce que fait apparaître la théorie économique de la propriété. Aussi bien le Conseil du 7 mars a-t-il bien fait d'ajouter une nouvelle disposition aux textes fondateurs.

5.a. La spécificité du logiciel.

Un logiciel, qui est une fonction ou un procédé, mais non, encore une fois, une idée abstraite, dispose d'une spécificité très particulière : son « secret de fabrication » peut être aisément conservé. L'inventeur d'un logiciel peut le mettre sur le marché sans révéler le code source et en ne dévoilant que le code binaire, lequel est, en principe, protégé par le droit d'auteur – *copyright*. Même en cas de dépôt de brevet, on l'a vu, le code source peut rester caché.

La distinction entre code source et code binaire est une bénédiction pour les gens qui veulent vendre du software parce qu'elle leur permet de maintenir secrète et de protéger leur investissement en propriété intellectuelle tout en distribuant seulement le code binaire.

Le code binaire est compréhensible par une machine mais opaque pour un être humain. Certaines techniques de « décompilation » permettent d'obtenir une idée du code source à partir du code objet, mais le processus est complexe, très imparfait et coûteux. On pourrait penser que le reverse engineering pourrait être automatisé, mais cela n'est pas. La compilation rend le code binaire opaque et garde tout son secret au code source, surtout s'il n'est accompagné d'aucun commentaire.

Le problème de l'ouverture au public est spécial pour la programmation. Pour d'autres formes de propriété intellectuelle, la révélation au public est une part inhérente du processus. Les gens peuvent objecter au fait d'une œuvre propriétaire, mais au moins ils peuvent la voir et travailler avec elle. Un professionnel de l'énergie par exemple peut examiner les brevets qui le concernent. Un professeur de droit lit des articles et a une latitude considérable pour faire un usage loyal des idées qu'ils contiennent sans autorisation et sans paiement. Un compositeur peut être inspiré par une musique. Seul le programmeur peut être obligé de regarder une structure opaque, mis hors du processus qui l'a produit.

La sensation de frustration s'étend non seulement à la personne qui essaie de travailler avec le programme lui-même, mais aussi à celle qui essaie d'écrire des programmes en interaction avec lui. La programmation n'est pas une science exacte. Mais tous les programmeurs n'ont pas toujours le même état d'esprit sur cette question. Le même, face à une situation dans

laquelle quelqu'un est en train d'essayer d'écrire une application qui utilisera son programme, peut avoir une vue différente des bénéfices de l'ouverture. Pourquoi aiderait-il un autre programmeur à écrire un programme concurrent, qui, à cause de coûts de développement plus faibles, pourrait ruiner l'original ? Finalement, les programmeurs sont confrontés au problème classique du *free ride*, c'est-à-dire que chacun est mieux s'il garde son code secret tandis que tous les autres rendent le leur public.

Que le logiciel soit si souvent assimilé au monde des idées vient de ce que la différence de nature entre l'information et le support sur lequel elle est mise en œuvre est très mince, alors qu'elle est plus nette pour les systèmes physiques, mécaniques ou chimiques.

5.b. Logique économique du copyright.

La théorie économique de la propriété explique facilement l'existence du droit d'auteur. Il faudrait en effet un hasard extraordinaire pour que d'un mélange de lettres agitées de manière aléatoire il sorte un poème de Victor Hugo. Autrement dit le poème en question est facilement reconnaissable, donc appropriable sans frais.

Il en est de même pour le code binaire d'un logiciel mis sur le marché. Il est appropriable et donc susceptible d'être protégé par un droit d'auteur. Ce droit confère un droit moral et un droit patrimonial à l'auteur d'un logiciel, dont le texte est considéré comme une œuvre originale couverte par le *copyright*. Ce type de protection est automatique. Du point de vue économique, les droits patrimoniaux instaurent un droit de propriété sur la forme du logiciel, caractérisé par la faculté d'interdire la copie du logiciel. L'auteur obtient ainsi un monopole dans l'exploitation et la distribution du logiciel, dans la mesure où l'utilisation d'un logiciel sur un ordinateur suppose la copie de tout ou partie du logiciel dans la mémoire de l'ordinateur¹⁴.

Comme on l'a dit, le droit d'auteur ne protège pas l'innovation. Il doit donc être éventuellement complété par un brevet.

5.c. Appropriable, donc brevetable.

Or, du fait que le code source n'est pas révélée lors de la mise du logiciel sur le marché. Cet aspect du logiciel est donc lui aussi appropriable. Comme il n'est pas protégeable par droit d'auteur, il faut bien qu'il le soit par une autre institution, qui ne peut être que le brevet. *Ergo*, le logiciel en tant que tel (*as such*) est brevetable, qu'il soit inscrit ou non dans du hardware. La protection par brevet ne naît pas au plan du code ; elle est à un niveau supérieur, fonctionnel

¹⁴ Au plan européen, le droit d'auteur sur logiciel est régi par la directive 91/250/CEE.

On a vu le cas de logiciel de calcul scientifique pour lequel de nouveaux résultats de mathématiques appliquées ont été développés et on ne voit pas au nom de quoi on ne brevetait pas ce type d'innovation. Par exemple, l'algorithme LZW a été breveté par Unisys, et concédé à 135 sociétés qui l'exploitent sous licence.

Le principe de base du droit de propriété est que le propriétaire est libre de faire ce qu'il veut de sa propriété si cette propriété existe. Ce principe devrait pouvoir être admis par la communauté du logiciel libre.

Personne ne peut obliger le propriétaire d'un logiciel à divulguer son invention alors même qu'il peut livrer ce logiciel au public sans la divulguer.

S'il se livre à cette divulgation, personne ne peut l'empêcher d'essayer d'en tirer profit. Il ne s'agit pas de s'interroger sur les conséquences éventuelles, positives ou négatives, de cette divulgation ou de cette non-divulgation. On n'a vu que cette discussion ne menait à rien. Et on retenu comme principe que ni la connaissance, ni l'innovation, ni la croissance économique étaient des valeurs en elles-mêmes. Il s'agit seulement d'appliquer le droit de propriété sur des objets où il peut s'appliquer, c'est-à-dire sur des objets appropriables.

5.d. Les raisons du succès du logiciel libre.

Puisque la brevetabilité du logiciel est dans la nature des choses et que la pratique de l'Office Européen des Brevets a évolué pour obéir à cette nature, on peut s'interroger sur le succès du logiciel libre. C'est que le mouvement de l'*open source* a de nombreux supporters¹⁵.

- a) Les programmeurs et les administrateurs des sociétés informatiques parce que le logiciel libre rend plus aisé l'accomplissement de leurs tâches ; les programmeurs aiment écrire des codes et ils veulent les faire marcher ;
- b) Les croyants dans le logiciel libre qui pensent que le logiciel et les créations intellectuelles ne doivent pas être traités comme des propriétés ;
- c) Les intellectuels qui prophétisent un nouveau mode collectif de production ;
- d) Les sceptiques de la propriété intellectuelle qui perçoivent correctement qu'il y a un problème, mais pensent incorrectement que le logiciel libre permet de l'éviter. Ces gens ont reçu l'appui
- e) Des utilisateurs de logiciel qui y voient une occasion de free ride et de faire baisser les prix ;
- f) Des développeurs de logiciel ;
- g) De grandes sociétés informatiques qui se servent de Linux de multiples façons : comme une arme de concurrence pour

¹⁵ DeLong, op. cit.

placer ses machines, comme une arme contre les autres producteurs de software ;

h) Des distributeurs de programmes open source pour vendre leurs services ;

i) Des intérêts idéologiques des socialistes du 21^e siècle cherchant à assurer à la collectivité le contrôle des secteurs clefs de l'économie ;

j) Des zélotes anticapitalistes ;

k) Des tenants de l'anti-américanisme primaire qui perçoivent l'industrie du logiciel comme dominée par les Etats-Unis.

5.e. Le conseil s'est conformé à la nature du logiciel.

De cette cacophonie, le conseil de l'Union Européenne du 7 mars a su s'extraire. Comme on l'a dit, il introduit une nouvelle disposition afin de préciser que, dans certaines circonstances et à des conditions très strictes, un brevet peut correspondre à une revendication pour un programme d'ordinateur, seul ou sur support. Le Conseil estime que cette disposition alignerait la directive sur ce qui est actuellement pratique courante, tant à l'OEB que dans les Etats membres.

Le conseil s'est conformé à la nature du logiciel.

