



La propriété intellectuelle face à l'ingénierie inverse

Bertrand Warusfel, Matthieu Dhenne

► **To cite this version:**

Bertrand Warusfel, Matthieu Dhenne. La propriété intellectuelle face à l'ingénierie inverse. Propriétés intellectuelles, Institut de Recherche en Propriété Intellectuelle, 2016, pp.20-32. <http://www.irpi.fr/revuepi/article.asp?ART_N_ID=820>. <hal-01852555>

HAL Id: hal-01852555

<https://hal-univ-paris8.archives-ouvertes.fr/hal-01852555>

Submitted on 13 Aug 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

La propriété intellectuelle face à l'ingénierie inverse

BERTRAND WARUSFEL

PROFESSEUR À L'UNIVERSITÉ DE LILLE II
AVOCAT

MATTHIEU DHENNE

AVOCAT
DOCTEUR EN DROIT

Polybe raconte que la capture puis l'analyse d'un navire de guerre carthaginois auraient été à l'origine de la puissance navale romaine¹. Toutefois, la contrefaçon demeurait encore inconnue du droit romain². La propriété intellectuelle, consacrée depuis, pourrait-elle aujourd'hui faire obstacle à une telle analyse, destinée à la reproduction d'un navire de guerre capturé ? Cette question vise in fine à fixer la frontière entre liberté de la recherche³ et propriété⁴ ; elle se pose donc en particulier avec l'ingénierie inverse. En effet, cette dernière, également désignée comme le *reverse engineering*, la rétro-ingénierie ou la rétroconception, a pour but la compréhension d'un objet par son analyse soit pour le reproduire servilement soit pour créer un nouvel objet. Il s'agit de réaliser à partir d'un produit tiers, matériel ou logiciel, différentes formes d'observations, de tests, de déconstructions, dans le but d'extraire, de comprendre, de reproduire, de modifier, tout ou partie des données techniques qu'il contient⁵.

De fait, le *reverse engineering* prend des formes diverses selon son objet. La pratique est apparue dans le secteur de la mécanique dans lequel soit on palpe un objet, soit on scanne numériquement sa surface pour dégager un nuage de points analysable par un logiciel. Le *reverse engineering* s'est ensuite, plus récemment, développé dans les secteurs de l'électronique et de l'informatique. Dans le premier secteur, l'étude porte sur le fonctionnement d'un circuit électronique. Dans le second, l'étude s'appuie sur trois outils : les désassembleurs, les débogueurs et les décompilateurs. Un désassembleur permet une analyse statique du programme. Il traduit le langage machine en une série d'instructions assembleur. Un débogueur permet d'étudier ce que fait le programme lorsqu'il est exécuté par une machine. Un décompilateur traduit le langage machine en un langage évolué aisément compréhensible pour l'Homme. Nous devons dès à présent souligner que décompilateur et désassembleur partagent leur capacité à transformer un code binaire (le langage machine) en un langage évolué. Cependant, le langage assembleur obtenu grâce au premier n'est ni concis ni structuré ; cinq à dix fois plus long, il est très difficilement compréhensible pour un praticien.



Cette étude est issue d'un rapport de recherches réalisé dans le cadre du Projet Marshal+, programme de recherches appliquées financé par le Fonds unique interministériel de soutien (FUI) et mis en œuvre par un consortium de recherche mené par Airbus Group et au sein duquel les auteurs ont représenté l'IRPI. L'objectif du projet était la mise en place d'un dispositif résistant à tout type d'« attaques » (matérielles et logicielles). Les auteurs étaient essentiellement chargés d'étudier les aspects juridiques de cette ingénierie inverse, afin de déterminer dans quelle mesure le droit est susceptible de faire obstacle à de telles « attaques ».

1. Polybe, Histoire générale de la République romaine, in *Ouvrages historiques de Polybe, Hérodien et Zozime*, trad. D. Thuillier et A. Desrez, 1836, p. 12. Le navire de guerre en cause était une quinquerème : une galère à cinq rangs de rames.
2. Elle a néanmoins été dénoncée par certains hommes de lettres. V. Horace, *Épîtres*, texte établi et trad. par F. Villeneuve, *Les Belles Lettres*, universités de France, 1989, Livre I, III, n° 15 s., p. 51 et 52. Martial, *Épigrammes*, T. I (Livre I-VII), texte établi et trad. par H. J. Izaac, *Les Belles Lettres*, universités de France, 3^e éd., 1969, Livre I, XXXVIII, p. 26, LII, p. 31, LIII, p. 31, LLXXII, p. 38.
3. C. Blaizot-Hazard, *Droit de la recherche scientifique*, PUF, Thémis, Droit public, 2003, p. 13 s. La liberté de la recherche scientifique n'est pas expressément prévue par un texte. On a proposé de la déduire de la liberté individuelle consacrée par l'article 4 de la Déclaration des droits de l'Homme et des citoyens. En France, le Conseil constitutionnel l'a déduite de la liberté d'expression consacrée à l'article 11 du même texte (Cons. const., déc. n° 94-345 DC, 29 juill. 1994, *Loi relative à l'emploi de la langue française*). Une telle position est critiquable. En effet, la liberté d'expression inclut la liberté d'opinion et la liberté de recevoir ou de communiquer des informations ou des idées sans qu'il puisse y avoir ingérence d'autorités publiques et sans considération de frontières (P. Wachsmann, *Libertés publiques*, Dalloz, Cours, 7^e éd., 2013, n° 461, p. 621). Or, la liberté de la recherche vise une réalité différente. Deux caractères la distinguent de la liberté d'expression. Premièrement, elle implique l'utilisation de méthodes de recherches (protocoles, expériences, etc.). Il ne s'agit jamais d'une expression entièrement libre. Deuxièmement, elle exige la production d'un résultat, autrement dit d'une chose nouvelle.
4. Dans les développements à venir, le concept de propriété sera entendu comme « la relation par laquelle une chose appartient à une personne » (F. Zenati-Castaing et T. Revet, *Les biens*, PUF, Droit fondamental, 3^e éd., 2007, n° 164, p. 260). V. F. Zenati, *La nature juridique de la propriété*, thèse dactyl. université Jean Moulin (Lyon III), 1981, *passim*. V. également S. Ginossar, *Droit réel, propriété et création*, LGDJ, 1960, *passim* et L. Rigaud, *Le droit réel*, A. Nauze, 1912, p. 48 s.
5. Rapp. P. Samuelson et S. Scotchmer, *The Law and Economics of Reverse Engineering*, Yale L. J., 2002, vol. 111, p. 1575 : « pratique d'extraction du savoir-faire à partir d'un artefact créé par l'Homme » (« process of extracting know-how or knowledge from a humanmade »).

Dans le secteur de l'informatique, l'ingénierie inverse constitue une ingérence au sein d'un appareil ou d'un programme d'ordinateur. Cette ingérence peut-elle porter atteinte à une propriété intellectuelle ? Une réponse affirmative s'impose. En effet, ce faisant l'on peut violer la propriété incorporelle d'un tiers, en particulier quand l'enseignement en cause constitue une chose intellectuelle au sens de la loi⁶. Ce type d'atteinte est plausible lorsque l'ingénierie inverse concerne une machine informatique (*hardware*) ou le programme la mettant en œuvre (*software*) ou encore les bases de données incluses dans la machine.

S'agissant d'un équipement de traitement de l'information (*hardware*), l'ingénierie inverse consistera surtout à déterminer comment fonctionnent les composants électroniques y participant tels que le microprocesseur ou des composants spécialisés. En théorie, cela pourrait mettre en cause certains droits de propriété intellectuelle, puisque la topographie de ces composants semi-conducteurs peut être protégée par un droit de propriété intellectuelle, en application de la directive 87/54/CE du 16 décembre 1986 relative à la protection des produits semi-conducteurs (transposée en France par la loi 87-890 du 4 novembre 1987 qui a notamment créé l'article L. 622-1 du CPI selon lequel la topographie finale ou intermédiaire d'un produit semi-conducteur est protégeable). Mais en réalité, cette protection, déjà peu usitée, est aussi limitée par l'exception prévue à l'article L. 622-5 du CPI qui permet la reproduction « aux fins d'évaluation, d'analyse ou d'enseignement » ainsi que la création « à partir de cette analyse ou de cette évaluation, d'une topographie distincte »⁷. Autant dire que l'ingénierie inverse d'un équipement ou de composants électroniques ne rencontrera que rarement d'obstacles dus au droit de la propriété intellectuelle. On ne s'attardera donc pas sur ces aspects, préférant nous consacrer aux difficultés de propriété intellectuelle plus fréquentes que recèle le *reverse engineering* de logiciels (y compris, le code embarqué dans la mémoire des composants électroniques spécialisés) et de bases de données.

La propriété intellectuelle est surtout de nature à entraver le *reverse engineering* quand elle concerne le logiciel. Cette restriction a suscité des débats à propos du programme d'ordinateur après la reconnaissance de sa protection par la propriété littéraire (I). Parallèlement, le *reverse engineering* autorise souvent un accès à des bases de données, lesquelles sont également l'objet d'un droit de propriété intellectuelle sui generis (II).

I. Programme d'ordinateur et ingénierie inverse

Il convient au préalable de déterminer ce qu'est un programme d'ordinateur. L'on admet par principe que c'est une suite d'instructions visant la réalisation de fonctions par une machine⁸. Au sens strict, un logiciel serait davantage qu'un programme. Il s'agit de l'« ensemble des programmes, procédés et règles, éventuellement de la documentation, relatif au fonctionnement d'un ensemble de traitement de données »⁹. Par conséquent,

un logiciel inclut tous les éléments exigés par le fonctionnement du programme d'ordinateur. L'article premier de la directive 91/250/CEE du Conseil du 14 mai 1991 (maintenant remplacée par la directive 2009/24/CE du 23 avril 2009) concernant la protection juridique des programmes d'ordinateur indique néanmoins que « les termes "programme d'ordinateur", aux fins de la présente directive, comprennent le matériel de conception préparatoire ». Cette assimilation se comprend aisément, car, dans la pratique, les deux dénominations, « logiciel » et « programme d'ordinateur », sont employées indifféremment. Elles sont en général considérées comme interchangeables¹⁰. Nous les emploierons donc tels des synonymes dans les développements à venir. En revanche, le programme doit être nettement distingué de l'algorithme. Ce dernier constitue une formule mathématique dont la réalisation dépend de l'intelligence de son exécutant. C'est la forme écrite d'un procédé de calcul composé d'une suite d'étapes. Ainsi, l'algorithme d'une fonction de la

6. M. Vivant, Ingénierie inverse, ingénierie perverse ? : *JCP E* 1991, n° 56.

7. Aux États-Unis, le législateur a également prévu une exception expresse au profit du *reverse engineering* des semi-conducteurs à l'article 306(a) du Titre 35 du Code des États-Unis. V. notam. L. Hsu, *Reverse Engineering Under the Semiconductor Chip protection Act : Complications for Standard of Infringement*, *Albany L.J. Sci. & Tech.*, 1996, vol. 5, p. 249.

8. J. Gualino, *Dictionnaire pratique informatique, Internet et nouvelles technologies de l'information et de la communication*, Gualino, 2006, v. « programme ». La Commission européenne définit ainsi le programme d'ordinateur comme « un ensemble d'instructions qui a pour but de faire accomplir des fonctions par un système de traitement de l'information, appelé ordinateur » (*Livre vert de la Commission des Communautés européennes sur le droit d'auteur et le défi technologique*, COM. [1988], p. 170, n° 5.1.1). De même, aux États-Unis, l'art. 101 du Titre 17 du Code des États-Unis dispose que le programme d'ordinateur est « une série d'instructions à utiliser, directement ou indirectement par l'ordinateur, afin d'obtenir un certain résultat » (Notre traduction de « *A "computer program" is a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result* »). Cette définition peut susciter des difficultés en présence de programmes rédigés en langage orienté objet. En effet, dans ce cas, on raisonne en termes d'« objets », et le programme constitue une réunion d'« objets », ce n'est pas véritablement une « suite » d'instructions. V. F. A. Koch, *Begründung und Grenzen des urheberrechtlichen Schutzes objektorientierter Software* : *GRUR* 2000, p. 191.

9. *Terminologie de l'informatique*, A. min. 22 déc. 1981 : *JORF* 17 janv. 1982, p. 624. De même, une instruction de la Direction générale des impôts, qui date du 12 oct. 1984, définit le logiciel comme « un ensemble d'instructions, de programmes, procédés et règles ainsi que la documentation qui leur est éventuellement associée, relatifs au fonctionnement d'un matériel de traitement de l'information ».

10. J. Huet, L'Europe des logiciels : le principe de la protection par le droit d'auteur (directive communautaire du 14 mai 1991) : *D.* 1992, p. 221, v. spéc. note 20, p. 222 ; A. Lucas, *Œuvres protégées – Logiciels*, in *J.-Cl. Prop. litt., art.*, fasc. 1160 (16 févr. 2015) n° 1 ; F. Pollaud-Dulian, *La brevetabilité des inventions. Étude comparative de jurisprudence France – OEB, Litec, IRPI*, t. 16, 1997, n° 78, p. 69, v. spéc. note 1 ; M. Vivant, Le programme d'ordinateur au pays des musées : *JCP E* 1991, I, 94. V. contra A. Bertrand, *Le droit d'auteur et les droits voisins*, Dalloz, 2^e éd., 1998, n° 12.31 ; J.-P. Martin, La protection des logiciels informatiques : droit d'auteur ou brevet d'invention ? : *JCP E* 1990, II, 15752.

vie quotidienne telle que se laver les cheveux, par exemple, serait défini de la sorte : mouiller les cheveux ; appliquer le shampoing sur le cuir chevelu ; masser ; émulsionner ; rincer abondamment¹¹. Dans le domaine de l'informatique, l'algorithme correspond à une fonction d'un programme d'ordinateur.

L'informatique est née aux États-Unis au cours de la deuxième moitié du vingtième siècle, après que les premiers calculateurs programmables aient été conçus et développés pour remplir des applications militaires durant la Seconde Guerre mondiale¹². La question de la protection des programmes d'ordinateur a donc initialement été soulevée outre-Atlantique¹³. En 1966, une commission présidentielle consacrée à l'appréhension des nouvelles technologies par le droit des brevets rendit un rapport défavorable à la protection des logiciels par brevet¹⁴. Cette exclusion a ensuite été justifiée par le rejet, plus général, de la brevetabilité des théories scientifiques. Il s'agissait alors de considérer qu'un programme d'ordinateur était assimilable soit à un algorithme soit à une suite d'algorithmes, qui ni l'un ni l'autre n'engendrent de transformation de la nature. Dans ce cas, le programme ne peut pas constituer une invention de procédé au sens de l'article 101 du Titre 35 du Code des États-Unis. De plus, il s'agit d'une suite d'étapes intellectuelles qui dépend de l'intelligence d'un exécutant. Sa description ne peut pas suffire à son exécution puis à sa reproduction. Elle est donc insuffisante au sens de l'article 112 du Titre 35 du Code des États-Unis. Ces deux arguments ont été réunis au sein de la théorie dite « des étapes mentales » (*mental steps*), selon laquelle le procédé dont la réalisation est uniquement subordonnée à l'intelligence de son exécutant n'est pas brevetable¹⁵.

Néanmoins, dès les années soixante, le rejet de la protection des programmes d'ordinateur par la propriété industrielle s'est accompagné de l'admission de sa protection par la propriété littéraire. En effet, dès 1964, l'*US Copyright Office* a reconnu que les logiciels étaient protégeables par copyright. Le législateur américain a consacré cette pratique avec la loi du 12 décembre 1980. Les législateurs français et allemand ont adopté des solutions similaires en 1985. La directive 2009/24/CE du 23 avril 2009¹⁶ et l'accord sur les Adpic du 15 avril 1994 reprennent aujourd'hui cette position en vertu de laquelle le logiciel est protégé par le droit d'auteur¹⁷ (A). Par ailleurs, la jurisprudence, aux États-Unis puis en Europe, a progressivement admis la brevetabilité des inventions portant sur des programmes d'ordinateur, en dépit de son rejet originel (B).

A. Au regard du droit d'auteur

Le *reverse engineering* implique-t-il une atteinte au droit d'auteur portant sur un logiciel ? On l'a souvent jugé comme tel, en particulier dans le cas de la décompilation, parce qu'elle autorise un accès à un code source proche du programme étudié, et implique de la sorte une divulgation ainsi qu'une reproduction partielle du

code initial. Mais plusieurs exceptions ont été adoptées afin de sauvegarder la liberté de la recherche face à la propriété en garantissant la licéité du *reverse engineering* dans les situations suivantes : la décompilation (1), les actes d'observation, d'étude et de test (2) et la maintenance (3).

1. La décompilation

La décompilation ne constitue qu'une partie du processus d'ingénierie inverse dans son ensemble. Elle concerne seulement la traduction du langage machine en un langage évolué compréhensible pour l'Homme. L'on estime souvent que la décompilation permet d'accéder au code source. En vérité, un tel accès est impossible. Un décompilateur autorise seulement la détermination d'un langage évolué le plus proche possible du code source initial. Un véritable accès au code source exigerait un accès à l'ensemble des sources du programme, et en particulier aux nombreux commentaires insérés dans le code, qui sont indispensables à son entière compréhension¹⁸.

Cette traduction d'un code objet en un code source constitue-t-elle un acte de contrefaçon au sens de la propriété littéraire ? La question a au départ été soulevée aux États-Unis, où la protection des logiciels par copyright a été admise pour la première fois. La cour d'appel du 5^e circuit a reconnu la licéité de la décompilation



11. X. Thunis, Les modes de protection juridique du logiciel : *DPCI* 1983, p. 121, note 4.

12. Ph. Breton, *Histoire de l'informatique*, La Découverte, 1987, p. 76 s.

13. Les premiers articles publiés à ce sujet l'ont naturellement été aux États-Unis. V. J. F. Banzhaf, Copyright Protection for Computer Programs : *Colum. L. Rev.* 1964, vol. 64, p. 1274 ; A. H. Seidel, Antitrust, Patent and Copyright Law. Implications of Computer Technology : *JPTOS* 1962, vol. 64, p. 116.

14. The President's Commission on the Patent System, *Report to Promote the Progress of Useful Arts*, 1966.

15. *Gottschalk v. Benson*, 409 US 67, 175 USPQ 673 (1972). V. P. Ambrose, The Mental Steps Doctrine : *Tenn. L. Rev.* 1981, vol. 48, p. 903 ; N. D. McClaskey, The Mental Process Doctrine : Its Origin, Legal Basis and Scope : *Iowa L. Rev.* 1970, vol. 5, p. 1149.

16. Directive 2009/24/CE du 23 avr. 2009 concernant la protection juridique des programmes d'ordinateur : *JOCE* 5 mai 2009, n° L 111, p. 16.

17. Art. 10 de l'accord sur les Adpic : « Les programmes d'ordinateur, qu'ils soient exprimés en code source ou en code objet, seront protégés en tant qu'œuvres littéraires en vertu de la Convention de Berne (1971) ».

18. H. Croze et F. Saunier, Logiciels : retour aux sources : *JCP G* 1996, I, 3909. Ces sources sont constituées par « toutes les informations non publiques nécessaires à un professionnel de l'informatique pour modifier ou maintenir le logiciel considéré ». Cela comprend non seulement le code source, mais également l'ensemble des informations nécessaires à son intelligibilité (liste des instructions dans un langage de programmation évolué, indications nécessaires à l'intelligibilité du code source, description des structures de programmation, algorithmes, description des formats de fichiers ou des bases de données utilisés). L'accès aux sources implique donc que le programme soit documenté à cet effet. Cette documentation est proche du « matériel de conception préparatoire » éligible à la protection par droit d'auteur (CPI, art. L. 112-2 [13]).

dans l'affaire *Vault contre Quaid*¹⁹. En l'espèce, la société Vault commercialisait un logiciel de « plombage » (Prolock), qui visait la protection d'autres logiciels contre des copies non autorisées. Quaid, une société concurrente, a décompilé Prolock afin de développer un logiciel de « déplombage » (Ramkey) capable de neutraliser Prolock. Vault a engagé une action en contrefaçon sur le fondement du copyright à l'encontre de Quaid. Elle fondait son action sur trois arguments. D'abord, la reproduction temporaire du code de Prolock dans la mémoire vive de l'ordinateur ne constituait pas une copie licite du programme au sens de l'article 107 du Titre 17 du Code des États-Unis²⁰, lequel consacre une exception au droit de l'auteur en vertu de laquelle un « usage acceptable » (*fair use*) de l'œuvre n'est pas constitutif d'une contrefaçon²¹. Ce texte n'étant, selon la demanderesse, invocable que si la copie était réalisée pour faire fonctionner le logiciel afin d'en obtenir l'usage auquel il était destiné. Ensuite, le logiciel Ramkey constituait une œuvre dérivée contrefaisante. Enfin, Quaid n'avait pas respecté ses obligations contractuelles puisque, conformément au *Louisiana Software License Enforcement Act* de 1987, la licence d'utilisation de Prolock interdisait toute copie du programme dans quelque intention que ce soit, y compris en vue de la *reverse engineering*. La décision de première instance, qui rejetait l'ensemble de ces arguments, a été confirmée en appel. Il a été jugé que rien n'indiquait que l'article 107 du Titre 17 du Code des États-Unis n'était invocable que pour une copie destinée à l'usage auquel son auteur l'avait destiné. Néanmoins, la reproduction temporaire qu'impliquait la décompilation ne constituait pas une contrefaçon et ne faisait donc pas intervenir la doctrine du *fair use*. Les juges ont aussi exclu la qualification d'œuvre dérivée. Ils ont également considéré que le *Louisiana Software License Enforcement Act* était inapplicable, car il était contraire au *Copyright Act*, ce qui rendait nulle la stipulation incluse dans la licence d'utilisation de Prolock.

À l'inverse, dans l'affaire *Sega contre Accolade*, il a été jugé que la décompilation constituait une contrefaçon, mais qu'elle entrait dans le champ d'application du *fair use*²². En l'espèce, le système d'exploitation de la console Genesis, l'équivalent américain de la Megadrive européenne, fabriquée par la société Sega contenait un code de sécurité. Ce code devait impérativement figurer dans le programme tournant sur la machine du fabricant. Un éditeur de jeux, Accolade, a cependant réussi à déterminer ce code via une décompilation, afin de faire tourner ses propres jeux sur la console Genesis. Sega a intenté une action en contrefaçon contre Accolade. La cour d'appel fit droit à cette demande. En effet, l'exception de l'article 107 du Titre 17 du Code des États-Unis ne pouvait pas être invoquée. Dès lors, la décompilation n'était pas en elle-même licite, comme cela avait été jugé dans le cas *Vault*. Toutefois, la cour a considéré qu'une telle contrefaçon pouvait être justifiée, et donc pas sanctionnée, si elle était le seul moyen d'accéder à des éléments – les interfaces en l'occurrence – non protégés par le copyright et pour lesquels des raisons législatives justifiaient un accès libre. En d'autres termes : si

l'argumentation diffère de celle adoptée dans l'affaire *Vault*, l'on aboutit à un résultat similaire : la licéité de la décompilation. Mais en la rattachant au *fair use* la cour limite cette licéité aux cas d'interopérabilité²³. Par la suite, la jurisprudence *Sega* a été suivie dans d'autres décisions, même si la défense fondée sur l'exception du *fair use* a été tantôt acceptée²⁴ tantôt rejetée²⁵, en fonction des circonstances de l'espèce.

Dans l'affaire *Nintendo*, le juge américain a assoupli sa position²⁶. Les faits étaient identiques à ceux du précédent *Sega*. En l'espèce, la société Atari avait réalisé une décompilation pour accéder au code de sécurité de la console NES. La décision a été fondée sur l'exception du *fair use*, tout en retenant une position plus souple que dans l'affaire *Sega*. Ainsi, il a été jugé que la copie intermédiaire pour accéder aux idées et aux méthodes de fonctionnement d'une œuvre protégée est licite si elle se justifie. Autrement dit, la licéité de la



19. *Vault Corp. v. Quaid Software Ltd.*, 847 F2d 255 (5th Cir. 1988). V. néanmoins J. Ginsburg, « *Reverse engineering* et protection du logiciel par le droit d'auteur aux États-Unis » : *Dr. informatique et télécoms* 1990, p. 89. J. Ginsburg s'interroge toutefois sur la question de savoir si cette copie en vue de la neutralisation du logiciel rentrait véritablement dans le champ de l'exception : « C'est une chose de démonter un logiciel dans un but éducatif [...] et une autre de le démonter afin de créer un logiciel concurrent ». Il semble qu'en réalité cette question concerne davantage l'utilisation ultérieure qui est faite de l'œuvre que la décompilation elle-même (F. Sardain, *La propriété intellectuelle à l'épreuve des logiciels. L'exemple des interfaces*, thèse dactyl. université de Poitiers, 2002, n° 155, p. 144).

20. V. en ce sens, *MAI Systems Corp. v. Peak Computer, Inc.*, 991 F2d 511 (9th Cir. 1993). V. également T. Lewis, « *Reverse Engineering of Software : An Assessment of the Legality of Intermediate Copying* : *Loy. L.A. Ent. L. Rev.* 2000, vol. 20, p. 561.

21. Art. 107 du Titre 17 du Code des États-Unis. V. J. C. Ginsburg et R. A. Gorman, *Copyrights Law*, Foundation Press, Concepts and Insights Series, 2012, p. 178 s.

22. *Sega Enterprises, Ltd. v. Accolade, Inc.*, 977 F2d 1510, 24 USPQ2d 1561 (9th Cir. 1992). V. J. E. Cohen, *Reverse Engineering and the Rise of Electronic Vigilantism : Intellectual Property Implications of « Lock-Out » Programs* : *S. Cal. L. Rev.* 1995, vol. 68, p. 1091 ; D. S. Karjala, *Copyright Protection of Computer Documents, Reverse Engineering, and Professor Miller* : *U. Dayton L. Rev.* 1994, vol. 19, p. 975 ; P. Samuelson, *Fair use for Computer Programs and Other Copyrightable Works in Digital Form : The Implications of Sony, Galoob and Sega* : *J. Intell. Prop. L.* 1993, vol. 1, p. 49 ; *Unbundling Fair Uses* : *Fordham L. Rev.* 2009, vol. 77, p. 2537, v. spéc. p. 2605 s.

23. R. H. Stern, *Reverse engineering of Software as Copyright Infringement – An Update* : *Sega Enterprises Ltd. v. Accolade Inc.* : *EIPR* 1993, vol. 1, p. 35.

24. *Sony Computer Entm't, Inc. v. Connectix Corp.*, 203 F3d 596, 599 (9th Cir. 2000) ; *DSC Commc'ns Corps. v. DGI Techs., Inc.*, 81 F3d 597, 601 (5th Cir. 1996) ; *Batteman v. Mnemonics, Inc.*, 79 F3d 1532, 1539, 38 USPQ2d 1225 (CAFC 1996).

25. *DSC Commc'ns Corp. v. Pulse Commc'ns, Inc.*, 170 F3d 1354, 1363 (CAFC 1999) ; *Compaq Computer Corp. v. Procom Tech., Inc.*, 908 Fsupp 1409, 1421 (S.D. Tex 1995). V. P. Samuelson, *Comparing US and EC Copyright Protection for Computer Programs : Are They More Different than They Seem* : *J.L. & Com.* 1994, vol. 13, p. 1279, v. spéc. p. 1285 s.

26. *Atari Games Corp. v. Nintendo of Am., Inc.*, 975 F2d 832, 839, 24 USPQ2d 1015 (CAFC 1992).

décompilation n'est pas limitée à l'interopérabilité : elle est également licite de façon plus générale lorsqu'elle est effectuée afin de comprendre et de réutiliser des éléments non protégeables relevant du domaine des idées²⁷.

La décompilation est donc susceptible d'être licite au regard du droit américain, tant qu'elle constitue un « usage acceptable ». L'est-elle en Europe au regard de la législation relative au droit d'auteur ? Contrairement aux œuvres littéraires, le code source d'un logiciel n'est généralement pas divulgué au public (tout au moins lorsqu'il s'agit de logiciels dits « propriétaires », par opposition aux logiciels dits « open source »). L'opération de décompilation aboutit à une divulgation partielle du code. S'agit-il pour autant d'une atteinte au droit de divulgation ? De plus, la décompilation exige des copies en mémoire vive et en mémoire morte. S'agit-il pour autant d'une atteinte au droit de reproduction ? C'est en particulier cette dernière question qui a été le cœur d'une importante controverse. Désormais, l'article 6(1) a) de la directive 2009/24/CE concernant la protection juridique des programmes d'ordinateur prévoit expressément que la décompilation est permise, tout en la subordonnant à la satisfaction de certaines exigences.

L'article 6(1) de la directive 2009/24/CE prévoit que « l'autorisation du titulaire des droits n'est pas requise lorsque la reproduction du code ou la traduction de la forme de ce code [...] est indispensable pour obtenir les informations nécessaires à l'interopérabilité d'un programme de façon indépendante avec d'autres programmes ». Seule la traduction du langage machine en un langage évolué est donc visée. Le texte précise en outre que la mise en œuvre de l'exception dépend de la réunion de plusieurs conditions : 1^o) les actes doivent être accomplis par un utilisateur légitime, le licencié ou une personne jouissant du droit d'utiliser une copie d'un programme ou pour leur compte par une personne habilitée à cette fin ; 2^o) les informations nécessaires à l'interopérabilité ne sont pas facilement et rapidement accessibles pour ces personnes ; 3^o) ces actes sont limités aux parties du programme nécessaires à l'interopérabilité. L'exigence d'interopérabilité est essentielle. Elle répond à trois objectifs : l'accès aux éléments du programme non protégés (les fonctionnalités) ; l'augmentation de l'efficacité des systèmes informatiques, les consommateurs ne devant pas demeurer prisonniers de certains de ces systèmes ; l'absence de limites imposées par des producteurs et qui auraient pour objectif de limiter la concurrence entre producteurs²⁸. La décompilation ne remplissant pas ces exigences porte atteinte au droit d'auteur.

L'autorisation de décompiler constitue une exception au droit de propriété de l'auteur. Et une telle exception est invocable si des conditions strictes sont remplies. Ce n'est en aucun cas une prérogative reconnue par le droit objectif, autrement dit, un droit subjectif²⁹. Il n'existe pas de « droit » de décompilation. Ainsi, la décompilation est en soi une contrefaçon. Mais elle n'est pas sanctionnée lorsqu'elle est effectuée dans un cadre précis. Pourtant, la cour d'appel de

Caen dans une décision rendue récemment dans une affaire *Skype* affirme l'inverse³⁰. En l'espèce, deux informaticiens avaient décompilé le logiciel de communication Skype, puis ils avaient divulgué le code source trouvé sur internet. Or, l'un des auteurs de la décompilation a été condamné tandis que l'autre a été relaxé. En effet, selon la cour, c'est « l'utilisation des données obtenues par la décompilation, qui constitue le délit de contrefaçon ». Le second programmeur n'avait donc pas participé au délit de contrefaçon : « La décompilation du logiciel Skype ne constituant pas un acte de contrefaçon, aucun élément ne vient établir que M. D., qui le conteste, ait participé comme auteur, coauteur ou complice, y compris par voie de fournitures d'instructions ou de moyens, aux actes matériels contrefaisants (publication d'un fichier reproduisant le code source du logiciel Skype) ». Cette motivation est surprenante. En effet, contrairement à ce que déclare la cour, c'est bien la décompilation qui constitue un acte de contrefaçon et non l'utilisation faite des données qu'elle permet d'obtenir.

C'est le plus souvent la question de l'interopérabilité qui cristallise les litiges relatifs à la décompilation. Et, de manière générale, les décisions rendues en la matière tendent à prouver que la technique, dans le domaine de l'informatique, demeure mal comprise par les juridictions. Nous en voulons d'abord pour preuve une décision rendue par la Cour de cassation le 20 octobre 2011³¹. En l'espèce, la société Fiducial informatique était titulaire du droit d'auteur relatif à un logiciel dénommé « H. open » destiné aux huissiers de justice. Le salarié à l'origine de ce programme a par la suite quitté la société Fiducial informatique pour proposer un logiciel concurrent. Un huissier souhaitait changer de système pour faire migrer ses données vers un autre. Dans ce cas, la difficulté résidait, pour l'utilisateur du premier programme, dans la récupération des données ou des fichiers issus de l'usage de cette solution logicielle, afin de continuer à les utiliser au sein du nouvel environnement. Une telle continuité implique une intervention sur le premier programme utilisé. La question posée à la Cour de cassation était donc la suivante : comment et dans quelles limites cette intervention est-elle envisageable ? Il a été jugé que « l'interopérabilité est la capacité d'échanger des informations et d'utiliser mutuellement les informations échangées ; dès lors, la cour d'appel a, à bon droit, énoncé que les opérations

27. G. Pellegrin, *Décompilation, l'Europe en retard d'une réforme : Expertises* 1994, p. 270.

28. J. Drexler, *What is Protected in a Computer Program ? Copyright Protection in the USA and Europe*, VCH, IIC Studies, vol. 15, 1994, p. 101.

29. *Vocabulaire juridique*, sous la dir. G. Cornu, PUF, Quadrige, 2011, v. « droit ».

30. CA Caen, 18 mars 2015, *Le ministère public – Skype Ltd et Skype Software Sarl c/ D. Christian et O. Sean*.

31. Cass. 1^{re} civ., 20 oct. 2011, pourvoi n° 10-14.069 : *RTD com.* 2011, p. 760, obs. F. Pollaud-Dulian.

de migrations de données, réalisées par M. X. et la société Alphapi, habilités à cette fin par les huissiers de justice titulaires de la licence d'utilisation du logiciel "H. Open", pour récupérer les fichiers de ce programme, s'inscrivaient dans les strictes nécessités de l'interopérabilité autorisée par l'article L. 122-6-1, IV du Code de la propriété intellectuelle (CPI) qui prévoit la nullité de toute stipulation contraire ».

L'on peut penser que le raisonnement de la Haute juridiction judiciaire était légitime, puisqu'un utilisateur ne doit pas demeurer prisonnier d'un système propriétaire³². Pour autant, cette décision et sa motivation nous paraissent assez discutables. L'article L. 122-6-1, IV du CPI prévoit seulement un droit de décompilation au profit d'un utilisateur pour lui permettre d'accéder à un langage évolué à partir de son langage machine. L'interopérabilité est la finalité justifiant une telle opération. À défaut d'une telle finalité, les actes de reproduction qu'exige la décompilation sont des actes de contrefaçon. Or, aucune des conditions auxquelles la décompilation est subordonnée – hormis la finalité poursuivie – n'a été évoquée par la Cour de cassation. Les juges se sont seulement fondés sur le but poursuivi, sans tenir compte du cadre dans lequel la décompilation était réalisée. Surtout, il semble que la Cour tienne les expressions « échanges entre des séries d'instructions » et « migrations de données » pour équivalentes. Or, elles ne le sont pas. Il est impossible d'assimiler des instructions à des données. Seules les premières appartiennent à la forme du programme et génèrent des interfaces permettant la communication entre logiciels. La migration désigne un transfert unidirectionnel alors que l'échange consiste dans une communication réciproque entre deux programmes. Dès lors, la migration de données est licite sans limitations (sauf si les données sont elles-mêmes protégées par le droit d'auteur) et seule une intervention visant la détermination d'instructions serait soumise aux conditions et limites de l'exception de décompilation.

Par ailleurs, la décompilation est souvent confondue avec l'ingénierie inverse, alors qu'elle n'en constitue qu'une partie³³. Cette erreur peut aboutir à des conséquences regrettables. D'une part, elle laisse à penser que le *reverse engineering* d'un logiciel serait licite sous certaines conditions alors que ce n'est pas le cas. Seuls certains actes, comme la décompilation ou la maintenance, le sont quand ils respectent des exigences précises. Le *reverse engineering* d'un logiciel est en soi illicite³⁴. D'autre part, une telle position peut aboutir à la confusion de la décompilation avec d'autres étapes de l'ingénierie inverse, en particulier avec le désassemblage. Cette dernière confusion est certes compréhensible : les praticiens eux-mêmes, par abus de langage, confondent souvent les termes de « décompilation » et de « désassemblage ». Ce sont pourtant de deux opérations distinctes. Le désassemblage permet de passer d'un langage machine à un langage assembleur, qui constitue un langage de bas niveau difficilement compréhensible pour l'Homme. La décompilation permet de passer d'un langage machine à un langage évolué d'un niveau plus élevé, dont la compréhension est bien plus aisée.

À l'inverse, le langage assembleur est un langage cinq à dix fois plus long, ni structuré ni concis, très difficile à déchiffrer et au sein duquel on distingue mal les lignes d'instructions des données.

Cette assimilation de la décompilation et du désassemblage est prégnante en droit américain³⁵. Cependant, elle est aussi apparue en Europe. Une décision rendue en 2006 par la cour d'appel de Paris semble réaliser une confusion de ce type³⁶. En l'espèce, Guillaume T., chercheur en biologie à l'université de Harvard, avait désassemblé un logiciel de sécurité (Viguard). Dans un premier temps, il en avait détecté les failles grâce à un désassemblage et au réassemblage des versions qu'il avait téléchargées frauduleusement sur internet. Dans un second temps, il avait réalisé des logiciels capables de neutraliser Viguard. Une action en contrefaçon a été intentée à l'encontre de Guillaume T. Concernant l'élément matériel de l'infraction, la cour constate que le désassemblage constitue un acte de contrefaçon. Les juges déclarent notamment « que la pratique du désassemblage d'un logiciel n'est licite que dans les strictes hypothèses prévues par l'article L. 122-6-1 – IV du CPI, mais constitue une contrefaçon dès lors que comme en l'espèce, l'auteur de la manipulation, non titulaire des droits d'utilisation, n'agit pas à des fins d'interopérabilité et met à disposition des tiers (les internautes) les informations obtenues ».

2. L'observation, l'étude et le test

L'article 5(3) de la directive 2009/24/CE concernant la protection juridique des programmes d'ordinateur dispose que « la personne habilitée à utiliser une copie

32. N. Binctin, Migration de données et interopérabilité. – (comm. Cass. 1^{re} civ., 20 oct. 2011, n° 10-14.069, F-P+B+I, SAS Fiducial Informatique v. DPSI) : *Comm. com. électr.* 2012, étude 6.

33. V. notam. H. Bitan, L'autonomie des droits de décompilation et d'analyse, in *Droit et technique. Études à la mémoire du Professeur Xavier Linant de Bellefonds*, Litec, 2007, p. 57, v. spéc. p. 57 ainsi que p. 59 s. ; B. Edelman, *Droit d'auteur. Droits voisins. Droit d'auteur et marché*, Dalloz, 2^e éd., 1993, n° 381, p. 290 ; T. de Galard, *Le reverse engineering est-il légal ?* : *Expertises* 1992, p. 215. V. néanmoins P. Belloir, *La décompilation d'une disquette est-elle illicite ?* : *Expertises* 1998, p. 190 ; S. Rozenfeld, *Reverse engineering* de logiciels. Des pratiques à double tranchant : *Expertises* 1991, p. 7 ; F. Sardain, préc., n° 147, p. 139.

34. F. Sardain, op. cit., n° 180 s., p. 166 s. ; M. Vivant, *Ingénierie inverse, ingénierie perverse ?*, op. cit. ; *Lamy droit du numérique* 2012, n° 200, p. 126, v. spéc. p. 127.

35. La confusion a été réalisée dans la décision *Sega v. Acolade*. préc. V. également D. L. Hayes, *The Legality of Disassembly of Computer Programs* : *Computer L.J.* 1993, vol. 12, p. 1 ; C. McManis, *Intellectual Property Protection and Reverse Engineering of Computer Programs in the United States and the European Community* : *High Tech. L.J.* 1993, vol. 8, p. 25. Mais cette confusion résulte certainement du fait que la décompilation soit admise en vertu de l'exception du *fair use*. En effet, le *fair use* paraît autant en mesure d'englober la décompilation que le désassemblage.

36. CA Paris, 21 févr. 2006 : *Juris-Data* n° 2006-311713 ; *Expertises* 2006, p. 366 ; *Propri. industr.* 2007, comm. 8, note J. Schmidt-Szalewski.

d'un programme d'ordinateur peut, sans l'autorisation du titulaire du droit, observer, étudier ou tester le fonctionnement de ce programme afin de déterminer les idées et les principes qui sont à la base de n'importe quel élément du programme, lorsqu'elle effectue toute opération de chargement, d'affichage, d'exécution, de transmission ou de stockage du programme d'ordinateur qu'elle est en droit d'effectuer ».

La Cour de justice a précisé les contours de ce droit d'observation, d'étude et de test du fonctionnement du programme d'ordinateur dans sa décision C-406/10 rendue le 2 mai 2012³⁷. En l'espèce, l'exception était invoquée par la société WPL qui souhaitait, avec le développement d'un logiciel alternatif, accéder à un marché occupé par un programme développé par la société SAS Institute. La demanderesse faisait notamment valoir le contrat de licence qui interdisait l'étude, l'observation et le test du programme. La Cour a jugé que la liberté offerte à l'utilisateur par le biais de l'exception était d'ordre public. Toute clause visant à retirer le bénéfice de cette faculté à l'utilisateur légitime est donc nulle. Une telle solution n'est pas étonnante pour un juriste français, dans la mesure où l'article L. 122-6-1 du CPI la consacrait déjà.

3. La maintenance

L'article 5(1) de la directive 2009/24/CE relative à la protection juridique des programmes d'ordinateur prévoit que sa reproduction et sa traduction sont autorisées quand elles sont nécessaires pour permettre à l'acquéreur légitime du programme de l'utiliser conformément à sa destination, notamment pour corriger des erreurs. Ce texte autorise autrement donc la maintenance informatique. Mais qu'est-ce que précisément la maintenance informatique³⁸ ? L'arrêté du 30 décembre 1983 sur l'enrichissement du vocabulaire relatif à l'informatique définit la maintenance comme l'ensemble des actions « tendant à prévenir ou à corriger les dégradations d'un matériel afin de maintenir ou de rétablir sa conformité aux spécifications », précisant que le mot ne doit pas être seulement employé pour les « améliorations fonctionnelles ou de performances apportées à chaque nouvelle version d'un logiciel »³⁹. Selon l'article 95 M, annexe 2 du Code général des impôts, « les activités qui relèvent du secteur de la maintenance [...] sont celles qui concourent, pour l'essentiel, à l'entretien ou à la réparation du matériel technique de production de biens ou de services exploités dans l'un des secteurs mentionnés à ce même alinéa ». Par conséquent, la maintenance implique la correction des éventuelles erreurs que contiendrait le programme. Cela explique que la jurisprudence ait déjà admis que les contrats de fourniture et de maintenance de logiciels étaient indivisibles⁴⁰. Finalement, la maintenance constitue l'ensemble des prestations nécessaires pour « maintenir » un système informatique dans un état de fonctionnement conforme au contrat de fourniture de matériel ou de logiciel.

L'article 5(1) de la directive susvisée autorise expressément la maintenance corrective. Qu'entend-on par

« maintenance corrective » ? Celle-ci intervient à chaque fois qu'un problème se fait jour afin d'y apporter une solution pour y remédier le plus rapidement possible. On peut séparer la maintenance corrective proprement dite de la maintenance curative, laquelle a une portée moins vaste. L'entretien curatif concerne seulement la réparation de pannes tandis que l'entretien correctif tend à remédier aux causes des pannes⁴¹.

Pourrait-on étendre la notion de maintenance corrective en considérant qu'elle comprend également la maintenance évolutive et qu'elle permet ainsi de faire évoluer le logiciel ? Une controverse sémantique est née à ce propos. Doit-on dans ce cas parler de « maintenance » ou de « suivi » ? Deux arguments militent en faveur du terme de « suivi »⁴². D'abord, le mainteneur n'est tenu que d'une obligation de moyens. Ensuite, le caractère hétérogène du contrat de suivi tient à ce que l'utilisateur attend à ce titre davantage que le simple maintien des performances ou la correction d'erreurs. Il veut pérenniser son investissement avec la correction des erreurs inévitables et l'adaptation à un nouvel environnement, voire l'amélioration des performances, avec la fourniture de nouvelles versions. Finalement, on parle improprement de « maintenance évolutive », dans la mesure où cette expression désigne un service par lequel le fournisseur ou un prestataire agréé par lui prend en charge les évolutions techniques visant à enrichir progressivement le logiciel en fonction du contexte technique. Nous opterons par conséquent pour le terme de « suivi » dans les développements à venir⁴³.



37. CJUE, 2 mai 2012, aff. C-406/10, *SAS Institute Inc. c/ World Programming Ltd.* : D. 2012, p. 2836, obs. P. Sirinelli : *JCP E* 2012, p. 1489, note M. Vivant, N. Mallet-Poujoul et J.-M. Bruguière : *Prop. industr.* 2012, comm. 61, note N. Bouche.

38. Sur la définition et la terminologie de la maintenance informatique, v. B. Warusfel, « Le contrat de maintenance », in M. Vivant, L. Rapp et B. Warusfel, *Droit du numérique*, éd. Lamy, 2015 (en particulier n° 910 s.).

39. A. min. 30 déc. 1983 : *JO* 19 févr. 1983, p. 1740.

40. Cass. com., 18 sept. 2002 : *Expertises* 2003, p. 149 ; Cass. com., 2 déc. 2008, n° 07-16.707 : *Juris-Data*, n° 2008-046119.

41. P. le Tourneau, *Contrats informatiques et électroniques*, Dalloz, Dalloz Réf., 7^e éd., 2013, n° 7.15, p. 311.

42. Sur ce débat et en faveur de la notion de « suivi » de logiciel, v. B. Warusfel, *Lamy, préc.*, n° 929.

43. *Contrats informatiques in J.-Cl. Contrats – Distribution*, fasc. 2410 (25 juill. 2001), n° 98, par I. de Lamberterie ; A. Hollande et X. Linant de Bellefonds, *Les contrats informatiques*, Delmas, 6^e éd., 2008, n° 712, p. 174, v. spéc. p. 175 ; J. Huet et Maisl, *Droit de l'informatique et des télécommunications*, Litec, 1989, n° 432 s., p. 405 s. ; A. Lucas, J. Devèze et J. Frayssinet, *Droit de l'informatique et de l'internet*, PUF, Thémis droit privé, 2001, n° 802, p. 540. V. néanmoins P. le Tourneau, op. cit. P. le Tourneau constate que la maintenance évolutive « est fausement nommée ; en effet, il s'agit en ce cas d'une prestation d'ingénierie (ou plus précisément, selon le vocabulaire américain de *reengineering*, c'est-à-dire de reconfiguration). La pratique tend à assimiler à la maintenance évolutive le "suivi" de logiciel ou de progiciel, ce qui est contestable » (n° 7.15.1, p. 311). En effet, selon l'auteur, « techniquement, modifier un logiciel, consiste toujours à développer un nouveau programme à l'intérieur de l'ancien » (n° 7.41.1, p. 318).

Ce suivi évolutif peut être distingué d'un suivi adaptatif consistant à maintenir la compatibilité du logiciel avec un certain environnement.

L'autorisation d'intervention pour garantir un usage du logiciel conforme à sa destination permet-elle d'admettre un suivi évolutif ou un suivi adaptatif ? Le Tribunal de l'Union européenne a répondu à cette question par une décision rendue le 16 décembre 2010 dans l'affaire *Systran*⁴⁴. La société Systran produit des logiciels de traduction automatique. De 1987 à 2002, elle a réalisé un programme adapté aux besoins de la Commission européenne : « EC-Systran ». La Commission a obtenu une licence d'utilisation pour ce programme. Par ailleurs, en 2003, elle a lancé un appel d'offres pour la maintenance et le renforcement linguistique de ce système. Il s'agissait de confier à l'attributaire du marché la réalisation d'améliorations, d'adaptations et d'ajouts de routines linguistiques. Le but de cet appel d'offres était ainsi de confier le suivi évolutif et adaptatif du logiciel à un tiers. Systran a estimé que les opérations envisagées constituaient des actes de contrefaçon. Le Tribunal de l'Union lui a donné raison en jugeant, notamment, que ce qu'autorise la directive ce n'est qu'une maintenance corrective et non du suivi évolutif ou adaptatif.

L'approche retenue par la jurisprudence *Systran* devrait à notre sens être nuancée. L'article 5(1) de la directive dispose expressément que l'utilisateur bénéficie d'un droit de maintenance corrective. Qu'en est-il du suivi ? Il n'est pas évoqué par le texte. Le programme d'ordinateur est une chose évolutive. L'on pourrait dès lors penser que son suivi est licite. Néanmoins, il paraît excessif de reconnaître cette licéité du seul fait du caractère évolutif du logiciel. La question centrale est en réalité de savoir si la modification envisagée est nécessaire à une utilisation du programme conforme à sa destination. L'évolution consistant à apporter des améliorations est sans aucun doute illicite. Des opérations comme la mise à jour du programme (*updating*) ou l'amélioration du programme (*specifications enhancements*) relèvent du contrôle de l'auteur. En revanche, l'adaptation garantissant que le programme puisse être utilisé conformément à sa destination est licite⁴⁵.

Une différence devrait être faite entre l'adaptation (suivi adaptatif), qui est licite, et l'évolution (suivi évolutif), qui ne l'est pas. Si un programme devient inutilisable en raison de circonstances extérieures, son adaptation doit être licite. Il en va par exemple ainsi si des règles de comptabilité sont modifiées par les pouvoirs publics. Dans ce cas, des modifications du programme sont nécessaires pour qu'il puisse fonctionner conformément à ces nouvelles règles. Mais ce type de suivi doit demeurer exceptionnel. L'on ne peut l'admettre que s'il est justifié par des circonstances insurmontables, étrangères à la volonté de son utilisateur. Une décision rendue le 28 octobre 2010 par la cour d'appel de Caen va au-delà en admettant un suivi évolutif ayant conduit à l'ajout de nouvelles fonctionnalités⁴⁶. En l'espèce, une polyclinique avait pris l'initiative d'ajouter des modifications au logiciel de gestion dont elle détenait une licence. La polyclinique avait réalisé

des corrections et ajouté de nouvelles fonctionnalités. Le titulaire du droit avait intenté une action en contrefaçon à son encontre. La cour a rejeté cette demande, parce que les interventions avaient été effectuées en vue de la « survie professionnelle » de la polyclinique. Les fonctionnalités développées étaient indispensables pour assurer la continuité de l'activité de l'établissement. Une utilisation du logiciel sans adaptation aurait constitué « une menace pour la survie de la Polyclinique du Parc et une source de risque pour ses patients ». Cette décision retient une vision étendue de la maintenance paraissant liée aux circonstances de l'espèce. La polyclinique avait fait appel au mainteneur et elle n'était intervenue qu'en l'absence de réaction de celui-ci, en vue de la survie de l'établissement. Une approche aussi large ne devrait donc pas être systématique. Elle offre cependant une nouvelle perspective vers une conception plus compréhensive des adaptations nécessaires au fonctionnement du logiciel conformément à sa destination, y compris, par exemple, lorsque l'adaptation vise le rétablissement de la compatibilité avec la nouvelle version d'un système d'exploitation tiers.

La forme d'expression d'un programme d'ordinateur, son code source, est susceptible d'être l'objet d'un droit d'auteur. Or, l'ingénierie inverse notamment sous la forme de la décompilation, implique une divulgation et une reproduction partielles, et donc une atteinte audit code. Cela explique que des exceptions spécifiques aient été instaurées pour éviter que la liberté de la recherche ne soit entravée dans le secteur de l'informatique. Aucune exception de ce type n'a été prévue en droit des brevets. Une telle absence s'explique par le rejet initial de la propriété industrielle pour les logiciels. Sa reconnaissance par la jurisprudence contemporaine conduit à s'interroger sur l'éventuel impact du droit des brevets sur l'ingénierie inverse.



44. TUE, 16 déc. 2010, aff. T-19/07, *Systran SA et Systran Luxembourg SA c/ Commission européenne* : RIDA janv. 2011, p. 243, obs. P. Sirinelli. V. S. Leriche, L'affaire Systran : une traduction judiciaire retentissante de la protection du logiciel en droit européen : RLDI 2011, n° 2372, p. 10. La décision en cause a été infirmée par la CJUE, mais sur un fondement juridique différent. V. CJUE, 18 avr. 2013, C-103/11, *Commission européenne c/ Systran SA et Systran Luxembourg SA*.

45. CA Versailles, 26 févr. 1999 : D. 1999, 97. Rapp. Y. Bismuth, Un logiciel doit vivre : *Expertises* 1990, p. 433, v. spéc. p. 435 ; J. Viet, Utilisateurs de logiciels. Quel droit d'adaptation et d'interopérabilité ? : *Expertises* 1995, p. 387. V. néanmoins J. Huet et H. Maisl, op. cit., n° 432, p. 407 citant CA Paris, 20 mai 1986 (pas d'adaptation à l'entrée en vigueur d'un nouveau plan comptable) : « il s'avère que le constructeur était seulement tenu de maintenir l'appareil en bon état de marche et n'était aucunement obligé, faute de dispositions expresses en ce sens, de prendre en compte les changements qui pourraient intervenir après la conclusion du contrat dans le plan comptable [...] » et CA Paris, 27 févr. 1987. V. également, CA Paris, 20 févr. 2009 : *Juris-Data* n° 2009-375693.

46. CA Caen, 28 oct. 2010 : RIDA janv. 2011, p. 289, obs. P. Sirinelli.

B. Au regard du droit des brevets

L'on considère généralement que le droit d'auteur à l'opposé du droit des brevets ne protège que la forme d'expression d'une idée, mais non l'idée en tant que telle⁴⁷. En d'autres termes : il n'autorise une protection que de la forme et non du fond. Ainsi, pour un logiciel, par exemple, une invention concernera une fonctionnalité et non son expression par le biais d'un code source⁴⁸. Nous pouvons alors aisément comprendre comment le droit des brevets pourrait intervenir pour faire obstacle à l'ingénierie inverse. Il est même susceptible de constituer de la sorte la protection juridique la plus efficace contre le *reverse engineering*, d'autant plus qu'il ne contient pas d'exceptions qui soient comparables à celles du droit d'auteur. Toutefois, l'exemption relative à l'expérimentation pourrait, sous certaines conditions, restreindre l'étendue du droit du breveté dans ce cas.

Or, l'on tend désormais à admettre que les inventions relatives à des programmes d'ordinateur sont des réalisations techniques, donc brevetables⁴⁹. Originellement, la brevetabilité de ce type d'inventions a été exclue tant aux États-Unis qu'en Europe. La Cour suprême américaine avait ainsi jugé que la demande relative à un convertisseur de nombres décimaux en nombres binaires n'était pas recevable⁵⁰. En effet, la Cour a estimé qu'en l'espèce l'invention revendiquée consistait essentiellement dans un algorithme. Un algorithme ne produit aucune transformation de la nature. Dès lors, il n'est pas qualifiable d'invention de procédé au sens de l'article 101 du Titre 35 du Code des États-Unis. Un tel algorithme, qui en tant que tel n'est pas brevetable, est à distinguer de la machine le réalisant, qui l'est. Cependant, cette position a été assouplie par la Haute juridiction dans l'affaire *Diamond contre Diehr*⁵¹ et la cour d'appel du circuit fédéral reconnaît, depuis le début des années 90, en particulier depuis la jurisprudence *Alappat*⁵², que les revendications produit-programme sont admissibles⁵³. La décision *Alice*, récemment rendue par la Cour suprême, ne remet pas en cause cette position, bien qu'elle semble aboutir à une limitation de la recevabilité des demandes relatives aux méthodes d'affaires⁵⁴. En Europe, la notion de caractère technique a permis à l'OEB puis aux juridictions nationales d'admettre progressivement la protection des inventions informatiques⁵⁵. Les revendications produit-programme sont admises par l'OEB depuis la décision T 1173/97 rendue en 1998⁵⁶. Dans cette affaire la brevetabilité demeure restreinte aux inventions portant sur des programmes d'ordinateur dont l'exécution peut engendrer un effet technique supplémentaire. Une telle restriction paraît amoindrie avec la décision T 424/03, dans laquelle la chambre a considéré que l'usage d'un médium lisible par un ordinateur suffisait à garantir le potentiel technique d'une invention⁵⁷. Par conséquent, l'insertion d'un support lisible par une machine dans une revendication devrait, semble-t-il, assurer la technicité de l'invention revendiquée. En outre, dans l'affaire T 935/97, la chambre de recours technique a aussi reconnu la recevabilité d'une revendication produit-programme qui ne concernait pas le

support lisible par un ordinateur, mais les codes sources qu'il contenait⁵⁸. En France, les juges semblent encore réticents face à l'approche de l'Office. Nous en voulons pour preuve une décision récente rendue par tribunal de grande instance (TGI) de Paris qui l'a récemment rejetée, en partie, en jugeant que les revendications de produit-programme n'étaient pas recevables en droit français⁵⁹. Cette position est à notre sens critiquable. Lorsqu'un logiciel comporte une dimension technique, son produit-programme doit pouvoir donc constituer un produit brevetable. Cette position est en outre à contre-courant de la tendance actuelle et elle laisse à craindre une insécurité juridique accrue préjudiciable aux acteurs économiques du secteur de l'informatique.

Concernant l'étendue de l'appropriation par un droit de brevet, elle est fonction de deux critères : la teneur des revendications du brevet (CPI, art. L. 613-2) et les actes que le droit permet d'interdire⁶⁰. Il revient donc



47. H. Desbois, *Le droit d'auteur en France*, Dalloz, 3^e éd., 1978, p. 22.

48. Sur les différences de protection par droit d'auteur et brevet en matière de logiciel, v. aussi B. Warusfel, « L'ambiguïté de la notion de brevetabilité du logiciel », in *La propriété intellectuelle en question(s) – Regards croisés européens*, Collection IRPI, éd. Litec, 2006, p. 165-170.

49. M. Dhenne, *Technique et droit des brevets*, thèse dactyl., université Panthéon-Assas (Paris II), 2013, *passim* et L'évolution du concept juridique d'invention dans la jurisprudence européenne : *Légicom* n° 53, 2014/2, p. 37.

50. *Gottschalk v. Benson*, 409 US 63, 175 USPQ 673 (1972). V. D. S. Chisum, *The Patentability of Algorithms* : *U. Pitt. L. Rev.* 1986, vol. 47, p. 959.

51. *Diamond v. Diehr*, 450 US 175, 209 USPQ 1 (1981).

52. *In re Alappat*, 33 F3d 1544, 31 USPQ2d 1557 (CAFC 1994). *Comp. In re Schrader*, 22 F3d 290, 30 USPQ2d 1455 (CAFC 1994). V. R. H. Stern, *Federal Circuit Equates Methods of doing Business to Algorithms for Patentability Purposes*, *In re Schrader* : *EIPR* 1994, vol. 16, p. 496.

53. *Court of Appeal of the Federal Circuit* (CAFC) : cour d'appel exclusivement compétente en matière de brevets depuis 1982.

54. *Alice Corp. v. CLS Bank Int'l*, 134 S. Ct. 2347 (2014). Cette interprétation de la décision ressort de la jurisprudence ultérieure des *districts courts* et de la *Court of Appeal of the Federal Circuit*. V. notam. *DDR Holdings, LLC v. Hotels.com, L.P.*, 773 F.3d 1245 (CAFC 2014).

55. M. Dhenne, *Technique et droit des brevets*, thèse préc., *passim*. V. M. Dhenne, *Réflexions sur le critère de la contribution technique* : *RLDI* 2014/108, n° 3575 ; La notion de caractère technique en droit des brevets : *Propr. intell.* 2015, n° 56, p. 253 ; B. May et M. Liens, *Le caractère technique et l'éléphant* : *Propr. industr.* 2011, étude 5 ; T. Pitois et M. Vivant, *La technique au cœur du brevet : un concept social évolutif* : *Dr. et société* 1996, n° 32, p. 117.

56. OEB, CRT 3.5.1, 1^{er} juill. 1998, T 1173/97, *Produit « programme d'ordinateur » c/ IBM I* : *JOOEB* 1999, p. 609. V. R. Whaite, *Analysis* : the EPO simplifies Software Patenting : *IPQ* 1999, vol. 4, p. 487.

57. OEB, CRT 3.5.1, 23 févr. 2006, T 424/03, *Clipboard formats I c/ Microsoft*, point 5.3.

58. OEB, CRT 3.5.1, 4 févr. 1999, T 935/97, *Produit « programme d'ordinateur » II/IBM*. V. R. McFarlane, *How to Patent Software ?* : *Patent World* 1999, p. 24.

59. TGI Paris, 18 juin 2015, *SA Orange c/ SAS Free et SAS Freebox*, RG n° 14/05735. La brevetabilité d'une revendication de programme produit a été annulée. Mais, dans le même temps, le critère de l'effet technique supplémentaire a été employé afin d'examiner la brevetabilité d'une invention portant sur un support lisible par un ordinateur.

60. F. Pollaud-Dulian, *Propriété intellectuelle. La propriété industrielle*, Économica, Corpus droit privé, 2010, n° 552, p. 306.

à l'inventeur, en pratique au conseil en propriété industrielle, de fixer lui-même l'objet du droit au moment il dépose sa demande de brevet. De plus, les articles L. 613-3 et L. 613-4 du CPI délimitent de propriété de manière négative en énumérant les actes auquel le droit permet de faire obstacle. Selon ces dispositions, le droit de brevet confère à son titulaire la possibilité d'interdire à tout tiers, en l'absence de son consentement : la fabrication, l'offre, la mise dans le commerce, l'utilisation ou bien l'importation ou la détention à ces fins du produit objet du brevet ; l'utilisation d'un procédé objet du brevet ou, lorsque le tiers sait ou lorsque les circonstances rendent évident que l'utilisation du procédé est interdite sans le consentement du propriétaire du brevet, l'offre de son utilisation sur le territoire français ; l'offre, la mise dans le commerce ou l'utilisation ou bien l'importation ou la détention à ces fins du produit obtenu directement par le procédé objet du brevet ; la livraison ou l'offre de livraison sur le territoire français à une personne autre que celle habilitée à exploiter l'invention brevetée, des moyens de mise en œuvre de cette invention se rapportant à un élément essentiel de celle-ci lorsque le tiers sait ou lorsque les circonstances rendent évident que ces moyens sont aptes et destinés à cette mise en œuvre. Il est à noter que la loi évoque un « droit d'interdire » et donne ainsi une conception négative de ce qu'est le droit de brevet. Pourtant, il s'agit bien d'un droit positif, d'une forme de propriété. D'ailleurs, le délit de contrefaçon s'analyse précisément comme l'atteinte illégitime portée aux prérogatives du breveté. C'est une atteinte au droit de propriété portant sur la chose incorporelle de l'invention.

Cependant, l'article 27 de la Convention sur le brevet communautaire, signée en 1975 à Luxembourg, exemptait plusieurs types d'actes de la qualification de contrefaçon. Malgré l'absence d'entrée en vigueur de la Convention, la plupart des pays européens ont intégré ces exemptions dans leurs droits positifs respectifs⁶¹. Aux États-Unis, l'article 271(e)(1) du Titre 35 du Code des États-Unis prévoit des exemptions du même genre⁶². Cette limitation du droit du breveté concerne d'une part des actes réalisés à titre privé sans finalité commerciale et d'autre part des actes réalisés à titre expérimental. En France, elle est prévue à l'article L. 613-5 du CPI.

Parmi les tolérances admises, celle relative aux actes expérimentaux est la plus souvent invoquée. Cette disposition a pour but d'éviter que l'octroi d'un droit de brevet ne soit de nature à compromettre l'évolution de la recherche et de la veille technologiques. Les actes expérimentaux autorisés doivent être circonscrits par deux critères : la finalité des actes et leur objet. En somme, il ne peut s'agir que d'actes réalisés dans le but de faire progresser la seule compréhension technique de l'invention. L'exemption n'est donc pas invocable quand les agissements en cause visent directement ou indirectement la commercialisation d'une chose incorporant l'enseignement technique protégé⁶³. A contrario, les agissements n'ayant pas une finalité commerciale immédiate sont permis.

L'exemption destinée aux actes d'expérimentation couvre-t-elle l'ingénierie inverse, notamment la décompilation de programmes d'ordinateur ? Cette traduction du code objet en code source exige nécessairement une reproduction partielle du programme, dont l'enseignement technique est protégé par un brevet et le code source par un droit d'auteur. C'est pourquoi l'atteinte au droit qu'implique la décompilation en matière de droit d'auteur a incité le législateur à prévoir une disposition destinée à l'autoriser sous certaines conditions limitatives. Mais elle n'a pas son équivalent en droit des brevets et il n'a donc pas été nécessaire de prévoir une dérogation légale spécifique. L'on peut donc estimer que l'exemption destinée à la réalisation d'actes expérimentaux peut couvrir des actes de décompilation et d'ingénierie inverse de manière plus générale. Mais dès lors que cette décompilation débouchera sur une réutilisation commerciale de ses résultats, l'on en reviendra à la qualification de contrefaçon. En effet, l'exemption relative aux expérimentations ne s'applique plus quand les essais ont pour objectif la commercialisation ultérieure d'un produit ou l'offre de l'utilisation d'un procédé. L'exercice du droit de brevet pour s'opposer à l'ingénierie inverse peut donc rester un moyen efficace, malgré l'exemption de recherche. Par conséquent, l'admission de la brevetabilité des inventions informatiques conduit à se poser dans le champ du droit des brevets la question de savoir dans quelles conditions et limites faudrait-il favoriser une certaine liberté d'ingénierie inverse, comme en droit d'auteur, afin d'encourager la recherche et le développement au-delà de la simple et restreinte exemption pour expérimentation ?

Permettant en principe de s'opposer à tous les actes d'expérimentation dont l'objectif est la commercialisation ultérieure d'un produit ou l'offre d'un procédé, le droit de brevet serait ainsi la protection la plus efficace pour faire obstacle au *reverse engineering* utilisé à des fins concurrentielles. Le droit sui generis des bases de données, qui ne connaît aucune restriction, pourrait aussi constituer lui aussi – lorsqu'il peut s'appliquer – un instrument efficace à cette même fin.



61. W. R. Cornish, *Experimental Use of the Patented Invention In European Community States* : IIC 1998, vol. 7, p. 735. V. T. Cook, *An European Perspective as to the Extent to which Experimental Use, and Certain Other, Defences to Patent Infringement, Apply to Differing Types of Research, A Report for the Intellectual Property Institute*, July 2006, § 3.1 s., p. 15 s.

62. Traditionnellement, l'exception était conçue strictement, conformément à la définition donnée par Robinson. V. W. C. Robinson, *The Law of Patent for Useful Inventions*, Vol. 1, Little Brown and Co., 1890, § 355, p. 488. Néanmoins, faisant suite à de nombreuses propositions doctrinales à la fin des années 80, le Congrès américain a modifié la loi afin que cette exemption soit souplement entendue pour être étendue aux essais cliniques. V. R. S. Eisenberg, *Proprietary Rights and the Norms of Science in Biotechnology Research* : *Yale L. J.* 1987, vol. 97, p. 177 (1987) ; *Science Patents and the Progress of Science : Exclusive Rights and Experimental Use* : *U. Chi. L. Rev.* 1989, vol. 56, p. 1017.

63. TGI Paris, 22 nov. 1996 : *PIBD* 1997, III, p. 571.

II. Base de données et ingénierie inverse

L'ingénierie inverse peut permettre d'accéder à une base de données dont l'accès est restreint. Or, le contenu d'une base de données est protégeable par une forme de propriété intellectuelle *sui generis*⁶⁴. Ce droit subjectif se différencie du droit d'auteur tant par les exigences auxquelles sa reconnaissance est subordonnée que par son régime. La forme de la base de données demeure par ailleurs protégée par un droit d'auteur, à condition d'être originale.

La directive 91/16/CE du 11 mars 1996 prévoit un régime de protection *sui generis* destiné aux contenus des bases de données⁶⁵. Cette directive a été transposée en droit français par la loi 98-536 du 1^{er} juillet 1998. Ainsi, l'article L. 341-1 du CPI prévoit que le producteur d'une base de données est « entendu comme la personne qui prend l'initiative et le risque des investissements correspondants » et qu'il bénéficie « d'une protection du contenu de la base lorsque la constitution, la vérification ou la présentation de celui-ci atteste d'un investissement financier, matériel ou humain substantiel. Cette protection est indépendante et s'exerce sans préjudice de celles résultantes du droit d'auteur ou d'un autre droit sur une base de données ou un de ses éléments constitutifs ».

Une distinction doit être faite entre le contenant et le contenu de la base, étant précisé que l'information en tant que telle demeure en principe toujours exclue de l'appropriation. Le contenant de la base est protégeable par le droit commun du droit d'auteur. Son contenu est protégeable par un droit *sui generis*. Selon l'article L. 112-3 du CPI, la base de données constitue « un recueil d'œuvres, de données, ou d'autres éléments indépendants, disposés de manière systématique ou méthodique, et individuellement accessible par des moyens électroniques ou par tout autre moyen ». Cette définition révèle que la protection est subordonnée à un travail d'arrangement au terme duquel les données sont disposées de manière systématique et sont individuellement accessibles. En d'autres termes : la juxtaposition de données est insuffisante. Sinon la simple compilation d'informations pourrait être appropriable. Tout débat sur les conditions de forme et d'originalité, que connaît le droit d'auteur, est donc vain en l'occurrence. L'on enferme uniquement ce droit *sui generis* dans des limites raisonnables autorisant une conciliation entre la rémunération de l'investissement, d'une part, et le principe de l'exclusion de l'appropriation des idées abstraites, d'autre part. Ainsi, la protection du contenu d'une base de données est liée à l'existence d'un investissement⁶⁶. Dès lors, les décisions fondées sur l'originalité de la réalisation du producteur sont à critiquer⁶⁷. Il est en ce sens regrettable que la Cour de cassation ait repris à son compte l'argument tiré de l'apport intellectuel pour justifier un droit *sui generis* sur une base annuelle⁶⁸.

Selon la directive 91/16/CE, la personne susceptible de bénéficier de ce droit *sui generis* est le « fabricant »

de la base de données. Ce mot constitue la traduction malheureuse du terme « maker » de la version anglaise. Selon l'article L. 341-1 du CPI, il s'agit d'un producteur, c'est-à-dire d'une « personne qui prend l'initiative et le risque des investissements correspondants ». Le texte ne concerne pas les intermédiaires, comme le sous-traitant par exemple. Il s'agit de récompenser une prise de risque dans la réalisation d'investissements. Ce risque consistant dans le fait de prendre principalement en charge l'investissement ou dans le fait d'avoir à répondre des dommages que l'utilisation de la base pourrait occasionner⁶⁹. C'est un investissement substantiel financier, matériel ou humain⁷⁰.

L'article L. 342-2 du CPI indique que le producteur peut interdire l'extraction d'une base de données, par « transfert permanent ou temporaire de la totalité ou d'une partie qualitativement ou quantitativement substantielle du contenu d'une base ». L'extraction est définie par le texte comme le « transfert [...] sur un

64. P. Catala, Au-delà du droit d'auteur, in *Écrits en hommage à Jean Foyer. Auteur et législateur*. Leges tulit, jura docuit, PUF, 1997, p. 215, v. spéc. p. 217 s. ; D. de Saint-Afrique, Le droit *sui generis* sur les bases de données : *Comm. com. électr.* 2004, chron. n° 5. La qualification de « propriété intellectuelle » nous paraît néanmoins inappropriée. En réalité, nous sommes certes en présence d'une propriété incorporelle, mais pas d'une propriété intellectuelle. La chose incorporelle de la base de données ne bénéficie pas du « supplément d'âme » caractérisant une chose intellectuelle. V. sur le concept de propriété intellectuelle J.-C. Galloux, Qu'est-ce que la propriété intellectuelle ?, in *Droits de la propriété intellectuelle. Liber amicorum Georges Bonet*, Litec, IRPI, t. 36, 2010, p. 199, v. spéc. p. 206.

65. Directive 96/9/CE du 11 mars 1996 : *JOCE* 27 mars 1996, n° L77, p. 20. V. G. Koumantos, Les bases de données dans la directive communautaire : *RIDA* janv. 1997, p. 79 ; F. Pollaud-Dulian, Brèves remarques sur la directive du 11 mars 1996 concernant la protection juridique des bases de données : *D. Aff.* 1996, p. 541.

66. B. Warusfel, La protection des bases de données en question : un autre débat sur la propriété intellectuelle européenne : *Propr. intell.* 2004, n° 13, p. 896.

67. CA Paris, 21 nov. 2008 : *RIDA* juill. 2009, p. 516 et p. 359, obs. P. Sirinelli ; *Propr. intell.* 2009, n° 31, p. 175, obs. A. Lucas ; *Propr. industr.* 2009, comm. 30, note J. Larrieu (rejet de la protection d'un site rassemblant des annonces de parents à la recherche d'un service de garde pour leurs enfants et des annonces d'assistantes maternelles).

68. Cass. com., 23 mars 2010 : *RIDA* mars 2010, p. 373, obs. P. Sirinelli ; *Propr. intell.* 2010, n° 36, p. 856, obs. J.-M. Bruguière ; *Propr. industr.* 2010, comm. 43, note N. Bouche ; *RLDI* 2010, p. 2030, note Castets-Renard (protection de la base de données constituée à partir des informations résultant de l'annuaire et enrichies par France Télécom).

69. CA Paris, 20 févr. 2004 : *Propr. industr.* 2004, comm. 57, note J. Schmidt-Szalewski (protection de l'annuaire de l'Association des ingénieurs de l'École nationale des télécommunications, qui assume les risques de ses investissements dans l'annuaire des anciens élèves, car en présence d'erreurs ou inexactitudes, ses membres cesseraient de payer leurs cotisations « qui contribuent à la faire vivre »). V. TGI Paris, 20 juin 2007 : *RTD com.* 2008, p. 93, note F. Pollaud-Dulian (protection de la base de données Infocentre PMU, parce que le PMU assume les risques des investissements « que ce soit en terme de bénéfice ou de déficit »).

70. CA Paris, 12 sept. 2001 : *JCP G* 2002, II, 10000, note F. Pollaud-Dulian : *RIDA* févr. 2002, p. 433, obs. A. Kéréver (protection d'un catalogue regroupant selon un mode particulier les informations afférentes aux exposants de différentes manifestations organisées).

autre support ». Un rapprochement est susceptible d'être fait avec le droit de reproduction connu en droit d'auteur⁷¹. Une différence avec le droit d'auteur est cependant à signaler. Tandis que toute fixation de l'œuvre constitue une reproduction contrefaisante, l'extraction n'est contrefaisante que si elle est réalisée à partir de la base produite par le titulaire du droit. Il n'existe pas d'extraction lorsque l'accès à la base de données résulte simplement du résultat affiché par un moteur de recherche. Jadis, le TGI de Paris s'était prononcé en sens contraire⁷². Dix ans plus tard, il s'est rallié à la position de la cour d'appel de Paris⁷³. Néanmoins, la Cour de justice a opté pour définition large de l'extraction en s'appuyant sur la précision selon laquelle le transfert peut être réalisé « par quelques moyens ou sous quelque forme que ce soit », ainsi que sur la finalité du texte, pour juger que l'extraction s'entendait de « tout acte consistant [...] à s'approprier [...], sans le consentement de la personne qui a constitué la base de données, les résultats de son investissement, privant ainsi cette dernière de revenus censés lui permettre d'amortir le coût de cet investissement »⁷⁴. Dès lors, l'objectif de l'extraction, commercial ou non commercial, n'influence pas la qualification de contrefaçon⁷⁵.

Le producteur est aussi en position d'interdire la réutilisation, « par la mise à disposition du public de la totalité ou d'une partie qualitativement ou quantitativement substantielle du contenu de la base ». Cette notion de réutilisation est proche de celles de droit de représentation, que connaît le droit d'auteur, et du droit de communication au public, que connaissent les droits voisins. Mais elle tend toutefois à se différencier de ceux-ci en incluant la mise à disposition par le biais d'un support tangible, lequel correspondrait plutôt à un droit de distribution. Ce caractère équivoque de la réutilisation conduit certains auteurs à considérer qu'il aurait mieux valu, comme en droit allemand, reconnaître cumulativement un droit de distribution et un droit de communication au public⁷⁶. De son côté, la Cour de justice a retenu une approche extensive de la notion de réutilisation en considérant qu'il s'agissait de « tout acte consistant [...] à mettre à la disposition du public, sans le consentement de la personne qui a constitué la base de données, les résultats de son investissement »⁷⁷.

L'ingénierie inverse, en ce qu'elle permet d'extraire ou de réutiliser le contenu d'une base de données est susceptible de constituer un acte de contrefaçon. D'autant plus que la finalité commerciale ou non commerciale n'importe pas. Le droit des bases de données ne prévoit aucune exception telle que celles prévues par le droit d'auteur, par le droit des brevets ou encore par le droit des topographies de semi-conducteurs. En l'absence de disposition spécifique, nous sommes amenés à penser que le droit sui generis relatif aux bases de données fait nécessairement obstacle à toute ingénierie inverse qui permettrait d'accéder à ces dernières. Afin d'éviter cette limitation excessive de la liberté de la recherche, il conviendrait de prévoir là aussi une disposition spécifique destinée au *reverse engineering* sous certaines conditions, telle que celle connue en droit d'auteur.

L'objectif du *reverse engineering* est la compréhension d'un objet par son analyse soit pour le reproduire servilement soit pour créer un nouvel objet. En tout cas, il vise l'obtention d'un enseignement susceptible d'être réutilisé pour constituer une nouvelle chose intellectuelle appropriable. Ce phénomène est en particulier visible dans le secteur des technologies de l'information. Ainsi, en droit d'auteur, le législateur a notamment instauré une exception spécifique pour la décompilation d'un code objet, qui constitue une partie de l'ingénierie inverse d'un programme d'ordinateur. Observée sous le prisme de la propriété intellectuelle, le *reverse engineering* fait surtout ressurgir la spécificité de ce type de propriété : il s'agit d'encourager la création ou l'innovation au sens strict. Cette spécificité distingue la propriété intellectuelle de la propriété corporelle, mais aussi d'autres formes de propriétés incorporelles. La valeur d'une chose intellectuelle est nettement délimitée par la loi, grâce aux conditions qu'elle pose, parce que dans les secteurs mentionnés la propriété limite la liberté de la création ou la liberté de la recherche. Dès lors, l'ingénierie inverse pourrait être vue comme un instrument qui, sous certaines conditions de loyauté et de respect du droit de la concurrence, constitue une limitation justifiée aux effets monopolistiques du droit des brevets et des bases de données. Il conviendrait donc à notre sens de prêter davantage d'attention à cette pratique créatrice qui – bien encadrée pour en éviter les abus – peut permettre d'éviter que la propriété intellectuelle ne puisse, dans certains domaines, devenir un frein à l'innovation.



71. J.-H. Reichman et P. Samuelson, *Intellectual Property Rights in Data ?* : *Vanderbilt L. Rev.* 1997, vol. 50, p. 51.

72. TGI Paris, ordonnance du 8 janvier 2001, infirmée par CA Paris, 25 mai 2001 : *PIBD* 2001, III, p. 455 ; *Comm. com. électr.* 2001, comm. n° 80, note C. Le Stanc (rejet de la protection du contenu d'un site d'offre d'emploi fonctionnant via un moteur de recherche). En l'espèce, il a été jugé que l'extraction n'était pas quantitativement ou qualitativement substantielle, parce que l'appelant, avec son moteur de recherches, ne téléchargeait pas la base de données de l'intimée afin d'alimenter son propre système, mais qu'il procédait à des interrogations ponctuelles sur le site de l'intimée en fonction de critères limités.

73. TGI Paris, 1^{er} févr. 2011 : *JCP E* 2011, 1417, note J. Larrieu ; *Prop. industr.* 2011, comm. 44, note J. Larrieu (rejet de la contrefaçon d'une base de données immobilières dont des éléments sont mis à disposition sur site internet via un moteur de recherche).

74. CJCE, 9 nov. 2004, aff. C-203/02, *The British Horseracing Board Ltd. et autres c/ William Hill Organization Ltd.* : *Comm. com. électr.* 2002, comm. 16, note C. Caron (bases de données sportives), point 51.

75. CJCE, 5 mars 2009, aff. C-545/07, *Apis-Hristovich c/ Lakorda AD* : *RIDA* juill. 2009, p. 415, obs. P. Sirinelli ; *Prop. intell.* 2009, n° 33, p. 383, note V.-L. Benabou ; *Prop. industr.* 2009, comm. 48, note J. Larrieu, point 52 (base de données juridique). La durée du transfert, caractéristique de l'extraction de données, est sans conséquence juridique. Le droit sui generis confère au producteur d'une base de données le droit d'interdire sa visualisation puisque celle-ci entraîne automatiquement un transfert de données.

76. A. Lucas, H.-J. Lucas et A. Lucas-Schloetter, *Traité de la propriété littéraire et artistique*, Litec, Traité, 4^e éd. 2012, n° 1191, p. 980.

77. CJCE, 9 nov. 2004, aff. C-203/02, *The British Horseracing Board Ltd. et autres v. William Hill Organization Ltd.*, loc. cit.

Le projet de directive relatif à la protection des secrets d'affaires prévoit d'ailleurs en ce sens une exception, à son article 4, au profit de la décompilation. Or, cette exception présente la particularité de ne pas être restreinte par la limite de l'interopérabilité que connaît le droit d'auteur. Une telle étendue – qui sera sans doute considérée comme excessive par les défenseurs des secrets d'affaires – ramène finalement à la question qui sous-tend toute la problématique de l'ingénierie inverse, à savoir : comment établir la juste balance des intérêts entre liberté de la recherche et respect de la réservation privative ? Dans quelle mesure le *reverse engineering* est-il légitime face à une propriété ou des droits privatifs auxquels il porte atteinte ? L'on perçoit aussi aisément que la question dépasse la sphère de la propriété pour s'étendre aux actes parasitaires, qui pourraient être couverts par une autorisation plus large de l'ingénierie inverse. Cela touche donc au-delà des aspects techniques de sa mise en œuvre, aux questions-clés du droit de l'information et particulièrement de son application dans le contexte numérique. En attendant d'éventuelles évolutions légales ou jurisprudentielles sur ce terrain, c'est au contrat qu'il revient sans doute d'encadrer cette pratique, puisque dans de nombreux cas (et notamment s'agissant de contrats de prestation de service), des clauses limitant ou interdisant certaines formes d'ingénierie inverse pourraient être considérées comme licites.

