

Analyse de l'accord politique du Conseil sur la brevetabilité des logiciels
(proposition de directive 2002/47/COD sur la brevetabilité des inventions mises
en oeuvre par ordinateur)

Contexte :

Cette analyse est produite sur la base de ce qui est aujourd'hui connu de l'accord politique adoptée à la majorité qualifiée lors du Conseil Compétitivité le 18 mai 2004. On trouvera à la fin de ce texte une analyse détaillée d'en quoi cet accord diffère du texte soumis par la présidence irlandaise au Comité des Représentants Permanents du 5 mai 2004. Ces différences peuvent être résumées ainsi : quelques rideaux de fumée supplémentaires pour cacher le fait que le texte ouvre grand la porte à la brevetabilité des idées sous-jacentes de logiciels et des méthodes de traitement de l'information, et pour justifier le revirement des pays qui s'étaient engagés à ne pas voter le texte irlandais, notamment l'Allemagne. Les pays n'ayant pas voté le texte au Conseil sont l'Espagne (vote contre), la Belgique, l'Italie, l'Autriche et la Pologne (qui a fortement réaffirmé son opposition au texte par la suite). La France, la Slovénie, et la Hongrie ont voté pour le texte, mais en insistant sur la nécessité de le clarifier d'ici l'adoption de la position commune formelle exprimant le résultat de la première lecture pour le Conseil.

Analyse :

Le texte proposé ouvre à mon avis encore plus la porte à la brevetabilité que la proposition originale de la Commission. Il l'empire sur 2 points :

- Il atteint des niveaux supérieurs d'hypocrisie et d'abstraction. Certains « considérant » sont écrit de façon à suggérer qu'il existerait une intention de restreindre la brevetabilité, alors que toutes les dispositions légales des articles officialisent la pratique récente de l'OEB et échouent à installer des obstacles solides à la monopolisation des algorithmes et des méthodes de traitement de l'information.. Certaines dispositions atteignent des nouveaux sommets de Novlangue¹ juridique.
- La proposition autorise les revendications logicielles (article 5b) alors qu'elles sont considérées par tous les analystes sérieux comme équivalentes à la brevetabilité des logiciels en tant que tels. J'ai personnellement entendu des représentants de haut niveau de l'OEB affirmer à répétition que la raison pour laquelle il fallait accepter les revendications logicielles était de rendre possible pour les détenteurs de brevets de poursuivre les créateurs, copieurs, distributeurs et éditeurs de logiciels.

Les deux points ci-dessus peuvent être appréciés lorsqu'on considère d'une part le nouvel article 4a (ancien considérant 7b) qui déclare :

(4anew) A computer program as such, in particular the expression of a computer program in source code or object code or in any other form, cannot constitute a patentable invention

¹Une langue fictive (enfin, on le croyait) introduite par Orwell dans 1984, et dont le but est de rendre impossible d'avoir des idées non-conformes en vidant les mots de tout sens.

et d'autre part l'article 5b :

(5b) A claim to a computer program, either on its own or in a carrier, shall not be allowed unless that program would, when loaded and executed in a computer, programmed computer network or other programmable apparatus, put into force a product or process claimed in the same patent application in accordance with paragraph 1.

En laissant de côté la négation retorse de la première phrase de l'article 5b, on peut résumer la situation qui résulterait de ce texte par : on prétendra que les logiciels, algorithmes et méthodes de traitement de données ne sont pas brevetables, mais on les brevetera, et on donnera de puissantes armes aux détenteurs des brevets correspondants pour anéantir les développeurs et distributeurs de logiciels qui ne leur plaisent pas.

Malheureusement, il ne s'agit pas des seules dispositions qui visent à officialiser le pillage du domaine public des idées et de l'information. La proposition a réintroduit (ignorant 43 amendements du Parlement) les pires dispositions de la proposition d'origine de la Commission :

- Le considérant 6 affirme que les inventions mises en oeuvre par ordinateur appartiennent à un domaine technique et donc relèvent de l'article 27.1 d'ADPIC et des autres dispositions d'ADPIC sur les droits conférés par les brevets. Puisqu'une invention mise en oeuvre par ordinateur est définie dans l'article 2a comme toute chose dont la réalisation implique un ordinateur, ... comme le fait d'exécuter un logiciel sur un ordinateur, cela signifie que tout logiciel appartient à un domaine technique (en contradiction avec la référence à l'article 52.2 de la Convention de Munich dans le considérant 7). C'est une affirmation extrêmement dangereuse, qui promeut une interprétation d'ADPIC (tout est dans un domaine technique pour peu que l'on puisse inventer une façon de dire que c'est une invention) qui aura des conséquences bien au-delà du domaine des logiciels. Le verbiage juridique confus du considérant 12 vise à prétendre que la notion de contribution technique peut constituer une barrière. Mais, aussi récemment que le 14 avril dernier, David Sant, représentant officiel de l'OEB auprès des institutions européennes a confirmé dans une conférence au Parlement européen l'interprétation officielle qui est si large qu'elle attribue une contribution technique à toute situation dans laquelle on peut dire qu'il faut considérer des facteurs techniques avant de commencer à programmer, situation à laquelle je n'ai jamais rencontré d'exception en 25 ans de pratique du logiciel et de la recherche informatique.

- L'article 2b permet de justifier l'attribution d'une contribution technique s'il existe une différence entre la revendication du brevet (y compris les caractéristiques non-techniques) et l'état de l'art. La limitation que l'article 4a prétend établir en rejetant la brevetabilité de méthodes qui seraient dites techniques simplement parce qu'elles utilisent un ordinateur, alors qu'elles ne produisent pas d'effets techniques au-delà des interactions physiques normales entre un programme et un ordinateur, réseau ou autre dispositif programmable est sans effet. Elle est déjà présente dans les instructions d'examen de l'OEB, et ceci n'a pas empêché l'OEB (ni les déposants) de trouver des façons toujours renouvelées d'affirmer qu'un programme a des effets au-delà des interactions physiques normales pour des raisons telles que de présenter de l'information mieux ou plus vite, ou de fonctionner plus vite, autrement dit tout ce que le logiciel essaie de faire. L'introduction d'un alinéa à l'article 4a selon lequel un programme en tant que tel n'est pas une invention brevetable est non moins hypocrite : le programme n'est soi-disant pas brevetable, mais ses idées sous-jacentes sont brevetables et il est revendicable, donc ceux qui écrivent des programmes ou les diffusent peuvent être accusés de contrefaçon.

- L'article 6 est supposé protéger l'interopérabilité mais y échoue complètement. Il maintient seulement les exceptions de rétro-ingénierie et de décompilation prévue par la directive de 1991 sur la protection du logiciel par le droit d'auteur. Mais alors que ces exceptions sont suffisantes dans le contexte du droit d'auteur, dans le contexte des brevets, il resterait impossible de mettre en oeuvre l'interopérabilité, puisque le faire serait une contrefaçon de brevet. Seule une exception explicite aux droits conférés par les brevets, (qui est pleinement autorisée par l'article 30 d'ADPIC) protégerait l'interopérabilité. L'amendement voté par le parlement établissait cette exception en excluant l'interopérabilité du champ de la contrefaçon possible. Un nouveau considérant affirme que les sociétés en position dominante devront donner accès à leurs technologies d'interopérabilité, mais sans créer de licence obligatoire gratuite et non-discriminatoire. Il sera non seulement inefficace, mais profondément nuisible à l'égard des logiciels libres, puisqu'il permettra aux détenteurs de brevets (rendus valides par les autres dispositions) d'exiger paiement de redevances sur les technologies d'interopérabilité, ce qui incompatible avec le principe même des logiciels libres.

Un seul des 4 points (revendications logicielles, faire tomber tout le logiciel dans le champ d'ADPIC, définir de façon laxiste la technicité, et l'absence de protection de l'interopérabilité) serait suffisante pour rejeter l'ensemble du texte.

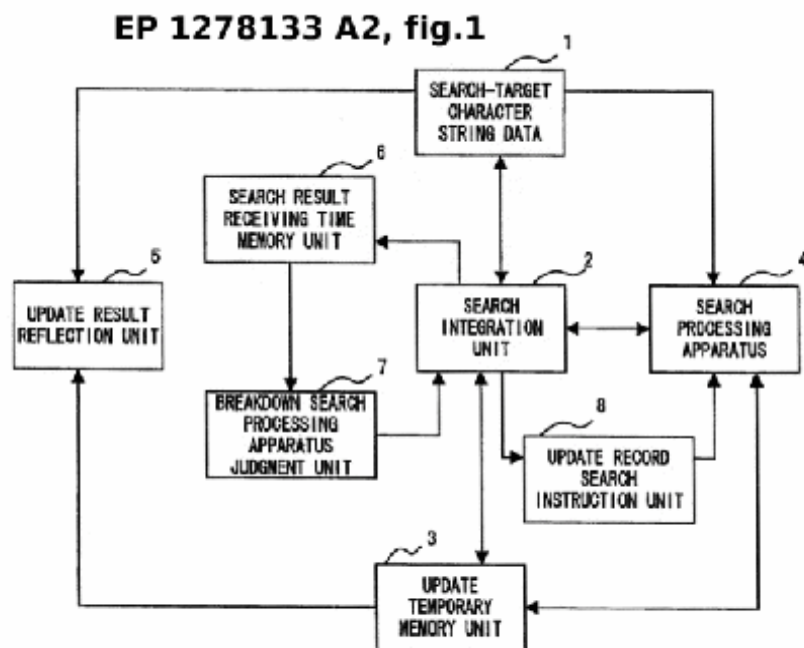


Illustration 1 : Un exemple d'illustration dans un brevet de l'OEB ou "comment donner une contribution technique à un algorithme basique (ici de recherche full-text) »

Changements introduits lors du Conseil du 18 mai 2004

Analyse basée sur un information fragmentaire, car le texte résultant n'a pas été communiqué.

- Le considérant 13 a été supprimé. Ce considérant disait explicitement qu'un algorithme est brevetable. Malheureusement, maintenant on ne le dit plus, mais les algorithmes sont toujours brevetables en tant que « solutions techniques dans le domaine du traitement de l'information », puisque l'exclusion précise des algorithmes et méthodes de traitement de l'information proposée par les amendements 69 et 55 du parlement et que l'Allemagne avait proposé de réintroduire dans l'article 2b n'est pas acceptée. Bref un texte plus confus, et tout

aussi nuisible.

- Le mot « new » a été introduit dans la définition de la contribution technique à l'article 2b. Mais cela ne change rien à la possibilité d'arguer de nouveauté dans l'ensemble de la revendication (caractéristiques non-techniques incluses). On n'a toujours pas imposé que la nouveauté réside dans les caractéristiques techniques proprement dites, et que celles-ci soient identifiées par la mise en oeuvre de moyens physiques pour obtenir des effets physiques. Si vraiment on voulait breveter des machines à laver et des systèmes de freinage incluant des logiciels, en quoi cela serait-il gênant? Mais on veut breveter le traitement, la présentation et l'échange de l'information.

- Un alinéa supplémentaire a été introduit comme article 4a(new) pour affirmer (ce qui n'était auparavant que dans les considérants) qu'un logiciel en tant que tel n'est pas brevetable que ce soit comme source, en binaire ou sous toute autre forme. Evidemment puisque sinon ce serait dire noir sur blanc que l'on s'assoit sur la Convention de Munich, ce qui est illégal pour un texte européen. Mais malheureusement, cela revient exactement au même puisqu'un logiciel devient « une solution technique » dès qu'il faut prendre en compte des considérations techniques pour le programmer.

- Enfin un considérant a été introduit pour affirmer que les sociétés en position dominante (suivez mon regard) devraient donner accès à l'information nécessaire pour l'interopérabilité, mais sans préciser autant qu'on le sache ni comment cela sera décidé (il faudrait pour cela que ce soit dans un article) ni que ce devra être avec une licence gratuite et non discriminatoire (par exemple à l'égard des licences comme la GPL).