Institutional Repository - Research Portal
Dépôt Institutionnel - Portail de la Recherche
researchportal.unamur.be
University of Namur

RESEARCH OUTPUTS / RÉSULTATS DE RECHERCHE

L'application de la directive communautaire du 25 juillet 1985 (responsabilité du fait des produits) au domaine du logiciel

Triaille, Jean-Paul

Published in:
Revue générale des assurances et des responsabilités

Publication date:
1990

Document Version
le PDF de l'éditeur

Link to publication

Citation for pulished version (HARVARD):
Triaille, J-P 1990, 'L'application de la directive communautaire du 25 juillet 1985 (responsabilité du fait des produits) au domaine du logiciel', *Revue générale des assurances et des responsabilités*, p. 11617.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 19. Apr. 2024

6.8071. F

BIBLIOTHEQUE
DROIT - NAMUR

L'APPLICATION DE LA DIRECTIVE COMMUNAUTAIRE DU 25 JUILLET 1985 (RESPONSABILITÉ DU FAIT DES PRODUITS) AU DOMAINE DU LOGICIEL (*)

par Jean-Paul TRIAILLE
C.R.I.D. Faculté de Droit (Namur)

PLAN

INTRODUCTION

1ère PARTIE : L'APPLICABILITÉ DE LA DIRECTIVE AUX LOGICIELS
«La directive est-elle applicable aux programmes d'ordinateur ?»

1ère OBJECTION
«La directive ne s'applique pas aux logiciels parce que le logiciel est un service et non un produit».

2ème OBJECTION
«La directive ne s'applique pas aux logiciels parce qu'un logiciel est un bien immatériel; or la directive ne vise que les biens matériels».

Réponse A : «La directive ne requiert pas que le bien soit matériel; et donc rien n'empêche de l'appliquer aux logiciels»

Réponse B : «La directive ne vise que les biens matériels, mais le logiciel est un bien corporel. Dès lors, la directive s'y applique»

Résumé
Avis de la Commission des communautés

2ème PARTIE : L'APPLICATION DE LA DIRECTIVE AUX LOGICIELS
«Quand la directive s'appliquera-t-elle ?»

I. La condition de défectuosité
II. Le dommage
III. Le lien de causalité
IV. La détermination du producteur responsable
V. La date de mise en circulation

QUELQUES MOTS DE CONCLUSION

La question de l'application de la directive européenne relative à la responsabilité du fait des produits défectueux aux logiciels est, à ce jour, fortement controversée.

L'importance de la réponse s'explique par les modifications qu'introduit la directive dans le champ du droit de la responsabilité du fait des produits en Europe, soit principalement :

- instauration d'une responsabilité sans faute (objective) à charge du producteur en cas de défaut du produit;
- interdiction des clauses limitatives ou exonératoires de responsabilité.

(*) Cet article reprend le texte d'un rapport présenté lors d'une journée d'études organisée par Euroforum à Bruxelles en octobre 1989 et consacrée à «La responsabilité du fait des produits».

Pour commencer cette étude, on peut citer quelques exemples de dommages qui ont été causés par des logiciels défectueux :

– Récemment, aux Etats-Unis, un patient devait subir un traitement aux rayons pour soigner une tumeur à l'épaule. L'appareil médical, actionné par un logiciel, a envoyé, en une seule fois, une dose de rayons beaucoup trop importante, qui s'est avérée mortelle; et l'accident s'est produit dans plusieurs hôpitaux aux Etats-Unis. On a alors découvert que c'est le logiciel qui était défectueux.

– En 1984, un ouvrier de Détroit a été tué par un robot dans les usines Ford. Ce genre d'accident s'est déjà produit plusieurs fois au Japon (avec à l'origine un défaut du logiciel).

– On a dit qu'une erreur de programmation avait failli déclencher une alerte nucléaire il y a quelques années... mais on peut trouver des exemples plus amusants : ainsi, en 1988 à Londres, où suite à une erreur dans un programme d'ordinateur, l'Administration des contributions a envoyé des déclarations fiscales à un million de personnes âgées qui ne devaient plus en remplir depuis des années. Or, la loi prévoit qu'une déclaration reçue doit être remplie et qu'une déclaration remplie doit être traitée par l'Administration...

Ces exemples ne relèvent certes pas tous du domaine de la directive; mais ils sont cités pour indiquer la diversité des situations et des dommages possibles quand un programme d'ordinateur se met à mal fonctionner.

1re PARTIE : L'APPLICABILITÉ DE LA DIRECTIVE AUX LOGICIELS «LA DIRECTIVE EST-ELLE APPLICABLE AUX PROGRAMMES D'ORDINATEUR ?»

Avant d'analyser *quand* (et dans quels cas) la directive s'appliquera aux logiciels, il faut d'abord régler une question préalable, c'est-à-dire voir *si* la directive s'y applique. On va voir en effet qu'une série d'objections ont été soulevées quant à l'application de la directive. Pour rester bref, on va mentionner rapidement ces quelques objections et indiquer pourquoi elles ne peuvent être retenues. On ne peut faire l'économie de l'examen de ces questions, car la réponse ne fait pas encore l'unanimité; or, il faut régler ces premiers obstacles (qu'on a appelé la «première génération de questions») avant de passer à l'étape suivante qui est l'analyse des conditions d'application du texte communautaire.

1re OBJECTION : «La directive ne s'applique pas aux logiciels parce que le logiciel est un service et non un produit»

C'est vrai que la directive ne s'applique pas aux services (pour lesquels la Commission a l'intention de proposer d'autres textes).

Mais même s'il est vrai que la fourniture de logiciels – et spécialement la production d'un logiciel sur mesure – est un service, le logiciel lui-même est un produit. De la même façon, un restaurant fournit un service aux clients, mais la nourriture que l'on y «sert» est un produit; et le simple fait pour un produit d'être fourni dans le contexte d'un service ne le soustrait pas du champ d'application de la directive.

On a parfois dit qu'il fallait distinguer entre les progiciels (logiciels standard vendus en grande série) et les logiciels écrits sur mesure pour les besoins spécifiques d'une entreprise, et que seuls les premiers devraient être considérés comme des produits. Or, il me semble que même si dans le cas des logiciels sur mesure, il y a eu effectivement un service rendu, avec une relation de personne à personne, cela n'empêche pas que l'objet même du service était la création d'un produit (qui a d'ailleurs les mêmes caractéristiques qu'un logiciel standard).

La situation est la même quand, au lieu d'acheter une voiture de série, l'acheteur passe commande d'une série de gadgets ou d'ajouts particuliers. Dans ce cas, comme pour un logiciel sur mesure, il y a un service qui aboutit à la création d'un «produit» dans le sens de la directive.

De plus, cette distinction logiciel standard/logiciel sur mesure se heurte à deux autres difficultés :

– beaucoup de programmes sont standard pour une grande partie, et adaptés aux besoins de l'utilisateur pour le reste; comment va-t-on les classer, comme produit ou comme service ? La réponse serait dans ce cas très importante pour décider si on a une responsabilité sans faute ou non;

– de plus, ce qui est fait sur mesure aujourd'hui peut être, si le fabricant le souhaite, produit en série demain : encore une fois, comment ferait-on la distinction ?

Enfin, il n'est de toute façon pas exact de prétendre que la directive ne s'applique qu'aux biens produits en série; même si l'exposé des motifs mentionne cette exigence, l'article 2 ne la reprend pas. Au Parlement européen, on avait suggéré d'exclure expressément les produits «artisanaux»; cela n'a pas été fait, et la Commission a elle même confirmé que les produits artisanaux étaient couverts par la directive (ainsi qu'on va le voir plus loin).

En résumé, on peut dire que la thèse selon laquelle le logiciel serait un service et non un produit ne peut être retenue. Pas davantage on ne peut isoler une catégorie de logiciels (ceux qui sont faits «sur mesure») pour ensuite les exclure du champ de la directive, puisqu'en effet, il y a d'abord des objections d'ordre matériel (où tracer la limite ?) et qu'en plus la directive ne prévoit pas un traitement différencié entre les biens faisant l'objet d'une production industrielle et les autres biens (produits de façon «artisanale»).

2e OBJECTION : «La directive ne s'applique pas aux logiciels, parce qu'un logiciel est un bien immatériel : or, la directive ne vise que les biens matériels»

Il y a deux façons de critiquer cet argument :

– soit (A) en soutenant que la directive s'applique également aux biens matériels (et donc au logiciel) (il ne nous semble pas que ce raisonnement soit exact);

– soit (B) en montrant que le logiciel n'est pas quelque chose d'immatériel, et que donc rien n'empêche l'application de la directive.

A. Premièrement, on peut dire : «La directive ne requiert pas le bien soit matériel, et donc rien n'empêche de l'appliquer au logiciel»

La question est donc de savoir si la directive vise aussi les biens intangibles; comme le texte n'est pas explicite à ce sujet, on a essayé de tirer argument de la terminologie, en comparant les différentes traductions. Il ne semble pas qu'une réponse claire se dégage de cette analyse.

On peut aussi essayer d'interpréter le texte de la directive d'une façon «téléologique» (conforme à son but). Mais quel est alors ce but ? On peut facilement démontrer que le but réel de la directive est d'améliorer la protection des consommateurs. Il est assez clair que les autres motifs invoqués (distortion de concurrence et effet sur la libre circulation des marchandises) sont plus formels que réels. Si tel est le cas (si le but de la directive est d'améliorer la protection des consommateurs), on pourrait dire : «En cas de doute, choisissons la voie la plus favorable aux consommateurs»; et dès lors, pour la question des biens immatériels, «élargissons le champ de la directive pour les y inclure, dans l'intérêt des consommateurs».

Cependant, à plusieurs reprises, l'exposé des motifs parle d'une «juste répartition des risques» entre la victime et le producteur. Dès lors, si la directive essaie d'établir un tel équilibre, il ne semble pas légitime de faire pencher la balance d'un côté plutôt que de l'autre; sinon cela reviendrait à donner à la directive un caractère unilatéral qu'en tous cas elle prétend et souhaite ne pas avoir.

Dès lors, ni la terminologie (dans les différentes versions), ni une éventuelle interprétation ne donne de réponse claire à la question de l'application de la directive aux biens immatériels.

Pour ma part, je ne pense pas que la directive s'applique aux biens immatériels, pour deux raisons : l'une tirée de l'analyse du texte (1o) et l'autre qui est davantage un argument d'opportunité (2o).

1o) Premièrement, à la lecture du texte, il apparaît clairement, qu'à aucun moment on n'a pensé aux biens intangibles. En effet, beaucoup de dispositions poseraient problème si on les appliquait aux biens immatériels :
- «matière première» d'une chose intangible
- «partie composante»

- «fabrication» (en anglais : manufacturer)
- «importateur»
- comment apposer son nom ou sa marque sur un bien immatériel ?
- qu'entendrait-on par sa «présentation» ?
- quand est-il «mis en circulation» ?
- quid de sa conception (design) ?
- quid du dommage «causé à une chose autre que le produit défectueux lui-même» ?

Cela montre bien que les rédacteurs n'ont jamais envisagé les biens intangibles.

2°) Par ailleurs – et j'en viens à l'argument d'opportunité –, si on vise les biens immatériels, où s'arrête-t-on ? On vise alors, entre autres choses, l'information. On y reviendra plus loin, mais est-il bien opportun d'instaurer une responsabilité sans faute en cas d'informations «défectueuses» ?

Est-ce à dire que la directive ne s'applique pas aux logiciels ? Non, parce qu'on peut tenir un raisonnement différent.

B. «Il est vrai que la directive ne vise que les biens corporels, mais le logiciel est un bien corporel. Dès lors, la directive s'y applique». Ici, l'objection ne porte plus sur le champ d'application de la directive mais sur la nature même du logiciel.

On définit souvent le logiciel comme étant une suite d'instructions données à un ordinateur et qui lui permettront d'exécuter une tâche déterminée. Le terme «instruction», à mon avis, induit en erreur, de même que quand on parle d'un «language» informatique. En réalité, ces instructions sont matérialisées sur un support (disquette, CD ROM, bande magnétique, etc...), et quand on l'introduit dans la machine, elles ont un effet matériel, concret et bien visible :
- soit, si on les introduit dans un ordinateur, celui-ci affichera des résultats à l'écran ou imprimera des données sur papier;
- soit, si on les introduit dans un robot, celui-ci sera actionné et pourra se déplacer, bouger des objets, etc...

De même à propos du language, il est vrai qu'on part, lors de la programmation, d'une étape où il y a un «texte» plus ou moins compréhensible; mais on aboutit, à la fin du processus, à une suite de O et de 1 (language binaire) qui ressemble beaucoup moins à un «texte».

En réalité, il ne faut pas comparer ces instructions à des lettres d'un texte, mais bien plutôt à la forme d'un outil ou aux caractéristiques techniques d'un instrument (comme son poids ou sa résistance). Si on voulait schématiser, on pourrait dire qu'en quelque sorte un logiciel, d'un point de vue concret, ne fait que canaliser, à travers mille ramifications, des impulsions électriques. Qu'y a-t-il d'immatériel là-dedans ? On peut dire qu'est réellement immatériel un droit sur une chose, une idée, ou une information; on voit mal pourquoi un logiciel le serait. Il semble qu'on confond ce qui est réellement immatériel (comme une information) et ce qui est seulement imperceptible pour nos sens.

Pour faire une comparaison, prétendrait-on qu'un microbe ou une bactérie sont immatériels ?

Le logiciel introduit dans une machine (comme le microbe dans l'organisme) y produit des effets matériels; inversément, ce qui est vraiment immatériel ne sait entraîner aucun changement matériel sauf (on y reviendra) si un être humain lui donne suite en décidant d'une action à effectuer (en prenant une décision concrète sur base d'une information, par exemple).

On peut se demander si cette idée selon laquelle le logiciel serait immatériel ne vient pas du fait que l'informatique et les ordinateurs sont encore parfois considérés comme quelque chose d'un peu diabolique ou magique... A cet égard, il était peut-être utile de démystifier un peu les choses.

On constate donc que le logiciel a un caractère hybride (à la fois «texte» et «outil» : ensemble d'instructions «écrites» par un humain, mais actionnant une machine); cela entraîne dès lors des malentendus. On retrouve d'ailleurs les mêmes difficultés quand on envisage la protection du logiciel par la propriété intellectuelle : en schématisant, on peut dire que les uns préconisent le droit d'auteur parce que le logiciel est un texte, et que les autres soutiennent la solution du brevet parce que le logiciel est un outil.

Plusieurs objections ont été soulevées à l'encontre du caractère matériel du logiciel :

– D'abord, le fait que le logiciel peut exister et être transféré indépendamment d'un support, par voie de télécommunication (par câble ou même par radio).

Dans ce cas, on peut dire que le programme lui-même n'est pas transféré : on en envoie seulement une copie; de cette façon, on met le programme de départ (bien matériel) à la disposition des utilisateurs, par la voie des moyens de télécommunication. La directive ne requiert pas pour son application que le bien soit physiquement transféré : toute «forme de distribution» (dans un but économique, voir article 7.c.) peut convenir. Dans ce cas («télé-software»), le logiciel de départ est effectivement mis en circulation et rendu disponible pour les utilisateurs.

– Ensuite, le fait que le logiciel ne soit pas vendu, mais donné en licence et qu'il fasse l'objet de droits de propriété intellectuelle.

Ainsi qu'on l'a dit, le mode de distribution, de même que la qualification du contrat par les parties (vente ou licence) n'a pas d'importance pour la directive. Par ailleurs, d'autres biens, comme par exemple une machine (au sens traditionnel), font l'objet de droits de propriété intellectuelle (brevet) sans qu'on ait jamais vu là une indication de leur caractère immatériel.

Résumé

Pour résumer cette première partie, on peut rappeler les différentes thèses de la façon suivante :

Les uns ont dit :
«Le logiciel est incorporel, or la directive ne vise que les biens tangibles, donc les logiciels ne sont pas visés».

D'autres on dit :
«Le logiciel est incorporel, mais la directive vise aussi ces biens immatériels; donc les logiciels sont inclus dans le champ de la directive».

Ce qu'on a essayé de montrer, c'est que «le logiciel est tangible et donc il est couvert par la directive (qui, à notre avis, ne vise que les biens matériels)».

Avis de la Commission des communautés

La commission des Communautés européennes s'est prononcée sur le sujet, dans une réponse parlementaire publiée en mai de cette année (voir bibliographie). A la question de savoir si la directive s'appliquait aux logiciels, la Commission, d'une façon très laconique, répond que «... la directive s'applique aux logiciels, comme elle s'applique d'ailleurs aux produits artisanaux et artistiques».

Cependant, elle ne précise pas à quel titre elle vise le logiciel : comme bien matériel ou comme bien immatériel ?

On aurait pu se limiter à donner cette réponse de la Commission, mais comme ses avis peuvent toujours être contestés par la suite, il nous semblait important de justifier cette réponse, ainsi que le principe (retenu par la Commission) de l'application de la directive aux logiciels.

On vient de voir que la directive s'appliquait aux logiciels; on peut dès lors passer à la deuxième partie qui concerne les conditions d'application de la directive.

2e PARTIE : L'APPLICATION DE LA DIRECTIVE AUX LOGICIELS : «QUAND LA DIRECTIVE S'APPLIQUERA-T-ELLE ?»

Cette question n'a jusqu'à présent été que fort peu étudiée par la doctrine. On va voir que beaucoup de points resteront sans réponse, dans l'attente de décisions judiciaires sur le sujet.

En admettant que la directive est applicable, pour que la norme nationale qui en reprendra les principes s'applique dans un cas concret, il faudra que la victime prouve trois choses :

I. le *défaut* du logiciel
II. un *dommage* couvert par la directive
III. un *lien de causalité*

De plus, il faudra déterminer qui est le producteur du logiciel (IV) et quand ce logiciel a été mis en circulation (V).

On va illustrer les explications par quelques exemples où un logiciel défectueux peut entraîner des dommages; au fur et à mesure, on verra ceux qui sont prévus par la directive et ceux qui sont à exclure.

1er exemple (le cas le plus fréquent où un logiciel peut causer des dommages) : une petite entreprise a acquis un logiciel qui lui permet de faire sa comptabilité, de gérer les stocks, de s'occuper de la facturation et, de façon générale, de stocker toutes les informations utiles à l'entreprise. Suite à un défaut dans le logiciel, le système est bloqué. Conséquence : désorganisation complète de l'entreprise (d'après les statistiques, dans trois cas sur quatre, l'entreprise est condamnée à plus ou moins brève échéance).

2e exemple : une entreprise a automatisé ses chaînes de montage et installé des robots qui travaillent à côté des ouvriers. Le cas s'est déjà présenté (aux Etats-Unis et au Japon) : le robot, actionné par un logiciel, devient incontrôlable et blesse grièvement un ouvrier.

3e exemple : (assez similaire au 2eme) en France, dans un hôpital, un appareil de radiographie informatisé ne répondant plus aux instructions de l'utilisateur, au lieu de se placer doucement sur la poitrine du patient, lui a enfoncé la cage thoracique.

4e exemple : Monsieur X a une cave à vin. Il la fait équiper d'un système qui en contrôle la température et l'humidité. Le système est actionné par un microprocesseur (instrument comprenant du hardware et du software). Le software est mal conçu, l'appareil se dérègle : tous les vins sont gâtés.

5e exemple : un système-expert médical (intelligence artificielle) est vendu aux médecins (et bientôt, aux Etats-Unis, aux consommateurs). Une erreur entraîne un faux diagnostic : le malade, mal soigné, subit un dommage corporel.

Revenons aux trois conditions d'application du texte : le défaut, le dommage et le lien de causalité.

I. La condition de défectuosité :

Pour reprendre les termes de l'article 6 : quand un logiciel n'a-t-il pas offert «la sécurité à laquelle on pouvait légitimement s'attendre» ? Un logiciel contenant un «bug» (une erreur) est-il automatiquement défectueux ?

Le problème est que de l'avis même des informaticiens, un programme qui ne contient aucune erreur n'existe pas ou presque : un tel programme est donc l'exception à la règle.

A quelle sécurité doit-on alors s'attendre ? Et à quelle sécurité peut-on légitimement s'attendre ? On voit que la réponse est plus une affaire d'opportunité qu'une argumentation juridique. D'une part, l'informatisation introduit dans beaucoup de domaines des progrès énormes dont tout le monde bénéficie; faut-il, pour en favoriser le développement, être assez souple en cas de défaut ? D'autre part, ses possibilités d'application sont immenses et vont s'étendre dans tous les domaines; faut-il dès lors, par mesure de prudence, et pour stimuler l'amélioration de la qualité des logiciels, être au contraire assez strict en cas de défaut ? On le voit, il s'agit d'une question politique. Le juge dispose, ici plus qu'ailleurs, d'un large pouvoir d'appréciation. Pour le guider, la directive mentionne certains facteurs à prendre en considération :

– *la présentation du produit :* on pense ici au guide d'utilisation du logiciel (qui peut contenir des centaines de pages) ou à l'affichage à l'écran de mises en garde pour l'utilisateur;
– *l'usage* qui peut être raisonnablement attendu; certains logiciels sont adaptables à beaucoup d'usages différents qui seront déterminés par l'acquéreur sans que le producteur puisse toujours les prévoir;
– *le moment* de la mise en circulation; ce facteur sera d'autant plus important que la technique évolue vite (voir aussi point V infra).

Les questions ne sont pas fondamentalement différentes que pour les autres produits : la difficulté vient principalement de leur nouveauté, de l'absence de références dans la jurisprudence antérieure et de l'inexistence (en Belgique du moins) de normes ou de labels de qualité pour les logiciels.

Autre difficulté de la matière : les preuves à apporter par la victime sont difficiles et nécessitent le recours à des experts, ce qui coûte cher inévitablement.

L'une des façons pour le producteur d'écarter sa responsabilité est de recourir à la défense des *«risques de développement»*; on a dit qu'un logiciel sans erreur serait l'exeption à la règle; il faut ajouter qu'il est impossible de vérifier à l'avance qu'un logiciel ne contient aucune erreur (il s'agit d'un principe connu des informaticiens, et d'un théorème dérivé de la théorie de la calculabilité). On est bien ici dans le cas d'un risque de développement avec une impossibilité absolue de découvrir le défaut. Ce moyen de défense pourrait dès lors s'avérer plus efficace pour les producteurs de logiciels que pour les fabricants d'autres produits. Son introduction dans le droit national est facultatif; l'option qui sera prise par le législateur national sera dès lors fort importante; et on peut regretter le manque d'harmonisation qui résultera des solutions divergentes des législateurs des Etats membres.

II. Le dommage :

Après avoir prouvé un défaut du logiciel, la victime doit établir l'existence d'un dommage couvert par la directive : en résumé, un dommage corporel ou un dommage a un bien privé.

On a donné au début des exemples possibles de tels dommages : les exemples «classiques» de dommages corporels sont ceux de robots qui blessent des ouvriers, ceux d'instruments médicaux qui peuvent blesser des patients ou d'avions pilotés avec l'aide d'ordinateurs.

Quant au dommages matériels, on citait l'exemple de la cave à vin; on peut en trouver d'autres et, à l'avenir, ces exemples se multiplieront avec l'usage croissant de petits micro-processeurs dans des appareils auxquels on est exposé tous les jours comme une télévision, un ascenseur, un système de chauffage central ou de climatisation, etc...

Cependant, là où les dommages les plus importants sont possibles (et courants), à savoir les problèmes dans les systèmes informatiques utilisés par les entreprises, les banques, les agences de voyage, les compagnies d'assurance, pour leur comptabilité, leur facturation, le stockage d'informations, l'échange d'information commerciales ou financières, etc... dans tous ces cas, la directive ne s'applique pas puisqu'il s'agit alors de dommages financiers ou économiques, qui ne sont pas couverts par la directive.

Dans tous les cas, ce sera normalement la responsabilité contractuelle qui jouera, et les clauses exonératoires de responsabilité restent valables.

Puisqu'on a beaucoup parlé dans la presse des «virus» (comme celui du vendredi 13 octobre 1989), on peut en dire quelques mots :

– Le virus est un petit programme inséré sur un logiciel pré-existant et qui a pour objectif de se multiplier le plus possible afin de contaminer (via les réseaux) d'autres programmes et d'autres ordinateurs;

– ce virus peut s'attaquer aux données stockées dans les systèmes informatiques des entreprises, de sorte que les dommages possibles sont normalement des dommages financiers ou des dommages causés à des biens commerciaux (non privés); donc la directive ne s'appliquerait pas.

De plus, le producteur du logiciel sur lequel s'est greffé un virus pourrait de toutes façons se défendre sur base de l'article 7.b. : le défaut (le virus) n'existait pas quand le logiciel a été mis en circulation.

En résumé, on a donc vu que la condition du *défaut* donnait lieu à beaucoup d'incertitude; on vient de voir que la condition du *dommage* (tel que la directive le prévoit) limite ses possibilités d'application).

III. Le lien de causalité

Il nous reste à analyser la condition de causalité entre le défaut et le dommage. On va voir que cette condition vient aussi limiter l'incidence de la directive pour les producteurs de logiciels.

En parlant du caractère du logiciel, on a insisté sur le fait que les logiciels avaient toujours un effet tangible : soit que, pour un ordinateur classique, les données s'affichent à l'écran, soit que, pour un robot (au sens large), celui-ci se mette à bouger et à effectuer des gestes mécaniques.

On voit immédiatement qu'on peut faire une *distinction* et qu'il y a donc deux sortes de logiciels :
- celui qui va traiter de l'information et donner un résultat «intellectuel»;
- celui qui va actionner une machine et entraîner un résultat «mécanique».

Le premier ne commande qu'à l'ordinateur, il n'a pas «d'interface machine». Le second commande à un robot; il a un «interface extérieur».

Le premier s'adresse à un utilisateur humain; quand l'opération est finie, le résultat s'affiche, et l'utilisateur est libre d'en faire ce qu'il veut. S'il n'utilise pas les informations (exactes ou erronées) données par l'ordinateur grâce au logiciel, absolument rien ne se passera (et aucun dommage n'est possible).

Le deuxième s'adresse au robot en lui donnant des instructions qui vont l'actionner, sans qu'une intervention humaine (une décision) soit nécessaire à ce stade. Le logiciel, en commandant la machine, a ici un effet direct sur son entourage; il joue le même rôle que n'importe quelle autre partie mécanique du robot.

Pour revenir aux exemples donnés :
- dans le cas de robot dans la chaîne de montage, il est actionné directement par un logiciel (et il y a dommage corporel);
- dans le cas de l'appareil de radiographie dans un hôpital, c'est la même chose (et il y a dommage corporel);
- dans le cas de la cave à vin, également, la hausse de température est un effet direct de l'erreur du micro-processeur (et il y a dommage matériel à des biens privés);
- par contre, dans le cas du système-expert médical, il n'y a dommage possible que suite à la décision du médecin (ou de patient) prise sur base d'informations inexactes.

Il me semble que c'est seulement pour les cas où il n'y a pas l'intervention d'une telle décision humaine que la responsabilité sans faute à été prévue et devrait s'appliquer, là où il y a un lien de causalité direct entre le défaut d'un produit et un dommage.

Dans l'hypothèse où des informations inexactes sont données par le logiciel à un être humain, ce n'est pas le produit qui peut causer lui-même les dommages (contrairement au logiciel dans le robot), c'est l'information contenue dans le «produit» ou fournie par ce produit qui, via l'interprétation et la pensée humaine, va entraîner une décision mal-fondée et, le cas échéant, des dommages.

Pour prendre l'exemple du système-expert médical, en cas d'erreur dans le logiciel, il peut y avoir un faux diagnostic et des dommages corporels pour le malade qui en suivrait les indications. Peut-on, dans ce cas-là, imposer au producteur une responsabilité sans faute ? Si oui, on peut alors imposer la même responsabilité aux éditeurs d'ouvrages médicaux (scientifiques ou de vulgarisation), car la situation est la même : il s'agit d'un support contenant des informations destinées à un utilisateur pour l'aider à prendre une décision. On en arriverait alors, via un texte sur la responsabilité du fait des produits, à un système de responsabilité sans faute du fournisseur d'informations, applicable aux systèmes-experts, mais aussi aux livres, à la presse, etc...

Or, d'une part, il est clair que ce n'est pas ce que recherchait la directive à l'origine. Et, d'autre part, il ne me semble pas qu'on puisse l'interpréter d'une façon tellement large qu'elle en viendrait à établir un système de responsabilité stricte pour informations défectueuses.

Une telle responsabilité constituerait en effet une modification tellement importante des principes en vigueur actuellement qu'il me semble que seul le législateur pourrait l'introduire (suite à un large débat d'idées sur la question) et qu'en tous cas, cette réforme ne peut découler de la seule interprétation, par des juristes, d'un texte qui n'avait absolument pas une telle réforme pour objectif.

Quelles sont alors les *conséquences de cette distinction* sur le régime de responsabilité de producteur et sur la possibilité de clauses exonératoires ? On peut reprendre la distinction de la façon suivante, selon que le dommage est causé par le contenu informationnel du programme (I) ou selon qu'il est causé directement (II).

(I). Dans certains cas (c'est l'exemple du système-expert médical ou également de la banque de données), le dommage ne peut arriver que suite à une décision humaine. Ces dommages, à mon avis, ne sont pas couverts par la directive. Dès lors, il reste possible de

les écarter par une clause exonératoire de responsabilité. Dans ce cas, il y a fourniture d'informations : la responsabilité qui entre en jeu est la responsabilité traditionnelle basée sur la faute. Cependant, il faut relativiser l'importance de la distinction entre responsabilité stricte et responsabilité traditionnelle de deux façons :

1) même en droit commun, les clauses exonératoires de responsabilité sont soumises à certaines limites;

2) une responsabilité sur base de la faute peut être stricte pour le producteur; progressivement, la jurisprudence (en Belgique, notamment) a évolué sur ce point et aboutit parfois à des résultats assez similaires à ce qu'entraînerait un système de responsabilité stricte. Dans ce cas de l'information, plus celle-ci a un caractère pratique et plus elle est censée entraîner le destinataire à un comportement précis, plus l'obligation du fournisseur d'informations sera lourde; il reste cependant la possibilité de s'exonérer de certains cas de responsabilité.

(II). Par ailleurs, on l'a vu, certains dommages peuvent découler directement du fonctionnement défectueux du logiciel dans un robot (ou un chauffage central, ou un appareil médical) : ces dommages ne peuvent plus être écartés par le biais d'une clause exonératoire. C'est ici l'effet le plus sensible de la directive. Le nombre d'accidents de ce genre risque d'augmenter avec l'utilisation de l'informatique et des micro-processeurs dans la vie de tous les jours.

L'application des principes de la directive dans ce cas est, me semble-t-il, assez logique : au fur et à mesure que l'utilisation des logiciels se banalise et se multiplie, il n'y a pas de raison de différencier ces produits des autres produits traditionnels; les raisons qui justifient l'instauration d'une responsabilité sans faute, si elles sont valables pour les produits traditionnels, le sont aussi pour les produits qui incorporent un logiciel.

Pour la victime, en cas d'accident, le résultat est le même; et on voit mal pourquoi on accorderait à une catégorie des producteurs (les producteurs de logiciels) un régime plus favorable que pour les autres producteurs de produits plus traditionnels comme les médicaments ou les voitures. C'est d'autant plus vrai qu'en général la victime pourra poursuivre le fabricant du chauffage ou de l'ascenseur, qui incorpore un logiciel ou un micro-processeur (si les juges considèrent par exemple, l'appareil de chauffage comme défectueux). Si on dit que le logiciel n'est pas couvert par la directive, cela veut dire que les fabricants des produits finis incorporant un logiciel supportent seuls les conséquences des défauts de ce logiciel, à l'exclusion de la responsabilité du fabricant du logiciel. On le voit, ce n'est donc pas seulement pour les consommateurs qu'une telle solution serait défavorable, mais pour beaucoup de fabricants, qui hésiteront dès lors (on pourrait le craindre) à recourir à ces nouvelles technologies.

Un logiciel peut donc produire deux sortes de résultats différents : *informer une personne* et l'aider dans une décision ou *actionner une machine*.

A nouveau, on retrouve un caractère hybride du logiciel qui peut jouer le rôle d'un texte (source d'information) ou le rôle d'un outil (actionnant le robot ou l'appareil médical); mais cette fois, on est dans le domaine des résultats que peut fournir le logiciel. On avait d'abord mentionné le caractère hybride du logiciel pour expliquer qu'il ne fallait pas être trompé par le fait qu'un programme était certes toujours «écrit» par un programmeur, mais qu'il entraînait toujours des résultats corrects sur une machine (ordinateur ou robot). On voit, à ce stade-ci, que même s'il y a toujours un effet concret et matériel (informations apparaissant à l'écran, ou mouvements du robot), il faut distinguer selon la possibilité effective de causer un dommage : soit directement (robot), soit indirectement (informations à l'utilisateur).

Une objection que l'on pourrait faire à ce raisonnement est que la directive elle-même ne précise pas le caractère (direct ou indirect) du lien de causalité. Il a donc fallu voir plus loin et analyser les conséquences qui découleraient de telle ou telle conception de la causalité. Or, à notre avis, la directive ne s'applique pas à la fourniture d'informations (pour les raisons que l'on a indiquées); donc, pour éviter cette conséquence, il faut limiter l'application de la directive aux dommages causés directement par le *«logiciel-outil»* (dans un robot), et exclure les dommages causés par le *«logiciel-information»* (ou plutôt par le contenu informationnel du logiciel).

Pour les produits tels que les livres, ou les logiciels traitant de l'information, il faut donc faire une distinction entre le support et le contenu et considérer, à mon avis, que la directive ne vise que les dommages causés par le support (directement). Par contre, pour les autres programmes, cette distinction ne tient pas : le support et les instructions sont inséparables. La façon dont les instructions sont rédigées, et leur contenu, vont directement actionner le robot, à la façon dont un engrenage ou une pièce mécanique le ferait : les instructions sont à comparer avec la forme de l'outil.

Pour mieux comprendre la distinction entre les dommages causés par le support et les dommages causés par le contenu, on peut prendre l'exemple d'un livre :

– Quelle sorte de dommages peut causer le contenu ? Pour prendre un exemple qui s'est passé en France (affaire Nathan), un guide sur les champignons comestibles contenait certaines imprécisions; conséquence : plusieurs personnes ont été empoisonnées.

– Quelle sorte de dommages pourrait causer le support ? L'hypothèse est plus théorique, c'est l'histoire du livre «Le Nom de la Rose», où c'est l'encre du livre qui est empoisonnée et qui entraîne la mort du Lecteur.

Pour le logiciel, on a vu qu'il pouvait occasionner un dommage de deux façons; pour l'illustrer, on peut dire qu'un logiciel peut me donner par exemple une recette de tarte aux pommes (et il est source d'informations), mais qu'un logiciel introduit dans un robot spécialisé pourrait aussi commander au robot de faire la tarte aux pommes (et il est alors outil). Dans le premier cas, j'obtiens des informations, dans le second cas, j'obtiens un résultat tangible.

Le raisonnement qu'on a suivi pour le système-experts peut être tenu pour les banques de données; certains soutiennent que la directive s'appliquerait en cas d'informations erronées contenues dans la banque de données. Je ne pense pas qu'on puisse suivre ce raisonnement : encore une fois, il s'agirait alors d'une responsabilité sans faute pour informations défectueuses.

Dernière remarque : la distinction est plus facile à faire en théorie qu'en pratique et il y a dans la réalité des situations mixtes : pensons à un logiciel qui aurait pour fonction de contrôle le déroulement d'un processus de fabrication dans une usine, d'avertir en cas de danger et de donner les indications à suivre pour éviter un accident. Dans ce cas, le logiciel donne certes des informations à un être humain, mais il s'agit d'un logiciel très lié à un produit; il ne fait en quelque sorte qu'en donner le mode d'emploi, et on peut dire qu'il fait partie de «la présentation du produit» (article 6.1.a de la directive). Donc, parmi les logiciels qui fournissent des informations (et qui, à mon avis, ne sont pas couverts par la directive – où plutôt dont les dommages qu'ils peuvent occasionner ne sont pas visés par la directive), il faut encore séparer ceux qui sont associés à l'utilisation d'un autre bien et dont ils constituent un élément (qui en améliore la sécurité) : pour ces logiciels, même s'ils s'agit d'informations, on retombe dans le champ d'application de la directive. On voit donc que la distinction entre les deux catégories de logiciels (et de dommages) laisse subsister des cas intermédiaires.

En résumé, de même que la définition du dommage couvert par la directive, la condition de causalité vient aussi en limiter les applications possibles.

Quand la victime est parvenue à prouver le *défaut* (et on a vu les imprécisions liées à cette notion), le *dommage* (et on a vu que dans beaucoup de cas la directive ne s'appliquerait pas) et le *lien de causalité* (et on a également vu que cette condition limitait le champ de la directive), il lui reste à établir deux choses; cela fait l'objet des points IV et V (pour lesquels on se limitera à quelques observations).

IV. *Détermination du producteur responsable :* un problème pourrait se poser dans le cas d'un logiciel conçu pour être adapté par l'utilisateur à ses propres besoins : en cas de défaut, il faudra retrouver son origine : soit dans le logiciel, soit dans les adaptations faites par l'utilisateur. Une expertise sera certainement nécessaire; mais il faut remarquer que la directive établit une présomption à charge du producteur : c'est lui qui devra prouver que le défaut est né après la mise en circulation du produit, car on présume que le défaut existait depuis le départ (article 7.b). S'il prouve cela, il y aura réduction ou suppression de sa responsabilité (article 8).

V. *Date de mise en circulation,* qui fait courir le délai de prescription de dix ans : une difficulté peut provenir du fait que les logiciels sont régulièrement améliorés et que de nouvelles versions sont distribuées dans le cadre des contrats de maintenance; si l'utilisateur est relié à un réseau, la nouvelle version peut être installée sans intervention de sa part, par voie de télécommunication.

Quelques mots de conclusion :

La directive s'applique aux logiciels parce que ceux-ci sont des biens corporels. Et les objections à l'application de la directive ne peuvent être retenues. La Commission a confirmé le principe de l'application aux logiciels.

Les tribunaux gardent un large pouvoir d'appréciation quant à la notion de défaut de logiciel (question d'opportunité).

Le moyen de défense des risques de développement devrait permettre au producteur d'écarter sa responsabilité, à la condition qu'il puisse prouver qu'il a effectivement effectué de nombreux tests (sinon, il pourrait être tenu responsable sur base de la faute).

Les cas les plus courants et les plus coûteux de dommages, à savoir les dommages financiers ou commerciaux d'une entreprise, ne sont pas visés par la directive : à cet égard, les clauses exonératoires restent valables.

Enfin, il faut distinguer deux sortes de logiciels, d'après la fonction qu'ils remplissent (soit donner des informations, soit actionner un robot).

Cette distinction permet, je pense, de limiter l'incidence de la directive aux domaines pour lesquels elle était destinée à l'origine, à savoir les dommages causés par un bien tangible, mais non les dommages causés par un bien intangible tel qu'une information : pour ces derniers aussi les clauses exonératoires restent licites.

Donc, la directive, ou plutôt la responsabilité sans faute et l'interdiction des clauses exonératoires ne seront appliquées que dans certains cas; beaucoup d'exemples de dommages causés par des logiciels resteront régis par le droit commun.

Quant à l'importance du changement qu'entraînera la mise en œuvre de la directive, on peut redire (pour les logiciels comme pour les autres produits) qu'il faut la relativiser :
- d'abord, parce qu'en matière de clauses exonératoires, déjà actuellement, il y a des limites à leur validité;
- mais surtout parce que les notions de faute (en rssponsabilité délictuelle) et de vice caché (en responsabilité contractuelle) sont parfois (et de plus en plus) interprétés par les juges d'une façon qui nous rapproche des principes de la directive.

Pour terminer, il faut préciser que l'on s'est limité à une analyse juridique, sans beaucoup parler des questions d'opportunité; ce qu'on a essayé de voir c'est : puisque la directive *peut* s'appliquer aux logiciels, *quand* s'appliquera-t-elle et à quelle conditions ?

La question de savoir s'il était opportun d'instaurer, dans ce secteur, une responsabilité sans faute n'a pas été longuement analysée pour deux raisons :
- parce que c'est une question «politique» où chacun peut avoir son opinion et défendre une option plutôt que l'autre;
- mais surtout, parce qu'il est trop tard pour poser la question : en effet, la réponse a été donnée implicitement lors de l'adoption de la directive en 1985 et une option a été choisie en faveur d'une responsabilité sans faute. Il est trop tard, nous semble-t-il, pour la remettre en cause.

Bibliographie

Directive du Conseil du 25 juillet 1985 relative au rapprochement des dispositions législatives, réglementaires et administratives des Etats membres en matière de responsabilité du fait des produits défectueux, (85/374/CEE), J.O.C.E., n° L 210/29, du 7-8-1985.

Question écrite n) 706/88 de M. Gijs de Vries à la Commission des Communautés européennes (5 juillet 1988); Réponse donnée par lord Cockfield au nom de la Commission (15 novembre 1988), J.O.C.E., n° C 114/42, du 8-5-1989.

STUURMAN C., «Product liability for software in Europe. A discussion the EC Directive of 25 July 1985», in Advanced Topics of Law and Information Technology, ed. C. VANDENBERGHE, Kluwer, Computer/Law Series, nº 3, 1989.

STUURMAN C. & VANDENBERGHE G., «Software fouten, een 'zaak' van leven of dood», Nederlands Juristen Blad, 24-31 Dec. 1988, p. 1667.

DOMMERING VAN RONGEN L., «Produktaansprakelijkheid en software», Computerrecht, 1988-5, p. 230.

DE RAADS W.A.P., chapitre sur la responsabilité du fait des produits, in Hoofdstukken Informaticarecht, ed. F. DE GRAAF, Samson, Alphen aan den Rijn, 1989.

POULLET Y., «Les contrats télématiques face au droit communautaire», Lamy Droit de l'Informatique, 1988, nº I, p. 22.

ZOPPINI A., «Les systèmes-experts et la responsabilité civile», Droit de l'informatique et des Télécoms, 1989/1.

BERTRAND A., «L'intelligence artificielle, la robotique, les systèmes-experts et le droit (droit français et américain)», Expertises, 1987, p. 219.

MORTIMER H., «Computer-aided medicine: present and future issues of liability», Computer Law Journal, vol.IX, 1989, p. 177.

HUET J., «Product Liability in the Information Field», in International contracts for sale of information services, The dossiers of the Institute of International Business Law and Practice, I.C.C., 1988.

BOWN C., «Liability for the supply of defective software», Computer Law and Practice, Sept./Oct. 1986.

SCOTT M., «Who is liable for software errors? Proposed new product liability law in Australia», The Computer Law and Security Report, May-June 1989, p. 28.

HIRSCHBAECK J., «Is software a product?», Computer Law and Practice, vol. 5, 1989, p. 154.