

RESEARCH OUTPUTS / RÉSULTATS DE RECHERCHE

L'applicazione della direttiva comunitaria sulla responsabilità del produttore nel campo del software

Triaille, Jean-Paul

Published in:

Il Diritto dell'Informazione e dell'Informatica

Publication date:

1990

Document Version

le PDF de l'éditeur

[Link to publication](#)

Citation for pulished version (HARVARD):

Triaille, J-P 1990, 'L'applicazione della direttiva comunitaria sulla responsabilità del produttore nel campo del software', *Il Diritto dell'Informazione e dell'Informatica*, no. 2, pp. 725-735.

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.

RICERCHE

L'APPLICAZIONE DELLA DIRETTIVA COMUNITARIA SULLA RESPONSABILITÀ DEL PRODUTTORE NEL CAMPO DEL SOFTWARE

SOMMARIO

1. Introduzione. — 2. Applicabilità della direttiva al *software* « La direttiva è applicabile ai programmi per *computer*? ». — 2.1. 1ª obiezione: « La direttiva non è applicabile al *software* perché il *software* è un servizio e non un prodotto ». — 2.2. 2ª obiezione: « La direttiva non è applicabile al *software* perché il *software* è un bene immateriale; e la direttiva non contempla che i beni materiali ». — Risposta A: « La direttiva non richiede che il bene sia materiale; e quindi nulla impedisce che essa venga applicata al *software* ». — Risposta B: « La direttiva contempla solo beni materiali, ma il *software* è un bene materiale. Di conseguenza, la direttiva è applicabile ». — 3. Applicazione della direttiva al *software*: « In quali casi si applicherà la direttiva? ». — 3.1. La condizione di difettosità. — 3.2. Il danno. — 3.3. Il nesso di causalità. — 3.4. La determinazione del produttore responsabile. — 3.5. La data di immissione sul mercato. — 4. Pochi cenni di conclusione. — Bibliografia.

1. INTRODUZIONE

Il problema dell'applicazione al *software* della direttiva relativa alla responsabilità da prodotti difettosi è, al momento, fortemente controverso.

L'importanza della risposta deriva dalla serie di modifiche introdotte dalla direttiva nel campo del diritto europeo riguardante la responsabilità dei produttori, cioè principalmente:

— introduzione di una responsabilità senza colpa (oggettiva) a carico del produttore in caso di difetto del prodotto;

— invalidità delle clausole limitative o esonerative della responsabilità.

Per cominciare, citeremo alcuni esempi di danno causato da prodotti difettosi:

— Recentemente, negli Stati Uniti, un paziente doveva essere sottoposto ad un trattamento di raggi per curare un tumore alla spalla. L'apparecchiatura medica, azionata da un *software*, ha inviato, in una sola volta, una dose troppo

elevata di raggi che si è rivelata mortale; e l'incidente si è ripetuto in numerosi ospedali degli Stati Uniti. Si è così scoperto che era il *software* ad essere difettoso.

— Nel 1984, un operaio di Detroit è stato ucciso da un robot negli stabilimenti della Ford. Questo tipo di incidente si è già verificato parecchie volte in Giappone (spesso un difetto del *software* ne era l'origine).

— Si dice che un errore di programmazione ha rischiato di far scattare un allarme nucleare alcuni anni fa ... ma si possono trovare anche esempi più divertenti: ad esempio, nel 1988 a Londra, in seguito ad un errore nel programma del *computer*, il Fisco ha inviato la dichiara-

* Il presente scritto costituisce la relazione ad una giornata di studi Euroforum sulla responsabilità del produttore svoltasi a Bruxelles il 20 ottobre 1989. La traduzione dal francese è di ANDREA ZOPPINI.

razione dei redditi ad un milione di persone anziane che da anni non erano più tenute a pagare le tasse. Ora, la legge prevede che se si riceve una richiesta di tasse la si deve anche compilare e che una dichiarazione compilata deve essere presa in esame dall'Amministrazione...

Certo questi esempi non riguardano tutto il campo coperto dalla direttiva; ma sono stati citati per far notare la varietà di situazioni e di danni possibili quando un *software* funziona male.

2. APPLICABILITÀ DELLA DIRETTIVA AL SOFTWARE: « LA DIRETTIVA È APPLICABILE AI PROGRAMMI PER COMPUTER? »

Prima di analizzare *quando* (in quale caso) si applicherà la direttiva al settore del *software*, bisogna risolvere un problema preliminare, riguardante l'applicabilità o meno della direttiva, visto che è stata sollevata una serie di obiezioni su questo punto. Verranno rapidamente citate queste obiezioni e si indicherà la ragione per cui esse non sono accettabili. Non è possibile fare un bilancio complessivo dell'analisi di questi problemi, perché il giudizio non è ancora unanime; bisogna eliminare questi primi ostacoli (ai quali si è dato il nome di « prima generazione di problemi ») prima di passare alla fase seguente che consiste nell'analisi delle condizioni di applicabilità del testo comunitario.

2.1. 1^a obiezione: « La direttiva non è applicabile al software perché il software è un servizio e non un prodotto. — È vero che la direttiva non riguarda i servizi (per i quali la Commissione ha intenzione di proporre altre norme).

Ma, anche se è vero che la fornitura di un *software* — e in particolare la produzione di un *software* su misura — è un servizio, il *software* in sé è un prodotto. Nello stesso modo un ristorante fornisce un servizio ai clienti, ma le pietanze che vengono « servite » sono un prodotto; e il semplice fatto che un prodotto venga fornito nell'ambito di un servizio non lo esonera dalla sfera di applicabilità della direttiva.

Si è spesso obiettato che si deve fare una distinzione tra *progiciel* (*software* standard venduti su larga scala) e *software* creati su misura per le necessità specifiche di una società. Solo i primi dovrebbero essere considerati come prodotti. Ma mi sembra che, anche se nel caso dei *software* su misura è stata effettivamente fornito un servizio, con un rapporto da persona a persona, ciò non impedisce che l'oggetto stesso del servizio fosse la creazione di un prodotto (che ha d'altronde le stesse caratteristiche di un *software* standard).

Si ha la stessa situazione quando un acquirente invece di acquistare una vettura di serie, ordina tutta una serie di *gadget* o di *optional* particolari. In questo caso, come per un *software* su misura, vi è un servizio che porta alla creazione di un « prodotto » secondo l'accezione della direttiva.

Inoltre, questa distinzione tra *software* standard e *software* su misura incontra altre due difficoltà:

— molti programmi sono in gran parte standard e vengono poi parzialmente adattati alle necessità dell'utente: come dovranno essere considerati, come prodotto o come servizio? La risposta sarebbe molto importante in questo caso per decidere se si ha più o meno a che fare con una responsabilità senza colpa;

— inoltre, ciò che oggi viene prodotto su misura può, se il fabbricante lo desidera, essere domani prodotto in serie: ancora una volta, su che base faremo la distinzione?

Infine, è del tutto inesatto pretendere che la direttiva possa essere applicata solo ai beni prodotti in serie; anche se l'esposizione dei motivi prevede questa esigenza, l'art. 2 non la contempla. In sede di Parlamento Europeo era stato suggerito di escludere espressamente i prodotti « artigianali »; la proposta non è stata accettata, e la Commissione stessa ha confermato che i prodotti artigianali erano coperti dalla direttiva (come vedremo più avanti).

Per riassumere, si può dire che non è accettabile la tesi secondo la quale il *software* sarebbe un servizio e non un prodotto. Né d'altronde si può isolare una categoria di *software* (quelli che vengono creati « su misura ») per poi escluderla dal campo della direttiva, perché in effetti, vi sono innanzitutto

delle obiezioni d'ordine materiale (dove fissare il limite?) e inoltre la direttiva non prevede comunque un trattamento differenziato tra i beni che sono oggetto di una produzione industriale e gli altri beni (prodotti di tipo « artigianale »).

2.2. 2^a obiezione: « La direttiva non è applicabile al software perché il software è un bene immateriale; e la direttiva non contempla che i beni materiali ». — Questa tesi può essere criticata in due modi:

A) sostenendo che la direttiva è applicabile anche ai beni immateriali (e quindi al software) (e non credo che questo ragionamento sia giusto);

B) dimostrando che il software non è un bene immateriale, e che dunque nulla impedisce l'applicazione della direttiva.

A. In primo luogo si può dire: « La direttiva non richiede che il bene sia materiale, e dunque nulla impedisce che essa venga applicata al software ».

Il problema sta dunque nel sapere se la direttiva contempla anche i beni intangibili; siccome il testo non è esplicito a questo riguardo, si è cercato di addurre come pretesto la terminologia, confrontando le varie traduzioni. Non sembra che da questa analisi si sia ottenuta una risposta esauriente.

Si può anche cercare di interpretare il testo della direttiva in modo « teologico » (in base allo scopo che essa si prefigge). Ma quale è allora questo scopo? Si può facilmente dimostrare che scopo della direttiva è di salvaguardare e proteggere i consumatori. È chiaro che gli altri motivi addotti (distorsione della concorrenza ed effetto sulla circolazione delle merci) sono più formali che reali. Se lo scopo è quello dichiarato (cioè la salvaguardia e la protezione dei consumatori), si potrebbe dire: « In caso di dubbio, scegliamo la soluzione più favorevole per i consumatori »; e di conseguenza, per quel che riguarda il problema dei beni immateriali, « ampliamo la sfera d'influenza della direttiva in modo da includerli, nell'interesse dei consumatori ».

Tuttavia, si è ripetutamente parlato nell'espone i motivi, di una « equa ripartizione dei rischi » tra la vittima ed il produttore. Di conseguenza, se la direttiva cerca di stabilire questo equilibrio, nulla autorizza a che la bilancia penda

più a favore dell'uno che dell'altro; ciò equivarrebbe infatti a dare alla direttiva un carattere unilaterale che in ogni caso essa pretende e si augura di non avere. Quindi, né la terminologia (nelle varie versioni), né una eventuale interpretazione, danno una risposta esauriente al problema dell'applicazione della direttiva ai beni immateriali.

Per quel che mi riguarda, io non penso che la direttiva sia applicabile ai beni immateriali, e questo per due ragioni: una che deriva dall'analisi del testo (1°) e l'altra che è innanzitutto una ragione di opportunità (2°).

1) In primo luogo, leggendo il testo, è chiaro che non si sono mai presi in considerazione i beni intangibili. In effetti, molte disposizioni creerebbero dei problemi qualora le si applicasse ad oggetti immateriali:

— « materie prime » di un oggetto intangibile;

— « componenti »;

— « produttore » (in inglese: *manufacturer*);

— « importatore »;

— come apporre il proprio nome o il proprio marchio su un bene immateriale?

— cosa intenderemmo per sua « presentazione »?

— quando sarebbe « messo in circolazione »?

— *quid* della sua concezione (*design*)?

— *quid* del danno « causato ad oggetto diverso dal prodotto difettoso »?

Ciò mostra chiaramente che i redattori non hanno mai preso in considerazione i beni intangibili.

2) D'altronde — e mi accingo a parlare delle ragioni di opportunità — se si considerano i beni immateriali, dove ci si deve fermare? Si considera ad esempio, tra gli altri, l'informazione. Ci torneremo più avanti, ma è veramente opportuno stabilire una responsabilità senza colpa in caso di informazione « difettosa »?

Riassumendo, non mi sembra esatto dire che la direttiva si applica al software perché essa è applicabile anche ai beni immateriali.

B. Allora la direttiva non è applicabile al software?

No, perché si può fare un ragionamento diverso e dire: « È vero che la direttiva

non contempla che i beni materiali, ma il software è un bene materiale. Quindi, la direttiva va applicata ». Qui, l'obiezione non riguarda più la sfera di applicazione della direttiva ma la natura stessa del *software*.

Il *software* viene spesso definito come « una serie di istruzioni date ad un *computer* e che rendono possibile l'esecuzione di un preciso compito ». Il termine « istruzione », a mio avviso, induce in errore, così come capita quando si parla di un « linguaggio » informatico. In realtà, queste istruzioni sono materializzate su un supporto (dischetto, CD ROM, nastro magnetico, ecc...) e quando lo si introduce nella macchina, esse hanno un effetto materiale, concreto e ben visibile:

— se vengono introdotte in un *computer*, questo mostrerà i risultati sullo schermo o stamperà dei dati su carta;

— se vengono introdotte in un *robot*, questo verrà azionato e potrà spostarsi, muovere degli oggetti, ecc. ...

Lo stesso a proposito del linguaggio perché è vero che si parte al momento della programmazione da un « testo » più o meno comprensibile, ma si giunge, alla fine del processo, ad una sequenza di 0 e di 1 (linguaggio binario) che assomiglia molto meno ad un « testo ».

In realtà, non bisogna paragonare queste istruzioni alle lettere di un testo, ma piuttosto alla forma di un utensile o alle caratteristiche tecniche di uno strumento (il suo peso, la sua resistenza). Volendo schematizzare, si potrebbe dire che praticamente il *software*, da un punto di vista concreto, non fa che canalizzare, attraverso mille ramificazioni, degli impulsi elettrici. Cosa c'è d'immateriale in questo? Si può dire che sia realmente immateriale il diritto su un oggetto, una idea, o una informazione; ma non si capisce perché dovrebbe esserlo il *software*. Mi sembra che si faccia confusione su ciò che è realmente immateriale (ad esempio una informazione) e ciò che è solo impercettibile ai nostri sensi.

Per fare un paragone, si potrebbe pretendere che un microbo o un batterio siano immateriali?

Il *software* introdotto in una macchina (come il microbo nell'organismo) provoca effetti materiali; inversamente, ciò che è realmente immateriale non può produrre alcuna trasformazione mate-

riale, salvo (e ci torneremo) nel caso in cui un essere umano continui il processo decidendo di effettuare una certa azione (ad esempio, prendendo una decisione in base ad una informazione). Ci si può domandare se questa idea, secondo la quale il *software* sarebbe immateriale, non derivi dal fatto che l'informatica e i *computer* vengano talvolta ancora considerati come qualcosa di diabolico o magico... A questo riguardo, era forse il caso di demistificare un po' la materia.

Si constata quindi che il *software* ha un carattere ibrido (di volta in volta testo e utensile: insieme di istruzioni « scritte » da un essere umano ma che azionano una macchina); ciò implica dei malintesi. Si incontrano d'altronde le stesse difficoltà quando si tenta di proteggere il *software* attraverso la proprietà intellettuale: schematizzando, si può dire che alcuni consigliano l'adozione del diritto d'autore perché il *software* è un testo, e che altri sostengono la soluzione che porti al brevetto perché il *software* è un utensile.

Numerose obiezioni sono state sollevate contro la tesi che difende il carattere materiale del *software*:

— Innanzitutto, il fatto che il *software* può esistere ed essere trasferito tramite telecomunicazioni anche in mancanza di un supporto (via cavo o persino per radio). In questo caso, si può dire che il programma non è realmente trasferito: si invia solamente una copia; in questo modo, si mette il programma di partenza (bene materiale) a disposizione degli utenti, attraverso i mezzi di telecomunicazione. Perché sia applicabile, la direttiva non richiede che il bene sia fisicamente trasferito: basta una qualsiasi « forma di distribuzione » (in senso economico, vd. art. 7 c). Nel nostro caso (*tele-software*), il *software* di partenza è effettivamente messo in circolazione e reso disponibile per gli utenti.

— In seguito, il fatto che il *software* non sia generalmente venduto, ma concesso in licenza e che esso diventi l'oggetto dei diritti della proprietà intellettuale. Come abbiamo detto, il modo di distribuzione, così come la qualificazione del contratto tra le parti (vendita o licenza) non ha importanza per la direttiva. Peraltro, altri beni, come ad esempio una macchina (nel senso tradizionale), sono soggetti ai diritti di proprietà

intellettuale (brevetto) senza che sia mai stato indicato il loro carattere immateriale.

Per riassumere questa prima parte, si possono ricordare le varie tesi:

Gli uni hanno detto: « Il *software* è immateriale, ma la direttiva non contempla che i beni tangibili, quindi i *software* non sono previsti ».

Altri hanno detto: « Il *software* è immateriale, ma la direttiva contempla anche i beni immateriali; quindi i *software* sono compresi nel campo d'applicazione della direttiva ».

Ciò che ho cercato di dimostrare è che « il *software* è tangibile e quindi è coperto dalla direttiva (che, a nostro parere, contempla solo i beni materiali) ».

La Commissione delle Comunità Europee si è pronunciata sul problema, in una risposta parlamentare pubblicata nel maggio di quest'anno. Riguardo al problema sulla applicabilità della direttiva al *software*, la Commissione ha risposto in modo alquanto laconico che « ...la direttiva si applica al *software*, così come essa si applica ai prodotti artigianali ed artistici ».

Tuttavia, la Commissione non precisa a che titolo la direttiva contempli il *software*: come bene materiale o come bene immateriale?

Ci si sarebbe potuti limitare a dare questa risposta della Commissione, ma poiché questi giudizi possono sempre essere contestati in seguito per via legale, ci sembra importante giustificare il principio (accettato dalla Commissione) dell'applicazione della direttiva al *software*. Abbiamo appena visto che la direttiva si applicava al *software*; possiamo quindi passare alla seconda parte che riguarda le condizioni di applicazione della direttiva.

3. APPLICAZIONE DELLA DIRETTIVA AL SOFTWARE: « IN QUALI CASI SI APPLICHERÀ LA DIRETTIVA? »

Questi problemi sono stati fino ad oggi molto sottovalutati dalla dottrina. Vedremo che molte domande resteranno senza risposta, nell'attesa che vengano prese decisioni di carattere giudiziario in proposito. Ammettendo che la direttiva

sia applicabile, perché la norma nazionale che ne riprende i principi, possa essere applicabile ad un caso concreto, bisognerà che la vittima riesca a provare tre cose:

- I) il difetto del *software*;
- II) un danno coperto dalla direttiva;
- III) un nesso di causalità.

Inoltre, bisognerà stabilire chi sia il produttore del *software* (IV) e quando il *software* sia stato immesso sul mercato (V).

Per spiegarci meglio ci serviremo di alcuni esempi in cui un *software* difettoso può causare dei danni; man mano, vedremo i casi che sono contemplati dalla direttiva e quelli che vanno esclusi.

1° esempio (è il caso più frequente in cui un *software* può causare un danno): una piccola impresa acquista un *software* che gli permette di gestire la contabilità, le scorte di magazzino, la fatturazione e, in linea generale, di registrare tutte le informazioni utili all'impresa stessa. In seguito ad un difetto del *software*, il sistema si blocca. Conseguenza: completa disorganizzazione dell'impresa (secondo le statistiche, in tre casi su quattro, l'impresa è condannata ad una chiusura più o meno lunga).

2° esempio: un'impresa automatizza la propria catena di montaggio ed installa dei *robot* che affiancano gli operai. Il caso si è già verificato (negli Stati Uniti e in Giappone): il *robot*, azionato da un *software*, diventa incontrollabile e ferisce gravemente un operaio.

3° esempio: (molto simile al 3°) in Francia, in un ospedale, un'apparecchiatura per radiografie computerizzate non rispondendo più ai comandi del medico che lo utilizzava, invece di posarsi delicatamente sul petto di un paziente, gli ha schiacciato la cassa toracica.

4° esempio: il Signor X ha una cantina vinicola. È necessario installare un sistema che ne regoli la temperatura e l'umidità. Il sistema è azionato da un microprocessore (strumento che ingloba *hardware* e *software*). Il *software* è stato mal concepito, l'apparecchio si guasta: tutto il vino si altera.

5° esempio: un sistema-esperto medico viene venduto ai medici (e presto, negli Stati Uniti, sarà venduto anche ai consumatori). Un errore provoca una diagnosi sbagliata: il malato, curato male, subisce un danno fisico.

Torniamo alle tre condizioni necessarie perché si possa applicare il testo: il difetto, il danno e il nesso di causalità.

3.1. *La condizione di difettosità.* — Riprendendo le parole dell'art. 6: quando un *software* non ha offerto « la sicurezza che ci si poteva legittimamente aspettare da esso »? Un *software* che contenga un « *bug* » (un errore) è automaticamente difettoso?

Il problema è che, anche secondo un esperto in informatica, non esiste praticamente nessun programma che non abbia neanche un errore: un tale programma è quindi l'eccezione alla regola. Quale sicurezza dobbiamo quindi aspettarci? E quale sicurezza possiamo legittimamente aspettarci? È chiaro che la risposta dipende più da ragioni d'opportunità che non da argomenti giuridici. Da una parte, l'automatizzazione permette, in numerosi campi, progressi enormi da cui tutti traggono vantaggio; bisogna, per favorirne lo sviluppo, chiudere un occhio in caso di difetto? D'altra parte le sue possibilità d'applicazione sono immense e si estenderanno a tutti i campi; bisogna allora, per prudenza e per migliorare sempre più la qualità del *software*, essere intransigenti in caso di difetto? È chiaro che si tratta di un problema politico. Il giudice dispone, quindi più che in altri casi, di un grande potere di valutazione. Per guidarlo, la direttiva menziona alcuni fattori che devono essere presi in considerazione:

— *la presentazione del prodotto*: ci si riferisce alle istruzioni del *software* (che possono essere anche di centinaia di pagine) o alla visualizzazione sullo schermo per avvertire l'utente;

— *l'utilizzazione* che si può ragionevolmente pretendere; alcuni *software* sono adattabili a usi diversi che saranno determinati dall'acquirente senza che il produttore li possa sempre prevedere;

— *il momento* dell'immissione sul mercato; questo fattore sarà tanto più importante vista la velocità d'evoluzione della tecnica (vd. anche il punto *V infra*).

I problemi non sono poi così diversi da quelli che si pongono per altri prodotti: la difficoltà sta principalmente nella loro novità, nella mancanza di casi di riferimenti nelle sentenze precedenti e (almeno in Belgio) di norme e di marchi di qualità. Altra difficoltà della materia: le

prove che le vittime devono presentare sono difficili e richiedono il ricorso ad esperti che sono inevitabilmente molto cari.

Uno dei sistemi cui può ricorrere il produttore per sfuggire alle proprie responsabilità è la difesa dai « *rischi di sviluppo* »; abbiamo detto che un *software* senza errori è l'eccezione alla regola: bisogna aggiungere che è impossibile verificare in anticipo se un *software* non contiene nessun errore (si tratta di un principio conosciuto dagli esperti di informatica, e di un teorema derivato dalla teoria della calcolabilità). Ci troviamo di fronte ad un caso di rischio di sviluppo con assoluta impossibilità di scoprire il difetto. Questo medoto di difesa potrebbe quindi rivelarsi più efficace per i produttori di *software* che non per i produttori di altri prodotti. La scelta che verrà operata dal legislatore nazionale sarà quindi molto importante; e si avrà purtroppo una mancanza di armonia come risultato delle soluzioni divergenti adottate dai legislatori nazionali.

3.2. *Il danno.* — Dopo aver provato il difetto del *software*, la vittima deve dimostrare l'esistenza di un danno coperto dalla direttiva: in breve, un danno fisico o un danno ad un bene privato.

Abbiamo dato all'inizio alcuni esempi possibili di danni di questo tipo: gli esempi « classici » di danni materiali sono quelli di *robot* che feriscono gli operai, quelli di strumenti medici che possono ferire i pazienti o di aerei pilotati con l'ausilio di *computer*.

Quanto ai danni materiali, abbiamo citato l'esempio della cantina vinicola; se ne possono trovare altri e, in futuro, questi esempi si moltiplicheranno con l'introduzione crescente di piccoli microprocessori in apparecchi di uso quotidiano quali la televisione, gli ascensori, i sistemi di riscaldamento centralizzato o di climatizzazione, ecc...

Tuttavia, là dove sono possibili i danni più gravi (e correnti), cioè problemi nei sistemi automatizzati utilizzati dalle imprese, le banche, le agenzie di viaggio, le compagnie di assicurazione, per la loro compatibilità, la fatturazione, la registrazione di informazioni, lo scambio di informazioni commerciali o finanziarie, ecc..., in tutti i casi, la direttiva non è applicabile perché si tratta di danni finan-

ziari o economici che non sono contemplati dalla direttiva.

In tutti questi casi, entrerà in gioco la responsabilità contrattuale, ferme restando le clausole esonerative della responsabilità.

Poiché la stampa ha parlato a lungo dei « virus » (come quello di venerdì 13 ottobre 1989) è il caso di dire qualcosa a proposito:

— il virus è un piccolo programma inserito in un *software* preesistente e che ha lo scopo di moltiplicarsi il più possibile e di contaminare (attraverso la rete) altri programmi e altri *computer*;

— questo virus contamina i dati registrati nei sistemi automatizzati delle imprese, così che i danni possibili sono normalmente di carattere finanziario o sono danni causati a beni commerciali (non privati); dunque la direttiva non sarebbe applicabile. Inoltre, il produttore del *software* su cui si è innestato il virus potrebbe comunque difendersi in base all'art. 7 b): il difetto (il virus) non esisteva quanto il *software* è stato immesso sul mercato.

Riassumendo, si è dunque visto che la condizione riguardante il difetto dava luogo a numerose incertezze; abbiamo appena visto che la condizione riguardante il danno (così come lo prevede la direttiva) limita le sue possibilità d'applicazione.

3.3. *Il nesso di causalità.* — Non ci resta che analizzare la condizione riguardante la causalità tra il difetto ed il danno. Vedremo che questa condizione limita ulteriormente l'incidenza della direttiva nei confronti dei produttori di *software*.

Parlando del carattere materiale del *software*, abbiamo insistito sul fatto che il *software* ha sempre un effetto tangibile: sia per un *computer* classico dove i dati vengono presentati sullo schermo, sia per un *robot* (in senso lato) che si muove o effettua gesti meccanici.

Si vede subito che è possibile fare una distinzione e che esistono quindi due tipi di *software*:

— quello che tratta l'informazione e da un risultato « intellettuale »;

— quello che aziona una macchina e produce un risultato meccanico.

Il primo dà ordini solo al *computer*; non esista una macchina che funga da

« interfaccia ». Il secondo dà ordini ad un *robot*; esiste una « interfaccia esterna ».

Il primo tipo di *software* si indirizza ad un utente umano; quando l'operazione è finita viene mostrato un risultato e l'utente è libero di farne ciò che vuole. Non succederà nulla (e non ci sono danni) se non utilizzerà le informazioni (esatte o sbagliate) fornite dal *computer* grazie al *software*.

Il secondo tipo di *software* si indirizza ad un *robot* dandogli istruzioni che lo azioneranno, senza che in questa fase sia necessario l'intervento umano (sotto forma di decisione). Il *software*, dando un comando alla macchina, ha qui un effetto diretto su ciò che lo circonda; svolge lo stesso ruolo di una qualsiasi altra parte meccanica del *robot*.

Per tornare agli esempi già presentati:

— nel caso del *robot* della catena di montaggio, esso è azionato direttamente da un *software* (e si ha un danno fisico);

— nel caso dell'apparecchiatura per radiografie nell'ospedale, è la stessa cosa (e si ha un danno fisico);

— nel caso della cantina vinicola, ugualmente, l'aumento di temperatura è una conseguenza diretta dell'errore del microprocessore (e si ha un danno materiale a dei beni privati);

— invece, nel caso del sistema-esperto medico, il danno è possibile solo in seguito ad una decisione del medico (o del paziente) presa in base ad informazioni inesatte.

Mi sembra che solo nel caso in cui non vi sia intervento decisionale da parte dell'uomo, sia stata prevista e si dovrebbe applicare la responsabilità senza colpa, purché esista un nesso diretto di causalità tra il difetto di un prodotto e un danno.

Nella prima ipotesi, in caso di informazioni inesatte fornite dal *software* ad un essere umano, non è il prodotto stesso che può causare dei danni (contrariamente a quanto avviene per il *software* che aziona il *robot*), è l'informazione contenuta nel « prodotto » o fornita da quel prodotto che, attraverso l'interpretazione e la mente umana, porta ad una decisione sbagliata e, talora, ai danni.

Per tornare all'esempio del sistema-esperto medico, in caso di errore nel *software*, si possono avere una diagnosi sbagliata e danni fisici per il malato che ne

seguisse le indicazioni. Possiamo, in questo caso, imporre al produttore una responsabilità senza colpa? Se sì, si può allora imporre la stessa responsabilità agli editori di opere mediche (scientifiche o di divulgazione), poiché la situazione è simile: si tratta di un supporto che contiene informazioni destinate ad aiutare un utente nel prendere delle decisioni. Si arriverebbe allora, attraverso un testo sulla responsabilità dei produttori, ad un sistema di responsabilità senza colpa del fornitore di informazioni, applicabile ai sistemi-esperti, ma anche ai libri, alla stampa, ecc. ...

Ora, da una parte, è chiaro che la direttiva non si prefiggeva all'inizio questo obiettivo. E, dall'altra, non mi sembra che la si possa interpretare in modo ampio da poter così stabilire un sistema di responsabilità rigida per le informazioni difettose.

Una tale responsabilità costituirebbe una modificazione talmente importante dei principi attualmente in vigore che mi sembra che solo il legislatore potrebbe introdurla (in seguito ad un ampio scambio d'idee sull'argomento) e che in ogni caso, questa riforma non può nascere dalla sola interpretazione, da parte dei giuristi, di un testo che non aveva assolutamente come scopo una tale riforma.

Quali sono allora le *conseguenze di questa distinzione* sul regime di responsabilità del produttore e sulla possibilità di clausole esonerative? Si può tornare sulla distinzione nel modo seguente, a seconda che il danno sia causato dal contenuto informatico del programma (I) o che sia causato direttamente (II).

(I) In certi casi (è l'esempio del sistema-esperto medico oppure della banca dati), il danno non può essere che la conseguenza di una decisione umana. Questi danni, a mio avviso, non sono coperti dalla direttiva. Quindi, resta possibile scartarli grazie ad una clausola esonerativa dalla responsabilità. In questo caso, vi è una fornitura di informazioni: la responsabilità che entra in gioco è la responsabilità tradizionale in base alla colpa. Tuttavia, bisogna considerare l'importanza della distinzione tra responsabilità oggettiva e responsabilità tradizionale in due modi:

1) anche per il diritto comune, le clausole esonerative dalla responsabilità sono sottoposte a certi limiti;

2) una responsabilità basata sulla colpa può essere molto severa per il produttore; progressivamente, la giurisprudenza (soprattutto in Belgio) si è evoluta su questo punto ed è giunta talvolta a risultati abbastanza simili a quelli cui si giungerebbe con un sistema di responsabilità oggettiva. Nel caso dell'informazione, più questa ha un carattere pratico e più si ritiene che essa debba costringere il destinatario ad un comportamento preciso, più l'obbligo del fornitore di informazioni si rivelerà pesante; resta tuttavia la possibilità di esonerarsi da certi casi di responsabilità.

(II) D'altronde, lo si è visto, alcuni danni possono derivare direttamente dal funzionamento difettoso del *software* in un *robot* (o un riscaldamento centrale, o un apparecchio medico): questi danni non possono essere evitati grazie ad una clausola esonerativa. Sta in questo l'effetto più sensibile della direttiva. Il numero di incidenti di questo tipo rischia di aumentare con l'utilizzazione dell'informatica e dei microprocessori nella vita di tutti i giorni.

Mi sembra che l'applicazione dei principi della direttiva sia, in questo caso, cosa abbastanza logica: man mano che l'utilizzazione dei *software* si banalizza e si moltiplica, non vi è ragione perché si faccia una differenza tra questi prodotti e gli altri prodotti tradizionali; se le ragioni che giustificano l'applicazione di una responsabilità senza colpa sono valide per i prodotti tradizionali, lo sono anche per i prodotti che incorporano un *software*.

Per la vittima, in caso d'incidente, il risultato è lo stesso; e non si capisce perché si dovrebbe concedere ad una categoria di produttori (i produttori di *software*) un regime più favorevole rispetto a quello cui sono soggetti i produttori di prodotti più tradizionali come le medicine e le automobili. Tanto più vero se la vittima potrà perseguire il fabbricante del riscaldamento, o dell'ascensore, che incorpora un *software* o un microprocessore (se i giudici considerano difettoso l'apparecchio). Se si dice che il *software* non è coperto dalla direttiva, si intende che il fabbricante di prodotti finiti che incorporano un *software* supporterà solo le conseguenze dei difetti di quel *software*, escludendo la responsabilità del produttore del *software*. È chiaro quindi che

una tale soluzione sarebbe sfavorevole non solo per i consumatori, ma anche per molti produttori che esiteranno (lo si potrebbe temere) a ricorrere a queste nuove tecnologie.

Un *software* può dunque produrre due tipi di risultati differenti: *informare una persona* e aiutarla a prendere una decisione o *azionare una macchina*.

Ci troviamo nuovamente di fronte ad un carattere ibrido del *software* che può giocare il ruolo di un testo (fonte d'informazione) o il ruolo di un utensile (che aziona il *robot* o l'apparecchiatura medica); ma questa volta, siamo nel campo dei risultati che possono essere forniti dal *software*. Avevamo menzionato prima il carattere ibrido del *software* per spiegare che non bisogna farsi ingannare dal fatto che un programma sia quasi certamente « scritto » da un programmatore, ma che portava sempre a risultati concreti per la macchina (*computer* o *robot*). Si vede, in questa fase, che anche se c'è sempre un effetto concreto e materiale (informazioni che appaiono sullo schermo o movimenti del *robot*, bisogna distinguere secondo la possibilità concreta di causare un danno: sia direttamente (*robot*), sia indirettamente (informazioni per l'utente)?

Una obiezione che si potrebbe muovere a questo ragionamento è che la stessa direttiva non precisa il carattere (diretto o indiretto) del nesso di causalità. Si è dovuto quindi guardare più lontano e analizzare le conseguenze che deriverebbero da questa o quella concezione della causalità. Ora, a nostro avviso, la direttiva non va applicata alla fornitura di informazioni (per le ragioni che ho indicato); dunque, per evitare questa conseguenza, bisogna limitare l'applicazione della direttiva ai danni causati direttamente dal « *software*-utensile » (in un *robot*) ed escludere i danni causati dal « *software*-informazione » (o piuttosto dal contenuto informatico del *software*).

Per i prodotti come i libri o i *software* che trattano l'informazione, bisogna dunque fare una distinzione tra il supporto e il contenuto e considerare, secondo me, che la direttiva non prevede che i danni causati dal supporto (direttamente). Invece, per gli altri programmi, questa distinzione non è valida: il supporto e le istruzioni sono inseparabili. Il modo in cui le istruzioni vengano redatte, e il loro contenuto, vanno ad azionare diretta-

mente il *robot*, così come lo farebbe un ingranaggio o un pezzo meccanico: le istruzioni sono come la forma dell'utensile.

Per capire meglio la distinzione tra i danni causati dal supporto e i danni causati dal contenuto, prendiamo l'esempio del libro:

— Che tipo di danni può causare il contenuto? Prendiamo per esempio il caso successo in Francia (affare Nathan), dove una guida sui funghi commestibili conteneva alcune imprecisioni. La conseguenza: parecchie persone sono rimaste avvelenate.

— Quale tipo di danni potrebbero essere causati dal supporto? L'ipotesi, più teorica, è quella del libro « Il Nome della Rosa » dove è l'inchiostro del libro ad essere avvelenato e a provocare la morte del lettore.

Per il *software*, si è visto che esso poteva provocare danni di due tipi; per dimostrarlo, si può dire che un *software* può darmi ad esempio la ricetta per la torta di mele (ed è così fonte di informazioni), ma che un *software* introdotto in un *robot* specializzato potrebbe anche comandare al *robot* di fare la torta di mele (ed è allora utensile). Nel primo caso, ottengo delle informazioni, nel secondo caso, ottengo un risultato tangibile.

Il ragionamento che abbiamo seguito per i sistemi-esperti può essere adottato alla banca dati; alcuni sostengono che la direttiva andrebbe applicata in caso di informazioni erronee contenute nella banca dati. Non credo che si possa seguire questo ragionamento: ancora una volta, si tratterebbe allora di una responsabilità senza colpa per informazioni difettose.

Ultima osservazione: la distinzione è più facile in teoria che in pratica ed esistono in realtà due situazioni miste: pensiamo ad un *software* che abbia come funzione di controllare lo svolgimento di un processo di fabbricazione, di avvertire in caso di pericolo e di fornire le indicazioni necessarie per evitare un incidente. In questo caso, il *software* fornisce certe informazioni ad un essere umano, ma si tratta di un *software* strettamente legato ad un altro prodotto; in un certo senso non fa altro che dare le istruzioni per l'uso, e si può dire che esso fa parte della « presentazione del prodotto » art. 6.1.a). Quindi, nel campo dei *software* che forniscono informazioni (e che,

secondo me, non sono coperti dalla direttiva — o meglio i danni da essi causati non sono previsti dalla direttiva), bisogna distinguere quelli che sono associati all'utilizzazione di un altro bene e del quale costituiscono un elemento (che ne migliora la sicurezza): per questi *software*, anche se si tratta di informazioni, ci si trova nella sfera di applicazione della direttiva. È chiaro quindi che la distinzione tra le due categorie di *software* (e di danni) lascia spazio ad una serie di casi intermedi.

Riassumendo, così come la definizione del danno coperto dalla direttiva, anche la condizione di causalità finisce con il limitarne le applicazioni possibili.

Quando la vittima giunge a provare il difetto (e abbiamo visto le imprecisioni che sono legate a questa nozione), il danno (e abbiamo visto che in molti casi la direttiva non potrebbe essere applicata) e il nesso di causalità (e abbiamo visto che questa condizione limitava anche il campo della direttiva), deve ancora stabilire due cose di cui ci occuperemo ai punti IV e V (per i quali ci limiteremo ad alcune osservazioni).

3.4. Determinazione del produttore responsabile. — Si potrebbe creare un problema nel caso di un *software* concepito in modo che l'utente lo possa adattare alle proprie necessità perché in caso di difetto, bisognerà ritrovare la sua origine: sia nel *software* che è stato venduto, sia nelle adattazioni fatte dall'utente. Si dovrà necessariamente ricorrere ad una perizia; ma bisogna sottolineare che la direttiva stabilisce una presunzione a carico del produttore: è lui che dovrà dimostrare che il difetto è subentrato dopo l'immissione sul mercato del prodotto, perché si deve presumere che il difetto esistesse dopo la partenza (art. 7 b). Se riesce a dimostrarlo, si avrà la riduzione o la soppressione della responsabilità (art. 8).

3.5. Data dell'immissione sul mercato, da cui decorre un termine di dieci anni. Può crearsi una difficoltà perché i *software* vengono regolarmente migliorati e nuove versioni vengono distribuite nel quadro dei contratti di manutenzione; se l'utente si appoggia ad una rete, la nuova versione può essere installata senza intervento delle parti, attraverso la telecomunicazione.

4. POCHI CENNI DI CONCLUSIONE

La direttiva è applicabile al *software* perché questo è un bene materiale. E si possono ignorare tutte le obiezioni all'applicazione della direttiva. La Commissione ha confermato il principio dell'applicazione al *software*.

I Tribunali si riservano un grande potere di valutazione per quel che riguarda la nozione di difetto del *software* (questioni di opportunità).

Il sistema di difesa dai rischi di sviluppo dovrebbe permettere al produttore di eludere la responsabilità, a condizione che egli possa provare di aver effettivamente effettuato numerosi test (altrimenti, egli potrebbe essere ritenuto responsabile in base alla colpa).

I casi più comuni e più costosi di danno, cioè i danni finanziari o commerciali di un'impresa, non sono contemplati dalla direttiva: a questo riguardo, restano valide le clausole esonerative.

Infine, bisogna distinguere due tipi di *software*, in base alla funzione che essi svolgono (sia fornire informazioni, sia azionare un *robot*). Questa distinzione permette, io penso, di limitare l'incidenza della direttiva ai campi ai quali era stata destinata all'inizio, cioè i danni causati da un bene tangibile, ma non i danni causati da un bene intangibile quale può essere un'informazione: anche in quel caso restano valide le clausole esonerative.

Dunque la direttiva, o meglio la responsabilità senza colpa e l'invalidità delle clausole esonerative, non saranno applicate che in alcuni casi; molti esempi di danni causati dal *software* continueranno ad essere trattati dal diritto comune.

Quanto all'importanza del cambiamento che si verificherà in seguito all'introduzione della direttiva, possiamo ripetere (sia per il *software* che per gli altri prodotti) che la si può ridimensionare:

— prima di tutto perché in materia di clausole esonerative, già attualmente esistono dei limiti alla loro validità;

— ma soprattutto perché le nozioni di colpa (nella responsabilità extra-contrattuale) e di vizio occulto (nella responsabilità contrattuale) sono talvolta (e sempre più) interpretate dai giudici in

un modo che si avvicina ai principi della direttiva.

Per terminare, bisogna precisare che ci siamo limitati ad un'analisi giuridica, senza parlare a lungo dei problemi di opportunità. Ciò che abbiamo cercato di capire è: visto che la direttiva può essere applicata ai software, quando la si potrà applicare e a quali condizioni?

Il problema che riguardava l'opportunità o meno di applicare, in questo settore, una responsabilità senza colpa non è stato analizzato a fondo per due ragioni:

— perché è un problema « politico » e quindi ognuno può avere la sua opinione a difendere un'opinione piuttosto che un'altra;

— ma soprattutto perché è troppo tardi per porsi il problema: in effetti, la risposta è stata data implicitamente al momento dell'adozione della direttiva nel 1985 ed avendo scelto l'opzione della responsabilità senza colpa. È troppo tardi, ci sembra, per rimetterla in discussione.

JEAN-PAUL TRIAILLE

BIBLIOGRAFIA

Direttiva del Consiglio del 25 luglio 1985 relativa al riavvicinamento delle disposizioni legislative, regolamentari e amministrative degli Stati membri in materia di responsabilità per danno da prodotti difettosi (85/374/CEE), in *JOCE*, legge n. 210/29 del 7 agosto 1985.

Domanda scritta n. 706/88 di Gijs de Vries posta alla Commissione delle Comunità Europee (5 luglio 1988); risposta data da lord Cockfield a nome della Commissione (15 novembre 1988), in *JOCE*, n. C. 114/42 del 8 maggio 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. G. Vandenberghe, Kluwer, Computer/Law Series, n. 3, 1989.

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

DOMMERING VAN RONGEN L., *Poduktaansprakelijkheid en software*, in *Computerrecht*, 1988, p. 230.

DE RAADT W.A.P., capitolo sulla responsabilità del produttore, in *Hoofdstukken Informaticarecht*, ed

F. De Graaf, Samson, Alphen aan den Rijn, 1989.

POULLET Y., *Les contrats télématiques face au droit communautaire*, in *Lamy Droit de l'Informatique*, 1988, n. 1, p. 22.

ZOPPINI A., *Les systèmes-experts et la responsabilité civile*, in *Droit de l'Informatique et des Télécoms*, 1989/1, p. 54.

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

MORTIMER H., *Computer-aided medicine: present and future issues of liability*, in *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 dossier of the Institute of International Business Law and Practice, ICC, 1988.

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

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

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