

EUI WORKING PAPERS

LAW No. 2006/25



L'accesso al codice sorgente:
Alcune considerazioni su libertà,
conoscenza e concorrenza
in margine al caso Microsoft.

GIOVANNI SARTOR AND
GUIDO SCORZA



EUROPEAN UNIVERSITY INSTITUTE

Department of Law

EUROPEAN UNIVERSITY INSTITUTE
DEPARTMENT OF LAW

*L'accesso al codice sorgente: Alcune considerazioni su libertà, conoscenza e concorrenza in
margine al caso Microsoft.*

GIOVANNI SARTOR AND GUIDO SCORZA

EUI working paper **LAW** No. 2006/25

This text may be downloaded for personal research purposes only. Any additional reproduction for such purposes, whether in hard copy or electronically, requires the consent of the author(s), editor(s). If cited or quoted, reference should be made to the full name of the author(s), editor(s), the title, the working paper or other series, the year, and the publisher.

The author(s)/editor(s) should inform the Law Department of the EUI if the paper is to be published elsewhere, and should also assume responsibility for any consequent obligation(s).

ISSN 1725-6739

© 2006 Giovanni Sartor and Guido Scorza

Printed in Italy
European University Institute
Badia Fiesolana
I – 50016 San Domenico di Fiesole (FI)
Italy

<http://www.iue.it/>
<http://cadmus.iue.it/dspace/index.jsp>

ABSTRACT:

Access to software is considered, with regard to the discipline provided by Art. 64 quater of the Italian Copyright law (which provides for a limited permission to decompile software). It is argued that in the proceedings stated by UE Commission against Microsoft a different approach emerges, according to which not only it is licit to decompile, even without the permission of the copyright holder, but the latter has the positive duty to provide sufficient information.

KEYWORDS:

Intellectual Property, Copyright

Contents

1	Introduzione	1
2	Il software tra codice sorgente e codice eseguibile	2
3	L'accesso al codice sorgente nella disciplina vigente	3
4	L'apertura del software	5
5	Il caso Microsoft	6

L'accesso al codice sorgente: Alcune considerazioni su libertà, conoscenza e concorrenza in margine al caso Microsoft.*

Giovanni Sartor
Guido Scorza

Abstract

Access to software is considered, with regard to the discipline provided by Art. 64 quater of the Italian Copyright law (which provides for a limited permission to decompile software). It is argued that in the proceedings stated by UE Commission against Microsoft a different approach emerges, according to which not only it is licit to decompile, even without the permission of the copyright holder, but the latter has the positive duty to provide sufficient information.

1 Introduzione

Sino a qualche anno fa il dibattito sulla disciplina giuridica del software appariva relegato alla storia del diritto dell'informatica, della quale aveva occupato qualche pagina sul finire degli anni '70, allorquando si esclude che i programmi per elaboratore costituissero invenzioni industriali, e nei primi anni '90, dopo che il nostro legislatore (seguendo le indicazioni del legislatore europeo) inserì gli stessi tra le opere dell'ingegno protette dal diritto d'autore (D.Lgs. 29 Dicembre 1992, n. 518).

Da allora, il giurista ha rivolto la sua attenzione quasi esclusivamente al tema della repressione delle riproduzioni non autorizzate (la cd. pirateria informatica), contro le quali il legislatore predisponendo disposizioni via via più efficaci, inasprendo le pene e ampliando l'ambito di applicazione delle sanzioni. La riflessione giuridica si è invece sostanzialmente disinteressata dell'adeguatezza funzionale e sistematica della scelta di considerare il software quale opera letteraria, e quindi di affidare alla disciplina autorale (legge 22 aprile 1941, n. 633) - pur con alcune norme sui generis - il compito di regolare la tutela e della circolazione dei relativi diritti (anche perché tale scelta è ribadita dalle normative di tutti i paesi avanzati, dai trattati internazionali, e dal legislatore europeo).

Negli ultimi anni, tuttavia, il dibattito sulla disciplina del software è stato ravvivato da fenomeni nuovi, diversi ma non privi di momenti di contatto: la proposta di una direttiva europea sulla brevettabilità delle invenzioni realizzate attraverso programmi per elaboratori, la crescita del fenomeno dell'open source, la diffusione dei contenuti digitali, le grandi controversie giudiziarie

*Supported by the EU projects ONE-LEX (Marie Curie Chair) and ALIS (IST-2004-027968). To be published in "Il diritto dell'Internet".

attenenti alla concorrenza nel mercato dell'informatica (che vendono Microsoft opporsi in Europa alla Commissione Europea e negli Stati Uniti d'America SCO opporsi a IBM ed ad altri leader di mercato vicini al mondo open source).

Gli studiosi del diritto dell'informatica hanno quindi ripreso ad occuparsi della software con ritrovato interesse, dando vita ad un dibattito vivace e stimolante, che unisce diversi temi del diritto: dalle libertà fondamentali alla proprietà intellettuale, dal diritto civile a quello della concorrenza.

A voler rintracciare un filo conduttore di tali questioni, non vi è alcun dubbio che esso debba essere rintracciato nel tema dell'accessibilità al codice sorgente.

2 Il software tra codice sorgente e codice eseguibile

Com'è noto il software viene di regola commercializzato in forma compilata: l'utilizzatore non acquisisce il programma informatico nella forma in cui è stato originariamente predisposto dall'autore, il c.d. codice sorgente, ma riceve la traduzione del codice sorgente in un c.d. codice eseguibile (o codice oggetto), che può essere elaborato automaticamente, ma non è comprensibile né modificabile da parte dell'uomo. Il passaggio inverso (la ricostruzione del sorgente a partire dall'eseguibile), la c.d. decompilazione, è operazione estremamente difficile, possibile solo in misura limitata con riferimento a software complessi. Inoltre la decompilazione è resa ancora più difficile dall'adozione di svariate tecniche di protezione (come quelle crittografiche).

Quindi, di regola, chi fruisce di un software non è in grado di comprenderlo, di esaminarne il contenuto (le istruzioni di programmazione, e la loro organizzazione secondo strutture e funzioni), e quindi è incapace di coglierne con precisione il funzionamento interno. Di conseguenza all'utilizzatore - anche se in possesso di buone competenze informatiche - è preclusa la piena comprensione delle dinamiche di funzionamento attuali e potenziali del software. Il software è una scatola nera, un oggetto opaco (non analizzabile, non "smontabile"), cui l'utente ha accesso solo attraverso la descrizione contenuta nei manuali d'uso e attraverso il comportamento osservabile (attraverso i messaggi visivi che il software stesso invia allo schermo o ai messaggi sonori che trasmette all'altoparlante).

L'opacità del software acquista un rilievo crescente in seguito all'accelerato progresso dell'informatizzazione. Sempre più viviamo in un mondo artificiale, dove interagiamo con artefatti tecnologici, e sempre più tali artefatti (per l'ufficio, lo svago, la casa, le cure mediche, la comunicazione a distanza, la memorizzazione e l'analisi dei dati, la gestione delle infrastrutture, il controllo dei processi di lavoro, ecc.) sono governati dal software. Gli artefatti tecnologici (e quindi il software che li governa) determinano gli effetti dei nostri comportamenti, condizionano le nostre possibilità stesse di agire e comunicare: nel mondo dell'artificiale, al funzionamento della natura (secondo le "istruzioni" della causalità naturale, immutabilmente scritte nel libro della natura, come diceva Galileo Galilei) si sovrappone sempre più il funzionamento dei dispositivi tecnologici (secondo le "istruzioni" dei programmi informatici che li governano, scritte dai programmatori che quei programmi hanno realizzato).

Nella società informatizzata l'inconoscibilità del software (cioè l'impossibilità accedere ad una sua presentazione accessibile alla mente umana, in forme che ne articolino la complessità rendendola comprensibile: istruzioni e dati di alto livello, accorpati in procedure, oggetti, metodi di elaborazione, protocolli di comunicazione, architetture) preclude la conoscenza del contesto dell'azione individuale e sociale.

Non sarà inutile collocare questa esperienza della nostra modernità nel quadro interpreta-

tivo offerto da alcuni riferimenti alla tradizione filosofica. Ricordiamo come il filosofo Hegel ravvisasse la dinamica dello sviluppo sociale e culturale nella oggettivazione dello spirito umano, cioè nella trasformazione del pensiero dell'uomo in realtà sociali ed istituzionali, e nella corrispondente autocoscienza, cioè nel ricondurre tali realtà al pensiero stesso che le ha originate. Nella società dell'informatica il processo dialettico di Hegel sembra destinato ad interrompersi: l'opacità del software esclude per sempre l'autocoscienza, la capacità dello spirito umano di ritrovare se stesso nelle proprie oggettivazioni e quindi di acquisirne consapevole controllo. Ricordiamo ancora che come insegnava Spinoza, la conoscenza è libertà: come chi non conosce il funzionamento della natura non è in grado di operare efficacemente in essa, così chi non conosce il funzionamento del software non è in grado di determinarsi liberamente nel mondo dell'artificiale. Chi non conosce il funzionamento del software non è in grado di conoscere gli effetti delle proprie azioni, e quindi non è in grado di effettuare e realizzare in libertà le proprie scelte (Che accadrà quando clicco su questa icona? Scaricherò una traccia musicale o un virus informatico? Attiverò involontariamente l'invio dei miei dati personali? Mi precluderò l'utilizzo di software che sto già utilizzando?). E' vero che anche i codici sorgenti restano incomprensibili per chi non abbia conoscenze informatiche adeguate, e che comunque il singolo individuo non potrebbe mai trovare il tempo e le energie per studiare tutti i software con i quali interagisce. Tuttavia, se il software fosse conoscibile (venendo fornito nel codice sorgente, accompagnato dalla relativa documentazione tecnica) le indicazioni sul suo funzionamento potrebbero essere fornite da specialisti imparziali, mediante consulenze informatiche. Inoltre lo studio del software potrebbe essere oggetto di un ampio dibattito scientifico, i cui risultati, pur frutto del dialogo degli esperti, sarebbero a tutti accessibili (in particolare per quanto attiene alla diffusione di conoscenze attinenti a disfunzioni e inganni nascosti nel codice).

L'inconoscibilità del software, oltre a incidere sulla libertà di autodeterminarsi incide negativamente sulla libertà di ricerca scientifica (alla quale viene escluso una componente fondamentale della nostra cultura e della nostra società), e pregiudica il funzionamento della concorrenza. Quest'ultima è già ristretta dal c.d. effetto di rete: i software di maggior successo tendono a prevalere perché il costo marginale di una copia è nullo o quasi (e ciò avvantaggia chi distribuisce volumi più ampi) e perché l'esigenza di comunicare e condividere le informazioni elaborate spinge tutti a convergere sui prodotti maggiormente diffusi. L'inconoscibilità del software allarga verticalmente gli spazi del monopolista: chi già utilizza un programma è maggiormente interessato ai software compatibili con questo (capaci di cooperare con esso, fornendogli dati e acquisendone i risultati: si pensi a come una videoscrittura possa interoperare con i programmi per gestire database, fogli elettronici, posta elettronica, agenda, ecc.) e la realizzazione di software compatibili con un programma dato (che in ipotesi già domina il mercato) richiede la conoscenza di quest'ultimo, conoscenza che deve essere aggiornata in occasione di ogni modificazione dello stesso, in seguito all'uscita di una nuova versione, di un aggiornamento o anche solo di un rattoppo (patch) che rimedia ad un errore di programmazione. Quindi il monopolista, precludendo o limitando l'accesso alle informazioni sul proprio software, può riservare a sé il mercato dei programmi compatibili, o può limitarlo selettivamente ai produttori amici, che si impegnino a non minacciare il monopolista e anzi a servire le sue politiche.

3 L'accesso al codice sorgente nella disciplina vigente

Dopo questa breve presentazione dei problemi inerenti alla conoscibilità del software, passiamo ad esaminare la disciplina nella quale essi vanno inquadrati. In astratto tre scelte sono possibili:

(1) un atteggiamento favorevole all'apertura, che imponga la visibilità del software, richiedendo la comunicazione del sorgente (e della relativa documentazione); (2) un atteggiamento neutrale, che ammetta la liceità del tentativo di conoscere il software (risalendo dall'eseguibile al sorgente), ma senza imporre l'obbligo di fornire tale conoscenza (il cui disvelamento verrebbe a dipendere dall'esito dello scontro tecnologico tra chi vuole nascondere e chi vuole svelare); (3) un'attiva tutela della chiusura, che non solo accetta che il software possa essere inconoscibile ma usa le sanzioni del diritto a garanzia di tale inconoscibilità. Quest'ultima è la scelta che emerge dall'art. 64 bis della nostra legge sul diritto d'autore, secondo il quale la traduzione del software - e quindi anche la sua decompilazione, cioè il tentativo di risalire dall'eseguibile al sorgente, ottenendo un risultato comprensibile alla mente umana - è soggetta all'autorizzazione del titolare (ed è quindi illecita in mancanza di tale autorizzazione). Non solo il diritto si astiene dall'imporre la comunicazione delle informazioni sul software, ma esso mette le proprie sanzioni al servizio del segreto sul sorgente.

L'unico limite a tale previsione si ritrova nell'art. 64 quater della legge sul diritto d'autore, secondo il quale:

“l'autorizzazione del titolare dei diritti non è richiesta qualora la riproduzione del codice del programma per elaboratore e la traduzione della sua forma ai sensi dell'art. 64 bis lettera a) e b), compiute al fine di modificare la forma del codice siano indispensabili per ottenere le informazioni necessarie per conseguire l'interoperabilità con altri programmi di un programma per elaboratore creato autonomamente”, a condizione che

“a) le predette attività siano seguite dal licenziatario o da altri che abbia il diritto di usare una copia del programma oppure per loro conto da chi è autorizzato a tal fine;
b) le informazioni necessarie per conseguire l'interoperabilità non siano già rapidamente e facilmente accessibili ai soggetti indicati alla lettera a);
c) le predette attività siano limitate alle parti del programma originale necessarie per conseguire l'interoperabilità.

Si noti che il 64 quater non impone al titolare del diritto sul software l'obbligo di fornire conoscenze su di esso, cioè l'obbligo di fornire il sorgente, neppure nei limiti necessari ad ottenere l'interoperabilità: esso si limita a riconoscere la liceità del tentativo di decompilare il software contro la volontà del titolare (tentativo spesso condannati all'insuccesso da ragioni tecnologiche), qualora tale tentativo sia finalizzato, appunto, al conseguimento dell'interoperabilità.

La lettura del combinato disposto del 64 bis e del 64 quater, già all'indomani dell'approvazione della Direttiva UE 91/250 e, più tardi, in occasione della sua attuazione nel nostro Paese attraverso il Decreto Legislativo 518/92 aveva lasciato perplessi in quanto in essa si registrava una rottura - più profonda che in relazione ad altri aspetti - tra la speciale legge a tutela dei programmi per elaboratore introdotta nella disciplina sul diritto d'autore ed i principi generali di tale sistema normativo.

Secondo tale normativa, il titolare del software gode di una privativa d'autore pur non ponendo integralmente a disposizione della collettività il proprio sforzo creativo: il codice sorgente, nel quale il risultato creativo viene presentato in forma comprensibile alla mente umana, non viene reso conoscibile alla collettività (anzi il tentativo di conoscerlo si scontra con le sanzioni del diritto). Si tratta di un'anomalia che all'interno della legge sul diritto d'autore si registra unicamente in relazione al software e che, all'epoca, fece dire ad autorevole dottrina che “sotto le vesti del diritto d'autore si” era “in sostanza introdotta un'esclusiva dal contenuto analogo a

quello delle privative industriali”¹. E' vero, tale anomalia si appoggia sulla particolare natura del software: da un lato esso può essere fruito senza aver coscienza del suo contenuto informativo (il suo utilizzo primario consiste nell'esecuzione automatica) e dall'altro eventuali software ispirati al sorgente originale (funzionamento analoghi ad esso, anche se leggermente diversi nella forma espressiva, in seguito a qualche ritocco) apparirebbero equivalenti al compratore. Tuttavia, la fornitura del software senza il sorgente (e senza la relativa documentazione) appare monca con riferimento ad uno dei valori fondamentali che ispirano la disciplina della proprietà intellettuale: lo sviluppo della conoscenza, e la garanzia delle libertà individuali e collettive che su questa si fondano.

L'accesso al codice sorgente è un presupposto necessario ed ineliminabile per la conoscenza dei principi e delle idee alla base di un programma per elaboratore ed è, pertanto, evidente che limitare la conoscibilità del codice significa, indirettamente, limitare anche la circolazione delle idee e dei principi sottostanti e, dunque, estendere la portata della privativa al di là di quelle che - stando alla richiamata previsione di cui all'art. 2, n. 8 - erano le stesse intenzioni del legislatore e comunque al di là dei limiti di elasticità del sistema del diritto d'autore, già messo a dura prova dal riconoscimento di privative su opere dell'ingegno dotate di una spiccata ed incontestabile vocazione all'utile più che al bello.

Tale singolarità della disciplina speciale sui programmi per elaboratore rappresenta un elemento di intrinseca instabilità capace di alterare - in modo talvolta sensibile - lo scambio di utilità tra autore e collettività sul quale, come è noto, riposa l'intero sistema del diritto d'autore.

In conclusione, la segretezza, giuridicamente garantita, del software (nella forma del codice sorgente), costituisce un importante elemento di rottura con il sistema del diritto d'autore nel quale pure trova ospitalità, e contraddice la definizione dell'oggetto della tutela d'autore che, a norma di quanto disposto dall'art. 2 n. 8 LDA, comprende i programmi per elaboratore in qualsiasi forma espressi (con la sola esclusione delle idee e dei principi che ne sono alla base). E' vero che il contenuto del diritto d'autore non rappresenta un'unità inscindibile, e che i diversi usi e le diverse componenti di un'opera protetta possono essere oggetto di specifiche e limitate autorizzazioni. Tuttavia, alla scissione dell'eseguibile dal sorgente si oppongono importanti ragioni, che fanno dubitare della sua legittimità, de jure condendo, e forse anche de jure condito: sul piano degli interessi privati, il nesso funzionale tra accesso al sorgente e piena e cosciente utilizzazione dell'eseguibile; sul piano degli interessi collettivi, il nesso funzionale tra tutela del sorgente e diffusione delle conoscenze in esso incorporate.

4 L'apertura del software

Come è noto, uno dei fenomeni più interessanti degli ultimi anni è consistito nella diffusione di nuovi modelli di distribuzione del software, modelli che spesso vengono identificati con la locuzione open source. E' appena il caso di rilevare che sotto questa dicitura si possono comprendere modelli produttivi e commerciali assai diversi in cui l'apertura riguarda solo la conoscibilità, in altri essa si estende alla cessione a terzi di copie del software (senza il consenso dell'autore o senza riconoscere ad esso un corrispettivo), all'utilizzo e all'ulteriore distribuzione, allo sviluppo di nuovi software che integrano o modificano il programma originario (tutte queste libertà sono riunite nel modello del c.d. software libero). E' opinione degli scriventi che queste diverse forme di apertura siano distinguibili tra loro, e che in particolare, l'apertura meramente conoscitiva,

¹V. Franceschelli, La tutela giuridica dei programmi per elaboratore, in Nuove leggi civili commentate, 1992, 260.

consistente nel conferire al licenziatario la concreta possibilità (non solo la mera liceità) di accedere al contenuto del programma meriti specifica attenzione. Questa forma di apertura è in fatti compatibile con la conservazione delle esclusive in capo al titolare del diritto d'autore: l'autore, anche quando conferisca al licenziatario il codice sorgente e la relativa documentazione, resta titolare esclusivo del potere di concedere ulteriori licenze (e quindi conserva intatti i propri diritti di sfruttamento economico). Si tratta di un'apertura limitata, che pare insufficiente a quanti si ispirano al modello del software libero, ma tuttavia non disprezzabile. In un modello pluralistico del mercato del software, in cui coesistano diversi modelli di produzione e di commercio, accanto all'idea radicale del software libero, ben può trovare spazio un pieno riconoscimento del diritto d'autore (una volta distinto dalla garanzia del segreto).

5 Il caso Microsoft

Non vi è dubbio che la questione dell'accessibilità al codice sorgente è al centro del caso Microsoft ovvero di una delle questioni antitrust più complesse ed articolate in ambito IT delle quali la Commissione Europea è stata chiamata ad occuparsi nell'ultimo decennio e che sembra, finalmente, pervenuta al suo epilogo.

La vicenda ha avuto inizio nel dicembre del 1998, allorquando la Sun Microsystems Inc. (società di diritto statunitense con sede a Palo Alto, in California) ha lamentato, dinanzi alla Direzione generale della Commissione UE, un presunto abuso di posizione dominante da parte della concorrente Microsoft. Più in particolare, stando a quanto riferito nella segnalazione della Sun, la Microsoft, operatore in posizione dominante nel mercato dei sistemi operativi per PC, avrebbe abusato di tale posizione. non divulgando talune informazioni relative ai propri programmi, informazioni indispensabili al fine di consentire ai concorrenti di realizzare soluzioni informatiche per server idonee ad interagire con detti sistemi operativi nell'ambito di reti aziendali².

Sulla scorta di tale segnalazione la Commissione ha dunque avviato un procedimento il cui oggetto è stato successivamente ampliato ricomprendendovi anche la presunta illegittimità della condotta commerciale tenuta dalla stessa Microsoft nell'inserire all'interno dei propri sistemi operativi il proprio lettore multimediale (Windows Media Player)³.

All'esito di tale procedimento, con Decisione del 24 marzo 2004, la Commissione Europea ha accertato che la Microsoft aveva effettivamente abusato della propria posizione dominante sotto entrambi i profili contestati:

(a) rifiutando di fornire informazioni necessarie per l'interoperabilità⁴ tra i propri prodotti e quelli sviluppati da terze parti e rifiutando di autorizzarne l'uso da parte di queste ultime al fine di consentire loro lo sviluppo e la distribuzione di soluzioni concorrenti nell'ambito del mercato dei sistemi operativi per server di gruppo di lavoro;

²Ricordiamo che il server è il computer che, nell'ambito di una rete informatica, offre servizi (coordinamento, fornitura di dati e software, archiviazione, posta elettronica, accesso ad internet, ecc.) agli altri computer della rete.

³Nel procedimento sono successivamente intervenuti: Association for Competitive technology (ACT), Time Warner Inc., Computer & Communications Industry Association (CCIA), Computing Technology Industry Association (CompTIA), Free software Foundation Europe (FSF Europe), Lotus Corporation, Novell Inc., RealNetworks Inc., Software & Informations Industry Association (SIIA).

⁴Tali informazioni sono definite nella decisione come Informations relatives à l'interopérabilité ovvero "les spécifications exhaustives et correctes de tous les protocoles implémentés dans les systèmes d'exploitation Windows pour serveurs de groupe de travail et qui sont utilisés par les serveurs de groupe de travail Windows pour fournir aux réseaux Windows pour groupe de travail des services de partage des fichiers et d'imprimantes, et de gestion des utilisateurs et de groupes, y compris les services de contrôleur de domaine Windows, le service d'annuaire Active Directory et le service "Group Policy".

(b) legando la fornitura del sistema operativo Windows all'acquisto contestuale del lettore multimediale Windows Media Player.

In ragione di tali violazioni la Commissione ha condannato la Microsoft al pagamento di una sanzione di 497196, 304 Euro nonché a porre fine alle infrazioni accertate. In particolare, per quanto concerne la prima delle due richiamate violazioni, la Microsoft doveva entro 120 giorni rendere disponibili le informazioni relative all'interoperabilità a tutte le imprese concorrenti intenzionate a sviluppare e distribuire sistemi operativi per server di gruppi di lavoro autorizzando tali imprese – secondo condizioni ragionevoli e non discriminatorie – all'utilizzo delle predette informazioni ai fini del conseguimento dell'interoperabilità.

A seguito della decisione, Microsoft ha quindi comunicato alla Commissione un insieme di documenti ed informazioni che, tuttavia, sono stati ritenuti dagli esperti nominati da quest'ultima insufficienti a consentire l'interoperabilità oggetto della decisione. Di conseguenza con una nuova decisione del 10 novembre 2005, la stessa Commissione ha rinnovato alla Microsoft l'invito ad adempiere alla precedente Decisione del marzo 2004 entro il successivo 15 dicembre con l'avvertimento che in difetto avrebbe proceduto ad irrogarle una sanzione sino a due milioni di euro per ogni giorno di ritardo.

Nei mesi scorsi, quindi, l'epilogo della vicenda allorquando la Microsoft ha comunicato alla Commissione di essersi determinata a porre a disposizione delle società concorrenti l'intero codice sorgente dei programmi oggetto della citata decisione attraverso uno speciale modello di licenza⁵.

Preso atto di tale volontà la Commissione ha, quindi, richiamato l'attenzione della Microsoft sulla circostanza che, in forza della Decisione del marzo 2004, essa non è obbligata a rendere accessibile tutto il proprio codice sorgente ma solo le informazioni necessarie a consentire ai concorrenti di perseguire le citate finalità di interoperabilità e che, d'altra parte, non può ritenersi pacifico che la messa a disposizione di detto codice valga a consentire a questi ultimi il raggiungimento di detta finalità.

Opportunamente la Commissione non ha adottato una nozione ideologica circa l'apertura del software, ma piuttosto ha messo in luce la connessione funzionale tra apertura e concorrenza: potrebbe non bastare l'accesso all'intero sorgente, quando per "sorgente" si intendesse sì l'insieme dei milioni di istruzioni di cui consta un programma complesso, ma quelle istruzioni non fossero presentate in forma adeguata, cioè aggregate ed illustrate in modo tale che sia possibile evincere le informazioni utili ai fini dell'interoperabilità. Si tratta, ovviamente, di una prospettiva limitata, che, coerentemente con i compiti che spettano alla Commissione, affronta solo uno dei valori su cui può incidere la chiusura del software (la concorrenza). Tuttavia si tratta di una profonda innovazione rispetto alla disciplina del 64 quater: non solo è lecito ai concorrenti cercare di "smontare" un eseguibile al fine di estrarne informazioni rilevanti per la realizzazione di prodotti compatibili, ma il produttore ha il dovere di consentire ai concorrenti l'accesso a tali informazioni. Alla mera liceità del tentativo di accedere alle informazioni necessarie per l'interoperabilità si sostituisce il diritto ad ottenerle dal produttore.

Allo stato, dunque, si è in attesa che l'esperto nominato dalla Commissione si pronunci, ancora una volta, sull'idoneità delle informazioni rese disponibili dalla Microsoft a consentire ai concorrenti di conseguire la predetta interoperabilità.

Risulta evidente – quale che sia l'epilogo della vicenda – che al centro della stessa vi è proprio la questione dell'accessibilità al codice sorgente e, più in generale, quella concernente la portata della privativa riconosciuta al titolare dei diritti sul software.

⁵Maggiori dettagli sulla licenza proposta dalla Microsoft sono pubblicati su <http://www.microsoft.com/resources/sharesource/licensing/windows.msp>.

La strada seguita da Microsoft nel tentativo di adeguarsi alle indicazioni provenienti dalla Commissione UE, cioè l'offerta di una licenza relativa all'utilizzo del proprio codice sorgente – salvo verificare gli esatti termini dell'accordo – potrebbe configurare un pentimento operoso in relazione all'interpretazione del significato e della portata della vigente disciplina della materia: la possibilità di acquisire, ad eque condizioni, una licenza d'uso relativa al sorgente, infatti, costituirebbe la prova che si è finalmente compreso che il codice sorgente costituisce una parte essenziale del software come opera dell'ingegno protetta ai sensi della legge sul diritto d'autore e che, pertanto, esso deve essere posto a disposizione della collettività a condizioni analoghe a quelle concernenti il codice oggetto, a pena di continuare a rendere asimmetrico ed instabile lo scambio di utilità tra autore e collettività da sempre alla base della disciplina sul diritto d'autore. Irrilevante, sul piano strettamente giuridico della garanzia della concorrenza, appare, invece, la circostanza che il codice sorgente venga posto in circolazione attraverso un modello di licenza e nell'ambito di un circuito diverso da quello riservato al software in codice oggetto. Tale scelta, infatti, rappresenta un modello di business evidentemente diverso da quello adottato nel mondo open source e recepito nella GPL ma, nondimeno potrebbe rivelarsi sufficiente a realizzare le esigenze di conoscibilità necessarie per la realizzazione di software compatibili, e quindi, suscettibile di incidere positivamente sull'assetto del mercato. Diversa potrebbe essere la valutazione per quanto attiene alle più ampie esigenze di libertà e diffusione della conoscenza sopra accennate, ma la loro discussione ci porterebbe al di là della prospettiva necessariamente adottata dalla Commissione Europea.