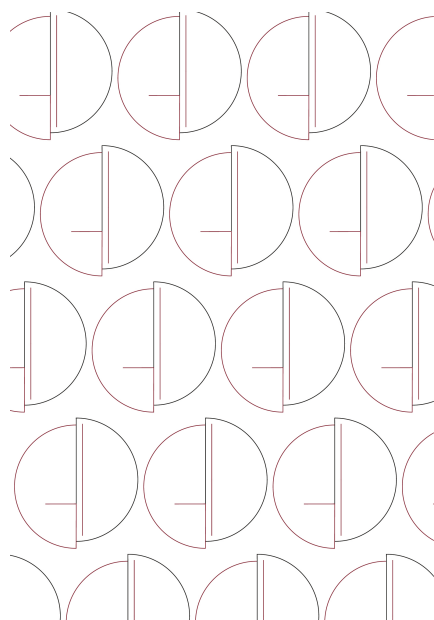


Annuario 2022 Osservatorio Giuridico sulla Innovazione Digitale

Yearbook 2022
Juridical Observatory on Digital Innovation

a cura di

Salvatore Orlando e Giuseppina Capaldo



Collana Materiali e documenti 90

Annuario 2022
Osservatorio Giuridico
sulla Innovazione Digitale

Yearbook 2022
Juridical Observatory on Digital Innovation

a cura di
Salvatore Orlando e Giuseppina Capaldo



SAPIENZA
UNIVERSITÀ EDITRICE
2022

Copyright © 2022

Sapienza Università Editrice

Piazzale Aldo Moro 5 – 00185 Roma

www.editricesapienza.it

editrice.sapienza@uniroma1.it

Iscrizione Registro Operatori Comunicazione n. 11420

Registry of Communication Workers registration n. 11420

ISBN 978-88-9377-256-3

DOI 10.13133/9788893772563

Publicato nel mese di dicembre 2022 | *Published in December 2022*



Opera distribuita con licenza Creative Commons Attribuzione –
Non commerciale – Non opere derivate 3.0 Italia e diffusa in modalità
open access (CC BY-NC-ND 3.0 IT)

*Work published in open access form and licensed under Creative Commons Attribution – NonCommercial –
NoDerivatives 3.0 Italy (CC BY-NC-ND 3.0 IT)*

Impaginazione a cura di | *Layout by:* Lucio Casalini e Enzo Maria Incutti

In copertina | *Cover image:* Michela Tenace, *Elaborazione grafica del logo OGDID/JODI, 2022, Archivio personale dell'a.*

Indice

Prefazione	7
1. Financial Markets and AI: the Algo-trading Regulation <i>Attilio Altieri</i>	9
2. Privacy Enhancing Technologies, trasparenza e tutela della persona nell'ambiente digitale <i>Alessandro Bernes</i>	23
3. Dati e identità personale. Note sparse a partire da una recente pronuncia del Consiglio di Stato <i>Lucio Casalini</i>	53
4. I procedimenti amministrativi di vigilanza sul mercato dei servizi digitali <i>Filippo D'Angelo</i>	73
5. Profili di tutela delle persone vulnerabili nell'ecosistema digitale. Il divieto di profilazione dei minori di età ai fini di marketing <i>Ilaria Garaci</i>	89
6. Diritti fondamentali e ambienti digitali: prime note di una ricerca sul diritto a non essere sottoposto a una decisione interamente automatizzata <i>Daniele Imbruglia</i>	113
7. La tutela giuridica del software: il caso Top System tra diritto di decompilazione e esigenze di conformità <i>Enzo Maria Incutti</i>	137

8. Platform economy e responsabilità delle piattaforme di intermediazione <i>Silvia Martinelli</i>	157
9. Neurorights. Una prospettiva di analisi interdisciplinare tra diritto e neuroscienze <i>Anita Mollo</i>	191
10. I sistemi di raccomandazione nelle interazioni tra professionisti e consumatori: il punto di vista del diritto dei consumi (e non solo) <i>Roberta Montinaro</i>	217
11. Linguaggi di programmazione e responsabilità <i>Salvatore Orlando</i>	267
12. L'intelligenza artificiale nel prisma dell'impresa: evoluzione normativa e prospettive di studio <i>Francesco Pacileo</i>	289
13. Trattamento dei dati personali e tutela dei minori <i>Federico Ruggeri</i>	325
14. Gli <i>smart contracts</i> nel settore finanziario: questioni irrisolte e prospettive regolatorie fra diritto nazionale e sovranazionale <i>Emanuele Tuccari</i>	343
Autori	367

7. La tutela giuridica del *software*: il caso *Top System* tra diritto di decompilazione e esigenze di conformità

Enzo Maria Incutti (Università di Roma La Sapienza)

7.1. Premessa: il caso *Top System*

Il presente studio prende spunto da una pronuncia della Corte di Giustizia dell'Unione Europea, che rappresenta l'occasione per riflettere sulla adeguatezza della tutela autoriale concessa al *software*, sui confini della autonomia privata e sulle potenziali prospettive evolutive della disciplina.

Il 6 ottobre 2021 la Quinta Sezione della Corte di Giustizia ha deciso sulla domanda di rinvio pregiudiziale promossa dalla *Cour d'appel de Bruxelles*, nell'ambito del procedimento *Top System SA vs. État belge* (C-13/20)¹, avente ad oggetto l'interpretazione dell'articolo 5, paragrafo 1², della direttiva 91/250/CEE del Consiglio, del 14 maggio 1991, relativa alla tutela giuridica dei programmi per elaboratore.

La controversia vedeva contrapporsi, da un lato, *Top System SA*, società di diritto belga che sviluppa programmi per elaboratore e fornisce prestazioni di servizi informatici e, dall'altro, *Selor*, l'ente pubblico belga responsabile della selezione e dell'orientamento dei collaboratori della pubblica amministrazione.

Su richiesta di *Selor*, *Top System* aveva sviluppato diverse specifiche applicazioni. Il 6 febbraio 2008 *Selor* e *Top System* avevano concluso un contratto avente ad oggetto l'installazione e la configurazione di un nuovo ambiente informatico di produzione, che, però, presentava

¹ Corte di giustizia dell'Unione Europea, 06 ottobre 2021, (causa C-13/20).

² «Salvo disposizioni contrattuali specifiche, non sono soggetti all'autorizzazione del titolare del diritto gli atti indicati nell'articolo 4, lettere a) e b), allorché tali atti sono necessari per un uso del programma per elaboratore conforme alla sua destinazione, da parte del legittimo acquirente, nonché per la correzione di errori».

subito dei difetti di funzionamento. Si noti a tal riguardo che Selor deteneva (e tutt'ora detiene) una licenza d'uso su tutti i programmi appositamente creati da Top System.

Il 6 luglio 2009 la Top System aveva proposto ricorso contro Selor e lo Stato belga dinanzi al *Tribunal de commerce de Bruxelles*, contestando l'illegittima attività di decompilazione effettuata da Selor sul programma per elaboratore e richiedendo il risarcimento dei danni per la decompilazione così effettuata.

Il 26 novembre 2009 la causa era stata rinviata dinanzi al *Tribunal de première instance de Bruxelles* che, il 19 marzo 2013, aveva dichiarato la domanda così proposta come infondata.

La *Top System* aveva, quindi, impugnato la sentenza di primo grado dinanzi al giudice del rinvio, la *Cour d'appel de Bruxelles*, sostenendo che la decompilazione possa essere effettuata solo in forza di un'autorizzazione dell'autore, o del suo avente diritto, o ancora a fini della c.d. interoperabilità - ovvero, l'interconnessione e l'interazione funzionale tra *software*.

Per contro, sulla base dell'interpretazione dell'art. 6 della Legge del 30 giugno 1994, che ha recepito nell'ordinamento belga la direttiva 91/250/CEE, Selor riteneva di essere legittimato a procedere alla decompilazione per correggere alcuni errori di funzionamento che rendevano impossibile un uso conforme alla destinazione del *software* e ad osservare, studiare e sperimentare il funzionamento del programma allo scopo di determinare le idee alla base delle funzionalità con l'obiettivo di prevenire le future interruzioni determinate da simili errori.

Alla luce dei presenti elementi di fatto ed avendo accertato l'avvenuta decompilazione da parte di Selor, la Corte d'Appello Belga aveva proposto, quindi, due differenti questioni pregiudiziali all'attenzione della Corte di Giustizia.

7.2. Alcune necessarie precisazioni in tema di decompilazione del *software*

Risulta opportuno ai fini di una corretta comprensione del fenomeno oggetto della controversia e, più in generale, del mondo informatico interessato dalla presente analisi, compiere alcune necessarie premesse.

In ambito informatico, per *software* si intende l'insieme composto del "sistema operativo" e di tutti i programmi in esso presenti³, mentre il programma per elaboratore è una sequenza di istruzioni⁴, espresse in linguaggio informatico⁵ che, per effetto del caricamento in un elaboratore⁶, interagendo con il sistema operativo di quest'ultimo comporta lo svolgimento di determinate attività⁷.

L'altro elemento determinante è il c.d. *hardware*, ovvero l'insieme dei dispositivi fisici costituenti un sistema elaborativo e/o operativo.

Il programma per elaboratore è di regola scritto dall'uomo nel c.d. codice sorgente per essere successivamente tradotto dall'elaboratore in una sequenza di *bit*⁸ che compongono il c.d. codice oggetto⁹.

Per poter essere intellegibile da parte dell'elaboratore, il codice sorgente (scritto in linguaggio «umano») deve essere tradotto nel c.d. codice oggetto, un linguaggio traducibile da parte della sola macchina.

In questa fase, quindi, le istruzioni di un programma scritto in un linguaggio di programmazione (sorgente) vengono trasformate in un codice oggetto (detto anche eseguibile) suscettibile di essere caricato ed eseguito da un elaboratore e dal relativo sistema operativo.

Entrando nel merito del processo informatico interessato dal presente giudizio, è essenziale delimitare i confini del processo di decompilazione.

La c.d. decompilazione va intesa come quella attività di *reverse engineering* che consiste nella ricostruzione di un prodotto già esistente,

³ Entità, quindi, ben più complessa rispetto al semplice programma che ne rappresenta solo una componente.

⁴ L'istruzione è una proposizione espressa sotto forma di codice diretta ad ordinare ad un elaboratore lo svolgimento di una determinata funzione o attività.

⁵ Il linguaggio è il sistema codificato di proposizioni che vengono espresse in forma di istruzioni

⁶ Il c.d. caricamento, consiste in quell'attività di inserimento di un programma nella memoria di un elaboratore.

⁷ Questa chiara distinzione tecnica non sembra, però, essere stata colta dal legislatore italiano che parla genericamente di programma per elaboratore, aprendo la porta a potenziali interpretazioni restrittive della disciplina. È indubbio che l'impropria restrizione terminologica non sia sinonimica di una concreta limitazione dell'ambito applicativo della normativa destinata ai *software*.

⁸ Sequenza che viene espressa in forma binaria di 0 e 1.

⁹ Dunque, il codice sorgente è scritto in un linguaggio di programmazione complesso ma comprensibile all'uomo (e non alla macchina); esso, infatti, deve essere tradotto dal compilatore in un linguaggio tecnico (il c.d. codice oggetto) attraverso la codificazione delle istruzioni in forma binaria.

in questo caso di un *software*. Il prodotto viene destrutturato per comprenderne l'architettura ed il suo funzionamento.

Lo scopo del *reverse engineering* in ambito *software* è quello di riprodurre il codice sorgente partendo dal codice oggetto di un programma esistente¹⁰. In questo modo è possibile ottimizzare il *software* correggendone gli errori di funzionamento, analizzare i programmi della concorrenza o sviluppare nuovi prodotti.

La ricostruzione di un *software* può riguardare diverse tipologie di attività, quali, a titolo esemplificativo, il recupero del codice sorgente del *software*; la comprensione delle regole di un protocollo di comunicazione; la creazione di un modello informatico; la ricerca dei c.d. *bug* nel programma; il miglioramento della compatibilità di un *software* con piattaforme e programmi di terze parti ed, anche, l'utilizzo di funzioni di piattaforme non documentate.

La decompilazione del *software* avviene di regola attraverso due specifiche modalità: a) attraverso la tradizionale decompilazione (c.d. *white box analysis*) del *software*; b) attraverso la tecnica della c.d. *black box analysis*, operazione consistente nell'osservazione del funzionamento del programma¹¹.

7.3. La tutela autoriale del *software*

L'esigenza di predisporre un'adeguata cornice di tutela per il *software* emerge nei primi anni '70 del secolo scorso, parallelamente all'evoluzione tecnico-informatica, trainata dai paesi occidentali¹².

¹⁰ «La decompilazione consente, in genere, non di ottenere il codice sorgente originale, bensì una terza versione del programma di cui trattasi, denominata “quasi-codice sorgente”, che potrà a sua volta essere compilata in un codice oggetto che consenta a tale programma di funzionare», così *Conclusioni Avvocato Generale, par. 41, Top System SA vs. État belge* (C-13/20).

¹¹ Molto spesso ciò avviene attraverso l'inserimento di *input* al fine di comprendere i corrispondenti *output* del programma. Questa distinzione informatica e procedurale risulta molto determinante sul piano normativo, come si vedrà più nel dettaglio successivamente.

¹² Sul punto, per un approfondimento v. L. MARTINAT e L. BOSSOTTI, *Tutela giuridica del software*, in S. VITRÒ (a cura di), *La tutela giuridica del software*, Key editore, 2022. p. 135 ss.

La ricerca di un corretto apparato normativo di tutela è stata, sin da subito, la principale e dirimente questione in materia¹³. Le due vie percorribili ma tra loro alternative – per approccio e per conseguenze – sono state individuate l’una nella tutela secondo il diritto d’autore e l’altra in quella brevettuale¹⁴.

A prevalere con nettezza è stata la prima opzione; pertanto, la tutela del *software* è tradizionalmente inserita nella disciplina del diritto d’autore.

Le ragioni giustificative di questa scelta risiedevano nella considerazione del *software* come creazione intellettuale e erano sorrette dal timore di vedere arrestato il processo tecnologico dalle restrizioni della tutela brevettuale.

Difatti, l’applicazione della disciplina brevettuale al *software*, in qualità di invenzione tecnica, avrebbe limitato notevolmente il progresso tecnico¹⁵ del mercato e avrebbe reso anche più complesse la commercializzazione e la diffusione dei programmi informatici a causa della rigidità del sistema dei brevetti.

Il diritto d’autore, invece, forniva - e fornisce tutt’ora - una soluzione più agevole sotto molteplici profili. Innanzitutto, si concede tutela sulla base della sola creazione del *software*, senza complesse procedure di “autorizzazione”, e si agevolano così i canali distributivi e commerciali connessi ai *software*.

Con il diritto d’autore, dunque, trovano tutela sia il codice sorgente nel particolare linguaggio con cui è stato scritto dal suo autore, e nello stesso modo è tutelato anche il codice oggetto ed il relativo materiale preparatorio¹⁶.

Non viene, invece, protetta l’idea che è alla base della creazione del *software*, permettendo di fatto che essa possa essere riutilizzata in un

¹³ Cfr. E. AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore. Profili e critica di una dicotomia normativa*, Giuffrè, Milano 2012; A.M. GAMBINO, *L’innovazione informatica tra brevettazione e diritto d’autore*, in *Dir. ind.*, 2010, II, p. 147 ss.; L. SCHIUMA, *Il software tra brevetto e diritto d’autore*, in *Riv. dir. civ.*, 2007, p. 683 ss.; G. DE SANTIS, *La tutela giuridica del software tra brevetto e diritto d’autore*, Giuffrè, Milano, 2000.

¹⁴ Per una dettagliata analisi e ricostruzione storico-giuridica, v. R. PARDOLESI e M. GRANIERI, *Il software*, in *AIDA*, 2007, p. 288-312, ed anche, G. NOTO LA DIEGA, *Le idee e il muro del suono. I programmi per elaboratore nella più recente giurisprudenza europea*, in *Europa e dir. priv.*, 2, 2013, p. 543 e ss.

¹⁵ Si tenga in considerazione la durata ventennale della tutela brevettuale con tutte le sue correlative limitazioni e restrizioni.

¹⁶ L. MARTINAT e L. BOSSOTTI, *op. cit.*, spec. p. 136.

numero indefinito di applicazioni a patto che venga diversamente espressa attraverso un autonomo codice sorgente¹⁷.

Nella prospettiva del sistema brevettuale avrebbero trovato tutela anche le idee e i principi che hanno dato origine all'invenzione informatica. Ciò che, invece, rileva ai fini della disciplina autoriale è l'espressione dell'idea, resa attraverso uno specifico programma informatico¹⁸.

Da un punto di vista legislativo, il primo intervento in questa direzione è ascrivibile al *Computer Software Copyright Act* statunitense¹⁹ del 1980²⁰.

In seguito, in Europa, è stata accolta la medesima impostazione, offrendo tutela autoriale per i *software* attraverso la Direttiva 91/250/CEE²¹ recepita, nel nostro ordinamento, dal D. Lgs. n. 518 del 1992.

In Italia quindi, a norma dell'art. 2, co. 1, n. 8 della legge sul diritto d'autore²², sono protetti «i programmi per elaboratore, in qualsiasi forma espressi purché originali quale risultato di creazione intellettuale dell'autore». Coerentemente alla *ratio* della disciplina autoriale,

¹⁷ Sul punto, in giurisprudenza, v. Trib. Bari Sez. IV, 14-3-2007, in *Dir. internet*, 2007, p. 447 ss., con nota di E. PELINO.

¹⁸ Come si legge in una importante pronuncia della Corte di Giustizia dell'UE, «[...] il vantaggio principale della tutela dei programmi per elaboratore mediante il diritto d'autore risiede nel fatto che essa concerne soltanto l'espressione individuale dell'opera e offre quindi uno spazio sufficiente a permettere ad altri autori di creare programmi simili, o perfino identici, purché essi si astengano dal copiare [...]» (cfr. Corte eur. giust. grande sez., C-406/10 del 2 maggio 2012, § 41).

¹⁹ *Computer Software Copyright Act*, L. n. 96-517, 94 Stat. 3015, 3028 (1980). Sul punto per una revisione critica, RICHARD H. STERN, *Another Look At Copyright Protection of Software: Did the 1980 Act Do Anything For Object Code?*, in *Computer L.J.*, 3, 1, 1981.

²⁰ Attualmente, negli Stati Uniti d'America la decompilazione dei software è fatta rientrare nella c.d. *fair use exception*, sulla base del Paragrafo 107 del Copyright Act. Si ritiene infatti, che l'utilizzo delle copie intermedie ai fini di *reverse engineering* sia pienamente legittimo anche in assenza di preventiva autorizzazione. Questa linea interpretativa sembra confermata dalla sentenza della Corte Suprema n. 18-956 del 5 aprile 2021, nel caso *Oracle vs. Google*.

²¹ Direttiva 91/250/CEE del Consiglio, del 14 maggio 1991, relativa alla tutela giuridica dei programmi per elaboratore. A conferma del paradigma autoriale di tutela messo precedentemente in evidenza, il secondo paragrafo dell'art. 1 recita: «la tutela ai sensi della presente direttiva si applica a qualsiasi forma di espressione di un programma per elaboratore. Le idee e i principi alla base di qualsiasi elemento di un programma per elaboratore, compresi quelli alla base delle sue interfacce, non sono tutelati dal diritto d'autore a norma della presente direttiva».

²² Legge, 22/04/1941 n. 633 (nel corso del presente studio, in via di semplificazione, anche l.d.a.)

restano, invece, esclusi dalla tutela accordata ai *software*, le idee e i principi che stanno alla base di qualsiasi elemento di un programma, compresi quelli alla base delle sue interfacce. È, invece, compreso nel termine «programma per elaboratore» il materiale preparatorio per la progettazione del software.

Come si vedrà in dettaglio successivamente, anche all'autore del programma per elaboratore vengono concessi diritti di utilizzazione economica (artt. 64-*bis* e ss. l.d.a.).

Sul piano internazionale, si conferma questo paradigma di tutela, come si evince dall'Accordo TRIPS²³ in aderenza a quanto già disposto dalla Convenzione di Berna.

Nello specifico, l'art. 10 sancisce, al primo paragrafo, che «i programmi per elaboratore, in codice sorgente o in codice oggetto, sono protetti come opere letterarie ai sensi della Convenzione di Berna (1971)»²⁴.

Anche il Trattato OMPI sul diritto d'autore (WCT)²⁵ si inserisce in questa linea regolatoria, affermando, all'art. 4, che i programmi per elaboratore sono tutelati secondo il diritto d'autore²⁶.

È utile ricordare, ai fini del presente discorso, che con il DPCM n. 244 del 3 gennaio 1994 sono state stabilite le modalità di registrazione all'interno del Registro pubblico speciale dei programmi per elaboratore, tenuto dalla S.I.A.E., tramite cui l'autore del programma può attestarne la paternità. Si tenga conto, però, il deposito all'interno del Registro non vale a costituire il momento iniziale della tutela autoriale, in quanto esso coincide con la creazione stessa dell'opera informatica. Esso rileva, pertanto, ai soli fini probatori, nell'ipotesi in cui dovessero

²³ Accordo TRIPs (*Trade Related Aspects of Intellectual Property Rights*), adottato a Marrakech 15 aprile 1994 – “Accordo relativo agli aspetti dei diritti di proprietà intellettuale attinenti al commercio” ratificato dall'Italia con legge 29 dicembre 1994, n. 747.

²⁴ Al secondo paragrafo, si specifica che «le compilazioni di dati o altro materiale, in forma leggibile da una macchina o in altra forma, che a causa della selezione o della disposizione del loro contenuto costituiscono creazioni intellettuali sono protette come tali. La protezione, che non copre i dati o il materiale stesso, non pregiudica diritti d'autore eventualmente esistenti sui dati o sul materiale».

²⁵ Anche, più comunemente, *World Copyright Act*, adottato a Ginevra il 20 dicembre 1996.

²⁶ «I programmi per elaboratore sono protetti in quanto opere letterarie ai sensi dell'articolo 2 della Convenzione di Berna. Tale protezione si applica a qualsiasi modo o forma di espressione di un programma per elaboratore».

essere rivendicati diritti di paternità sull'opera²⁷ o se dovessero essere presentate richieste di risarcimento per plagio.

Focalizzandosi sulla tutela autoriale, i programmi per elaboratore sono tutelati in qualsiasi forma siano espressi a condizione che siano una originale creazione intellettuale dell'autore. Come si evince dall'art. 1, co. 1, n.8 della l.d.a., la tutela predisposta per i *software* comprende anche il materiale preparatorio utile alla progettazione del programma informatico²⁸.

Il requisito dell'originalità per i *software*, al pari di ogni altra opera dell'ingegno, richiede che l'opera sia frutto dell'espressione creativa ed originale dell'autore, anche qualora essa sia frutto della rielaborazione di idee e concetti diffusamente condivisi nel settore di riferimento. Questo ultimo aspetto è particolarmente rilevante nell'ambito *software* in quanto la condivisione di principi informatici è fondamentale per lo sviluppo di nuovi programmi e per il progresso del mercato²⁹. Laddove, invece, si dovesse applicare la disciplina brevettuale, anche le idee e i principi alla base della invenzione sarebbero protetti e, quindi, non accessibili agli altri operatori del mercato³⁰.

Si deve mettere in evidenza, inoltre, che al fianco dei diritti di utilizzazione economica l'autore acquisisce anche diritti morali, tra cui,

²⁷ «La registrazione, dunque, è una forma di pubblicità notizia che non produce un effetto suo proprio e autonomo ma si somma e si sovrappone a quello già prodotto dall'atto della creazione», così L. MARTINAT e L. BOSSOTTI, *op. cit.*, spec. p. 144.

²⁸ In tal senso, «[...] per dissipare ogni dubbio, occorre precisare che solo l'espressione di un programma per elaboratore è oggetto di tutela e che le idee e i principi alla base di qualsiasi elemento di un programma, compresi quelli alla base delle sue interfacce, non sono tutelati dal diritto d'autore a norma della presente direttiva. Conformemente a detto principio del diritto d'autore, le idee e i principi che sono alla base della logica, degli algoritmi e dei linguaggi di programmazione non sono tutelati a norma della presente direttiva. [...]» (cfr. *Considerando 11, Direttiva 2009/24/CE*).

²⁹ Sulla tutelabilità della funzionalità di un programma per elaboratore, del linguaggio di programmazione e del formato dei file di dati utilizzati, v., in giurisprudenza, CGUE, Grande sez., C-406/10, cit.. In dottrina, v. A. GERACI, *Copia di un software e violazione del diritto d'autore: la Corte di giustizia sul caso SAS c. WP*, *Dir. ind.*, 2012, V, 457 ss.; G. NOTO LA DIEGA, *op. cit.*

³⁰ In merito al requisito dell'originalità del *software*, «[...] sono tutelati dal diritto d'autore, quale risultato di creazione intellettuale, i programmi per elaboratore elettronico, intesi come un complesso di informazioni o istruzioni idonee a far eseguire al sistema informatico determinate operazioni, che siano completamente nuovi o forniscano un apporto innovativo nel settore, esprimendo soluzioni migliori o diverse da quelle preesistenti», cfr. Cass. pen., III, 16 marzo 2018, n. 30047, in *Repertorio Foro Italiano*, 2018, *Diritti d'autore*, n.° 67; in *Ced. Cass. pen.*, rv. 273757 (m).

su tutti, il diritto al riconoscimento della paternità dell'opera da lui creata³¹.

Volgendo, infine, lo sguardo verso il sistema brevettuale, è possibile notare che la scelta a favore di una disciplina di diritto d'autore per i *software* sembra essere confermata dalle disposizioni della Convenzione sul brevetto europeo³² e, specificatamente, dai limiti imposti alle invenzioni brevettabili *ex art.* 52.

È importante chiarire, però, che la Direttiva 2009/24/CE³³ non ha escluso l'applicazione di altre disposizioni, anche in tema brevettuale, a patto che ciò non comporti la violazione dei diritti dell'autore del programma per elaboratore così come sanciti dal quadro normativo, nazionale ed europeo, in tema di diritto d'autore³⁴.

Questa netta presa di posizione del legislatore europeo lascia, per certi versi, aperto l'interrogativo sull'adeguatezza del sistema autoriale di tutela per programmi informatici così complessi ed in così notevole evoluzione. Dopotutto, l'agevole paradigma di disciplina imposto per i programmi per elaboratore, favorisce correttivi, modifiche e riutilizzazioni che se inclini ad un progresso tecnico-scientifico e rispondenti ad esigenze concorrenziali di mercato, risultano essere, parimenti, occasione di controversie circa la titolarità dei diritti di utilizzazione economica ed il loro legittimo esercizio.

Non a caso, unitamente alla disciplina sulle banche dati, la tutela del *software* è stata saggiamente definita come «la «più dirompente espressione dello “sconfinamento” del diritto d'autore contemporaneo»³⁵.

³¹ Questi sono diritti inalienabili, imprescrittibili e irrinunciabili che si acquistano in modo automatico con la creazione dell'opera dell'ingegno.

³² Conclusa a Monaco il 5 ottobre 1973 e riveduta a Monaco il 29 novembre 2000.

³³ Direttiva 2009/24/CE del Parlamento europeo e del Consiglio, del 23 aprile 2009 relativa alla tutela giuridica dei programmi per elaboratore.

³⁴ Al considerando 16 della Direttiva si legge, infatti, che «la tutela dei programmi per elaboratore a norma delle leggi sul diritto d'autore non deve pregiudicare l'applicazione, in casi opportuni, di altre forme di tutela; tuttavia qualsiasi disposizione contrattuale non conforme alle disposizioni della presente direttiva riguardanti la decompilazione o alle eccezioni di cui alla presente direttiva relative alla possibilità di fare una copia di riserva o all'osservazione, studio o sperimentazione del funzionamento di un programma dovrebbe essere considerata nulla». Come ribadito, inoltre, dall'art. 8 della Direttiva, «le disposizioni della presente direttiva non ostano all'applicazione di altre eventuali disposizioni giuridiche come quelle in materia di diritti brevettuali [...]».

³⁵ G. GHIDINI, *Profili evolutivi del diritto industriale. Innovazione - Concorrenza - Benessere dei consumatori - Accesso alle informazioni*, II ed., Giuffrè, Milano, 2008, spec. p. 193.

7.4. Il diritto di decompilazione

Come anticipato ed in modo speculare alle altre ipotesi di opere dell'ingegno, anche nel caso del *software*, l'autore diviene titolare di specifici diritti di utilizzazione economica³⁶.

In virtù dell'art. 64-*bis* della l.d.a., all'autore del *software* spettano i diritti di: a) riproduzione, permanente o temporanea, totale o parziale, con qualsiasi mezzo o in qualsiasi forma; b) traduzione, adattamento, trasformazione, modificazione nonché il diritto di riproduzione dell'opera, senza che ciò arrechi pregiudizio dei diritti di chi modifica il programma; c) distribuzione, in qualsiasi forma.

Al fianco, però, dell'autore del programma per elaboratore esistono diversi soggetti che possono legittimamente disporre di una copia del *software*, solitamente a seguito della conclusione di specifici contratti di cessione o di licenza d'uso³⁷.

³⁶ Tutelati, da un punto di vista processuale, dagli artt. 156 e ss. della l.d.a.

³⁷ Con il contratto di cessione di *software*, l'acquirente diviene titolare di tutti i diritti patrimoniali connessi al programma (e quindi, anche del codice sorgente), mentre nel caso di contratto di licenza d'uso, l'autore del *software* rimane titolare dei diritti di utilizzazione economica ma concede al licenziatario di poterne godere nei limiti delle condizioni contrattuali. Le licenze sono contratti atipici che possono assumere diverse connotazioni a seconda della tipologia del *software* e dei soggetti coinvolti. A titolo esemplificativo, esse possono comprendere tutti i diritti economici o soltanto una parte di esse, possono essere limitate territorialmente o possono essere concesse in esclusiva. Una delle licenze più diffuse nel mercato digitale, è la c.d. *End-User License Agreement* (EULA), con cui l'utente finale ha la possibilità di esercitare il diritto d'uso del *software* nei limiti espressi dalla licenza. Essa può essere perpetua, dietro il pagamento di una somma unica, o a durata, in cui è previsto il versamento di un canone periodico. Sul punto, v. L. MARTINAT e L. BOSSOTTI, *op. cit.*, p. 160 e ss. Altra ipotesi particolare è quella della licenza c.d. *click-through* che non consente all'utente di accettare i termini della licenza prima dell'accesso al *software*. A tal proposito, bisogna ricordare che per fronteggiare l'improprio utilizzo delle suddette licenze, spesso lesive dei diritti dell'utente, negli Stati Uniti sin dal 1998, con il *Digital Millenium Copyright Act* (DMCA), si è cercato di porre un argine legislativo e limitare il fenomeno. Per una primissima disamina sul punto, v. G. L. FOUNDS, *Shrinkwrap and Clickwrap Agreements: 2B or Not 2B?*, in *Federal Communications Law Journal*, vol. 52, 1, 1999, p. 99 e ss. Come ricordato anche in giurisprudenza, « accanto al modello di licenza tradizionale, che prevede il pagamento di un corrispettivo a fronte della concessione del diritto d'uso, si sono imposti, nel mondo dell'informatica, schemi negoziali alternativi, i quali consentono all'utilizzatore del programma di avere una disponibilità completa sul codice sorgente e d'impiegare il software anche senza corrispettivo», così Corte cost., 26 marzo 2010, in *Foro it.*, n. 122, 2010, 1, X, p. 2650 ss.

Le disposizioni contrattuali, diversamente formulate, redistribuiscono, dunque, i diritti di utilizzazione economica così come predisposti dalla l.d.a. In questo, si erge la funzione “riequilibratrice” della autonomia privata che rappresenta il principale strumento tramite cui modellare le esigenze concorrenziali del mercato e gli interessi degli operatori coinvolti al paradigma normativo costruito sulle trame della l.d.a.

Nell’ambito della cornice legislativa ora in analisi ed a norma del primo comma dell’art. 64-ter l.d.a., l’utente legittimo ha il diritto di riprodurre, tradurre, adattare e modificare il *software*, «allorché tali attività sono necessarie per l’uso del programma per elaboratore conformemente alla sua destinazione da parte del legittimo acquirente, inclusa la correzione degli errori».

Il secondo comma, invece, concede all’utente legittimo di «effettuare una copia di riserva del programma, qualora tale copia sia necessaria per l’uso». Non è ammessa la possibilità di pattuizioni contrarie, a pena di nullità, diversamente da quanto sancito nel primo comma, in cui, invece, non si escludono le eventuali differenti previsioni contrattuali.

La copia di riserva non legittima l’utente a realizzare più copie private del programma, ma semplicemente ad utilizzare una copia di riserva in caso di malfunzionamento o di errori procedurali. La copia privata, ovviamente, deve rimanere nella disponibilità dell’autore del programma per elaboratore e non può essere venduta a terzi.

Ai fini del presente studio, estremamente rilevante risulta essere quanto previsto dal comma terzo dell’art. 64-ter l.d.a.

Infatti, l’utente legittimo può, anche senza l’autorizzazione del titolare dei diritti, «osservare, studiare o sottoporre a prova il funzionamento del programma, allo scopo di determinare le idee ed i principi su cui è basato ogni elemento del programma stesso».

Questa attività di decompilazione, la c.d. *black box analysis*, è ammessa qualora venga compiuta nell’ambito di operazioni di caricamento, visualizzazione, esecuzione, trasmissione o memorizzazione³⁸.

³⁸ In merito, v. G. NOTO LA DIEGA, *op. cit.* spec. p. 543 e ss. L’Autore correttamente mette in evidenza che «occorre precisare che: a) non è vietato il reverse engineering, ma solo quella sua *species* nota come decompilazione; b) non si tratta di un divieto, ma di un’attività consentita a determinate condizioni; c) quelli in discorso sono limiti imposti dal fondamento stesso della c.d. proprietà intellettuale, cioè dalla necessità di compensare gli investimenti, in pari tempo incentivando l’innovazione».

Al pari del secondo comma, le clausole contrattuali pattuite in violazione di quanto disposto nel terzo comma sono nulle.

In via generale, quindi, l'attività di decompilazione non è sempre ammessa, se non nei limiti previsti dalla legge e limitatamente alla tipologia della c.d. *black-box analysis*.

La previsione di nullità è volta a rafforzare il progresso tecnologico in ambito *software*, conformemente all'idea di fondo della tutela autoriale: proteggere l'espressione creativa ed originale ma non le idee ed i principi ad essa correlati³⁹.

Si tenga, infine, conto che l'attività di decompilazione, oltre alle limitazioni legislative e contrattuali, incontra anche notevoli costi e complesse operazioni tecniche che, di fatto, attribuiscono un livello ulteriore ed implicito di tutela per il codice sorgente⁴⁰.

7.5. Le soluzioni proposte dalla Corte di Giustizia nel caso *Top System*

Ritornando sulle tracce della pronuncia in esame, è utile ripercorre le argomentazioni a sostegno delle soluzioni proposte, al fine di vagliarne l'adeguatezza nell'ottica di una più generale ricalibrazione della tutela del *software*.

In merito alla prima questione, ci si domanda se l'articolo 5, paragrafo 1, della direttiva 91/250 (*Deroghe relative alle attività riservate*)⁴¹

³⁹ Sul punto, v. L. MARTINAT e L. BOSSOTTI, *op. cit.*, p. 154-155.

⁴⁰ In questi termini, v. R. PARDOLESI e M. GRANIERI, *op. cit.*, spec. p. 303 (cfr. «[...] il codice diviene inespugnabile, sia fisicamente, sia dal punto di vista economico, nella misura in cui il reverse engineering sui circuiti e la decompilazione divengono attività dispendiose e con non adeguate percentuali di successo»).

⁴¹ «Salvo disposizioni contrattuali specifiche, non sono soggetti all'autorizzazione del titolare del diritto gli atti indicati nell'articolo 4, lettere a) e b), allorché tali atti sono necessari per un uso del programma per elaboratore conforme alla sua destinazione, da parte del legittimo acquirente, nonché per la correzione di errori».

Le attività dell'articolo 4 che vengono richiamate sono «a) la riproduzione, permanente o temporanea, totale o parziale di un programma per elaboratore con qualsivoglia mezzo, in qualsivoglia forma. Nella misura in cui operazioni come il caricamento, la visualizzazione, l'esecuzione, la trasmissione o la memorizzazione del programma per elaboratore richiedono una riproduzione, tali operazioni devono essere sottoposte ad autorizzazione da parte del titolare del diritto ; b) la traduzione, l'adattamento, l'adeguamento e ogni altra modifica di un programma per elaboratore e la riproduzione

debba essere interpretato nel senso che il legittimo acquirente di un programma per elaboratore ha il diritto di procedere alla decompilazione di tutto o parte di esso al fine di correggere errori che incidono sul funzionamento di tale programma, anche quando la correzione consista nel disattivare una funzione che pregiudica il buon funzionamento dell'applicazione di cui fa parte detto programma.

A tal riguardo, la Corte ricorda che la decompilazione di un *software* è una attività di *reverse engineering* che consiste nella ricostruzione del codice sorgente partendo da un codice oggetto di un programma esistente. Attraverso la decompilazione, di regola, si ottiene un "quasi codice sorgente", non perfettamente corrispondente al codice sorgente originale. L'attività di *reverse engineering* rappresenta il contraltare, in ambito *software*, della c.d. compilazione, attività che attiene, invece, al processo di creazione del codice oggetto sulla base delle istruzioni contenute nel codice sorgente⁴².

Siccome la decompilazione non è testualmente ricompresa tra gli atti disciplinati dall'art. 4 lett. a) e b) della direttiva, ai quali fa riferimento l'art. 5 par. 1, la Corte, nelle sue ricostruzioni, pone l'interrogativo sulla possibilità di estendere tale disciplina anche agli atti di decompilazione.

La decompilazione costituisce un'operazione di trasformazione della forma del codice di un programma che implica una riproduzione, almeno parziale e provvisoria, di tale codice, nonché una traduzione della forma di quest'ultimo.

A tal riguardo, la Corte sostiene che il legittimo acquirente di un programma non solo ha il diritto di decompilazione ai fini di interoperabilità a norma dell'art. 6 della direttiva, ma ha anche il diritto di decompilazione nel caso in cui ciò sia necessario per risolvere errori che incidono sul buon funzionamento del *software*, come previsto dall'art. 5, par. 1⁴³.

del programma che ne risulti, fatti salvi i diritti della persona che modifica il programma».

⁴² «[...] la «decompilazione» è diretta a ricostituire il codice sorgente di un programma a partire dal suo codice oggetto. La decompilazione è effettuata mediante un programma denominato «decompilatore», cfr. CGUE, C-13/20, cit., § 37.

⁴³ La Corte ribadisce che sia il codice sorgente sia il codice oggetto trovano tutela autorale, come peraltro già dichiarato in un'altra sentenza (Terza Sezione) del 22 dicembre 2010, *Bezpečnostní softwarová asociace - Svaz softwarové ochrany* contro *Ministerstvo kultury* (C-393/09).

Queste due ultime disposizioni hanno, però, finalità differenti, in quanto, mentre l'art. 6⁴⁴ riguarda gli atti necessari a garantire l'interoperabilità di programmi creati autonomamente, l'art. 5, par. 1 mira a consentire al legittimo acquirente di un programma di farne un uso conforme alla sua destinazione.

Dunque, l'articolo 5, paragrafo 1 della direttiva 91/250 deve essere interpretato nel senso che il legittimo acquirente di un programma per elaboratore ha il diritto di procedere alla decompilazione di tutto o parte di esso al fine di correggere errori che incidono sul funzionamento di tale programma, anche quando la correzione consiste nel disattivare una funzione che pregiudica il buon funzionamento dell'applicazione di cui fa parte detto programma.

In merito, invece, alla seconda questione (ovvero, se il legittimo acquirente di un programma per elaboratore che intenda procedere alla decompilazione al fine di correggere gli errori che incidono sul suo funzionamento debba soddisfare i requisiti previsti all'articolo 6 della direttiva), la Corte chiarisce i confini ed i presupposti dell'attività di decompilazione *ex art. 5*.

Nella ricostruzione della Corte, il compimento degli atti di decompilazione di un programma è soggetto a determinati requisiti, a norma dell'art. 5, par. 1.

La decompilazione, infatti, deve essere necessaria per la correzione di errori e per consentire al legittimo acquirente un uso conforme alla destinazione del programma. Inoltre, la rettifica degli errori secondo il modello dell'art. 5 deve rispettare le specifiche previsioni contrattuali che non possono in ogni modo vietare simili atti di correzione. Infine il legittimo acquirente non può utilizzare il risultato della decompilazione per fini diversi dalla correzione di errori di funzionamento.

Pertanto, come si legge nella sentenza, il legittimo acquirente che intenda procedere alla decompilazione al fine di correggere errori di funzionamento del *software* potrà agire soltanto nella misura necessaria a tale correzione e nel rispetto, se del caso, delle condizioni contrattualmente previste con il titolare del diritto d'autore su detto

⁴⁴ «Per gli atti di riproduzione del codice e di traduzione della sua forma ai sensi dell'articolo 4, lettere a) e b), non è necessaria l'autorizzazione del titolare dei diritti qualora l'esecuzione di tali atti al fine di modificare la forma del codice sia indispensabile per ottenere le informazioni necessarie per conseguire l'interoperabilità con altri programmi di un programma per elaboratore creato autonomamente», nel rispetto di specifiche condizioni dettate nell'art. 6 della direttiva.

programma, non dovendo, però rispettare i requisiti dettati dall'art. 6 della direttiva.

Nella prima e, soprattutto, nella seconda soluzione interpretativa emerge quale elemento di valutazione il giudizio di "conformità dell'uso" alla destinazione economica del programma⁴⁵.

Infatti, l'attività di decompilazione deve essere funzionale ad un uso conforme e legittimo del codice in analisi, nel rispetto di quanto previsto contrattualmente.

Il confine della legittimità di una specifica attività sembrerebbe, quindi, essere segnato dalla conformità di quell'atto al regolamento contrattuale ed alla destinazione economica ivi esplicitata.

In questo, si evince la centralità dell'autonomia privata che, di fatto, può estendere o restringere le maglie di accesso alle idee ed ai funzionamenti di base dei programmi per elaboratore, ricalibrando notevolmente l'assetto della tutela autoriale.

Parallelamente a ciò, dalla lettura della presente pronuncia, sembra emergere una chiara tendenza della CGUE favorevole alla condivisione delle idee e incline a realizzare un mercato europeo dei *software* aperto, accessibile e altamente concorrenziale⁴⁶.

7.6. Il *software* come "bene immateriale": quale tutela più adeguata?

L'annosa – e forse mai risolta – questione dell'individuazione della più adeguata tutela per i *software* ritorna con vigore alla luce delle grandi trasformazioni del mercato digitale.

⁴⁵ Riecheggia, indirettamente, la formulazione, in tema di usufrutto, dell'art. 981 cod. civ., laddove si prescrive «l'obbligo di non mutare la destinazione economica» in capo all'usufruttuario. Continuando in questo tentativo di assimilazione si dovrebbe immaginare, allora, che il mancato rispetto della destinazione economica del *software*, farebbe venir meno il diritto dell'utente legittimo di compiere attività di decompilazione, nel rispetto della l.d.a.

⁴⁶ Sulla tendenza a proteggere un nucleo centrale di idee e principi non soggetti ad atti appropriativi o a diritti proprietari, v. su tutti, C. HESS e E. OSTROM (a cura di), *Understanding Knowledge as a Commons. From Theory to Practice*, MIT press, Cambridge, 2007; ma anche, G. RESTA, G. RESTA, *The Case against the Privatization of Knowledge: Some Thoughts on the Myriad Genetics Controversy, Biotech Innovations and Fundamental Right*, in *Biotech Innovations and Fundamental Rights*, a cura di R. Bin, S. Lorenzon e N. Lucchi, Springer, 2012, p. 11-36

Aspetto centrale della questione è l'evidente disallineamento tra la natura dell'oggetto di tutela (il *software*, quale invenzione informatica priva di materialità) e la disciplina autoriale di riferimento⁴⁷.

Questa distanza tra l'immaterialità del bene e la circoscrizione della tutela all'espressione creativa, sembrerebbe essere rimarcata laddove si dovesse ritenere esistente l'esigenza che il programma protetto fornisca soluzioni concrete e migliorative dello stato della tecnica⁴⁸.

Nell'attuale conformazione della tutela del *software* un conto è l'idea, altro è la sua espressione. Solo la seconda trova tutela nella l.d.a., mentre la prima rimane liberamente accessibile, rappresentando di fatto lo strumento tramite cui migliorare ed adattare il processo informatico.

Il binomio idea – espressione rispecchia la tradizionale distinzione *corpus mysticum* e *corpus mechanicum* che nel mondo virtuale e dei beni immateriali tende a scomparire⁴⁹. Difatti, il *software*, seppur operante su sistemi materiali come gli *hardware*, rimane una creazione tecnico-informatica priva di un supporto fisico⁵⁰ che esteriorizzi l'espressione di quella idea.

Non bisogna, però, cadere nell'errore di una graduale assimilazione dei due concetti⁵¹ poiché, in questo caso, il rischio sarebbe quello di vedere eccessivamente ampliato l'ambito applicativo della disciplina⁵², tutelando aspetti ed elementi che andrebbero, piuttosto,

⁴⁷ Cfr. G. NOTO LA DIEGA, *op. cit.* p. 560 e ss.

⁴⁸ «[...] l'evocata riemersione della natura è tangibile là dove si richiama la necessità che i programmi costituiscano soluzioni in grado di migliorare lo stato della tecnica, principio proprio della tradizione brevettuale, ma il cui richiamo in sede autoriale è, se non altro, inusitato», *ivi*, p. 3 e ss.

⁴⁹ Si rimanda alla tradizionale distinzione tra una creazione intellettuale in quanto idea (il c.d. *corpus mysticum*) e una opera dell'ingegno che si estrinseca in un supporto materiale (il c.d. *corpus mechanicum*). La digitalizzazione ha, difatti, spezzato il legame tra la dimensione immateriale e quella materiale dell'opera dell'ingegno: da un lato, infatti, può essere facilmente trasferita su qualsiasi supporto fisico, dall'altro, non necessita più, ai fini della sua circolazione del *corpus mechanicum*, potendo essere distribuita attraverso gli strumenti del Web che ne aumentano la capacità di fruizione simultanea.

⁵⁰ A meno che esso non venga caricato su un CD-rom o su altro dispositivo mobile.

⁵¹ A favore di questa posizione, v. P. A. FRASSI, *Creazioni utili e diritto d'autore. Programmi per elaboratore e raccolte di dati*, Giuffrè, Milano, 1997.

⁵² Sui delicati limiti del concetto di originalità del software in relazione al binomio idea e espressione, v. V. MOSCON, *Diritto d'autore e protezione del software: l'irrisolta questione dell'originalità*, in *Dir. internet*, IV, 2007, p. 350 ss.

inseriti in quel «*retrotterra non proprietario*»⁵³, ovvero quel vasto patrimonio di idee, conoscenze e informazioni liberamente accessibili e sottratte a qualsivoglia forma di personale appropriazione, in quanto condizione essenziale per la creatività futura dell'intera collettività.

Si deve, inoltre, considerare che tassello centrale in questo complesso puzzle è costituito dalla tradizionale predisposizione del mondo digitale ed informatico a favore di processi accessibili e di dati liberamente condivisi⁵⁴. Non a caso lo spazio lasciato all'autonomia privata dalla I.d.a. è abbastanza ampio, con il limite invalicabile del divieto di monopolizzazione delle idee e dei principi⁵⁵.

Ed allora, tra la via autoriale, che pone tuttora gli interrogativi anzidetti⁵⁶, e la scelta altamente protezionistica e restrittiva del brevetto, potrebbe stagliarsi una terza soluzione intermedia⁵⁷. Si potrebbe, difatti, valutare l'opportunità di una ulteriore e più approfondita ricalibrazione della disciplina autoriale del *software*, implementando regole che aderiscano maggiormente alla natura immateriale del bene ed alle tendenze del mercato dei programmi per elaboratore.

⁵³ Stefano Rodotà elabora questa espressione particolarmente colorita e significativa, nel contesto di una più generale rivalutazione del rapporto beni pubblici-beni privati, in *Il terribile diritto. Studi sulla proprietà privata*, II ed., Il Mulino, Bologna, 1990.

⁵⁴ Si pensi al ruolo dei c.d. *open-source software* che rappresentano una soluzione comune in materia e sarebbero del tutto estranei ad una disciplina protezionistica come quella brevettuale. Sul punto, v. F. MARABINI, *La tutela giuridica del "software" e l'"open source"*, in *Cyberspazio e diritto*, 2017, 2, p. 405-422; C. PIANA e S. ALIPRANDI, *Il Free and Open Source software nell'ordinamento italiano: principali problematiche giuridiche*, in *Informatica e diritto*, 2012, 1, p. 79-96; A. ROSSATO, *Diritto e architettura nello spazio digitale. Il ruolo del software libero*, Padova, CEDAM, 2006; E. LOFFREDO, *Open source e appartenenza del software*, in *AIDA*, 2004, 67.

⁵⁵ «[...] l'autonomia privata incontra limiti stringenti imposti dalla esigenza di evitare che possano essere monopolizzate le idee», così G. NOTO LA DIEGA, *op. cit.*, spec. p. 559 e ss.

⁵⁶ «D'altro canto, il diritto d'autore come strumento di protezione del software continua a mostrare il fianco alle debolezze tipiche di quella forma di protezione e alla pericolosità insita nel riconoscimento automatico della protezione», così R. PARDOLESI e M. GRANIERI, *op. cit.*, spec. p. 303.

⁵⁷ Come già si metteva in evidenza in dottrina, circa la soluzione regolatoria più opportuna, bisogna «o denervare, a tutto campo e sino in fondo, quel che resta della già scossa matrice del diritto d'autore. Oppure ricavare al suo interno regole specificamente conformate alla natura adespota del software: opzione, quest'ultima, resa nella sostanza indisponibile dall'isteresi di scelte internazionali difficilmente riplasmabili nel breve periodo». così R. PARDOLESI, «*Software*», *property rights*» e *diritto d'autore: il ritorno dal paese delle meraviglie*, in *Foro it.*, 1987, 3, II, p. 289 e ss, spec. p. 300.

In campo software, si enfatizza la tensione tra le spinte verso l'innovazione e il modello privatistico di tutela dei diritti di proprietà intellettuale⁵⁸.

In questa tensione, sembra poter essere leso «quell'equilibrio originario tra le prospettive di breve periodo (riguardanti appunto la protezione accordata agli inventori ed ai creatori) e quelle di lungo periodo (in termini di incentivo al progresso e alla promozione della cultura a beneficio dell'umanità) che, in definitiva, è alla base dell'*intellectual property*»⁵⁹.

Una disciplina speciale per un bene (immateriale) speciale sembra poter essere una – valida – soluzione di compromesso tra le contrapposte istanze.

7.7. L'esigenza di conformità tra progresso scientifico ed esigenze contrattuali: riflessioni conclusive

La pronuncia qui in esame ha messo in evidenza peculiari aspetti della tutela giuridica del software.

È evidente che, data la specificità del bene tutelato, la disciplina autoriale subisca – anche involontariamente – un riadattamento che fa leva sulla funzione di equilibrio dell'autonomia privata.

Infatti, nel caso di specie, l'esigenza di conformità dell'uso del programma alla sua destinazione economica riflette l'incidenza delle clausole contrattuali sulle necessità tecniche di studio e di analisi.

Per questa ragione, interviene la l.d.a. sanzionando con la nullità i patti che impediscano legittime attività di decompilazione dell'utente legittimo, al fine di studiare il funzionamento del *software* e di correggere eventuali errori.

Dunque, da un lato esigenze contrattuali, dall'altro istanze di tipo tecnico-informatico e di progresso scientifico.

⁵⁸ Sul tema dell'accesso a dati ed informazioni comuni in ambito digitale, v. *ex multis*, G. GHIDINI, *Evoluzioni del diritto d'autore e promozione di informazione e cultura. Nuove luci e nuove ombre*, in *Scenari e prospettive del del diritto d'autore*, a cura di A.M. Gambino e V. Falce, Art ,Roma, 2009.

⁵⁹ A. IANNARELLI, "Proprietà", "immateriale", "atipicità": i nuovi scenari di tutela, in G. RESTA (a cura di), *Diritti esclusivi e nuovi beni immateriali*, Utet, 2011, spec. p. 166.

Il diritto, per sua stessa attitudine, ricerca costantemente nuove vie da percorrere, per trovare le soluzioni più adeguate alla realtà da regolare.

Il diritto d'autore, in particolare, ha da sempre risentito dello sviluppo delle nuove tecnologie. Il fenomeno nel suo complesso ha originato nuovi interrogativi, dovuti alla difficoltà di recepire il cambiamento e di porre un limite al dilagare delle ipotesi di elusione o violazione del sistema di diritto d'autore nel contesto virtuale.

Si è, dunque, imposta una politica legislativa europea orientata ad un sistema uniforme, attraverso la creazione di un mercato unico digitale che possa permettere un più alto tasso di efficienza ed adeguatezza delle regole ed un rilevante grado concorrenziale.

Allo stesso tempo, però, il diritto d'autore avrebbe il dovere rispondere alla sua seconda indole intrinseca, ovvero la sua tendenza a promuovere lo sviluppo culturale e scientifico, attraverso la creazione di un sistema il più possibilmente inclusivo che sia capace di cogliere appieno le nuove forme di business e le innovative attività del mondo digitale.

Il modello strettamente patrimonialista e volontaristico di concedere tutela all'opere del proprio ingegno, soprattutto in relazione alla sua utilizzazione economica, sembra non rispondere più perfettamente alle mutate caratteristiche della realtà digitale.⁶⁰

Alla luce di queste considerazioni, appare ancor più complesso considerare come appetibile la scelta brevettuale di tutela per il *software*. Essa, infatti, rappresenterebbe un "aggravamento" delle restrizioni ad ora previste, impedendo al mercato dei programmi per elaboratore di evolversi liberamente, costituendo monopoli sulle idee e i principi informatici.

Di certo, ciò non significa che la soluzione di diritto d'autore non possa essere corretta ed integrata. L'esistenza di una soluzione più restrittiva ed inadatta non legittima il legislatore ad una statica e passiva ricerca di alternative migliorative.

⁶⁰ Si discute ampiamente su forme di remunerazione alternative, su base strettamente morale e fondata sui rapporti interpersonali tra i soggetti del mondo digitale. (cfr. A. MUSSO, *Grounds of Protection: How Far Does The Incentive Paradigm Carry?*, in A.OHLY (a cura di), *Common Principles of European Intellectual Property Law*, Tubingen : Mohr Siebeck, 2012, spec. p. 33 e ss).

