



UNIVERSITÀ DEGLI STUDI DI TRENTO

Facoltà di Giurisprudenza
Corso di Laurea Magistrale in Giurisprudenza

**IL SOFTWARE COME OGGETTO DELLA
TUTELA BREVETTUALE.**
Una comparazione tra Europa e Stati Uniti

Relatore:
Prof. Roberto Caso

Laureando:
Lorenzo Radice

Anno Accademico 2016/2107



UNIVERSITÀ DEGLI STUDI DI TRENTO

Facoltà di Giurisprudenza
Corso di Laurea Magistrale in Giurisprudenza

**IL SOFTWARE COME OGGETTO DELLA
TUTELA BREVETTUALE.**
Una comparazione tra Europa e Stati Uniti

Relatore:
Prof. Roberto Caso

Laureando:
Lorenzo Radice

PAROLE CHIAVE:
Proprietà intellettuale - brevetto per invenzione - software - programma per elaboratore
- diritto comparato

Anno Accademico 2016/2107

Ai miei genitori.

INDICE

IL SOFTWARE COME OGGETTO DELLA TUTELA BREVETTUALE.

Una comparazione tra Europa e Stati Uniti

INTRODUZIONE.....	1
CAPITOLO I, Individuazione del bene giuridico: il <i>software</i>.....	3
<i>Premessa</i>	3
1. Breve storia del computer e dell'informatica	3
1.1. La nascita del computer	3
1.2. Verso la dicotomia hardware-software	5
1.3. La nascita dell'industria software	10
1.4. Il software nell'era digitale.....	12
2. La particolare natura del software. I beni informazionali.....	15
3. La necessità di una tutela del software tra progresso e incentivo alla creatività	18
4. Definizioni di software tra diritto e informatica	20
5. Software applicativi e software operativi. L'interoperabilità e il caso Microsoft.....	23
CAPITOLO II, Il software come oggetto della tutela	28
<i>Premessa</i>	28
1. L'oggetto della tutela autoriale: il <i>computer program</i> . La dicotomia idea-espressione nel software	29
2. L'oggetto della tutela brevettuale: Il <i>computer process</i> e le criticità del sistema statunitense	33
3. Il software come oggetto di segreto industriale	35
4. La <i>fictio iuris</i> come paradigma della tutela del software	38
5. I contratti di licenza: software proprietari e software liberi	39
6. Le misure tecnologiche di protezione, verso la chiusura del software? ...	43
7. La tutela del software e la sua percezione	45

CAPITOLO III, Il quadro legislativo e le *guidelines* degli Uffici Brevetti48

<i>Premessa</i>	48
1. Cenni alla legislazione europea prima della EPC	49
1.1. La Francia	49
1.2. L'Inghilterra.....	51
2. La Convenzione sul Brevetto Europeo	54
2.1. EPO, struttura e organizzazione	54
2.2. (<i>Segue</i>): il brevetto unitario europeo	57
2.3. EPC: normativa	60
2.4. Le Linee Guida: contributo tecnico, programma per elaboratore e invenzione attuata a mezzo di elaboratore elettronico.....	64
3. Le convenzioni internazionali sui brevetti	67
4. Legislazione dell'Unione Europea	69
4.1 Le direttive europee e la tutela giuridica dei programmi per elaboratore (cenni).....	69
4.2. Il fallimento della direttiva relativa alla brevettabilità delle invenzioni attuate per mezzo di elaboratori elettronici	70
5. La legislazione degli Stati Uniti	73
5.1. Il Patent Act del 1952	73
5.2. Dalla Commissione CONTU al Computer Software Copyright Act del 12 dicembre 1980	76
5.3. Le <i>Guidelines</i> del Patent Office statunitense	78

CAPITOLO IV, Il ruolo della giurisprudenza nell'individuazione del software brevettabile82

1. La giurisprudenza statunitense	82
1.1. <i>Premessa</i> : una analisi in chiave cronologica	82
1.2. Gottschalk v. Benson: il software come formula matematica	82
1.3. Diamond v. Diehr: il software come procedimento	85
1.4. In re Alappat: il software come macchina.....	86
1.5. State Street Bank & Trust Co. v. Signature Financial Group: l'utilità pratica del software	87
1.6. Bilski v. Kappos: il <i>machine or transformation test</i>	88
2. La giurisprudenza di EPO	92
2.1. <i>Premessa</i> : l'importanza delle linee guida.....	92
2.2. Il caso Vicom: la differenza tra un metodo matematico e un processo tecnico	93
2.3. (<i>Segue</i>): il carattere tecnico tra il <i>contribution approach</i> e il <i>problem-solution approach</i>	95
2.4. Le decisioni gemelle IBM: il software rivendicato di per sé	96
2.5. La rimodulazione del carattere tecnico: il caso Hitachi	98
2.6. "La via dell'originalità"? Il parere G 3/08 e i problemi aperti.....	99

RIFLESSIONI CONCLUSIVE, Prospettive di riforma della materia.....	101
<i>Premessa</i>	101
1. La privativa brevettuale come parte di una tutela stratificata: il livello di tutela del software.....	102
2. Il vero ostacolo al riconoscimento di una piena tutela brevettuale al software nell'Unione Europea: la concorrenza internazionale e l'accesso alla tutela.....	104
3. L'introduzione di regole <i>ad hoc</i> nel diritto brevettuale.....	105
4. L'opportunità di separare le tutele.....	108
 BIBLIOGRAFIA	110

INTRODUZIONE

Questa tesi nasce con lo scopo di illustrare in quali circostanze, entro quali limiti, e con quali criteri viene concessa al software la tutela di tipo brevettuale, evidenziando le principali problematiche, sia teoriche che pratiche, che riguardano la materia in oggetto.

Le dinamiche su cui insiste il ragionamento, non sono sintetizzabili in orientamenti giurisprudenziali piuttosto che in precise scelte legislative. Sono coinvolti fattori diversi tra loro, passibili di piena comprensione solo attraverso un'analisi interdisciplinare.

In particolare, è importante considerare il contesto storico di riferimento per individuare da un lato "lo stato della tecnica" e, dall'altro, la rilevanza economica del mercato dell'informatica. Nell'effettuare una tale ricerca, non si può prescindere da un'analisi di tipo comparato, vista la grande influenza che gli Stati Uniti, considerati il paese guida nella materia, hanno avuto sull'Europa.

Negli anni ottanta, momento di grande espansione dell'industria di software statunitense, le corti iniziarono a mutare i propri orientamenti, prospettando teorie volte alla inclusione dei programmi per elaboratore all'interno del *patentable subject matter*. Venne così invertita quella tendenza, maturata per non "soffocare" un ramo economico in pieno sviluppo, a tutelare il software solo mediante il *copyright*. Nonostante questo approfondimento si concentri sullo strumento brevettuale, si è ritenuta necessaria un'analisi, sebbene circostanziata, del diritto d'autore, stante il suo sovrapporsi alla privativa brevettuale.

Secondo il diritto dell'Unione Europea, infatti, il programma per elaboratore è tutelato attraverso il diritto d'autore. Nello strutturare tale privativa, tuttavia, se ne sono ampliate le prerogative, al fine di concedere agli operatori del settore uno strumento competitivo. L'operazione ha l'indubbio vantaggio di proteggere le industrie europee, meno sviluppate nel mercato di software rispetto alle concorrenti statunitensi e asiatiche, ma il risultato è stato la trasformazione del copyright in una tutela simil-brevettuale.

Nel momento in cui il diritto d'autore veniva potenziato mediante lo strumento legislativo, l'European Patent Office (EPO) interpretava le norme della Convenzione sul Brevetto Europeo, nel senso di concedere brevetti che insistevano su invenzioni di software, traendo ispirazione dalle decisioni assunte oltreoceano.

Poste le basi per l'applicazione della tutela brevettuale, non si è contemporaneamente agito nel senso di limitare i confini delle altre privative attinenti il software, ma, anzi, il progresso tecnologico ha reso possibile lo svilupparsi di ulteriori forme di tutela come le licenze e le misure tecnologiche di protezione. Il risultato è che in Europa il software è stato eccessivamente protetto, mediante una tutela stratificata: le diverse privative infatti non si cumulano tra loro ma si sovrappongono, insistendo su di un oggetto in parte identico.

I problemi negli Stati Uniti sono diversi e riguardano invece la "leggerezza" che molti autori denunciano in tema di concessione del brevetto. Questa

circostanza porta a rilevanti problemi nel mercato di software statunitense contraddistinto da una corsa, quasi predatoria, alla brevettabilità del trovato.

Lo sviluppo di un ragionamento il più completo possibile non può prescindere da una analisi preliminare volta alla esatta comprensione di cosa sia un software. Per tale motivo il primo capitolo ripercorre le tappe fondamentali della storia del computer e dell'informatica. Comprendendo l'evoluzione del software sarà possibile capire l'oggetto anche dal punto di vista squisitamente tecnico e se ne apprezzerà l'importanza che ricopre a livello economico, soprattutto nell'odierno contesto dell'era digitale.

Compreso il bene in questione, nel secondo capitolo si procede alla illustrazione delle diverse tipologie di tutele ad esso applicabili, attraverso una disamina volta non tanto alla mera descrizione di tali strumenti, quanto piuttosto alla comprensione di quale sia l'effettivo oggetto su cui essi insistono.

Nel terzo capitolo si analizzerà il quadro legislativo del diritto brevettuale europeo, internazionale e statunitense, vagliando con particolare attenzione in quali termini il software sia stato o meno preso in considerazione dai legislatori. Inoltre si evidenzierà come le normative di riferimento siano state interpretate dagli Uffici Brevetti, competenti a concedere o meno le rivendicazioni.

Il quarto capitolo consta di una analisi della giurisprudenza prima statunitense e poi europea. Il passaggio è fondamentale perché individua i criteri in base ai quali un brevetto su di un software possa o meno essere rilasciato.

Infine verranno prospettate delle riflessioni conclusive dove si affrontano i problemi prospettati in questa ricerca, cercando di trarne spunti di riflessione e proponendo qualche possibile soluzione.

CAPITOLO I

Individuazione del bene giuridico: il *software*

SOMMARIO: *Premessa* - 1. Breve storia del computer e dell'informatica - 1.1. La nascita del computer - 1.2. Verso la dicotomia hardware-software - 1.3. La nascita dell'industria software - 1.4. Il software nell'era digitale - 2. La particolare natura del software. I beni informazionali - 3. La necessità di una tutela del software tra progresso e incentivo alla creatività - 4. Definizioni di software tra diritto e informatica - 5. Software applicativi e software operativi. L'interoperabilità e il caso Microsoft

Premessa

Un approccio corretto alla questione della brevettabilità del software non può prescindere da una analisi preliminare volta alla esatta comprensione del bene oggetto della ricerca. Molte delle questioni dibattute dalla dottrina in merito alla tutela del software derivano proprio dalle particolarità del bene in questione.

Per comprendere cosa sia un programma per elaboratore si deve ripercorrere la storia del computer, vagliando quei passaggi che hanno reso possibile il distacco del software dall'hardware. L'innovazione tecnologica ha profondamente riformato la concezione del programma per elaboratore e per verificare tale circostanza si prenderanno in considerazione quelli che possono essere considerate come le fasi salienti della evoluzione dell'informatica, che hanno portato il software ad affrancarsi, prima tecnicamente, poi economicamente, ed infine giuridicamente, dalla macchina.

Acquisite le basi di una tale evoluzione da un punto di vista storico, si affronterà il bene oggetto della ricerca, il software, alla luce della scienza informatica ed economica. Si illustreranno le particolarità del software come bene dell'informazione, e si analizzerà cosa un software fa e cosa un software è, raffrontando tali concezioni con le definizioni fornite dai legislatori.

1. Breve storia del computer e dell'informatica

1.1. La nascita del computer

Curiose, affascinanti nonché drammatiche furono le circostanze nelle quali cominciò a delinearsi l'utilizzo della parola software. Nonostante la questione circa la paternità del termine, dal punto di vista etimologico non abbia soluzione

certa¹, i primi ad avere consapevolezza dell'esistenza di un qualcosa di ulteriore da implementare in una macchina, al fine di risolvere un certo problema, furono i tecnici britannici guidati da Alan Turing preposti alla decrittazione del linguaggio in codice tedesco Enigma, durante la seconda guerra mondiale².

Invero non vi sono fonti affidabili che legano la nascita della parola a questo preciso periodo storico, ma è innegabile che fu a partire dalla creazione di questi primi computer che cominciò a delinearsi la dicotomia tra componente dura della macchina, il cosiddetto *hardware*, e componente morbida, il *software*, ossia il cervello programmabile.

Alan Turing, che non a caso è considerato il padre della scienza informatica³, fu preposto negli anni della guerra alla decifrazione dei messaggi in codice tedeschi presso il Department of Communications inglese, a Bletchley Park. Fino a quel momento gli Inglesi decrittavano i codici tedeschi grazie all'utilizzo della macchina Bomba di Marian Rejewski, matematico polacco, ultimata nel 1938. I tedeschi a quel punto, consci della vulnerabilità delle loro comunicazioni, modificarono Enigma in modo tale da rendere la cifratura del messaggio ancora più complicata. Turing sviluppò allora il concetto di macchina di Turing, che implementò sulla Bomba, realizzandone una nuova versione, riuscendo così nell'intento di decifrare i codici tedeschi.

Pochi anni dopo, nel 1942 Max Newman sulla base della macchina di Turing progettò Colossus, il primo computer elettronico programmabile mai creato, realizzato da Tommy Flowers nel 1943. La macchina fu preposta alla decrittazione della cifratrice Lorenz SZ40/42, macchina molto più sofisticata di Enigma, ma più ingombrante, quindi utilizzata solo per comunicazioni ad alti livelli⁴.

Si può quindi evidenziare come la nascita del computer ebbe luogo da una necessità ben precisa: risolvere, attraverso una soluzione tecnica, una

¹ P. NIQUETTE nell'introduzione a *The software Age*, 1995, asserisce di avere per primo coniato la parola: "in October, 1953, I coined the word software. The notion of software as a separate thing from hardware took years to assert itself. Sure, the computer (popularly referred to as a "giant brain" in the early fifties) was unable to do anything but consume electrical power until a "programmer" came along to "program" it, and the consequent "routines" resided in the computer's "memory" thereafter. One did not, in the beginning, take a program written for one computer and put it into another. A half-century later, most people will find that hard to imagine. As originally conceived, the word 'software' was merely an obvious way to distinguish a program from the computer itself. A program comprised sequences of written instructions each endowed with the power to command the behavior of the permanently crafted machinery, the "hardware." For the origin of the word software most dictionaries give an unknown source and 1960 as the date, but I was there!". Ancora nel New York Times D. LEONHARDT, in un articolo intitolato "John Tukey, 85, statistician; coined the word 'software'", del 28 luglio 2000 cita come fonte un articolo (di cui però si omette il titolo) pubblicato nel 1958 nella rivista *American Mathematical Monthly*, dove John Wilder Tukey, matematico, avrebbe per primo fatto uso della parola in ambito accademico. Il riferimento pare corretto, tanto che lo stesso articolo è quotato nell'Oxford English Dictionary, in Rete: <http://www.oed.com/view/Entry/183938?redirectedFrom=software#eid>.

² Si veda per un approfondimento storico sul ruolo di Alan Turing nelle vicende in oggetto N. CAWTORNE, *L'enigma di un genio. La vera storia di Alan Turing il matematico inglese che decrittò il codice nazista*, Newton Compton, Roma, 2014.

³ In questo senso si veda B. J. COPELAND, *The Essential Turing. The ideas that gave birth to the computer age*, Oxford University Press, Oxford, 2004.

⁴ P. S. SANNA, *Manuale di informatica di base*, CEDAM, Padova, 2014, pp. 7-10.

operazione complessa nel minor tempo possibile. Un sistema di calcolo fornisce un contributo tecnico non ovvio, implementante un processo industriale, che produce un risultato utile, concreto e tangibile⁵. Si nota immediatamente come tale ultima definizione riassuma in sé molte caratteristiche e condizioni richieste dai legislatori o dalle corti al fine della concessione di una tutela all'invenzione.

Sempre in Inghilterra, a Cambridge, nel 1949 Maurice Wilkes ed il suo team realizzarono quello che si può definire come uno dei primi antenati del moderno computer, l'EDSAC (Electronic Delay Storage Automatic Calculator). In questa fase di sviluppo dell'informatica, nacquero diversi sistemi di calcolo analoghi, come per esempio l'UNIVAC, l'ENIAC e l'EDVAC⁶. Il tratto comune di questi computer risiede nella programmazione rudimentale della macchina e nelle funzioni svolte, che consistevano principalmente in calcoli matematici e aritmetici⁷.

In questa prima fase di sviluppo storico, nelle macchine per elaborare dati, la distanza tra hardware e software è nella prassi assente. I due elementi sono fusi in un sistema pensato ed ideato al fine di compiere quella determinata operazione. Gli antenati del computer, inteso come sistema automatico di elaborazione di dati, nascono quindi come sintesi di hardware e software combinati insieme al fine di risolvere determinati problemi.

Questi problemi, sebbene risolvibili anche attraverso un procedimento umano, hanno una complessità tale da rendere più efficiente lo sviluppo di un sistema di un calcolo automatizzato, rispetto all'impiego diretto del tempo umano per la risoluzione del problema stesso. La macchina così realizzata è un mezzo che si sostituisce all'uomo per la velocità, l'efficienza, ed il ristretto, se non nullo margine di errore nel compimento di determinate operazioni.

1.2. Verso la dicotomia hardware-software

Una concezione volta alla dicotomia tra hardware e software comincia a palesarsi nel momento in cui la macchina inizia ad acquisire caratteristiche standard. Tale procedimento fu possibile grazie ad una continua e repentina innovazione tecnologica. Il miglioramento di un sistema, considerato nel suo complesso, può avvenire o attraverso l'implementazione di nuove tecnologie hardware, oppure attraverso la creazione di nuove funzioni che si innestano sulla macchina stessa⁸.

⁵ P. E. CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, Apogeo Editore, Milano, 2006, pp. 10-14.

⁶ Per una analisi approfondita di questi dispositivi v. *ibidem*, pp. 23-47.

⁷ *Ibidem*, pp. 32-36. È da evidenziare come la realizzazione del computer fu possibile grazie al documento di J. VON NEUMANN, *First draft of a report on the EDVAC*. Tale documento è fondamentale perché fissa i principi dell'architettura hardware di base, sulla quale si fonderanno tutti i futuri computer digitali. Questo modello di architettura, in informatica è infatti detto "di von Neumann" in quanto il matematico la sviluppò per la creazione dell'EDVAC. Il documento è Rete: <http://www.virtualtravelog.net/wp/wp-content/media/2003-08-TheFirstDraft.pdf>. Per un ulteriore approfondimento tecnico riguardante l'architettura v. SANNA, *Manuale di informatica di base*, pp. 153-188.

⁸ In questo senso v. P. LEITH, *Software and patents in Europe*, Cambridge intellectual property Rights and information law, Cambridge University Press, 2007, pp. 1-5.

In questa fase, definibile come “intermedia”, che si sviluppa grosso modo dagli anni sessanta sino agli anni ottanta, inizia ad affermarsi l'idea dell'esistenza, anche giuridica, del software considerato di per sé, ossia come entità autonoma e distinta dalla macchina, con la quale opera in sinergia⁹.

Se prima il computer era concepito come oggetto da stanza, principalmente usato da enti governativi e da università, comincia in questo periodo ad affermarsi l'idea del computer da tavolo, un oggetto di dimensioni più ridotte che si rivolge ad una clientela più ampia composta da società e da professionisti¹⁰.

Il passaggio cruciale è l'avvento dei microprocessori¹¹, attraverso il cui utilizzo fu possibile la realizzazione dei cosiddetti microcomputer. La differenza sostanziale di tali processori rispetto ai predecessori, risiede nelle dimensioni contenute e nei prezzi, che subirono una drastica diminuzione. Grazie a questi processori nacquero appunto i primi microcomputer, anche detti *home computer*.

Una vicenda particolarmente interessante sotto diversi aspetti si verificò nel 1975. In quell'anno vide la luce il fortunato Altair 8800, un microcomputer dalle eccellenti performance.

Bill Gates, all'epoca ancora studente, assieme a Paul Allen, comprese, visto il basso costo del computer, che sarebbe stato molto appetibile per una nuova fascia di mercato, quella degli hobbisti, ossia persone appassionate di computer¹². Ritennero così che sarebbe stato proficuo abbinare all'Altair un software interprete per il linguaggio di programmazione ad alto livello più

⁹ In questo senso anche G. GHIDINI, *Profili evolutivi del diritto industriale*, Giuffrè Editore, Milano, 2015, 277, ove si individua “la richiesta di una tutela esclusiva del software in coincidenza con il graduale sviluppo e la relativa commercializzazione dei programmi come prodotti autonomi dall'hardware” in un percorso che inizia verso la fine degli anni sessanta e culmina agli inizi degli anni novanta con la diffusione del personal computer; CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, pp. 129-132.

¹⁰ In questo senso v. G. ZICCARDI, *L'eredità culturale del secondo dopoguerra e le nuove tecnologie: gli anni Cinquanta e Sessanta, il computer e le prime problematiche sociali e giuridiche*, in *Cyberspazio e diritto*, Vol. 9, n. 2, 163-206, 2008; il punto è evidenziato anche dalla Commissione CONTU, *Final Report*, 1978: “The first commercial computers, built shortly after World War II, were based largely on vacuum tubes and were so expensive that only the government or the largest corporations could even consider owning them. To function, the typical early computer required an environment in which temperature and humidity were carefully monitored. It was controlled by programs created by its manufacturer and users exclusively for that particular computer. Subsequent generations of computers have been characterized by dramatic reductions in the size, energy requirements, and price for a given amount of computational power”. Cit., Chapter 3, Computers and Copyright, Background.

¹¹ Uno dei primi microprocessori fu l'Intel 8080, realizzato a metà del 1974. Grazie a questo processore la MITS realizzò l'Altair 8800, un computer molto economico che passò alla storia grazie alla sua diffusione. Ancora, sulla base del processore MOS 6502, realizzato nel 1975, vide la luce nel 1976 l'Apple I, cui seguirà il più completo e avanzato Apple II, nel 1978, importanti perché considerati i primi home-computer. Venivano infatti venduti compresi di monitor e di tastiera, in un case di dimensioni relativamente ridotte. V. CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, pp. 250-256.

¹² Sul ruolo degli hobbisti in questa fase della storia del computer v. CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, pp. 265-266.

sviluppato all'epoca: il BASIC¹³. Contattarono quindi Ed Roberts, fondatore di MITS, proponendogli la commercializzazione abbinata del loro interprete BASIC, e chiedendo di poter mostrare per un test il loro software. Roberts fissò un incontro per il mese di marzo dello stesso anno. Iniziarono così il progetto ed assieme Monte Davidoff, studente di Harvard amico di Gates, realizzarono il linguaggio in due mesi¹⁴. La MITS fu soddisfatta e nel mese di aprile fu concesso alla neonata Micro-soft¹⁵ un contratto di licenza esclusiva per la distribuzione dell'Altair 8800 abbinato al software interprete di Gates e Allen: l'Altair BASIC.

Il contratto, di durata decennale, prevedeva delle royalties in favore di Micro-Soft per ogni copia venduta, con la clausola di "usare i migliori sforzi per dare in licenza, promuovere e commercializzare il programma".

La nascita della casa produttrice di software che diverrà da lì a pochi anni leader nel settore, unitamente ad uno dei primi contratti di licenza stipulato tra una azienda produttrice di hardware ed una software house, sono elementi estremamente rilevanti. Si riconosce infatti, attraverso un accordo commerciale, che attraverso l'implementazione di un determinato software su di una macchina, è possibile creare un prodotto più appetibile per il mercato¹⁶.

Ma la vicenda in questione non è interessante solo per questi motivi. La MITS, per pubblicizzare il proprio prodotto, mise in atto un tour promozionale, organizzando convegni di città in città.

Durante un incontro, tuttavia, una copia dell'Altair BASIC venne trafugata. Il software era incorporato su un supporto tangibile, un nastro perforato¹⁷, che, una volta ricopiato attraverso l'utilizzo di una perforatrice per nastri, venne distribuito nei circoli di appassionati di computer¹⁸. Tale circostanza, unitamente al fatto che il software creato da Micro-soft e distribuito da MITS era molto caro, se non acquistato unitamente all'Altair 8080, portò ad un insuccesso del progetto. Invero il software era molto utilizzato, ma solamente nella sua versione "piratata".

¹³ Così vengono riportate tali vicende da P. FREIBERGER, M. SWAINE, *Fire in the Valley: The Making of the Personal Computer*, McGraw-Hill, New York, 2000, pp. 53-55. Per una panoramica del linguaggio BASIC v. CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, pp. 274-279; Per una illustrazione tecnica dei linguaggi di programmazione ad alto livello v. SANNA, *Manuale di informatica di base*, pp. 141-149.

¹⁴ Così riporta J. WALLACE, J. ERICKSON, *Hard Drive: Bill Gates and the Making of the Microsoft Empire*, John Wiley & Sons, New York, 1992, pp. 77-79.

¹⁵ La parola "Micro-soft", non a caso, è la sintesi tra i vocaboli "microcomputer" e "software".

¹⁶ Le vicende sono riportate dalla stessa Microsoft sulla sua pagina web. In Rete <https://news.microsoft.com/2000/05/09/microsoft-fast-facts-1975/#wOBhaQGvKd5H0bcP.97>.

¹⁷ Il nastro perforato è un sottile nastro fatto di carta o di altri materiali plastici, utilizzato per memorizzare informazioni codificate attraverso fori praticati ad intervalli regolari sul nastro stesso.

¹⁸ La circostanza viene descritta da F. MOORE, *It's a Hobby*, in *Homebrew Computer Club Newsletter*, vol. 1, n° 4, 7 giugno 1975. In Rete http://www.digibarn.com/collections/newsletters/homebrew/V1_04/index.html.

Gates decise così di scrivere una lettera aperta alla comunità di hobbisti¹⁹ nella quale spiega la sua posizione in merito al fenomeno della pirateria informatica²⁰. Dopo aver illustrato i costi sostenuti per la realizzazione del software, rileva come il suo guadagno sia stato irrilevante, e di certo non costituisce un incentivo per continuare ad innovare e ad introdurre novità nel mercato. Infine definisce ladro chiunque copi e utilizzi il programma da lui progettato senza averlo pagato²¹.

Il fatto descritto è molto importante perché anticipa nei tempi il problema cruciale della distribuzione del software: la facilità di essere copiato e trasmesso. Le opinioni in merito alla posizione di Gates furono contrastanti.

Alcuni appoggiarono tale posizione accostando le royalties provenienti dalla vendita di un software alla stregua di quelle che spettano ad un autore di libri o di musica²². Altri, invece, accusarono Gates di volersi semplicemente arricchire, vendendo il prodotto ad un costo spropositato, considerando anche sovrastimati i costi riportati per la realizzazione del software²³. Si distinguono già nettamente le posizioni a favore di un software-proprietario, ossia licenziato dall'autore, e quelle a favore di un software libero, aperto, ossia con un codice sorgente pubblico, visibile e modificabile da chiunque.

Una interessante posizione mediana è affermata da Jim Warren, membro dell'Homebrew Computer Club, il quale, sul numero di luglio del 1976 della newsletter "SIGPLAN Notices", afferma che la soluzione più efficace al problema della copia illegittima di software, sia la sua commercializzazione e distribuzione ad un prezzo basso, così che gli utenti siano maggiormente invogliati all'acquisto piuttosto che al furto²⁴.

¹⁹ Il destinatario principale di "An open letter to hobbyists" era l'Homebrew Computer Club. Questo circolo fu uno dei primi a riunire molti appassionati del mondo dell'informatica, e molte copie del software interprete di Micro-soft furono proprio distribuite all'interno del club, come dichiara lo stesso Fred Moore nel suo articolo, v. *ibidem*.

²⁰ La lettera fu pubblicata da diverse riviste, vedi tra tutte *Homebrew Computer Club Newsletter*, vol. 2, n° 1, Homebrew Computer Club, 01/1976, p. 2.

²¹ Estratto dalla lettera aperta agli hobbisti: "Why is this? As the majority of hobbyists must be aware, most of you steal your software. Hardware must be paid for, but software is something to share. Who cares if the people who worked on it get paid? Is this fair? One thing you don't do by stealing software is get back at MITS for some problem you may have had. MITS doesn't make money selling software. The royalty paid to us, the manual, the tape, and the overhead make it a break-even operation. One thing you do do is prevent good software from being written. Who can afford to do professional work for nothing? What hobbyist can put 3-man years into programming, finding all bugs, documenting his product and distribute for free? The fact is, no one besides us has invested a lot of money in hobby software. We have written 6800 BASIC, and are writing 8080 APL and 6800 APL, but there is very little incentive to make this software available to hobbyists. Most directly, the thing you do is theft". In Rete la lettera completa: http://www.digibarn.com/collections/newsletters/homebrew/V2_01/index.html.

²² D. BUNNEL, *Across the Editor's Desk*, in *Computer Notes*, vol. 1, n° 5, MITS, 09/1975.

²³ In effetti Gates ed Allen, per sviluppare l'interprete del BASIC utilizzarono, in modo illegittimo, in quanto il fine era commerciale ed extra-accademico, un PDP-10 dell'Harvard's Aiken Computer Center che era reso disponibile gratuitamente agli studenti in time-sharing. Così riportato in WALLACE, ERICKSON, *Hard Drive: Bill Gates and the Making of the Microsoft Empire*, pp. 81-83.

²⁴ J. C. WARREN JR., *Correspondence, SIGPLAN Notices*, July 1976, in K. DRISCOLL, *Professional work for nothing: Software commercialization and "An Open Letter to Hobbyists"*, University of Southern California, 2015, pp.10-12, in Rete: http://kevindriscoll.info/docs/driscoll.professional_work_for_nothing.2015.PREPRINT.pdf.

Ulteriore vicenda interessante, che si sviluppò attorno all'Altair BASIC, fu lo scioglimento del contratto tra Micro-soft e MITS. Nel 1977, quando la Pertec Computer, che aveva recentemente acquisito MITS, si rifiutò di concedere in licenza il BASIC ad altre società interessate al prodotto, Gates sollevò in tribunale la violazione del contratto di licenza, sulla base della particolare clausola che obbligava MITS a compiere tutti gli sforzi necessari a pubblicizzare e commercializzare l'Altair BASIC nel migliore dei modi.

Micro-soft iniziò quindi a commercializzarlo in proprio e tale circostanza fece la fortuna della società. Mentre precedentemente la società di Gates, a causa dello svantaggioso contratto di licenza, era di fatto una succursale della MITS, azienda che produceva hardware, comincia ora ad affermarsi come una vera e propria società produttrice di beni dotati di un certo valore economico: i software²⁵.

Il miglioramento che investì il computer in questa fase storica, non riguardò solamente la parte hardware²⁶. L'utilizzo di un computer era pensato per applicazioni tecniche, scientifiche e gestionali. Le funzioni implementate erano in continuo aumento, e si sostanziano in fogli di calcolo elettronico, in programmi di videoscrittura, in programmi per l'ufficio, e videro la luce i primi giochi.

Per il funzionamento di tali applicazioni si svilupparono interfacce basilari, ossia sistemi operativi, che comunicavano con l'utente attraverso il monitor e la tastiera. Il sistema operativo più utilizzato in questo periodo è il DOS (*Disk Operating System*). Ne furono create moltissime versioni, questo perché ogni casa produttrice aveva il proprio sistema DOS²⁷.

Il suo funzionamento avviene attraverso comandi diretti digitati dall'utente su di una linea chiamata *prompt dei comandi*, in grado di eseguire solo un programma per volta. Era necessario conoscere il nome e la dicitura tecnica esatta di molti programmi per poter sfruttare a pieno le funzioni del sistema. Questo modo di interagire, anche se di fatto aumentò la possibilità e la facilità di interazione tra uomo e macchina, non favorì di certo la diffusione del personal computer in ambito domestico.

²⁵ Per una ricostruzione dei diversi passaggi che portarono Microsoft ad un ruolo di predominanza e di dominio nel mercato di software v. S. MANES, P. ANDREWS, *Gates: How Microsoft's Mogul Reinvented an Industry and Made Himself the Richest Man in America*, Touchstone, Simon and Schuster, New York, 1994.

²⁶ La Commissione CONTU, *Final Report*, 1978, nota, quasi profeticamente, come: "*These people -riferito ai produttori di software- may be users or they may be, and increasingly are, programmers or small firms who market their wares for use by individual machine owners who are not in a position to write their own programs (...). If present computer industry trends continue, it is all but certain that programs written by nonmachine manufacturers will gain an increasing share of the market, not only because writing programs and building machines are two very different skills that need not necessarily occur simultaneously, but also because program writing requires little capital investment*". Cit., Cap. 3, *Computers and Copyright*, *Foundation for the Recommendations, Computer Programs*. Si evidenzia quindi, anche in questo documento, come il periodo in questione possa essere definito "di passaggio". L'approccio ad una utenza più ampia è evidente ed è fagocitato anche dall'innovazione apportata dai creatori di software.

²⁷ Per citarne alcuni: Apple DOS, Atari DOS, Commodore DOS9, PC DOS.

Tale circostanza, unitamente costo medio di questi sistemi, che era comunque alto, si risolse in una utenza principalmente di tipo commerciale e professionale²⁸.

1.3. La nascita dell'industria software

Con l'avvento dei microprocessori, le grandi aziende produttrici cominciano ad elaborare e a sviluppare l'idea di un computer portatile, personale o da casa²⁹.

IBM, allora azienda leader del mercato della tecnologia hardware mainframe³⁰, nel 1981, vista la crescente espansione dell'informatica distribuita, lanciò sul mercato il PC IBM che divenne lo standard dell'industria dei personal computer³¹. Nonostante il costo elevato il sistema era stabile, l'hardware di qualità e l'assistenza di livello.

Per la realizzazione di un sistema operativo IBM nel 1980 si rivolse a Microsoft, che realizzò MS-DOS³², il primo sistema operativo di larga diffusione per personal computer. Nonostante all'inizio non era obbligatorio acquistare una copia di MS-DOS insieme ai PC IBM, ma era possibile scegliere fra quattro sistemi operativi, a causa del prezzo meno elevato, il software sviluppato da Bill Gates e Paul Allen fu il più venduto. La fortuna di Microsoft si deve anche alla sinergia che instaurò con IBM³³.

Ma il passaggio definitivo che consacra la separazione tra macchina e programma si ha nel momento in cui tale tecnologia diviene fruibile dal grande pubblico. Tale passaggio si concretizza attraverso due fattori: il progressivo abbassamento dei prezzi e la crescente facilità di utilizzo dei sistemi. Il mercato che prima era sostanzialmente un *business to business* innesta i suoi rapporti con un gruppo considerevole di consumatori³⁴.

Ciò fu reso possibile grazie allo sviluppo di una interfaccia grafica avanzata in grado di mettere in diretta comunicazione l'utenza con il programma. Le interfacce grafiche, o GUI (*Graphics User Interface*) vengono create per consentire una interazione con la macchina senza dover conoscere

²⁸ CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, pp. 282-285.

²⁹ *Ibidem*, pp. 308-323.

³⁰ Sull'ascesa di IBM nell'industria hardware, v. *Ibidem*, pp. 86-95.

³¹ *Ibidem*, p. 314.

³² È da segnalare che la prima versione di MS-DOS sul PC IBM prese il nome di PC DOS. Infatti, il software sviluppato da Gates dopo un controllo di qualità di IBM, che ne rivelò diversi errori, fu riscritto in alcune sue parti. Inizialmente quindi il sistema operativo venne licenziato congiuntamente da IBM e Microsoft. Successivamente, con una nuova versione del software, Microsoft si affrancò e MS-DOS venne licenziato autonomamente da Microsoft. CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, p. 315.

³³ Per una lettura di carattere generale sulla progressiva espansione di Microsoft v. WALLACE, ERICKSON, *Hard Drive: Bill Gates and the Making of the Microsoft Empire*.

³⁴ Tale importante circostanza è evidenziata anche nel CONTU, *Final Report*, 1978: "Computers have become less cumbersome and expensive, so that individuals can and do own computers in their homes and offices with more power than the first commercial computers, while at the same time, programs have become less and less frequently written to comply with the requirements imposed by a single-purpose machine". Cit., Cap. 3, *Computers and Copyright*, Foundation for the Recommendations, Computer Programs.

la esatta sintassi dei comandi. Da tale idea nascono elementi come le finestre, le icone, i menù e i sotto-menù che possono essere utilizzati dall'utente attraverso un sistema di puntamento controllato dal *mouse*³⁵.

La rivoluzione si verificò a metà degli anni 80, prima con il Macintosh di Apple nel 1984, e poi con Windows di Microsoft nel 1985.

Invero, Apple sviluppò una interfaccia grafica anche prima dell'uscita di Macintosh³⁶. Il progetto iniziò nel 1978 sulla base dello Xerox Alto³⁷. La Xerox Corporation, che aveva progettato il computer in oggetto permise agli ingegneri Apple di analizzarlo in cambio del futuro acquisto di azioni Apple da parte della stessa Xerox³⁸. Così nel 1983 fu lanciato sul mercato Lisa, un computer fortemente innovativo. Il costo esorbitante, abbinato alle scarse performance del prodotto ne decretarono il fallimento.

La circostanza interessante è che Lisa conteneva un sistema operativo molto avanzato, in grado di effettuare più operazioni contemporaneamente³⁹. La componente hardware, tuttavia, non era in grado di sopportare una quantità tale di informazioni. Un anno dopo i problemi fondamentali di Lisa furono risolti nel Macintosh, anche grazie ad uno snellimento della componente software⁴⁰.

L'evoluzione di Windows fu invece più lenta. Le prime versioni di Windows, invero, non erano un sistema operativo vero e proprio. Il sistema operativo rimaneva MS-DOS, e Windows era una sorta di espansione mediante la quale l'utente poteva accedere ad una interfaccia grafica. Solo nel 1995, i servizi DOS vengono integrati in Windows 95, primo sistema operativo di Microsoft in cui compare l'icona "start" che diverrà in seguito uno standard caratterizzante⁴¹.

Questo approccio *user-friendly* allargò progressivamente il mercato di riferimento. La facilità e l'immediatezza di utilizzo del computer, unitamente alla presenza di software che implementavano funzioni sempre più interessanti, decretò il successo dei computer. Inoltre, l'avvento di processori sempre più

³⁵ SANNA, *Manuale di informatica di base*, pp. 21-24.

³⁶ W. ISAACSON, *Steve Jobs*, Simon and Schuster, New York, 2011, pp. 92-102.

³⁷ CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, pp. 301-308.

³⁸ Per una analisi approfondita del progetto Xerox v. M. A. HILTZIK, *Dealers of Lightning. Xerox PARC and the dawn of the computer age*, HarperCollins, New York, 1999.

³⁹ Il software di Lisa fu il primo sistema operativo commercializzato che poteva eseguire funzioni *multitasking*. Era possibile l'utilizzo contemporaneo di più programmi applicativi, attraverso finestre grafiche che si sovrapponevano. La componente software era tuttavia troppo avanzata per le allora soluzioni utilizzate nell'hardware.

⁴⁰ ISAACSON, *Steve Jobs*, pp. 108-117; sul Macintosh in particolare v. CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, p. 318.

⁴¹ Invero bisogna citare il fatto che Microsoft, nello sviluppo di Windows, quantomeno si ispirò alla interfaccia grafica di Macintosh. La prima versione di Windows, realizzata nell'85, fu progettata sulla base di una serie di licenze che Microsoft, riesci a farsi rilasciare grazie ad un accordo con Apple. Il sistema operativo di quest'ultima azienda era infatti superiore, anche se molto meno diffuso. Il grande successo di Windows 3.0, presentato nel 1990, fu accompagnato da azioni legali da parte di Apple a causa dell'utilizzo di finestre sovrapposte e di altre caratteristiche che Apple pretendeva appartenere all'aspetto caratteristico del suo sistema operativo. Il tribunale attestò 10 delle 189 violazioni evidenziate da Apple. La controversia si risolse tuttavia con un accordo traslativo, del quale non si conoscono i dettagli esatti.

evoluti ha reso possibile la realizzazione di computer sempre più performanti, di dimensioni sempre più ridotte e a costi sempre più appetibili⁴².

Il progresso tecnologico nell'ambito dei computer ha reso possibile il delinearsi di una dicotomia tra hardware e software. Allo stesso tempo, lo sviluppo del computer è avvenuto attraverso il miglioramento sia della componente dura, l'hardware, sia della componente morbida, il software.

La separazione tecnica del software dall'hardware ha reso possibile lo svilupparsi di una industria di software economicamente molto rilevante, affrancata, ma ancora complementare all'industria hardware.

Nella progettazione e sviluppo di un programma applicativo era fondamentale sapere non solo il sistema operativo sul quale sarebbe stato eseguito, ma anche le caratteristiche della macchina sulla quale sarebbe stato utilizzato. La programmazione di un software non poteva prescindere dal considerare l'effetto che il programma avrebbe avuto sulla macchina. Le capacità della macchina, in termini di prestazioni, individuavano il mercato di riferimento. Così, per esempio, un semplice programma di videoscrittura poteva essere pensato per una ampia utenza, mentre un software di progettazione di interni o gestionale era rivolto a professionisti, stanti le risorse hardware richieste per fare funzionare il programma in maniera performante.

Tali considerazioni, sono sicuramente valide anche oggi, ma solo in una determinata quota di mercato di software. L'interesse degli sviluppatori si è infatti rivolto verso un mercato nuovo, diverso, fatto di servizi. Un mercato che, come si vedrà, è in continua crescita ed espansione.

1.4. Il software nell'era digitale

Ai giorni d'oggi, in quella che potremmo definire era digitale⁴³, la complementarità tra hardware e software, anche se ineliminabile da un punto di vista teorico, è di fatto irrilevante nella prassi.

Il mercato dei software si è di fatto evoluto in un mercato volto alla produzione di servizi internet⁴⁴. Nel percorso che ha portato a questo punto si possono individuare due momenti fondamentali: la grande espansione di internet e l'avvento dei cosiddetti *smart-devices*⁴⁵.

⁴² Tale circostanza viene desunta dai dati che riguardano il mercato delle vendite dei computer. In uno studio Gartner, Inc. (NYSE: IT), che considera le vendite dal 1981 al 2015 si nota come si registrano picchi nelle vendite negli anni 1985-1990 e 1995-2000, ossia in prossimità di quelle che abbiamo analizzato essere i periodi in cui vengono immessi prodotti fortemente innovativi. In Rete: <https://www.extremetech.com/wp-content/uploads/2016/01/computer-shipments-2015.png>. Il dato risulta interessante se raffrontato con quello relativo all'andamento dei prezzi medi del personal computer, che, soprattutto a partire dagli anni 90, subirono un grosso calo. Fonte: <https://www.bls.gov/data/#prices>.

⁴³ Per una panoramica dell'incidenza delle rivoluzioni tecnologiche sul diritto, ed in particolare dell'ultima rivoluzione, quella digitale, si veda su tutti G. PASCUZZI (a cura di), *Il diritto nell'era digitale*, il Mulino, Bologna, 2016.

⁴⁴ Per un approfondimento in questo senso v. E. DE SIMONE, *Storia economica: dalla rivoluzione industriale alla rivoluzione informatica*, Angeli, Milano, 2011.

⁴⁵ Il termine *smart device* comprende tutti quei dispositivi dotati di portabilità e che possono connettersi alla rete attraverso diversi protocolli wireless come Bluetooth, NFC, Wi-Fi, 3G, etc. Per esempio rientrano nella categoria i computer portatili, i tablet, e gli smartphone.

Internet è una rete in grado di mettere in comunicazione i dispositivi di tutto il mondo. A partire dal 1995 cominciò a divenire il principale strumento di comunicazione di massa, e nel biennio 1998/1999 la diffusione del World Wide Web⁴⁶ assunse ritmi esponenziali⁴⁷.

Internet nacque per mettere in comunicazione tutti gli utenti ed i loro dispositivi, indipendentemente dalla loro dotazione hardware e software, garantendo così l'intercomunicazione tra sistemi differenti. Internet è una rete onnicomprensiva e non discriminatoria. Tutti i computer parlano la stessa lingua, o meglio, hanno lo stesso protocollo di comunicazione. Internet consiste in un insieme di protocolli di comunicazione. I protocolli non sono proprietari, ma pubblici, quindi tutti li possono usare. Ad esempio "http" è un protocollo applicativo alla base del web. Ma internet non è solo questo; oggi internet è un insieme di servizi come ad esempio motori di ricerca, streaming, podcasting, web radio, social networks, che aggiungono importanti funzionalità per tutti gli utenti di computer.

L'avvento e la diffusione di Internet e dei suoi servizi hanno rappresentato una vera e propria rivoluzione tecnologica⁴⁸, nonché uno dei principali elementi di sviluppo economico nell'ambito dell'*Information and Communication Technology* (ICT)⁴⁹.

È sicuramente anche grazie ad internet ed alla miriade di servizi offerti che il computer, strumento essenziale per accedere alla rete, crebbe ancora di importanza⁵⁰.

Con il susseguirsi degli anni la rete diventa sempre più grande, e i dispositivi sempre più piccoli e portatili. Il progresso tecnologico migliora la rete internet rendendola mobile, grazie della tecnologia GSM, sempre più veloce, e con una copertura territoriale sempre maggiore.

Nascono così gli *smartphone*. Tali dispositivi uniscono le funzioni proprie della telefonia mobile a quelle proprie degli elaboratori elettronici. All'inizio uno smartphone non era altro che un cellulare. L'accezione odierna che si conferisce alla parola è tuttavia diversa e identifica un bene ulteriore. L'impulso allo sviluppo del mercato degli smartphones, avviene nel 2007 con l'avvento del

⁴⁶ WWW o Web è uno dei principali servizi internet che consente la consultazione, detta anche navigazione, di moltissimi contenuti. La facilità nella ricerca delle informazioni fu resa possibile grazie allo sviluppo di web browser sempre più efficienti, il cui compito è quello di selezionare i contenuti presenti nel web sulla base di un algoritmo, che collega la ricerca dell'utente alle pagine di presumibile maggior interesse per lo stesso. Per citare i più famosi: Google Chrome, Safari, Internet Explorer, Mozilla Firefox. Per un approfondimento sulle diverse tipologie di reti internet, v. CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, pp. 338-354; sui motori di ricerca v. pp. 381-383.

⁴⁷ Si veda G. T. ELMÌ. *Informatica e Diritto. Presupposti, storia, disciplina, insegnamento, "ius condendum"*, in *Informatica e diritto*, Vol. XXIII, 2014, n. 2, 85-123. In particolare nel paragrafo 3.3 ove si considera il periodo in questione come "il periodo della rete globale".

⁴⁸ *Ibidem*. In particolare il passaggio è evidenziato nel paragrafo 2.2 dove si analizza il passaggio "dall'informatica classica all'informatica della rete".

⁴⁹ "Le tecnologie dell'informazione e della comunicazione sono i metodi e le tecnologie che realizzano sistemi di trasmissione e di telecomunicazione nonché di ricezione ed elaborazione di informazioni". In questi termini Enciclopedia Treccani, in Rete: http://www.treccani.it/enciclopedia/ict_%28Dizionario-di-Economia-e-Finanza%29/.

⁵⁰ Per approfondire il "tempo di internet" v. CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, pp. 357-401.

primo iPhone⁵¹. Anche in questo frangente storico, grazie all'implementazione di una tecnologia hardware e software *user-friendly* si rivoluzionò il modo di concepire il "telefonino".

Tutti quei servizi internet che erano fruibili a pieno solo tramite computer, diventano ora mobili e portatili. Lo sviluppo in termini di diffusione nel mercato che hanno avuto gli *smart devices* è stata, ed anche oggi è, maggiore rispetto alla diffusione dei classici computer. Ma i dati mostrano un ulteriore ed elemento di riflessione, ossia la circostanza che l'utenza, per usufruire di internet e dei servizi connessi, si rivolge principalmente a piattaforme mobili⁵².

È cambiato il modo di utilizzare il computer e internet, e ciò sicuramente a vantaggio dell'industria ICT. I contatti tra utente e macchina sono aumentati, le persone sono sempre connesse, così come lo sono i loro dispositivi⁵³. Di conseguenza sono cambiati i software che "servono" a questo tipo di mercato. Gran parte delle moderne *start up* investono tempo e denaro nella progettazione e programmazione di software volti alla produzione di servizi internet usufruibili attraverso dispositivi mobili.

Si pensi ad esempio al grandissimo mercato delle Applicazioni, che implementano sterminate funzioni e servizi, al mondo dei social network, agli strumenti di messaggistica, ai servizi di acquisto o vendita di beni on-line. Il mercato ICT ha assunto con il tempo dimensioni molto rilevanti. Analizzando i dati si nota come la parte più rilevante di tale mercato sia rappresentata dai servizi, seguiti dai dispositivi, dai software e dalla pubblicità digitale⁵⁴. Anche quando si parla di servizi, si deve rilevare, come spesso e volentieri questi siano implementati ed immessi nel mercato a mezzo di software. Tutti questi trovati, sono spesso tutelati anche mediante brevetto o brevetti⁵⁵.

Nell'odierna società dell'informazione, il software riveste quindi un importante ruolo economico. È lo stesso legislatore europeo a ricordarlo nel

⁵¹ Così ISAACSON, *Steve Jobs*, pp. 465-476. Circostanza confermata dai dati, che evidenziano come il 2007, sia stato l'anno cruciale che, nel mercato dei sistemi operativi, portò al successo determinati prodotti (IOS, Android), a scapito di altri (Palm, RIM, Symbian). Dati comScore U.S. smartphone market share (2005-2012) e nello stesso senso Needham & Co. analyst Charles Wolf, Global smartphone OS shares (2005-2008).

⁵² Dati audiweb, IAB Seminar Mobile 2016, in Rete: <http://www.audiweb.it/news/audiwebiab-seminar-mobile-2016/>. Nel rapporto si legge che "Il 73% del tempo totale dedicato alla navigazione è generato dalla fruizione dai device mobili (smartphone e tablet), +48% negli ultimi due anni".

⁵³ V. L. MANOVICH, *Software culture*, Edizioni Olivares, Milano, 2010, che definisce la società odierna come una "*software society*", dove il software è l'elemento che funge da collante in relazione al sistema economico, sociale e anche culturale. Soprattutto in merito a tale ultimo elemento l'autore sottolinea come il software sia di fatto lo strumento principale di acquisizione, distribuzione e produzione della cultura. Circostanza questa che non deve essere interpretata secondo una accezione completamente negativa, a patto che si abbia coscienza delle dinamiche intorno allo strumento in questione.

⁵⁴ Dati Gartner Market Databook, relativi al 2017. In Rete: <http://www.gartner.com/technology/research/it-spending-forecast/>.

⁵⁵ Se si analizzano, relativamente ad EPO, i campi tecnologici con più applicazioni, si nota come le "digital communication" e le "computer technology" ricoprano il terzo ed il quarto posto. Questi settori possono dirsi in un certo senso complementari e hanno a che fare, direttamente o indirettamente con il software. Se ricomponessimo queste due categorie si otterrebbero quasi 20000 *applications*, il doppio della "medical technology" tutt'ora al primo posto. Dati forniti da E. Archontopoulos, EPO, relativi al mese di novembre del 2015.

considerando numero tre della direttiva 2009/24/CE relativa alla tutela giuridica dei programmi per elaboratore, dove si evidenzia la sua fondamentale importanza per lo sviluppo industriale della comunità stessa: *“I programmi per elaboratore hanno un ruolo sempre più importante in una vasta gamma di industrie e, di conseguenza, si può affermare che la tecnologia dei programmi per elaboratore riveste una fondamentale importanza per lo sviluppo industriale della Comunità”*⁵⁶.

Negli ultimi anni l'innovazione è passata anche e soprattutto attraverso il software. Non solo; il software è al tempo stesso veicolo e strumento di innovazione nonché costituisce esso stesso innovazione.

Da mezzo dell'industria, l'informatica è diventata il fine dell'industria.

2. La particolare natura del software. I beni informazionali

Il software è un bene particolare sotto due aspetti. Il primo aspetto attiene all'analisi economica, mentre il secondo aspetto richiede una analisi giuridica.

Entrambi gli elementi sono sintetizzati nel considerando numero due della direttiva 2009/24/CE: *“Per creare programmi per elaboratore è necessario investire considerevoli risorse umane, tecniche e finanziarie, mentre è possibile copiarli a un costo minimo rispetto a quello necessario per crearli autonomamente”*⁵⁷. L'intreccio tra le discipline si realizza perché il software, sia esso incorporato o meno su un supporto tangibile, richiede costi di elaborazione e creazione molto alti, mentre invece i costi di riproduzione di una copia sono prossimi allo zero.

La circostanza avviene in considerazione del fatto che il software è un bene digitale, o anche detto bene di informazione o informazionale. Il valore economico di questa categoria di beni *“non dipende dalle sue caratteristiche fisiche o materiali, ma dalle informazioni che contiene”*⁵⁸.

Il *costo marginale di produzione*⁵⁹ di un bene di informazione è molto basso. Si pensi ai costi sostenuti da Microsoft per la realizzazione di una copia del pacchetto Microsoft Office. Il costo marginale di produzione risiede negli oneri di cui si fa carico la società per l'incorporazione del software all'interno del supporto materiale (CD), l'imballaggio e la distribuzione del prodotto. Questi costi sono molto bassi e incidono per una percentuale trascurabile sul prezzo del bene. Si deve inoltre constatare come a oggi i costi marginali sono di fatto nulli; la tendenza delle case produttrici è infatti quella di rendere disponibile il programma attraverso il download del prodotto digitale, di fatto elidendo quelli che sono i costi della incorporazione su supporto e successiva distribuzione nel mercato. Ciò non significa tuttavia che i costi di produzione di un software siano bassi.

⁵⁶ Direttiva 2009/24/CE del parlamento europeo e del consiglio, del 23 aprile 2009, relativa alla tutela giuridica dei programmi per elaboratore, considerando n. 3.

⁵⁷ Direttiva 2009/24/CE, considerando n. 2.

⁵⁸ P. R. KRUGMAN, R. WELLS, *Microeconomia*, Zanichelli, Bologna, 2013, cit. 486.

⁵⁹ Per costo marginale di produzione si intende *“il costo sostenuto per la produzione di una unità aggiuntiva”*, v. *ibidem*, cit. 176.

I *costi fissi* di produzione di un prodotto digitale sono molto alti. Per produrre un software è infatti necessaria una lunga fase di programmazione ad opera di personale tecnico qualificato, su supporti informatici adeguati. Tale fase può durare anche diversi anni durante i quali molte persone sono coinvolte non solo per progettare, ma anche per testare il prodotto. Tali costi sono appunto detti fissi perché “*non dipendono dal numero di copie prodotte e distribuite*”⁶⁰. Che venda milioni di copie o non venga messo in commercio, il costo di produzione di un software è il medesimo. Il prezzo di un software non può quindi consistere nel costo marginale di produzione. Le imprese che innovano il mercato devono poter recuperare i propri costi fissi e realizzare un profitto affinché abbiano un incentivo a produrre il bene.

Il valore di un software per il consumatore finale può dipendere non solamente dal suo contenuto e dalle funzioni che offre, ma anche dalla quantità di utenti che lo possiedono o lo utilizzano. Il fenomeno è denominato *esternalità di network*⁶¹, e vi sono soggetti tutti quei beni, tra cui è opportuno annoverare alcuni software, il cui “*valore per un individuo è tanto maggiore quanto è estesa l’utenza*”⁶².

L’esempio più eclatante è costituito dai social network. Il fine ultimo di qualsiasi social network è creare una rete di persone più vasta possibile; ciò accade perché più ampia è la rete di riferimento e più utenti saranno propensi a farne parte. Un altro esempio rilevante, nel settore della messaggistica istantanea lo ritroviamo in WhatsApp. Molti più utenti saranno propensi ad utilizzare un determinato servizio di messaggistica, quanto maggiori saranno gli utenti che, tramite la stessa, potranno contattare. Il tratto comune di queste applicazioni risiede nel grande successo che le stesse hanno avuto in un breve lasso di tempo; questo perché le esternalità di rete sono alla base di una caratteristica peculiare dei beni di informazione: il *feedback positivo*. Se infatti il successo del bene è legato all’aumento dell’utenza, ciò significa che se un grande numero di individui fruisce del bene è probabile che anche altri lo facciano. Allo stesso modo, se gli utenti non utilizzano il prodotto è meno probabile che altri lo utilizzino; “*Successo e fallimento tendono ad autoalimentarsi*”⁶³.

Ma quale è la via per raggiungere tale successo? L’analisi economica teorizza che una volta raggiunta una certa dimensione di utenza, detta *massa critica*⁶⁴, il network “esplode”. I percorsi di crescita delle applicazioni sopracitate sono infatti molto simili; inizialmente si rileva un periodo prolungato di crescita lenta. In questa fase il prodotto sazia la *domanda iniziale incondizionata*. Successivamente si ha una espansione rapida e improvvisa che può essere causata da diversi fattori; riprendendo gli esempi già citati, Facebook raggiunge la massa critica implementando costantemente funzioni aggiuntive e rendendo

⁶⁰ *Ibidem*, cit. 487.

⁶¹ Il fenomeno è detto esternalità di network, o esternalità di rete, perché le ipotesi più eclatanti di detto fenomeno si verificano quando i beni coinvolti formano appunto una rete di comunicazione (telegrafo, telefono, fax, indirizzi di posta elettronica, messaggistica).

⁶² KRUGMAN, WELLS, *Microeconomia*, cit., 491.

⁶³ *Ibidem*, cit., 492.

⁶⁴ La massa critica è quella “*soglia superata la quale un network comincia a crescere rapidamente*”, v. *ibidem*, cit., 495.

disponibile il servizio in mercati sempre più ampi. WhatsApp invece “esplode” nel momento in cui rende gratuita l'applicazione⁶⁵. Quella che era *domanda condizionata* viene ora rapidamente saziata. Quando quest'ultima viene soddisfatta, il mercato diviene maturo e l'espansione scema lentamente, a meno che non si riesca, attraverso un nuovo step innovativo a raggiungere un nuovo livello di domanda condizionata.

In queste applicazioni l'esternalità di rete si è manifestata in maniera diretta, ma il fenomeno è rilevante e interessante anche nella sua forma indiretta. L'esempio classico di una esternalità di network indiretta è quella dei sistemi operativi.

Nella maggior parte dei computer di tutto il mondo è installato Microsoft Windows⁶⁶. Una minoranza significativa di utenti di computer possiede prodotti Apple, i quali hanno un proprio sistema operativo, Macintosh. Infine un numero esiguo di personal computer utilizzano Linux, un sistema operativo open source. Perché Windows ha un così forte dominio sul mercato? Sarebbe sicuramente errato affermare che un computer su cui è installato Windows, è utile nella misura in cui la maggior parte dell'utenza utilizza Windows. Non vi è una esternalità di rete diretta che rende Windows il sistema operativo preferito.

Ciononostante il dominio di Microsoft si è nel tempo autoalimentato anche per via di due importanti ragioni: essendo Windows il sistema più diffuso, è più facile per un utente Microsoft ottenere assistenza e consigli. Secondariamente, essendo ampia la quota di mercato posseduta, Windows attira maggiormente l'attenzione degli sviluppatori di software applicativi. Di conseguenza le applicazioni disponibili per Windows sono maggiori, e ciò rappresenta un fattore ulteriore con il quale viene attirata l'utenza.

Nel 1984 la Apple Computer lanciò Macintosh, un computer fortemente innovativo. Tale prodotto ha introdotto per primo nel mercato l'utilizzo di una interfaccia utente grafica avanzata, che attraverso il sistema “punta e clicca” permetteva di compiere operazioni mediante l'utilizzo di un cursore. Prima di tale invenzione il funzionamento di un personal computer avveniva attraverso la scrittura di comandi, o al più scegliendo un comando attraverso un menù. Il sistema operativo di questo tipo più diffuso era DOS, e la variante più diffusa era creata da Microsoft. Apple era conscia di avere un prodotto nettamente superiore, e dunque decise di venderlo ad un prezzo più elevato. L'utenza rimase tuttavia legata al DOS, benché tecnologicamente inferiore. Questo perché MS-DOS era ampiamente disponibile e decisamente meno caro. Tale situazione trovò conferma negli anni successivi. Microsoft sviluppò infatti una propria interfaccia grafica, Windows, e l'utenza vi rimase fedele nonostante nelle sue prime versioni vennero riscontrati problemi rilevanti, come la lentezza nell'avviamento e le frequenti paralisi del sistema. Anche in tal caso le

⁶⁵ Dati comScore Media Metrix, che incrociano i primi quattro anni di crescita sul mercato dopo il lancio dei prodotti in oggetto. In Rete: <https://i1.wp.com/www.zerozone.it/wp-content/uploads/2014/02/crescita-whatsapp.png>.

⁶⁶ Secondo gli ultimi dati dell'anno corrente (2017) forniti da netmarketshare.com la quota di mercato occupata dai sistemi operativi Microsoft, che comprendono diverse versioni di Windows, è di circa il 90%. Mac OS X si attesta intorno al 7%, mentre Linux ricopre il 2%.

esternalità già raggiunte dal DOS, hanno convinto la maggior parte dell'utenza a non abbandonare Microsoft benché tecnologicamente inferiore⁶⁷.

Un vantaggio iniziale può quindi decretare il trionfo di un prodotto sul suo diretto concorrente. Quando infatti il mercato si pronuncia a favore di un bene, l'altro prodotto gradualmente va a ricoprire sul mercato una posizione marginale, di nicchia, oppure scompare. Questo perché *“il vantaggio iniziale si autoalimenta nel momento in cui la diffusione del prodotto raggiunge la sopraddeffta massa critica e di conseguenza la domanda del diretto concorrente si contrae”*⁶⁸. Tale fenomeno prende il nome di *tipping*. Col senno di poi, non sarebbe assurdo ipotizzare, sulla base di quanto illustrato, che se Apple, comprendendo la forza delle esternalità, avesse abbassato il prezzo dei propri prodotti, avrebbe dominato il mercato⁶⁹.

3. La necessità di una tutela del software tra progresso e incentivo alla creatività

La necessità di una forte protezione del software si è palesata nel momento in cui si è compresa la vulnerabilità del prodotto digitale. Come ricordato nel sopracitato considerando 2 della direttiva 24/2009/CE, i costi fissi necessari per creare un software sono molto alti, mentre invece i costi della singola copia sono prossimi allo zero⁷⁰.

Qualora un soggetto terzo rispetto all'autore vendesse o distribuisse il prodotto, verrebbe frustrato l'investimento iniziale. Tale circostanza è la base della tutela della proprietà intellettuale, ma il software, essendo un bene informazionale, è, se vogliamo, ancor più sensibile a tali problematiche⁷¹.

Un primo problema attiene alla facilità di duplicazione. Duplicare un software è una operazione attraverso la quale si ricrea un prodotto assolutamente equivalente all'originale, senza compiere sforzi in termini economici o temporali. Inoltre, una volta duplicato un bene digitale, è anche estremamente facile la trasmissione dello stesso. Attraverso la rete internet è infatti possibile una distribuzione del prodotto efficiente, su larga scala e a costi zero. Infine, chi possiede determinate competenze informatiche può modificare

⁶⁷ In questo senso anche CERUZZI, *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, p. 318.

⁶⁸ KRUGMAN, WELLS, *Microeconomia*, cit., 495.

⁶⁹ In questo senso v. *ibidem*, p. 497.

⁷⁰ Circostanza sottolineata anche dalla Commissione CONTU, *Final Report*, 1978: *“The cost of developing computer programs is far greater than the cost of their duplication”*, cit., Cap. 3, Computers and Copyright, Foundation for the Recommendations, Computer Programs.

⁷¹ Circostanza enfatizzata dal legislatore europeo nella Direttiva 2001/29/CE del parlamento europeo e del consiglio, del 22 maggio 2001, sull'armonizzazione di taluni aspetti del diritto d'autore e dei diritti connessi nella società dell'informazione. Su tutti il considerando numero 10: *“Per continuare la loro attività creativa e artistica, gli autori e gli interpreti o esecutori debbono ricevere un adeguato compenso per l'utilizzo delle loro opere, come pure i produttori per poter finanziare tale creazione. Gli investimenti necessari a fabbricare prodotti quali riproduzioni fonografiche, pellicole o prodotti multimediali e servizi quali i servizi su richiesta («on-demand») sono considerevoli. È necessaria un'adeguata protezione giuridica dei diritti di proprietà intellettuale per garantire la disponibilità di tale compenso e consentire un soddisfacente rendimento degli investimenti”*.

un software con relativa facilità, riproponendo il prodotto modificato come originale. Tutte queste attività sono estremamente lesive per l'industria di software.

È importante che chi crea un software possa venderlo, o comunque commercializzarlo, godendo di una esclusiva su quel prodotto. Tale esclusiva non può tuttavia essere illimitata né nel tempo, né nella sostanza.

Le forme di tutela della proprietà intellettuale, che si realizzino mediante copyright o mediante brevetto, sono pensate per instaurare un bilanciamento di interessi. Tale bilanciamento è una questione che ha sempre insistito storicamente nelle dinamiche di protezione della proprietà intellettuale⁷². L'innovazione va tutelata affinché chi investe energie e danaro nella creazione ne sia ripagato. Questo incentivo, di carattere economico, deve tuttavia essere bilanciato con il fatto che l'innovazione non possa essere concepita come una esclusiva infinita nelle mani di un soggetto.

Deve quindi essere predisposta una limitazione alla privativa concessa, sia dal punto di vista sostanziale che temporale. Questo perché l'innovazione non solo può portare utilità alla collettività, ma anche perché genera essa stessa innovazione.

Il punto in questione è ben evidenziato anche nella costituzione degli Stati Uniti, ove si statuisce che *“il congresso avrà il potere di di promuovere il progresso della scienza e delle arti utili, assicurando per periodi limitati di tempo agli Autori ed agli Inventori il diritto esclusivo sui loro scritti e scoperte”*⁷³.

Per promuovere il progresso da un lato si devono incentivare gli autori e gli inventori concedendo loro una esclusiva, un monopolio, ma dall'altro si deve anche permettere la circolazione delle informazioni, limitando l'esclusiva nel tempo e nella sostanza. L'utilizzo di diversi strumenti di tutela della proprietà intellettuale applicati al software risiede proprio nella diversa volontà di limitare in modo più o meno penetrante tale esclusiva realizzando un bilanciamento di interessi più o meno equo.

Come afferma il legislatore europeo, *“Ogni armonizzazione del diritto d'autore e dei diritti connessi dovrebbe prendere le mosse da un alto livello di protezione, dal momento che tali diritti sono essenziali per la creazione intellettuale. La loro protezione contribuisce alla salvaguardia e allo sviluppo della creatività nell'interesse di autori, interpreti o esecutori, produttori e consumatori, nonché della cultura, dell'industria e del pubblico in generale”*⁷⁴.

La questione è quindi importante sotto molti punti di vista, e la circostanza che il software sia un bene di informazione rende ancora più delicato tale problema.

⁷² Su tutti, con riferimento al diritto d'autore, si veda I. UMBERTO, *Alle origini del copyright e del diritto d'autore, Tecnologia, interessi e cambiamento giuridico*, Carocci (collana Biblioteca di testi e studi), Roma, 2010. Per una panoramica storica con riferimento allo strumento brevettuale v. C. A. NARD, *The Law of Patents*, Wolters Kluwer, Aspen Publishers, New York, 2008, in particolare il cap. 1, *“History and Architecture of the Patent System”*, pp. 1-33; e, sul bilanciamento di interessi pp. 64-67.

⁷³ Costituzione degli Stati Uniti d'America, articolo 1, sez. 8, clausola 8, trad.

⁷⁴ Direttiva 2001/29/CE, considerando n. 9.

4. Definizioni di software tra diritto e informatica

Ma cosa è in concreto un software? Dal punto di vista tecnico una definizione completa di software non è semplice da dare. Per iniziare la disamina, si analizzerà come il legislatore europeo abbia inquadrato il bene giuridico in questione.

Una prima definizione viene offerta all'interno della direttiva 2009/24/CE. Al considerando numero sette si individua l'oggetto della tutela, il software o programma per elaboratore, che viene tutelato in tutte le sue forme. Questa "non definizione" non ci dice cosa è un software ma serve per individuare e circoscrivere l'oggetto della tutela, il programma per elaboratore, che viene per l'appunto protetto in tutte le sue forme⁷⁵.

In tale frangente il legislatore vuole assicurare una tutela al software indipendentemente dal supporto su cui lo stesso insiste. Il software, infatti, essendo un bene digitale può essere conservato, distribuito o venduto attraverso diverse modalità. Anzitutto può essere preinstallato nell'hardware con il quale viene commercializzato. Si pensi ad esempio alla dotazione di software, invero molto ricca, che si trova già installata all'interno di un personal computer, oppure di uno smartphone, al momento dell'acquisto. La tendenza odierna consiste nel predisporre determinati pacchetti software, composti sia dal software di base, o sistema operativo, che da software applicativi, per l'esercizio delle funzioni e delle utilities principali, installati e pronti all'uso nei dispositivi⁷⁶.

In passato, invece, il software veniva acquistato a parte e veniva distribuito e venduto attraverso supporti tangibili, i CD-ROM, che l'utente installava nel proprio hardware. È inoltre importante notare come a oggi il modello generale di commercializzazione e distribuzione dei software si basa sull'acquisto on-line con conseguente download del file. Si deve quindi guardare al contenuto e non al contenitore.

L'importanza del software risiede nelle informazioni che contiene, perché, come si è illustrato, è un bene di informazione. Anche per questo motivo la tutela del legislatore europeo, come specificamente richiamato dall'articolo 1 della Direttiva 2009/24/CE, si estende ai lavori preparatori per la realizzazione di un programma⁷⁷. Considerati gli investimenti che essi richiedono, e dato che portano comunque a un determinato risultato, si ritengono meritevoli di tutela alla stregua di un "software finito", qualora tali lavori, se ulteriormente sviluppati in una successiva fase, portino alla realizzazione di un programma⁷⁸.

Una definizione funzionale la troviamo invece nel considerando numero dieci della Direttiva 2009/24/CE. In questo frangente il legislatore europeo ci

⁷⁵ V. Direttiva 2009/24/CE, art. 1, comma 1.

⁷⁶ Sulla differenza tra software applicativo e software di base, e sulle problematiche accessorie v. *infra*, par. 5.

⁷⁷ Direttiva 2009/24/CE, art. 1, comma 1: "Ai fini della presente direttiva, il termine «programma per elaboratore» comprende il materiale preparatorio per la progettazione di un programma".

⁷⁸ In particolare, sulla tutela dei lavori preparatori, v. *infra*, cap. II, par. 1.

dice cosa un software fa, ma non cosa è. La definizione fornita⁷⁹ è tuttavia interessante perché utile a comprendere il compito del software.

La funzione primaria di un software è quella di creare un collegamento. All'interno di un sistema i collegamenti sono molteplici; infatti, esistono software progettati per creare una comunicazione tra la macchina e l'utente; si pensi anzitutto ai sistemi operativi, la cui ideazione nasce dalla volontà di creare una diretta e facile interazione tra utente, macchina ed altri software. Questi ultimi, come vedremo nel paragrafo successivo, sono detti software applicativi e svolgono invece il compito di ricevere istruzioni dall'utente ed elaborarle affinché questi possa, mediante l'utilizzo del programma, svolgere le funzioni per le quali lo stesso è preposto.

Esistono quindi software che non possono funzionare senza altri software, così come è un dato di fatto rilevare la complementarietà tra hardware e software: l'uno non può funzionare senza l'altro.

È importante capire cosa è in concreto un software per valutare su cosa ricada la tutela. Il legislatore statunitense per primo diede una definizione sostanziale di software nel 1978, descrivendolo come *“una serie di comandi e di istruzioni usate direttamente o indirettamente in un computer al fine di avere un certo risultato”*⁸⁰.

Sempre nello stesso anno l'Organizzazione Mondiale per la Proprietà Intellettuale (OMPI) pubblicò un documento sulla protezione del software definendolo come *“un insieme di istruzioni che se registrate su uno strumento leggibile da una macchina, possono essere elaborate al fine di selezionare, eseguire, o produrre una determinata funzione, attività o risultato”*⁸¹.

Anche sulla base di queste definizioni, il legislatore europeo, nella proposta originaria di quella che diverrà la direttiva 91/250/CE sulla tutela giuridica dei programmi per elaboratore, definì il software come *“l'espressione in ogni forma, linguaggio, notazione o codice di un insieme di istruzioni il cui scopo è quello di permettere ad un congegno di elaborazione di informazioni, un computer, di svolgere le proprie funzioni”*⁸². Tale definizione è poi stata eliminata, non venendo inclusa né nel corpo, né nei considerando della sopracitata direttiva per una precisa scelta del legislatore, che ritenne come una

⁷⁹ Direttiva 2009/24/CE, considerando n. 10: *“I programmi per elaboratore svolgono la funzione di comunicare e operare con altri componenti di un sistema informatico e con gli utenti; a tale scopo è necessaria un'interconnessione e un'interazione logica e, ove opportuno, materiale per consentire a tutti i componenti software e hardware di operare con altri software e hardware e con gli utenti in tutti i modi in cui sono destinati a funzionare. Le parti del programma che assicurano tale interconnessione e interazione fra gli elementi del software e dell'hardware sono generalmente denominate «interfacce». Tale interconnessione e interazione funzionale è generalmente denominata «interoperabilità»; l'interoperabilità può essere definita come la capacità di due o più sistemi di scambiare informazioni e di usare reciproca mente le informazioni scambiate”*.

⁸⁰ La definizione maturata in seno alla commissione CONTU, fu adottata dal congresso per modificare il paragrafo 101 del Copyright Act inserendovi la definizione di “computer program” come *“a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result”* (17 U.S.C. S. 101).

⁸¹ Documento WIPO, *Model provision on the protection of computer software*, Ginevra, 1978, trad., pp. 4-5. In Rete: http://www.wipo.int/mdocsarchives/LPCS_I_79/LPCS_I_2_E.pdf.

⁸² Proposal for a Council Directive on the legal protection of computer programs, *Explanatory Memorandum*, 1989, trad., p. 3. In Rete: [http://aei.pitt.edu/13138/1/COM_\(88\)_816_final.pdf](http://aei.pitt.edu/13138/1/COM_(88)_816_final.pdf).

definizione fornita allo stato dell'arte, sarebbe stata in breve tempo divenuta obsoleta⁸³.

In questi frangenti i legislatori fanno riferimento a concetti propri dell'informatica. È quindi necessario addentrarci in tale ambito per avere una definizione completa ed esauriente del bene giuridico in questione.

In estrema sintesi si definisce software un insieme di procedure e di istruzioni redatte secondo un determinato linguaggio che permettono alla macchina di eseguire determinate funzioni⁸⁴. Ma cosa si intende per linguaggio? Il software è costituito fondamentalmente da due codici: il codice sorgente ed il codice oggetto.

Il codice oggetto, o codice binario, impartisce comandi all'elaboratore, attraverso sequenze ed alternanze logiche, sviluppate attraverso formule matematiche, gli algoritmi. Questi sono costituiti da un insieme di istruzioni che definiscono una sequenza di operazioni⁸⁵.

Il programma non può tuttavia essere scritto in questo linguaggio (detto anche linguaggio macchina). È quindi scritto attraverso il codice sorgente, ossia il linguaggio di programmazione, che, attraverso un ulteriore componente software, generalmente un compilatore, viene tradotto in codice binario⁸⁶.

Nella realizzazione di un software il punto fondamentale è quindi la sua scrittura in linguaggio di programmazione, che porterà alla realizzazione del codice sorgente, il quale viene scritto sulla base di quegli algoritmi studiati in via preliminare.

Il codice sorgente non è tuttavia comprensibile dalla macchina. Il programma viene quindi tradotto attraverso software specifici: i compilatori e gli interpreti. La differenza tra queste due tipologie di software risiede nella circostanza che nei primi la traduzione investe tutte le istruzioni del programma in un unico procedimento, mentre nei secondi, la traduzione viene effettuata di volta in volta, a seconda delle specifiche istruzioni del codice⁸⁷.

La scelta di utilizzo di un compilatore piuttosto che di un interprete, deve essere predeterminata prima della scrittura del codice sorgente, e la scelta dell'uno rispetto all'altro dipende da molte variabili quali la potenza della macchina per cui è pensato il programma, e la complessità del software progettato⁸⁸.

Una volta tradotto il codice sorgente in codice oggetto, questo riesce ad impartire comandi alla macchina.

⁸³ *Ibidem*: "The words computer program are not defined for the purposes of this article. It has been recommended by experts in the field that any definition in a directive of what constitutes a program would of necessity become obsolete as future technology changes the nature of programs as they are known today.", cit., 17.

⁸⁴ R. BORRUSO, *Computer e diritto, analisi giuridica del computer*, Giuffrè, Milano, 1998, p. 129; E. GIANNANTONIO, *Manuale di diritto dell'informatica*, CEDAM, Padova, 1997, p. 111.

⁸⁵ Sul concetto di algoritmo v. SANNA, *Manuale di informatica di base*, pp. 84-89.

⁸⁶ Per una analisi tecnica approfondita dei differenti linguaggi di programmazione, v. *ibidem* pp. 130-152; Con riguardo invece alle modalità in cui una informazione viene rappresentata e codificata v. *ibidem* pp. 27-81.

⁸⁷ In particolare sulla traduzione dei comandi v. *ibidem*, pp. 217-218.

⁸⁸ P. S. MENNEL, *An Analysis of the Scope of Copyright Protection for Application Programs*, 41 *Stanford Law Review*, 1045, 1055, (1988).

5. Software applicativi e software operativi. L'interoperabilità e il caso Microsoft

Il software è abitualmente classificato in software di base e software applicativo. I termini sono già stati utilizzati in precedenza ma è bene illustrare tali elementi, cosa li differenzia nonché il rapporto che li lega.

Il primo, anche detto di sistema, comprende l'insieme dei programmi predisposti a favorire un efficiente impiego dell'hardware, e generalmente include i compilatori, gli interpreti, gli assemblatori⁸⁹ e i sistemi per la gestione delle basi di dati⁹⁰. Il suo cuore è costituito dal sistema operativo. Questi programmi svolgono funzioni di servizio e di gestione del sistema di elaborazione. I compiti del software di sistema possono sintetizzarsi in: gestione e controllo di tutte le componenti hardware del computer; gestione dei file e delle memorie di massa che li contengono; svolgimento del ruolo di interfaccia tra l'uomo e la macchina, permettendo all'utente di interagire con le componenti hardware. Si sono già citati in precedenza esempi eloquenti di software operativi come MS-DOS, Windows e Macintosh.

Il secondo è costituito dai programmi volti alla risoluzione di singoli problemi nonché alla predisposizione delle singole funzioni ed allo svolgimento di determinati compiti⁹¹. Ad esempio, prendendo in considerazione i software applicativi più diffusi, ritroviamo programmi volti alla elaborazione testi (scrittura, formattazione, stampa); oppure programmi che implementano fogli di calcolo (inserimento ed analisi di dati numerici); ancora, software adibiti alla gestione di basi di dati (creazione e gestione di archivi dati tra loro correlati) e browser Web che consentono di sfogliare gli ipertesti che formano il "www." come ad esempio Internet Explorer, Firefox, Netscape.

Le due tipologie di software sono tra loro molto diverse ma complementari. Un software applicativo non può funzionare senza l'appoggio di un sistema operativo su cui "girare". Allo stesso tempo un sistema operativo senza alcun software applicativo è di fatto inutile, perché il dispositivo hardware-software non sarebbe in grado di svolgere alcuna funzione.

Lo sviluppo di un software applicativo non può prescindere dal considerare la struttura del sistema operativo sul quale dovrà svolgere le funzioni a cui è preposto. Tale circostanza può portare a numerosi problemi, soprattutto se il produttore del software di base non rende noto ai programmatori del software applicativo determinate informazioni.

Tale circostanza avviene in considerazione del fatto che, per avere il corretto funzionamento di un programma, è necessario che questo riesca a scambiare correttamente le informazioni con un altro programma: tale forma di comunicazione prende il nome di interoperabilità.

⁸⁹ Nel precedente paragrafo si è spiegato cosa sono gli interpreti e i compilatori. Gli assemblatori, trasformano il linguaggio *assembly* in linguaggio macchina. La funzione è quindi la stessa degli interpreti e dei compilatori con la differenza del particolare linguaggio tradotto. Il linguaggio assemblativo è infatti un linguaggio di programmazione, vicino e simile al linguaggio macchina vero e proprio.

⁹⁰ SANNA, *Manuale di informatica di base*, pp. 195-196.

⁹¹ V. *ibidem*.

Nel 1998 il dipartimento di giustizia degli Stati Uniti e 19 Stati americani, intrapresero un procedimento antitrust contro Microsoft, accusando l'azienda di monopolio e di pratiche commerciali sleali. In quell'anno Microsoft era l'azienda produttrice di software di maggior successo, e il suo fondatore Bill Gates era già l'uomo più ricco del mondo⁹².

Microsoft aveva costituito un monopolio; su quasi tutti i computer del mondo era installato il sistema operativo Windows. Le questioni sollevate dal governo statunitense, tuttavia, non erano volte allo smantellamento del monopolio di Microsoft nel mercato dei sistemi operativi. La situazione monopolistica, nell'industria informatica, può derivare da conseguenze naturali e non deve, in questa sua forma, essere ostacolata⁹³.

Il governo sosteneva tuttavia che Microsoft avesse usato la sua posizione da monopolista nel settore dei sistemi operativi, per assicurare ai propri prodotti, facenti parte di altri segmenti di mercato, un vantaggio ingiusto sui concorrenti. Per fare degli esempi, Microsoft includeva all'interno del suo sistema operativo Internet Explorer, prevalendo in tale modo nel mercato dei web browser, a scapito della concorrenza.

Ma gli ostacoli creati da Microsoft alla concorrenza non si sostanziano solamente in questo tipo di pratiche commerciali. L'architettura di Windows era tale da rendere molto difficile la compatibilità tra il sistema operativo di Microsoft e i software applicativi di terze parti. I software sviluppati dalla concorrenza non riuscivano ad integrarsi con il sistema operativo, circostanza questa che porta il verificarsi di determinati problemi tecnici. Tale fatto indirizzò l'utenza verso le applicazioni Microsoft, non soggette a questi problemi.

Attraverso tali pratiche commerciali, Microsoft creava ulteriori monopoli, scoraggiando così l'innovazione. Secondo l'opinione del governo, gli operatori del mercato di software applicativi non erano disposti ad investire somme considerevoli nei loro prodotti, perché temevano che Microsoft, sfruttando la sua posizione, potesse produrre una applicazione concorrente e commercializzarla con maggiore successo, indipendentemente dalla effettiva qualità del prodotto offerto, implementandola direttamente e senza alcuna difficoltà tecnica nel proprio sistema operativo.

Il 19 ottobre iniziò il processo. La corte federale fu presieduta dal giudice Thomas Jackson, che, nella sentenza preliminare del novembre del 1999 condannò Microsoft. Il 3 Aprile del 2000 fu emessa la sentenza definitiva, che, confermando la preliminare, ritiene colpevole la società di avere violato la legge

⁹² Per una analisi squisitamente economica del caso si veda KRUGMAN, WELLS, *Microeconomia*, pp. 499-500; per una analisi giuridica "a caldo" v. G. DE SANTIS, *La tutela brevettuale del software tra brevetto e diritto d'autore*, Giuffrè Editore, Milano, 2000, pp. 84-89. La sentenza relativa al caso in oggetto *United States v. Microsoft Corporation*, 253 F.3d 34 (D.C. Cir. 2001), in Rete: <http://law.justia.com/cases/federal/appellate-courts/F3/253/34/576095/>.

⁹³ Secondo la legislazione americana non è infatti illegale l'acquisizione di un monopolio attraverso un prodotto superiore rispetto a quello della concorrenza. È tuttavia illecito l'utilizzo di questa posizione di dominanza per il mantenimento del monopolio ovvero per la sua estensione in altri settori.

federale antitrust USA⁹⁴. Successivamente alla sentenza, il dipartimento di Giustizia e i diciannove Stati che si erano uniti nella causa chiedono formalmente la divisione della società di Bill Gates. Un mese dopo tale richiesta, il giudice ordinò lo smembramento dell'impresa in due rami: uno adibito alla produzione di software operativi, e l'altro rivolto alla produzione di software applicativi.

Microsoft presentò istanza di appello e chiese la sospensione della sentenza di primo grado. A tal fine la società presentò un rapporto della Securities and Exchange Commission, organo di controllo della borsa americana, sottolineando come la decisione del giudice Jackson avrebbe potuto causare un effetto avverso sul prezzo del titolo, il che avrebbe messo in serio pericolo non solo la società, ma anche, vista la larga diffusione azionaria di Microsoft, e il suo ruolo da assoluta protagonista nell'ambito dell'ICT, tutta l'industria delle tecnologie di informazione.

Il rischio di manovre speculative sulla borsa convinse la Corte d'Appello ad accettare la richiesta dei legali di Bill Gates e sospese l'ordinanza di primo grado.

Nel frattempo avvenne il cambio di amministrazione alla Casa Bianca e George W. Bush venne eletto Presidente degli Stati Uniti. Tale avvicendamento si fece sentire anche sul caso in corso, e il nuovo Dipartimento di giustizia rinunciò a chiedere lo smembramento di Microsoft.

Così, nel 2001, il dipartimento e Microsoft giunsero ad un accordo traslativo. Con tale accordo l'impresa si impegnavo a fornire ai produttori di software applicativi la tecnologia e le informazioni necessarie, affinché le loro applicazioni potessero facilmente integrarsi con il sistema operativo Windows e con tutte le altre applicazioni di Microsoft.

Tale accordo fu successivamente accettato anche da dieci Stati americani dei diciannove in causa.

I restanti stati, prevalentemente ad amministrazione democratica, che non aderirono all'accordo, continuarono il processo che si concluse il primo novembre del 2002. Il giudice Colleen Kollar-Kotelly approvò l'accordo amichevole, ritenendolo idoneo ad una equa risoluzione della controversia, e respinse le sanzioni supplementari chieste dai nove Stati. Solo lo stato del Massachusetts non aderirà a tale transazione.

Tali vicende sono molto importanti, anche se la loro risoluzione è stata, dal punto vista giuridico, deludente. La decisione del caso è stata di fatto condizionata da pesanti influenze prima finanziarie e poi politiche. L'accordo sottoscritto presentava innumerevoli scappatoie per Microsoft, che avrebbe potuto continuare ad abusare della sua posizione di monopolio nell'ambito del mercato dei sistemi operativi⁹⁵.

⁹⁴ I passaggi salienti della sentenza sono ben evidenziati in DE SANTIS, *La tutela brevettuale del software tra brevetto e diritto d'autore*, cit., 74-87. L'autore individua 4 punti: 1. La posizione monopolistica è stata utilizzata per "ostacolare lo sviluppo o l'entrata nel mercato dei pc"; 2. "La barra delle applicazioni Windows ha impedito l'interoperabilità con altri programmi"; 3. I consumatori non hanno avuto la "possibilità di scegliere un sistema alternativo a Windows"; 4. Microsoft ha usato la sua posizione anche "nello sviluppo dei programmi in rete".

⁹⁵ Per una più dettagliata analisi cronologica dei fatti in oggetto si veda S. COSTA, *Antitrust e software: il caso Microsoft*, Carocci, Roma, 2004.

Le aziende concorrenti, infatti, protestarono vivamente e, negli anni successivi, Microsoft, per evitare altre vicende processuali sottoscrisse ulteriori accordi. Per esempio nel 2004 Microsoft e Sun Microsystems firmarono un accordo per appianare le loro controversie avviando una collaborazione volta a garantire una migliore integrazione tra i propri prodotti. Sun ricevette 700 milioni di dollari per la risoluzione delle controversie in materia di concorrenza e 900 milioni di dollari per quelle sui brevetti.

Ancora, nel 2003 fu avviata una *class action* da parte di un gruppo molto rilevante di consumatori che citò in giudizio Microsoft per abuso di posizione dominante⁹⁶. L'accusa fu ritirata solo dopo il pagamento di una notevole somma di denaro: 1 miliardo e 100 milioni di dollari.

La questione fu sollevata anche in sede europea. Nel 2004, dopo una indagine durata cinque anni, la Commissione Europea sanzionò Microsoft per abuso di posizione dominante⁹⁷. L'accusa era mirata e investì la circostanza che Microsoft, abbinando il programma Media Player a Windows, avrebbe deliberatamente annullato la concorrenza nel settore dei lettori multimediali.

La condanna irrorata consistette in una multa di 497 milioni di euro, che l'allora Commissario europeo Mario Monti definì come una efficace misura deterrente, idonea a delineare la posizione della Commissione *pro futuro*.

La decisione della Commissione, inoltre, obbligò Microsoft anche a trasmettere ai suoi concorrenti, entro 120 giorni, le "informazioni sulle interfacce richieste" in modo da permettere in modo agevole l'installazione delle applicazioni delle aziende concorrenti sui dispositivi con sistema operativo Windows. Infine, Microsoft fu obbligata a rilasciare sul mercato, una versione di Windows priva del lettore multimediale Media Player⁹⁸.

La decisione della Commissione, se raffrontata alle modalità in cui agì la procura statunitense, è sicuramente più coerente. Tale coerenza deriva dalla scelta della Commissione di circoscrivere la propria decisione ad un determinato mercato di software applicativi. La soluzione prospettata inizialmente dal dipartimento di giustizia americano era drastica perché voleva risolvere tutte le problematiche inerenti il mercato dei software applicativi con una unica decisione, ma tale scelta si è rilevata fallimentare.

Nel mercato odierno il rapporto tra software operativi e software applicativi è di fatto cambiato. La ragione risiede nello sviluppo di un nuovo modello

⁹⁶ L'azione fu promossa nel 1999 in California. Dopo tale accordo fu possibile alle imprese ed ai consumatori ricevere una indennità, con rimborso voucher, pari a 5 dollari per ogni copia di Microsoft Word acquistata, 16 dollari per ogni copia di Windows e 29 dollari per ogni copia Microsoft Office.

⁹⁷ V. Decisione della Commissione, del 24 maggio 2004, relativa ad un procedimento a norma dell'articolo 82 del trattato CE e dell'articolo 54 dell'accordo SEE contro Microsoft Corporation (Causa n. COMP/C-3/37.792 — Microsoft). In Rete: <http://eur-lex.europa.eu/legal-content/IT/TXT/?uri=CELEX%3A32007D0053>.

⁹⁸ Microsoft fece ricorso e fu aperto un caso (Microsoft Corp. contro Commissione delle Comunità europee, COMP/C-3/37.792) in rete: <http://eur-lex.europa.eu/legal-content/IT/TXT/?uri=CELEX:62004TJ0201>. Il ricorso fu fondamentalmente respinto eccetto nella parte in cui la decisione della Commissione obbligava alla "designazione di un mandatario indipendente dotato dei poteri di accedere, indipendentemente dalla Commissione, all'assistenza, alle informazioni, ai documenti, ai locali e ai dipendenti della Microsoft nonché al «codice sorgente» dei prodotti rilevanti della Microsoft". Per un commento critico al caso v. S. ANDERMAN, *Microsoft e il problema dell'interoperabilità*, in *Mercato, Concorrenza, Regole*, 3, 2007, 569.

commerciale che ha instaurato una sinergia tra produttori di sistemi operativi e applicazioni. Grazie infatti al nuovo, e più diffuso, metodo di distribuzione di software, che avviene attraverso il download della copia digitale, i produttori dei sistemi operativi hanno reso disponibile ai programmatori una piattaforma, volta alla commercializzazione dei loro prodotti⁹⁹. Unitamente a tale piattaforma si rendono disponibili anche gli strumenti di programmazione volti ad un corretto funzionamento del software sul sistema operativo¹⁰⁰, nonché servizi di marketing, come per esempio piani pubblicitari, volti ad una efficace sponsorizzazione del prodotto.

Il guadagno è reciproco. Il titolare della piattaforma di vendita, ad ogni copia venduta, trattiene parte del corrispettivo, e ha interesse nell'implementazione di quante più funzioni applicative nel proprio prodotto, in modo tale da attirare quanta più utenza possibile, sviandola così dalla concorrenza. I produttori di software applicativi, di contro, hanno molti più strumenti di programmazione, che consentono una più veloce realizzazione del progetto, e possono appoggiarsi ad una piattaforma di vendita sicura, controllata e della quale gli utenti si fidano. Questi ultimi infatti acquistano in sicurezza, con una scelta molto ampia, e a prezzi concorrenziali.

⁹⁹ Per citare i più famosi si pensi all'App Store, al Google Play Store, al Microsoft Store

¹⁰⁰ Ad esempio Apple, al fine di permettere lo sviluppo di applicazioni sul sistema operativo iOS, rende disponibile, una volta effettuata l'iscrizione all'Apple Developer Program, il kit SDK per iOS. L'iOS Developer Center offre strumenti di programmazione come codici di esempio e documentazione tecnica.

CAPITOLO II

Il software come oggetto della tutela

SOMMARIO: *Premessa* - 1. L'oggetto della tutela autoriale: il *computer program*. La dicotomia idea-espressione nel software - 2. L'oggetto della tutela brevettuale: Il *computer process* e le criticità del sistema statunitense - 3. Il software come oggetto di segreto industriale - 4. *La fictio iuris* come paradigma della tutela del software - 5. I contratti di licenza: software proprietari e software liberi - 6. Le misure tecnologiche di protezione, verso la chiusura del software? - 7. La tutela del software e la sua percezione

Premessa

Compresi quelli che sono i concetti di fondo che riguardano la natura del software, la sua funzione pratica, e la sua definizione tecnica, si procede ora con una analisi volta a comprendere su quale parte del software ricada in concreto la privativa brevettuale.

In questo capitolo si analizzerà in particolare il diverso oggetto su cui concretamente insistono, o dovrebbero insistere, le differenti protezioni di cui il software può beneficiare, in modo tale da poter individuare e delimitare con chiarezza l'ambito di applicazione della tutela mediante brevetto.

Attraverso questo ragionamento si cercherà inoltre di evidenziare la differenza sostanziale tra le differenti privative, che non si riducono nel dualismo brevetto - diritto d'autore. Infatti, il programma per elaboratore, stante la sua particolare natura, è passibile di essere tutelato anche attraverso altri strumenti inquadrabili nella categoria del *public enforcement* e del *private enforcement*.

Ciò detto è d'uopo iniziare il ragionamento prendendo in considerazione una domanda: è possibile ricreare il codice sorgente partendo dal codice oggetto? La risposta a tale quesito è positiva, ma, pur esistendo sistemi che decompilano il codice binario, il risultato di questo procedimento è un codice sorgente che differisce dall'originario pur essendone funzionalmente equivalente. Tale pratica prende il nome di ingegneria inversa. È, quindi, astrattamente possibile creare un codice sorgente "originale" sulla base di un codice oggetto copiato.

Da questa circostanza emerge, come si illustrerà *ultra*, la differenza tra una tutela attuata attraverso diritto d'autore, ed una mediante brevetto.

1. L'oggetto della tutela autoriale: il *computer program*. La dicotomia idea-espressione nel software

Attraverso il diritto d'autore viene protetta l'espressione dell'idea, qualora la stessa sia originale. La tutela non si estende tuttavia alle idee e i principi che sono alla base dell'opera¹⁰¹. Il legislatore europeo ha cristallizzato il principio all'interno della direttiva 2009/24/CE¹⁰².

Applicando detto principio al software, l'oggetto della tutela autoriale ricade sul codice, su quel determinato linguaggio, ossia sull'espressione dell'idea. Non si estende invece all'idea stessa, ossia alla funzione che si vuole eseguire. Riprodurre la medesima funzione, la medesima idea, attraverso un codice o un linguaggio diverso, quindi originale, non costituisce violazione del diritto d'autore. Potrebbe tuttavia sorgere il dubbio su quale codice, ossia su quale linguaggio, ricada la tutela. È limitata al solo codice sorgente?

Il legislatore europeo ha avuto cura di indicare come meritevoli di tutela anche i lavori preparatori alla realizzazione di un programma per elaboratore¹⁰³. Tali lavori sono per loro natura incompleti, sovente redatti in un linguaggio non

¹⁰¹ Il principio risale al caso *Backer v Selden*, 101 U.S. 99 (1879), ripreso successivamente in diverse pronunzie, e codificato nel Copyright Act americano (S. 17 U.S.C. 102 (b)): "*In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work.*". Il famoso caso scaturisce nel 1859, quando Charles Selden ottenne il copyright su un suo libro chiamato "*Selden's Condensed Ledger, or Bookkeeping Simplified*". Nel libro Selden descriveva un sistema di contabilità. A questo libro ne seguirono altri simili tutti coperti da copyright. I libri erano composti per la maggior parte di schemi e formulari e contenevano poche parole. L'operazione editoriale di Selden non ebbe successo, e Selden ebbe considerevoli perdite, stanti i debiti contratti per la stampa dei propri libri. Nel 1867, W.C.M. Baker mise in commercio un libro che descriveva un sistema di contabilità molto simile a quello illustrato da Selden, ed ebbe un grande successo editoriale, stante la maggior comprensibilità ed il minor costo. Dopo la morte di Selden la moglie, in qualità di erede, muove causa a Baker, con l'aiuto dei creditori. La decisione di primo grado del 1875 statuisce che il libro di Baker è per molte e sostanziali parti identico a quello di Selden e viola il copyright di quest'ultimo ed emana una *permanent injunction* ordinando a Baker di sospendere ogni pubblicazione e vendita così come ogni altro atto di disposizione del suo libro. La decisione della Corte Suprema di pochi anni dopo rovesciò tuttavia questa decisione. La tecnica in questione fu resa pubblica con un libro, e non protetta con un brevetto. Il libro è coperto da copyright, ma la tecnica in sé no (Selden fece richiesta per il patent, richiesta che venne respinta). Il copyright è valido al di là dalla novità dell'argomento, della tecnica o della materia contenuta, ma la sua tutela si limita alla espressione con cui quella tecnica è espressa. Per un approfondimento v. P. SAMUELSON, *Freedom of Ideas and of Competition, the story of Backer v Selden: Sharpening the distinction between authorship and innovation*, in *Intellectual property stories*, R. DREYFUSS e J. C. GINSBURG (a cura di), Foundation Press, New York, 157, 2006.

¹⁰² Direttiva 2009/24/CE, considerando n. 11: "*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. Conformemente alla legislazione e alla giurisprudenza degli Stati membri, nonché alle convenzioni internazionali sul diritto d'autore, l'espressione di tali idee e principi deve essere tutelata dal diritto d'autore.*"

¹⁰³ V. *supra*, cap. I, par. 4.

comprensibile alla macchina, e neppure in un linguaggio di programmazione “completo”.

Quindi la tutela non è necessariamente limitata al solo codice sorgente.

La circostanza si evince anche dalla definizione contenuta nel documento che accompagnava la prima direttiva europea sulla tutela del software, già analizzata¹⁰⁴. Non è richiesta la diretta eseguibilità sulla macchina, ma solo la possibilità che tale “prodotto” possa essere elaborato mediante una macchina, direttamente o indirettamente, ovvero dopo essere stato tradotto in un linguaggio eseguibile.

Nella concessione della tutela autoriale si prescinde quindi dalla forma utilizzata per la realizzazione del programma, sempre che questa forma sia comunque idonea ad essere successivamente implementata su di un computer. La tutela copre quindi ogni linguaggio, suscettibile di essere sviluppato su macchina, attraverso cui viene espressa una idea di programma per elaboratore.

Quest’ultimo ragionamento trova conferma alla luce sia della direttiva europea, sia della legge sul copyright statunitense¹⁰⁵, ma non alla lettera del documento OMPI. In questa circostanza il software è stato identificato come “*a set of instruction capable, when incorporated in a machine (...) to perform a particular function*”¹⁰⁶. Si presuppone quindi che il programma sia espresso in un linguaggio direttamente eseguibile dalla macchina. È, pertanto, esclusa la descrizione degli algoritmi attraverso altri linguaggi, strutture grafiche o diagrammi di flusso, ossia strumenti utili alla definizione dell’algoritmo alla base del linguaggio di programmazione, ma non comprensibili dalla macchina.

In questo senso si deve quindi considerare il concetto di “programma per elaboratore”, o, secondo l’accezione statunitense “*computer-program*”, all’interno del paradigma del diritto d’autore: un certo linguaggio, una modalità di espressione di una idea, idonea, anche non nell’immediatezza, a fornire istruzioni ad un computer.

Tornando al ragionamento di partenza, l’attività di decompilazione, alla luce di quanto appena detto sembrerebbe essere sempre illecita¹⁰⁷. Se infatti la tutela ricade su ogni forma di espressione dell’idea, compreso il codice oggetto, allora utilizzare il codice oggetto per ottenere un codice sorgente originale si deve considerare come attività vietata¹⁰⁸. Questa circostanza costituisce

¹⁰⁴ V. *supra*, cap. I, par. 4.

¹⁰⁵ V. 17 U.S.C. S. 101, dove viene rimarcata la circostanza che istruzioni possono essere implementare direttamente o indirettamente in un elaboratore elettronico.

¹⁰⁶ WIPO, *Model provision on the protection of computer software*, Ginevra, 1978, cit., 4.

¹⁰⁷ In tema di decompilazione v. C. REED, *Reverse Engineering Computer Programs without Infringing Copyright*, in 13 *E.I.P.R.* 47, 1991.

¹⁰⁸ Sulla necessità di una tutela forte del codice oggetto, stante la sua utilizzabilità al fine di creare un codice sorgente “originale”, v. DE SANTIS, *La tutela brevettuale del software tra brevetto e diritto d’autore*, pp. 69-71.

tuttavia una forte contraddizione al principio della dicotomia idea-espressione¹⁰⁹.

Le idee e i principi alla base dell'opera dovrebbero essere fruibili, ma di fatto non lo sono.

A livello legislativo, l'attività di decompilazione, è generalmente vietata, salvo una eccezione operante al verificarsi di diverse condizioni. La Direttiva 2009/24/CE, all'articolo 6, statuisce infatti che non è necessaria l'autorizzazione del titolare dei diritti per l'attività di decompilazione se *"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"*¹¹⁰. Per mettere in atto la pratica, tuttavia, il soggetto deve sottostare a stringenti condizioni. La procedura infatti deve essere messa in atto solo dal licenziatario e limitatamente al fine di acquisire le informazioni necessarie ad assicurare l'interoperabilità con un programma creato autonomamente. I risultati inoltre non possono essere comunicati a terzi e soprattutto, elemento più sconcertante della normativa, non possono essere *"utilizzate per lo sviluppo, la produzione o la commercializzazione di un programma sostanzialmente simile nella sua espressione, o per ogni altro atto che violi il diritto di autore"*¹¹¹.

Per comprendere tale assetto basti leggere il considerando n. 15 della Direttiva 2009/24/CE, dove si evidenzia come *"l'obiettivo di tale eccezione è di consentire l'interconnessione di tutti gli elementi di un sistema informatico, compresi quelli di fabbricanti differenti, perché possano funzionare insieme"*¹¹².

La norma è quindi rivolta agli operatori del settore ed è volta a garantire l'interoperabilità¹¹³. Nel mettere in atto tale necessaria salvaguardia, si tradisce tuttavia il principio della dicotomia idea-espressione, già evidenziato dal considerando n.11. Se infatti un soggetto non può utilizzare la decompilazione per giungere all'idea che sottende al software, al fine di utilizzare quell'idea per

¹⁰⁹ A tale proposito risulta molto interessante la teoria di Vittorio De Sanctis. L'autore sottolinea come nell'ordinamento italiano l'attività di decompilazione è generalmente vietata dalla legge per il disposto degli articoli 64 ter, e 64 quater della legge 633/1941. Allo stesso tempo è vietata la commercializzazione di software progettati al fine di eseguire la decompilazione, per il disposto dell'articolo 171 bis, della medesima normativa italiana. Posti tali divieti, di fatto viene contraddetto il principio secondo cui le idee alla base del programma sono disponibili a chiunque. V. V. DE SANCTIS, *A proposito della protezione del format*, in *Dir. Aut.*, 1, 2007, 58, p. 60.

¹¹⁰ Direttiva 2009/24/CE, art. 6.

¹¹¹ *Ibidem*.

¹¹² Direttiva 2009/24/CE, considerando n. 15.

¹¹³ Si deve rilevare come nel licenziare i propri software, le aziende generalmente vietano l'attività di decompilazione all'interno delle clausole dei contratti di licenza e, affinché il contratto sia rispettato implementano dei programmi delle misure tecnologiche di protezione volte ad inibire la pratica. Anche se i contratti prevedono che il licenziatario possa svolgere tale attività in conformità alla legge, si è già notato, parlando del caso Microsoft, come nella prassi tale circostanza non sia così semplice da realizzare. Si porta ad esempio un estratto della licenza di Windows 8.1: *"In particolare, la presente licenza non concede al licenziatario alcun diritto in merito a quanto segue e pertanto il licenziatario non potrà: (...) tentare di aggirare le precauzioni tecniche di protezione presenti nel software, decompilare né disassemblare il software, salvo nei casi in cui le leggi del Paese di residenza del licenziatario lo consentano, sebbene il contratto lo vieti. In tal caso, il licenziatario potrà effettuare unicamente quanto consentito dalla legge applicabile"*. V. *infra*, cap. il par. 5.

stimolare la propria vena creativa, allora il necessario contrappeso della piena fruibilità delle idee contro il monopolio sull'espressione delle stesse viene meno¹¹⁴.

Nel prevedere una eccezione così stringente alla pratica della decompilazione, in palese contrasto con il principio della dicotomia idea-espressione, si può desumere un elemento molto rilevante, ossia che la tutela autoriale nella sua "purezza" non sia sufficiente, o meglio competitiva, se applicata al software. La tutela è stata quindi estesa, ricomprendendo non solo ogni possibile forma di espressione, ma anche ampliando di fatto quelle che sono le attività riservate al titolare dei diritti¹¹⁵.

Nel diritto dell'Unione Europea, infatti, si è cristallizzata solo la tutela autoriale del programma per elaboratore. La tutela è economica e permette alle piccole realtà imprenditoriali di affermarsi nel settore proteggendo i propri trovati. Per questo motivo si è predisposta una tutela forte. Ciò non priva tuttavia le aziende, o perlomeno quelle che possono permetterselo, della possibilità di aggiungere al diritto d'autore anche una tutela brevettuale, in conformità con i parametri di EPO.

La contraddizione sopra esposta potrebbe derivare dalla circostanza che è astrattamente impossibile effettuare una scissione idea-espressione in un software¹¹⁶. Da un punto di vista meramente economico, infatti, un software dotato di valore si contraddistingue non certo per la sua espressività, ma per le sue funzioni. Ma anche da un punto di vista tecnico-fattuale *"Le forme espressive non sono altro che la specificazione nel linguaggio di programmazione della struttura e delle subroutine di un programma. Di conseguenza, ciò che viene generalmente chiamato forma espressiva del software, in realtà non è altro che un insieme di istruzioni sviluppate in maniera sequenziale per produrre un certo risultato in maniera costante."*¹¹⁷.

La modalità con cui un programmatore rappresenta un certo programma per elaboratore, deriva non di certo dal suo estro artistico, quanto piuttosto dalla necessità di rappresentare in un certo modo determinati algoritmi, alla base delle funzioni che si vogliono implementare nel software. Tanto è vero che è possibile scegliere tra differenti rappresentazioni, quanto è vero che tale scelta sarà fondata su ragioni tecnico-pratiche, piuttosto che su motivi meramente soggettivi-rappresentativi¹¹⁸.

¹¹⁴ Sul bilanciamento di questi interessi *"di rilievo costituzionale"* v. GHIDINI, *Profili evolutivi del diritto industriale*, pp. 59-71.

¹¹⁵ In questo senso V. FRANCESCHELLI, *La direttiva CEE sulla tutela del software: trionfo e snaturamento del diritto d'autore*, in *Rivista diritto industriale*, 1991, p. 169.

¹¹⁶ In questo senso si segnalano su tutti M. RICOLFI, *La tutela della proprietà intellettuale: fra incentivo all'innovazione e scambio ineguale*, in *Riv. Dir. Industriale*, I, 2002, 511, p. 516; e P. S. MENELL, *An Analysis of the Scope of Copyright Protection for Application Programs*, 41 *Stanford Law Review*, 1045, (1988), p. 1047, il quale nota come anche la commissione CONTU, nel 1978, avesse già espresso perplessità nel rinvenire una netta linea di confine tra espressione e idea nel programma per elaboratore.

¹¹⁷ E. AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, Giuffrè Editore, 2012, cit., 49.

¹¹⁸ Cfr. M. FABIANI, *Software: prodotto dell'ingegno in cerca di protezione*, in *Dir. Inf.*, 1989, pp. 561 ss. Secondo l'autore, il programmatore, in sede di scrittura del software, avrebbe invece ampie libertà di scelta relativamente al metodo con cui implementare le funzioni, la procedura e l'architettura del programma.

2. L'oggetto della tutela brevettuale: il *computer process* e le criticità del sistema statunitense

Se un software è protetto mediante brevetto, l'attività di decompilazione è sostanzialmente inutile, stante la non possibilità di "copiare" quelle idee, funzioni e algoritmi che sono alla base del programma¹¹⁹.

In questo caso la protezione è più penetrante da un punto di vista sostanziale. Infatti i singoli algoritmi che generano le istruzioni, se coperte da brevetto, non sono più utilizzabili, almeno a fini commerciali, da altri programmatori.

Sulla base di tali elementi si palesa la distinzione tra "*computer program*", ossia un programma per computer descritto in un determinato linguaggio, e coperto dal diritto d'autore, e "*computer process*", ossia il processo, l'insieme delle istruzioni implementate nel programma¹²⁰. Su quest'ultimo oggetto ricade la tutela brevettuale, qualora sottenda ai requisiti previsti dai legislatori, nonché ai criteri sviluppati dalle corti. Sulla base di tale distinzione negli Stati Uniti si delinearono le rispettive competenze del Patent Office e del Copyright Office nel momento in cui venne aperta la strada alla brevettabilità dei programmi per elaboratore¹²¹.

Il processo, la funzione, è un elemento che prescinde dalla modalità con cui la stessa è rappresentata. Esprime piuttosto il cuore del programma, i motivi per cui lo stesso è stato progettato. Ogni software è infatti creato per espletare determinate funzioni, ma ciò non significa che ogni programma in grado di svolgere determinate mansioni sia automaticamente brevettabile.

Per esserlo dovrà *in primis* sottostare ai requisiti previsti dai legislatori¹²², ed in particolare dovrà essere considerato come rientrante nel *patentable subject matter*, ossia quell'insieme di oggetti ritenuti meritevoli di ricevere protezione brevettuale. In tale ambito le corti ebbero, e hanno, un ruolo centrale, interpretando in maniera estensiva il concetto di oggetto dell'invenzione, onde ricomprendervi quelle attuate a mezzo di software¹²³.

Ma non solo; il software per accedere alla privativa dovrà essere "nuovo" e "non ovvio". Se questi "baluardi" venissero sottovalutati nell'analisi della rivendicazione, allora si rischierebbe di andare nella direzione di un

¹¹⁹ "Inutile" ma non esplicitamente vietata da alcuna normativa brevettuale. Va da se tuttavia che sussistendo il divieto, in quanto il programma è coperto anche dal copyright, tale pratica sarà comunque irrealizzabile, almeno stando alla legge dell'Unione Europea.

¹²⁰ Nelle invenzioni di software, il *process* non è solamente il procedimento industriale predisposto per la fabbricazione del prodotto finale, oggetto ultimo della tutela. Costituisce infatti allo stesso tempo anche l'oggetto della tutela: "*il programma per elaboratore è allo stesso tempo sia process che product*", E. AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, Giuffrè Editore, Collana Univ. Milano - Fac. giur. Studi Dir. Privato, 2012, cit., 266-267.

¹²¹ In questo senso G. GUGLIELMETTI, *L'invenzione di software. Brevetto e diritto d'autore*, Giuffrè Editore, Milano, 1997, p. 26; C. GATTEI, *Il brevetto di software, guida teorico pratica*, LaTribuna, Piacenza, 2003, pp. 27-31.

¹²² V. *infra*, cap. III.

¹²³ V. *infra*, cap. IV.

indiscriminato riconoscimento della tutela brevettuale. Circostanza questa evidenziata da molti studiosi con riguardo al sistema brevettuale americano¹²⁴.

Ne è un esempio pratico la rivendicazione di Amazon concernente il *“Method and system for placing a purchase order via a communications network”*, più comunemente noto come sistema *“one-click-buy”*. L'invenzione in questione utilizza i cookies per facilitare l'identificazione dell'utente in modo tale da poter rendere immediate le operazioni di pagamento.

La decisione della Board of Appeal di EPO del 2009¹²⁵ chiude una *querelle* durata circa un decennio, stanti i continui rifiuti delle Divisioni di EPO a riconoscere il brevetto ad Amazon, e l'ostinazione della società alla presentazione di domande di registrazione. Il fondamento della decisione risiede nella circostanza che l'invenzione in questione è ovvia. Infatti, secondo la decisione di EPO, Amazon utilizza elementi già presenti allo stato della tecnica, che una persona mediamente competente nella materia avrebbe potuto realizzare senza particolari difficoltà inventive.¹²⁶

Se in Europa il trovato è considerato come “ovvio”, negli Stati Uniti, l'invenzione è stata invece brevettata¹²⁷. Analizzando il quadro legislativo si evincerà come a livello internazionale ci sia convergenza attorno ai criteri di novità ed ovvietà¹²⁸. Tuttavia, in questo caso, è palese come essi siano stati interpretati in maniera diversa.

La differenza di trattamento della invenzione potrebbe non essere causata tanto da una distanza in termini legislativi, quanto piuttosto da una differente organizzazione dei rispettivi Uffici Brevetti.

Negli Stati Uniti l'Ufficio brevetti nel 1982 fu trasformato in agenzia di servizi. A seguito di tale trasformazione, i costi necessari al suo mantenimento non vennero più erogati dal governo federale, ma fu previsto un sistema di autofinanziamento. In questo modo la struttura si regge attraverso le *fees dei patent applicants*¹²⁹.

Alcuni autori individuano quindi più nei bilanci del Patent Office, che nelle sue decisioni, la causa delle difficoltà del sistema brevettuale statunitense,

¹²⁴ V. DAN, L. BURK, M. A. LEMLEY, *The Patent Crisis and How the Courts Can Solve It*, University Of Chicago Press, Ltd., London, 2010; A. B. JAFFE, J. LERNER, *Innovation and Its Discontents: How Our Broken Patent System is Endangering Innovation and Progress, and What to Do About It*, Princeton University Press, New Jersey, 2004; J. FELLAS, *The Patentability of Software-Related Inventions in the United States*, in *E.I.P.R.*, 21(6), 1999, 330.

¹²⁵ Decision of the Technical Board of Appeal, Gift order/AMAZON, T 1616/08, 11.11.2009, in Rete <http://www.epo.org/law-practice/case-law-appeals/pdf/t081616eu1.pdf>.

¹²⁶ *Ibidem*, p. 11: “The mere wish to automate process steps that have previously been performed manually is usually regarded as obvious. The automation details may naturally be inventive, but in the present case the problem of how to extract the delivery information is left entirely to the skilled person. Thus an inventive step is neither involved in the idea to extract information automatically, nor in its implementation. The subject-matter of claim 1 is therefore obvious”.

¹²⁷ US 5960411 A, “Method and system for placing a purchase order via a communications network”, in Rete: <http://www.google.com/patents/US5960411>.

¹²⁸ Su tutti v. The Agreement on Trade Related Aspects of Intellectual Property Rights (TRIPs), 1995, art. 27; Cfr. 35 U.S. Code § 102, § 103 e EPC art. 54 e art. 56. V. *infra*, cap. III, par. 3.

¹²⁹ V. JAFFE, LERNER, *Innovation and Its Discontents: How Our Broken Patent System is Endangering Innovation and Progress, and What to Do About It*. Gli autori sottolineano come tra il 1982 e il 2002, la concessione di brevetti sia aumentata al ritmo medio del 5,7% l'anno, contro l'1% medio annuo del periodo 1930-1982. Da tali circostanze giungono alla conclusione che l'esplosione dell'attività brevettuale sia da riferire a circostanze di carattere economico.

contraddistinto da una crescita esponenziale nel numero dei contenziosi giudiziari, da una sostanziale perdita di rigore nelle procedure di valutazione delle domande, nonché da un aumento dei costi di transazione per l'acquisto e la cessione di licenze sui brevetti¹³⁰.

La proliferazione di brevetti di scarsa o nessuna rilevanza tecnologica e i costi sempre più elevati di difesa dei brevetti in sede giudiziale, possono portare al fenomeno del c.d. *patent trolling*¹³¹.

Negli Stati Uniti, si sono da tempo sviluppate moltissime società il cui metodo business consiste nella compra-vendita di brevetti¹³². Tale prassi è sicuramente lecita, e non va per l'appunto confusa con il fenomeno dei *troll* o anche detti PAE (*Patent Assertion Entity*).

Tale pratica predatoria consiste nel richiedere royalties per l'utilizzo di brevetti, legittimamente registrati, ma di dubbia rilevanza allo stato dell'arte, ad aziende che li stavano inconsapevolmente utilizzando senza il consenso del titolare dei diritti. Stanti gli alti costi della giustizia, chi viene chiamato in causa generalmente paga, oppure viene ad accordi traslativi con il titolare dei diritti¹³³. Il fenomeno è anche stato descritto alla stregua della "semplice estorsione"¹³⁴.

Il problema è particolarmente presente nell'industria di software¹³⁵. Questo perché il mercato ICT ha basse barriere di entrata e l'ottenimento di brevetti, di sospetta qualità, nel campo informatico, viene considerato, dagli operatori del settore, relativamente semplice.

3. Il software come oggetto di segreto industriale

Come meglio si analizzerà nel corso del prossimo capitolo, la prassi, sia di EPO, che dell'USPTO, non obbliga all'allegazione del codice sorgente del programma di cui si richiede la brevettabilità. Si ritiene sufficiente, per

¹³⁰ V. *supra*, nota n. 124.

¹³¹ V. W. J. WATKINS JR., W. F. SHUGHART II, *Patent Trolls: Predatory Litigation and the Smothering of Innovation*, The Independent Institute, Oakland, 2014.

¹³² Nel gergo commerciale statunitense PHC, ossia Patent Holding Company.

¹³³ Il fenomeno è così descritto sulla rivista Economist, N.V., *Obituary for software patents*, 14/12/2013: "Hiding behind shell companies, trolls prey mostly on small firms, sending out thousands of 'demand letters' at a time. Over the past couple of years, the practice has reached epidemic proportions, as trolls have broadened their targets to include not just high-tech firms, but also universities, retailers, hospitals, charities and even consumers themselves. With their lawyers working on a contingency basis (ie, no award, no fee), trolls have low upfront costs and little to lose. Most of their law-suits are filed in eastern Texas, where the courts have a reputation for being troll-friendly". In Rete: <http://www.economist.com/blogs/babbage/2013/12/difference-engine-0>.

¹³⁴ In questo senso K. OSENGA, *Debugging Software's Schemas*, in *Intellectual property Law Review*, 2015, Edited by Karen B. Tripp, Thomson Reuters, 2015, 203-228.

¹³⁵ M. A. LEMLEY, *Software patents and the return of functional claiming*, Stanford Public Law Working Paper No. 2117302, 2012.

soddisfare la *disclosure*, una dettagliata descrizione, tale da consentire, ad una persona esperta della materia, l'attuazione dell'invenzione¹³⁶.

Men che meno tale onere di pubblicazione sussiste nel momento in cui viene applicata una tutela mediante il diritto d'autore.

Ciò implica che le stringhe dei codici, le informazioni più importanti di un programma, che esprimono l'invenzione nella sua tecnicità più pura, non sono rese note alla comunità scientifica. *“Accade l'esatto contrario di ciò che sarebbe legittimo aspettarsi: il codice sorgente, l'unico plausibilmente proteggibile, rimane segreto e ciò è reso possibile proprio grazie alla protezione del codice oggetto, la cui tutela deriva, a sua volta, da quella accordata al sorgente”*¹³⁷.

Dette informazioni, sono di fatto tenute segrete, e, secondo la disciplina odierna¹³⁸, sono destinate a rimanere tali. Affrontando il tema della decompilazione, si è visto come il margine di azione nell'attuazione di tale pratica sia molto limitato, e quindi limitante per il progresso tecnologico¹³⁹.

È bene ribadire come tale fattore sia sicuramente molto preoccupante.

Quando un prodotto tradizionale entra in commercio, il legittimo acquirente può usarlo in qualsiasi modo preferisce, fino anche a smontarlo per comprenderne i principi di funzionamento. Per comprendere i margini di liceità di tale pratica, detta “ingegneria inversa”, bisogna tuttavia distinguere a seconda che tale operazione sia effettuata per fini commerciali o meno.

¹³⁶ V. Guidelines for Examination in the EPO, Part F – Chapter II, November 2016, 4.12, Computer programs: *“In the particular case of inventions in the computer field, program listings in programming languages cannot be relied on as the sole disclosure of the invention. The description, as in other technical fields, should be written substantially in normal language, possibly accompanied by flow diagrams or other aids to understanding, so that the invention may be understood by a person skilled in the art who is deemed not to be a specialist in any specific programming language, but does have general programming skills. Short excerpts from programs written in commonly used programming languages can be accepted if they serve to illustrate an embodiment of the invention”*, v. *infra*, cap. III, par. 2.4 e 5.3.

¹³⁷ AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 56, che desume, da tale circostanza, una ulteriore causa di inidoneità del paradigma autoriale alla tutela del software; se infatti è superfluo per le tradizionali opere dell'ingegno la previsione di un onere di pubblicazione, stante il fatto che al momento della commercializzazione dell'oggetto, i contenuti dell'opera diventano fruibili, tale onere sarebbe invece estremamente necessario per i programmi, stante la loro particolare natura.

¹³⁸ In particolare si fa riferimento alla Direttiva 2009/24/CE, art. 6.

¹³⁹ Si segnala, in tema di decompilazione, una interessante sentenza della Corte di Giustizia dell'Unione Europea. La sentenza del 2 maggio 2012, Causa C-406/10, SAS Institute Inc, contro World Programming Ltd, chiarisce in sede giurisdizionale i concetti che si sono già richiamati in precedenza. La Corte rimarca come *“non costituiscono una forma di espressione di un programma per elaboratore e non sono, a tale titolo, tutelati dal diritto d'autore sui programmi per elaboratore ai sensi della predetta direttiva né la funzionalità di un programma siffatto né il linguaggio di programmazione e il formato di file di dati utilizzati nell'ambito di un tale programma per sfruttare talune delle sue funzioni”*. Se ne deduce che *“colui che ha ottenuto su licenza una copia di un programma per elaboratore può, senza l'autorizzazione del titolare del diritto d'autore, osservare, studiare o sperimentare il funzionamento di detto programma al fine di determinare le idee e i principi su cui si basa ogni elemento di tale programma”*. Tuttavia, tale atteggiamento permissivo è confermato *“allorché egli effettua operazioni coperte da tale licenza nonché operazioni di caricamento e svolgimento necessarie all'utilizzazione del programma e a condizione che non leda i diritti esclusivi del titolare del diritto d'autore sul programma di cui trattasi”*. Sebbene a livello teorico si affermi la dicotomia idea/espressione, nella prassi le limitazioni alla decompilazione sono tali da rendere impossibile una legittima fruizione delle idee alla base dell'opera. In Rete: http://eur-lex.europa.eu/legal-content/it/TXT/PDF/?uri=uriserv%3AOJ.C_.2012.174.01.0005.01.ITA.

Nel secondo caso, *nulla questio*, non palesandosi alcun danno per chi beneficia del segreto. Nel primo caso, invece, è bene fare riferimento, per una piena comprensione del quadro legislativo, alla normativa internazionale, ed in particolare all'Accordo TRIPs, che fornisce agli Stati aderenti una forma di tutela comune.

Le informazioni tenute segrete non possono essere rivelate, acquisite o utilizzate *“in un modo contrario a leali pratiche commerciali”*¹⁴⁰ e tale circostanza non accade qualora dette informazioni siano: note o facilmente accessibili; dotate di valore commerciale a prescindere dalla loro segretezza; non adeguatamente protette.¹⁴¹

Questa pratica, tuttavia, diviene in ogni caso vietata dalla legge qualora applicata al software, ed i parametri sopracitati diventano di fatto irrilevanti¹⁴².

Ma non solo; anche una eventuale applicazione della lettera della disciplina sarebbe paradossale perché concederebbe una più lata tutela a quei programmi la cui decompilazione risulterebbe più difficoltosa.¹⁴³

È d'uopo notare come la suddetta pratica non sia a priori, ed in ogni caso vietata qualora i prodotti in questione siano coperti da brevetto¹⁴⁴. Anche in quest'ultimo caso l'utente può analizzare e studiare il prodotto a fini non commerciali. Non solo; se l'attività di analisi e di sperimentazione (*Experimental Use*) porta a nuove acquisizioni tecniche, allora queste potranno confluire in nuovi brevetti, che, saranno dipendenti dal brevetto oggetto di studio, e, qualora utilizzati in ambito commerciale, dovranno sottostare all'autorizzazione del titolare del primo brevetto, nonché al pagamento di determinate royalties.

L'innovazione può quindi in concreto generare innovazione.

Il risultato è paradossale perché, *“si dà spazio ad una forma di segreto industriale che da una parte è impropria, in quanto cumulativa e non alternativa al diritto d'autore e, dall'altra, è eccessiva, nel senso che consente una protezione ben maggiore di quella generalmente garantita dal tradizionale segreto industriale”*¹⁴⁵.

¹⁴⁰ Accordo TRIPs, art. 39.

¹⁴¹ Accordo TRIPs, art. 39: *“a) siano segrete nel senso che non sono, nel loro insieme o nella precisa configurazione e combinazione dei loro elementi, generalmente note o facilmente accessibili a persone che normalmente si occupano del tipo di informazioni in questione; b) abbiano valore commerciale in quanto segrete; e c) siano state sottoposte, da parte della persona al cui legittimo controllo sono soggette, a misure adeguate nel caso in questione intesa a mantenerle segrete.”*

¹⁴² Per una disamina generale sul segreto industriale v. A. VANZETTI, V. DI CATALDO, *Manuale di diritto industriale*, Giuffrè Editore, Milano, 2012, pp. 499-502; in particolare sui rapporti tra segreto, brevetto e altri diritti di proprietà intellettuale, v. G. GUGLIELMETTI, *La tutela del segreto*, in C. GALLI, *Le nuove frontiere del diritto dei brevetti*, Giappichelli, Torino, 2003 pp. 109-131.

¹⁴³ Circostanza che può verificarsi nel caso in cui il titolare del software abbia deciso di implementare delle misure tecnologiche di protezione. V. *infra*, cap. II, par. 6.

¹⁴⁴ Sul tema v. E. AREZZO, G. GHIDINI, *Biotechnology and software patent law, a comparative review of new developments*, Collana New Directions in Patent Law, Edward Elgar Publishing, Cheltenham, 2011. In particolare V. DI CATALDO, *The experimental use of the patented invention: a free Use or an infringing use?* pp. 91 ss.; NARD, *The Law of Patents*, pp. 732-739.

¹⁴⁵ AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 62-63.

4. La *fictio iuris* come paradigma della tutela del software

Il software è stato da sempre protetto paragonandolo ed accostandolo ad altre entità, invero molto diverse dalla tipologia del bene in questione.

Secondo la legge dell'Unione Europea, il software è tutelato con il diritto d'autore attraverso una finzione volta a considerarlo come un'opera letteraria¹⁴⁶. *“L'assimilazione del software ad un'opera dell'ingegno è consentita tramite il ricorso alla c.d fictio iuris che paragona i listati di un programma alla prosa di un romanzo”*¹⁴⁷.

Come si è precedentemente analizzato, il legislatore europeo concede una tutela autoriale ai programmi per elaboratore espressi in qualsiasi forma. Se si può ritenere coerente l'accostamento ad una opera letteraria di un programma scritto in codice sorgente, comprensibile per l'uomo, e frutto della sua creazione intellettuale, altrettanto non può dirsi per tutte le altre forme di scrittura del programma come ad esempio il codice oggetto¹⁴⁸.

La finzione viene quindi sfruttata per estendere la tutela anche alle altre forme di linguaggio di programmazione¹⁴⁹. Ma non solo; come si è visto in precedenza la finzione viene sfruttata sino a trasformare il diritto d'autore in una privativa *“simil-brevettuale”*¹⁵⁰, volta a proteggere quegli aspetti funzionali del programma che resterebbero di fatto non tutelati se fosse rigidamente applicata la disciplina del diritto d'autore.

La pratica di tutelare il software alla stregua di una entità che di fatto non è, non si limita alla applicazione del paradigma autoriale.

Grossomodo fino agli anni ottanta, in Inghilterra e negli Stati Uniti, l'unico modo per dare al software una tutela mediante brevetto era quello di considerarlo come una macchina, ossia come un sistema integrato all'hardware¹⁵¹. Questa era l'unica strada percorribile per assicurargli una tutela

¹⁴⁶ Direttiva 2009/24, art. 1, comma 1: *“Conformemente alle disposizioni della presente direttiva, gli Stati membri tutelano i programmi per elaboratore, mediante diritto d'autore, come opere letterarie ai sensi della convenzione di Berna sulla tutela delle opere letterarie e artistiche. Ai fini della presente direttiva, il termine «programma per elaboratore» comprende il materiale preparatorio per la progettazione di un programma”*.

¹⁴⁷ AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 46.

¹⁴⁸ Circostanza questa tutt'altro che pacifica; infatti è sovente, nella stesura dei programmi, il ricorso ad apparecchiature informatiche. In particolare vengono utilizzati altri software, progettati con il proficuo scopo di convertire algoritmi molto complessi, che costituiscono le istruzioni di base, in linguaggio di programmazione. Tale pratica consente ai programmatori un considerevole risparmio di tempo ed una diminuzione del margine di errore durante la programmazione, ma, per converso, tale traduzione è difficile da considerare come creazione artistico-intellettuale del programmatore. M. A. LEMLEY, P. S. MENELL, R. P. MERGES, P. SAMUELSON, B. W. CARVER, *Software and Internet Law*, Wolters Kluwer, Aspen Publishers; 4th Revised ed., New York, 2011. 1, pp. 25 ss.

¹⁴⁹ In questo senso GUGLIEMMETTI, *L'invenzione di software. Brevetto e diritto d'autore*, il quale conclude che la tutela è efficace solo se estesa anche al codice oggetto, stante tuttavia l'impossibilità di considerare tale linguaggio alla stregua di una opera letteraria.

¹⁵⁰ In questi termini AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 67.

¹⁵¹ Si veda su tutti P. LEITH, *Patenting programs as machines*, Volume 4, Issue 2, SCRIPT-ed, 2007.

brevettuale, stante le prime teorie dottrinali che, considerando il software alla stregua di una formula matematica, lo consideravano non brevettabile¹⁵².

Successivamente la giurisprudenza americana cambiò rotta, concedendo una protezione brevettuale ai programmi per elaboratore. Il punto è tuttavia delicato perché le basi su cui poggiavano le decisioni erano invero differenti. Un primo filone giurisprudenziale accosta il software ad un procedimento tecnico¹⁵³. Un secondo filone fa leva sulla tesi della materialità del software, dal momento che, quando lo stesso viene implementato su di una macchina, la stessa acquisisce la possibilità di svolgere nuove funzioni¹⁵⁴. Infine si statui l'idoneità di una formula matematica ad essere brevettata se la stessa è idonea a produrre un risultato utile, concreto e tangibile¹⁵⁵.

In Europa, invece, gli stati nazionali, attenendosi alla normativa della Convenzione Europea sul Brevetto, entrata in vigore nel 1977, interpretarono la propria normativa in modo tale da rendere brevettabili non i software in quanto tali, ma solamente le invenzioni attuate a mezzo di software.

Questa sottile linea di demarcazione delinea la situazione che sussiste oggi in tutta l'Unione Europea, dove, ai sensi di quanto previsto dalla EPC e dalle direttive UE, un software può essere brevettato solo in determinati casi, scandagliati dalla giurisprudenza dell'EPO, mentre in tutti gli altri casi sarà soggetto comunque ad una tutela autoriale.

Che sia tutelato alla stregua di un'opera letteraria, di una macchina, di un processo industriale, o di una formula matematica, il software non è mai stato protetto attraverso l'implementazione di una privativa esclusivamente ispirata alle sue particolari caratteristiche. Di contro, l'adattamento al software di normative "neutrali" ha permesso alle corti di avere un elevato grado elasticità nell'impostazione della materia, permettendo al diritto di "rimanere al passo" con l'inevitabile evoluzione dello stato della tecnica¹⁵⁶.

La predisposizione di strumenti classici di tutela della proprietà intellettuale, se può dirsi efficace per prevenire o bloccare fenomeni di furto della proprietà intellettuale da parte della concorrenza, può rivelarsi non idonea quando le violazioni sono messe in atto da parte degli utenti finali.

Quando la tutela dell'autorità pubblica non è sufficiente, sono gli stessi autori ad autotutelarsi, e ciò avviene attraverso la predisposizione di contratti di licenza, nella implementazione di misure tecnologiche di protezione, nonché per mezzo di strategie commerciali, volte a dissuadere l'utente medio dall'utilizzo di una copia contraffatta.

5. I contratti di licenza: software proprietari e software liberi

Il primo metodo con il quale i titolari dei diritti proteggono il software è attraverso uno strumento di natura contrattuale: i contratti di licenza.

¹⁵² V. *infra*, cap. IV, par. 1.2.

¹⁵³ Diamond contro Diehr, 450 U.S 175 (1981). V. *infra*, cap. IV, par. 1.3.

¹⁵⁴ In re Alappat, 33 F.3d 1526 (Fed. Circ. 1994). V. *infra*, cap. IV, par. 1.4.

¹⁵⁵ State Street Bank & Trust Co. v. Signature Financial Group Inc, 149 F.3d 1368 (1998). V. *infra*, cap. IV, par. 1.5.

¹⁵⁶ V. *infra*, riflessioni conclusive, par. 3.

La licenza è il permesso, l'autorizzazione di utilizzare il software nei termini e nei limiti specificati nel contratto. Chi acquista un software non ne diventa il proprietario, proprio perché sta acquisendo il diritto all'utilizzo dello stesso, e non la proprietà. Il proprietario è e rimane il licenziante¹⁵⁷.

I contratti in questione sono individuali, e per lo più scritti e sottoscritti attraverso modelli standard. Si sta parlando di licenze non esclusive che si applicano a una vasta gamma di utenti.

Per potere utilizzare un software, nella maggior parte dei casi, si dovrà accettare un *license agreement*. Questo è un vero e proprio contratto la cui sottoscrizione è condizione necessaria per l'utilizzo del programma a cui accede, e, nonostante esso consti di pagine e pagine di contenuto, è una constatazione meramente empirica rilevare come l'utente accetti le clausole senza nemmeno leggerle.

I contratti di licenza offrono, ai titolari dei diritti di distribuzione del software, una forte prerogativa di controllo sul prodotto. Infatti, se il trasferimento del programma per elaboratore non è configurato come una vendita, allora non si perfeziona l'esaurimento del diritto di distribuzione.

Alle licenze d'uso, infatti, non si applica la *first sale doctrine*¹⁵⁸. In questo modo l'utenza non può creare mercati secondari. Il titolare dei diritti può quindi segmentare il mercato attuando una *discriminazione dei prezzi*¹⁵⁹. È il titolare che controlla la distribuzione del software anche dopo la prima "vendita"¹⁶⁰.

Il principio dell'esaurimento del diritto è fondamentale al fine di avere una tutela bilanciata del copyright. Attraverso di esso, infatti, si ha una maggiore circolazione dell'innovazione e della cultura. L'assenza di tale principio nell'ambito del software è sicuramente un punto importante da rilevare.

Il principio impedisce al titolare dei diritti di controllare l'ulteriore distribuzione della copia dopo la prima vendita. Il compratore può quindi rivendere la copia a un prezzo più basso di quello di mercato. In questo modo si ha una maggiore espansione della cultura perché l'opera arriva dove, per ragioni economiche, non potrebbe arrivare.

¹⁵⁷ Per un approfondimento del tema si veda E. PROSPERETTI, *La circolazione dell'opera digitale*, Giappichelli, Torino, 2012. In questo paragrafo si analizzeranno i contratti di licenza limitatamente alla fattispecie *business to consumer*.

¹⁵⁸ La dottrina della prima vendita, incorpora il principio di esaurimento del diritto d'autore. Il titolare del copyright può controllare solo la prima vendita, che esaurisce il diritto vantato dall'autore su quella singola copia. Si permette all'acquirente di trasferire successivamente un prodotto soggetto a proprietà intellettuale, senza necessità di ottenere il consenso del titolare. Tale principio è una limitazione, assieme al *fair use* (17 U.S.C. § 107. *Limitations on exclusive rights: Fair use*), del diritto di proprietà intellettuale, che tradizionalmente consentono un equo bilanciamento tra i diritti dei titolari ed il pubblico interesse alla diffusione della cultura. Per un approfondimento v. G. COLANGELO, *Diritto comparato della proprietà intellettuale*, Il Mulino, Bologna, 2011, in particolare cap. I, par. 4.3.

¹⁵⁹ La discriminazione di prezzo è "*la strategia di prezzo di una impresa che per lo stesso bene applica prezzi diversi a diverse categorie di consumatori*", KRUGMAN, WELLS, *Microeconomia*, cit., 326.

¹⁶⁰ Circostanza ben riassunta nella Direttiva 2009/24/CE, articolo 4, comma 2: "*La prima vendita della copia di un programma nella Comunità da parte del titolare del diritto o con il suo consenso esaurisce il diritto di distribuzione della copia all'interno della Comunità, ad eccezione del diritto di controllare l'ulteriore locazione del programma o di una copia dello stesso*".

Il principio non si applica ai software perché essi non sono venduti, ma concessi in licenza d'uso. È importante rilevare come non influisca il fatto che il software sia o meno incorporato su di un supporto tangibile. Se c'è vendita il diritto si esaurisce, se c'è licenziamento no¹⁶¹. L'esaurimento del software può quindi avvenire anche tramite trasferimento *on-line* ma tale trasferimento deve essere qualificato come vendita¹⁶².

Lo strumento della licenza è rilevante anche per un altro motivo; le licenze c.d. *shrinkwrap* o *clickwrap*, “spesso impongono al consumatore, legittimo acquirente condizioni svantaggiose ovvero la rinuncia a prerogative riconosciute dalla legge”¹⁶³. Nonostante la dottrina prevalente ritenga che tali limitazioni, essendo in palese contrasto con la legge, siano invalide, è già di per sé rilevante notare come lo strumento contrattuale comprima, o provi a comprimere, le attività dell'utilizzatore¹⁶⁴.

I contratti di licenza ricoprono quindi un importante ruolo nella tutela dei software, nonché della loro circolazione. A seconda della tipologia di licenza rilasciata si innesta la differenza tra software proprietari e software liberi o open-source¹⁶⁵.

Il software proprietario, detto anche privato, non libero, o closed source, è licenziato attraverso contratti, le cui condizioni impediscono determinati atti come la modifica, la ridistribuzione, la condivisione, lo studio, nonché le azioni di ingegneria inversa. Essendo il software pubblicato soltanto in codice binario, il codice sorgente non viene mostrato all'acquirente del prodotto. Tale pratica rende lo studio e la modifica del programma per elaboratore tecnicamente molto difficile, essendo necessarie specifiche competenze tecniche per risalire al codice sorgente, attività questa che è tuttavia interdetta al licenziatario.

¹⁶¹ Come si evidenzia nella Direttiva 2001/29/CE, articolo 3, comma 3: “I diritti di cui ai paragrafi 1 e 2 non si esauriscono con alcun atto di comunicazione al pubblico o con la loro messa a disposizione del pubblico, come indicato nel presente articolo”.

¹⁶² Causa C-128/11, 03.07.2012 (UsedSoft GmbH contro Oracle International Corp.): tale sentenza della Corte di Giustizia UE è importantissima perché chiarisce le problematiche che si innestano sul diritto di esaurimento applicato al bene software nell'Unione Europea. Secondo i giudici il diritto di distribuzione di cui all'articolo 4, paragrafo 2, della direttiva 2009/24 non riguarda unicamente beni tangibili ma anche copie immateriali di software scaricate via Internet. La vendita di un programma per elaboratore mediante supporto tangibile e mediante download da Internet sono quindi analoghe. Nel caso di specie fu qualificata alla stregua di una vendita il licenziamento di un software senza limiti di durata, a fronte di un pagamento di un prezzo. Sussistendo quindi una vendita e non un licenziamento, il diritto di distribuzione era quindi soggetto ad esaurimento. Per un approfondimento v. A. GERACI, *L'applicazione del principio di esaurimento nei contratti di licenza d'uso del "software"*, in *Il Diritto industriale*, 2015, 5, pp. 447-453.

¹⁶³ AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 11.

¹⁶⁴ V. M. RICOLFI, *Software e limitazioni del licenziatario*, in *AIDA* 2004, 358; M. A. LEMLEY, *Intellectual property and shrinkwrap Licenses*, in 68 *S. Cal. L. Rev.*, 1239, (1995).

¹⁶⁵ I termini, in questa parte della ricerca possono essere usati come sinonimi ma invero, esistono differenze tra un software libero ed uno open-source. La disponibilità del codice sorgente è il requisito fondamentale che accomuna il software libero ed il software open source. Tale condizione è necessaria e sufficiente perché un software sia definibile come open source. È invece necessaria ma non sufficiente affinché un software sia libero. Per dirsi libero, deve essere possibile l'esercizio delle 4 libertà fondamentali, ideate dalla FSF. Quest'ultima ne evidenzia le differenze sul suo sito web: <https://www.gnu.org/philosophy/free-software-for-freedom.it.html>.

I software liberi sono invece rilasciati con una licenza molto permissiva¹⁶⁶. La licenza GNU/GPL ne è l'esempio più famoso nonché di successo¹⁶⁷. È pensata per consentire la condivisione e la modifica del software. Ne esistono diverse versioni, ma lo snodo principale è che il codice sorgente deve essere trasmesso a tutti coloro che utilizzano il programma.

L'idea nasce negli Stati Uniti ad opera di un informatico, Stallman, che nel 1985 fondò la Free Software Foundation, organizzazione no-profit volta alla promozione e allo sviluppo del software libero. La prima versione di tale licenza fu rilasciata nel 1989 nell'ambito del progetto GNU, un sistema operativo.

Lo strumento della licenza viene rovesciato ed utilizzato per permettere determinate attività quali l'accesso, la riproduzione, la modifica e la distribuzione¹⁶⁸. Questo utilizzo dello strumento contrattuale è efficace, perché, se fosse stata scelta l'alternativa del pubblico dominio, ossia una pubblicazione incondizionata del codice sorgente, si sarebbe tornati al punto di partenza, perché il miglioramento dello stesso da parte di terzi ne avrebbe legittimato la sua tutela¹⁶⁹.

Tali licenze introducono un vero e proprio modello di business nuovo. La continua rielaborazione dell'esistente porta a un costante e veloce miglioramento del software. Le barriere in entrata sono molto basse essendo il lavoro iniziale condiviso, e ciò si traduce in un vantaggio concorrenziale. Il guadagno per chi investe sulla elaborazione di software liberi c'è, ed è rappresentato, per esempio, dai servizi aggiuntivi non essenziali.

La differenza fondamentale tra software proprietari e software liberi viene descritta con successo in *La cattedrale e il bazaar*, un saggio di Eric S. Raymond del 1997. Esso descrive un nuovo modello di sviluppo, paragonato appunto ad un bazaar, il cui esempio più famoso ed efficace, per il tempo in cui il saggio fu realizzato, fu la modalità di costruzione del kernel Linux¹⁷⁰.

Il modello a cattedrale, tipico delle aziende commerciali, è invece basato sulla realizzazione del software ad opera di un limitato gruppo di esperti, il cui lavoro viene ordinato e coordinato. Il modello a bazaar non è gerarchico, e lo

¹⁶⁶ Per un approfondimento sul tema v. M. DA BORMIDA, D. DOMENICI, *Software libero, copyleft e digital divide*, in *Diritto di Autore e Nuove Tecnologie*, 2006, 2, 143-170.

¹⁶⁷ Tali licenze sono liberamente accessibili in rete: <https://www.gnu.org/licenses/licenses.it.html>.

¹⁶⁸ Queste rappresentano le 4 libertà fondamentali, che devono sussistere affinché un software possa essere definito come libero; Libertà 0: libertà di eseguire il programma per ogni scopo; Libertà 1: libertà di studiare il programma e modificarlo; libertà 2: Libertà di ridistribuire copie del programma; libertà 3: Libertà di migliorare il programma e di distribuirne pubblicamente i miglioramenti. In rete: <https://www.gnu.org/philosophy/free-sw.it.html>. Cfr. con i principi a cui si ispira il software open source: <https://opensource.org/docs/osd>.

¹⁶⁹ Da tale circostanza deriva il concetto di *copyleft*, ideato dallo stesso Stallman, ossia l'utilizzo del copyright per realizzare l'opposto del suo scopo; non si impongono restrizioni ma si concedono diritti, in modo tale da garantire che tali diritti non possano essere successivamente annullati. In Rete: <https://www.gnu.org/licenses/copyleft.it.html>.

¹⁷⁰ Il kernel Linux fu distribuito a partire dal 1991 grazie al lavoro di Linus Torvalds. Inizialmente, Linux utilizzava l'userspace di Minix, un sistema operativo che all'epoca veniva utilizzato per finalità didattiche. Successivamente, Linus decise di rendere il sistema indipendente da Minix, e decise di sostituire parte del sistema operativo con il software del progetto GNU. Per fare ciò, Torvalds cambiò la licenza e adottò la GPL. La sinergia tra Linux e la licenza GPL fu la fortuna della comunità del software libero. Infatti, Linux potette svilupparsi e migliorarsi velocemente, e, allo stesso tempo, le licenze GPL della FSF crebbero di fama grazie ad un prodotto estremamente interessante.

sviluppo è decentralizzato. La sua forza risiede nella vastità dei soggetti che fanno parte della fase realizzativa¹⁷¹. Questi elementi si traducono in un prodotto in continua evoluzione, ed in veloce miglioramento¹⁷².

6. Le misure tecnologiche di protezione, verso la chiusura del software?

Un secondo strumento attraverso cui i titolari dei diritti possono proteggere le loro opere, in questo caso il software, sono le misure tecnologiche di protezione (MTP).

Una MTP è una tecnologia formata o da soli software o dalla combinazione di strumenti hardware e software, destinata a impedire o limitare atti, su opere o altri materiali protetti, non autorizzati dal titolare del brevetto, del diritto d'autore, o del diritto connesso al diritto d'autore¹⁷³.

Questi strumenti vengono implementati nei software affinché l'utente non possa andare oltre a ciò che gli è permesso fare. E si deve ricordare, come si è visto in precedenza, che le libertà dell'utente vengono enucleate nella licenza.

Il software quindi, una volta implementate tali misure, "si protegge da solo".

Per effetto di tali misure è quindi impossibile per l'utente medio compiere attività che andrebbero al di là del contratto di licenza (*"enforcement tecnologico del contratto"*¹⁷⁴). Per esempio, acquistato un determinato software, è fisicamente impossibile copiarlo su un altro dispositivo, installarlo su un'altra macchina, così come trasmetterlo ad un altro utente, a meno che tali attività non siano permesse dalla licenza.

La MTP diviene ancor più penetrante quando impedisce quelle attività, che, in determinati casi stabiliti dalla legge, l'utente può mettere in atto: si fa riferimento alla attività di decompilazione, che abbiamo visto essere lecita in determinati casi, oppure alla estrazione di una copia del programma finalizzata al *back-up*.

Una MTP su un software veglia sull'integrità del contratto di licenza. Si potrebbe considerare come una tutela privata, implementata affinché l'utente

¹⁷¹ V. E. S. RAYMOND, *The Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*, O'Reilly Media Inc., Revised Edition, Sebastopol, 2001.

¹⁷² Per una disamina approfondita dell'Open Source, sia dal punto di vista economico che giuridico, v. M. BERTANI, *Open Source*, Quaderni di AIDA n. 13, Giuffrè Editore, Milano, 2005.

¹⁷³ Una efficace definizione di misura tecnologica di protezione viene fornita dal legislatore europeo nella Direttiva 2001/29/CE, articolo 6, comma 3: *"Ai fini della presente direttiva, per «misure tecnologiche» si intendono tutte le tecnologie, i dispositivi o componenti che, nel normale corso del loro funzionamento, sono destinati a impedire o limitare atti, su opere o altri materiali protetti, non autorizzati dal titolare del diritto d'autore o del diritto connesso al diritto d'autore, così come previsto dalla legge o dal diritto sui generis previsto al capitolo III della direttiva 96/9/CE. Le misure tecnologiche sono considerate «efficaci» nel caso in cui l'uso dell'opera o di altro materiale protetto sia controllato dai titolari tramite l'applicazione di un controllo di accesso o di un procedimento di protezione, quale la cifratura, la distorsione o qualsiasi altra trasformazione dell'opera o di altro materiale protetto, o di un meccanismo di controllo delle copie, che realizza l'obiettivo di protezione"*.

¹⁷⁴ Si veda R. CASO, *Digital rights management: il commercio delle informazioni digitali tra contratto e diritto d'autore*, CEDAM, Padova, 2004, ristampa 2006, disponibile in Rete: <http://www.jus.unitn.it/users/caso/pubblicazioni/drm/homeDRM.asp?cod=roberto.caso>.

rispetti i propri impegni contrattuali. In ogni caso, sempre di una tutela aggiuntiva si parla, tutela che non è disponibile per altri beni protetti con il copyright o con il brevetto¹⁷⁵.

A fronte di tali strumenti non mancano sicuramente metodi per aggirarli, e, il legislatore europeo, si adopera affinché gli Stati vegolino, attraverso il diritto penale, sull'integrità delle MTP¹⁷⁶. In particolare, essi devono predisporre un adeguato *public enforcement* contro le attrezzature, i prodotti, i componenti o i servizi che abbiano come scopo principale quello di eludere efficaci misure tecnologiche di protezione¹⁷⁷.

Il discorso che coinvolge le misure tecnologiche di protezione si fa sicuramente molto interessante se si parla di modifiche all'hardware¹⁷⁸. Questo perché la macchina è di proprietà dell'utente, mentre il software no. Una MTP che impedisce di modificare e di migliorare un bene di proprietà dell'utente è sicuramente un freno all'innovazione e alla creatività. E ciò è paradossale

¹⁷⁵ Per esempio, non esiste una tecnologia che si applica alle opere letterarie, protette dal diritto d'autore, volta a bloccare la sua riproduzione a mezzo di fotocopiatrice.

¹⁷⁶ La circostanza viene ben evidenziata nella direttiva 2001/29/CE, consistendo n. 47: *“Lo sviluppo tecnologico consentirà ai titolari dei diritti di far ricorso a misure tecnologiche per impedire o limitare atti non autorizzati dal titolare del diritto d'autore, dei diritti connessi o del diritto sui generis sulle banche dati. Esiste tuttavia il rischio di attività illegali intese a rendere possibile o a facilitare l'elusione della protezione tecnica offerta da tali misure. Per evitare soluzioni legislative frammentarie che potrebbero ostacolare il funzionamento del mercato interno è necessario prevedere una protezione giuridica armonizzata contro l'elusione di efficaci misure tecnologiche e contro la fornitura di dispositivi e prodotti o servizi a tal fine”*;

Ancora al considerando n. 48: *“Una siffatta protezione giuridica dovrebbe essere accordata alle misure tecnologiche che limitano in modo efficace atti non autorizzati dai titolari del diritto d'autore, dei diritti connessi o del diritto sui generis sulle banche dati, senza tuttavia impedire il normale funzionamento delle attrezzature elettroniche ed il loro sviluppo tecnologico. Tale protezione giuridica non implica alcuna obbligazione di adeguare i dispositivi, i prodotti, le componenti o i servizi a tali misure tecnologiche, purché detti dispositivi, prodotti, componenti o servizi non rientrino nel divieto di cui all'articolo 6. Tale protezione giuridica dovrebbe rispettare il principio della proporzionalità e non dovrebbe vietare i dispositivi o le attività che hanno una finalità commerciale significativa o un'utilizzazione diversa dall'elusione della protezione tecnica. Segnatamente, questa protezione non dovrebbe costituire un ostacolo alla ricerca sulla crittografia”*. Sul punto, R. FLOR, *Tutela penale e autotutela tecnologica dei diritti d'autore nell'epoca di Internet. Un'indagine comparata in prospettiva europea ed internazionale*, CEDAM, Padova, 2010, p. 604.

¹⁷⁷ Direttiva 2001/29, art. 6, commi 1 e 2: *“1. Gli Stati membri prevedono un'adeguata protezione giuridica contro l'elusione di efficaci misure tecnologiche, svolta da persone consapevoli, o che si possano ragionevolmente presumere consapevoli, di perseguire tale obiettivo. 2. Gli Stati membri prevedono un'adeguata protezione giuridica contro la fabbricazione, l'importazione, la distribuzione, la vendita, il noleggio, la pubblicità per la vendita o il noleggio o la detenzione a scopi commerciali di attrezzature, prodotti o componenti o la prestazione di servizi, che: a) siano oggetto di una promozione, di una pubblicità o di una commercializzazione, con la finalità di eludere, o b) non abbiano, se non in misura limitata, altra finalità o uso commercialmente rilevante, oltre quello di eludere, o c) siano principalmente progettate, prodotte o realizzate con la finalità di rendere possibile o di facilitare l'elusione di efficaci misure tecnologiche”*.

¹⁷⁸ Su tutti si veda R. CASO, *Technological Protection Measures: Fifty (and More) Shades of Grey of the European Court of Justice*, Trento Law and Technology Research Group, Research Paper n. 19, 2014.

perché il fine ultimo della tutela della proprietà intellettuale è il progresso della cultura e dell'innovazione¹⁷⁹.

Questi argomenti costituiscono tuttavia proprio la giustificazione usata del legislatore europeo per legittimare l'ammissibilità di una così penetrante tutela delle MTP, che, come emerge nel considerando della direttiva 2001/29/CE, risiede per l'appunto nel necessario incentivo che gli Stati devono fornire agli autori affinché questi, vedendo il loro investimento protetto, possano continuare a creare innovazione¹⁸⁰.

7. La tutela del software e la sua percezione

Si è visto come il diritto offra all'autore di un software numerosi strumenti di tutela.

Interessante è anche valutare se e come l'utenza percepisca queste forme di protezione. Una utenza disinteressata, passiva e poco preparata dal punto di vista tecnico, sicuramente avvantaggia una concezione chiusa del software. Questo perché se l'utente non ha alcun interesse, o capacità, nell'apportare modifiche o miglioramenti nel software al fine di adattarlo alle proprie esigenze, è il produttore a stabilire cosa serva all'utenza, e quest'ultima si adatta a tali scelte.

Si deve ricordare che la funzione del software è quella di mettere in contatto l'utente con la macchina. Quest'ultimo elemento è importante perché negli ultimi anni le strategie commerciali dei produttori stanno portando non solo a un software chiuso, ma anche a una macchina chiusa.

¹⁷⁹ Sul punto v. W. SELTZER, *The Imperfect Is the Enemy of the Good: Anticircumvention versus Open User Innovation*, 25 *Berkeley Tech. L.J.* 909, 2010. In Rete: <http://scholarship.law.berkeley.edu/cgi/viewcontent.cgi?article=1847&context=btlj>.

¹⁸⁰ Il punto, nella direttiva in oggetto, è ampiamente evidenziato nel considerando n. 9: "Ogni armonizzazione del diritto d'autore e dei diritti connessi dovrebbe prendere le mosse da un alto livello di protezione, dal momento che tali diritti sono essenziali per la creazione intellettuale. La loro protezione contribuisce alla salvaguardia e allo sviluppo della creatività nell'interesse di autori, interpreti o esecutori, produttori e consumatori, nonché della cultura, dell'industria e del pubblico in generale. Si è pertanto riconosciuto che la proprietà intellettuale costituisce parte integrante del diritto di proprietà";

In particolare, con riferimento ai beni di informazione v. il considerando n. 10: "Per continuare la loro attività creativa e artistica, gli autori e gli interpreti o esecutori debbono ricevere un adeguato compenso per l'utilizzo delle loro opere, come pure i produttori per poter finanziare tale creazione. Gli investimenti necessari a fabbricare prodotti quali riproduzioni fonografiche, pellicole o prodotti multimediali e servizi quali i servizi su richiesta («on-demand») sono considerevoli. È necessaria un'adeguata protezione giuridica dei diritti di proprietà intellettuale per garantire la disponibilità di tale compenso e consentire un soddisfacente rendimento degli investimenti";

Infine, anche con riferimento allo sviluppo culturale della comunità v. il considerando n. 11: "Un sistema efficace e rigoroso di protezione del diritto d'autore e dei diritti connessi è uno dei principali strumenti in grado di garantire alla creazione e alla produzione culturale europea le risorse necessarie nonché di preservare l'autonomia e la dignità di creatori e interpreti o esecutori" e 12: "Un'adeguata protezione delle opere tutelate dal diritto d'autore e delle opere tutelate dai diritti connessi assume grande importanza anche sotto il profilo culturale. L'articolo 151 del trattato obbliga la Comunità a tener conto degli aspetti culturali nell'azione da essa svolta".

La connessione tra software e hardware è sempre più stretta; si prenda ad esempio il mercato degli smart devices. Questi dispositivi sono prevalentemente chiusi sia nel software che nell'hardware. Molti software -e non solo quello operativo- sono preinstallati e la modifica della parte hardware è, se non impossibile, quantomeno difficile per ragioni tecniche o economiche.

Una situazione di tale tipo sicuramente non sussisteva fino ad un decennio fa, quando la macchina era, da un punto di vista tecnico, liberamente e quasi completamente modificabile. Ora, guardando anche alla tipologia di computer più diffusi, i laptop, la tendenza è quella di chiudere la componente hardware¹⁸¹.

Un hardware non modificabile vuole dire non migliorabile. Se un dispositivo non si migliora nel tempo, diventa obsoleto. Nell'era digitale l'obsolescenza di un sistema è una questione di pochi anni. Con il cambiamento della macchina viene automaticamente imposto il cambiamento del sistema operativo, nonché di tutti gli altri software applicativi proprietari, solitamente preinstallati, sui quali vengono implementate misure tecnologiche di protezione sempre più penetranti.

Le strategie commerciali dei principali operatori nel mercato informatico vanno verso una chiusura del software e dell'hardware, e ad un tale cambiamento l'utenza sembra essere passiva¹⁸². Tale atteggiamento è probabilmente conseguenza di un nuovo modello di commercializzazione e distribuzione dei prodotti digitali. Mentre in passato i software venivano licenziati a costi molto elevati, oggi i costi sono di molto inferiori. Dal 1996 al 2016 i prezzi dei software sono diminuiti in media del 66%¹⁸³.

¹⁸¹ Ciò avviene tecnicamente attraverso saldature praticate direttamente sulla scheda madre. In questo modo i componenti saldati, ossia la CPU e le SSD, non sono modificabili e sostituibili dall'utente.

¹⁸² Per precisione e correttezza del ragionamento, si deve tuttavia sottolineare come il sistema operativo più diffuso a livello mondiale nei dispositivi mobili, Android, sia un software open source. Secondo gli ultimi dati del 2017, Android ricopre una quota del 64%, riferita al mercato mondiale dei sistemi operativi mobili (Fonte: netmarketshare.com). In precedenza si è analizzato il software libero, ed è bene in questo frangente sottolineare come Android non sia libero, ma open source. Il codice sorgente è pubblicato da Google (che sviluppò Android sulla base del kernel di Linux), ma non è prevista nella licenza l'esercizio delle 4 libertà fondamentali che caratterizzano le licenze libere.

In ogni dispositivo Android, che è marchiato a seconda del produttore dell'hardware, sono inclusi driver, firmware e librerie i cui codici non sono pubblicati. Ciò significa che quel dispositivo, per esempio Samsung o LG, non potrebbe funzionare correttamente con una versione di Android completamente libera, perché le aziende produttrici della parte hardware hanno introdotto questi elementi senza pubblicarli. Facendo un esempio pratico: installando una versione di Android modificata da un utente, su di un dispositivo Samsung, alcune funzionalità sarebbero precluse perché sono necessari determinati driver per farle funzionare (ad esempio è necessario uno specifico driver per il funzionamento della fotocamera, del wi-fi, del microfono, e così via). Questi driver sono proprietari, e raramente sono resi disponibili agli utenti.

Ulteriore snodo critico risiede nel fatto che una applicazione molto importante, Google Play, attraverso la quale gli utenti acquistano moltissime applicazioni, è di proprietà di Google, ma non fa parte di Android. Google quindi pur non traendo direttamente profitto da Android, riesce a capitalizzare attraverso il proprio Store Online, che è cresciuto di importanza proprio grazie alla grande diffusione di Android. Questa, almeno da un punto di vista etico, è sicuramente una pratica incoerente. Criticità queste sottolineate dalla stessa FSF. In Rete: <https://fsfe.org/campaigns/android/android.it.html>.

¹⁸³ Fonte: BLS (U.S. Bureau of Labor Statistics) in rete: <https://www.bls.gov/data/#prices>.

È inoltre rilevante la circostanza che molti software sono forniti all'utente in modo gratuito; o meglio, il prezzo del software non è rappresentato da una somma di danaro che si deve immediatamente pagare. Molti software generano profitti con altre strategie di prezzo. Ad esempio si implementano servizi pubblicitari grazie alla collaborazione con aziende terze. Il software licenziato gratuitamente avrà quindi una maggiore diffusione nel pubblico, e più numerosi sono gli utenti, maggiore sarà l'interesse delle aziende ad inserirvi la propria pubblicità.

In altri casi il software viene offerto gratuitamente ma sono previsti servizi o funzioni aggiuntive a pagamento. Ancora, un altro modo per favorire una efficiente discriminazione di prezzo è concedere licenze di utilizzo di breve durata, solitamente un anno, ad un prezzo decisamente accessibile. Infine, in alcuni casi, il prezzo da pagare per l'utilizzo di un software risiede nella cessione, a volte inconsapevole, dei propri dati personali¹⁸⁴. In ogni caso un prezzo c'è sempre.

Attraverso tali meccanismi la maggior parte degli utenti non solo si trovano nella condizione di acquistare il programma, ma sono anche decisamente meno propensi a mettere in atto comportamenti volti alla realizzazione o all'utilizzo della copia illegittima¹⁸⁵.

¹⁸⁴ La questione è molto rilevante ed investe il diritto alla *privacy* degli utenti. La circostanza è divenuta rilevante con l'avvento del web 2.0. Spesso i contenuti sono elaborati direttamente dall'utenza (si pensi ad esempio ai social network) che non è soltanto passiva ma ha assunto un ruolo attivo. Tali contenuti, unitamente ai dati, anche basilari, dell'utente sono il prezzo da pagare per l'accesso a determinati prodotti e servizi. Si veda su tutti PASCUZZI (a cura di), *Il diritto nell'era digitale*, pp. 43-77.

¹⁸⁵ Una interessante ricerca evidenzia come tale circostanza non sia esatta con riferimento al mercato dei videogames: C. DARROCH, *Problems and Progress in the Protection of Videogames: A Legal and Socio-logical Perspective*, The University of Manchester, 2012, in Rete: [http://www.humanities.manchester.ac.uk/medialibrary/law/main_site/Research/Student_Law_Review1/MSLR_Vol1_10\(Darroch\).pdf](http://www.humanities.manchester.ac.uk/medialibrary/law/main_site/Research/Student_Law_Review1/MSLR_Vol1_10(Darroch).pdf).

CAPITOLO III

Il quadro legislativo e le *guidelines* degli Uffici Brevetti

SOMMARIO: *Premessa* - 1. Cenni alla legislazione europea prima della European Patent Convention - 1.1. La Francia - 1.2. L'Inghilterra - 2. La Convenzione sul Brevetto Europeo - 2.1. EPO, struttura e organizzazione - 2.2. (Segue): il brevetto unitario europeo - 2.3. EPC: normativa - 2.4. Le Linee Guida: contributo tecnico, programma per elaboratore e invenzione attuata a mezzo di elaboratore elettronico - 3. Le convenzioni internazionali sui brevetti - 4. Legislazione dell'Unione Europea - 4.1 Le direttive europee e la tutela giuridica dei programmi per elaboratore (cenni) - 4.2. Il fallimento della direttiva relativa alla brevettabilità delle invenzioni attuate per mezzo di elaboratori elettronici - 5. La legislazione degli Stati Uniti - 5.1. Il Patent Act del 1952 - 5.2. Dalla Commissione CONTU al Computer Software Copyright Act del 12 dicembre 1980 - 5.3. Le *Guidelines* del Patent Office statunitense

Premessa

Ora che si è a conoscenza delle caratteristiche salienti del software e della storia della sua evoluzione è possibile guardare con più coscienza alla legislazione. Nel farlo il lettore avrà cura di rammentare da un lato lo stato della tecnica valevole in quel determinato periodo storico, dall'altro l'oggetto che si vuole tutelare, considerato nella sua essenza, in modo tale da vagliare in modo critico l'opportunità delle scelte operate dal legislatore.

Il punto centrale del capitolo è la comparazione tra la normativa europea e quella statunitense.

La prima "oscilla" tra la Convenzione Europea sul Brevetto, che non esclude *tout court* il software dall'alveo delle invenzioni brevettabili, e le direttive dell'Unione Europea, che muovono nel chiaro senso di riconoscere una tutela forte mediante il solo diritto d'autore, state il fallimento della direttiva europea sulla brevettabilità delle invenzioni attuate a mezzo di elaboratore elettronico.

La seconda disciplina, prima interpretata in una maniera tale da essere avversa ad una tutela brevettuale del programma per elaboratore, poi indirizzata verso il prevalente (per non dire indiscriminato) riconoscimento di una tutela brevettuale.

Nello svolgere questa analisi si prenderanno in considerazione anche le linee guida fornite dagli uffici brevetti, in modo tale da ottenere una interpretazione "autentica" della disciplina.

1. Cenni alla legislazione europea prima della European Patent Convention

1.1. La Francia

Diversi studi di diritto comparato, illustrando le normative europee nazionali prima della Convenzione Europea sui brevetti, mostrano come il software fosse brevettato principalmente solo in Inghilterra¹⁸⁶. Mentre negli altri paesi europei, l'orientamento da seguire in merito alla brevettabilità dei programmi per elaboratore, era incerto o "neutrale", in Francia tale circostanza fu esclusa esplicitamente per legge¹⁸⁷.

Per comprendere i motivi alla base di una tale scelta normativa bisogna analizzare i lavori preparatori alla legge n°68-1 del parlamento francese¹⁸⁸. Il primo e dibattuto argomento investì la natura del software. La questione principale consisteva nel considerare o meno il software come una entità astratta. La circostanza era importante perché la legge sul brevetto non consentiva la brevettabilità di invenzioni astratte.

Si riscontrarono posizioni diverse tra il Senato e l'Assemblea Nazionale. Il primo aveva una posizione più flessibile, evidenziando come in taluni casi il software potesse avere un carattere industriale¹⁸⁹. Prevalse tuttavia la valutazione dell'Assemblea che identificava il software in un insieme di algoritmi, che, per loro stessa natura, sono astratti¹⁹⁰. Infine si rilevò come il concedere la brevettabilità ai programmi per elaboratore, avrebbe potuto portare al riconoscimento di una tutela brevettuale anche alle formule matematiche¹⁹¹.

Queste non furono tuttavia le sole considerazioni che portarono alla negazione della tutela brevettuale al software. Anzitutto si tenne in forte

¹⁸⁶ In questo senso si esprimono A. MANCINI, *International patent law is obsolete*, Buenos Books America, 2006, che si focalizza sull'esperienza francese e LEITH, *Software and patents in Europe*, che esamina con accuratezza le vicende inglesi. Per una disamina di tipo comparato v. GUGLIELMETTI, *L'invenzione di software. Brevetto e diritto d'autore*, parte prima.

¹⁸⁷ Art. 7, Loi n°68-1 du 2 janvier 1968 sur les brevets d'invention: "*Est considérée comme industrielle toute invention concourant dans son objet, son application et son résultat, tant par la main de l'homme que par la machine, à la production de biens ou de résultats techniques. Ne constituent pas, en particulier, des inventions industrielles :*

1 Les principes, découvertes et conceptions théoriques o u purement scientifiques ;

2 Les créations de caractère exclusivement ornemental;

3 Les méthodes financières ou comptables, les règles de jeux et tous autres systèmes de caractère abstrait, et notamment les programmes ou séries d'instructions pour le déroulement des opérations d'une machine calculatrice."

¹⁸⁸ Per una analisi approfondita dei lavori preparatori alla Loi n°68-1 du 2 janvier 1968 v. A. MANCINI, *International patent law is obsolete*, Buenos Books America, 2006, in particolare il Capitolo I "*The history of legal protection for computer software in France, in reaction to the American position*". Parte dei lavori preparatori sono disponibili in Rete: <http://www.senat.fr/dossier-legislatif/a76772902.html#block-timeline>.

¹⁸⁹ V. Report di MARCILHACY pp. 18 ss., in Rete: http://www.senat.fr/rap/1977-1978/i1977_1978_0281.pdf

¹⁹⁰ MANCINI, *International patent law is obsolete*, cit., 5.

¹⁹¹ *Ibidem*, cit., 6.

considerazione quella che era al tempo la posizione degli Stati Uniti¹⁹², la cui industria nell'ambito dei computer era la più sviluppata. In questo periodo negli USA si stava lavorando all'inclusione del software nel Copyright Act, che vedrà luce nel 1980.

A questo argomento di natura comparata, si aggiunse una considerazione di carattere politico¹⁹³; la concessione di una tutela brevettuale in soli determinati casi avrebbe lasciato un potere discrezionale troppo lato alle corti, volgendo quindi la questione in uno stato di incertezza.

Infine un certo peso ebbero le considerazioni di carattere economico. Le aziende più sviluppate nell'ambito dei software erano statunitensi, e concedere una tutela brevettuale avrebbe significato largire un monopolio a soggetti stranieri paralizzando il mercato tecnologico francese¹⁹⁴.

Sulla base di queste considerazioni l'articolo 7 della legge francese, nel richiedere che le invenzioni brevettabili siano atte ad avere una applicazione industriale, esclude i programmi per elaboratore dall'ambito del brevettabile, inserendoli tra le procedure meramente astratte, prive del carattere di industrialità.

La prima applicazione pratica di questa legge si rinviene nel caso Mobil Oil del 22 maggio 1973¹⁹⁵. In quell'anno la Mobil Oil chiese la brevettabilità di un procedimento computerizzato per la preparazione di tinte. La richiesta di registrazione fu respinta dall'ufficio brevetti in base alla circostanza, desunta dall'articolo 7 della legge sui brevetti, che l'invenzione in questione era non brevettabile in quanto classificabile come "programma per elaboratore".

L'azienda fece ricorso presso la Corte d'Appello di Parigi. Il ricorso si basava sul fatto che l'invenzione non riguardava un programma "in sé", ma era invero un procedimento industriale che era implementato e realizzato anche a mezzo di un programma per elaboratore elettronico. Il ricorso fu tuttavia respinto con una interpretazione estensiva dell'articolo 7, volto a escludere dalla privativa brevettuale tutti i programmi per elaboratore, indipendentemente dalla idoneità degli stessi ad avere una applicazione industriale. Tale decisione fu successivamente confermata dalla Cour de Cassation, ribadendo la mancanza della caratteristica del risultato industriale¹⁹⁶.

L'atteggiamento assunto dal legislatore francese è sicuramente attribuibile ad evidenti incertezze circa la comprensione del bene in oggetto, riferibili anche al periodo storico di riferimento. Si rileva tuttavia un ulteriore elemento molto importante: la circostanza che la protezione della proprietà intellettuale ha una forte connotazione nazionale.

La posizione assunta dalla legge tende sempre a tutelare e avvantaggiare l'economia interna dello stato nel quale la legge trova applicazione. Paesi come per esempio la Francia, dove l'industria software non era, e non è, di certo

¹⁹² *Ibidem*, cit., 6.

¹⁹³ *Ibidem*, cit., 8.

¹⁹⁴ *Ibidem*, cit., 7.

¹⁹⁵ Cour D'Appel de Paris, 22 mai 1973, in Ann. 1973, 275.

¹⁹⁶ Cour de Cassation, Chambre commerciale, du 28 mai 1975, 73-12.820, in Rete: <https://www.legifrance.gouv.fr/affichJuriJudi.do?idTexte=JURITEXT000006994145>.

paragonabile a quella anglosassone¹⁹⁷, propendono per un software meno protetto, meno chiuso, in modo tale da poter permettere alle proprie aziende di trarne un piccolo vantaggio¹⁹⁸. A ben vedere, tali considerazioni possono valere anche oggi, anche se è sicuramente controverso stabilire attraverso quali strumenti del diritto si possa ottenere una protezione più o meno penetrante.

1.2. L'Inghilterra

Le vicende giuridiche che riguardano la brevettabilità del software in Europa, in questo periodo storico, sono molto interessanti con riguardo alle vicende inglesi, dove si registra un orientamento non avverso allo strumento brevettuale applicato al software. L'apertura concettuale alla privativa in oggetto può essere causata da molteplici fattori.

Come si è visto nel primo capitolo, l'Inghilterra svolse un ruolo importante nella nascita dei primi computer¹⁹⁹. Inoltre, nel dopoguerra, il paese assunse una posizione di supremazia economica in Europa, ed infine non è da sottovalutare la vicinanza, in primis linguistica, con l'industria americana, molto avanzata in questo ambito²⁰⁰.

Per queste ragioni storiche ed economiche ivi si sviluppò una interessante giurisprudenza, resa possibile da richieste di registrazione di determinati sistemi integranti una parte hardware e una software. L'innovazione e il mercato hanno introdotto l'argomento prima del legislatore. Questi, o meglio le corti, si sono prodigate al fine di offrire agli operatori economici una tutela adeguata, conformemente ai principi generali stabiliti dalla legge del tempo.

Una primo dato rilevante attiene ad uno dei primi brevetti rilasciati su di un programma per elaboratore. La richiesta di registrazione fu depositata nel Regno Unito il 21 maggio 1962. L'invenzione consisteva in un "computer predisposto alla soluzione automatica di problemi di programmazione lineare"²⁰¹. La richiesta venne accolta e il brevetto rilasciato nel 1966.

Invero, in tale periodo, non si ha ancora una piena consapevolezza di cosa sia un software considerato in sé e per sé. Non è un caso che il brevetto in questione parli di un computer, ossia fondamentalmente di un sistema composto da una parte di hardware unita da un legame indissolubile ad una parte software. L'invenzione così concepita era quindi materiale, tangibile e brevettabile alla luce della legislazione inglese.

Non essendoci riferimenti specifici al software nel Patent act del 1949, era d'obbligo fare riferimento ai criteri e alle condizioni generali stabiliti nella legge,

¹⁹⁷ Sul punto v. GUGLIELMETTI, *L'invenzione di software. Brevetto e diritto d'autore*, cit., 5: "uno strumento di tutela forte avrebbe potuto ulteriormente accrescere nel mercato francese il vantaggio competitivo di cui già allora godevano le grandi imprese statunitensi, ostacolando in patria lo sviluppo dell'industria nazionale".

¹⁹⁸ In questo senso LEITH, *Software and patents in Europe*, pp. 21-22.

¹⁹⁹ V. *supra*, cap. I, par. 1.1.

²⁰⁰ LEITH, *Software and patents in Europe*, pp. 21-22.

²⁰¹ GB1039141: "A computer arranged for the automatic solution of linear programming problems". Il brevetto in Rete: https://worldwide.espacenet.com/publicationDetails/biblio?DB=EPODOC&II=0&ND=3&adjacent=true&locale=en_EP&FT=D&date=19660817&CC=GB&NR=1039141A&KC=A.

dove la brevettabilità dell'invenzione era sostanzialmente collegata alla scoperta di un nuovo metodo di fabbricazione²⁰².

Solo negli anni 70 si consuma il passaggio da una concezione di innovazione meramente legata alla materialità, a una concezione legata alla applicazione. Il passaggio è molto importante perché la componente innovativa portata dal software non investe una innovazione legata alla macchina. La macchina rimane la stessa da un punto di vista materiale, ma attraverso il software riesce a svolgere funzioni e applicazioni nuove e ulteriori²⁰³.

Indicativo di tale passaggio è il caso che si è innestato in Inghilterra sul brevetto Nymeyer²⁰⁴. La richiesta di registrazione dell'invenzione in questione fu esposta allo UK Patent Office nel 1971 con il nome di "miglioramenti relativi all'elaborazione dei dati", sulla base di un brevetto già registrato negli Stati Uniti²⁰⁵.

L'invenzione in questione permette che, attraverso calcoli automatizzati, si conosca immediatamente il valore corrente di quelle merci precedentemente comprate, il valore complessivo del portfolio, nonché l'aumento del capitale complessivo espresso in percentuale. In pratica si tratta di una agenda finanziaria. Punto fondamentale è che l'idea in questione, pur essendo un nuovo metodo per aumentare la produttività, non implementa una innovazione tangibile nella tecnica informatica (*business idea via a computer system*)²⁰⁶.

Al momento della richiesta di registrazione del brevetto, Nymeyer rappresenta la sua invenzione su schemi e diagrammi concettuali che mischiano linguaggi, concetti e processi propri dell'economia e dell'informatica, sia nella parte hardware che nella parte software²⁰⁷. Attraverso una tale descrizione l'inventore è riuscito a sottolineare quelle qualità che hanno reso un software paragonabile ad una macchina vera e propria, e così facendo il brevetto inglese gli fu accordato nel 1974.

²⁰² Patent Act 1949, section 101 (1): *"Invention means any manner of new manufacture the subject of letters patent and grant of privilege within section six of the Statute of Monopolies and any new method or process of testing applicable to the improvement or control of manufacture, and includes an alleged invention"*.

²⁰³ V. LEITH, *Software and Patents in Europe*, cit., 11, dove si evidenzia tale circostanza parlando di *"hardware-oriented patent application"* e di *"application-oriented patent"*: *"in the 1970s there were attempts to move from the solely hardware-oriented patent application to that of application-oriented patent, where the novelty lay in the software being run on non-novel hardware"*.

²⁰⁴ GB1352742: *"Improvements Relating Data Processing"*. Il brevetto in Rete: https://worldwide.espacenet.com/publicationDetails/biblio?DB=EPODOC&II=0&ND=3&adjacent=true&locale=en_EP&FT=D&date=19740508&CC=GB&NR=1352742A&KC=A.

²⁰⁵ US 3,581,072 *"Auction Market Computation System"*, registrato nel 1968 e pubblicato nel 1971. In Rete: <https://www.google.com/patents/US3581072>.

²⁰⁶ V. LEITH, *Software and Patents in Europe*, cit. 12: *"These patents are essentially implementing business ideas via a computer system (...) Neither of these patents utilise 'novel' computing techniques"*.

²⁰⁷ V. L. FLECK, *Genesis and Development of a Scientific Fact*, Chicago, IL, Chicago University Press, 1979, cit., 137, in LEITH, *Software and Patents in Europe*, cit., 13, che definisce la rappresentazione in oggetto come: *"a graphic representations of certain ideas and certain meanings. It involves a kind of comprehending where the meaning is represented as a property of the object illustrated"*.

In seguito IBM fece richiesta di revoca presso il Patent Appeal Tribunal²⁰⁸. Punto controverso era la possibilità di ricomprendere una tale invenzione sotto il concetto di “*invention*” così come stabilito dal patent act del 1949.

IBM sostenne che l’idea di Nymeyer era applicabile su ogni computer standard così da non comportare nessun miglioramento alla macchina in sé. Secondo la posizione della nota azienda, anche se una idea è nuova e non ovvia, non è un nuovo metodo di fabbricazione, per il solo fatto di permettere ad un computer ordinario, ossia senza modifiche alle sue componenti, di svolgere ciò che a ben vedere anche una persona può già fare.

La corte tuttavia adottò una prospettiva diversa, guardando all’invenzione nel suo complesso, valorizzando l’elemento inventivo, rapportandolo alle innovazioni meramente tangibili. Quando l’invenzione viene implementata su di una macchina standard, questa si specializza acquisendo particolari funzioni anche se non vi sono state modifiche o migliorie nelle componenti materiali.

Risulta quindi evidente come in questa fase, pure essendo il software come oggi noi lo intendiamo, tutelato anche mediante brevetto, le giustificazioni della tutela muovevano dalla premessa dello stretto legame tra programma e macchina. Un legame che va capito, anche e soprattutto tenendo in considerazione il periodo storico; le macchine erano molto costose e la parte software era generalmente legata quasi indissolubilmente alla parte hardware, in un mercato radicalmente diverso da quello odierno.

La tangibilità dell’invenzione, l’analogia tra software e macchina era il punto centrale, nonché condizione necessaria per vedere accolta la richiesta di registrazione. Software e hardware si considerano uniti e formano un sistema unico. Una tale impostazione, non fu invero *tout court* abbandonata con il corso degli anni, ma anzi venne ripresa nella sua essenza in pronunzie relativamente recenti²⁰⁹.

Lo stesso ufficio dei brevetti, basandosi su questa finzione, attestò la sua posizione ufficiale²¹⁰, almeno fino al 1977, ossia con l’introduzione del nuovo Patent Act, che segue in linea temporale la Convenzione Europea sul Brevetto.

In questa fase il software è inteso ancora come macchina, o meglio come parte di un sistema complesso, e brevettabile solo in tale senso. Il programma per elaboratore non è quindi brevettabile in quanto tale, ma se esso è

²⁰⁸ Application by IBM for the Revocation of Letters Patent N. 1,352,742 In the Name of Frederick Nymeyer, Patents Appeal Tribunal, 16 October 1978. Il documento è reperibile in UNH School of Law, *Diamond V. Diehr - a brief amicus curiae for Chevron Research Company*, appendice A, in Rete: <https://ipmall.law.unh.edu/content/diamond-v-diehr-brief-amicus-curiae-chevron-research-company-support-affirmance>.

²⁰⁹ Si fa riferimento a *In re Alappat*, 33 F.3d 1526 (Fed. Circ. 1994), vedi *infra*, cap. IV, par. 1.4.

²¹⁰ UK Patent Office, Linee guida, Febbraio 1969: “*patents are not granted for computer programmes expressed as such. No objection is, however, raised in respect of inventions for novel methods of programming computers to operate in a specified way, or for computers so programmed, or for tape etc. having recorded on it a novel programme to control a computer in a started way. Nor, in general, is objection taken to inventions involving new uses of computer in controlling manufacturing processes or to methods of testing, involving novel programmes, for computers under manufacture*”. Il testo è reperibile nella rivista *New Scientist* del 30 novembre 1972, p. 510.

incorporato in una macchina, ovvero è considerato alla stregua della stessa, allora la situazione, come si è visto, cambia²¹¹.

Una tale posizione non deve sorprendere essendo sicuramente presente anche l'influenza della dottrina e della giurisprudenza nord americana. Come si vedrà meglio nel proseguo del lavoro, un punto fermo in questo periodo storico era la non brevettabilità delle formule matematiche; ed essendo il software, almeno da un punto di vista meramente fattuale, un insieme di numeri e di formule che vanno a comporre algoritmi, allora, sulla base di tale visione del programma per elaboratore, sicuramente riduttiva, non si può brevettare²¹².

2. La Convenzione sul Brevetto Europeo

2.1. EPO, struttura e organizzazione

Importanza cruciale, per l'elaborazione di una tutela del software coerente ed uniforme in Europa, ebbe la Convenzione sul Brevetto Europeo²¹³. L'accordo fu firmato a Monaco di Baviera il 5 ottobre 1973, ed entrò in vigore il 7 ottobre 1977. L'ultima revisione fu fatta il 29 novembre 2000.

Molti dei principi affermati erano già stati enunciati nella Convenzione di Strasburgo del 1963 sull'unificazione di taluni elementi del diritto dei brevetti d'invenzione, promossa dal Consiglio d'Europa ed in vigore solo dal 1980. L'importanza pratica di tale ultima convenzione è invero limitata²¹⁴ ma fissò principi di base che vennero appunto recepiti nella EPC.

Dalla Convenzione Europea sul Brevetto nacque l'Organizzazione Europea dei Brevetti²¹⁵, istituzione indipendente dalla allora Comunità Europea e controllata dagli stessi stati membri dell'organizzazione che si riuniscono nella Conferenza dei Ministri²¹⁶.

La funzione di esaminare le domande, verificando la sussistenza dei requisiti formali e sostanziali, al fine concedere o meno il brevetto è esercitata da un organo ad hoc: l'Ufficio Europeo dei Brevetti²¹⁷. Questo al suo interno si articola in diverse Divisioni d'Esame, ossia commissioni di esperti la cui composizione varia a seconda della natura della domanda da analizzare. Oltre ai membri tecnici è presente anche uno staff giuridico specializzato. L'ufficio comprende anche le Commissioni Tecniche di Ricorso, competenti ad esaminare le controversie, ossia le situazioni in cui il richiedente, vedendosi negare la richiesta di registrazione dell'invenzione, promuove ricorso.

²¹¹ V. LEITH, *Patenting programs as machines*.

²¹² Si fa in particolare riferimento alla tesi dei "*mental steps*". V. *infra*, cap. IV, par. 1.2, nota n. 355.

²¹³ Di seguito EPC, secondo l'acronimo internazionale di European Patent Convention. Il testo in Rete: <http://www.epo.org/law-practice/legal-texts/epc.html>.

²¹⁴ Stante il cospicuo numero di paesi firmatari, 14, nonché la generalità delle prescrizioni contenute. Punti centrali della Convenzione sono i requisiti di brevettabilità e i contenuti delle domande di brevetto. Il testo in Rete: <http://www.coe.int/en/web/conventions/full-list/-/conventions/treaty/047>.

²¹⁵ Di seguito EPOrg, secondo l'acronimo internazionale di European Patent Organization.

²¹⁶ A oggi i membri che compongono l'EPOrg, ossia gli stati firmatari dell'EPC, sono 38.

²¹⁷ Di seguito EPOff, secondo l'acronimo internazionale di European Patent Office.

È necessario sin da ora sottolineare come, mancando ad oggi la figura del brevetto europeo, inteso come titolo unitario immediatamente valido in tutti i paesi dell'Unione, l'EPO abbia svolto, e svolge, un ruolo di fondamentale importanza dal punto di vista della armonizzazione del diritto brevettuale europeo.

Il brevetto europeo concesso dall'EPO, è un fascio di brevetti nazionali che vengono contestualmente rilasciati da un unico organo competente. La comodità del sistema è indubbia in quanto si evita che il richiedente debba espletare la procedura in ogni singolo Stato.

Tuttavia il sistema ha precise criticità²¹⁸. Anzitutto ogni brevetto europeo è soggetto non solo alla convenzione, ma anche alle singole normative e discipline nazionali. Può quindi accadere che un brevetto valido in Italia venga invece invalidato in Germania, perché, per esempio, una invenzione analoga era già stata registrata direttamente presso l'ufficio nazionale di quel paese.

In questo modo può palesarsi la situazione che siano messi in circolazione prodotti contenenti brevetti tutelati solo in determinati stati, mentre in altri gli stessi prodotti possono considerarsi come lesivi di altrui invenzioni. Tale circostanza può portare a problematiche di spessore essendo il mercato interno europeo basato sul libero scambio e sulla libera circolazione delle merci²¹⁹.

Se si trasferisce tale rilievo ad un brevetto su un software, il problema assume dimensioni ancor più rilevanti, essendo la circolazione, nella società dell'informazione, ancor più celere e insensibile ai confini nazionali.

Rilievi ancora più critici sono mossi con riguardo alla giurisdizione. Se un soggetto ritiene che il brevetto concesso ad un terzo leda i propri diritti, perché per esempio copia un altro brevetto già rilasciato, allora dovrà opporsi separatamente in ogni stato, ove è consumata la violazione, dando luogo ad una pluralità di ricorsi.

Considerando i costi e i tempi medi della giustizia è palese che la fase di contenzioso costituisce un notevole disincentivo alla brevettabilità delle invenzioni, a favore invece del segreto industriale.

A tali conclusioni è giunta anche la Commissione europea, evidenziando come *“il sistema esistente, che comporta il pericolo di controversie multiple in materia di brevetti, ha molte conseguenze che indeboliscono il sistema dei brevetti in Europa e rendono meno attraenti i brevetti europei, soprattutto per le PMI”*²²⁰. Inoltre, esiste il rischio che *“la controversia multipla in materia di*

²¹⁸ Per una approfondita disamina v. Comunicazione della commissione al parlamento europeo e al consiglio, *Migliorare il sistema dei brevetti in Europa*, Bruxelles, 03/04/2007 COM(2007) 165 definitivo. In Rete: <http://eur-lex.europa.eu/legal-content/IT/TXT/PDF/?uri=CELEX:52007DC0165&from=it>.

²¹⁹ La libera circolazione delle merci, è una delle quattro libertà fondamentali dell'Unione europea. V. Trattato sul funzionamento dell'Unione europea (TFUE), Artt. 28 - 37.

²²⁰ Comunicazione della commissione al parlamento europeo e al consiglio, *Migliorare il sistema dei brevetti in Europa*, Bruxelles, 03/04/2007, cit., 6.

*brevetti possa portare a risultati variabili, ove non contraddittori, nei vari Stati*²²¹ (...) *nonché la possibilità del verificarsi di casi di forum-shopping*²²².

Le piccole e medie imprese, sono quindi frustrate da un tal sistema di ricorsi, soprattutto a causa dei costi esorbitanti cui dovrebbero andare in contro²²³.

I costi, non sono tuttavia esorbitanti solo nella eventuale fase di contenzioso. Le spese che sono necessarie per la registrazione dell'invenzione, variano a seconda del numero di stati presso cui si vuole fare la domanda.

La Commissione Europea, ha calcolato che i costi procedurali, comprensivi di traduzioni, di deposito, e di rilascio del brevetto europeo, con successiva nazionalizzazione in tredici Paesi dell'Unione ammonterebbero a € 129.183,00 nell'arco di venti anni, nonché a € 43.228,00 per una registrazione limitata a solo tre paesi dell'Unione (Germania, Francia, Inghilterra). Il dato è ancora più disarmante se si confrontano tali cifre con i costi sostenuti per una tutela brevettuale, sempre nel medesimo arco temporale di venti anni, negli Stati Uniti (14.556 USD) e nel Giappone (17.341YEN)²²⁴.

Svolti tali rilievi, non deve stupire il dato inerente ai soggetti che richiedono la registrazione delle proprie invenzioni. Il 66% delle richieste proviene infatti da grandi aziende, delle quali, circa il 50% sono extraeuropee. Solo il 28% delle registrazioni è effettuato da PMI o da persone fisiche²²⁵.

²²¹ *Ibidem*, cit., 6.

²²² *Ibidem*, cit., 7: "differenze possono dar luogo ad una caccia al tribunale più favorevole, nella quale le parti scelgono di avviare un'azione in una giurisdizione poiché riceveranno un miglior trattamento rispetto ad un'altra. Le differenze di costo e di rapidità dei procedimenti hanno un impatto significativo sulla scelta della giurisdizione".

²²³ *Ibidem*, cit., 8: "In Germania, il costo complessivo sostenuto da ciascuna parte in una vertenza in materia di brevetti del valore medio di circa 250 000 euro è stimato a circa 50 000 euro in primo grado e a 90 000 euro in appello, tanto per le azioni di nullità che di violazione. In Francia, il costo medio di una controversia in materia di brevetti del valore suddetto va da 50 000 euro a 200 000 euro in primo grado e tra 40 000 euro e 150 000 euro in appello. Nei Paesi Bassi, il medio costo stimato per una causa in materia di brevetti varia da 60 000 a 200 000 in primo grado e tra 40 000 euro e 150 000 euro in appello. Nei Paesi Bassi, il medio costo stimato per una causa in materia di brevetti varia da 60 000 a 200 000 in primo grado e tra 40 000 euro e 150 000 euro in appello. Nel Regno Unito il costo di una causa analoga può variare da 150 000 euro (procedura accelerata) a 1 500 000 euro in primo grado e da 150 000 euro a 1 000 000 euro in appello. Ciò significa che i costi cumulativi di una controversia parallela in questi quattro Stati membri varierebbero tra 310 000 euro e 1 950 000 euro in primo grado e tra 320 000 euro e 1 390 000 euro in secondo grado".

²²⁴ *Ibidem*, allegato I, p. 18, "Struttura dei costi inerenti al deposito delle domande di brevetto dirette e del mantenimento dei brevetti".

²²⁵ Dati forniti da Benoît Battistelli all' European Parliament's Legal Affairs Committee, 23 Marzo 2017. La presentazione in Rete: <http://www.uibm.gov.it/attachments/article/2008274/juri-hearing-unitary-patent-batistelli.pdf>.

2.2. (Segue): il brevetto unitario europeo

Le ragioni sopra addotte sono alla base del progetto di costituzione di un brevetto unitario europeo, che dovrebbe concretizzarsi in questo anno²²⁶.

Il percorso volto alla predisposizione di un sistema brevettuale unitario nell'Unione Europea è stato, è, e forse sarà, lungo e travagliato. Il 15 dicembre 1975, dopo complessi lavori preparatori, fu sottoscritta dai paesi membri della allora Comunità Europea, la Convenzione di Lussemburgo sul brevetto comunitario (CBC). La Convenzione statuiva che i brevetti rilasciati da EPO costituissero un titolo valido all'interno di tutta la CE. L'accordo, non è mai entrato in vigore, a causa di resistenze politiche di certi Stati come ad esempio Irlanda e Danimarca. Per superare queste frizioni il testo fu modificato e l'accordo fu nuovamente sottoscritto a Lussemburgo il 15 dicembre 1989. Neppure questa convenzione ricevette tuttavia le necessarie ratifiche per la sua entrata in vigore. Così, la Commissione si mise al lavoro per rilanciare il brevetto europeo attraverso uno strumento diverso dalla Convenzione, il Regolamento.

La difficoltà della uniformazione della materia risiede nella forte connotazione nazionale della tutela brevettuale. Infatti, i fallimenti dei sopracitati accordi mostrano come gli Stati siano restii a demandare ad una giurisdizione esterna la competenza sulla materia. Un ulteriore problema evidenziato durante le negoziazioni riguarda la lingua; ogni stato vorrebbe che la domanda di brevetto fosse depositata nella propria lingua, incrementando però a dismisura i costi di traduzione per ogni singolo brevetto²²⁷.

In un mondo globalizzato, tuttavia, l'Europa può avere un ruolo rilevante nell'economia mondiale solo se agisce come soggetto unico. La circolazione del prodotto brevettato è una questione di rilevanza internazionale ed è ancora più rilevante all'interno dell'Unione Europea, dove vige la libera circolazione delle merci.

Consentire l'acquisizione di una privativa nazionale a soggetti stranieri, e a costi per giunta bassi, è sicuramente un deterrente per quei paesi che a livello economico non ricoprono una posizione di supremazia²²⁸. A oggi, sembra tuttavia che i benefici portati dal brevetto unitario, soprattutto per le piccole e medie imprese europee, siano maggiori rispetto ai "danni" pronosticati per i singoli Stati nazionali. Soprattutto nel settore della tecnologia, dove molti protagonisti del mercato sono asiatici o statunitensi, se il costo di un brevetto europeo è alto, allora vi accederanno solo quelle grandi multinazionali. In questo modo lo stato "capitalizza" nell'immediato ma non tutela quei soggetti medio-piccoli, economicamente impossibilitati ad accedere alla tutela²²⁹.

²²⁶ Così ha dichiarato il 7 marzo 2017 il presidente dell'Ufficio europeo dei brevetti Benoit Battistelli a Bruxelles, pronosticando che il nuovo sistema potrebbe divenire operativo "dal primo dicembre prossimo". Fonte ANSA, in Rete: http://www.ansa.it/europa/notizie/rubriche/altrenews/2017/03/07/brevetto-unitario-ue-incognita-brexit-su-entrata-in-vigore_3fab216f-7e72-4578-9002-e8c4709c9d35.html.

²²⁷ V. VANZETTI, DI CATALDO, *Manuale di diritto industriale*, pp. 519-521.

²²⁸ V. *Ibidem*, pp. 521-526.

²²⁹ V. *infra*, riflessioni conclusive, par. 2.

I motivi che hanno portato gli Stati europei ad adottare il provvedimento in esame, desunti dalle criticità evidenziate nel precedente paragrafo, si rinviengono nei considerando dei regolamenti europei che sono alla base del progetto²³⁰.

In primo luogo si fa riferimento al mercato unico europeo, ed il brevetto unitario è considerato uno strumento rivolto alla internazionalizzazione europea delle imprese dell'Unione²³¹. I singoli diritti nazionali in tema di tutela brevettuale ostano infatti ad una produzione e distribuzione di prodotti secondo regole omogenee. Una tutela unitaria, inoltre, favorirà lo sviluppo scientifico e tecnologico dell'Unione, aiuterà le imprese, e aumenterà la competitività del mercato Europeo²³².

Gli obiettivi sono ambiziosi e, nel marzo di questo anno, sono stati esaminati dal presidente di EPO, in una relazione in seno alla commissione JURI al parlamento europeo.

Si definisce il brevetto unitario europeo come *“A simplified and cost-effective patent system for Europe”*²³³. La vera rivoluzione riguarderà infatti il profilo dei costi necessari per accedere alla tutela. Il presidente di EPO stima che con un importo inferiore a € 5.000 sarà possibile ottenere una protezione di durata decennale in tutti i 26 paesi membri aderenti all'accordo.

“Il totale complessivo da pagare per il suo mantenimento per l'intera durata di 20 anni ammonterà a poco più di EUR 35.500. Per fare un confronto, gli importi da corrispondere in base al sistema attuale negli stessi 25 stati membri arrivano a EUR 29.500 per i primi dieci anni e a quasi EUR 159.000 per l'intera durata di 20 anni. In altre parole, l'ammontare delle tasse attualmente

²³⁰ Le fonti normative sulle quali si fonda il brevetto unitario europeo sono: Il Regolamento (UE) n. 1257/2012 del Parlamento Europeo e del Consiglio del 17 dicembre 2012 relativo all'attuazione di una cooperazione rafforzata nel settore dell'istituzione di una tutela brevettuale unitaria; Il Regolamento (UE) n. 1260/2012 del Consiglio del 17 dicembre 2012 relativo all'attuazione di una cooperazione rafforzata nel settore dell'istituzione di una tutela brevettuale unitaria con riferimento al regime di traduzione applicabile; L'Accordo del Consiglio (2013/C 175/01) su un tribunale unificato dei brevetti. I documenti sono disponibili in Rete: <http://www.uibm.gov.it/index.php/brevetti/brevettare-all-estero/brevetto-unitario>.

²³¹ Regolamento 1257/2012, considerando n. 1: *“La creazione delle condizioni giuridiche che permettano alle imprese di adattare le loro attività nella produzione e nella distribuzione di prodotti attraverso i confini nazionali e che offrano loro maggiori scelte e opportunità contribuisce al conseguimento degli obiettivi dell'Unione di cui all'articolo 3, paragrafo 3, del trattato sull'Unione europea. È necessario che una protezione brevettuale uni forme nel mercato interno, o almeno una parte significativa dello stesso, figuri fra gli strumenti giuridici a disposizione delle imprese”*.

²³² Regolamento 1257/2012, considerando n. 4.: *“La tutela brevettuale unitaria favorirà il progresso scientifico e tecnologico e il funzionamento del mercato in terno rendendo l'accesso al sistema brevettuale più facile, meno costoso e giuridicamente sicuro. Essa migliorerà altresì il livello della tutela brevettuale rendendo possibile l'ottenimento di una protezione brevettuale uniforme ne gli Stati membri partecipanti e l'eliminazione dei costi e della complessità a beneficio delle imprese di tutta l'Unione. Essa dovrebbe essere disponibile per i titolari di un brevetto europeo, sia degli Stati membri partecipanti che degli altri Stati, indipendentemente dalla loro cittadinanza, residenza o luogo di stabilimento”*.

²³³ Benoît Battistelli, relazione innanzi al European Parliament's Legal Affairs Committee, 23 marzo 2017, cit., 20.

*approvato per il brevetto unitario nell'ambito della proposta "True Top 4" corrisponde a una riduzione del 78% rispetto alla situazione corrente"*²³⁴

Tale circostanza unitamente alla previsione di un sistema di tribunali unificato, volge sicuramente a favore delle realtà economiche medio-piccole, così come ai singoli individui ed alle università o enti di ricerca con budget ridotti.

L'accesso alla tutela sarà più facile, più efficace e sicuramente più competitivo su un piano internazionale.

Attraverso tale strumento con una unica procedura sarà possibile ottenere una tutela che si estende a tutti gli stati aderenti²³⁵. Ma l'innovazione non insiste solo nella fase di rilascio del brevetto, perché è anche prevista l'istituzione di un tribunale unificato che gestirà le controversie²³⁶.

Il brevetto europeo sarà rilasciato dall'EPO, si aggiungerà agli strumenti già esistenti e non si sostituirà alla tutela brevettuale nazionale. Esso infatti si fonda su una disciplina che statuisce il primato del diritto dell'Unione Europea. Le fonti sono quindi, oltre al diritto dell'Unione Europea, i regolamenti 1257/2012 UE e 1260/2012 UE, l'accordo del consiglio sull'istituzione del tribunale unificato, l'EPC e gli altri accordi internazionali applicabili ai brevetti e vincolanti per gli stati contraenti, ed infine solo in via residuale il diritto nazionale²³⁷.

Si verificherà quindi una uniformazione sia nel diritto applicabile, che nel sistema dei ricorsi. Tale circostanza, è bene ricordarlo, non si tradurrà in uno stravolgimento della normativa dell'EPC nonché della consolidata giurisprudenza dell'EPO. Infatti i regolamenti sopracitati riprendono la disciplina

²³⁴ EPO, Adozione di una struttura di tasse favorevole alle imprese per il brevetto unitario (c.d. "documento true top 4"), 24 giugno 2015, cit., 1. Disponibile in Rete: http://www.uibm.gov.it/attachments/article/2008274/Tasse%20brevetto%20unitario_%20italiano.pdf.

²³⁵ V. Regolamento 1257/2012, articolo 3, brevetto europeo con effetto unitario: "1. Un brevetto europeo concesso con la stessa serie di rivendicazioni con riguardo a tutti gli Stati membri partecipanti beneficia di un effetto unitario in detti Stati membri, a condizione che il suo effetto unitario sia stato registrato nel registro per la tutela brevettuale unitaria. Un brevetto europeo concesso con una diversa serie di rivendicazioni per diversi Stati membri partecipanti non beneficia dell'effetto unitario. 2. Un brevetto europeo con effetto unitario possiede un carattere unitario. Esso fornisce una protezione uniforme e ha pari efficacia in tutti gli Stati membri partecipanti. Esso può essere limitato, trasferito o revocato, o estinguersi unicamente in relazione a tutti gli Stati membri partecipanti. Esso può essere concesso in licenza in relazione all'intero territorio degli Stati membri partecipanti o ad una parte di esso. 3. L'effetto unitario di un brevetto europeo si considera come non prodotto nella misura in cui il brevetto europeo è stato revocato o limitato".

²³⁶ La necessità di un tribunale che eserciti una giurisdizione unificata è palesata nel Regolamento 1257/2012, considerando n. 25: "È essenziale istituire un tribunale unificato dei brevetti incaricato di giudicare le cause concernenti i brevetti europei con effetto unitario al fine di garantire il corretto funzionamento di tali brevetti, la coerenza della giurisprudenza e quindi la certezza del diritto, nonché l'efficienza dei costi per i titolari dei brevetti. È pertanto di fondamentale importanza che gli Stati membri partecipanti ratifichino l'accordo su un tribunale unificato dei brevetti conformemente alle rispettive procedure costituzionali e parlamentari nazionali e adottino le misure necessarie affinché tale tribunale divenga operativo quanto prima".

²³⁷ V. il capo IV dell'Accordo del consiglio su un tribunale unificato dei brevetti (2013/C 175/01), rubricato "primato del diritto dell'Unione e responsabilità degli stati contraenti", e il capo V rubricato "fonti del diritto e diritto sostanziale", entrambi nella parte I.

della convenzione europea sul brevetto. La novità risiede nell'effetto unitario che sarà possibile accordare al brevetto europeo in fase di registrazione.

Il sistema sarà operativo con l'entrata in funzione del tribunale unificato dei brevetti, che avrà la sede della divisione centrale a Parigi con sezioni a Londra e a Monaco. Ulteriori sedi locali potranno essere create sulla base del numero di richieste di registrazione che avverranno negli stati firmatari. È istituita anche una corte di appello nonché una cancelleria entrambe con sede in Lussemburgo. Infine è prevista l'istituzione di un centro di mediazione e arbitrato per i brevetti, con sede a Lubiana e a Lisbona.

Il nuovo sistema diverrà operativo quando saranno raggiunte almeno 13 ratifiche dei paesi UE coinvolti, ivi comprese le ratifiche di Francia, Germania e Regno Unito. Ad oggi sono intervenute le ratifiche di 12 stati, compresa quella della Francia²³⁸.

È d'obbligo tuttavia rammentare una circostanza che preoccupa gli operatori del settore nonché i commentatori, ossia l'inizio della procedura d'uscita del Regno Unito dalla Unione Europea, dopo la cosiddetta "*Brexit*", a seguito del referendum inglese del 23 giugno 2016.

Il ruolo dell'Inghilterra nell'accordo che ha portato, o meglio che porterà, all'adozione di un brevetto unitario è stato di fondamentale importanza. Infatti, oltre ad essere un paese firmatario, la sua ratifica è essenziale affinché si perfezioni il sopracitato accordo. Inoltre una sede del tribunale, secondo l'accordo del consiglio, dovrebbe essere locata a Londra. Il governo inglese ha tuttavia dichiarato, attenuando le preoccupazioni, l'intenzione di firmare la ratifica²³⁹. Se ciò non dovesse accadere, si dovrà trovare, un'altra soluzione. Si possono quindi presumere due strade; nella prima ipotesi la procedura di uscita farà salva la permanenza del Regno Unito all'interno dell'accordo del brevetto unitario, nella seconda ipotesi si avrà la completa uscita di scena del Regno Unito. Al momento non sembra prospettabile la circostanza che l'intero accordo non vedrà la luce.

Nonostante siamo dinanzi ad una innovazione profonda del sistema brevettuale nell'Unione Europea, risulta essere ancora attuale una analisi della disciplina dell'EPC, nonché della giurisprudenza dell'EPO. Il brevetto europeo ad effetto unitario, quando vedrà la luce, poggerà infatti sulle stesse basi di diritto sostanziale che vi sono oggi.

²³⁸ Risultano avere ratificato l'accordo Austria, Belgio, Bulgaria, Danimarca, Finlandia, Francia, Italia, Lussemburgo, Malta, Olanda, Portogallo, Svezia. Tra i 26 paesi coinvolti vi sono anche: Germania, Regno Unito, Polonia, Cipro, Grecia, Estonia, Lettonia, Lituania, Rep. Ceca, Rep. Slovacca, Slovenia, Irlanda, Romania, Ungheria. Da notare che la Spagna non ha aderito all'accordo.

²³⁹ Così ha dichiarato il ministro britannico per la Proprietà Intellettuale, Lucy Neville-Rolfe con una comunicazione formale al Consiglio Competitività a Bruxelles nel mese di novembre 2016. Fonte: ANSA, riportata in Rete: <http://www.ilsole24ore.com/art/impresa-e-territori/2016-11-28/londra-dice-si-brevetto-unitario-doccia-fredda-milano-183215.shtml?uuid=ADRM9J3B>.

2.3. EPC: normativa

Rilevato il ruolo centrale di EPO si deve ora analizzare la disciplina dell'EPC, nella parte in cui statuisce sulle tematiche di interesse ai fini del ragionamento.

Punto centrale del testo è l'articolo 52 rubricato invenzioni brevettabili²⁴⁰. Il primo comma stabilisce le condizioni generali affinché una invenzione possa essere oggetto di brevetto: deve essere nuova, implicare una attività inventiva ed essere suscettibile di una applicazione industriale. Ma cosa si intende per invenzione? Nonostante al comma uno si stabilisca come il brevetto europeo possa essere concesso in ogni campo tecnologico²⁴¹, il comma due prevede una serie di eccezioni ben specifiche²⁴². In particolare non si considerano invenzioni ai sensi della lettera a) i metodi matematici e ai sensi della lettera c) i programmi informatici. La limitazione non è quindi generica ma specifica, e rimanendo alla lettera della convenzione non sembra esservi spazio per una interpretazione diversa da quella fatta palese testo. Tuttavia, se tale ultimo comma sembra chiudere la porta alla brevettabilità dei programmi informatici, il comma terzo apre uno spiraglio. Si stabilisce infatti che i casi elencati dal comma due sono esclusi dalla brevettabilità se la domanda concerne *“uno solo di tali oggetti o attività, considerati come tali”*.

È quindi esclusa la brevettabilità del software in quanto tale, perché, secondo la convenzione, non implementa una invenzione, mentre non è esclusa la brevettabilità di una invenzione che implementi un software.

Il confine è sicuramente labile, ma una tale impostazione non deve sorprendere, alla luce di quanto si è analizzato nelle tendenze legislative e giurisprudenziali pre-convenzione. A livello internazionale vi era incertezza circa la tipologia di tutela più idonea all'oggetto software. Così la convenzione si è tradotta in una sintesi delle posizioni degli Stati firmatari, introducendo la dicitura *“come tali”* per fornire un criterio interpretativo volto a distinguere tra programmi brevettabili e non.²⁴³

Le *“non-invenzioni”*, di cui al secondo comma dell'articolo 52, sono accomunate dalla circostanza di essere entità astratte, ossia prive di quel

²⁴⁰ Art. 52, Invenzioni brevettabili: *“1. I brevetti europei sono concessi per le invenzioni in ogni campo tecnologico, a condizione che siano nuove, implicino un'attività inventiva e siano atte ad avere un'applicazione industriale. 2. Non sono considerate invenzioni ai sensi del paragrafo 1 in particolare: a) le scoperte, le teorie scientifiche e i metodi matematici; b) le creazioni estetiche; c) i piani, principi e metodi per attività intellettuali, per giochi o per attività commerciali, come pure i programmi informatici; d) le presentazioni di informazioni. 3. Il paragrafo 2 esclude la brevettabilità degli oggetti o delle attività che vi sono enumerati soltanto nella misura in cui la domanda di brevetto europeo o il brevetto europeo concerna uno solo di tali oggetti o attività, considerati come tali”*.

²⁴¹ La specificazione *“in ogni campo tecnologico”* fu aggiunta nell'ultima modifica alla convenzione del 29 novembre 2000.

²⁴² Eccezioni che tuttavia non formano un gruppo omogeneo, ma sono individuate in quattro sottoinsiemi. La logica alla base dell'escussione operata sulla base di tali sottoinsiemi non è chiara. Così VANZETTI, DI CATALDO, *Manuale di diritto industriale*, pp. 333 ss.

²⁴³ V. GATTEI, *Il brevetto di software*, cit. pp. 40-41.

carattere tecnico necessario per una applicazione industriale²⁴⁴. Tale caratteristica *“fungerebbe da discrimen tra le creazioni industriali ed intellettuali, essendo queste ultime deputate esclusivamente alla fruizione estetica ed astratta”*²⁴⁵.

L'articolo 53 della Convenzione, rubricato eccezioni alla brevettabilità²⁴⁶, è interessante nella parte in cui non fa alcun riferimento ai programmi informatici. La norma statuisce infatti che non vengono concessi brevetti europei sulle innovazioni contrarie all'ordine pubblico e al buon costume, sulle innovazioni che insistono su varietà vegetali e razze animali e sulle innovazioni riguardanti trattamenti chirurgici o terapeutici sul corpo umano o animale.

L'articolo 54 introduce il requisito della novità²⁴⁷. Una invenzione per essere brevettabile deve essere nuova, ossia non deve essere compresa nello stato della tecnica. Per stato della tecnica si intende ciò che è accessibile, fruibile, consultabile dal pubblico ovvero invenzioni già depositate nell'ufficio ancorché non ancora pubblicate. Si escludono quindi le invenzioni già conosciute, e quelle in fase di registrazione.

²⁴⁴ V. GUGLIELMETTI, *L'invenzione di software. Brevetto e diritto d'autore*, cit., 31, che individua nella dottrina tedesca del secolo XIX un elemento di influenza a tale circostanza. Tale dottrina (*kerntheorie*) incentra nel possesso del carattere tecnico il punto fondamentale per l'accesso ad una tutela brevettuale. L'invenzione era identificata come la combinazione di mezzi fisici, che attraverso l'utilizzo di forze naturali, ottenevano un certo risultato. L'algoritmo di software alla luce di una tale visione risultava privo del carattere tecnico perché *“concepito come una procedura astratta per la soluzione di un problema, la cui validità e risultato prescindono dall'impiego diretto di forze dalla natura, e nella quale l'uso dei mezzi materiali è dunque successivo al completamento dell'idea inventiva riguardando soltanto uno dei suoi possibili modi di attuazione”*.

²⁴⁵ E. AREZZO, *La brevettabilità del software e dei metodi commerciali elettronici nella giurisprudenza dell'Ufficio Europeo Brevetti*, Assonime, 12/2009, cit., 15. Paper in Rete: https://www.tuttocamere.it/files/studi/2009_12_Assonime.pdf.

²⁴⁶ Art. 53, Eccezioni alla brevettabilità: *“Non vengono concessi brevetti europei per: a) le invenzioni il cui sfruttamento commerciale sarebbe contrario all'ordine pubblico o al buon costume; tale contrarietà non può essere dedotta dal solo fatto che lo sfruttamento è vietato da una disposizione legale o amministrativa in tutti gli Stati contraenti o in parte di essi; b) le varietà vegetali o le razze animali come pure i procedimenti essenzialmente biologici per l'ottenimento di vegetali o di animali. Tale disposizione non si applica ai procedimenti microbiologici e ai prodotti ottenuti mediante questi procedimenti; c) i metodi di trattamento chirurgico o terapeutico del corpo umano o animale e i metodi di diagnosi applicati al corpo umano o animale. Tale disposizione non si applica ai prodotti, in particolare sostanze o miscele di sostanze, destinati all'utilizzazione in uno di questi metodi”*.

²⁴⁷ Art. 54, Novità: *“1. Un'invenzione è considerata nuova se non è compresa nello stato della tecnica. 2. Lo stato della tecnica è costituito da tutto ciò che è stato reso accessibile al pubblico prima della data di deposito della domanda di brevetto europeo mediante una descrizione scritta o orale, un'utilizzazione o un qualsiasi altro mezzo. 3. È pure considerato compreso nello stato della tecnica il contenuto, secondo il testo depositato in principio, di domande di brevetto europeo che hanno una data di deposito anteriore a quella citata nel paragrafo 2 e sono state pubblicate soltanto in questa data o più tardi. 4. I paragrafi 2 e 3 non escludono la brevettabilità di una sostanza o di una miscela di sostanze appartenente allo stato della tecnica per l'utilizzazione in uno dei metodi di cui all'articolo 53 lettera c, a condizione che la sua utilizzazione in uno qualsiasi di questi metodi non sia compresa nello stato della tecnica. 5. I paragrafi 2 e 3 non escludono neppure la brevettabilità di una sostanza o di una miscela di sostanze di cui al paragrafo 4 per qualsiasi utilizzazione specifica in ognuno dei metodi di cui all'articolo 53 lettera c, a condizione che tale utilizzazione non sia compresa nello stato della tecnica”*.

L'articolo 56 illustra l'ulteriore caratteristica, già richiamata dall'articolo 52 affinché una innovazione possa essere considerata brevettabile: deve implicare una attività inventiva²⁴⁸.

Il requisito è adempiuto nel momento in cui una persona esperta nel campo specifico in cui l'invenzione insiste, ritenga che essa, allo stato della tecnica, non sia evidente. È bene chiarire tale requisito perché è possibile la sua sovrapposizione con la caratteristica della novità. Quest'ultima rappresenta una valutazione di carattere oggettivo: non potrà essere concesso un brevetto per una invenzione nota, in quanto già brevettata in precedenza, ovvero depositata e in attesa di registrazione. Il requisito di cui all'articolo 56 richiede invece una valutazione circa l'effettivo contributo che l'innovazione porta in quel determinato ambito. Si vogliono quindi escludere i brevetti su tecniche note, palesi o di uso comune ancorché non brevettate. Deve sussistere una concreta attività inventiva. Potrebbe quindi ben palesarsi la situazione che una invenzione pur potendosi considerare nuova, non integri invece il requisito della inventività.

L'ultimo requisito necessario, affinché l'invenzione possa essere brevettata, risiede nella sua idoneità ad avere applicazione industriale. Ai sensi dell'articolo 57 ciò si palesa nel momento in cui l'oggetto dell'invenzione è atto all'utilizzo o alla fabbricazione industriale, considerata in ogni sua tipologia di genere²⁴⁹.

Alla luce di tale normativa, la valutazione circa la brevettabilità di un software necessita di un esame caso per caso. I maggiori problemi interpretativi si hanno con riferimento all'articolo 52. È infatti indispensabile una specifica e concreta valutazione per differenziare un software non brevettabile (perché considerato di per sé) da un software brevettabile (perché costituisce invero una invenzione che implementa un software).

Importanza centrale ebbe quindi la Commissione di Ricorso dell'Ufficio Europeo dei Brevetti, il cui contributo fu fondamentale per delineare una certa linea interpretativa e rappresentando al contempo un fermo punto di riferimento per gli stati dell'Unione Europea.

Una volta depositata la domanda di registrazione del brevetto, e qualora questa sia stata accolta e pubblicata, i diritti conferiti dal brevetto europeo, sono i medesimi concessi dal brevetto nazionale per il quale è stata richiesta e pubblicata la registrazione²⁵⁰. I diritti conferiti dal brevetto europeo variano

²⁴⁸ Art. 56, Attività inventiva: *“Un'invenzione è considerata come implicante un'attività inventiva se, per una persona del mestiere, essa non risulta in modo evidente dallo stato della tecnica. Se lo stato della tecnica comprende documenti di cui all'articolo 54, paragrafo 3, questi documenti non sono presi in considerazione per l'apprezzamento dell'attività inventiva”*.

²⁴⁹ Art. 57, Applicazione industriale: *“Una invenzione è considerata atta ad avere un'applicazione industriale se il suo oggetto può essere fabbricato o utilizzato in un qualsiasi genere di industria, compresa l'agricoltura”*.

²⁵⁰ Art. 64, Diritti conferiti dal brevetto europeo: *“1. Salvo il paragrafo 2, il brevetto europeo conferisce al suo titolare, a decorrere dal giorno di pubblicazione della menzione della sua concessione e in ciascuno Stato contraente per il quale è stato concesso, i medesimi diritti che gli conferirebbero un brevetto nazionale concesso in questo Stato. 2. Se l'oggetto del brevetto europeo è un procedimento, i diritti conferiti da questo brevetto si estendono ai prodotti ottenuti direttamente mediante questo procedimento. 3. Ogni contraffazione del brevetto europeo è apprezzata conformemente alle disposizioni della legislazione nazionale”*.

quindi a seconda degli stati nazionali presso cui il brevetto viene registrato. A livello temporale tuttavia la convenzione statuisce che la durata del brevetto europeo è di venti anni, salva tuttavia la facoltà degli stati di prorogare tale termine conformemente alla legislazione nazionale²⁵¹, e in determinati casi eccezionali.

2.4. Le Linee Guida: contributo tecnico, programma per elaboratore e invenzione attuata a mezzo di elaboratore elettronico

Un elemento molto rilevante nell'analisi del brevetto di software in Europa, si fonda sulla interpretazione del concetto di programma per elaboratore considerato "in quanto tale" e di invenzione attuata a mezzo di programma per elaboratore. Per dare una corretta interpretazione alla lettera della convenzione è necessario rifarsi alle linee guida dettate da EPO.

Le Linee Guida furono pubblicate il 1° giugno 1978, e hanno il fine di fornire determinati criteri, basati sulle norme della convenzione, attraverso cui eterodeterminare l'operato degli uffici.

Con riguardo alla brevettabilità di una invenzione, le Linee Guida richiedono che abbia le caratteristiche della concretezza e della tecnicità.

Si esclude quindi, attraverso una interpretazione letterale dell'articolo 52, la brevettabilità di quelle entità che ex art. 52, comma 2, sono da considerarsi astratte e prive di carattere tecnico²⁵².

Ma cosa si intende per carattere tecnico? Stando alle Linee Guida: *"the concept of technical character implies that a patentable invention must relate to tecnica field, must be concerne with a technical problem and must be defined in the claim by technical features"*²⁵³.

Applicando tali concetti al software, rinveniamo una rigida interpretazione. Si stabilisce infatti che, a prescindere dal contenuto, un programma per elaboratore non è brevettabile in quanto tale²⁵⁴. Non solo; la brevettabilità è

²⁵¹ Art. 63, Durata del brevetto europeo: "1. La durata del brevetto europeo è di venti anni a decorrere dalla data di deposito della domanda. 2. Il paragrafo 1 non pregiudica il diritto di uno Stato contraente di prorogare la durata di un brevetto europeo o di concedergli corrispondente protezione alla stessa stregua di quella dei brevetti nazionali allo scadere di detto termine, a) per tenere conto di un caso di guerra o di crisi simile di questo Stato; b) se l'oggetto del brevetto europeo è un prodotti o un procedimento di fabbricazione o un'utilizzazione di un prodotto che prima della sua messa in circolazione in questo Stato sottostà a una procedura amministrativa di autorizzazione prescritta per legge. 3. Le disposizioni di cui al paragrafo 2 sono applicabili per analogia ai brevetti europei rilasciati congiuntamente da gruppi di Stati contraenti ai sensi dell'articolo 142. 4. Ogni Stato contraente che prevede una proroga della durata di un brevetto o una protezione corrispondente conformemente al paragrafo 2 lettera b può, in base a un accordo concluso con l'Organizzazione, trasferire all'Ufficio europeo dei brevetti compiti relativi all'applicazione di dette disposizioni".

²⁵² European Patent Office, *Guidelines*, 1978 cit., cap. IV, 2.3, "An invention must be both a concrete and a technical character, in contrast to the list of things in Art. 52(2) with shall not be regarded as invention since they are either abstract or non-technical".

²⁵³ *Ibidem*, cit., cap. IV, 2.3.

²⁵⁴ *Ibidem*, cit., cap. IV, 2.3: "a computer program claimed by itself or as a record on a carrier is unpatentable irrespective of its content".

esclusa qualora quel programma venga semplicemente caricato su un computer noto allo stato dell'arte.²⁵⁵

Tuttavia, qualora il programma per elaboratore, fornisca, unitamente al sistema a cui accede, un determinato contributo tecnico allo stato dell'arte, allora l'invenzione, in questo senso considerata, -ossia l'invenzione di software- può essere brevettata²⁵⁶.

Nel 1985 l'Ufficio Brevetti pubblicò il primo documento di revisione per le linee guida che le Divisioni d'Esame dovevano seguire durante la valutazione e l'analisi delle domande sui brevetti. Il documento, confermando la posizione già adottata dall'ufficio, stabilisce la brevettabilità dell'invenzione di software, quando lo stesso è in grado di fornire un contributo tecnico allo stato dell'arte (c.d. *contribution approach*).

Tale verifica va effettuata anche sulla base del c.d. "*problem-solution approach*"²⁵⁷. Si deve sottolineare come tale criterio è utile per sottoporre l'innovazione al vaglio dell'originalità. Per valutarne la aderenza all'interno del *patentable subject matter*, il criterio di riferimento è invece il *contribution approach*²⁵⁸.

Se il problema viene risolto attraverso strumenti tecnici, e non meramente astratti (come ad esempio l'applicazione di formule matematiche), tali da portare un effettivo contributo allo stato della tecnica, allora la richiesta di brevettabilità può essere presa in considerazione.

Lo "strumento tecnico" richiesto, consta di un sistema tecnologico complesso di hardware e software, dove il "nuovo programma" o il "nuovo algoritmo", siano stati ideati al fine di raggiungere un determinato risultato²⁵⁹. Quando il software è in grado di controllare un apparato informatico, allora il sistema integrato sarà brevettabile. Questa importante revisione delle linee guida rese possibile nello stesso anno la decisione del caso Vicom, di cui si parlerà nel capitolo successivo²⁶⁰.

Nel 1994, fu presentato il rapporto annuale che rappresentava la posizione ufficiale di EPO in materia di brevetti su programmi per elaboratore. Relativamente all'esclusione dalla tutela brevettuale del software in quanto tale si registra che: "*EPO is thus debarred from granting patents for computer programs as such; they can be protected by copyright. On the other hand, the EPC in no way prohibits the patenting of computer-related inventions or, more*

²⁵⁵ *Ibidem*, cit., IV, 2.3: "*The situation is not normally changed when the computer program is loaded into a known computer*".

²⁵⁶ *Ibidem*, cit., IV, 2.3: "*Where subject-matter claimed is only concerned with the program-controlled internal working of a known computer, the subject matter could be patentable if it provides a technical effect*" e ancora, "*If the subject matter, considered as a whole makes a technical contribution to the known art, patentability should not be denied merely on the ground that a computer program is involved in its implementation*".

²⁵⁷ V. European Patent office, *Annual Report*, 1985, pp.11-12.

²⁵⁸ V. *infra*, cap. IV, par. 2.3

²⁵⁹ EPO, *Annual Report*, 1985, cit., 12: "*Programmable controllers and manufacturing or monitoring systems operating under computer control can therefore generally patented. This is also the case when the internal operation of a known computer is controlled by a program so that a technical effect results. Crucial to this approach is that the inventive achievement may be based on a new program or algorithm*".

²⁶⁰ V. *infra*, cap. IV, par. 2.2.

*precisely, of methods, systems, devices or products in which new computer software plays an important role*²⁶¹.

Si denota quindi come la tutela brevettuale venga concessa al prodotto tangibile, frutto di una interazione tra macchina e software. Nel contesto brevettuale europeo, quando si parla di “invenzione di software” non si intende il “software” o “il programma per elaboratore” nel senso lato del termine²⁶² ma un “trovato più complesso in cui il programma è una componente centrale”²⁶³.

Quindi; qualora il programma per elaboratore venisse classificato “in quanto tale”, alla luce della convenzione sarebbe privo di una concreta applicazione pratica ovvero di un determinato risultato tecnico, e non sarebbe quindi brevettabile²⁶⁴.

L'applicazione concreta richiede la partecipazione di un agente esterno, come ad esempio un determinato supporto hardware, in grado di implementare il contributo astratto in un risultato ora concreto e tangibile²⁶⁵.

Similmente a quanto accadeva in Inghilterra prima della Convenzione, la tutela del software è ottenuta mediante la rappresentazione del legame che si instaura con la componente hardware²⁶⁶. La sinergia tra le componenti è brevettabile perché idonea ad apportare un contributo tecnico²⁶⁷.

Con il progresso tecnologico, si deve preliminarmente rilevare come sia cambiata l'interpretazione della normativa dell'EPC, grazie all'impulso della giurisprudenza dell'EPO. Di conseguenza sono anche cambiate le linee guida fornite dall'ufficio brevetti. Quelle fornite oggi dall'Ufficio²⁶⁸, sebbene nella loro essenza non si discostino da quelle sopracitate, aggiungono sicuramente elementi di grande interesse, che tuttavia, stante appunto la loro origine giurisprudenziale, verranno analizzati nel proseguo del lavoro²⁶⁹.

Si noti tuttavia come sia ancora presente il requisito del “contributo tecnico”, che il software può portare solo nel momento in cui “gira” su una

²⁶¹ European Patent Office, *Patenting Computer Software*, in Annual Report, 1994, cit., 10.

²⁶² Ossia il programma per elaboratore, espresso in qualsivoglia forma, come ad esempio in codice sorgente od oggetto. Questo, se originale, può essere tutelato mediante diritto d'autore, in conformità della legge.

²⁶³ AREZZO, *La brevettabilità del software e dei metodi commerciali elettronici nella giurisprudenza dell'Ufficio Europeo Brevetti*, cit., 11.

²⁶⁴ Sarebbe comunque un programma per elaboratore, e, come tale, suscettibile di una tutela autoriale.

²⁶⁵ Interpretazione maggiormente accolta. V. GUGLIELMETTI, *L'invenzione di software. Brevetto e diritto d'autore*; GATTEI, *Il brevetto di software, guida teorico pratica*; AREZZO, *La brevettabilità del software e dei metodi commerciali elettronici nella giurisprudenza dell'Ufficio Europeo Brevetti*; AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*.

²⁶⁶ LEITH, *Software and patents in Europe*, pp. 22-26.

²⁶⁷ Emblematica in tal senso la decisione del caso Schlumberger del 15 giugno 1981 (Cour d'Appel de Paris, in Ann. 1982, 24). Pochi anni dopo il caso Mobil Oil, la corte, stante l'entrata in vigore della Convenzione concesse la brevettabilità di “un procedimento per l'analisi morfologica del sottosuolo”, che l'azienda aveva sviluppato per la ricerca di giacimenti petroliferi. Il sistema comprendeva al suo interno l'utilizzo di un programma per elaboratore, ma la corte interpretò l'articolo 7 della legge francese in maniera restrittiva, in linea con la EPC.

²⁶⁸ Guidelines for Examination in the EPO, Novembre 2016, Parte G, Capitolo II. In Rete: <https://www.epo.org/law-practice/legal-texts/guidelines.html>.

²⁶⁹ V. *infra*, cap. IV, par. 2. ss.

macchina²⁷⁰. “L’invenzione di software”, o meglio una “computer-implemented invention”²⁷¹ può anche consistere in “un programma considerato in quanto tale”, qualora lo stesso, implementato su un sistema hardware standard, permetta appunto di realizzare quel particolare contributo tecnico.²⁷²

Se il software -inteso in quanto tale, nella sua accezione di “copyrightable subject matter”- è, in molti casi, una “computer-implemented invention”, quest’ultima solo in determinati casi, filtrati dalla giurisprudenza EPO, si esaurisce nel programma²⁷³.

3. Le convenzioni internazionali sui brevetti

Con l’approvazione della EPC, alcuni Stati europei adottarono la già citata Convenzione di Strasburgo sulla unificazione dei principi in materia di brevetti. La Convenzione è interessante nella parte in cui, prevedendo una lista di eccezioni tassative alla brevettabilità, non cita i programmi per elaboratore. Questi, quindi, potranno essere tutelati qualora sottendano ai requisiti generali richiesti per la registrazione del brevetto: novità, attività inventiva, suscettibilità di applicazione industriale, ricalcando le condizioni richieste ex articolo 52 EPC.

Una ulteriore fonte internazionale è l’accordo TRIPs²⁷⁴, amministrato oggi dalla WTO²⁷⁵, concluso al termine del negoziato “Uruguay Round”²⁷⁶, e adottato il 15 aprile 1994 con la firma degli accordi di Marrakech.

L’accordo, similmente alla EPC ed alla convenzione di Strasburgo, conferma per le invenzioni i requisiti della novità, della attività inventiva e della industrialità. Molto interessante è l’articolo 27 che prevede la possibilità per gli stati aderenti di escludere in via facoltativa, la brevettabilità di determinati oggetti.

²⁷⁰ *Ibidem*, cit., 6: “Likewise, although it may be said that all computer programming involves technical considerations since it is concerned with defining a method which can be carried out by a machine, that in itself is not enough to demonstrate that the program which results from the programming has technical character; the programmer must have had technical considerations beyond “merely” finding a computer algorithm to carry out some procedure (G 3/08)”.

²⁷¹ *Ibidem*, cit., 5: “an expression intended to cover claims which involve computers, computer networks or other programmable apparatus whereby prima facie one or more of the features of the claimed invention are realised by means of a program or programs”.

²⁷² *Ibidem*, cit., 5: “A computer program claimed by itself is not excluded from patentability if it is capable of bringing about, when running on or loaded into a computer, a further technical effect going beyond the “normal” physical interactions between the program (software) and the computer (hardware) on which it is run (T 1173/97 and G 3/08)”.

²⁷³ In questo senso anche AREZZO, *La brevettabilità del software e dei metodi commerciali elettronici nella giurisprudenza dell’Ufficio Europeo Brevetti*, pp. 10-11.

²⁷⁴ Accordo sugli aspetti commerciali dei diritti di proprietà intellettuale (The Agreement on Trade Related Aspects of Intellectual Property Rights). Il testo in Rete: http://www.uibm.gov.it/attachments/Accordo_trips.pdf.

²⁷⁵ Organizzazione mondiale del commercio (World Trade Organization).

²⁷⁶ Ciò fu il frutto di un negoziato di quasi 8 anni che ha riguardato 123 paesi, portando alla creazione della WTO e alla ratifica di tre accordi principali, tra cui appunto il TRIPs.

La lista delle eccezioni alla brevettabilità non cita tuttavia il software²⁷⁷. Dei programmi per elaboratore viene fatta esplicita menzione nell'articolo 10²⁷⁸. Si stabilisce che essi siano protetti alla stregua di opere letterarie qualora siano rappresentati in codice sorgente ovvero in codice oggetto ovvero in una altra forma espressiva tale da poter essere classificata alla stregua di creazione intellettuale.

L'accordo TRIPs, quindi, non esclude in modo esplicito il software dall'alveo delle invenzioni brevettabili²⁷⁹, sempre che lo stesso soddisfi i criteri generali per il suo rilascio, pur tuttavia riconoscendo come punto fermo la tutela mediante diritto d'autore del c.d "computer program".

In conformità con la Convenzione di Berna, l'opera viene tutelata qualora originale, frutto della creazione intellettuale dell'autore e *"la protezione del diritto d'autore copre le espressioni e non le idee, i procedimenti, i metodi di funzionamento o i concetti matematici in quanto tali"*²⁸⁰.

Una tale posizione compromissoria non deve sorprendere, essendo l'accordo in oggetto una sintesi tra le diverse posizioni di molti paesi aderenti. L'obiettivo dell'accordo è infatti quello di fornire una base comune per la concessione di una tutela minima, riconosciuta in tutti gli stati firmatari²⁸¹.

Similmente all'art. 10 dell'accordo TRIPs, statuisce il WIPO Copyright Treaty (WCT)²⁸², adottato nel dicembre del 1996 e che si innesta sulle basi già poste dalla convenzione di Berna²⁸³. L'articolo 4 stabilisce infatti che *"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"*²⁸⁴.

La tendenza dei trattati internazionali è quella di individuare il software come bene tutelabile dal diritto d'autore, lasciando tuttavia aperta la strada alla

²⁷⁷ TRIPs, art. 27, comma 3. *"I Membri possono inoltre escludere dalla brevettabilità: a) i metodi diagnostici, terapeutici e chirurgici per la cura dell'uomo o degli animali; b) i vegetali e gli animali, tranne i microorganismi, e i processi essenzialmente biologici per la produzione di vegetali o animali, tranne i processi non biologici e microbiologici"*.

²⁷⁸ TRIPs, art. 10, Programmi per elaboratore e compilazioni di dati *"1. I programmi per elaboratore, in codice sorgente o in codice oggetto, sono protetti come opere letterarie ai sensi della Convenzione di Berna (1971). 2. 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"*.

²⁷⁹ Conclusione cui giunge anche GATTEI, *Il brevetto di software, guida teorico pratica*, cit., 45, stante la non menzione delle c.d invenzioni di software.

²⁸⁰ TRIPs, articolo 9, *"Rapporto con la convenzione di Berna"*.

²⁸¹ VANZETTI, DI CATALDO, *Manuale di diritto industriale*, pp. 512-514.

²⁸² Trattato World Intellectual Property Organization (WIPO) o Organizzazione Mondiale Proprietà Intellettuale (OMPI). Il testo in Rete: http://www.aesvi.it/cms/attach/editor/INTERNAZIONALE_01.pdf.

²⁸³ La Convenzione di Berna per la protezione delle opere letterarie e artistiche, è stata, ed è il primo e fondamentale accordo internazionale sul reciproco riconoscimento tra gli stati aderenti del diritto d'autore. Fu adottata a Berna nel 1886 e a oggi i paesi aderenti sono 164. Il testo in Rete: <https://www.admin.ch/opc/it/classified-compilation/19710188/201402070000/0.231.15.pdf>

²⁸⁴ WIPO Copyright Treaty, art. 4.

sua brevettabilità, non rientrando tra quei beni o trovati *tout court* esclusi dal panorama brevettabile²⁸⁵.

4. Legislazione dell'Unione Europea

4.1 Le direttive europee e la tutela giuridica dei programmi per elaboratore (cenni)

Le fonti normative della tutela del software nell'Unione Europea si rinvengono in tre direttive. La prima, in ordine cronologico, è la direttiva 91/250/CEE²⁸⁶, sostituita nel 2009 dalla direttiva 2009/24/CE²⁸⁷. I contenuti riprodotti sono sostanzialmente analoghi e vertono sulla tutela giuridica dei programmi per elaboratore. Ulteriore direttiva molto importante è la 2001/29/CE²⁸⁸, sull'armonizzazione di taluni aspetti del diritto d'autore e dei diritti connessi nella società dell'informazione.

Si è detto che la protezione della proprietà intellettuale muove dalla necessità di assicurare un incentivo alla creatività. Incentivo finalizzato allo sviluppo generale della cultura nell'intera società²⁸⁹. Per salvaguardare tale obiettivo il legislatore europeo ha deciso di proteggere il software mediante il diritto d'autore, accostando il programma per elaboratore alle opere letterarie, e tutelandolo ai sensi della convenzione di Berna²⁹⁰.

La tutela è automatica, non richiede alcuna forma di registrazione, ha una durata limitata nel tempo²⁹¹ ed è rivolta all'autore dell'opera²⁹².

Sostanzialmente la tutela copre l'espressione dell'idea qualora la stessa sia originale e non si estende alle idee e i principi che sono alla base dell'opera²⁹³.

Dal punto di vista sostanziale il legislatore europeo prevede una gamma di eccezioni e limitazioni che si contrappongono alle attività riservate²⁹⁴.

²⁸⁵ In questo senso anche S. MAGELLI, *Innovazione informatica e diritti di brevetto*, in GALLI, *Le nuove frontiere del diritto dei brevetti*, pp. 76-77.

²⁸⁶ Direttiva 91/250/CEE del Consiglio, del 14 maggio 1991, relativa alla tutela giuridica dei programmi per elaboratore. In Rete: <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31991L0250:IT:HTML>.

²⁸⁷ Direttiva 2009/24/CE del parlamento europeo e del consiglio del 23 aprile 2009 relativa alla tutela giuridica dei programmi per elaboratore. In Rete: <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2009:111:0016:0022:IT:PDF>.

²⁸⁸ Direttiva 2001/29/CE del parlamento e del consiglio del 22 maggio 2001 sull'armonizzazione di taluni aspetti del diritto d'autore e dei diritti connessi nella società dell'informazione. In Rete: <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2001:167:0010:0019:IT:PDF>.

²⁸⁹ Il punto emerge chiaramente nei considerando delle direttive europee. In particolare V. considerando n. 9-12 direttiva 2001/29/CE.

²⁹⁰ V. Direttiva 2009/24/CE, art. 1 comma 1.

²⁹¹ V. Convenzione di Berna, art. 7, comma 1, che recita *“La durata della protezione concessa dalla presente Convenzione comprende la vita dell'autore e un periodo di cinquanta anni dopo la sua morte”*.

²⁹² L'autore, può essere una persona fisica o giuridica, v. Direttiva 2009/24/CE, art. 3.

²⁹³ V. Direttiva 2009/24/CE, considerando n.11; art. 1, comma 3. Sulla dicotomia espressione nei programmi per elaboratore, v. *supra*, cap. II, par. 1.

²⁹⁴ V. Direttiva 2009/24/CE, artt. 14 e 15.

Se, quindi, la tutela sembra essere bilanciata da un punto di vista sostanziale, altrettanto non si può dire per la sua estensione nel tempo. Un'estensione temporale che va oltre la vita dell'autore non sembra essere ideale sotto più punti di vista.

Nell'era digitale l'innovazione è in continua crescita ed espansione. La "vita" di un software non può essere paragonata a quella di una opera letteraria. La durata della prima può infatti esaurirsi dopo soli pochi anni a causa della velocità nell'innovazione e del progresso.

Secondariamente, è paradossale legare la durata della tutela alla vita dell'autore perché, nella maggior parte dei casi, lieve è il collegamento tra il titolare dei diritti e l'opera. Non sussiste infatti un legame intimo, paragonabile a quello che sussiste tra uno scrittore e la sua opera letteraria. Un software, infatti, il più delle volte è creato da più soggetti sotto le dipendenze di una persona giuridica.

Il diritto d'autore è una tutela pensata per creare un equo bilanciamento in relazione ad altre opere, su tutte quelle letterarie. La lunga durata del diritto deriva dal legame intimo tra creatore e creazione e poco si addice alla nuova concezione di tempo dell'era digitale.

Le criticità dello strumento del copyright non si esauriscono in tale circostanza, ma affondano anche e soprattutto nella incoerenza della tutela applicata al programma per elaboratore. Circostanza questa rilevabile e su di un livello sostanziale, stante l'eccessivo grado di tutela, e su di un livello teorico, stante l'impraticabilità del principio della dicotomia idea espressione²⁹⁵

4.2. Il fallimento della direttiva relativa alla brevettabilità delle invenzioni attuate per mezzo di elaboratori elettronici

Nonostante l'EPO, con le sue decisioni, abbia con il tempo tracciato una certa direzione, volta ad includere in certi casi il software nell'ambito della tutela brevettuale, il legislatore prima comunitario, poi europeo, ha scelto, una strada diversa.

Il processo di armonizzazione della legislazione europea nell'ambito della tutela dei programmi per elaboratore, è andato nella chiara direzione di tutelare il software alla stregua di una opera letteraria, ossia per mezzo del diritto d'autore, così come stabilito dalla convenzione di Berna²⁹⁶.

La medesima strada è stata percorsa dagli accordi internazionali, laddove si è stabilito che la tutela minima da concedere al software fosse appunto quella del copyright²⁹⁷.

Si deve ora notare come la situazione normativa sopra esposta nasce da una visione sostanzialmente diversa della materia da parte dell'Unione Europea e dell'EPO. La prima, cristallizza il fatto che la tutela del software debba

²⁹⁵ V. *supra*, cap. II, par. 1; per una disamina approfondita di tali criticità v. AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, pp. 25-69.

²⁹⁶ Si veda su tutti la Direttiva 91/250/CEE, art. 1.

²⁹⁷ Si veda su tutti l'Accordo TRIPs, art. 10.

avvenire per mezzo del copyright, mentre, le linee guida dettate dall'ufficio brevetti, unitamente alla giurisprudenza²⁹⁸, tende in determinati casi a concedere la tutela brevettuale.

È necessario ricordare che l'EPOrg è una organizzazione internazionale autonoma e distinta dall'UE, la quale, assieme ad altre organizzazioni internazionali, come ad esempio la WIPO, ne è un membro osservatore presso il consiglio amministrativo.

Tale indipendenza ha permesso all'EPO di divenire una organizzazione rilevante nonché di sviluppare un proprio orientamento nella materia.

Nel 2002, la commissione europea propose una direttiva relativa alla brevettabilità delle invenzioni attuate per mezzo di elaboratori elettronici²⁹⁹.

Il dibattito intorno alla proposta fu lungo e diede adito a molte controversie. Per comprendere le argomentazioni pro o contro la stessa, ritengo che sia utile analizzare il contenuto e gli obiettivi che si prefiggeva la proposta.

L'obiettivo principale della direttiva era quello di armonizzare la tutela mediante brevetto dei programmi per elaboratore, cristallizzando quella che era la posizione *de facto* dell'EPO, e fissando il concetto di "contributo tecnico", all'interno di una legge. Allo stesso tempo, la commissione mirava a concedere un forte incentivo economico alle aziende produttrici di software, attraverso una emulazione del modello statunitense, che, in quegli anni, aveva già "virato" verso una tutela del software anche attraverso il brevetto³⁰⁰.

I più vivaci oppositori della direttiva facevano leva fundamentalmente su due ragioni.

²⁹⁸ V. *infra*, cap. IV.

²⁹⁹ Proposta 2002/0047 del 20 febbraio 2002, presentata dalla Commissione Europea con relatrice Arlene McCarthy. Tale proposta è il risultato della consultazione pubblica iniziata dalla Commissione Europea il 19 ottobre 2000. A sua volta tale consultazione fu avviata sulla base del *Libro verde sul brevetto comunitario e sul sistema dei brevetti in Europa, Promuovere l'innovazione tramite il brevetto*, 24 giugno 1997, dove la Commissione si interroga sull'opportunità di introdurre una tutela brevettuale sul software. A pagina 4 si legge: "*Il Libro verde persegue un triplice obiettivo: offrire un quadro il più possibile completo della situazione in materia di protezione dell'innovazione con il sistema dei brevetti nella Comunità europea; valutare l'opportunità di nuove azioni comunitarie e/o di una modifica degli attuali regimi; studiare forma e contenuto di eventuali nuove azioni*". La Commissione invita a riflettere sul fatto che in altri paesi, uno su tutti gli Stati Uniti, sia riconosciuta la brevettabilità del software. Nel farlo sottolinea le più recenti linee guida del Patent Office, notando, a pagina 17 come "*mentre in passato una rivendicazione relativa ad un algoritmo matematico era accettata soltanto se vi era una trasformazione fisica, oggi viene raccomandato un approccio più pragmatico, basato sull'esigenza di "utilità" dell'invenzione. Questo ha come effetto un ampliamento del campo delle invenzioni connesse ai programmi per elaboratore che sono brevettabili. La pratica negli Stati Uniti era comunque già molto vasta: un programma per elaboratore incorporato in un mezzo tangibile come un dischetto, era brevettabile (26 aprile 1996, in re Beauregard) già prima della pubblicazione dei nuovi orientamenti e che a livello internazionale non sia esclusa da alcun trattato la non brevettabilità del software*". Sottolinea inoltre, sempre a pagina 17, come le convenzioni internazionali non vietano la brevettabilità del software *tout court*: "*l'articolo 27 dell'accordo TRIP non esclude la brevettabilità dei programmi per elaboratore. Per questo motivo alcuni paesi terzi autorizzano il rilascio di un brevetto per tali programmi*". Il documento in Rete: <http://eur-lex.europa.eu/legal-content/IT/TXT/PDF/?uri=CELEX:51997DC0314&from=EN>.

³⁰⁰ Così emerge dal *Libro verde sul brevetto comunitario e sul sistema dei brevetti in Europa*, pp. 17-18 e della *Proposta di direttiva del Parlamento europeo e del Consiglio relativa alla brevettabilità delle invenzioni attuate per mezzo di elaboratori elettronici*, considerando n. 4, 7, 8.

La prima motivazione, di carattere tecnico, riteneva la proposta essere in palese contrasto con l'articolo 52 dell'EPC nella parte in cui vieta la brevettabilità del programma per elaboratore in quanto tale³⁰¹.

La seconda motivazione faceva invece leva sul potenziale danno che avrebbero avuto i piccoli sviluppatori e la comunità dell'*open source*. In tale opposizione un ruolo predominante lo ebbe la FFII, organizzazione no-profit tedesca³⁰², da sempre, ed ancora oggi, impegnata alla difesa del mercato del software libero.

La proposta del 2002, venne modificata significativamente dal parlamento europeo nel 2003, attraverso molti emendamenti volti a impedire la brevettabilità del software in quanto tale³⁰³.

Nonostante tale modifica, nel maggio del 2004 il Consiglio approvò la direttiva nel suo testo originale, suscitando molte proteste. Da luglio dello stesso anno sino a marzo 2005 la proposta fu bloccata dall'ostruzionismo della Polonia. L'8 marzo 2005 il Consiglio ha adottato la direttiva, con il voto contrario di Danimarca, Portogallo, Polonia, Spagna e l'astensione dell'Italia. Il parlamento europeo tuttavia respinse la direttiva a maggioranza qualificata il 6 luglio 2005.

Ma cosa prevedeva in concreto la proposta³⁰⁴?

Anzitutto all'articolo 2 venivano definiti i concetti di "invenzione attuata per mezzo di elaboratori elettronici" e di "contributo tecnico".

Il primo venne descritto come *"un'invenzione la cui esecuzione implica l'uso di un elaboratore, di una rete di elaboratori o di un altro apparecchio programmabile e che presenta a prima vista una o più caratteristiche di novità che sono realizzate in tutto o in parte per mezzo di uno o più programmi per elaboratore"*³⁰⁵. Si noti subito come tale definizione differisca da quelle fornite nella disamina del primo capitolo³⁰⁶.

³⁰¹ V. M. A. Rossi, *Software patents: a closer look at the European commission's proposal*, Siena Memos and Papers in Law and Economics n. 30, Univ. Siena, 2005. Si rileva che, nonostante la commissione dichiarò di voler rispettare lo *status quo* adottato da EPO, le conseguenze della direttiva consisterebbero in una estensione della tutela mediante brevetto, il che sarebbe in palese contrasto con l'articolo 52 della stessa convenzione, che vedrebbe il divieto di brevettare il software in quanto tale, di fatto non rispettato. Per le stesse ragioni chi appoggiava la direttiva sosteneva la necessità di una modifica dell'articolo 52 dell'EPC, proprio nella parte in cui vieta la brevettabilità dei programmi in quanto tali. Il paper in Rete: https://people.ffii.org/~jmaebe/swpat/studies/docs/rossi05-ROSSI_Paper.pdf.

³⁰² Foundation for a Free Information Infrastructure, organizzazione no-profit con sede a Monaco di Baviera, fondata nel 1999 dedicata alla creazione di un libero mercato nella tecnologia dell'informazione.

³⁰³ Oltre alla già citata FFII, pareri contrari alla proposta di direttiva che influenzarono il parlamento ad una sostanziale modifica del testo giunsero del mondo accademico, del governo francese, della Commissione tedesca dei monopoli, da EuroLinux e della Free Software Foundation.

³⁰⁴ Proposta di direttiva del Parlamento europeo e del Consiglio relativa alla brevettabilità delle invenzioni attuate per mezzo di elaboratori elettronici, (2002/C 151 E/05), COM(2002) 92 def., 2002/0047(COD), Presentata dalla Commissione il 20 febbraio 2002, in Rete <http://eur-lex.europa.eu/legal-content/IT/TXT/PDF/?uri=CELEX:52002PC0092&from=IT>. Per una disamina approfondita della proposta v. G. GUGLIELMETTI, *La proposta di direttiva sulla brevettabilità delle innovazioni in materia di software*, in *Rivista di diritto industriale*, 2002, 440.

³⁰⁵ Proposta di direttiva del Parlamento europeo e del Consiglio relativa alla brevettabilità delle invenzioni attuate per mezzo di elaboratori elettronici, art. 2, a).

³⁰⁶ Cfr. *supra*, cap. I, par. 4.

Il discrimine risiede nel fatto che, in tale circostanza, la Commissione mirava a definire non il “programma per elaboratore”, ma quella che in precedenza abbiamo chiamato “invenzione di software”. La proposta in oggetto, infatti, lasciava impregiudicata la normativa derivante dalla direttiva 91/250/CEE³⁰⁷. L’oggetto delle rispettive tutele è infatti diverso.

Il contributo tecnico, invece, veniva definito come *“un contributo allo stato dell’arte in un settore tecnico, giudicato non ovvio da una persona competente nella materia”*³⁰⁸. La Commissione definì tale elemento perché fu considerato fondamentale nella concessione di una tutela brevettuale ad una invenzione di software. L’articolo 4, che stabilisce le condizioni alla brevettabilità, prevedeva infatti che l’invenzione doveva essere *“atta ad un’applicazione industriale, presenti un carattere di novità e implichi un’attività inventiva”*³⁰⁹.

Con particolare riferimento all’attività inventiva questa sussiste se *“l’invenzione attuata per mezzo di elaboratori elettronici arrechi un contributo tecnico”*³¹⁰.

Novità, attività inventiva ed applicazione industriale sono quelle condizioni che, come si è visto, sono previste dalla EPC e dalle altre convenzioni internazionali affinché una invenzione possa essere brevettata.

In particolare, la Convenzione Europea sul brevetto, come si è visto, all’articolo 56, specifica il concetto di attività inventiva, ma solo le Linee Guida di EPO e la sua giurisprudenza applicano il concetto all’invenzione di software, trovando nel contributo tecnico il punto di separazione tra una invenzione di software, quindi brevettabile, ed una “non-invenzione di software”³¹¹.

La proposta aveva quindi lo scopo di trasferire questi concetti, maturati in seno all’EPO, in una direttiva.

5. La legislazione degli Stati Uniti

5.1. Il Patent Act del 1952

Negli Stati Uniti la normativa di riferimento in tema di brevetti è il Patent Act del 1952. In particolare, la *section 101* individua l’ambito del *patentable subject matter*, definendo i requisiti per la brevettabilità delle invenzioni.

Tale norma può essere considerata come il reciproco dell’articolo 52 della EPC, ma, raffrontata a questo, la *section 101* ne è speculare perché enuncia

³⁰⁷ V. Proposta di direttiva del Parlamento europeo e del Consiglio relativa alla brevettabilità delle invenzioni attuate per mezzo di elaboratori elettronici, art. 6: *“La protezione conferita da i brevetti per le invenzioni che rientrano nel campo d’applicazione della presente direttiva lascia impregiudicate le facoltà riconosciute dalla direttiva 91/250/CEE relativa alla tutela giuridica dei programmi per elaboratore mediante il diritto d’autore, in particolare le disposizioni relative alla decompilazione e all’interoperabilità o le disposizioni relative alle topografie dei semiconduttori o ai marchi commerciali.”*, che codifica quanto affermato anche nel considerando n. 18.

³⁰⁸ *Ibidem*, art. 2, b).

³⁰⁹ *Ibidem*, art. 4.

³¹⁰ *Ibidem*, art. 4.

³¹¹ V. considerando n.10-13, dove si affronta la questione circa il contributo tecnico.

espressamente le categorie di oggetti che possono essere oggetto di brevetto, non prevedendo invece alcuna lista di eccezioni³¹².

La norma prevede che: *"Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title"*³¹³.

Traducendo tale norma si evince che le categorie che possono essere oggetto di brevetto sono i procedimenti, o i processi, le macchine, i manufatti e *"le composizioni ottenute tramite la miscela i sostanze diverse"*³¹⁴. Si noti come non sia previsto alcun caso di esclusione del software.

La norma in questione non individua solo l'oggetto del brevettabile ma pone anche il requisito della utilità. L'*utility* si esprime da un lato nel corretto funzionamento dell'invenzione, dall'altro nella realizzazione di quel determinato scopo pratico per cui è stata progettata³¹⁵.

La *section* 102 stabilisce il criterio della novità dell'invenzione (*novelty*). Si esclude la brevettabilità se l'invenzione è conosciuta o usata da terzi, ovvero già brevettata o pubblicata³¹⁶. Con riguardo alla applicazione del criterio è da segnalare un elemento che contraddistingue il sistema statunitense dagli standard internazionali. Storicamente infatti il criterio usato per verificare la novità dell'invenzione negli USA risiede nel *first-to-invent*. A livello internazionale, invece, il criterio utilizzato è il *first-to-file*. La differenza è palese perché, solo nel secondo caso, vige la presunzione di perfetta corrispondenza tra momento di registrazione e momento di invenzione, prevenendo così delicati problemi di paternità. Con l'America Invents Act (AIA) del 2011, gli Stati Uniti passano ad un regime di *first-to-file* "attenuato"³¹⁷, operativo dal 16/03/2013.

Un ulteriore requisito è enunciato nella *section* 103 dove viene illustrato il criterio della originalità ovvero della c.d. *"non obviousness"*, che specifica come l'invenzione, considerata nella sua interezza, non debba risultare ovvia, alla luce delle conoscenze possedute da una persona competente in quel determinato ambiente tecnico³¹⁸. Alla luce di tale criterio risulta ovvia l'applicazione o l'utilizzo essenziale di una tecnica già nota allo stato dell'arte³¹⁹.

³¹² In questi termini di raffronto AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, pp. 104-111.

³¹³ 35 U.S. Code § 101 - Inventions patentable.

³¹⁴ Le *composition of matter* vengono così tradotte da E. AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 104.

³¹⁵ In questa accezione M. A. LEMLEY, P. S. MENELL, R. P. MERGES, *Intellectual Property in the New Technological Age: 2016, Vol. I Perspectives, Trade Secrets and Patents*, Clause 8 Publishing, 2016, Chapter 3, p. 10-11. Nel dettaglio v. pp. 93-108.

³¹⁶ V. 35 U.S. Code § 102 - Conditions for patentability; novelty. V. C. A. NARD, *The Law of Patents*, Aspen Publishers, 2008, pp. 187-259.

³¹⁷ Come si specifica in LEMLEY, MENELL, MERGES, *Intellectual Property in the New Technological Age: 2016, Vol. I Perspectives, Trade Secrets and Patents*, Chapter 3, pp. 19 ss. il regime è una *"variation"* del sistema internazionale *first-to-file*, stante il *"grace period"* di un anno concesso all'inventore a certe condizioni. Cit., 55-56. Cfr. 35 U.S. Code § 102, b, 1.

³¹⁸ V. 35 U.S. Code § 103 - Conditions for patentability, non-obvious subject matter; Il caso di riferimento è *Graham v. John Deere Co.*, 383 U.S. 1, 1966, per una analisi v. C. A. NARD, *The Law of Patents*, Aspen Publishers, 2008, pp. 327-339.

³¹⁹ Per una analisi approfondita v. LEMLEY, MENELL, MERGES, *Intellectual Property in the New Technological Age: 2016: Vol. I Perspectives, Trade Secrets and Patents*, Chapter 3, pp. 58-93.

L'ultima condizione risiede nella “*disclosure*”, codificata nella *section* 112. La rivendicazione deve contenere una descrizione dell'oggetto del brevetto, tanto dettagliata da essere in grado di informare adeguato sulla scoperta una persona esperta del settore³²⁰. Questa caratteristica è molto importante perché costituisce il principale contrappeso al monopolio conferito dallo stato all'inventore.

Il combinato disposto di queste norme non esclude il software dall'alveo degli oggetti brevettabili. Ai sensi della *section* 101, infatti, se il programma per elaboratore fosse considerato alla stregua di una macchina o di un processo, allora sarebbe brevettabile.

La categoria dei procedimenti ha da sempre rappresentato una categoria ad oggetto molto incerto. Il processo è infatti, per sua natura, astratto “*e si avvicina decisamente, per tale aspetto, proprio a quegli elementi esclusi dal brevettabile, quali le leggi naturali o le idee astratte*”³²¹. Da questa circostanza emerge la difficoltà nel riconoscere al software, un oggetto composto da algoritmi, la natura di procedimento brevettabile (in quanto trovato di istruzioni codificate in un linguaggio informatico), ovvero di idea astratta o legge della natura, e, come tale, non brevettabile.

Proprio per tale difficoltà, è sempre stata forte la “tentazione” di concedere la tutela brevettuale soltanto a quelle invenzioni di software caratterizzate da un certo grado di materialità, il che significa nella prassi brevettare il sistema hardware-software, e non invece il programma considerato nella sua essenza.³²²

Una tale interpretazione sarebbe necessaria ma non sufficiente per concedere ad un software la tutela brevettuale. Il Patent Act richiede infatti una verifica sulla novità del trovato e sulla sua originalità³²³.

Similmente alla normativa dell'EPC, anche quella statunitense può essere interpretata in diversi modi onde ammettere o escludere il programma per elaboratore dalla cerchia degli oggetti brevettabili. Un ruolo importante nel delineare i criteri interpretativi lo ebbero la giurisprudenza da un lato, e le *Guidelines* del Patent Office dall'altro.

È opportuno evidenziare come il cambiamento da una interpretazione restia a ricomprendere il software all'interno delle invenzioni ex *section* 101, ad una volta invece alla sua inclusione, iniziò negli anni ottanta, ossia all'indomani della codifica del programma per elaboratore nel Copyright Act³²⁴. Ruolo

³²⁰ V. *Ibidem*, pp. 108-138.

³²¹ AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 106.

³²² Circostanza che si verificò nei primi brevetti che implementavano nell'invenzione anche una parte software, come ad esempio, US 3,581,072 “*Auction Market Computation System*” registrato nel 1968, v. LEITH, *software and patents in Europe*, pp.12-13 ; ma anche in tempi “più recenti” v. *infra*, In re Alappat, 33 F.3d 1526 (Fed. Circ. 1994), cap. IV, par. 1.4.

³²³ Una posizione molto interessante con riguardo al “problema” della brevettabilità dei programmi per elaboratore si legge in J. FELLAS, *The Patentability of Software-Related Inventions in the United States*, cit., 333: “*the real question is no longer whether software-related inventions claim patentable subject matter, but, rather, whether such inventions are novel and non obvious*”.

³²⁴ Così AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, pp. 101-104.

centrale lo ebbero sicuramente le corti, ma, per comprendere a fondo il passaggio, si deve necessariamente fare riferimento anche ad altre circostanze.

Si è collocata la nascita dell'industria software proprio negli anni ottanta, e questo elemento deve sicuramente fare riflettere. Se negli anni settanta il mercato della tecnologia era gestito dai produttori di hardware, a partire dagli anni ottanta le *software houses* iniziano ad assumere una certa rilevanza economica, il che le porta ad avere un certo peso, esercitabile sulla politica attraverso gruppi di pressione (*lobby*)³²⁵.

Ma non furono solamente circostanze di tipo economico-politico ad "influenzare" le corti. È necessario infatti guardare anche alla normativa relativa al Copyright, ed in particolare alla relazione sulle nuove tecnologie ad opera della commissione CONTU, che pose importanti basi sulla differenziazione di una tutela mediante copyright ed una mediante brevetto.

5.2. Dalla Commissione CONTU al Computer Software Copyright Act del 12 dicembre 1980

Nel 1976 fu compiuta una rilevante revisione del copyright statunitense, che modificò la precedente normativa del 1909. Prima della stesura della legge venne istituita la *National Commission on New Technology Uses of Copyrighted Works (CONTU)*, con l'obiettivo di effettuare una valutazione sull'utilizzo del copyright relativamente ai nuovi rami della tecnologia, in particolare con riferimento ai programmi per elaboratore.

Le analisi effettuate dalla commissione sono molto importanti, e confluiranno in un documento³²⁶ che influenzerà negli anni seguenti la posizione statunitense, sia in riferimento al software come oggetto di una tutela autoriale, sia in riferimento al software come oggetto di tutela brevettuale.³²⁷

³²⁵ LEITH, *Software and patents in Europe*, cit., 25. L'autore riferisce come negli Stati Uniti, negli anni settanta, le grandi aziende, che ai tempi fatturavano attraverso la produzione di hardware, fossero ostili alla brevettabilità dei programmi per elaboratore e agirono attraverso il *lobbying*. Riporta una comunicazione di Jim Sprowl: "IBM addressed this problem by lobbying for software to be unpatentable. When the President set up a commission to study the U.S. patent system, a senior IBM manager was appointed head of this commission. Not surprisingly, the commission concluded that software should not be patentable because the patent examiners were not prepared to examine software patents". Si fa riferimento alla President's Commission on the Patent System, *To promote the Useful Arts*, del 1966. Come riferisce anche GATTEI, *Il brevetto di software, guida teorico pratica*, pp. 74-75, la Commissione ritenne sufficiente la tutela mediante copyright, evidenziando come tale strumento fosse idoneo a stimolare la ricerca, ed evidenziando come allo stesso tempo l'industria informatica fosse in crescita anche in assenza di una protezione brevettuale. Oltre a queste motivazioni di stampo economico la Commissione sottolineò come gli uffici, non sarebbero stati in grado di valutare correttamente lo stato dell'arte, a causa della mancanza di una precisa classificazione tecnica del software, impedendo così una cosciente valutazione l'effettiva brevettabilità dei trovati. Sulla rilevanza del contesto politico economico di riferimento v. anche AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, p. 105.

³²⁶ Final Report of the National Commission on New Technology Uses of Copyrighted Works, 31/07/1978. Il documento è disponibile in Rete: <http://digital-law-online.info/CONTU/contu1.html>.

³²⁷ Così GATTEI, *Il brevetto di software, guida teorico pratica*, pp. 76-78.

La commissione anzitutto analizza come l'industria informatica sia in rapida evoluzione, sottolineando come il progresso della tecnologia derivi non solo da un miglioramento delle macchine, ma anche da un costante ampliamento delle loro funzioni³²⁸.

Dopo tali valutazioni preliminari si fornisce una definizione di programma per elaboratore come *"a sets of instructions which, when properly drafted, are used in an almost limitless number of ways to release human beings from such diverse mundane tasks (...). Computer programs are prepared by the careful fixation of words, phrases, numbers, and other symbols in various media"*³²⁹.

Da un lato, si individua il software come un insieme di istruzioni, guardando quindi al suo effettivo contenuto, ed alla funzione che tali istruzioni predispongono, dall'altro, si enfatizza la componente della sua espressione e della sua scrittura, notando anche come *"The need for protecting the form of expression chosen by the author of a computer program has grown (...)."*³³⁰.

La Commissione, sembra quindi non escludere la brevettabilità del software³³¹, ma distingue tra ambiti di protezione. Il primo è costituito dalla espressione del programma per elaboratore, ossia la sua scrittura, la forma con cui è realizzato, tutelabile mediante copyright. Il secondo è il contenuto del software, il processo, l'insieme di istruzioni che vengono implementate dagli algoritmi, cui accede la tutela brevettuale.³³²

Il codice del programma andrebbe quindi considerato alla stregua di una opera letteraria, mentre non si esclude che il processo implementato possa essere brevettato, sempre che sottenda ai requisiti di novità ed originalità previsti dal Patent Act.

Sulla base di tale relazione si adottò il Computer Software Copyright Act del 12 dicembre 1980. In particolare la *section 101* fu emendata, incorporando

³²⁸ CONTU, *Final Report*, 1978, Cit., Capitolo 3, Computers and Copyright, Foundation for the Recommendations, Computer Programs: *"As the number of computers has increased dramatically, so has the number of programs with which they may be used. While the first computers were designed and programmed to perform one or a few specific tasks, an ever increasing proportion of all computers are general-purpose machines which perform diverse tasks, depending in part upon the programs with which they are used. Early programs were designed by machine manufacturers to be used in conjunction with one model or even one individual computer. Today, many programs are designed to operate on any number of machines from one or more manufacturers. In addition, and perhaps even more importantly, there is a growing proportion of programs created by persons who do not make machines"*.

³²⁹ Cit., *ibidem*.

³³⁰ Cit., *ibidem*.

³³¹ In questo senso v. GATTEI, *Il brevetto di software, guida teorico pratica*, pp. 76-78.

³³² La commissione CONTU conclude illustrando un ventaglio di possibili tutele applicabili al software, cit., Capitolo 3, Computers and Copyright, Foundation for the Recommendations, Computer Programs: *"The Commission is unanimous in its belief that computer programs are entitled to legal protection. But the unanimity has not extended to the precise form that protection should take. The law as it exists today with respect to the protection of computer programs is not totally clear. What is clear is that today there are different and often conflicting methods used by proprietors to attempt to protect their products. These include patent and copyright – exclusively federal statutory methods; trade secret law – derived from statutory and judicial state law; and unfair competition – based on elements of common law and federal statute"*.

una definizione di programma per elaboratore³³³, cristallizzando quindi la sua idoneità ad accedere ad una tutela -anche- autoriale.

5.3. Le *Guidelines* del Patent Office statunitense

A metà degli anni novanta il Patent Office statunitense pubblicò un documento di revisione delle *Guidelines*. Come si è già notato con riguardo alle Linee Guida di EPO, le linee guida ricoprono un ruolo di primaria importanza, perché cristallizzano i criteri fondamentali che gli uffici devono utilizzare durante l'analisi delle domande di brevetto.

La revisione delle *Guidelines* fu sicuramente fagocitata dallo svilupparsi di determinati orientamenti giurisprudenziali, dei quali si darà atto nel successivo capitolo, ma anche da un generale movimento volto ad assicurare al software una adeguata tutela³³⁴.

La revisione delle *Guidelines* fu accompagnata da un documento molto importante perché direttamente riferito alle invenzioni di software: *Legal Analysis to Support Proposed Examination Guidelines for Computer-Implemented Inventions*³³⁵. Attraverso tale documento si fissano gli step da verificare al fine di concedere, o meno, il brevetto all'invenzione.

Anzitutto si deve individuare l'oggetto della domanda, considerando l'invenzione nella sua interezza.

³³³ V. Final Report, cit., Chapter 3 – Computers and Copyright, Recommendations for Statutory Change: “*That section 101 be amended to add the following definition: A computer program is a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result*”. Tale definizione fu ripresa letteralmente nel 7 U.S. Code § 101, Definitions, “computer program”.

³³⁴ Ne è un esempio il ciclo di *Public Hearings on Software Patents* indetti dal Patent and Trademark Office nel 1994 ad Arlington e a San Jose, finalizzati “*in obtaining public input on issues associated with the patenting of software-related inventions*”. A tal fine furono posti determinati argomenti di discussione riassunti in tre macro-aree: “*Use of the patent system to protect software-related inventions; Standards and practices used in examination of patent applications for software-related inventions; Significance of and protection for visual aspects of software-related inventions*.” Il documento di notizia dell'evento, dove vengono fissati i *topic* di discussione è reperibile in Rete: <https://www.uspto.gov/web/offices/com/hearings/software/notices/notice94.html>.

La discussione che scaturì da tali incontri può essere riassunta a tre fondamentali conclusioni. Anzitutto si rileva come la tutela mediante copyright non risulta essere completa ed efficace a garantire una protezione nei confronti della concorrenza, frustando così gli investimenti delle imprese. Da qui si concorda sulla predisposizione di una tutela brevettuale del software, giudicando come idonei a fornire tale tutela gli strumenti già predisposti dal legislatore. Il secondo, fa notare come talvolta il Patent Office abbia concesso la tutela brevettuale a trovati di software che non sembrano incontrare i necessari requisiti di originalità e novità. L'ultimo punto denota come le aziende sentano la necessità di una maggiore specializzazione tecnica del Patent Office, lamentando come lo stesso non sia aggiornato allo stato dell'arte. Le trascrizioni degli incontri sono disponibili in Rete, incontri di San Jose, Wednesday, January 26, 1994 Thursday, January 27, 1994: <https://www.uspto.gov/web/offices/com/hearings/software/sanjose/sjhrng.pdf>; incontri di Arlington, Thursday, February 10, 1994, Friday, February 11, 1994: <https://www.uspto.gov/web/offices/com/hearings/software/arlington/vahrng.pdf>.

³³⁵ Patent and Trademark Office, United States Department of Commerce, *Legal Analysis to Support Proposed Examination Guidelines for Computer-Implemented Inventions*, 3/10/1995. In Rete: <https://www.uspto.gov/web/offices/com/hearings/software/analysis/softlaw.pdf>.

Per esempio, se è vero che una rivendicazione avente ad oggetto un mero algoritmo matematico vada rigettata³³⁶, una rivendicazione costituita in parte da un algoritmo, la si deve considerare alla luce dell'intero processo implementato nell'invenzione. Non si può quindi scorporare la formula matematica, alla luce della sua isolata non brevettabilità, dal resto dell'invenzione³³⁷, per valutarla separatamente dal resto degli elementi propri dell'intero processo o sistema presentato.

Per individuare l'oggetto della invenzione di software, il personale deve: determinare la funzionalità del programma, ossia ciò che il software produce quando viene eseguito dal computer; determinare la configurazione della macchina; determinare la relazione tra il programma e gli altri elementi dell'invenzione³³⁸.

Dopo tale triplice analisi, l'oggetto, così individuato, deve presentare una utilità pratica, vagliata tenendo conto dello stato dell'arte e della tecnologia³³⁹, e non deve essere annoverabile tra le entità astratte. L'invenzione deve quindi produrre un contributo pratico; infatti: *"If the utility of an 'invention' is only as an object of philosophical inquiry or to be appreciated in terms of its literary or artistic value, the claimed invention should be rejected under § 101"*³⁴⁰.

Accertata la brevettabilità del trovato nel suo complesso, si deve individuare la sua categoria di appartenenza. Si deve quindi fare riferimento ai tali oggetti la cui brevettabilità è prevista dalla *section 101*, ossia: *process, machine, manufacture, composition of matter*. Le invenzioni potranno quindi essere classificate come *Non-Statutory Subject Matter* o *Statutory Subject Matter* a seconda della loro idoneità o meno ad essere classificati alla stregua di tali oggetti.

Le linee guida individuano tre rivendicazioni che non rientrano nel campo della *section 101*.

Il primo caso fa riferimento alle strutture di dati e ai programmi per elaboratore in quanto tali³⁴¹. Questi oggetti sono considerati come puramente astratti e non tangibili, quindi non brevettabili, perché non descrivono azioni o procedimenti da eseguirsi mediante elaboratore. Infatti, la rivendicazione intorno a tali oggetti diviene ammissibile solo quando la descrizione contenuta nella domanda inquadra la struttura di dati o il software in quanto tale all'interno

³³⁶ *Ibidem*, cit., 2: "Office personnel will no longer begin examination by determining if a claim recites a mathematical algorithm".

³³⁷ *Ibidem*, cit., 7: "Office personnel may not dissect a claimed invention into discrete elements and then evaluate the elements in isolation. Instead, each claimed element of the invention must be considered in the context of the claim as a whole".

³³⁸ *Ibidem*, cit., 3: "determine what the programmed computer does when it performs the processes dictated by the software (i.e., the functionality of the programmed computer); determine how the computer is to be configured to provide that functionality (i.e., what elements constitute the programmed computer and how are those elements to be configured to provide the specified functionality); and if applicable, determine the relationship of the programmed computer to other subject matter that constitutes the invention (e.g., machines, devices, materials, or process steps other than those that are part of or performed by the programmed computer)".

³³⁹ *Ibidem*, cit., 5: "The utility of an invention must be within the technological arts".

³⁴⁰ *Ibidem*, cit., 6.

³⁴¹ *Ibidem*, cit., 7: "Data structures per se or Computer programs per se".

di un procedimento eseguibile, ovvero come componente di una invenzione fisica.³⁴²

Nel secondo caso il Patent Office si riferisce alle *“Non-Functional Information”*. Le informazioni o anche contenuti non-funzionali sono *“parole, immagini o altre informazioni che non forniscono un'utilità pratica all'invenzione”*³⁴³. Ogni rivendicazione su tali contenuti non è ammissibile, perché questi non impartiscono nessuna funzionalità all'invenzione, risolvendosi in semplici oggetti che potranno tuttalpiù, qualora originali e frutto della creazione artistica dell'autore, essere oggetto di tutela mediante copyright.

Ultimo elemento che non rientra nelle categorie brevettabili, è costituito dai fenomeni fisici e naturali. Questi non possono essere oggetto di tutela brevettuale, anche se possono comunque determinare il funzionamento di un programma per elaboratore³⁴⁴.

Dopo aver fissato i criteri di inammissibilità delle rivendicazioni, il documento si preoccupa di classificare le invenzioni tutelabili, distinguendo a tal fine quelle che fanno riferimento ai prodotti e quelli che invece si riferiscono ai processi³⁴⁵.

I primi implementano un macchinario o un manufatto dotato di una certa utilità pratica, dove il programma per elaboratore compone un sistema composto da una struttura fisica che altro non è che una composizione hardware-software.

I secondi sono una serie di istruzioni eseguibili, che, una volta elaborati, trasformano quei dati o quelle entità eseguite in altrettanti dati o entità. Questo passaggio è fondamentale perché qualora il processo operi ad un livello meramente astratto, non sarà tutelabile³⁴⁶.

L'elaborazione deve condurre o ad una attività fisica diversa da quella di partenza, ovvero alla medesima ma modificata o trasformata al fine di ottenere una certa utilità pratica³⁴⁷. I processi esclusi dalla brevettabilità sono quindi quei

³⁴² *Ibidem*, cit., 7: *“If the claimed subject matter cannot be treated as a process and does not have any physical structure, then it is non-statutory information”*.

³⁴³ *Ibidem*, cit., 8, trad.

³⁴⁴ *Ibidem*, cit. 8: *“A claim directed to a natural phenomenon such as energy or magnetism, which does not recite the practical application of that phenomenon in a process or a product, is to be rejected under § 101”*.

³⁴⁵ C.d *“Statutory Subject Matter”*, v. *ibidem*, pp. 8 ss.

³⁴⁶ *Ibidem*, cit., 15: *“If the ‘acts’ of a process manipulate only numbers, abstract concepts or ideas, the acts are not being applied to appropriate subject matter. Thus, a process consisting solely of mathematical operations does not manipulate appropriate subject matter and thus cannot constitute a statutory process”*.

³⁴⁷ *Ibidem*, cit. 15: *“Results in a transformation or reduction of the subject matter manipulated into a different state or into a different thing to achieve a practical application. (...) Importantly, the subject matter manipulated by a process does not have to be a physical object; it may be ‘intangible subject matter representative of or constituting physical activity or objects’*. Tale impostazione riprende la decisione *Diamond v. Diehr*, 450 U.S. 175 (1981), v. *infra*, cap IV, par. 1.3.

trovati che eseguono meri calcoli matematici, ovvero elaborano entità astratte senza portare ad un risultato concreto e tangibile³⁴⁸.

Al momento della richiesta del brevetto sarà quindi importante fornire una completa descrizione del processo e del suo risultato. A tal proposito le linee guida forniscono i criteri per la verifica di una completa ed esauriente descrizione, da effettuarsi onde soddisfare il requisito della *section* 112. Questa dovrà contenere tutti gli elementi dell'invenzione, ossia il programma, le sue funzionalità, le interazioni logiche del sistema informatico ed i passi compiuti dal processo di elaborazione dei dati³⁴⁹.

Ultimo passo da compiere è la verifica dei requisiti di novità e di non ovvietà. Per potersi classificare come nuova l'invenzione non deve essere disponibile allo stato della tecnica, considerando il momento in cui è presentata domanda. L'ovvietà va valutata invece attraverso un raffronto con i programmi già disponibili.

Se un perito della scienza informatica fosse in grado di giungere a quella invenzione senza alcuno sforzo intellettuale, allora l'invenzione non può essere brevettata. Si escludono quindi quei trovati che pur non essendo direttamente disponibili allo stato della tecnica, ne sono direttamente derivabili senza alcun particolare sforzo³⁵⁰.

Svolti tali passaggi, l'ufficio può concludere l'analisi decretando se il trovato può o meno essere idoneo ad una tutela brevettuale³⁵¹.

³⁴⁸ Tale circostanza favorisce il riconoscimento della tutela brevettuale alle invenzioni aventi ad oggetto i "metodi di business". Si pensi ad esempio ai software gestionali, come ad esempio un software che implementa un metodo contabile. L'elemento del risultato concreto è costituito dalla trasformazione dei dati iniziali, in un certo dato finale, che non si risolve semplicemente in una serie di calcoli matematici, ma determina un certo risultato pratico. I dati inseriti ed ottenuti, infatti, sono riferibili ad elementi tangibili nella realtà fisica (come ad esempio fatture, ordini, ecc.) operanti in un contesto pratico. La sentenza che per prima ammise la brevettabilità dei metodi di business fu *State Street Bank & Trust Co. v. Signature Financial Group*, v. *infra*, cap. IV, par. 1.5. Per un approfondimento sul tema v. L. MANSANI, *La Brevettabilità dei Metodi Commerciali e delle invenzioni attuate per mezzo di elaboratori elettronici*, in Studi di diritto industriale in onore di Adriano Venzetti. Proprietà Intellettuale e concorrenza, II, Giuffrè, Milano, 2004, pp. 949 e ss. e ancora MANSANI, *I brevetti relativi a business methods e a computer implemented inventions*, in GALLI, *Le nuove frontiere del diritto dei brevetti*, pp. 41-66.

³⁴⁹ È importante notare come la descrizione debba essere fatta in lingua inglese. Quindi la mera allegazione del codice del programma non costituisce descrizione esauriente. Si deve quindi sottolineare come l'allegazione del programma in forma di codice sorgente sia facoltativa e non obbligatoria. La pubblicazione del codice non è richiesta in quanto non è considerata di per sé idonea a fornire quella dettagliata spiegazione dell'invenzione, richiesta *ex section* 112, tale da rendere un perito della materia conscio dell'innovazione apportata dall'invenzione. V. Patent and Trademark Office, United States Department of Commerce, *Legal Analysis to Support Proposed Examination Guidelines for Computer-Implemented Inventions*, 3/10/1995, pp. 17-21. Di contro si deve altresì notare come per la legislazione americana una descrizione tecnico-letterale dell'invenzione sia sufficiente a soddisfare il requisito della *disclosure*. Quest'ultimo elemento è invero molto importante perché costituisce il contrappeso che gli inventori devono soddisfare a fronte del monopolio fornito dallo stato.

³⁵⁰ *Ibidem*, pp. 21-22.

³⁵¹ *Ibidem*, cit., 22: "Once Office personnel have concluded an analysis of the claimed invention under all the statutory provisions, including §§ 101, 112, 102 and 103, when composing any Official action, they should review all the proposed rejections and their bases to confirm their correctness. Only then should any rejection be imposed".

CAPITOLO IV

Il ruolo della giurisprudenza nell'individuazione del software brevettabile

SOMMARIO: 1. La giurisprudenza statunitense - 1.1. *Premessa*: una analisi in chiave cronologica - 1.2. *Gottschalk v. Benson*: il software come formula matematica - 1.3. *Diamond v. Diehr*: il software come procedimento - 1.4. *In re Alappat*: il software come macchina - 1.5. *State Street Bank & Trust Co. v. Signature Financial Group*: l'utilità pratica del software - 1.6. *Bilski v. Kappos*: il *machine or transformation test* - 2. La giurisprudenza di EPO - 2.1. *Premessa*: l'importanza delle linee guida - 2.2. Il caso *Vicom*: la differenza tra un metodo matematico e un processo tecnico - 2.3. (*Segue*): il carattere tecnico tra il *contribution approach* e il *problem-solution approach* - 2.4. Le decisioni gemelle *IBM*: il software rivendicato di per sé - 2.5. La rimodulazione del carattere tecnico: il caso *Hitachi* - 2.6. "La via dell'originalità"? Il parere G 3/08 e i problemi aperti

1. La giurisprudenza statunitense

1.1. *Premessa*: un'analisi in chiave cronologica

A partire dagli anni ottanta, nel medesimo anno in cui si cristallizza all'interno del *Copyright Act* la tutela del software mediante *copyright*, le corti "intrapresero un lento cammino volto a scalfire, passo dopo passo, i pregiudizi contro lo strumento brevettuale"³⁵². Ritengo che nel procedere alla disamina della giurisprudenza americana sia utile utilizzare un approccio "evolutivo-cronologico".

In questo modo sarà possibile evidenziare non solo quegli elementi che hanno reso possibile il consolidarsi del software come *patentable subject matter*, ma anche quei passaggi che hanno portato il software ad essere oggetto non esclusivamente ad una tutela mediante *copyright*.

1.2. *Gottschalk v. Benson*: il software come formula matematica

La prima importante decisione assunta dalla Corte Suprema degli Stati Uniti, in merito alla possibilità o meno di brevettare un software, risale al 1972 con il caso *Gottschalk v. Benson*³⁵³.

³⁵² AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 101.

³⁵³ *Gottschalk v. Benson*, 409 U.S. 63 (1972). In Rete: <https://supreme.justia.com/cases/federal/us/409/63/case.html>.

L'invenzione, oggetto del contenzioso, riguardava un algoritmo matematico, che, una volta implementato in un sistema di elaborazione di dati, era in grado di convertire i numeri decimali in formato binario.

Il Patent Office rifiutò l'istanza di rivendicazione, mentre la corte d'Appello per i brevetti rigettò la decisione dell'Ufficio. Fu così richiesto il pronunciamento della Corte Suprema, che dovette valutare se l'invenzione in questione fosse brevettabile alla luce dei requisiti fissati dalla legge.

La Corte rilevò due fondamentali circostanze. Anzitutto sostenne che l'algoritmo non avrebbe potuto avere alcuna applicazione pratica fuori dal suo utilizzo all'interno di un elaboratore.

Secondariamente espose come la concessione della tutela brevettuale al trovato, così come rivendicato, avrebbe significato la sostanziale estensione della privativa alle formule matematiche, evenienza questa esclusa dalla legge, sulla base del fatto che esse sono entità astratte³⁵⁴.

Un algoritmo, così come una formula matematica, rivendicato di per sé è non brevettabile perché rientra nella categoria dei fenomeni naturali e dei concetti astratti, che sebbene siano sicuramente oggetto di scoperta, sono strumenti basilari per l'attività scientifica e quindi non possono essere oggetto di privativa brevettuale³⁵⁵.

La decisione rappresentò il punto di riferimento per l'esclusione dei programmi per elaboratore dall'ambito del brevettabile. Tuttavia, una disamina più approfondita del caso ci mostra come la Corte, già all'epoca, non fosse a priori avversa all'utilizzo dello strumento brevettuale per la tutela dei software³⁵⁶.

La brevettabilità fu esclusa nel caso in esame perché l'invenzione non insisteva su alcuna applicazione sostanziale. Questo elemento di discriminare può essere spiegato più nella modalità in cui è stata presentata in concreto la domanda, che nella effettiva inidoneità di un programma per elaboratore alla risoluzione di determinati problemi tecnici³⁵⁷.

³⁵⁴ *Ibidem*, cit. p. 409 U.S. 71: "The mathematical formula involved here has no substantial practical application except in connection with a digital computer, which means that, if the judgment below is affirmed, the patent would wholly preempt the mathematical formula and, in practical effect, would be a patent of the algorithm itself".

³⁵⁵ Il principale mezzo per escludere gli algoritmi dalla tutela brevettuale fu la teoria del "mental steps". Tale dottrina nacque negli anni sessanta e si basava sul non considerare brevettabile tutto ciò che fosse riconducibile ad una sequenza di passaggi mentali. Tale tesi trovò applicazione nel programma per elaboratore stante la sua riduzione, in quel determinato periodo storico, ad un semplice algoritmo. Lo stesso algoritmo, quindi, viene ridotto a processo mentale. L'impiego di mezzi fisici, per produrre un certo risultato pratico, entra in gioco in un momento successivo all'attività inventiva. Il prodotto iniziale, il vero oggetto del brevetto, deve essere dunque considerato come una idea astratta. V. P. SAMUELSON, *Benson Revisited: The Case Against Patent Protection For Algorithms and Other Computer Program-Related Inventions*, Berkeley Law Scholarship Repository, 39 *Emory L.J.* 1025 (1990), in Rete: <http://scholarship.law.berkeley.edu/cgi/viewcontent.cgi?article=1908&context=facpubs>; GATTEI, *Il brevetto di software, guida teorico pratica*, pp. 185-189; GUGLIELMETTI, *L'invenzione di software. Brevetto e diritto d'autore*, pp. 32-35.

³⁵⁶ In questo senso anche AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, p. 101; GATTEI, *Il brevetto di software, guida teorico pratica*, pp. 153-154.

³⁵⁷ *Gottschalk v. Benson*, cit. p. 409 U.S. 63: "It is said that the decision precludes a patent for any program servicing a computer. We do not so hold".

La richiesta della privativa brevettuale fu quindi rigettata perché era inerente ad un algoritmo matematico “puro”, ossia ad una idea astratta, senza che ne fosse individuata dalla domanda una specifica applicazione pratica.

Dopo tale decisione, la tendenza fu quella di escludere i programmi da elaboratore dell’ambito del brevettabile, paragonandoli a formule matematiche e considerandoli privi di una concreta e sostanziale applicazione industriale.

L’idoneità di ricavare un’interpretazione “alternativa” della decisione di *Gottschalk v. Benson*, si rinviene nel caso *Parker v. Flook*³⁵⁸.

L’invenzione in oggetto consisteva in un metodo di calcolo volto al controllo del processo di conversione del greggio in idrocarburi³⁵⁹. Nella fattispecie, un algoritmo gestiva e monitorava attraverso un procedimento computerizzato molte variabili chimiche, per garantire che le stesse rientrassero in determinati limiti.

La richiesta di *Flook* è particolare perché in essa il richiedente si premura di individuare l’ambito di applicazione pratica della formula. Sebbene il Patent Office in prima battuta respinse la richiesta, la corte di Appello ritenne il trovato brevettabile, sulla base dell’indicazione del settore di applicazione industriale. Tale interpretazione del precedente non persuase tuttavia la Corte Suprema.

La Corte individuò nell’algoritmo la sola componente originale dell’intenzione; componente che tuttavia, per sua natura, non poteva essere brevettata³⁶⁰.

Tale circostanza si palesò perché nella richiesta di brevetto l’algoritmo rimase ad un livello astratto³⁶¹. Nella fattispecie, non fu sufficiente la mera illustrazione dell’ambito industriale di applicazione, ma venne richiesta una completa descrizione del processo industriale in cui l’algoritmo è coinvolto. È necessario dare atto al concreto collegamento materiale, che deve sussistere tra utilizzo dell’algoritmo e risultato finale tangibile.

Le argomentazioni della Corte, tuttavia, trascendono i meri aspetti tecnici, focalizzandosi anche su questioni di più ampio respiro.

La Corte, in particolare, ritiene di non doversi sbilanciare sulla tipologia di tutela da offrire ai programmi per elaboratore stante l’enorme crescita che gli

³⁵⁸ *Parker v. Flook*, 437 U.S. 584 (1978). In Rete: <https://supreme.justia.com/cases/federal/us/437/584/case.html>.

³⁵⁹ L’invenzione prendeva il nome di: “*Automatic monitoring alarming*”.

³⁶⁰ *Parker v. Flook*, cit. p. 437 U.S. 594: “*Respondent’s process is unpatentable under § 101 not because it contains a mathematical algorithm as one component, but because, once that algorithm is assumed to be within the prior art, the application, considered as a whole, contains no patentable invention*”.

³⁶¹ La corte sviluppò quella che venne successivamente chiamata come dottrina o teoria del “*point of novelty approach*”. Durante l’esame di una domanda l’elemento di novità deve ricadere su un elemento tecnico affinché l’intera invenzione possa essere brevettabile. Nel caso di specie l’unica innovazione nel trovato, secondo la corte, era l’algoritmo, entità priva di contenuto tecnico e quindi non brevettabile. Sull’argomento v. M. A. LEMLEY, *Point of Novelty*, *Northwestern University Law Review*, Vol. 105, No. 3, 2011, in Rete: <http://scholarlycommons.law.northwestern.edu/cgi/viewcontent.cgi?article=1159&context=nulr>.

stessi stanno avendo nel mercato. Il timore è quello di soffocare una industria in piena espansione³⁶².

1.3. Diamond v. Diehr: il software come procedimento

Nel famoso caso Diamond v. Diehr³⁶³ la Corte Suprema si pronunciò a favore della brevettabilità di una invenzione di software³⁶⁴.

È bene comprendere in cosa consisteva l'invenzione e come venne presentata dall'inventore, al fine di capire la differenza tra questo trovato, brevettabile, e i precedenti considerati non meritevoli di tutela.

La domanda aveva ad oggetto un procedimento volto alla produzione di gomma sintetica. Per produrre correttamente tali manufatti, il brevetto implementava un software che controllava la temperatura e il tempo di sottoesposizione al trattamento termico, calcolando nell'immediatezza eventuali modifiche ai parametri sulla base della nota "Equazione di Arrhenius".

Il procedimento in questione sfruttava la matematica astratta in un procedimento industriale, idoneo al raggiungimento di un risultato concreto, tangibile e individuato³⁶⁵. La descrizione dell'invenzione comprendeva ogni fase del procedimento di lavorazione della gomma sintetica e la suddetta equazione di Arrhenius era stata implementata in un software, che gestiva e calcolava i parametri del processo di manipolazione.

Nello specifico, la Corte valutò l'invenzione idonea alla trasformazione di un oggetto in un oggetto ulteriore ovvero alla sua modificazione. L'invenzione era così brevettabile alla stregua di un processo, a prescindere dunque dalla materialità o meno del trovato.

³⁶² Parker v. Flook, cit. p. 437 U.S. 595: *"The youth of the industry may explain the complete absence of precedent supporting patentability. Neither the dearth of precedent nor this decision should therefore be interpreted as reflecting a judgment that patent protection of certain novel and useful computer programs will not promote the progress of science and the useful arts, or that such protection is undesirable as a matter of policy. Difficult questions of policy concerning the kinds of programs that may be appropriate for patent protection and the form and duration of such protection can be answered by Congress on the basis of current empirical data not equally available to this tribunal"*. Cfr. SAMUELSON, *Benson Revisited: The Case Against Patent Protection For Algorithms and Other Computer Program-Related Inventions*. Con riguardo a tale circostanza, l'autrice sostiene come i fautori della dottrina del *mental steps* avessero anche l'obiettivo di sottrarre gli algoritmi, o meglio i software nella forma di algoritmi, dall'oggetto brevettabile onde permetterne l'utilizzo alla collettività e alla comunità scientifica. Ciò detto rileva che nel momento in cui l'oggetto del brevetto non è più l'algoritmo, ma una sua applicazione pratica in un determinato ambito della tecnologia, allora verrebbe meno tale giustificazione su cui la teoria si basa.

³⁶³ Diamond v. Diehr, 450 U.S. 175 (1981). In Rete: <https://supreme.justia.com/cases/federal/us/450/175/case.html>.

³⁶⁴ Per una ricostruzione completa del caso v. M. A. O'ROURKE, *The story of Diamond v. Diehr: Toward Patenting Software*, in *Intellectual property stories*, DREYFUSS, GINSBURG (a cura di), p. 200.

³⁶⁵ Diamond v. Diehr, cit. p. 450 U.S. 176: *"When a claim containing a mathematical formula implements or applies the formula in a structure or process which, when considered as a whole, is performing a function which the patent laws were designed to protect (e.g., transforming or reducing an article to a different state or thing), then the claim satisfies § 101's requirements"*.

La Corte superò dunque la dottrina del *point of novelty*³⁶⁶, considerando l'invenzione non alla stregua di tanti elementi frazionabili, ma nel suo complesso, inaugurando di fatto la stagione del c.d. “*concrete, tangible, useful test*” che sarà modulato nelle successive pronunzie ad opera del CAFC.

Tale decisione aprì le porte alla brevettabilità dei software, ponendosi in evidente discontinuità con i precedenti della materia. Nel compiere tale passaggio è evidente l'*escamotage* utilizzato dalla Corte al fine di concedere la tutela brevettuale solo in quei casi in cui si fosse dimostrato che la rivendicazione implicasse passaggi di un determinato procedimento³⁶⁷.

1.4. In re Alappat: il software come macchina

I principi enunciati in *Diamond v. Diehr* vennero confermati nella decisione della Corte di Appello federale in re Alappat³⁶⁸.

Oggetto della rivendicazione era un software in grado, una volta implementato su un computer, di modificare l'intensità dei *pixel* di una immagine, in modo tale da renderla più dettagliata e nitida nei suoi particolari³⁶⁹.

Il trovato venne giudicato non brevettabile dal Patent Office, sulla base dell'assunto che l'invenzione consisteva in una mera formula matematica.

Il software, infatti, era idoneo a produrre quel determinato risultato a prescindere dall'implementazione dello stesso su di uno specifico apparato. I risultati prodotti dal software vennero presentati alla commissione esaminatrice attraverso un computer con caratteristiche standard. L'ufficio ritenne che la rivendicazione insistesse su una entità astratta e quindi non brevettabile.

La Corte d'Appello non si trovò d'accordo con questa visione. Mentre una formula matematica consiste in una idea astratta, l'oggetto della rivendicazione considerato nella sua interezza era un prodotto tangibile, un nuovo sistema informatico composto da una parte hardware e da una parte software.

Quest'ultimo elemento, in particolare, era idoneo a fornire all'elaboratore un miglioramento sotto il profilo funzionale tale da rendere il trovato così costituito meritevole di tutela brevettuale.

³⁶⁶ V. *supra*, nota n. 361.

³⁶⁷ V. J. E. COHEN, M. A. LEMLEY, *Patent Scope and Innovation in the Software Industry*, Georgetown Law Faculty Publications, 89 *Cal. L. Rev.* 1-57 (2001), 2010, cit., 9, ove si definisce questo *escamotage* come “*doctrine of the magic words*”, secondo cui si brevetta il programma per elaboratore attraverso la finzione di rivendicare una invenzione di fatto diversa. “*Under this approach, software was patentable subject matter, but only if the applicant recited the magic words and pretended that she was patenting something else entirely. During the 1980s and early 1990s, knowledgeable patent attorneys did exactly that, claiming software inventions as hardware devices, pizza ovens, and other “machines”*”, Ancora, per una visione critica, v. LEMLEY, MENELL, MERGES, SAMUELSON, CARVER, *Software and Internet Law*, pp. 281-283, dove si rileva come attraverso tale finzione sia tuttavia possibile anche brevettare non già invenzioni di software ma semplici algoritmi matematici.

³⁶⁸ In re Alappat, 33 F.3d 1526, U.S. Court of Appeals Federal Circuit (CAFC), 1994. In Rete: <http://digital-law-online.info/cases/31PQ2D1545.htm>.

³⁶⁹ *Ibidem*, cit., A. Alappat's Invention: “*Employing this anti-aliasing technique eliminates any apparent discontinuity, jaggedness, or oscillation in the waveform, thus giving the visual appearance of a smooth continuous waveform. In short, and in lay terms, the invention is an improvement in an oscilloscope comparable to a TV having a clearer picture*”.

La decisione è fondamentale perché per prima riconosce al software la capacità di implementare, con evidente contributo tecnico, determinate funzioni su di un computer *general-purpose*, rendendo così l'invenzione, considerata nel suo complesso, idonea ad eseguire precise funzioni di carattere industriale³⁷⁰.

Viene in particolare evidenziato dalla corte, attraverso l'utilizzo di un mero espediente, come il computer di fatto si trasformi, si specializzi, nel momento in cui viene implementato al suo interno un software.

Da un punto di vista pratico tale sentenza è molto importante perché renderà possibile la brevettabilità di molti programmi per elaboratore, mediante la loro implementazione su un hardware, ancorché standard.

Da un punto di vista teorico, tuttavia, essa rappresenta una sorta di "ritorno alle origini". Se infatti è necessario il "caricamento" del programma su un supporto materiale per riconoscere la brevettabilità all'invenzione così ottenuta, ciò significa riconoscere come il software non sia in grado di apportare alcuna applicazione pratica qualora considerato di per sé.

Viene così brevettato il sistema hardware-software, o meglio viene brevettato il software come parte integrante di una macchina³⁷¹.

Tali importanti concetti confluiranno, l'anno successivo alla decisione, nelle linee guida dell'USPTO³⁷². Il riflesso principale risiede nella possibilità di brevettare un programma per elaboratore se incorporato all'interno di un dispositivo di memoria leggibile da un computer, come ad esempio un CD, o meglio, considerando gli standard del 1995, un *floppy disk*³⁷³.

1.5. State Street Bank & Trust Co. v. Signature Financial Group: l'utilità pratica del software

Il caso State Street Bank & Trust Co. v. Signature Financial Group³⁷⁴ si innesta sulla rivendicazione di un brevetto concernente -anche- un metodo commerciale puro³⁷⁵.

L'invenzione in questione consiste in un sistema gestionale rivolto all'amministrazione finanziaria dei fondi comuni. Il software permetteva la gestione congiunta di più investimenti, effettuati da una pluralità di soggetti, in un unico portafoglio di valori, senza che ci fossero rischi di commistione dei

³⁷⁰ *Ibidem*, cit., D. Analysis, (2) Section 101, b). "We have held that such programming creates a new machine, because a general purpose computer in effect becomes a special purpose computer once it is programmed to perform particular functions pursuant to instructions from program software".

³⁷¹ Cfr. LEITH, *Patenting programs as machines*; v. *supra*, cap. III, par. 1.2.

³⁷² V. *supra*, cap. III, par. 5.3.

³⁷³ Patent and Trademark Office, United States Department of Commerce, *Legal Analysis to Support Proposed Examination Guidelines for Computer-Implemented Inventions*, 3/10/1995, cit., 11: "When functional descriptive material is recorded on some computer-readable medium it becomes structurally and functionally interrelated to the medium and will be statutory in most cases".

³⁷⁴ State Street Bank & Trust Co. v. Signature Financial Group, 149 F.3d 1368 (Fed. Cir. Jul. 23, 1998). In Rete: <http://digital-law-online.info/cases/47PQ2D1596.htm>.

³⁷⁵ Sui metodi commerciali v. *supra*, cap. III, par. 5.3 nota n. 348.

patrimoni. In questo modo l'amministratore avrebbe potuto risparmiare considerevoli somme di danaro in tasse e spese di gestione.

State Street Bank & Trust Co. chiese alla corte distrettuale di invalidare il brevetto detenuto dalla Signature Financial Group. Il brevetto fu revocato sulla base del fatto che, secondo la corte, consisteva in un puro calcolo matematico.

La Corte d'Appello Federale mutò tuttavia la decisione.

Secondo la corte, infatti, il software in questione era degno di ricevere protezione brevettuale fondamentalmente a causa della sua idoneità al raggiungimento di un'utilità pratica e di una effettiva applicazione industriale³⁷⁶.

Il *concrete, tangible, useful test*, viene applicato alle invenzioni di software, con una interpretazione tale da ridurre "l'analisi della section 101 del Patent Act (...) ad una mera valutazione sulla utilità pratica del trovato"³⁷⁷.

La corte ritenne che la trasformazione dei dati iniziali (che comunque rappresentano entità concrete, ossia somme di danaro), attraverso calcoli matematici, in un determinato risultato finale, rappresentasse un concerto ed utile risultato³⁷⁸.

Il principale prodotto di tale ragionamento risiede nella brevettabilità di un software rivendicato di per sé.

La decisione fu invero molto importante anche per un ulteriore motivo, che non è relativo, almeno direttamente, con l'oggetto di questa diserzione. Infatti tale sentenza decretò la brevettabilità dei "metodi commerciali", o metodi di *business*, elidendo quel divieto di brevettabilità che era stato portato avanti negli anni dalle corti minori³⁷⁹.

1.6. Bilski v. Kappos: il *machine or transformation test*

La pronuncia della Corte Suprema nel caso Bilski v. Kappos³⁸⁰ è di fondamentale importanza perché trattasi di una decisione in cui la Corte riprende in mano la materia dopo circa trent'anni. In questo lasso di tempo il CAFC ha rielaborato la teoria del *concrete, tangible, useful test*, portandola, con

³⁷⁶ State Street Bank & Trust Co. v. Signature Financial Group, cit., The "Mathematical Algorithm" Exception: "Unpatentable mathematical algorithms are identifiable by showing they are merely abstract ideas constituting disembodied concepts or truths that are not "useful." From a practical standpoint, this means that to be patentable an algorithm must be applied in a "useful" way".

³⁷⁷ AREZZO, Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa, cit., 112.

³⁷⁸ State Street Bank & Trust Co. v. Signature Financial Group, cit. The "Mathematical Algorithm" Exception: "Today, we hold that the transformation of data, representing discrete dollar amounts, by a machine through a series of mathematical calculations into a final share price, constitutes a practical application of a mathematical algorithm, formula, or calculation, because it produces "a useful, concrete and tangible result"--a final share price momentarily fixed for recording and reporting purposes and even accepted and relied upon by regulatory authorities and in subsequent trades".

³⁷⁹ Per un approfondimento sulla sentenza in oggetto, utilizzando la chiave di lettura dei *business methots*, v. GATTEI, Il brevetto di software, guida teorico pratica, pp. 175-179; NARD, The Law of Patents, pp. 140-160.

³⁸⁰ Bilski v. Kappos, 561 U.S. 593 (2010), in Rete: <https://www.supremecourt.gov/opinions/09pdf/08-964.pdf>

la decisione *State Street Bank & Trust Co. v. Signature Financial Group*, alla sua massima espansione. Dopo tale sentenza, infatti, si è di fatto aperta la strada alla brevettabilità, a volte indiscriminata, delle invenzioni di software.

Questo fenomeno di proliferazione di brevetti di scarsa o quantomeno dubbia rilevanza tecnologica e pratica è stata fortemente criticata, soprattutto dal mondo accademico³⁸¹.

Tali elementi portarono gli operatori del settore quantomeno a sperare in una rimodulazione della materia ad opera della Corte Suprema.

Per comprendere la decisione bisogna tuttavia illustrare quella che fu la decisione del CAFC sul caso in oggetto³⁸². La Corte d'Appello, in tale frangente, abbandonò il *useful, concrete and tangible result test*, così come risultante dalla elaborazione di *State Street*, inaugurando un nuovo metodo o criterio di valutazione: il *machine or transformation test*.

Secondo questo nuovo standard di analisi, la brevettabilità di un software deve essere verificata applicando due criteri o vagli tra lo alternativi. Da un lato si deve verificare il legame sussistente tra procedimento e hardware programmato, dall'altro si deve vagliare l'idoneità del programma a mettere in atto un procedimento volto alla trasformazione o modifica di un determinato oggetto.

Si nota come vengono uniti in un unico criterio quelli che sono stati i risultati *de facto* prodotti dai precedenti *In re Alappat*, ossia la brevettabilità del software stante la materialità dell'invenzione, e *State Street*, ossia la brevettabilità del metodo commerciale, stante l'utilità pratica del procedimento³⁸³. Criteri che già prima della decisione in oggetto potevano dirsi alternativi, ma comunque legati dal *concrete, tangible, useful test*.

È necessario effettuare sin da ora una disambiguazione. Si è infatti parlato, e nel proseguo del paragrafo si parlerà, di procedimento e di metodi di business. Si potrebbe quindi pensare che questa sentenza non ha nulla a che fare con l'oggetto di questa ricerca: il software. In realtà non è così, infatti: "*subject matter category delineation is notoriously elusive. There is no clear division between "software" and "nonsoftware" patents, or a computer system that implements a "business method" and one that implements another type of process, or between "technological" and "nontechnological" inventions*"³⁸⁴.

L'invenzione di *Blisky* era un metodo di business, che, attraverso un determinato procedimento, trasformava certi dati iniziali, per ottenere un risultato finale. Guardando alla sostanza, trattasi di un certo algoritmo di calcolo

³⁸¹ V. *supra*, cap. II, par. 2.

³⁸² *In re Bilski*, 545 F.3d 943, 88 U.S.P.Q.2d 1385 (Fed. Cir. 2008).

³⁸³ Per un approfondimento v. S. FUSCO, *Is In re Bilski a Déjà Vu?*, Stanford Tech. Law Rev., P1, 2009, in Rete: <http://stlr.stanford.edu/pdf/fusco-bilski-deja-vu.pdf>.

³⁸⁴ M. A. LEMLEY, M. RISCH, R. P. SICHELMAN, R. P. WAGNER, *Life after Bilsky*, 63 *Stan. Law. Rev.* 1315, 2011, cit., 1327.

implementato su di un programma per elaboratore. La decisione in questione è quindi perfettamente attinente all'ambito della ricerca³⁸⁵.

Il nuovo criterio suscitò da subito molte perplessità, stante soprattutto l'incertezza circa l'interpretazione del criterio del *machine*³⁸⁶. Se si interpreta il test in maniera lata, sarebbe infatti possibile implementare finanche un qualsiasi algoritmo (e non software), su una macchina, per potere accedere al brevetto.

Il caso giunse in esame alla Corte Suprema, che, il primo giugno 2009, accolse il ricorso relativo alla sentenza *In re Bilski*.

Il test elaborato dal CAFC fu ridotto ad un mero "indizio", in grado di fornire elementi circostanziali con cui valutare la brevettabilità del trovato³⁸⁷.

La Corte rileva come l'elevazione di detto standard ad unico criterio, andrebbe in senso contrario alla legge e specifica come le corti inferiori non dovrebbero interpretare la legge aggiungendo limitazioni ed eccezioni non espressamente previste dal legislatore³⁸⁸. La corte adotta quindi una interpretazione rigida, basata sulla lettera della legge³⁸⁹. In effetti il test avrebbe avuto l'effetto di considerare un procedimento brevettabile "*solo lì dove legato ad un macchinario ovvero in grado di trasformare un oggetto in un bene diverso*"³⁹⁰, in contrasto, almeno dal punto di vista letterale, con la legge³⁹¹.

Secondariamente la Corte fa leva su di argomentazioni più ampie.

I giudici ritengono infatti il criterio formulato dalla Corte d'Appello inadatto a valutare la brevettabilità dei procedimenti, stante il nuovo contesto dell'*Information Age* in cui gli stessi insistono.

³⁸⁵ La linea di confine tra un "metodo di business" ed un "programma per elaboratore" è alquanto labile, stante appunto il medesimo oggetto su cui le stesse possono insistere. Se un metodo di business è implementato su di un computer, allora lo stesso è nei fatti un software. Un metodo di business "puro" sarebbe semplicemente una idea astratta; da qui l'introduzione della categoria dei *business methods*, ad opera delle corti inferiori statunitensi, nell'ambito del non brevettabile, con lo scopo pratico di ridurre le rivendicazioni in oggetto. V. AREZZO, *La brevettabilità del software e dei metodi commerciali elettronici nella giurisprudenza dell'Ufficio Europeo Brevetti*, pp. 11-13.

³⁸⁶ Su tutti v. LEMLEY, RISCH, SICHELMAN, WAGNER, *Life after Bilsky*, cit. pp. 1322-1323: "*The machine or transformation test is problematic. It contains a number of ambiguities, leads to some bizarre results, and poorly tracks the stated goal of preventing the patenting of abstract ideas. (...) If the machine or transformation test were limited to process claims, it would generally be easy to draft around; most computer-implemented process claims can be rewritten as a machine or article of manufacture claims*".

³⁸⁷ *Bilski v. Kappos*, Cit., b), p. 2: "*The machine-or-transformation test is not the sole test for patent eligibility under §101. The Court's precedents establish that although that test may be a useful and important clue or investigative tool, it is not the sole test for deciding whether an invention is a patent-eligible 'process' under §101*".

³⁸⁸ *Ibidem*, cit., b), p. 2: "*the Federal Circuit violated two principles of statutory interpretation: Courts 'should not read into the patent laws limitations and conditions which the legislature has not expressed'*".

³⁸⁹ Per una critica che muove da questo punto v. P. S. MENELL, *Forty Years of Wondering in the Wilderness and No Closer to the Promised Land: Bilski's Superficial Textualism and the Missed Opportunity to Return Patent Law to Its Technology Mooring*, 63 *Stan. L. Rev.* 1289 (2010). In rete: <http://scholarship.law.berkeley.edu/facpubs/1303>.

³⁹⁰ AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 209.

³⁹¹ Cfr. 35 U.S. Code § 100 - Definitions, b) "*The term 'process' means process, art or method, and includes a new use of a known process, machine, manufacture, composition of matter, or material.*"

Nello spiegare come la repentina innovazione della tecnica deve portare all'utilizzazione di un approccio il più elastico possibile³⁹², la Corte in parte contraddice l'assunto, dell'interpretazione letterale della legge, da cui aveva mosso i passi³⁹³.

Una critica all'impostazione adottata dalla Corte, può essere mossa sottolineando come questa elasticità possa essere raggiunta solo dando la possibilità alle corti inferiori di interpretare il diritto alla luce delle esigenze del periodo di riferimento³⁹⁴.

Esigenze che a oggi, negli Stati Uniti, si muovono dalla necessità di restringere l'ambito del brevettabile. Il criterio utilizzato dalla Corte D'Appello non convinse tuttavia la Corte Suprema, nonostante si fregiasse di avere il pratico scopo di limitare l'ampiezza della tutela brevettuale³⁹⁵.

Infatti, una rivendicazione legata a quel preciso apparato, piuttosto che relativa a quel determinato processo di trasformazione, avrebbe consentito la non estensione della tutela, della medesima idea collegata ad una macchina diversa o ad una trasformazione differente³⁹⁶.

Mossa tali considerazioni, come fu risolto il caso dalla Corte?

Anzitutto bisogna rilevare come l'innovazione in questione consistesse in un metodo commerciale. La domanda di brevetto insisteva infatti in un metodo utilizzabile per il calcolo del prezzo fisso di una fornitura energetica. Nei contratti di energia a canone fisso, i clienti pagano prezzi mensili per il loro futuro consumo di energia, sulla base del loro passato utilizzo energetico. I prezzi mensili rimangono uguali, indipendentemente da quanta energia utilizzino. Il software calcolava i consumi passati, li raffrontava al totale pagato e prospettava un preventivo.

Con State Street fu decretata la brevettabilità dei metodi commerciali, e la Corte, nel caso in oggetto, nel negare il brevetto alla innovazione in questione,

³⁹² Bilski v. Kappos, cit., 8: "*But times change. Technology and other innovations progress in unexpected ways*".

³⁹³ Così anche AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, pp. 208-211.

³⁹⁴ MENELL, *Forty Years of Wondering in the Wilderness and No Closer to the Promised Land: Bilski's Superficial Textualism and the Missed Opportunity to Return Patent Law to Its Technology Mooring*, cit., 1313-1314: "*The Supreme Court's textualist turn has worked a great disservice to the promotion of progress in the useful arts and preserving free enterprise. By replacing the patent system's rich jurisprudential tradition with dictionary definitions, the Bilski decision fundamentally misconstrues the contours of patentable subject matter and undermines the dynamic process that has guided the evolution of patent law since the nation's founding (...) a flexible and evolving body of common law that is sensitive to history, statutory evolution, constitutional constraints, and understanding of modern science and technology*".

³⁹⁵ In re Bilski, cit., 959: "*A claimed process involving a fundamental principle that uses a particular machine or apparatus would not pre-empt uses of the principle that do not also use the specified machine or apparatus in the manner claimed. And a claimed process that transforms a particular article to a specified different state or thing by applying a fundamental principle would not preempt the use of the principle to transform any other article, to transform the same article but in a manner not covered by the claim, or to do anything other than transform the specified article*".

³⁹⁶ Tuttavia, tale circostanza potrebbe spingere gli inventori a brevettare l'idea alla base del trovato attraverso quante più apparecchiature possibili ovvero attraverso la trasformazione o modificazione di quanti più oggetti possibili. In questo modo non solo si vanificherebbe il supposto restringimento dell'ambito della tutela, ma si avrebbe una proliferazione di brevetti pressoché identici. In questo senso V. LEMLEY, RISCH, SICHELMAN, WAGNER, *Life after Bilsky*,

non si mosse nel chiaro senso di decretare la non aderenza dei metodi commerciali all'interno del *patentable subject matter*³⁹⁷. Utilizzò infatti un altro argomento, *“riportando alla luce il principio del divieto di brevettabilità delle idee astratte, ancorandolo alla dottrina della pre-emption”*³⁹⁸.

La dottrina in questione, partendo dall'assunto della non brevettabilità delle idee astratte, considera come tali anche quei procedimenti che in una certa misura implementano una conoscenza astratta, volta ad essere applicata non solo in quel determinato ambito industriale. In un certo senso si riafferma, attraverso un differente procedimento argomentativo, la decisione *Diamond v. Diehr* del 1981.

La Corte non definisce chiaramente cosa si debba intendere per “idea astratta”, ne tantomeno illustra come eseguire un siffatto test, onde guidare in tale direzione i giudici inferiori. Di contro lascia alle corti quel margine di discrezionalità idoneo sia a concedere il brevetto ad invenzioni meritevoli di tutela, sia a respingere quelle non-innovazioni di cui tanto viene denunciata la proliferazione nel mercato brevettuale statunitense.

La giustificazione di tale scelta può forse essere descritta con le stesse parole della Corte:

*“This Age puts the possibility of innovation in the hands of more people and raises new difficulties for the patent law. With ever more people trying to innovate and thus seeking patent protections for their inventions, the patent law faces a great challenge in striking the balance between protecting inventors and not granting monopolies over procedures that others would discover by independent, creative application of general principles. Nothing in this opinion should be read to take a position on where that balance ought to be struck.”*³⁹⁹

2. La giurisprudenza di EPO

2.1. Premessa: l'importanza delle linee guida

Nell'esaminare le più rilevanti decisioni di EPO, volte a definire i confini della tutela brevettuale del software, ritengo sia utile al ragionamento, una contestuale, seppur parziale, analisi delle odierne linee guida fornite dall'Ufficio Brevetti Europeo. Queste, infatti, nell'evidenziare come e quando un software possa essere meritevole di tutela, richiamano quelle sentenze che hanno contribuito a fornire quella visione della materia che oggi è confluita all'interno delle linee guida stesse. In questo modo è possibile ottenere un “vaglio di rilevanza” all'interno della vasta giurisprudenza di EPO.

Nel capitolo III, analizzando la disciplina EPC, si è già fatto riferimento alle linee guida⁴⁰⁰, onde comprendere come EPO abbia inteso interpretare la propria normativa. Si è già quindi parlato del necessario requisito che deve

³⁹⁷ Per una lettura del caso volta a sottolinearne le implicazioni nell'ambito dei *business methods v. AREZZO*, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, pp. 211-213.

³⁹⁸ *Ibidem*, cit., 214.

³⁹⁹ *Bilski v. Kappos*, cit., 10.

⁴⁰⁰ V. *supra*, cap. III, par. 2.4.

avere una invenzione di software: il carattere tecnico. Non si è tuttavia spiegato, ed è questo lo scopo di tale diserzione, come EPO sia arrivato ad una tale impostazione.

2.2. Il caso Vicom: la differenza tra un metodo matematico e un processo tecnico.

La prima rilevante decisione emanata dalla Commissione di Ricorso dell'European Patent Office⁴⁰¹, è relativa al caso Vicom⁴⁰².

La domanda di brevetto insisteva su un programma per elaboratore in grado di modificare le immagini, attraverso la modifica di determinati dati, relativi alle immagini stesse. L'immagine veniva digitalizzata ed "editata" mediante il cambiamento di determinati parametri, sotto forma di dati, in grado di controllare certi elementi, come per esempio la saturazione dei colori e i contorni dell'immagine. Apportate tali modifiche, queste venivano applicate all'immagine, mostrando sul monitor il risultato finale.

La Divisione di Esame esclude la brevettabilità del trovato sulla base dell'assunto che la rivendicazione insistesse su di un programma per elaboratore in quanto tale. Nello specifico si rilevò come l'applicazione del metodo, all'apparato *general-purpose*⁴⁰³, non apportasse alcun contributo tecnico.

La Commissione adottò tuttavia una prospettiva diversa.

Enunciò che la brevettabilità andasse riconosciuta alla invenzione oggetto del ricorso, sulla base dell'assunto che essa non insistesse sul metodo matematico in se, quando piuttosto sull'innovazione che, nel suo complesso, utilizza un metodo matematico.

Il passaggio è di fondamentale importanza ed è bene descriverlo con le parole della Commissione:

"A basic difference between a mathematical method and a technical process can be seen, however, in the fact that a mathematical method or a mathematical algorithm is carried out on numbers (whatever these numbers may represent) and provides a result also in numerical form, the mathematical method or algorithm being only an abstract concept prescribing how to operate on the numbers. No direct technical result is produced by the method as such. In contrast thereto, if a mathematical method is used in a technical process, that process is carried out on a physical entity (which may be a material object but equally an image stored as an electric signal) by some technical means implementing the method and provides as its result a certain change in that

⁴⁰¹ Nella struttura delineata dalla European Patent Convention, la Commissione di ricorso è quell'organo preposto a vagliare i ricorsi presentati contro le decisioni delle sezioni di esame. Quest'ultime hanno invece la competenza a decidere sulle domande di brevetto presentate. Si veda, EPC, art. 21.

⁴⁰² Vicom/Computer-related invention, T 0208/8, 15.07.1986, in Rete: <https://www.epo.org/law-practice/case-law-appeals/pdf/t840208ep1.pdf>.

⁴⁰³ Il trovato, era infatti in grado di funzione su sistemi hardware standard.

*entity. The technical means might include a computer comprising suitable hardware or an appropriately programmed general purpose computer*⁴⁰⁴.

Si distingue quindi tra un metodo fondato sulla matematica astratta, che non è in grado di apportare alcun contributo tecnico allo stato dell'arte, ed un procedimento tecnico che implementa un metodo matematico, idoneo invece ad avere applicazione industriale. In questo modo la tutela si estende solamente a quel procedimento tecnico, e non al metodo matematico in sé.

Deve sussistere una applicazione di tipo concreto e tangibile che porti a determinati risultati pratici.

La decisione fissa quindi molti punti di interesse centrale; anzitutto esplica la differenza tra software considerato di per sé ed innovazione attuata a mezzo di software. Quest'ultima è inquadrabile alla stregua di un processo tecnico perché idonea a produrre, mediante l'utilizzo di mezzi tecnici, e nel suo complesso, un risultato concreto, suscettibile di applicazione industriale⁴⁰⁵.

La decisione in esame è invero rilevante anche per un ulteriore motivo. Nello specificare come la presenza del contributo tecnico vada vagliata attraverso una analisi della invenzione nel suo complesso, si specifica come sia irrilevante la circostanza che lo step innovativo sia effettuato a mezzo di hardware o a mezzo di software⁴⁰⁶.

Per comprendere tale assunto si deve considerare lo stato della tecnica del periodo di riferimento. A metà degli anni ottanta erano presenti sul mercato sia computer *finite-purpose*, ossia adibiti a specifici compiti, sia computer *general-purpose*, ossia standard. In questi ultimi, in particolare, l'innovazione non risiede tanto nella componente hardware, che è appunto standard, quanto piuttosto nelle funzioni implementate attraverso software.

Se il carattere tecnico deve essere valutato guardando l'invenzione nel suo complesso, si stabilisce indirettamente l'idoneità del solo software ad apportare all'hardware un determinato contributo tecnico⁴⁰⁷.

⁴⁰⁴ Vicom/Computer-related invention, cit., 7-8.

⁴⁰⁵ Si registra quindi il superamento e della dottrina *point of novelty approach*, di matrice statunitense e già in precedenza richiamata (v. *supra*, cap. IV, par. 1.3.), e della *kerntheorie*, di matrice tedesca, già citata (v. *supra*, cap. III, par. 2.2.). Per un approfondimento in questo senso v. AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, pp. 119-127.

⁴⁰⁶ *Ibidem*, cit., 11: “a distinction between embodiments of the same invention carried out in hardware or in software is inappropriate as it can fairly be said that the choice between these two possibilities is not of an essential nature but is based on technical and economical considerations which bear no relationship to the inventive concept as such”.

⁴⁰⁷ Assunto che venne confermato nel caso Koch & Sterzel/X-Ray Apparatus, T 26/86, 21.05.1987, cit., 3.3., in Rete: <https://www.epo.org/law-practice/case-law-appeals/recent/t860026ep1.html>. La Commissione stabilì infatti che l'oggetto del brevetto può consistere in una invenzione composta da elementi tecnici e da elementi privi di tecnicità. Il trovato deve essere valutato nel suo complesso, ed è irrilevante la circostanza che la componente innovativa sia costituita dal solo programma per elaboratore: “The computer program used in a general-purpose computer is thus considered to be a program as such and hence excluded from patentability by Article 52(2)(c) EPC. But if the program controls the operation of a conventional general-purpose computer so as technically to alter its functioning, the unit consisting of program and computer combined may be a patentable invention”. Per una più ampia lettura del caso in oggetto v. AREZZO, *La brevettabilità del software e dei metodi commerciali elettronici nella giurisprudenza dell'Ufficio Europeo Brevetti*, pp. 21-24.

2.3 (segue): il carattere tecnico tra il *contribution approach* e il *problem-solution approach*

Tale decisione aprì la strada, negli anni successivi, all'applicazione del criterio del c.d. "*contribution approach*". Secondo tale metodologia di analisi l'invenzione, per essere considerata come tale, doveva produrre un contributo tecnico. La verifica di tale contributo da un lato avviene attraverso una analisi onnicomprensiva del trovato, dall'altra attraverso un giudizio sul risultato prodotto dall'interazione di quegli stessi elementi tecnici, o non tecnici, nuovi o acquisiti, che doveva essere tecnico per soddisfare i requisiti di brevettabilità. Ma di quali requisiti si sta parlando nello specifico?

Il problema di tale criterio risiede nella circostanza che tende a sovrapporre due distinti elementi di analisi: il primo, che insiste sulla identificazione del trovato all'interno del *patentable subject matter*⁴⁰⁸; il secondo, che insiste sulla verifica del necessario requisito di originalità⁴⁰⁹.

Si ha una confusione con il criterio del *problem-solution approach*. Tale criterio è usato da EPO per sottoporre i trovati ad un vaglio di originalità⁴¹⁰.

Analizzando le linee guida si apprende che esso consta di tre passaggi: anzitutto si deve individuare la tecnica esistente che più si avvicina all'invenzione in oggetto, secondariamente si deve stabilire l'obiettivo problema tecnico da risolvere e, infine, si deve considerare se l'invenzione rivendicata sarebbe risultata ovvia ad una persona specializzata nel settore di riferimento⁴¹¹.

Il criterio sopra descritto risulta molto utile se applicato ad una specifica categoria di invenzioni: le c.d. *mixed-type-claim-feature*.

In questa tipologia di trovati si ha una commistione di elementi tecnici e non tecnici. Nonostante non sia a priori esclusa la brevettabilità di questo tipo di invenzioni, che sovente investono le invenzioni attuate a mezzo di elaboratore⁴¹², viene stabilito che lo step inventivo debba essere vagliato con riguardo solo ai quegli elementi tecnici brevettabili:

" (...) *features which do not contribute to the technical character of the invention cannot support the presence of an inventive step. Such a situation may arise, for instance, if a feature contributes only to the solution of a non-technical problem, e.g. a problem in a field excluded from patentability.*"⁴¹³

Si noti come nei fatti l'applicazione di tale criterio sia molto affine alla teoria dei *mental steps*⁴¹⁴.

⁴⁰⁸ V. art. 52 EPC.

⁴⁰⁹ V. art. 56 EPC.

⁴¹⁰ Guidelines for Examination in the EPO, Novembre 2016, Parte G, cap. VII, cit. p. 2: "*In order to assess inventive step in an objective and predictable manner, the so-called 'problem-and-solution approach' should be applied*".

⁴¹¹ *Ibidem*, pp. 2-5.

⁴¹² *Ibidem*, cit., 5, 6: "*It is legitimate to have a mix of technical and non-technical features appearing in a claim, as is often the case with computer-implemented inventions*".

⁴¹³ *Ibidem*, cit., 6.

⁴¹⁴ V. *supra*, cap. IV, par. 1.2, nota n. 355.

Se ai sensi del *contribution approach* l'invenzione deve essere guardata nel suo complesso, il *problem-solution approach*, considera essenziale il sezionamento della stessa nei suoi elementi tecnici e non tecnici.

La circostanza che, alla lettera della EPC, il software sia un elemento non tecnico, e quindi non brevettabile di per sé, ha portato, nelle decisioni post-Vicom, a risultati in parte contraddittori⁴¹⁵.

La situazione tuttavia cambiò nel corso degli anni successivi grazie a due importanti decisioni, volte da un lato, a riconoscere al software la caratteristica della tecnicità e, dall'altro, a rimodulare le modalità di verifica del carattere tecnico dell'invenzione.

2.4. Le decisioni gemelle IBM: il software rivendicato di per sé

Le decisioni in oggetto⁴¹⁶ ricoprono un posto di fondamentale importanza nella giurisprudenza di EPO. Non a caso, la decisione Ibm/Computer Program Product I, è la più citata all'interno delle linee guida che devono seguire le Divisioni d'Esame⁴¹⁷.

Ponendosi in un'ottica di rottura con i precedenti della materia, la Commissione di Ricorso sostenne la brevettabilità del software rivendicato di per sé, qualora lo stesso sia dotato delle capacità di produrre un effetto tecnico ulteriore, una volta eseguito su di un elaboratore.

Se prima i brevetti non erano rivolti al software, ma piuttosto ad innovazioni contenenti una parte software, ora si ha un affrancamento della "parte morbida" dalla "parte dura", che permette, soddisfatti determinati requisiti, l'innalzamento del software a *patentable subject matter*.

Lo snodo centrale della decisione risiede nella constatazione -se vogliamo, concettualmente e tecnicamente banale, come in parte ammette anche la stessa commissione⁴¹⁸- che l'hardware è un supporto materiale su cui opera il programma, espletando le funzioni cui è preposto.

Queste, infatti, risiedono nel software che è il vero elemento inventivo. Se infatti è vero che il programma per elaboratore è in grado di produrre un risultato tecnico, fisico e tangibile solo una volta eseguito da un computer, è

⁴¹⁵ Su tutti cfr. Ibm/Text processing, T 0038/86, 14.02.1989, in Rete: <https://www.epo.org/law-practice/case-law-appeals/recent/t860038ep1.html>, Ibm/Editable document form, T 0110/90, 15.04.1993, in Rete <https://www.epo.org/law-practice/case-law-appeals/recent/t970935eu1.html>. Per un approfondimento sulle decisioni post-Vicom v. AREZZO, *La brevettabilità del software e dei metodi commerciali elettronici nella giurisprudenza dell'Ufficio Europeo Brevetti*, pp. 21-29; AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, pp. 123-134; GATTEI, *Il brevetto di software, guida teorico pratica*, pp. 112-124.

⁴¹⁶ Ibm/Computer Program Product I, T 1173/97, 01.07.1998, in Rete: <https://www.epo.org/law-practice/case-law-appeals/pdf/t971173ex1.pdf>; Ibm/Computer Program Product II, T 0935/97, 04.02.1999, in Rete: <http://www.epo.org/law-practice/case-law-appeals/pdf/t970935eu1.pdf>.

⁴¹⁷ V. Guidelines for Examination in the EPO, Novembre 2016, Parte G, cap. II, pp. 5 ss.

⁴¹⁸ Ibm/Computer Program Product I, cit., 20: "It is self-evident that in this instance the basic idea underlying the invention resides in the computer program. It is also clear that, in such a case, the hardware on which the program is intended to run is outside the invention, ie the hardware is not part of the invention".

altrettanto vero che tali effetti sono comunque intrinseci nel software, che possiede la latente potenzialità di produrre questi effetti⁴¹⁹.

Questi effetti, possono anche essere classificati come tecnici, da qui la capacità di un software di implementare (in via potenziale) quell'effetto ulteriore, che è motivo di brevettabilità⁴²⁰.

Dati questi elementi: *"the Board sees no good reason for distinguishing between a direct technical effect on the one hand and the potential to produce a technical effect, which may be considered as an indirect technical effect, on the other hand. A computer program product may therefore possess a technical character because it has the potential to cause a predetermined further technical effect in the above sense. According to the above, having technical character means not being excluded from patentability under the 'as such' provision pursuant to Article 52(3) EPC"*⁴²¹.

Ciò significa che un programma per elaboratore, che ha il potenziale di provocare (pre)determinati effetti tecnici ulteriori, è, in linea di principio, non escluso dalla brevettabilità.

Rimane da chiarire un elemento, ossia cosa intenda il Board per effetto tecnico⁴²². L'importanza è centrale perché idoneo a distinguere *"programs for computers with a technical character, from programs for computers as such"*⁴²³.

L'effetto tecnico va ricercato in quel *"plus"* che deriva dall'esecuzione delle istruzioni del programma sul computer, istruzioni che si devono tradurre in un determinato risultato tecnico, tale da non ridursi alla mera e necessaria interazione tra programma e macchina.

È da sottolineare anche la circostanza, che trova conferma nelle linee guida odierne, che il carattere tecnico deve essere valutato senza riguardo allo stato della tecnica. Questo perché l'accertamento in questione ha in oggetto la brevettabilità del bene alla luce dell'articolo 52 EPC, e non il vaglio di originalità del trovato, da effettuarsi con un successivo step⁴²⁴.

Sulla base di queste considerazioni, le linee guida di EPO, richiamando la decisione in oggetto, recitano:

"A computer program claimed by itself is not excluded from patentability if it is capable of bringing about, when running on or loaded into a computer, a further technical effect going beyond the 'normal' physical interactions between the program (software) and the computer (hardware) on which it is run (...).

⁴¹⁹ Ibidem, cit., 21: *"Every computer program product produces an effect when the program concerned is made to run on a computer. The effect only shows in physical reality when the program is being run. Thus the computer program product itself does not directly disclose the said effect in physical reality. It only discloses the effect when being run and consequently only possesses the "potential" to produce said effect"*.

⁴²⁰ Ibidem, cit., 21: *"This effect may also be technical, in the sense explained in reason 6, in which case it constitutes the 'further technical effect' mentioned there. This means that a computer program product may possess the potential to produce a "further" technical effect"*.

⁴²¹ Ibidem, cit., 21.

⁴²² V. ibidem, pp.15-17.

⁴²³ Ibidem, cit., 15.

⁴²⁴ Guidelines for Examination in the EPO, Novembre 2016, Parte G, cap. II, cit., 5: *"Technical character should be assessed without regard to the prior art (see T 1173/97, confirmed by G 3/08). Features of the computer program itself (see T 1173/97) as well as the presence of a device defined in the claim (see T 424/03 and T 258/03) may potentially lend technical character to the claimed subject-matter as explained below"*.

*Whether a computer program can contribute to the technical character of the claimed subject-matter is frequently an issue separate and distinct from the technical character of the hardware components which may be defined in order to execute the computer program. When a computer program produces a further technical effect (T 1173/97), it is by itself considered technical and not excluded*⁴²⁵.

2.5. La rimodulazione del carattere tecnico: il caso Hitachi

L'innovazione oggetto del ricorso insisteva sulla rivendicazione di un sistema di vendita online, attuato attraverso l'asta olandese. L'oggetto della domanda di brevetto consisteva in un programma per elaboratore, deputato ad impartire quelle istruzioni descritte dal metodo commerciale.

La Divisione d'Esame negò la tutela, stante la considerazione che l'oggetto della domanda era un metodo commerciale "in quanto tale".

Il Board, nell'affrontare il caso, cristallizzò in tale decisione⁴²⁶ come la verifica del carattere tecnico, necessaria per poter classificare il trovato come "invenzione" ai sensi dell'art. 52 EPC, debba essere preliminare ad ogni altro tipo di analisi.

Nel ribadire tale concetto viene separato il carattere tecnico dalla prova del contributo tecnico, analisi quest'ultima che attiene ora alla sussistenza della originalità del trovato. Tale approccio restringe il campo degli oggetti non brevettabili, relegando la categoria ai soli concetti astratti, privi di qualsivoglia implicazione tecnica⁴²⁷.

Tale metodo di analisi è confluito nelle linee guida, la dove si stabilisce che:

*"If claimed subject-matter relating to a computer program does not have a technical character, it should be rejected under Art. 52(2) and (3). If the subject-matter passes this test for technicality, the examiner should then proceed to the questions of novelty and inventive step"*⁴²⁸.

Si potrebbe ora pensare che la decisione in oggetto metta da parte il criterio di analisi del *contribution approach*. Di fatto è così, ma relativamente a quella analisi preliminare da effettuarsi per poter considerare la rivendicazione brevettabile alla luce dell'articolo 52 EPC. In questa fase si deve prendere in considerazione la sussistenza del mero carattere tecnico, anche se il Board non si sbilancia fornendone una definizione, né uno specifico test per vagliarne la

⁴²⁵ Guidelines for Examination in the EPO, Novembre 2016, Parte G, cap. II, cit., 5-6.

⁴²⁶ Auction method/HITACHI, T 258/03, 21.04.2004, in Rete: <https://www.epo.org/law-practice/case-law-appeals/pdf/t030258ep1.pdf>.

⁴²⁷ Nelle parole del Board, *ibidem*, cit., 6: "What matters having regard to the concept of "invention" within the meaning of Article 52(1) EPC is the presence of technical character which may be implied by the physical features of an entity or the nature of an activity, or may be conferred to a non-technical activity by the use of technical means. In particular, the Board holds that the latter cannot be considered to be a non-invention "as such" within the meaning of Article 52(2) and (3) EPC. Hence, in the Board's view, activities falling within the notion of a non-invention "as such" would typically represent purely abstract concepts devoid of any technical implications".

⁴²⁸ Guidelines for Examination in the EPO, Novembre 2016, Parte G, cap. II, cit., 6.

sussistenza. Si limita ad affermare che il carattere tecnico può essere implicito nelle caratteristiche fisiche di un'entità o nella natura di un'attività o, ancora, può essere conferito ad un'attività non tecnica mediante l'utilizzo di mezzi tecnici⁴²⁹.

Ma la concessione del brevetto è legata alla sussistenza di ulteriori requisiti:

*“Needless to say, however, this does not imply that all methods involving the use of technical means are patentable. They still have to be new, represent a non-obvious technical solution to a technical problem, and be susceptible of industrial application.”*⁴³⁰

Che sia il vaglio di queste ultime condizioni a rappresentare il vero “filtro” in grado di individuare il software meritevole di tutela?

2.6. “La via dell’originalità”? Il parere G 3/08 e i problemi aperti

Da una disamina delle decisioni successive al caso appena analizzato, si nota come l’allentamento delle maglie nell’analisi della sussistenza dell’invenzione, non ha fatto corrispondere un siffatto atteggiamento, con riguardo alla valutazione dei restanti requisiti di brevettabilità.

Analizzando alcuni casi post-Hitachi si può notare come, accertata la sussistenza del carattere tecnico, ci sia poi stato diniego del brevetto, in base alla circostanza che il trovato era da considerarsi ovvio alla stato della tecnica⁴³¹.

Si ha quindi una certa rivisitazione del metodo di analisi delle rivendicazioni, che vede ora la sua parte centrale, non tanto nell’accertamento del carattere tecnico, quanto piuttosto, nella sussistenza degli altri requisiti, ed in particolare modo di quello relativo alla originalità.

Il criterio del *contribution approach* sembra essere di fatto confluito nella metodologia di analisi della originalità dell’invenzione, aggiungendosi così al *problem solution approach*⁴³². I criteri di cui si lamentava la sovrapposizione, sembra che ora, operando in sinergia all’interno di un medesimo ambito di analisi, abbiano trovato una coerenza sistematica.

Coerenza che tuttavia non sembra ritrovarsi nell’applicazione di tali criteri, stante la sussistenza di determinate decisioni che, in contrasto con la posizione

⁴²⁹ V. Auction method/HITACHI, 4.5, p. 6.

⁴³⁰ *Ibidem*, cit., 4.6, p. 6.

⁴³¹ V. Citibank/Integrated account, T 368/05, 15.05.2007. L’invenzione in questione consisteva in un metodo finanziato in grado di agevolare la gestione di determinati servizi bancari online. La Commissione stabilì che l’invenzione non apportava alcun contributo alla tecnica di riferimento, trattandosi invero della semplice applicazione di tecniche di programmazione di routine. In Rete: <https://www.epo.org/law-practice/case-law-appeals/pdf/t050368eu1.pdf>. In Game machine/GAMEACCOUNT, T 1543/06, 29.06.2007, l’invenzione rendeva possibile un determinato tipo di interazione tra giocatore e macchina, rendendo disponibile al primo, sulla base delle mosse disponibili, una sorta di illustrazione delle possibili giocate. La Commissione spiegò come per un soggetto perito in quel determinato ambito la soluzione adoperata risultasse ovvia. In Rete: <https://www.epo.org/law-practice/case-law-appeals/pdf/t061543eu1.pdf>.

⁴³² Di questa linea di pensiero AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, pp. 153 ss.

assunte nei casi sopracitati, rinvengono il gradiente di originalità anche là dove pare non esserci⁴³³.

Nonostante l'Enlarged Board of Appeal, con il parere G 3/08⁴³⁴, si sia prodigata a rispondere a determinati quesiti⁴³⁵, riguardanti gli snodi problematici della materia, su tutti l'apparente, o palese, contraddittorietà tra determinate decisioni, permangono determinati nodi irrisolti.

Sul tema l'EBA, pur rigettando dal punto di vista formale il *referral*, di fatto fornì risposta alle domande prospettatagli, risposte che costituiscono piuttosto una rilettura organica delle precedenti decisioni (in parte già analizzate in questo stesso capitolo), volte a fornire una parvenza di coerenza e rigore sistematico alla materia⁴³⁶.

Ma i problemi, in parte evidenziati dai quesiti posti all'Enlarged Board of Appeal, permangono.

Anzitutto è dubbio il significato da attribuire al termine "tecnico". Venne infatti confermato, anche dall'EBA, ciò che fu affermato in *Hitachi*, ossia il fatto che il carattere tecnico possa essere dedotto dalle caratteristiche fisiche di una certa entità, dalla natura di una certa attività o ancora potrebbe essere conferito mediante l'utilizzo di mezzi tecnici anche ad una attività non tecnica. Questa è di fatto una "non-definizione" di carattere tecnico, non in grado di fornire un preciso metodo di verifica della caratteristica⁴³⁷.

Secondariamente non è chiaro, e non è stato chiarito, in cosa debba consistere nella prassi delle Divisioni d'Esame, il test per verificare l'originalità dell'invenzione. Test che riveste un ruolo molto importante, stante la riduzione a mera formalità della sussistenza del carattere tecnico dell'invenzione.

⁴³³ Si veda a titolo esemplificativo *Video game/KONAMI*, T 0928/03, 02.06.2006, dove l'invenzione fu considerata come una non banale implementazione di tipo tecnico volta a risolvere il problema della scarsa visibilità sullo schermo di gioco, migliorando quindi la fruibilità. Invero il problema in questione insisteva negli indicatori grafici, nella fattispecie rappresentati da triangoli, che, nel gioco di calcio in oggetto, servivano ad identificare il giocatore controllato dall'utente. Questi risultavano fastidiosi, così si decise di sostituirli con un anello posto ai piedi del giocatore controllato. Si denota come invero la soluzione tecnica apportata, consti di una mera modifica visiva, che muove dallo stesso stato della tecnica adottato nella precedente versione del gioco. In Rete: <https://www.epo.org/law-practice/case-law-appeals/pdf/t030928eu1.pdf>.

⁴³⁴ *Programs for computers*, G 0003/08, 02.05.2010, in Rete: <http://www.epo.org/law-practice/case-law-appeals/pdf/g080003ex1.pdf>.

⁴³⁵ Tali quesiti sono stati posti dall'allora Presidente di EPO, Alison Brimelow, il 22.10.2008, sulla base della facoltà riconosciutogli ex art. 112, comma 1, lett. B) EPC. G 0003/08, *Referral by the President of the EPO in relation to a point of law pursuant to Article 112(1)(b) EPC*, 16.10.2009, in Rete: <https://www.epo.org/law-practice/case-law-appeals/pdf/g080003eu1.pdf>.

⁴³⁶ Per una analisi critica alla decisione dell'EBA v. J. PILA, *Software Patents, Separation of Powers, and Failed Syllogisms: A Cornucopia from the Enlarged Board of Appeal of the European Patent Office*, *The Cambridge Law Journal*, Vol. 70, No. 1 (March 2011), pp. 203-228.

⁴³⁷ Per una trattazione completa sul requisito del carattere tecnico si segnala R. FRUSCALZO, *Il carattere tecnico dell'invenzione*, Università degli studi di Parma, Dottorato di ricerca in diritto commerciale: proprietà intellettuale e concorrenza, Ciclo XXIV, 2009, in Rete: http://dspace.unipr.cineca.it/bitstream/1889/2148/1/Tesi_dottorato_%20Riccardo%20Fruscalzo.pdf.

RIFLESSIONI CONCLUSIVE

Prospettive di riforma della materia

SOMMARIO: *Premessa* - 1. La privativa brevettuale come parte di una tutela stratificata: il livello di tutela del software - 2. Il vero ostacolo al riconoscimento di una piena tutela brevettuale al software nell'Unione Europea: la concorrenza internazionale e l'accesso alla tutela - 3. L'introduzione di regole ad hoc nel diritto brevettuale - 4. L'opportunità di separare le tutele

Premessa

Il tema affrontato in questo lavoro è sicuramente ampio e complesso. Si è cercato di prospettare l'argomento in modo tale da rendere consapevole il lettore dei tanti elementi che rendono la materia particolarmente delicata.

Credo tuttavia che il ragionamento fin qui svolto possa già offrire alcune chiavi di lettura utili a sciogliere nodi problematici di rilievo. In questa parte finale del lavoro si cercherà da un lato di evidenziare quei particolari problemi che investono la materia in questione, e, dall'altro si prospetteranno degli spunti di riflessione in un'ottica di miglioramento della disciplina.

L'annosa questione circa la tutela del software non risiede tanto nella scelta tra tutela brevettuale e tutela autoriale, quanto nell'oggetto delle rispettive tutele, nel coordinamento delle stesse, e nei margini più o meno ampi della concessione del brevetto.

Si sono analizzate le tendenze legislative e giurisprudenziali, comprendendo le difficoltà riscontrate e le soluzioni trovate al fine di concedere al software una tutela anche brevettuale. Non tutti i problemi sono tuttavia stati superati. Ritengo sia quindi utile evidenziare tali aspetti, utilizzando anche un atteggiamento di tipo propositivo.

In tale disamina non ci si può porre su un piano meramente astratto, di diritto "puro", al di sopra del contesto storico in cui l'analisi insiste.

Contesto storico che va valutato con riguardo a diversi elementi.

In primis, si deve considerare il livello tecnologico, ossia "lo stato dell'arte", cercando quindi anche di prevedere in quale direzione si muoveranno negli anni avvenire le invenzioni di software.

Secondariamente, si deve considerare l'impatto dell'industria di software sull'economia; l'industria ICT è molto rilevante e la competizione tra Stati nazionali, volti a salvaguardare le proprie aziende, è un elemento che sicuramente influenza la materia.

Infine, è d'uopo porsi in un'ottica di possibili mutamenti istituzionali, su tutti si fa riferimento al probabile, o per meglio dire eventuale, per non dire possibile, o ancora auspicabile, passaggio al sistema del brevetto unitario europeo.

1. La privativa brevettuale come parte di una tutela stratificata: il livello di tutela del software

Analizzando le tendenze legislative statunitensi, europee ed internazionali, si è rimarcato come la tutela del software, sia essa realizzata esclusivamente mediante copyright, o anche mediante brevetto, sia congegnata per raggiungere un alto livello di protezione. Le ragioni di tale scelta sono principalmente due. La prima è insita nella natura del bene; come già si è dato atto nel primo capitolo, il software è un bene particolare perché, sia esso incorporato o meno su un supporto tangibile, richiede costi di elaborazione e creazione molto alti, mentre invece i costi di riproduzione di una copia sono prossimi allo zero⁴³⁸.

Il rischio di chi innova, di chi crea, di chi fa impresa, è di vedere vanificato il proprio lavoro, ed il proprio profitto, da utilizzi non consentiti del programma.

Nello sviluppo di un programma per elaboratore, il fine ultimo sovente si sostanzia nella distribuzione delle copie, al fine di ricavarne un certo utile. La circostanza interessa non solo all'autore, ma anche ai singoli Stati. Questi si prodigano per predisporre strumenti di tutela adeguati, in grado di soddisfare le esigenze delle imprese operanti nel mercato nazionale, e ciò accade non solo per facilitare le aziende nostrane, ma, anche, al fine di attirare investimenti stranieri⁴³⁹.

La seconda ragione è strettamente collegata alla prima, perché il fine ultimo della tutela è incentivare la creatività proteggendo gli autori. In questo modo si stimola l'innovazione, a vantaggio dell'intera società⁴⁴⁰.

Avendo analizzato gli strumenti attraverso i quali un software viene protetto⁴⁴¹, è possibile rilevare il livello di tutela del bene in questione, al fine di valutare se esso sia o meno compatibile con il bilanciamento di interessi da cui muovono i diritti di proprietà intellettuale.

Questi, come si è visto, sono finalizzati prima di tutto al progresso, allo sviluppo della cultura e dell'innovazione. Il diritto di esclusiva concesso all'autore è un incentivo alla creazione. Tale esclusiva, tuttavia, non deve trasformarsi in un monopolio illimitato, perché in tale circostanza i vantaggi che ne deriverebbero alla collettività sarebbero annullati⁴⁴².

⁴³⁸ V. *supra*, cap. I, par. 2.

⁴³⁹ V. Direttiva 2009/24/CE, considerando n. 2; Direttiva 2001/29/CE considerando n. 10; v. *supra* cap. I, par. 3.

⁴⁴⁰ V. Direttiva 2001/29/CE considerando n. 9 e 11; Costituzione degli Stati Uniti d'America, articolo 1, sez. 8, clausola 8.

⁴⁴¹ V. *supra*, cap. II.

⁴⁴² Per una disamina approfondita degli interessi in conflitto nel moderno diritto industriale v. GHIDINI, *Profili evolutivi del diritto industriale*, "Il sistema, gli interessi, i conflitti", pp. 1-71. Cfr. V. SHIVA, *Il mondo sotto brevetto*, Feltrinelli, Milano, 2002 pp. 23-40, dove l'autrice, definisce alla stregua di "leggende", ossia di false credenze, quelli che dovrebbero essere gli effetti benevoli apportati alla società dalle privative di proprietà intellettuale. In particolare, lo stimolo della creatività, l'innovazione e la formazione del sapere, secondo tale tesi, sarebbero effetti connaturati nella comunità umana e non incentivabili attraverso monopoli, che hanno l'effetto opposto di trasformare il sapere in capitale.

Si è visto come il copyright sia stato adattato al software. Ma il copyright è una tutela pensata per creare un equo bilanciamento in relazione ad altre opere, su tutte quelle letterarie. Per esempio, la lunga durata del diritto deriva dal legame intimo tra creatore e creazione e poco si adatta alla nuova concezione di tempo dell'era digitale.

Si è anche visto come i titolari dei diritti si autotutelino attraverso strumenti contrattuali e tecnici. Questa è una peculiarità del software perché l'autore ha il controllo non solo al momento della cessione del prodotto ma anche dopo la cessione stessa. Nei rapporti *business to consumer* il software non viene venduto, ma concesso in licenza d'uso, e le potenziali violazioni del contratto di licenza vengono controllate dal software stesso mediante le MTP implementate.

Il software è quindi protetto dallo stato, dal contratto di licenza, e infine si protegge da solo. Tali argomentazioni vengono fatte in un determinato contesto economico e sociale, a mio avviso rilevante, ossia quello proprio dell'era digitale.

Determinati rilievi di carattere economico, su tutti la volontà di avvantaggiare le aziende nostrane, hanno sicuramente influenzato i legislatori nella predisposizione di una tutela al software, circostanza che, in Unione Europea, ha portato all'adozione di una privativa autoriale. Lo strumento in questione, tuttavia, risulta inidoneo e incoerente per come è stato applicato al software⁴⁴³.

La sua inadeguatezza da un punto di vista teorico è data dal fatto che questa tutela fu applicata al bene in oggetto in un momento storico in cui si tendeva a negare la brevettabilità del programma per elaboratore. Il diritto d'autore fu quindi "potenziato" per essere idoneo a fornire una protezione soddisfacente al software⁴⁴⁴.

Ma, come si è visto, al copyright si sono aggiunti ulteriori livelli di protezione⁴⁴⁵; nel momento in cui si è iniziato a concedere al software la tutela

⁴⁴³ V. *supra*, cap. II, par. 1.

⁴⁴⁴ La tutela introdotta dalla direttiva 91/250/CE, come si è visto, si estende ad ogni forma espressiva del software, compresi i lavori preparatori, qualora idonei, attraverso un successivo step, ad essere implementati su di un computer. Tuttavia, si deve sottolineare come la tutela non si fermi qui. Anche le GUI, ossia le interfacce grafiche utente, secondo il prevalente orientamento sono tutelabili, ai sensi della normativa europea. V. GUGLIELMETTI, *L'invenzione di software. Brevetto e diritto d'autore*, pp. 299 ss. Questa teoria ha trovato applicazione in sede giurisprudenziale, nella pronuncia della Corte di Giustizia dell'Unione Europea, causa C-393/09, Sentenza della Corte (Terza Sezione) del 22 dicembre 2010, *Bezpečnostní softwarová asociace v. Svaz softwarové ochrany/Ministerstvo kultury*. Nel dispositivo si legge: "*L'interfaccia utente grafica non costituisce una forma di espressione di un programma per elaboratore ai sensi dell'art. 1, n. 2, della direttiva del Consiglio 14 maggio 1991, 91/250/CEE, relativa alla tutela giuridica dei programmi per elaboratore, e non può fruire della tutela mediante diritto d'autore sui programmi per elaboratore in virtù di detta direttiva. Nondimeno, essa può godere della tutela mediante diritto d'autore in quanto opera, ai sensi della direttiva del Parlamento europeo e del Consiglio 22 maggio 2001, 2001/29/CE, sull'armonizzazione di taluni aspetti del diritto d'autore e dei diritti connessi nella società dell'informazione, qualora detta interfaccia costituisca una creazione intellettuale del suo autore.*". In Rete: http://eur-lex.europa.eu/legal-content/it/TXT/PDF/?uri=uriserv%3AOJ.C_.2011.063.01.0008.01.ITA.

⁴⁴⁵ AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 4. L'autrice parla di "stratificazione" alla luce del fatto che "le diverse forme di tutela non sai sono alternate del tempo, ma si sono aggiunte l'un l'altra man mano che l'industria esigeva una protezione più forte".

brevettuale, non si è pensato di porre in essere un sistema di coordinamento tra le tutele⁴⁴⁶. Come risultato si ha una tutela non bilanciata, che pende troppo a favore dell'esclusiva privata.

Le ragioni di una situazione di questo tipo sono più di tipo storico economico che giuridico in senso stretto. Le diverse tipologie di tutele si sono infatti accavallate con il tempo a causa della continua crescita dell'industria di software e del contemporaneo abbandono di quel pregiudizio che inizialmente era manifestato con riguardo alla tutela brevettuale.

L'essere oggetto ad una tutela stratificata *“costituisce una tipica forma di overprotectionism ove l'eccesso di protezione si ravvisa nella coesistenza di paradigmi proprietari differenti ognuno a tutela di porzioni spesso solo apparentemente distinte dal medesimo bene intangibile”*⁴⁴⁷.

Una maggiore utilità per la collettività potrebbe essere ottenuta attraverso un equo bilanciamento in grado di facilitare la circolazione dell'innovazione e allo stesso tempo favorire la creatività.

2. Il vero ostacolo al riconoscimento di una piena tutela brevettuale al software in Unione Europea: la concorrenza internazionale e l'accesso alla tutela

Nel terzo capitolo si è analizzato il fallimento della proposta della Commissione di una Direttiva Europea, che avrebbe introdotto, all'interno della legge degli stati membri, una tutela mediante brevetto del programma per elaboratore⁴⁴⁸.

Chi sostiene l'idoneità del brevetto a tutelare il software, potrebbe pensare che quella fu una occasione mancata. Sicuramente, a livello di coerenza sistematica, l'introduzione di una normativa che avrebbe legittimato l'operato di EPO, non restio alla concessione di brevetti su invenzioni di software, allineando allo stesso tempo l'Unione Europea alla normativa statunitense, sarebbe stata, e forse è, auspicabile.

Come si è cercato di dimostrare in questo lavoro, le problematiche in merito al riconoscimento di una tutela brevettuale al software, non risiedono tanto nella coesistenza della privativa autoriale con quella brevettuale, quanto piuttosto al loro sovrapporsi su di un identico oggetto⁴⁴⁹.

Tuttavia, ad opinione di chi scrive, in quel determinato momento storico, i vivi oppositori di una tutela anche brevettuale del software avevano ragione a contestare quella proposta, ma per i motivi errati.

⁴⁴⁶ Non è inoltre da sottovalutare l'apporto, portato alla tutela nel suo complesso, dal segreto industriale. Come si è già rilevato nel capitolo II, l'accesso alla tutela e autoriale e brevettuale, non comporta alcun onere di pubblicazione dei codici sorgenti. Secondo parte della dottrina, tale elemento non escluderebbe in linea teorica l'applicazione della disciplina del segreto a tali elementi. V. G. GUGLIELMETTI, *Analisi e decompilazione dei programmi*, in *La legge sul software*, Commentario sistematico, Quaderni di AIDA, L. C. UBERTAZZI (a cura di), Giuffrè, Milano, 1992, 152, pp. 162-164.

⁴⁴⁷ AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit., 12.

⁴⁴⁸ V. *supra*, cap. III, par. 4.2.

⁴⁴⁹ V. *supra*, riflessioni conclusive, par. 1.

Alle argomentazioni avverse alla cristallizzazione nel diritto dell'Unione Europea di una tutela brevettuale, di carattere puramente giuridico/tecnico⁴⁵⁰ o di principio⁴⁵¹, sembrano più convincenti rilievi di natura economica.

Come accadeva dieci anni fa, anche allo stato odierno, ottenere un brevetto Europeo con una copertura territoriale rilevante è molto costoso⁴⁵². Ampliare dal punto di vista legislativo le possibilità di brevettare i software significherebbe concedere un enorme vantaggio alle aziende straniere a scapito delle piccole e medie imprese operanti sul mercato europeo. Queste ultime, infatti, non avrebbero i mezzi economici per ottenere una tutela brevettuale.

Gli stati membri finirebbero quindi con il concedere un enorme vantaggio industriale alle grandi multinazionali straniere, a scapito delle proprie realtà economico-industriali.

Tale assetto potrebbe cambiare qualora divenisse operativo il brevetto europeo ad effetto unitario. La tutela diventerebbe più accessibile a causa di un drastico ridimensionamento dei costi, e ciò sicuramente faciliterebbe molti operatori medio-piccoli a rivendicare brevetti di invenzione.

In tal caso, sarà tuttavia auspicabile non incorrere in quell'ordine di problemi, che oggi contraddistinguono il mercato brevettuale americano.

A tal fine ritengo che dovranno essere le corti a porre determinati limiti in merito ai software degni di accedere ad una tutela brevettuale. Ad opinione di chi scrive, il criterio di riferimento non dovrebbe insistere sull'oggetto della rivendicazione, ossia il "*patentable subject matter*", criterio già ampiamente trattato, e idoneo ad essere allargato e ristretto a seconda della vena interpretativa.

Applicando in modo rigoroso i requisiti della novità e della non ovvietà, si potrebbe ottenere il risultato di tutelare mediante brevetto solamente quei software che sono effettivamente meritevoli di accedere a tale forma di tutela, mantenendo nello stesso tempo una interpretazione coerente e costante dell'art. 52 dell'EPC.

3. L'introduzione di regole *ad hoc* nel diritto brevettuale

Alla luce del ragionamento sin ora condotto, si potrebbe ritenere che lo strumento più idoneo, in grado di fornire al software una protezione quanto più equa ed adeguata, vada ricercato nella predisposizione di regole *ad hoc*, finalizzate a limitarne il livello generale di tutela.

Bisogna tuttavia fare chiarezza; anzitutto con regole *ad hoc* non si intende la predisposizione di un diritto *sui generis* così come quello introdotto in Europa

⁴⁵⁰ Se è vero che la EPC prevede la non brevettabilità del software considerato di per sé, è anche vero che è stata la stessa EPO a dare una interpretazione della normativa volta a concedere al software una tutela brevettuale. V. *supra* Cap. III par. 2.4 e cap. IV.

⁴⁵¹ Che, per quanto possano essere condivisibili, sembrano essere in contro-tendenza rispetto al generale andamento del mercato globale. V. *supra*, cap. II, par. 7.

⁴⁵² V. *supra* cap. III, par. 2.1 e 2.2.

per la tutela giuridica delle banche dati⁴⁵³. In quel caso, infatti, il legislatore europeo introdusse un nuovo diritto, stante la considerazione che le banche dati non fossero “*sufficientemente tutelate in tutti gli Stati membri dalle normative esistenti*”⁴⁵⁴. Si è infatti ampiamente dimostrato come il software benefici di una tutela ampia e stratificata, quindi il livellamento della tutela andrebbe effettuato, nel caso, verso il basso.

Diverso sarebbe invece ragionare sulla predisposizione di regole particolari volte a plasmare lo strumento brevettuale per meglio adattarlo al software. Si potrebbe prendere ad esempio la Direttiva sulla protezione delle invenzioni biotecnologiche⁴⁵⁵. Tale direttiva prevede una particolare disciplina per questa specifica categoria di oggetti. Nella fattispecie stabilisce una specifica lista di esclusioni⁴⁵⁶, una precisa rimodulazione del requisito della novità⁴⁵⁷, e fissa l'ambito di protezione concesso dalla tutela⁴⁵⁸.

Tale tutela è stata espressamente pensata per la particolarità e la delicatezza delle innovazioni biotecnologiche⁴⁵⁹. Una sua diretta trasposizione all'oggetto software sarebbe quindi irrealizzabile⁴⁶⁰. Ben si potrebbe, tuttavia, condividerne l'approccio utilizzato.

A oggi, le normative internazionali sui brevetti uniformano l'oggetto della tutela alle innovazioni “*in tutti i campi della tecnologia*”⁴⁶¹. Non si differenzia quindi, a seconda del settore tecnologico di appartenenza.

Tuttavia, è un dato di fatto rilevare come gli uffici brevetti, per valutare la meritevolezza della rivendicazione, si affidino a differenti Divisioni, a seconda del ramo tecnologico in questione.

Se da un punto di vista teorico il brevetto è neutrale, nella prassi la sua concessione è influenzata dal ramo tecnologico cui appartiene l'invenzione. Sulla base di tale assunto si può dire che già oggi esistono all'interno delle commissioni di valutazione dei Patent Offices più diritti brevettuali⁴⁶². In tale contesto, la teoria dell'*Industry-specific Patent Legislation*, sembra quindi persuasiva.

⁴⁵³ Direttiva 96/09/CE del Parlamento europeo e del Consiglio dell'11 marzo 1996 relativa alla tutela giuridica delle banche di dati. In Rete: <http://eur-lex.europa.eu/legal-content/IT/TXT/PDF/?uri=CELEX:31996L0009&from=IT>.

⁴⁵⁴ *Ibidem*, considerando n. 1.

⁴⁵⁵ Direttiva 98/44/CE del parlamento e del consiglio del 6 luglio 1998 sulla protezione giuridica delle invenzioni biotecnologiche. In Rete: <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:1998:213:0013:0021:IT:PDF>.

⁴⁵⁶ *Ibidem*, art. 6.

⁴⁵⁷ *Ibidem*, art. 3.

⁴⁵⁸ *Ibidem*, artt. 8 e 9.

⁴⁵⁹ Per un approfondimento v. AREZZO, GHIDINI, *Biotechnology and software patent law, a comparative review of new developments*, Part III, *Emerging themes in the biotech industries*, pp. 221-339, in particolare v. P. L. C. TORREMAN, *Patentability of human stem cell or synthetic biology based inventions*.

⁴⁶⁰ Per una trasposizione dei concetti della direttiva 98/44/CE nelle invenzioni di software, in particolare con riferimento all'articolo 5, comma 3, v. AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, pp. 261-265.

⁴⁶¹ Così recitano sia l'articolo 27 TRIPs, che l'art. 52 EPC.

⁴⁶² In questo senso v. D. L. BURK, M. A. LEMLEY, *Tailoring patents to different industries*, in AREZZO, GHIDINI, *Biotechnology and software patent law, a comparative review of new developments*, cit., 13: “*Getting a patent is quicker, cheaper, and easier in some industries than in others*”.

Nella dottrina statunitense, molteplici sono le teorie che illustrano la direzione in cui dovrebbe muovere una specifica modifica legislativa, volta ad una sostanziale segmentazione o specializzazione del diritto brevettuale. Lo scopo di tale paragrafo non è tuttavia illustrare in cosa potrebbe o dovrebbe consistere una tale modifica⁴⁶³, quanto piuttosto evidenziare se sia o meno opportuno effettuare una operazione di questo tipo.

Nel contrastare questa visione si possono opporre diverse argomentazioni⁴⁶⁴. Anzitutto, stante la velocità del progresso, una legislazione ad hoc potrebbe essere fallimentare a lungo termine stante la velocità dell'innovazione. Regole che sembrano coerenti oggi, potrebbero risultare incoerenti già nell'arco di pochi anni⁴⁶⁵.

Rappresentare in una legge i dettagli, anche tecnici, di un settore tecnologico, al fine di descriverne la tutela, potrebbe quindi rappresentare un "azzardo".

In primis, si rischia il fenomeno "dell'obsolescenza legislativa", e, secondariamente, stante la probabile influenza di determinati gruppi di interesse sul potere legislativo⁴⁶⁶, esiste il concreto pericolo che la tutela vada nel senso di concedere determinati vantaggi a determinate realtà di nicchia. Il che significherebbe rincorrere il progresso tecnologico attraverso una legge, che nella sua formulazione potrebbe non risultare neutrale.

Se quindi è vero che l'adattamento al software di tutele pensate per proteggere altri oggetti, porta a rilevanti problemi sia teorici che pratici⁴⁶⁷, è altrettanto vero che gli stessi strumenti sono passibili di un certo grado di elasticità. Elasticità che invece verrebbe meno, qualora venissero cristallizzati principi e regole da applicarsi direttamente al software.

Tra le alternative prospettate, ritengo questa ultima visione maggiormente condivisibile. In un quadro legislativo destinato dunque a non mutare, un ruolo rilevante e di primo piano potrebbero svolgerlo le corti, nonché le commissioni esaminatrici degli uffici, che potrebbero beneficiare di quel grado di elasticità, al fine di vagliare con equità e coscienza le differenti rivendicazioni.

Per ottenere un tale risultato, è tuttavia necessario da un lato un elevato livello di specializzazione delle commissioni tecniche di esame, dall'altro un mutato atteggiamento delle corti, volto a considerare la rivendicazione all'interno della categoria tecnologica di riferimento. Ciò non tanto al fine di evidenziare come l'invenzione possa o meno rientrare nell'alveo del brevettabile, quanto piuttosto per evidenziare con rigore il rispetto di quei principi di novità e di non ovvietà che possono essere compresi solo alla luce di una profonda conoscenza dello stato della tecnica.

⁴⁶³ Per una disamina approfondita delle teorizzazioni più rilevanti v. *ibidem*, pp. 15-32.

⁴⁶⁴ V. *ibidem*, pp. 35-39.

⁴⁶⁵ BURK, LEMLEY, *Tailoring patents to different industries*, cit., 36. Gli autori evidenziano come tale circostanza si sia realizzata negli stati uniti con riferimento alla normativa SCPA (*Semiconductor Chip Protection Act*). La legge in questione (17 U.S.C. §§ 901-914) non fu di fatto mai utilizzata in quanto il mutamento della tecnica di produzione di tali oggetti rese la normativa obsoleta.

⁴⁶⁶ È un dato empirico rilevare come il fenomeno delle *lobby*, oltre che essere innegabile, sia anche connaturato soprattutto nella legislazione americana e dell'Unione Europea.

⁴⁶⁷ V. *supra*, cap. II, par. 1.

4. L'opportunità di separare le tutele

Un'ulteriore soluzione al problema della tutela eccessiva del software, consiste nel cercare una modalità attraverso cui separare e delimitare l'ambito, o meglio l'oggetto, di applicazione delle rispettive tutele.

In parte convincente sembra la tesi che trova nella pubblicazione del codice sorgente e dei cosiddetti listati, al momento di registrazione del brevetto, la soluzione alla sovrapposizione delle privative⁴⁶⁸.

Si è già dato atto di come i codici sorgenti siano di fatto assoggettabili ad una peculiare tutela mediante segreto, per effetto dell'applicazione della privativa autoriale⁴⁶⁹. Segreto che quindi non è alternativo al brevetto, ma cumulativo. Il problema di fondo non risiede tanto nella coesistenza di queste diverse forme di tutela quanto piuttosto al loro sovrapporsi su di un identico oggetto⁴⁷⁰.

L'onere di pubblicazione del codice sorgente, nel momento in cui si accede alla tutela brevettuale, avrebbe l'effetto di differenziare il concreto ambito della tutela conferita da ogni privativa.

La tutela brevettuale coprirebbe la funzione, gli algoritmi alla base del programma. Il copyright continuerebbe a tutelare l'espressione dell'idea, ossia quei determinati codici (sorgente ed oggetto) con cui è stato scritto il programma, ma non ci sarebbe spazio per il segreto industriale, essendo il codice sorgente pubblicato. In questo modo, verrebbe soddisfatta a pieno la *disclosure* e ripristinato il necessario equilibrio tra concessione di monopolio e divulgazione della scoperta⁴⁷¹. Essendo i sorgenti disponibili non ci sarebbe più alcun motivo per decompilare un programma.

Di contro, se un soggetto sceglierà di tutelare il programma solo con il copyright, senza l'onere di pubblicare i sorgenti, beneficerà anche del segreto sui codici sorgenti.

La tutela complessiva così ottenuta permette, in una certa misura, il ripristino di un equilibrio tra le tutele. In particolare il rapporto tra tutela brevettuale e segreto tornerebbe alla "normalità", ossia l'applicazione di una disciplina, escluderebbe a priori l'altra. L'innovatore che ritiene opportuno assicurare al suo prodotto una più "forte" protezione sostanziale, sarà obbligato alla pubblicazione di determinate informazioni tecniche -il codice sorgente- che il diritto d'autore gli permetteva di mantenere segrete.

Se la soluzione prospettata sembra essere persuasiva in relazione all'onere di pubblicazione dei sorgenti al momento di rivendicazione del

⁴⁶⁸ AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, pp. 265 ss.

⁴⁶⁹ V. *supra*, cap. II par. 3.

⁴⁷⁰ In altre parole: "La tutela non può semplicemente dirsi cumulativa, nel senso che la prima si aggiunge alla seconda, come spesso accade nel diritto industriale, bensì coincidente". AREZZO, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, cit. 272.

⁴⁷¹ Circostanza che andrebbe a vantaggio in primis della concorrenza, soprattutto nell'ottica di garantire un elevato grado di interoperabilità nel mercato di software.

brevetto, altrettanto non può dirsi nella parte in cui nulla prevede in tema di restringimento della privativa autoriale.

Un tale assetto potrebbe dirsi compensato dagli esorbitanti costi che oggi, almeno in Europa, devono essere sostenuti per l'ottenimento di un brevetto⁴⁷², ragion per cui sembra coerente concedere una tutela forte anche a quei soggetti che una rivendicazione brevettuale non possono oggettivamente permettersela⁴⁷³.

La teoria in questione avrebbe quindi anche il merito di differenziare la tipologia di tutela sulla base degli *applicants* e dell'invenzione. A seconda infatti della tipologia di programma sviluppato, l'inventore, sulla base di una analisi di mercato, potrebbe infatti ritenere più o meno strategico accedere ad una tutela anche brevettuale.

La situazione tuttavia cambierebbe radicalmente nel caso in cui diventasse operativo il brevetto europeo ad effetto unitario⁴⁷⁴. In tal caso, come si è visto, i costi per ottenere un brevetto si abbasserebbero drasticamente, il che renderebbe privo di senso il ragionamento appena effettuato. Verrebbe infatti meno il "contrappeso" dell'alto costo della tutela brevettuale, incentivando invece gli operatori del settore a presentare domanda per l'ottenimento della tutela.

Tale circostanza, tuttavia, potrebbe paradossalmente portare ad una situazione molto vantaggiosa per il mercato. Quanti più saranno i software brevettati, tanto più sarebbero i codici sorgenti disponibili alla conoscenza scientifica. In questa ultima ipotesi, tuttavia, una tutela così penetrante mediante il solo diritto d'autore non sarebbe in alcun modo giustificabile.

Ma una situazione ancor più preoccupante si verificherebbe nel caso in cui si dovesse trasferire lo stato attuale della legislazione al modello brevettuale unificato. In questo ultimo frangente la "supertutela" concessa al software mediante lo stratificarsi di differenti privative, compresa quella brevettuale, sarebbe accessibile quasi da chiunque, eccetto naturalmente che dalle realtà molto esigue, senza alcun contrappeso in termini di *disclosure* dell'opera.

⁴⁷² V. *supra*, cap. II, par. 2.

⁴⁷³ V. *supra*, riflessioni conclusive, par. 2.

⁴⁷⁴ V. *supra*, riflessioni conclusive, par. 2.

BIBLIOGRAFIA

- ANDERMAN S., *Microsoft e il problema dell'interoperabilità*, in *Mercato, Concorrenza, Regole*, 3, 2007, 569.
- AREZZO E., *La brevettabilità del software e dei metodi commerciali elettronici nella giurisprudenza dell'Ufficio Europeo Brevetti*, Assonime, 12/2009.
- AREZZO E., *Nuovi scenari in materia di brevettabilità delle invenzioni attuate a mezzo di elaboratore elettronico: dal tramonto della proposta di direttiva europea alla recente opinione della Commissione Allargata dei Ricorsi dell'UEB nel caso G 0003/08*, in *Riv. dir. ind.*, 2011, 106.
- AREZZO E., *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa*, Giuffrè Editore, Collana Univ. Milano - Fac. giur. Studi Dir. Privato, 2012.
- AREZZO E., GHIDINI G., *Biotechnology and software patent law, a comparative review of new developments*, Collana New Directions in Patent Law, Edward Elgar Publishing, Cheltenham, 2011.
- AUTERI P., FLORIDIA G., MANGINI V., OLIVIERI G., RIVOLFI M., SPADA P., *Diritto industriale. Proprietà intellettuale e concorrenza*, Giappichelli, Torino, 2016.
- BALLARDINI R. M., *Software patents in Europe: the technical requirement dilemma*, in *Journal of Intellectual Property Law & Practice*, 2009, 563.
- BERTANI M., *Open Source*, Quaderni di AIDA n. 13, Giuffrè Editore, Milano, 2005.
- BLIND K., EDLER J., FRIEDEWALD M., *Software patents, economic impacts and policy implications*, Collana New horizons in intellectual property, Edward Elgar Pub, Northampton, 2006.
- BORRUSO R., *Computer e diritto, analisi giuridica del computer*, Giuffrè, Milano, 1998.
- BURK D. L., A. LEMLEY M. A., *The Patent Crisis and How the Courts Can Solve It*, University Of Chicago Press, Ltd., London, 2010.
- CASO R., *Digital rights management: il commercio delle informazioni digitali tra contratto e diritto d'autore*, CEDAM, Padova, 2004, ristampa 2006.
- CASO R., *Technological Protection Measures: Fifty (and More) Shades of Grey of the European Court of Justice*, Trento Law and Technology Research Group, Research Paper n. 19, 2014.

- CASO R., *The Origins of Copyright and Droit d'Auteur: Some Insights in the Law and Technology prospective*, Trento Law and Technology Research Group, Research Paper Series n. 2, 2010.
- CAWTHORNE N., *L'enigma di un genio. La vera storia di Alan Turing il matematico inglese che decrittò il codice nazista*, Newton Compton, Roma, 2014.
- CERUZZI, P. E., *A History of Modern Computing*, MIT Press, Cambridge, 2003.
- CERUZZI P. E., *Storia dell'informatica. Dai primi computer digitali all'era di Internet*, Apogeo Editore, Milano, 2006.
- COHEN J. E., LEMLEY M. A., *Patent Scope and Innovation in the Software Industry*, Georgetown Law Faculty Publications, 89 Cal. L. Rev. 1-57 (2001), 2010.
- COLANGELO G., *Diritto comparato della proprietà intellettuale*, Il Mulino, Bologna, 2011.
- COPELAND B. J., *The Essential Turing. The ideas that gave birth to the computer age*, Oxford University Press, Oxford, 2004.
- COSTA S., *Antitrust e software: il caso Microsoft*, Carocci, Roma, 2004.
- DA BORMIDA M., DOMENICI D., *Software libero, copyleft e digital divide*, in *Diritto di Autore e Nuove Tecnologie*, 2006, 2, 143-170.
- DARROCH C., *Problems and Progress in the Protection of Videogames: A Legal and Socio-logical Perspective*, The University of Manchester, 2012.
- DE SANTIS G., *La tutela brevettuale del software tra brevetto e diritto d'autore*, Giuffrè Editore, Milano, 2000.
- DE SANCTIS V., *A proposito della protezione del format*, in *Dir. Aut.*, 1, 2007, 58.
- DE SIMONE E., *Storia economica: dalla rivoluzione industriale alla rivoluzione informatica*, Angeli, Milano, 2011.
- DI GIANDOMENICO G., *Natura giuridica e profili negoziali del software*, Collana Univ. Molise, Edizioni scientifiche italiane, Roma, 2000.
- DRISCOLL K., *Professional work for nothing: Software commercialization and "An Open Letter to Hobbyists"*, University of Southern California, 2015.
- ELMI G. T., *Informatica e Diritto. Presupposti, storia, disciplina, insegnamento, "ius condendum"*, in *Informatica e diritto*, Vol. XXIII, 2014, n. 2, 85-123.
- FABIANI M., *Software: prodotto dell'ingegno in cerca di protezione*, in *Dir. Inf.*, 1989, 561.
- FELLAS J., *The Patentability of Software-Related Inventions in the United States*, in *E.I.P.R.*, 21(6), 1999, 330.

- FLOR R., *Tutela penale e autotutela tecnologica dei diritti d'autore nell'epoca di Internet. Un'indagine comparata in prospettiva europea ed internazionale*, CEDAM, Padova, 2010.
- FRANCESCHELLI V., *La direttiva CEE sulla tutela del software: trionfo e snaturamento del diritto d'autore*, in *Rivista diritto industriale*, 1991, 169.
- FREIBERGER P., SWAINE M. *Fire in the Valley: The Making of the Personal Computer*, McGraw-Hill, New York, 2000.
- FRUSCALZO R., *Il carattere tecnico dell'invenzione*, Università degli studi di Parma, Dottorato di ricerca in diritto commerciale: proprietà intellettuale e concorrenza, Ciclo XXIV, 2009.
- FUSCO S., *Is In re Bilski a Déjà Vu?*, Stanford Tech. Law Rew, P1, 2009.
- GALLI C., *Le nuove frontiere del diritto dei brevetti*, Giappichelli, Torino, 2003
- GATTEI C., *Il brevetto di software, guida teorico pratica*, LaTribuna, Piacenza, 2003.
- GERACI A., *L'applicazione del principio di esaurimento nei contratti di licenza d'uso del "software"*, in *Il Diritto industriale*, 2015, 5, 447-453.
- GIANNANTONIO E., *Manuale di diritto dell'informatica*, CEDAM, Padova, 1997.
- GHIDINI G., *Profili evolutivi del diritto industriale*, Giuffrè Editore, Milano, 2015.
- GUARDA P., *Creation of Software Within the Academic Context: Knowledge Transfer, Intellectual Property Rights and Licences*, in *IIC - International Review of Intellectual Property and Competition Law*, 2013, vol. 44, n. 5, 494-523.
- GUARDA P., *Looking for a Feasible Form of Software Protection: Copyright or Patent, is That the Question?*, in *EIPR*, 2013, issue 8, 445-454.
- GUGLIELMETTI G., *Analisi e decompilazione dei programmi*, in *La legge sul software*, Commentario sistematico, Quaderni di AIDA, L. C. UBERTAZZI (a cura di), Giuffrè, Milano, 1992, 152.
- GUGLIELMETTI G., *L'invenzione di software. Brevetto e diritto d'autore*, 2^a ed., Giuffrè Editore, Milano, 1997.
- GUGLIELMETTI G., *La proposta di direttiva sulla brevettazione delle innovazioni in materia di software*, in *Rivista di diritto industriale*, 2002, 440.
- HILTZIK M. A., *Dealers of Lightning. Xerox PARC and the dawn of the computer age*, HarperCollins, New York, 1999.
- ILARDI A., *il nuovo brevetto europeo*, Zanichelli, Bologna, 2013.
- ISAACSON W., *Steve Jobs*, Simon and Schuster, New York, 2011.

- ITALIANI M., SERAZZI G., *Elementi di Informatica: algoritmi, architetture, strutture dati, linguaggi di programmazione, Pascal, sistemi operativi, Unix*, Etas Libri, Milano, 1993.
- IZZO U., *Alle origini del copyright e del diritto d'autore, Tecnologia, interessi e cambiamento giuridico*, Carocci, Roma, 2010.
- JAFFE A. B., LERNER J., *Innovation and Its Discontents: How Our Broken Patent System is Endangering Innovation and Progress, and What to Do About It*, Princeton University Press, New Jersey, 2004.
- KRUGMAN P. R., WELLS R., *Microeconomia*, Zanichelli, Bologna, 2013.
- LEITH P., *Patenting programs as machines*, Volume 4, Issue 2, SCRIPT-ed, 2007.
- LEITH P., *Software and patents in Europe*, Cambridge intellectual property Rights and information law, Cambridge University Press, 2007.
- LEMLEY M. A., *Intellectual property and shrinkwrap Licenses*, in 68 S. Cal. L. Rev., 1239, (1995).
- LEMLEY M. A., *Point of Novelty*, Northwestern University Law Review, Vol. 105, No. 3, 2011.
- LEMLEY M. A., *Software patents and the return of functional claiming*, Stanford Public Law Working Paper No. 2117302, 2012.
- LEMLEY M. A., RISCH M., SICHELMAN T., WAGNER R. P., *Life after Bilsky*, 63 Stan. Law Rev. 1315, 2011.
- LEMLEY M. A., MENELL P. S., MERGES R. P., *Intellectual property in the new technological age, 2012 statutory Supplement*, Wolters Kluwer, Aspen Publishers; Supplement edition, New York, 2012.
- LEMLEY M. A., MENELL P. S., MERGES R. P., *Intellectual Property in the New Technological Age: 2016, Vol. I Perspectives, Trade Secrets and Patents*, Clause 8 Publishing, 2016.
- LEMLEY M. A., MENELL P. S., MERGES R. P., *Intellectual Property in the New Technological Age: 2016, Vol II. Copyrights, Trademarks and State IP Protections*, Clause 8 Publishing, 2016.
- LEMLEY M. A., MENELL P. S., MERGES R. P., SAMUELSON P., CARVER B. W., *Software and Internet Law*, Wolters Kluwer, Aspen Publishers; 4th Revised ed., New York, 2011. 1, pp. 25 ss.
- MANCINI A., *International patent law is obsolete*, Buenos Books America, 2006.
- MANES S., ANDREWS P., *Gates: How Microsoft's Mogul Reinvented an Industry and Made Himself the Richest Man in America*, Touchstone, Simon and Schuster, New York, 1994.

- MANOVICH L., *Software culture*, Edizioni Olivares, Milano, 2010,
- MANOVICH L., *Software takes command*, Bloomsbury Academic, INT edition, New York, 2013.
- MANSANI L., *La Brevettabilità dei Metodi Commerciali e delle invenzioni attuate per mezzo di elaboratori elettronici*, in Studi di diritto industriale in onore di Adriano Venzetti. Proprietà Intellettuale e concorrenza, II, Giuffrè, Milano, 2004.
- MENELL P. S., *An Analysis of the Scope of Copyright Protection for Application Programs*, 41 *Stanford Law Review*, 1045, (1988).
- MENELL P. S., *Forty Years of Wondering in the Wilderness and No Closer to the Promised Land: Bilski's Superficial Textualism and the Missed Opportunity to Return Patent Law to Its Technology Mooring*, 63 *Stan. L. Rev.* 1289, (2010).
- MOORE F., *It's a Hobby*, in *Homebrew Computer Club Newsletter*, vol. 1, n° 4, 7 giugno 1975.
- NARD C. A., *The Law of Patents*, Wolters Kluwer, Aspen Publishers, New York, 2008.
- OHLY A., *Common principles of european intellectual property law*, Mohr Siebeck, Tübingen, 2012.
- OHLY A., PILA J., *The europeanization of intellectual property law, towards a european legal methodology*, Oxford University Press, Oxford, 2013.
- O'ROURKE M. A., *The story of Diamond v. Diehr: Toward Patenting Software*, in *Intellectual property stories*, DREYFUSS R., GINSBURG R. C. (a cura di), Foundation Press, New York, 2006.
- OSENGA K., *Debugging Software's Schemas*, in *Intellectual property Law Review*, 2015, Edited by Karen B. Tripp, Thomson Reuters, 2015, 203-228.
- PASCUZZI G. (a cura di), *Il diritto nell'era digitale*, il Mulino, Bologna, 2016.
- PILA J., *Software Patents, Separation of Powers, and Failed Syllogisms: A Cornucopia from the Enlarged Board of Appeal of the European Patent Office*, *The Cambridge Law Journal*, Vol. 70, No. 1 (March 2011), 203-228.
- PROSPERETTI E., *La circolazione dell'opera digitale*, Giappichelli, Torino, 2012.
- PROSPERETTI E., *L'opera digitale tra regole e mercato*, Giappichelli, Torino, 2013.
- C. REED, *Reverse Engineering Computer Programs without Infringing Copyright*, in 13 *E.I.P.R.* 47, 1991.

- RAYMOND E. S., *The Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*, O'Reilly Media Inc., Revised Edition, Sebastopol, 2001.
- RICOLFI M., *La tutela della proprietà intellettuale: fra incentivo all'innovazione e scambio ineguale*, in *Riv. Dir. Industriale*, I, n. 6, 2002, 511.
- RICOLFI M., *Software e limitazioni del licenziatario*, in *AIDA* 2004, 358.
- ROSSI M. A., *Software patents: a closer look at the European commission's proposal*, Siena Memos and Papers in Law and Economics n. 30, Univ. Siena, 2005.
- SAMUELSON P., *Benson Revisited: The Case Against Patent Protection For Algorithms and Other Computer Program-Related Inventions*, Berkeley Law Scholarship Repository, 39 *Emory L.J.* 1025 (1990).
- SAMUELSON P., *Freedom of Ideas and of Competition, the story of Backer v Selden: Sharpening the distinction between authorship and innovation*, in *Intellectual property stories*, DREYFUSS R., GINSBURG J. C. (a cura di), Foundation Press, New York, 2006.
- SANNA P. S., *Manuale di informatica di base*, CEDAM, Padova, 2014.
- SELTZER W., *The Imperfect Is the Enemy of the Good: Anticircumvention versus Open User Innovation*, 25 *Berkeley Tech. L.J.* 909, 2010.
- SHIVA V., *Il mondo sotto brevetto*, Feltrinelli, Milano, 2002.
- SUTTON E., *Software Patents: A Practical Perspective*, CreateSpace Independent Publishing Platform, 2016.
- VAN EECHOUD M., HUGENHOLTZ P. B., VAN GOMPEL S., GUIBAULT L., HELBERGER N., *Harmonizing european copyright law*, Wolters Kluwer, Kluwer Law International, The Netherlands, 2009.
- VANZETTI A., DI CATALDO V., *Manuale di diritto industriale*, Giuffrè Editore, Milano, 2012.
- WALLACE J., ERICKSON J., *Hard Drive: Bill Gates and the Making of the Microsoft Empire*, John Wiley & Sons, New York, 1992.
- WALTER M. M., VON LEWINSKI S., *European copyright law, a commentary*, Oxford University Press, Oxford, 2010.
- WATKINS JR. W. J., SHUGHART II W. F., *Patent Trolls: Predatory Litigation and the Smothering of Innovation*, The Independent Institute, Oakland, 2014.
- ZICCARDI G., *L'eredità culturale del secondo dopoguerra e le nuove tecnologie: gli anni Cinquanta e Sessanta, il computer e le prime problematiche sociali e giuridiche*, in *Cyberspazio e diritto*, Vol. 9, n. 2, 163-206, 2008.

FONTI LEGISLATIVE E DOCUMENTI ISTITUZIONALI

Unione Europea

Proposal for a Council Directive on the legal protection of computer programs, *Explanatory Memorandum*, 1989.

Direttiva 91/250/CEE del Consiglio, del 14 maggio 1991, relativa alla tutela giuridica dei programmi per elaboratore.

Direttiva 96/09/CE del Parlamento europeo e del Consiglio dell'11 marzo 1996 relativa alla tutela giuridica delle banche di dati.

Direttiva 98/44/CE del parlamento e del consiglio del 6 luglio 1998 sulla protezione giuridica delle invenzioni biotecnologiche.

Direttiva 2001/29/CE del parlamento europeo e del consiglio, del 22 maggio 2001, sull'armonizzazione di taluni aspetti del diritto d'autore e dei diritti connessi nella società dell'informazione.

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

Proposta di direttiva del Parlamento europeo e del Consiglio relativa alla brevettabilità delle invenzioni attuate per mezzo di elaboratori elettronici, (2002/C 151 E/05), COM(2002) 92 def., 2002/0047(COD), Presentata dalla Commissione il 20 febbraio 2002.

Regolamento (UE) n. 1257/2012 del Parlamento Europeo e del Consiglio del 17 dicembre 2012 relativo all'attuazione di una cooperazione rafforzata nel settore dell'istituzione di una tutela brevettuale unitaria.

Regolamento (UE) n. 1260/2012 del Consiglio del 17 dicembre 2012 relativo all'attuazione di una cooperazione rafforzata nel settore dell'istituzione di una tutela brevettuale unitaria con riferimento al regime di traduzione applicabile.

Accordo del Consiglio (2013/C 175/01) su un tribunale unificato dei brevetti.

Comunicazione della commissione al parlamento europeo e al consiglio, *Migliorare il sistema dei brevetti in Europa*, Bruxelles, 03/04/2007 COM(2007).

Decisione della Commissione, del 24 maggio 2004 , relativa ad un procedimento a norma dell'articolo 82 del trattato CE e dell'articolo 54 dell'accordo SEE contro Microsoft Corporation (Causa n. COMP/ C-3/37.792 — Microsoft).

Libro verde sul brevetto comunitario e sul sistema dei brevetti in Europa, Promuovere l'innovazione tramite il brevetto, 24 giugno 1997.

Organizzazione Europea dei Brevetti (EPOrg)

European Patent Office, *Guidelines*, 1978.

European Patent office, *Annual Report*, 1985.

European Patent Office, *Patenting Computer Software*, in Annual Report, 1994.

EPO, “True top 4”, Adozione di una struttura di tasse favorevole alle imprese per il brevetto unitario, 24 giugno 2015.

Guidelines for Examination in the EPO, November 2016.

Stati Uniti

Costituzione degli Stati Uniti d’America, 15 settembre 1787.

CONTU, *Final Report of the National Commission on New Technology Uses of Copyrighted Works*, 31/07/1978.

U.S. Code: Title 17 - Copyrights.

U.S. Patent Act -- 35 USCS Sects. 1 - 376.

United States Patent and Trademark Office, *Public Hearing on Use of the Patent System to Protect Software-Related Inventions*, Transcript of Proceedings Wednesday, January 26, 1994 Thursday, January 27, 1994, Thursday, February 10, 1994, Friday, February 11, 1994.

United States Patent and Trademark Office, United States Department of Commerce, *Legal Analysis to Support Proposed Examination Guidelines for Computer-Implemented Inventions*, 3/10/1995.

Internazionali

Convenzione di Berna, 1886.

Convenzione di Strasburgo del 1963.

European Patent Convention, 1973.

Documento WIPO, *Model provision on the protection of computer software*, Ginevra, 1978.

The Agreement on Trade Related Aspects of Intellectual Property Rights (TRIPs), 1995.

WIPO Copyright Treaty (WCT), 1996.

Altro

Legge 22 aprile 1941 n. 633, Protezione del diritto d'autore e di altri diritti connessi al suo esercizio, (G.U. n.166 del 16 luglio 1941).

Loi n°68-1 du 2 janvier 1968 sur les brevets d'invention.

UK, The Patents Act 1949.

UK Patent Office, Linee guida, Febbraio 1969.

FONTI GIURISPRUDENZIALI

Decisioni di EPO

Vicom/Computer-related invention, T 0208/8, 15.07.1986.

Koch & Sterzel/X-Ray Apparatus, T 26/86, 21.05.1987.

Ibm/Text processing, T 0038/86, 14.02.1989.

Ibm/Editable document form, T 0110/90, 15.04.1993.

Ibm/Computer Program Product I, T 1173/97, 01.07.1998.

Ibm/Computer Program Product II, T 0935/97, 04.02.1999.

Auction method/HITACHI, T 258/03, 21.04.2004.

Video game/KONAMI, T 0928/03, 02.06.2006.

Citibank/Integrated account, T 368/05, 15.05.2007.

Game machine/GAMEACCOUNT, T 1543/06, 29.06.2007.

Gift order/AMAZON, T 1616/08, 11.11.2009.

Programs for computers, G 0003/08, 02.05.2010.

Corte Suprema

Backer v Selden, 101 U.S 99 (1879).

Gottschalk v. Benson, 409 u.s. 63 (1972).

Parker v. Flook, 437 U.S. 584 (1978).

Diamond v. Diehr, 450 U.S. 175 (1981).

Bilski v. Kappos, 561 U.S. 593 (2010).

Corti inferiori

United States v. Microsoft Corporation, 253 F.3d 34 (D.C. Cir. 2001).

In re Alappat, 33 F.3d 1526, U.S. Court of Appeals Federal Circuit.

State Street Bank & Trust Co. v. Signature Financial Group, 149 F.3d 1368 (Fed. Cir. Jul. 23, 1998).

In re Bilski, 545 F.3d 943, 88 U.S.P.Q.2d 1385 (Fed. Cir. 2008).

Unione Europea

Sentenza del Tribunale di primo grado (grande sezione) del 17 settembre 2007, Microsoft Corp. contro Commissione delle Comunità europee, COMP/C-3/37.792.

Causa C-406/10, SAS Institute Inc, contro World Programming Ltd, 2 maggio 2012.

Corte di Giustizia dell'Unione Europea, causa C-393/09, Sentenza della Corte (Terza Sezione) del 22 dicembre 2010, Bezpečnostní softwarová asociace contro Svaz softwarové ochrany/Ministerstvo kultury.

Altro

Cour D'Appel de Paris, 22 mai 1973, in Ann. 1973, 275.

Cour de Cassation, Chambre commerciale, du 28 mai 1975, 73-12.820.

Caso Schlumberger del 15 giugno 1981 (Cour D'Appel de Paris, in Ann. 1982, 24).

ALTRE FONTI RILEVANTI

Enciclopedie: Oxford English Dictionary (oed.com), Treccani.it.

Dati statistici: netmarketshare.com e bls.gov.

Portali di organizzazioni per lo sviluppo dell'open software: gnu.org, opensource.org. e fsfe.org.

Consultazione dei brevetti registrati: google.com/patents e epo.org.

Ringraziamenti

Desidero ora ringraziare tutti coloro che mi hanno aiutato nell'ideazione e nella stesura della tesi con suggerimenti, critiche ed osservazioni: a loro va la mia gratitudine, anche se a me spetta la responsabilità per ogni errore contenuto in questo lavoro.

Ringrazio anzitutto il professor Roberto Caso, Relatore, per avermi trasmesso, attraverso i suoi corsi, non solo i fondamenti della materia, ma anche la passione per la stessa, nonché lo stimolo di vagliare in modo critico gli argomenti oggetto di studio. Grazie anche per la costante disponibilità e comprensione; senza questi elementi questa tesi non avrebbe sicuramente visto la luce dell'estate.

Rivolgo un particolare e sentito ringraziamento al professor Paolo Guarda, per i preziosi consigli, per la pazienza profusa nell'aiutarmi, e per il tempo che mi ha concesso.

Vorrei infine ringraziare le persone a me più care.

I miei amici; ed in particolare Martino, da sempre interessato al mio percorso di studi. La mia famiglia; i miei nonni che sebbene lontani, da sempre mi sostengono e mi motivano, mia sorella Sara, e naturalmente i miei genitori, a cui questo lavoro è dedicato, che durante il mio cammino universitario non mi hanno mai fatto mancare nulla.

Infine ringrazio Maria Vittoria, per il costante incoraggiamento e supporto quotidiano, e Y., per le lunghe e piacevoli passeggiate.

A handwritten signature in blue ink, consisting of stylized, flowing letters that appear to be 'S.A.' or similar initials.