



**FACULTADE DE DEREITO**  
**Departamento de dereito mercantil**  
**e do traballo**

**Tesis Doctoral**  
**Mención Europea**

**EL DEBATE SOBRE LA INTEROPERABILIDAD**  
**INFORMÁTICA EN EL DERECHO DE AUTOR**  
**COMUNITARIO.**

**THE SOFTWARE INTEROPERABILITY**  
**DEBATE ON EUROPEAN COPYRIGHT LAW.**

AUTORA:

LIC. BEGOÑA GONZÁLEZ OTERO

DIRECTORES:

PROF. DR. ÁNGEL FERNÁNDEZ-ALBOR BALTAR

DR. FRANCISCO JAVIER FRAMIÑÁN SANTAS

Santiago de Compostela, 2013.





**UNIVERSIDAD DE SANTIAGO DE COMPOSTELA  
FACULTAD DE DERECHO  
DEPARTAMENTO DE DERECHO MERCANTIL  
Y DEL TRABAJO**

**EL DEBATE SOBRE LA INTEROPERABILIDAD  
INFORMÁTICA EN EL DERECHO DE AUTOR COMUNITARIO.  
THE SOFTWARE INTEROPERABILITY DEBATE ON  
EUROPEAN COPYRIGHT LAW.**

Trabajo de investigación inédito realizado por la Licenciada Begoña González Otero, bajo la dirección del Catedrático de Derecho mercantil Prof. Dr. Ángel Fernández-Albor Baltar y el Profesor Contratado Dr. Francisco Javier Framiñán Santas, para la obtención del grado de Doctora en Derecho con Mención Europea.

Fdo.: Begoña González Otero

Vº Bº de los Directores de la Tesis Doctoral.



# ÍNDICE

	<b>Pág.</b>
<b>Abreviaturas y acrónimos.</b>	3

## **Introducción.**

1. El programa de ordenador como factor clave de la Sociedad de la Información.	11
2. El debate jurídico en torno a la interoperabilidad en el marco del derecho de autor comunitario.	13

## **Capítulo 1**

### **Los elementos del debate en el marco del Derecho de Autor.**

1. La interoperabilidad.	21
1.1. La interoperabilidad: un concepto multidimensional.	21
1.2. La interoperabilidad vertical y horizontal.	31
1.3. El papel de la interoperabilidad en el mercado del software.	33
A) La atipicidad del mercado: software, network effects e interoperabilidad.	35
B) Efectos económicos de la interoperabilidad.	38
B.1) Clases de network effects.	38
B.2) Efectos positivos: mejora de la competitividad y aumento del bienestar del consumidor.	39
B.3) Efectos negativos: estandarización susceptible de frenar la innovación.	43
2. Los programas de ordenador.	46
2.1. Los programas de ordenador desde la perspectiva técnica.	46
A) El concepto de programa de ordenador.	46

B) Características y clasificaciones.	47
C) La creación de un programa de ordenador: De la programación a la ingeniería informática.	52
2.2. Los programas de ordenador desde la perspectiva jurídica.	58
A) Marco normativo internacional.	58
B) El programa de ordenador como obra de derecho de autor asimilada a una obra literaria.	60
C) Las características especiales de los programas de ordenador.	64
a) La funcionalidad.	64
b) La ausencia de finalidad comunicativa.	67
D) La ficción jurídica de tratar al programa de ordenador como obra literaria en el marco del Convenio de Berna.	70
3. La ingeniería inversa del software en general y la descompilación en particular.	73
3.1. Ingeniería inversa e interoperabilidad.	79
3.2. El carácter infractor de la descompilación para el derecho de autor.	80
A) El derecho de reproducción.	82
B) El derecho de transformación.	85
3.3. La ingeniería inversa entre el impulso y el freno a la innovación.	90
4. La información interoperable contenida en las interfaces del programa.	94
4.1. El concepto de interfaz desde la perspectiva técnica.	94
4.2. El concepto de interfaz desde la perspectiva jurídica: ¿idea, expresión o método de operación?	97
4.3. Protección sobre las interfaces y derecho a acceder a la información interoperable: ¿un posible nuevo conflicto para el derecho comunitario?	101
A) Planteamiento de la cuestión: el conflicto entre dos derechos fundamentales.	101

B) La propiedad sobre la información interoperable frente al derecho del público a acceder a la información.	105
<b>5. Conclusión de capítulo.</b>	<b>107</b>

## **Capítulo 2**

### **El debate sobre la interoperabilidad informática en la Copyright Act de Estados Unidos.**

1. La protección de los programas de ordenador a través de la Copyright Act.	111
2. La garantía de la interoperabilidad en la Copyright Act.	114
2.1. El ámbito de protección del programa.	114
2.1.1. La extensión de la protección a las interfaces del programa.	117
A) Apple contra Franklin.	117
B) Whelan contra Jaslow.	119
2.1.2. La restricción del ámbito de protección sobre algunos elementos del programa en favor de la libre competencia.	121
A) Pains Cotton contra Goodpasture.	121
2.1.3. La exclusión de las interfaces del ámbito de protección sobre el programa a través de un análisis analítico.	123
A) Computer Associates contra Altai.	123
B) Apple Computer contra Microsoft y Hewlett-Packard.	125
C) Lotus contra Borland (Lotus II).	126
2.1.4. El aparente retorno hacia la doctrina extensiva.	127
A) Bateman contra Mnemonics.	127
B) Dun & Bradstreet contra Grace Consulting.	128
C) Oracle contra Google.	128

2.2. La limitaciones de la Copyright Act.	131
2.2.1. La excepción de uso limitado del artículo 117.	132
A) Fundamento de la excepción.	132
B) Requisitos subjetivos: la persona autorizada.	134
C) Requisitos objetivos: la conducta permitida.	138
a) El acto constituye un paso esencial.	139
b) La expresión “que no se use de ningún otro modo”.	142
2.2.2. La excepción del fair use del artículo 107.	143
3. La Digital Millennium Copyright Act (DMCA) como garante de la interoperabilidad informática.	153
3.1. Introducción: La DMCA y su relación con la interoperabilidad informática.	153
3.2. La Digital Millennium Copyright Act.	154
3.3. Las conductas prohibidas por el artículo 1201 DMCA.	159
3.4. La excepción de la ingeniería inversa del artículo 1201(f)	161
3.4.1. Examen de la excepción.	162
a) Requisitos subjetivos: la persona autorizada.	162
b) Requisitos objetivos de la conducta permitida.	164
1. El control efectivo al acceso de la obra protegida.	165
2. Los elementos necesarios para alcanzar la interoperabilidad.	167
3. Los mecanismos tecnológicos de inutilización.	168
4. El uso de la información adquirida: la posibilidad de proporcionar información o medidas para alcanzar la interoperabilidad a terceros.	170
3.4.2. Relación entre el artículo 1201(f) y la doctrina del fair use.	172



<b>4. Conclusión de capítulo.</b>	178
-----------------------------------	-----

### Capítulo 3

#### **El debate sobre la interoperabilidad informática en el derecho de autor europeo.**

<b>1. La protección de los programas de ordenador en el derecho europeo antes de la Directiva 91/250/CEE de 14 de mayo de 1991.</b>	183
1.1. El camino hacia la Directiva sobre protección jurídica de los programas de ordenador.	187
1.2. Principales diferencias entre los regímenes nacionales de derecho de autor con efectos sobre la interoperabilidad.	188
1.2.A) La originalidad del programa.	188
A.1) Alemania.	188
A.2) Francia.	193
A.3) Reino Unido.	196
A.4) España.	198
1.2.B) Las facultades exclusivas de reproducción y transformación.	199
B.1) El concepto de reproducción del programa.	199
B.2) El concepto de transformación del programa (adaptation right).	204
<b>2. La protección de los programas de ordenador en la Directiva 91/250/CEE de 14 de mayo sobre protección jurídica de los programas de ordenador.</b>	206
2.1. La finalidad protectora de la Directiva 91/250/CEE.	206
2.2. El debate sobre la interoperabilidad en la elaboración de la Directiva.	207
2.3. La bandera blanca: La excepción de la descompilación del artículo 6.	213
2.4. Análisis del artículo 6 de la Directiva.	215
A) Principios que rigen la excepción de la descompilación.	216

B) Actos permitidos por la excepción.	218
C) El objeto: La información interoperable de las interfaces.	223
C.1) Las interfaces del programa ante la dicotomía idea-expresión.	223
C.2) Las interfaces del programa ante el requisito de originalidad.	233
D) Las condiciones autorizantes de la descompilación: Artículo 6.1 de la Directiva.	239
D.1) La interoperabilidad de un programa creado de forma independiente con otros programas.	239
D.2) La persona autorizada.	242
D.3) La información necesaria para conseguir la interoperabilidad.	244
D.4) La indispensabilidad de la descompilación.	244
E) El uso de la información obtenida con la descompilación.	247
E.1) La excesiva restricción sobre los usos de la información descompilada.	248
E.2) El procedimiento de “sala blanca”: ¿Un salvoconducto?	250
E.3) La invitación implícita a negociar entre las partes.	252
<b>3. La interpretación del artículo 6 de la Directiva por los tribunales de los Estados miembro.</b>	254
3.1. Alemania.	254
3.2. Francia.	257
3.3. Reino Unido.	262
3.4. España.	264
<b>4. Conclusión de capítulo.</b>	266

## Capítulo 4

### **La relación de la Directiva sobre protección jurídica de los programas de ordenador con otras ramas del ordenamiento jurídico en materia de interoperabilidad: Más razones para la reforma del artículo 6.**

<b>1. La relación entre la Directiva y el Derecho de defensa de la competencia en materia de interoperabilidad.</b>	271
1.1. Políticas fundamentales de protección de los derechos de Propiedad Industrial e Intelectual.	271
1.2. Políticas fundamentales de protección del Derecho de defensa de la competencia.	273
1.3. La relación entre el Derecho de defensa de la competencia y el Derecho de autor en la Directiva de software.	275
1.4. El creciente papel tutelar del Derecho de defensa de la competencia.	279
<b>2. El papel de la doctrina de las “essential facilities” en el debate sobre la interoperabilidad informática.</b>	281
2.1. La doctrina de las “essential facilities” en la jurisprudencia comunitaria como correctora excepcional del ejercicio abusivo de los derechos de Propiedad Industrial e Intelectual.	281
A) Volvo.	283
B) Magill.	284
C) Oscar Bronner.	285
D) IMS Health.	287
2.2. La aplicación de la doctrina de las “essential facilities” en situaciones donde no se proporciona la información interoperable a los competidores.	289
A) Microsoft.	289
B) Apple.	294
C) The Mathworks.	297
2.2.1. La indispensabilidad de la información interoperable en Microsoft y su relación con la descompilación.	298

2.2.2 Abuso del derecho de autor y abuso de posición dominante en relación a la interoperabilidad.	300
2.2.3. ¿Son las licencias obligatorias la solución más equilibrada para el debate de la interoperabilidad?	302
<b>3. La relación entre la Directiva y el derecho contractual en materia de interoperabilidad.</b>	307
3.1. El principio de la libertad contractual ante el Derecho de autor en el caso del software.	307
3.2. La imperatividad del artículo 8 de la Directiva.	309
<b>4. Propuestas de <i>lege ferenda</i> en relación con el artículo 6 y sus consecuencias.</b>	313
4.1. Propuestas de <i>lege ferenda</i> .	313
4.1.1. La propuesta de una excepción general para la ingeniería inversa.	313
4.1.2. La propuesta de un nuevo artículo 6 en dos niveles.	316
4.2. Una enseñanza de derecho comparado: La reforma de la excepción sobre ingeniería inversa en la Ley sobre Propiedad Intelectual de Chile.	321
4.3. El fomento de los estándares abiertos.	323
<b>5. Conclusión de capítulo.</b>	325
<b>Conclusiones finales.</b>	329
<b>Anexo Mención Europea (Introduction and Conclusions).</b>	339
<b>Índice de jurisprudencia.</b>	355
<b>Bibliografía.</b>	363

## *Abreviaturas y acrónimos*

§	Sección
A.	Repertorio Aranzadi de Jurisprudencia
AAVV	Autores varios
ABLJ	American Business Law Journal (Estados Unidos)
AC	Aranzadi Civil
ACM	Association of Computing Machinery
Act. Civ.	Actualidad Civil (España)
ADI	Actas de Derecho Industrial y Derecho de Autor (España)
ADPIC	Acuerdo sobre los Aspectos de los Derechos de Propiedad Intelectual relacionados con el Comercio (TRIPs, en inglés).
AIA	Actualidad Informática Aranzadi
AIPL	Annual of Industrial Property Law (Gran Bretaña)
AIPPI	Asociación Internacional para la Protección de la Propiedad Industrial
AJ	Aranzadi. Repertorio de Jurisprudencia (España)
ALAI	Asociación Literaria y Artística Internacional
Alb. L.J.Sci. & Tech	Albany Law Journal of Science and Technology (Estados Unidos)
AMI	Auteurs-, Media-, Informatierrecht (Holanda)
AN	Audiencia Nacional
Antitrust L.J	Antitrust Law Journal (Estados Unidos)
AOJP	Australian Office Journal of Patents
AP	Audiencia Provincial
Art.	Artículo
Arts.	Artículos
BB	Der Betriebs-Berater (Alemania)
BDA	Boletín de Derecho de Autor de la UNESCO
BeckRS	Beck Rechtsprechung (Alemania)
B2B	Business to Business (comercio electrónico entre profesionales)
B2C	Business to Consumer (comercio electrónico entre profesionales y consumidores)
BGBI	Bundesgesetzblatt (Alemania)
BGH	Bundesgerichtshof (Tribunal Supremo Federal Alemán)
BOCG	Boletín Oficial de las Cortes Generales (España)
BOE	Boletín Oficial del Estado (España)
BOPI	Boletín Oficial de la Propiedad Industrial (España)
BPatGer.	Bundespatentgericht (Alemania)
Boston College Intellec. Prop. & Tech. F.	Boston College Intellectual Property and Technology Forum (Estados Unidos)
BSA	Business Software Alliance
BTLJ	Berkeley Technology Law Journal (Estados Unidos)
BYU L. Rev.	Birgham Young University Law Review (Estados Unidos)
C.Co.	Código de Comercio (España)
CA	Cour d'Appel (Corte de Apelación francesa)
Ca.L.Rev.	California Law Review (Estados Unidos)
CAFC	Court of Appeals for the Federal Circuit (Estados Unidos)

Cass	Unidos)
CB	Cour de Cassation (Francia)
	Convenio de Berna para la protección de obras literarias y artísticas, revisado en París el 24 de julio de 1971
Cc.	Código Civil
CCE	Communications – Commerce électronique (Francia)
CDPA	Copyright, Designs and Patents Act 1988
CE	Constitución española o Comunidad Europea (según contexto)
CEE	Comunidad Económica Europea
CENDOJ	Centro de Documentación Judicial (España)
Cir.	Circuit
Cit.	Citado
CJPI	Cuadernos de Jurisprudencia sobre la Propiedad Industrial
CLJ	Computer/Law Journal
CLP	Computer Law & Practice (Reino Unido)
CLSR	Computer Law and Security Report
Colum.L.R.	Columbia Law Review (Estados Unidos)
Colum.-VLA J.L. & Arts	Columbia Journal of Law and the Arts (Estados Unidos)
CONTU	Commission on New Technological Uses of Copyrighted Works
CPC	Convenio de Luxemburgo sobre la Patente Comunitaria
CPE	Convenio de Munich sobre la Patente Europea
CPI	Code de la Propriété Intellectuelle
CPIDC	Comunicaciones en Propiedad Industrial y Derecho de la competencia (España)
CR	Computer und Recht (Alemania)
CRi	Computer Law Review international (Alemania)
CTLR	Computer & Telecommunications Law Review (Reino Unido)
CUP	Convenio de la Unión de París
CV	Convención de las Naciones Unidas sobre los contratos de compraventa de mercaderías, hecha en Viena el 11 de abril de 1980
D	Recueil Dalloz (Francia)
DdA	Le Droit d'auteur (OMPI/BIRPI)
Dec. Leg.	Decreto legislativo
Dec.Ley	Decreto Ley
DN	Derecho de los Negocios (España)
DOCE	Diario Oficial de las Comunidades Europeas
DOUE	Diario Oficial de la Unión Europea
Directiva Software	Directiva 91/250 CEE, del Consejo, de 14 de mayo de 1991, sobre la Protección Jurídica de los Programas de Ordenador (DOCE L/122 de 17 de mayo), codificada en la Directiva 2009/24/CE del Parlamento Europeo y del Consejo de 23 de abril de 2009 sobre la protección jurídica de programas de ordenador (DO L111 de 5 de mayo)
Directiva 2001/29	Directiva 2001/29 CE, del Parlamento Europeo y del Consejo, de 22 de mayo de 2001 relativa a la armonización de determinados aspectos de los derechos de autor y derechos afines a los derechos de autor en la Sociedad de la Información (DOCE

DMCA	L/167 de 22 de junio) Digital Millenium Copyright Act (Estados Unidos, 1998, S.2037)
EIPR	European Intellectual Property Review (Gran Bretaña)
EPI	Estatuto de la Propiedad Industrial
EPO	European Patent Office
EuR.	Europarecht (Alemania)
Eur.L.Rev.	European Law Review
F	Federal Reporter
FSF	Free Software Foundation
F.Supp.	Federal Supplement
F.Supp.2d	Federal Supplement, Second Series
F.2d	Federal Reporter, Second Series
F.3d	Federal Reporter, Third Series
FJ	Fundamento Jurídico
FSR	Fleet Street Law Reports (Gran Bretaña)
FTC	Federal Trade Commission (Estados Unidos)
Gaz. Pal.	Gazette du Palais (Francia)
Geo.Wash.J.Int.L.&Econ.	George Washington Journal of International Law and Economics (Estados Unidos)
Geo.Wash.L.Rev.	George Washington Law Review (Estados Unidos)
GRUR	Gewerblicher Rechtsschutz und Urheberrecht (Alemania)
GRUR Int.	Gewerblicher Rechtsschutz und Urheberrecht International (Alemania)
Harv. L. Rev.	Harvard Law Review (Estados Unidos)
High Tech. L. J	High Technology Law Journal (Estados Unidos)
Ibid.	Ibidem
IDEI	Instituto de Derecho y Ética Industrial
IDIUS	Instituto de Derecho Industrial de la Universidad de Santiago de Compostela
IIC	International Review of Industrial Property and Copyright Law
ICTSD	International Centre for Trade and Sustainable Development (Suiza)
IDP	Internet, Derecho y Política (España)
IJLIT	International Journal of Law and Information Technology (Gran Bretaña)
IEEE	Institute of Electrical and Electronics Engineers
InDret	InDret. Revista para el análisis del Derecho (España)
Ind. L.J	Indiana Law Journal (Estados Unidos)
Informe Directiva 2000	Informe de la Comisión al Consejo, al Parlamento Europeo y al Comité Económico y Social sobre la transposición y los efectos de la Directiva 91/250/CEE sobre la protección jurídica de programas de ordenador (COM/2000/0199 final)
Int.Prop.L.Rev.	Intellectual Property Law Review (Estados Unidos)
JET Law	Vanderbilt Journal of Entertainment and Technology Law
J.L. & Inf. Sci.	Journal of Law and Information Science (Estados Unidos)
J. Marshall J Computer & Info. L.	The John Marshall Journal of Computer & Information Law (Estados Unidos)
JIPITEC	Journal of Intellectual Property, Information

JOLT	Technology and E-Commerce Law (Alemania) Harvard Journal of Law & Technology (Estados Unidos)
JORF	Journal Officiel de la Republique Française (Francia)
JPI	Juzgado de Primera Instancia
Jurimetrics J.	Jurimetrics Journal (Estados Unidos)
JUR	Base de datos Aranzadi Westlaw
JZ	Juriszeitung (Alemania)
La Ley	La Ley, Revista española de jurisprudencia y legislación (España)
LCAI	Law, Computers and Artificial Intelligence (Estados Unidos)
LDC	Ley 15/2007 de 3 de julio de defensa de la competencia (BOE núm. 159 de 4 de julio)
LIPRO	Ley 16/1993 de 23 de diciembre, de incorporación al Derecho español de la Directiva 91/250/CEE, de 14 de mayo de 1991, sobre protección jurídica de los programas de ordenador.
LPI 1987	Ley de Propiedad Intelectual de 10 de noviembre de 1987 (BOE núm. 275 de 17 de noviembre)
MTP	Medida tecnológica de protección
NJW	Neue Juristische Wochenschrift (Alemania)
Notre Dame L. Rev	Notre Dame Law Review (Francia)
NW.J.TECH. & INTELL. PROP.	Northwestern Journal of Technology and Intellectual Property (Estados Unidos)
NYU	New York University (Estados Unidos)
OEP	Oficina Europea de Patentes
OEPM	Oficina Española de Patentes y Marcas
OMPI	Organización Mundial de la Propiedad Intelectual
Op.cit.	Obra citada
PCT	Patent Cooperation Treaty
RCDI	Revista Crítica de Derecho Inmobiliario (España)
R.D.	Real Decreto
RdNT	Revista Aranzadi de Derecho y nuevas tecnologías (España)
R.D.Leg.	Real Decreto Legislativo
RDM	Revista de Derecho Mercantil (España)
RdP	Revista de Derecho Patrimonial (España)
RDPI	Revue du Droit de la propriété intellectuelle (Francia)
Rec.	Recopilación de jurisprudencia del Tribunal de Justicia de la Unión Europea.
REDI	Revista española de Derecho Internacional
Reg.PI	Registro de la Propiedad Intelectual
Rep.	Repertorio de la jurisprudencia del Tribunal de Justicia de las Comunidades Europeas (desde 1986)
Rev. OMPI	Revista de la OMPI
Rev.P.I.	Revista de la Propiedad Intelectual (España)
RGD	Revista General de Derecho (España)
RIDA	Revue Internationale de Droit d'Auteur (Francia)
RIETI	Research Institute of Economy, Trade & Industry (Japón)
Riv. Dir. Comm.	Rivista del Diritto Commerciale (Italia)
RJ	Repertorio de Jurisprudencia Aranzadi Westlaw
RPC	Reports of Patent, Design and Trade Mark Cases



Santa Clara J.C.H.T.	(Gran Bretaña) Sanata Clara Journal of Computer and High Technology Law (Estados Unidos)
SAP	Sentencia de la Audiencia Provincial
S.D.N.Y.	Southern District of New York
SIERP	Standford Institute for Economic Policy Research (Estados Unidos)
SSRN	Social Science Research Network
STJCE	Sentencia del Tribunal de Justicia de la Comunidad Europea
STJUE	Sentencia del Tribunal de Justicia de la Unión Europea
STS	Sentencia del Tribunal Supremo
Sup. Ct.	Supreme Court of the United States
TC	Tribunal Constitucional
TCE	Tratado constitutivo de la Comunidad Europea
TFUE	Tratado de funcionamiento de la Unión Europea
TI	Tecnologías de la Información
TJCE	Tribunal de Justicia de las Comunidades Europeas
TJUE	Tribunal de Justica de la Unión Europea
TRANSNAT'L LAW	Transnational Law (Estados Unidos)
TRLGDCU	Texto Refundido de la Ley general para la defensa de los consumidores y usuarios (BOE núm. 287 de 30 de noviembre)
TRLPI	Texto Refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia (Real Decreto Legislativo 1/1996, de 12 de abril) (BOE núm. 97 de 22 de abril)
TS	Tribunal Supremo (España)
Tul. L. Rev.	Tulane Law Review (Estados Unidos)
U.C. Davis L. Rev.	University of California, Davis, Law Review (Estados Unidos)
UE	Unión Europea
U. MIAMI L.REV	University of Miami Law Review (Estados Unidos)
UPLR	University of Pennsylvania Law Review (Estados Unidos)
UWG	Unlauteren Wettbewerb Gesetz (Alemani)
Vol.	Volumen
W. New Eng. L. Rev.	Western New England Law Review (Estados Unidos)
WCT	Tratado de la OMPI sobre derecho de autor adoptado en Ginebra el 20 de diciembre de 1996.
WIPO	World Intellectual Property Organization
WL	Westlaw
WPPT	Tratado de la OMPI sobre Interpretación o Ejecución y Fonogramas adoptado en Ginebra el 20 de diciembre de 1996.



## **Introducción.**



## **1. El programa de ordenador como factor clave de la Sociedad de la Información.**

Los países industrializados se han transformado en los últimos treinta años en Sociedades de Información. El progreso técnico, el descubrimiento y el desarrollo de nuevos medios de información y técnicas de comunicación han exigido desde el primer momento un ejercicio de esfuerzo y una rápida adaptación del Derecho.

La evolución de los medios de comunicación tradicionales (telégrafo, teléfono, gramófono, radio y televisión), debido al avance de la microelectrónica y al desarrollo de programas informáticos cada vez más sofisticados, y la proliferación de Internet nos han llevado a una sociedad donde se dice que “el software está por todas partes”<sup>1</sup>. La frase puede sonar a cliché pero lo cierto es que el software es uno de los motores de esta sociedad del conocimiento y va mucho más allá del ordenador personal. Pensemos en las aplicaciones de domótica que sirven para mejorar el confort de viviendas o para perfeccionar aspectos de seguridad; los denominados teléfonos móviles inteligentes (*smartphones*), los ordenadores portátiles y las tabletas; los sistemas informáticos integrados en los automóviles como el sensor de aparcamiento, sistemas de freno, control del motor, etc. El listado podría llenar páginas y páginas. Lo cierto es que, como consumidores, en términos generales confiamos en este software sin necesidad de entenderlo ni conocerlo en profundidad.

Hoy por hoy podría decirse que el software es el alma mater de la computadora. Es el factor clave que diferencia a los productos y servicios modernos. Está inmerso en sistemas de todo tipo: de transportes, médicos, de telecomunicaciones, militares, procesos industriales, entretenimientos, productos de oficina, etc. La lista es casi interminable. El software es casi ineludible en un mundo moderno.

Las “*Copyright Industries*” se han convertido en un sector de producción propio y dentro de éste, el software es su epicentro. La protección jurídica de este bien sigue planteando retos al derecho de autor, que no hacen sino poner a prueba su capacidad de evolución.

Muestra de que la aparición de los ordenadores y con ellos, los programas de ordenador, constituyeron lo que algunos autores han denominado como la segunda

---

<sup>1</sup> Cada vez más autores emplean esta expresión. Entre otros: R., PRESSMAN, *Ingeniería del Software. Un enfoque práctico*, adaptado por D., INCE, España, 5ª ed., Mc Graw Hill, 2005, pág. 4; P., NORTON, *Introducción a la computación*, México, 6ª ed., Mc Graw Hill, 2006, pág. 3.

revolución industrial<sup>2</sup> es que no han pasado todavía sesenta años desde que se planteó por primera vez la idea de vender un programa de ordenador de forma separada al ordenador, iniciativa pionera de IBM<sup>3</sup>, y el sector del software no ha dejado de crecer de tal forma que se ha convertido en uno de los grandes motores de la economía mundial<sup>4</sup>.

Dos razones justifican la afirmación anterior:

En primer lugar, el sector de las tecnologías de la información y de las comunicaciones en general (TICs), del que el software forma parte, es un sector en constante ebullición. El programa de ordenador que ayer era la última novedad, puede estar hoy obsoleto por la aparición de uno nuevo. Actualizar es una palabra clave, y sucede a un ritmo casi vertiginoso, de modo que la oferta de programas de ordenador en el mercado se ve a cada instante renovada. Las empresas siguen la máxima “*time to market*”<sup>5</sup>, es decir, ser el primero en llegar al lineal de venta, y ello supone que el factor tiempo se consagre como un elemento clave, siempre contando con la constante del riesgo de que el producto quede obsoleto antes de recuperar la inversión realizada.

En segundo lugar, vivimos en una sociedad cada vez más interconectada, donde se han modificado de manera significativa las formas de relacionarse entre personas,

---

<sup>2</sup> Expresión empleada por R., KÖHLER, *Der urheberrechtliche Schutz der Rechenprogramme*, München, Beck, 1968, pág. 1.

<sup>3</sup> Para una visión histórica completa del nacimiento de la industria del software vid. J. BAND, M. KATOH, *Interfaces on trial: Intellectual Property and Interoperability in the Global Software Industry*, Boulder, Westview Press, 1995, pág. 2; G.A., STOBBS, *Software patents*, Gaithersburg, Aspen Law & Business, 2000, págs. 38; R., MERGES, *Intellectual Property in the new technology age*, New York, Aspen, Wolters Kluwer Law & Business, 2009, pág. 603.

<sup>4</sup> Desde una perspectiva estrictamente económica, Gartner Inc, empresa líder en asesoramiento e investigación de tecnología de la información, afirmaba, en los datos presentados a principios de 2010, que a pesar de la complicada situación financiera mundial, el gasto global estimado en innovación tecnológica para este año ascendería a 3.4 trillones de dólares, un 4,6 por ciento más que el año 2009. (más información en [www.gartner.com](http://www.gartner.com)).

A nivel institucional, la Comisión Europea ha declarado: “La creación de programas informáticos ha registrado un crecimiento constante en los últimos años. Ha influido significativamente en el conjunto de la industria europea y contribuye de forma sustancial al PIB y al empleo. En 1998, el valor de mercado de los programas de ordenador en Europa ascendía a 39.000 millones de euros”. (Propuesta de Directiva del Parlamento Europeo y del Consejo sobre la patentabilidad de las invenciones implementadas en ordenador, COM (2002) 92 final, pág. 2). Más recientemente: “Los programas de ordenador están desempeñando un papel de creciente importancia en una amplia gama de sectores y, en consecuencia, cabe considerar la tecnología informática como de capital importancia para el desarrollo industrial de la Comunidad”. (Considerando Segundo de la Directiva 2009/24/CE del Parlamento Europeo y del Consejo de 23 de abril de 2009 sobre protección jurídica de los programas de ordenador (versión codificada), DO 5 de mayo de 2009, L111, pág. 16).

<sup>5</sup> K., KENNETH B., (ed.), *PDMA Handbook of New Product Development*, 2ª ed., John Wiley & Sons, 2005, págs. 173-187; disponible online: [http://www.knovel.com/web/portal/browse/display? EXT\\_KNOVEL\\_DISPLAY\\_bookid=1416&Vertical\\_ID=0](http://www.knovel.com/web/portal/browse/display? EXT_KNOVEL_DISPLAY_bookid=1416&Vertical_ID=0) (última consulta 1 de junio, 2013).

empresas, y gobiernos: las empresas usan Internet para relacionarse con clientes y proveedores, emplean herramientas de gestión del conocimiento para ser más eficientes; los gobiernos mejoran su presencia en Internet y los servicios a los ciudadanos a través de la prestación de servicios por vía electrónica y los consumidores utilizamos cada vez más herramientas informáticas en nuestras relaciones profesionales y personales.

En una sociedad de estas características, cuyo eje fundamental es el acceso a la información, el programa de ordenador se posiciona como el gran intermediario entre la información y el ser humano, cuya meta debería ser que todos nos podamos comunicar con todos sin trabas.

La propia Internet es el mejor ejemplo de ello: No importa si somos usuarios de un sistema operativo Windows, Linux o de Mac IOS: todos podemos acceder a Internet sin trabas, a cambio de que dichas comunicaciones se realicen mediante el uso de estándares abiertos como el TCP/IP, HTML, POP o SMTP.

Ahora bien, algo que tenemos tan claro en las comunicaciones de Internet, no lo tenemos tan claro con la forma en que almacenamos o distribuimos los datos, o incluso, en la forma que establecemos ciertos servicios a través de Internet. La mayor parte de los usuarios nos hemos dejado llevar de la mano por los creadores de tecnologías que tienen éxito en el mercado, que están basadas en estándares privativos y que son también tecnologías privativas.

Con independencia de la tecnología que hayamos decidido usar, lo cual es muy respetable, debemos ser conscientes de que nuestra opción tecnológica nos puede jugar malas pasadas si no tomamos ciertas precauciones.

## **2. El debate jurídico en torno a la interoperabilidad en el marco del derecho de autor comunitario.**

Las Instituciones europeas empezaron a tomar conciencia de la importancia de la interoperabilidad a principios de este siglo y dos grandes iniciativas se han puesto en marcha desde entonces.

La primera, iniciada en Junio de 2002, donde los Jefes de Estado europeos aprobaron el Plan de Acción eEurope 2005<sup>6</sup>. En él hacían un llamamiento a la Comisión

---

<sup>6</sup> Plan de Acción eEurope 2005, COM(2002)263.

Europea para que hiciesen público un marco consensuado sobre interoperabilidad para apoyar la prestación de servicios administrativos paneuropeos a los ciudadanos y las empresas.

La más reciente, la Agenda Digital Europea 2010-2020 de la Comisión, enumera entre sus pilares, la interoperabilidad y los estándares. La propia Agenda reconoce que el problema de la interoperabilidad es una cuestión prioritaria que afecta a una amplia gama de leyes y políticas sobre tecnologías de la información<sup>7</sup>.

Lo sorprendente de estas iniciativas, es que ninguna de ellas recoge entre sus medidas alguna relativa al derecho de autor sobre el programa de ordenador.

Cuando la comunidad internacional decidió proteger los programas de ordenador a través de este derecho de propiedad exclusiva, la garantía de la interoperabilidad quedó también afectada por él.

Este estudio no ha pasado por alto el hecho de que existen otros instrumentos de protección para los programas de ordenador así como para la información interoperable que contienen: el derecho de patente, el secreto comercial, el derecho contractual, incluso el derecho penal. Tampoco que el derecho de autor juega un papel importante, aunque limitado, en la protección de las innovaciones informáticas. Sin embargo, y siguiendo lo declarado en el preámbulo del Tratado WCT<sup>8</sup>, este trabajo defiende que el mantenimiento de normas que rompen el equilibrio de intereses entre creadores y nuevos competidores o que comprometen el interés público, debieran ser modificadas o derogadas, no solo por respeto al principio de proporcionalidad sino para evitar que otras ramas del ordenamiento tengan que ser modificadas, comprometiendo en algún caso sus principios fundamentales, y poniendo en jaque la seguridad jurídica de todo el sistema.

La garantía de la interoperabilidad en el derecho comunitario se contiene en una excepción al derecho de autor. El artículo 6 de la Directiva sobre protección jurídica de programas de ordenador, titulado “descompilación”, es una norma que a nuestro juicio rompe el equilibrio entre derechos de autor e interés público, que no garantiza la

---

<sup>7</sup> <http://ec.europa.eu/digital-agenda/en/our-targets/pillar-ii-interoperability-standards>

<sup>8</sup> Preámbulo del Tratado de la OMPI sobre Derecho de Autor (en adelante Tratado WCT) adoptado en Ginebra el 20 de diciembre de 1996, BOE núm. 148, de 18 junio 2010, pág. 9638: “Las Parte Contratantes, (...) *reconociendo* la necesidad de mantener un equilibrio entre los derechos de los autores y los intereses del público en general, en particular en la educación, la investigación y el acceso a la información, como se refleja en el Convenio de Berna”.



interoperabilidad y que tiene importantes repercusiones para la innovación del sector informático. En ella reside, en nuestra opinión, el epicentro del debate sobre la interoperabilidad en el derecho de autor.

Un dato muy sorprendente sobre este artículo, es que desde su aprobación hace más de veinte años, ni se han planteado cuestiones sobre su interpretación ante el Tribunal de Justicia de la Comunidad Europea, ni ha sido objeto de más que unas pocas decisiones judiciales.

Estos hechos nos han llevado a plantearnos las siguientes preguntas:

1º Si el texto de la norma es tan claro que proporciona la seguridad jurídica suficiente a los operadores del mercado como para eludir el recurso a esta norma, ¿por qué seguimos hablando de problemas de interoperabilidad afectados por el derecho de autor?

2º Si por el contrario, consecuencia de la complejidad y excesiva restricción de esta norma, se ha generado un desequilibrio entre el derecho de autor y la garantía de un interés público como la interoperabilidad, habrá que replantearse su utilidad.

El presente trabajo responde a las anteriores cuestiones y propone una solución al debate de la interoperabilidad desde el derecho de autor.

Para ello, el trabajo se ha estructurado del siguiente modo: El primer capítulo parte de una exposición de conjunto de los elementos del debate: La interoperabilidad, los programas de ordenador, la descompilación y las interfaces.

Así, en primer lugar se responde a cuestiones como: ¿qué es la información interoperable de un programa?; ¿dónde se encuentra dicha información? y ¿cómo el derecho de autor afecta el acceso y el uso de esta información? También se han analizado los efectos económicos más importantes de la interoperabilidad para el mercado de las TICs.

La segunda parte del capítulo, analiza los conceptos de programas de ordenador y descompilación, prestando especial atención a cómo este proceso de ingeniería inversa interfiere con el derecho de autor sobre el programa. La última parte del capítulo, examina las interfaces, parte específica del programa que es clave para lograr la interoperabilidad.

A partir de aquí, la investigación se centra exclusivamente en el análisis del debate desde la perspectiva jurídica. En primer lugar, presentamos el estado del debate sobre la interoperabilidad en el derecho de autor norteamericano. Las razones que justifican a la elección de Estados Unidos son que fueron los primeros en proteger los programas de ordenador a través del derecho de autor y sus tribunales, los que han tratado más cuestiones relacionadas con la ingeniería inversa por razones de interoperabilidad. Además, Estados Unidos, al contrario que los países de tradición civilista europeos, utiliza un sistema de excepciones abierto, lo cual nos ha permitido observar cuál de los dos sistemas resulta más eficiente ante los problemas planteados por la ingeniería inversa para lograr interoperabilidad.

El tercer capítulo contiene el examen del derecho europeo, cuyo epicentro es el artículo 6 de la Directiva sobre protección de programas de ordenador. Para contestar a las preguntas planteadas por este trabajo, se han realizado tres exámenes. En primer lugar, hemos analizado la Directiva desde una perspectiva histórica, para comprender cómo y porqué los legisladores europeos llegaron al texto del artículo 6. En segundo lugar, se ha realizado un examen de la norma, poniendo de relieve las dudas interpretativas que plantea, teniendo en cuenta los criterios jurisprudenciales del Tribunal de Justicia de la Comunidad Europea sobre otros artículos de la misma norma que afectan al artículo 6. En tercer lugar, se han elegido cuatro Estados miembro: Alemania, Francia, Reino Unido y España, para examinar cómo han implementado la excepción en sus ordenamientos y cómo la han puesto en práctica sus tribunales.

El cuarto capítulo examina la relación entre la Directiva sobre protección jurídica de programas de ordenador con otras dos ramas del ordenamiento jurídico a las que el texto de la norma comunitaria menciona a efectos de garantizar la interoperabilidad. En primer lugar, el Derecho de defensa de la competencia, que menciona el Considerando Decimoséptimo de la Directiva. En segundo lugar, el Derecho contractual, al que se refiere el párrafo segundo del artículo 8 de la norma comunitaria. Con este análisis se pretende dar respuesta a la cuestión de si las licencias obligatorias sobre información interoperable son la solución más eficiente para este debate y comprobar si la imperatividad de la excepción de la descompilación, tiene alguna relevancia práctica.

El capítulo se cierra con un análisis de las propuestas de *lege ferenda* más recientes sobre la descompilación; se sugiere una propuesta concreta de revisión y se pronostica alguna de las consecuencias que podrían derivarse para la interoperabilidad de una regulación más flexible sobre esta técnica de ingeniería inversa.

Finalmente, una serie de conclusiones resumen los resultados más relevantes de toda la investigación.



## **Capítulo 1**

**Los elementos del debate en el marco del Derecho de Autor.**



## **1. La interoperabilidad.**

### *1.1. La interoperabilidad: un concepto multidimensional.*

El primer reto que plantea el debate en torno a la interoperabilidad del software es la definición del propio término. Interoperabilidad es un término complejo y dinámico. A ello se suma la existencia de distintos niveles de interoperabilidad, cuestiones tecnológicas muy complicadas, y efectos económicos difíciles de explicar.

Lo que sí puede decirse con total seguridad sobre el concepto de interoperabilidad es que no existe una única definición global que recoja todos sus aspectos. La concepción más extendida del término es la que se refiere las tecnologías de la información y la comunicación, las TICs. En este contexto, interoperabilidad es la habilidad para transferir y traducir datos útiles y otra clase de información entre sistemas operativos, aplicaciones o componentes. Pero es necesario ir más allá de esta definición para entender la trascendencia de lo que la interoperabilidad conlleva en distintos contextos y a diferentes niveles.

Desde un punto de vista teórico, la interoperabilidad opera en cuatro niveles distintos de cualquier sistema complejo. Comprender esta estructura es fundamental para entender cómo la interoperabilidad funciona y cómo la sociedad puede garantizarla o frustrarla. Esto implica que la interoperabilidad no es únicamente una cuestión que afecta al flujo de datos o la información; atañe también a cuestiones esenciales de interacción humana e institucional. Por lo tanto, los problemas asociados a la interoperabilidad son tanto de índole cultural como de índole tecnológico.

Al tratar la interoperabilidad, la doctrina especializada se viene refiriendo a cuatro aspectos distintos: el aspecto organizativo, el semántico, el técnico y el temporal<sup>9</sup>. Sin embargo, en este trabajo hemos optado por una clasificación un poco diferente, resultado de combinar la clasificación clásica con la recientemente publicada por PALFREY y GASSER<sup>10</sup>, a fin de adaptarla a la cuestión central que se tratará, el debate jurídico en el marco del derecho de autor. Así, en lugar de aspectos, se habla

---

<sup>9</sup> European Interoperability Framework for Pan-European eGovernment Services (EIF), Comisión Europea, Luxemburgo, 2004, pág. 9, disponible en: <http://europa.eu.int/idabac>, (última consulta 1 de junio, 2013)

<sup>10</sup> J. PALFREY, U. GASSER, *Interop: The Promise and Perils of Highly Interconnected Systems*, New York, Basic Books, 2012, pág. 6.

cinco dimensiones: la tecnológica, la de información, la humana, la institucional y la temporal.

1. La dimensión tecnológica. Basta pensar en el hardware y en los códigos informáticos. A este nivel, interoperabilidad significa la capacidad de los programas para intercambiar información y utilizar mutuamente la información así intercambiada<sup>11</sup>.
2. La dimensión de la información. Este nivel se encuentra muy cercano al nivel anterior, el tecnológico. Es más, ambos niveles acostumbran a entrecruzarse, porque la interoperabilidad tecnológica no es posible si la información que dos sistemas intercambian no puede ser descifrada, comprendida por ambos sistemas. Si la interoperabilidad de la información falla, la interoperabilidad tecnológica es inútil. Un simple ejemplo puede ilustrar mejor esta interacción dimensional: Si alguien nos envía un correo electrónico desde su computador personal con un archivo adjunto y lo recibimos en nuestro *Smartphone* sin problemas, la interoperabilidad tecnológica se habrá producido. Pero si el software contenido en nuestro *Smartphone* no es capaz de entender y traducir la información contenida en el archivo adjunto, de manera que no podemos acceder a la información en él contenida, la interoperabilidad de la información no se habrá producido, y esta segunda hace fútil o banal que se haya producido la primera. Estas son las dos dimensiones en las que se centra el debate sobre el derecho de autor y la protección que ofrece al programa de ordenador.
3. La dimensión humana. Este nivel es mucho más abstracto que los dos anteriores, pero es de gran importancia para que se produzca la interoperabilidad. La utilización de lenguajes para codificar la información resulta clave cuando, siguiendo con el anterior ejemplo, tanto la interoperabilidad tecnológica como la de información tienen lugar. Si las dos personas que han intercambiado ese archivo, puedan leer y entender la información contenida en el archivo adjunto, se habrá logrado la interoperabilidad en las tres dimensiones hasta ahora citadas.

---

<sup>11</sup> Considerando Décimo de la Directiva Software.



El uso de lenguajes estandarizados para que además del intercambio de información tenga lugar la comunicación es el punto de partida de esta dimensión de la interoperabilidad. Además del lenguaje, la dimensión humana abarca la capacidad de trabajar juntos. La interoperabilidad suele triunfar o fallar dependiendo de la capacidad de trabajo en equipo o de lo dispuestas que estén a negociar las personas. El caso Microsoft, en el cual una empresa se niega a ofrecer a otra, bajo ningún concepto, la información interoperable de algunos de sus programas informáticos, es un claro ejemplo de esta dimensión de la interoperabilidad. Otro ejemplo es el software de código abierto, que permite el trabajo conjunto entre las personas y facilita la consecución de interoperabilidad.

4.La dimensión institucional. Este nivel es el que se encuentra a un mayor nivel de abstracción, pero es sencillo de entender si se ha comprendido la perspectiva humana de la interoperabilidad. Tan importante es que las personas trabajen en equipo como que las instituciones garanticen que lo anterior pueda tener lugar en condiciones de equidad. El ordenamiento jurídico es un claro ejemplo de la importancia que reviste esta dimensión: a través de la concesión de un derecho de propiedad exclusiva como es el derecho de autor sobre el programa de ordenador, el legislador establece las reglas de juego entre los operadores del mercado para que un nuevo competidor pueda utilizar la información interoperable de un operador que ya está en el mercado. Según estas reglas del juego faciliten o impidan esta entrada, el ordenamiento estará fomentando o no la interoperabilidad. Aquí se incluyen aspectos clave como los estándares abiertos, los servicios de interconexión, la integración de datos y el middleware, la presentación e intercambio de datos, la accesibilidad y la seguridad de los servicios<sup>12</sup>. Un claro ejemplo puede observarse en las instituciones de la Unión Europea, que han declarado la interoperabilidad en el ámbito de las TICs como asunto de interés público. La Comisión Europea lo ha declarado en diversas ocasiones en los últimos diez años, primero en la Comunicación de 2004

---

<sup>12</sup> European Interoperability Framework (EIF), *op.cit.*, pág. 16.

“Retos para la sociedad de la información europea con posterioridad a 2005”, donde afirma que a medida que crece la disponibilidad de aplicaciones basadas en las TIC, se hace mayor la necesidad de hacerlas compatibles, y reconoce que la interoperabilidad presenta múltiples facetas. Para los operadores, equivale a la posibilidad de interconexión con otras redes, mientras que para los proveedores de contenidos o servicios significa poder prestar un servicio sobre cualquier plataforma adecuada. Para los consumidores, la interoperabilidad permite adquirir un dispositivo y utilizarlo para tener acceso a distintos servicios y descargar contenidos de diversas fuentes. Además, declaraba la Comisión que los poderes públicos en determinadas circunstancias, tendrán que prestar apoyo a las partes interesadas en su búsqueda de soluciones comunes, llegando a reconocer que podría resultar preciso imponer el uso de normas abiertas<sup>13</sup>. Poco después, en la Comunicación de 2005 “i2010- Una sociedad de la información europea para el crecimiento y el empleo”, declaraba que la convergencia digital exige que los dispositivos, plataformas y servicios sean interoperables. Esta exigencia puede llevar a que en determinados casos sea necesario emplear instrumentos de obligado cumplimiento<sup>14</sup>. Más recientemente, en la Comunicación de 2010 “Una Agenda Digital para Europa”, la Comisión vuelve a insistir en la necesidad de alcanzar interoperabilidad efectiva entre los productos y servicios de las tecnologías de la información<sup>15</sup>.

5. En último lugar, la dimensión temporal. En el ordenamiento jurídico español, el Real Decreto 4/2010 de 8 de enero la define como “la dimensión relativa a la interacción entre elementos que corresponden a

---

<sup>13</sup> Comunicación de la Comisión al Consejo, al Parlamento Europeo, al Comité Económico y Social Europeo y al Comité de las Regiones de 19 de noviembre de 2004: «Retos para la sociedad de la información europea con posterioridad a 2005», COM (2004) 757 final - no publicada en el Diario Oficial, pág. 10.

<sup>14</sup> Comunicación de la Comisión al Consejo, al Parlamento Europeo y al Comité Económico y Social Europeo y al Comité de las Regiones “i2010 – Una sociedad de la información europea para el crecimiento y el empleo”, COM (2005) 229 final - no publicada en el Diario Oficial, pág. 7.

<sup>15</sup> Comunicación de la Comisión al Consejo, al Parlamento Europeo y al Comité Económico y Social Europeo y al Comité de las Regiones “Una Agenda Digital para Europa”, COM(2010) 245 final/2 – no publicada en el Diario Oficial, pág. 17.

diversas oleadas tecnológicas; se manifiesta especialmente en la conservación de la información en soporte electrónico”<sup>16</sup>. Es decir, la interoperabilidad garantiza al usuario de información digital el acceso a la misma en el futuro, sin perjuicio de los cambios en la tecnología<sup>17</sup>. Esta dimensión tiene especial relación con la privacidad y la seguridad de la información, dos de los principios fundamentales de las iniciativas institucionales antes mencionadas. La vulnerabilidad de la privacidad y la seguridad de las redes es una consecuencia de la cada vez mayor complejidad de los sistemas por razón de la interoperabilidad. El nivel de privacidad y seguridad en un sistema interoperable depende de cómo sean diseñadas, implementadas y monitorizadas las medidas en las dimensiones de tecnología, de información, humanas e institucionales. En este punto cobra también importancia la ingeniería inversa del software. La ingeniería inversa, como proceso de recuperación del diseño del programa, permite detectar vulnerabilidades del mismo que ayuden a evitar la introducción por terceros de virus informáticos, errores de programación o la facilidad para craquear el código. Un claro ejemplo del impacto de esta dimensión y de los principios a que se asocia fue lo acontecido a Sony en abril de 2011, cuando las cuentas de 77 millones de usuarios de su PlayStation Network (PSN) fueron forzadas o “hackeadas”. El sistema de juego en red se basaba en controles técnicos y de usuario que no se habían establecido de forma correcta. El “hacker” había obtenido las direcciones de los usuarios (ciudad, estado, código postal), la dirección de correo electrónico, fechas de nacimiento, claves de acceso a la red de juego, nombres de usuario, identificaciones online, y datos de sus perfiles así como datos del historial de compras, direcciones de facturación e incluso números de tarjetas de crédito y débito. Si cada sistema operativo de la PlayStation hubiese estado en aparatos independientes, si no se tratase de una red, los niveles de

---

<sup>16</sup> Real Decreto 4/2010, de 8 de enero, por el que se regula el Esquema Nacional de Interoperabilidad en el ámbito de la Administración Electrónica, BOE núm. 25 de 29/01/2010, anexo, pág. 8155.

<sup>17</sup> Esta dimensión de la interoperabilidad guarda relación con el concepto de neutralidad tecnológica, en tanto que el cambio de una tecnología a otra no conlleve unos costes inasumibles o la pérdida de los datos almacenados. (Sobre la situación de encierro o *lock-in* tecnológico, vid. infra apartado 1.3).

seguridad del sistema no hubiesen importado. Ciertamente es también que la interoperabilidad no era el problema principal, pero sí se convirtió en un problema porque los controles de seguridad y privacidad eran inadecuados.

Desde una perspectiva global, las anteriores dimensiones de la interoperabilidad, pueden verse reflejadas en distintas iniciativas electrónicas, todas ellas de gran envergadura: el comercio electrónico; los servicios prestados por las Administraciones Públicas, como servicios sanitarios, de justicia, de administración electrónica; la digitalización de información en bibliotecas o en cualquier clase de ente, tanto de naturaleza privada como pública.

Debido a este elevado nivel de asociación, y aquí radica la complejidad del término, la interoperabilidad no puede tratarse por igual en cada situación, y por ello, ofrecer una única definición del concepto que sea satisfactoria, se muestra una tarea muy complicada.

A modo ejemplificativo, algunas de las definiciones empleadas por diferentes instituciones sobre interoperabilidad:

- 1) La Decisión 2004/387/CE del Parlamento Europeo y del Consejo de 21 de abril de 2004, relativa a la prestación interoperable de servicios paneuropeos de administración electrónica al sector público, las empresas y los ciudadanos (IDABC) define interoperabilidad como “la capacidad de los sistemas de tecnologías de la información y de las comunicaciones (TIC) y de los procesos empresariales a los que apoyen, de intercambiar datos y posibilitar la puesta en común de información y conocimientos”.
- 2) El Libro Blanco de Microsoft “Interoperabilidad en la Administración Pública – Facilitando la prestación de e-Servicios”, define interoperabilidad como “la posibilidad de que los sistemas de las Administraciones Públicas trabajen juntos de forma satisfactoria y productiva – independientemente de la tecnología o la aplicación que se utilice, o qué proveedor ha suministrado el sistema adyacente”<sup>18</sup>.

---

<sup>18</sup> Microsoft, “Libro Blanco: Interoperabilidad en la Administración Pública – Facilitando la prestación de e-Servicios”, 2005, pág. 9.

- 3) En la obra *e-Government Interoperability: Guide*<sup>19</sup>, parte del programa de desarrollo de la ONU, se habla de la interoperabilidad como “la característica que permitirá el uso en conjunto de la información compilada por diferentes agencias para tomar mejores decisiones”; y como una cuestión de carácter general que afecta no solo a gobiernos de países desarrollados sino como a toda la humanidad porque contribuye a un mejor gobierno<sup>20</sup>.
- 4) El Glosario AAP-6 de la OTAN, define interoperabilidad a efectos militares como: “La habilidad de los sistemas, unidades o fuerzas para proveer servicios a y para aceptar servicios de otros sistemas, unidades o fuerzas, y para usar los servicios así intercambiados para operar efectivamente juntos<sup>21</sup>”.
- 5) El Real Decreto 4/2010 de 8 de enero por el cual se regula el Esquema Nacional de Interoperabilidad en el ámbito de la administración electrónica, define interoperabilidad como “la capacidad de los sistemas de información y de los procedimientos a los que éstos dan soporte, de compartir datos y posibilitar el intercambio de información y conocimiento entre ellos<sup>22</sup>”.

Si aceptamos la afirmación de que hoy día vivimos en un mundo interoperable, donde los productos informáticos, tanto estructurales (hardware) como coyunturales (programas), son fabricados por distintas empresas, y que dichos productos han de compartir datos tanto en redes locales como en redes globales a través de Internet, y que esos datos han de poder comunicarse entre personas, la interoperabilidad resulta ser una de las piedras angulares de la industria informática<sup>23</sup> y un factor clave para la sociedad del siglo XXI.

Ante la dificultad de dar una definición del término más allá de los ejemplos citados y vistas las diferentes dimensiones del mismo, lo que sí es posible es diferenciar interoperabilidad de otros conceptos a los que normalmente se la asocia, pero que no son sinónimos.

---

<sup>19</sup> E.C., LALLANA, following inputs from the UNDP GIF Study Group, *e-Government Interoperability: Guide*, UNDP, 2007, pág. 1.

<sup>20</sup> La definición original dispone: “*Interoperability will allow data compiled by different agencies to be used together to make better decisions*”.

<sup>21</sup> NATO Glossary of Terms and Definitions (English and French), version 2, 2012, pág. 2-F-5.

<sup>22</sup> Real Decreto 4/2010 de 8 de enero, *op.cit.*, Sec. 1, pág. 8139.

<sup>23</sup> J. BAND, M., KATOH, *Interfaces on Trial 2.0, versión pre publicación*, Cambridge, London, The MIT Press, 2010, pág. 1, disponible en: [mitpress.mit.edu/books/full\\_pdfs/Interfaces\\_on\\_Trial\\_2.0.pdf](http://mitpress.mit.edu/books/full_pdfs/Interfaces_on_Trial_2.0.pdf), (última consulta 1 de junio, 2013); P., POIKELA, *IPR Protection of application programming interfaces*, Saarbrücken, LAP, Lambert Acad. Publ., 2010, pág. 17.

El primer falso sinónimo de interoperabilidad que se ha utilizado con frecuencia es “**compatibilidad**”.

La compatibilidad informática significa que dos productos (sistemas operativos, aplicaciones o componentes) pueden funcionar de manera conjunta sin producir errores. Ambos productos se comprenden correctamente, sin embargo desconocen porqué y cómo se produce la comunicación. En este caso se podría hablar de una coexistencia pacífica de productos en el mercado, pero no de interoperabilidad. Por ejemplo, hay programas relacionados con la seguridad informática, como los antivirus, que por regla general no son compatibles entre sí de modo que si tenemos dos antivirus instalados en nuestro ordenador, si los ponemos a funcionar simultáneamente, colapsarán el sistema operativo.

Otro ejemplo ilustrativo de la diferencia entre compatibilidad e interoperabilidad puede verse en un videojuego. Para poder funcionar en una computadora, puede requerir algún tipo de sistema operativo específico, de manera que sea compatible con el sistema operativo Linux, y no con el sistema operativo Microsoft Windows, o viceversa. Para que el fabricante del videojuego pueda poner a disposición del consumidor en el mercado distintas versiones del videojuego, compatibles con según cual sistema operativo, normalmente habrá tenido que negociar con los fabricantes de los sistemas operativos con los que quiera que su videojuego sea compatible las condiciones de compatibilidad. Si al fabricante del sistema operativo le interesa que ese videojuego sea compatible con su software, probablemente habrá firmado un contrato de licencia con el fabricante del videojuego, a través del cual le proporciona una información que está codificada, contenida en unos componentes denominados librerías y APIs<sup>24</sup>, que el fabricante del videojuego no podrá descodificar. Esta es una práctica respetable en el mercado, pero no es interoperabilidad.

---

<sup>24</sup> Las librerías, también denominadas bibliotecas informáticas o *libraries* en inglés, son un conjunto de subprogramas utilizados para desarrollar software. Las bibliotecas contienen código y datos, que proporcionan servicios a programas independientes, es decir, pasan a formar parte de estos. API (del inglés *Application Programming Interface*, y en español, interfaz de programación de aplicaciones – IPA) es una clase de interfaz que contiene un conjunto de funciones y procedimientos que ofrece cierta librería para ser utilizado por otro programa como una capa de abstracción. Sobre esta clase de interfaces vid. infra. apartado 4.1.

La interoperabilidad implica que el fabricante del videojuego tendría acceso a la información codificada que facilita el funcionamiento entre su programa y el sistema operativo, sin tener que depender de la voluntad del fabricante del sistema operativo.

En condiciones ideales de interoperabilidad, cualesquiera dos programas de ordenador que un consumidor elija podrán funcionar conjuntamente y del mismo modo esos programas con cualesquiera otros. Que el fabricante de uno de los programas tenga una mejor posición en el mercado no es un factor determinante en una situación de interoperabilidad porque el proveedor no tiene el control sobre las condiciones de compatibilidad. Así, en mercados caracterizados por interoperabilidad, el éxito de un programa no dependerá únicamente de que éste sea más o menos compatibles con un sistema dominante o viceversa, sino porque la aplicación es sencilla o el servicio es más rápido, o más seguro, o más confiable, o tiene una mejor interfaz de usuario o proporciona un servicio de mayor calidad. La decisión final del consumidor no está condicionada por el grado de compatibilidad del programa<sup>25</sup>.

La interoperabilidad se basa en estándares abiertos, que en teoría no favorecen a ningún fabricante de software sobre los demás. El mejor ejemplo de interoperabilidad lo proporciona Internet. Da igual si somos usuarios de un sistema operativo Windows, Linux o Mac IOS, cualquiera de ellos puede acceder a Internet sin trabas, a cambio de que dichas comunicaciones se realicen mediante el uso de estándares abiertos como el HTML, POP, SMTP, etc.

El segundo término con el cual en ocasiones hay confusión es **“intraoperabilidad”**.

Intraoperabilidad es lo opuesto a interoperabilidad. En una situación de intraoperabilidad existe no sólo un programa informático efectivamente dominante, central, sea por su grado de penetración en el mercado, por su posición de dominio o por el nivel de aceptación del consumidor, sino toda una plataforma. La compatibilidad con este plataforma se produce a través de estándares privativos, establecidos y controlados por su fabricante, y que no suelen adecuarse a las definiciones de estándares de

---

<sup>25</sup> P. SAMUELSON, “Computer Programs, user interfaces and Section 102(b) of the Copyright Act of 1976: A critique of Lotus v. Paperback”, High Tech. L. J. 6, 1991, pág. 209 (268); D., LIM, “Copyright under siege: An economic analysis of the essential facilities doctrine and the compulsory licensing of copyright works”, Alb. L.J.Sci. & Tech, 17, 2007, pág. 481 (522).

organizaciones neutrales<sup>26</sup>. En esta situación es probable que un nuevo competidor en el mercado tenga interés económico en aceptar los términos y condiciones de compatibilidad impuestos por el fabricante, aunque al mismo tiempo ello suponga la prestación de un consentimiento tácito de participar en un entorno asimétrico, intraoperable, donde las mayores ventajas, por no decir, todas las ventajas, van a parar a ese fabricante central.

Desde la perspectiva de los consumidores, esta situación de intraoperabilidad conllevará que el consumidor verá reducidas las opciones que pueda tomar a la hora de elegir un programa o un dispositivo electrónico, ya que de elegir uno que proceda de un fabricante central, sus posteriores adquisiciones se verán siempre condicionadas a que puedan conectarse con él. Dicho de otro modo, la postura del propietario de la plataforma ante el consumidor es: “si quieres interoperabilidad, cómprame solo a mí tus programas y dispositivos. No te preocupes, yo te garantizo que funcionarán sin problema”. Posiblemente el consumidor tendrá que pagar un precio mayor por esa posterior adquisición, al tratarse de un producto protegido y garantizado del fabricante central.

Hasta hace pocos años, la situación de la industria y el mercado del software eran de intraoperabilidad casi al 100%. No fue hasta la decisión en el caso Microsoft, tanto en la Unión Europea como en Estados Unidos, que las instituciones y la sociedad han cuestionado la legitimidad de estas situaciones. Cuando el propietario de tecnologías centrales no estandarizadas es capaz, con sus comportamientos, de frenar la innovación en beneficio propio, y/o producir situaciones anticompetitivas en el mercado, la legitimidad de su comportamiento desaparece.

Por lo tanto, la interoperabilidad con carácter global podría definirse como un concepto multidimensional que conlleva un reto tecnológico, aspectos legales, organizativos y semánticos de tratamiento de la información<sup>27</sup>. La frase “el software está en todas partes” es una realidad en nuestra era y por tanto, la garantía de interoperabilidad una necesidad fundamental.

---

<sup>26</sup> Entre otras: IEEE (Institute of Electrical and Electronic Engineers - [ieeexplore.ieee.org](http://ieeexplore.ieee.org) -), ETSI (European Telecommunications Standards Institute - [etsi.org](http://etsi.org) -) or IDABC (European Union's programme for Interoperable Delivery of European eGovernment Services to public Administrations, Businesses and Citizens - [ec.europa.eu/idabc](http://ec.europa.eu/idabc) -).

<sup>27</sup> The European eGovernment Action Plan 2011-2015, COM (2010) 743, 15 de Diciembre 2010, pág. 14.



Aclarados todos los aspectos que envuelve la interoperabilidad y la multidimensionalidad del término, corresponde indagar en las dos dimensiones que afectan al debate jurídico suscitado por este término en el marco del Derecho de autor: las dimensiones tecnológica e informativa de la interoperabilidad.

### *1.2. La interoperabilidad vertical y horizontal.*

Todos los usuarios de tecnologías queremos que nuestro ordenador personal pueda funcionar con nuestra cámara y con nuestro *Smartphone*. Nada saca más de quicio a los consumidores en la economía digital que los sistemas tecnológicos que no son capaces de funcionar sin problemas de manera conjunta. La consecuencia inmediata de esta situación es que la interoperabilidad vende. Este aspecto comercial de la interoperabilidad puede apreciarse más claramente en las dos dimensiones menos abstractas del término: la técnica y la de información. Si un sitio web no puede cargar vídeos de manera simple, porque necesita que instalemos un complemento (plug-in) que complica el proceso, probablemente tendrá menos visitas que otro que no necesite la instalación de plug-in. El programa de procesamiento de textos que es capaz de reconocer y leer documentos en la mayoría de los formatos existentes en el mercado sin esfuerzo, tendrá más usuarios que aquel otro procesador que aún siendo de código abierto, requiere mayor esfuerzo y conocimientos por parte del usuario.

La mayoría de usuarios de tecnologías de la información realizamos como regla general cuatro acciones básicas: enviamos y recibimos correos electrónicos; escribimos, editamos y almacenamos documentos; enviamos y recibimos datos y señales de voz a través de dispositivos móviles, y escuchamos música. Para realizar cualesquiera de estas funciones, disponemos de una amplia variedad de productos entre los que escoger. Podemos elegir entre los productos y servicios de plataformas como Microsoft, Google o Apple, entre otras opciones. Lo más sencillo suele ser utilizar los productos y servicios de la misma plataforma, aunque también tenemos la opción de combinar distintos productos y servicios de distintas plataformas. Sin embargo esto último, requiere mayor habilidad por nuestra parte.

La primera opción refleja una situación que garantiza lo que se ha denominado interoperabilidad vertical. Es decir, nuestros datos han de poder ser intercambiados e

interpretados por nuestros dispositivos y programas, para que se puedan utilizar sin restricciones dentro de una misma organización, o para que los podamos usar sin trabas con los distintos programas que usamos siempre que provengan del mismo fabricante. No importa la dimensión del sistema y garantiza la experiencia más satisfactoria para el usuario. Así, si editamos un documento con un procesador de texto de Microsoft, ese documento será 100% interoperable en cualquier dispositivo Microsoft.

La segunda opción se refiere a la interoperabilidad horizontal, a la capacidad de intercambiar información con organizaciones distintas a la nuestra, con usuarios de otras plataformas y con nosotros mismos, si optamos por combinar dispositivos y programas de diferentes plataformas. En este caso, los estándares abiertos y la ingeniería inversa cobran relevancia para garantizar la interoperabilidad.

Si la industria informática utilizara estándares abiertos en sus programas, la interoperabilidad horizontal estaría garantizada, pero la realidad es bien distinta. La ausencia de una política de estándares oficiales consolidada en el sector informático y la presencia de estándares *de facto* de carácter privativo han hecho de la interoperabilidad horizontal una cuestión complicada, pero no de imposible solución. Así, recientemente en el marco de la Agenda Digital de la Unión Europea, se ha aprobado un Reglamento en septiembre de 2012 sobre la normalización europea, en el que por primera vez se definen los requisitos para la identificación de las especificaciones técnicas de las TIC<sup>28</sup>. En relación a los estándares abiertos, si bien no existe una definición universalmente aceptada de los mismos, en el ámbito de la Unión Europea, el Marco Europeo para la Interoperabilidad los ha definido como aquellos que satisfacen las siguientes condiciones<sup>29</sup>:

- Los costes de uso del estándar son bajos y no constituyen un obstáculo para acceder a él.
- El estándar se ha publicado.

---

<sup>28</sup> Reglamento (UE) nº 1025/2012 del Parlamento Europeo y del Consejo, de 25 de octubre de 2012, sobre la normalización europea, por el que se modifican las Directivas 89/686/CEE y 93/15/CEE del Consejo y las Directivas 94/9/CE, 94/25/CE, 95/16/CE, 97/23/CE, 98/34/CE, 2004/22/CE, 2007/23/CE, 2009/23/CE y 2009/105/CE del Parlamento Europeo y del Consejo y por el que se deroga la Decisión 87/95/CEE del Consejo y la Decisión nº 1673/2006/CE del Parlamento Europeo y del Consejo, DO nº 316 de 14 de noviembre, 2012, L-2012-82195, Anexo II, pág. 89.

<sup>29</sup> “European Interoperability Framework ...”, *op.cit.*, pág. 9.

- El estándar se ha adoptado en el marco de un proceso de decisión abierto (acuerdo por consenso o por mayoría, etc.)
- Los derechos de propiedad intelectual del estándar se encuentran depositados en una organización sin ánimo de lucro que opera sobre la base de una política de libre acceso.
- No hay restricciones para la reutilización del estándar.

En cuanto a la ingeniería inversa, en ocasiones, ante la ausencia de interoperabilidad entre programas y de la imposibilidad de acceder a la información que la proporciona (sea por cuestiones de costes o por la negativa a proporcionar dicha información de su propietario), un tercero que quiera entrar a competir en el mercado puede optar por esta opción para alcanzar la interoperabilidad horizontal entre su nuevo producto y los de otros fabricantes. Esta opción es arriesgada por dos motivos. En primer lugar por los costes que pueda llevar asociados, que se examinarán más adelante, y que pueden suponer una alta barrera de entrada al mercado. En segundo lugar, porque la protección que confiere el Derecho de autor sobre el programa de ordenador en la Unión Europea, plasmada en la Directiva de software, establece un marco más que cuestionable para la ingeniería inversa sobre el programa, que ofrece poca o ninguna seguridad jurídica a los operadores del mercado y no garantiza la interoperabilidad de la información entre tecnologías. De tal modo, que a los costes intrínsecos del procedimiento hay que añadir el posible cometimiento de una infracción del ordenamiento jurídico vigente, lo cual complicada bastante la consecución de esta clase de interoperabilidad.

### *1.3. El papel de la interoperabilidad en el mercado del software.*

La última pieza del puzle para comprender porqué la interoperabilidad informática es materia de interés público, pasa por entender cuáles son los efectos que la ausencia o presencia de interoperabilidad producen dentro del sector de las tecnologías de la información y de la comunicación. Aquí se encuentran las claves económicas que mueven a los operadores del mercado a facilitar o impedir la interoperabilidad con sus

programas. No puede perderse de vista que la interoperabilidad afecta no solo a la competencia del mercado, sino también al consumidor y a la innovación en el sector.

Desde el punto de vista económico, el éxito de una plataforma informática se representa a través de una curva de retroalimentación positiva: cuanto mayor sea el número de programas de aplicación disponibles para la plataforma, el mayor atractivo que en general tendrá la plataforma en el mercado, y el mayor número de consumidores que la plataforma tendrá. Este bucle de retroalimentación, al que se denomina *network effect*<sup>30</sup>, tiene una importancia decisiva en las motivaciones que llevan a las empresas a publicar o no la información que posibilita la interoperabilidad<sup>31</sup>. Empresas como Apple, Microsoft o Facebook, por nombrar solo tres, se enfrentan al reto de desarrollar y ajustar sus estrategias competitivas continuamente para beneficiarse de los *network effects*. Las decisiones estratégicas que tomen grandes empresas como estas tienen un profundo impacto en la cuestión fomentar o no la interoperabilidad<sup>32</sup>. Dicho de otro modo, ¿por qué empresas tecnológicas como Microsoft, adoptan una estrategia de negocio que depende en última instancia de controlar la interoperabilidad? La respuesta es que si se limita la interoperabilidad, en ocasiones, los beneficios para la fabricante son mayores que si se facilita.

El creador de un programa que pueda permitirse controlar el nivel de interoperabilidad de los demás competidores en el mercado, tendrá una posición privilegiada. Por ejemplo, cuando Apple lanzó iTunes al mercado para comprar música digital a los consumidores de su tecnología *iPod*, la pretensión era establecer una red propia y generar así unos *network effects* sin competencia directa de cualquier otra plataforma musical digital. El considerable éxito de Apple con su estrategia motivó que un competidor, RealNetworks, realizara un proceso de ingeniería inversa sobre la tecnología de Apple para hacer que su reproductor *RealPlayer* fuese compatible con la música de *iTunes*. A pesar de que la respuesta inicial de Apple fue amenazar a RealNetworks con demandarlos por haber infringido sus derechos de propiedad

---

<sup>30</sup> Según KEMPER, los *network effects* son “the changes in decision variables of an economic agent, such as benefits, are based on choices of other agents consuming similar goods”, vid. A., KEMPER, *Valuation of Network Effects in Software Markets. A Complex Networks Approach*, Springer, Berlin, Heidelberg, 2010, pág. 67.

<sup>31</sup> J., FARRELL, “Standardization and Intellectual Property”, *Jurimetrics*, 30, 1989, pág. 35 (36).

<sup>32</sup> U. GASSER, U., J. PALFREY, “Breaking down digital barriers: when and how ICT interoperability drives innovation”, Berkman Publication Series, Noviembre 2007, pág. 9, disponible en: <http://cyber.law.harvard.edu/interop> (Última consulta 1 de junio, 2013).

intelectual, Apple decidió modificar la interfaz de *iTunes* para inutilizar la interoperabilidad de *RealPlayer* con *iTunes*.

*A). La atipicidad del mercado: software, network effects e interoperabilidad.*

El mercado del software está dominado por los antes mencionados *network effect*, y aunque esta no es una característica exclusiva de este sector<sup>33</sup>, sin embargo unido a las características del software y a la interoperabilidad, el análisis económico de este sector del mercado muestra que nos encontramos ante un mercado atípico<sup>34</sup>. Veamos las razones que sostienen esta afirmación:

En relación al software, se presenta en el mercado como un bien para el que no existe rivalidad en el consumo porque, en primer lugar, que un usuario lo utilice no impide que otro pueda hacerlo también y en segundo lugar, porque es parcialmente excluible a través de mecanismos de precios<sup>35</sup>. El proveedor tiene la capacidad de poner un precio en cualquiera o en ambos extremos de la red. Un operador de telecomunicaciones podría fijar un precio por la llamadas que realiza un cliente, por las que recibe o por ambas. En el mercado del software, los proveedores también eligen el extremo de la red que desean cobrar. Un ejemplo claro es Adobe, que distribuye el programa Acrobat Reader de forma gratuita, mientras vende Acrobat Distiller, el programa que permite la creación de archivos que pueden ser leídos por el primero.

---

<sup>33</sup> Los efectos en red están presentes en otros sectores industriales como las telecomunicaciones, el transporte, la electricidad, la banca o la atención sanitaria. Vid. P. DAVID, “Clio and the economics of qwerty”, *American Economic Review*, vol. 2, n° 75, 1985, pág. 332 (333); N., ECONOMIDES, L., WHITE, “One-way networks, two-way networks, compatibility, and public policy”, Department of Economics Working Paper ED-93-14, NYU, 1993, pág. 1331 (1355); N., ECONOMIDES, “Public Policy in Network Industries”, Law and Economics Research Paper No. 06-49, Septiembre, NYU, 2006, pág. 3; disponible en SSRN: <http://ssrn.com/abstract=936469> o <http://dx.doi.org/10.2139/ssrn.936469> (última consulta 1 de junio, 2013).

<sup>34</sup> M.M., MÄKELÄ, “Software Business: Position As a Field of Research and Avenues for Scholarly Contributions”, 14th International Conference of the International Association for Management of Technology (IAMO), Viena, Austria, Mayo 22-26, 2005, pág. 1 (3). Un estudio completo puede verse en D.G., MESSERSCHMITT, C. SZYPERSKI, “Industrial and Economic Properties of Software: Technology, Processes, and Value”, Report UCB//CSD-00-1130, Berkeley University, California, (2000), disponible en: <http://www.eecs.berkeley.edu/~messer/PAPERS/01/Software-econ/Software-econ-TR-footnotes.PDF>, (última consulta 1 de junio, 2013)

<sup>35</sup> Vid. F., LÉVÊQUE, Y., MÉNIÈRE, *The Economics of Patents and Copyright*, Paris, Berkeley Electronic Press, 2004, pág. 79, disponible en: [ssrn.com/abstract=642622](http://ssrn.com/abstract=642622) (última consulta 1 de junio, 2013); N., ECONOMIDES, “The economics of Networks”, *International Journal of Industrial Organization*, vol. 14, n° 2, 1996, pág. 1 (3).

Otra especial característica del software es que la inversión necesaria para generar software se produce en su mayor parte en la fase de desarrollo, mientras que los costes de producción son prácticamente cero.

En tercer lugar, el software, aunque en el mercado se presenta como un producto, debido a su inmaterialidad, se comporta como un servicio, sin perjuicio además de que puede ser adaptado y personalizado a las necesidades de los distintos segmentos de mercado, áreas de negocio y clientes. Se trata de un bien que se valora por su comportamiento o funcionalidad<sup>36</sup>.

Desde el punto de vista de los *network effects* en el mercado, se produce lo que en Economía se denomina una externalidad cuando un nuevo cliente que entra a formar parte de la red. Es decir, el adquirente de un nuevo programa normalmente no es recompensado por el beneficio que produce en los demás a causa de su entrada, de manera que los costes sociales son mayores que los costes del productor, o los beneficios sociales superan a los beneficios del productor<sup>37</sup>.

Los *network effects* también conllevan que la senda de penetración o de expansión en la red sea mucho más rápida que en sectores donde no se producen estos efectos.

La tercera característica a destacar de los *network effects* es la posible aparición de guerras de estándares. La interoperabilidad es la cualidad que permite hacer realidad la complementariedad entre productos; en algunas ocasiones esto sucede de manera directa, pero en la mayoría de las ocasiones la interoperabilidad sólo puede lograrse a través de la aceptación de unos estándares técnicos específicos. Los creadores de programas tienen en su mano hacerlos parcial o totalmente incompatibles con el resto, mediante la creación de diseños propietarios y rechazando la interconexión, impidiendo el acceso a competidores a la información interoperable.

---

<sup>36</sup> Vid. M.M., MÄKELÄ, “Software Business...”, *op.cit.*, pág. 2; D.G., MESSERSCHMITT, C. SZYPERSKI, “Industrial and Economic Properties ...”, *op.cit.*, pág. 4; S. VON ENGELHARDT, “The Economic Properties of Software”, Jena Economic Research Papers, Max Plank Institute of Economics, nº 45, 2008, pág. 1 (19); U., MYLLY, “An Evolutionary Economics Perspective on Computer Program Interoperability and Copyright”, IIC 41(3), 2010, pág. 284 (288); P., SAMUELSON, S., SCOTCHMER, “The Law and Economics of Reverse Engineering”, Yale Law Journal vol.111(7), 2002, pág. 1575 (1613).

<sup>37</sup>Vid. J., PASCUAL Y VICENTE, *Diccionario de derecho y economía de la competencia en España y Europa*, Madrid, Civitas, 2002, pág. 219.

Otra característica significativa de los *network effects* es la desigualdad que crean en el mercado, consecuencia del éxito social de una red creada en torno a un producto concreto.

Los mercados con fuertes *network effects* donde los operadores eligen sus propios estándares, son mercados con grandes desigualdades en términos de cuotas de mercado y beneficios, ya que un operador con una cuota de mercado elevada puede ofrecer mayor variedad de productos complementarios y, por tanto, su producto será más valorado por los consumidores, lo que repercute en mayores ventas. Un ejemplo es el mercado de sistemas operativos para ordenadores personales.

Esta estructura monopolística también puede maximizar el beneficio social: cuando los *network effects* son muy fuertes y conllevan una cuota de mercado muy elevada para una plataforma concreta, tanto consumidores como operadores salen beneficiados. De hecho, una situación en la que un monopolio de esta clase se escinde en dos operadores con sistemas no interoperables reduciría el beneficio social en vez de incrementarlo, ya que los *network effects* serían menores. Por ello, no debe presumirse la existencia de acciones anticompetitivas al observar cuotas desiguales de mercado o elevados beneficios de un operador.

Además, esta estructura de desigualdad no implica que no exista competencia en el mercado. Lo que sucede es que la competencia es de naturaleza distinta: se compete por el mercado, en vez de “en el mercado”. Un contexto donde el operador que crea el programa o la plataforma predominante obtiene gran parte del beneficio, suele llevar a una competencia intensa para convertirse en el operador dominante. Es el tipo de estrategias que se han observado en torno a operadores como Google, Amazon o eBay.

En relación a la interoperabilidad como elemento que condiciona la atipicidad del mercado de software, hay que situarse en el lado de la oferta. La consecución de interoperabilidad es el reto constante de la industria que se manifiesta en la satisfacción del usuario. Es debido a los elevados costes de alcanzar interoperabilidad que entran en juego los procesos de ingeniería inversa. Por supuesto, cualquier nuevo operador del mercado o aquel que no quiera quedarse fuera valorará una serie de factores antes de optar por acudir a la ingeniería inversa. Factores que con carácter general serán tenidos en cuenta son: el objetivo de la ingeniería, el contexto industrial donde tiene lugar, el

coste de realización, el coste temporal, así como si existen vías alternativas a la ingeniería inversa como el licenciamiento, y el uso de la información adquirida durante el proceso<sup>38</sup>.

La consecución de interoperabilidad a través de un proceso de esta clase o a través de otras vías, puede producir una serie de efectos económicos en el sector, tanto en el lado de la oferta como en el de la demanda, de una magnitud suficiente como para situar al legislador en un papel clave a la hora de mantener el equilibrio entre los derechos de todas las partes implicadas, en relación al modo de acceder a la información necesaria para lograr el objetivo de la interoperabilidad.

#### *B). Efectos económicos de la interoperabilidad.*

Para examinar los efectos que la interoperabilidad ocasiona en el mercado de software, es necesario diferenciar entre las distintas clases de *network effects* que puedan producirse. Según la interoperabilidad tenga lugar en una u otra clases de efectos, las consecuencias podrán ser positivas o negativas para el sector.

##### *B.1). Clases de network effects.*

Los *network effects* directos u horizontales significan que el valor del programa no depende de sus cualidades intrínsecas sino que su valor tiende a incrementarse cuanto mayor es el número de usuarios de ese producto o de productos compatibles con el primero<sup>39</sup>. Es decir, cuanto mayor sea el número de personas con un sistema operativo determinado, mayor es la utilidad de poseer uno. Lo mismo puede predicarse de un sistema de correo electrónico o de un sistema de mensajería instantánea.

Esta clase de efectos se producen si existe interoperabilidad vertical: a mayor capacidad de interoperabilidad de un programa, mayor será la capacidad funcional de este con un mayor número de programas y dispositivos que formen parte de la misma organización o provengan del mismo fabricante. Desde esta perspectiva, la interoperabilidad aumenta la funcionalidad del programa en el mercado<sup>40</sup>.

---

<sup>38</sup> Vid. P., SAMUELSON, S., SCOTCHMER, "The Law & Economics ...", *op.cit.*, pág. 1586.

<sup>39</sup> N., ECONOMIDES, "The economics of ...", *op. cit.*, pág. 9.

<sup>40</sup> A., KEMPER, *Network...*, *op.cit.*, pág. 70.



Frente a los efectos directos, existen los efectos indirectos o verticales, donde el valor adicional del producto deriva de la dependencia colectiva de algún tipo de bien complementario, como la información disponible o la existencia de aplicaciones complementarias al producto. En este caso, el consumidor no se beneficia directamente de un mayor tamaño de la red, sino que, la mayor dimensión de la red propicia mayores inversiones en productos complementarios, y esta mayor inversión genera valor para el usuario de forma indirecta. Son el reflejo de la interoperabilidad horizontal.

Traducido en términos económicos, los efectos verticales aumentan la demanda del programa<sup>41</sup>. El ejemplo más claro es Internet; cuanto mayor sea el número de usuarios de la Red, mayor será la cantidad de contenido que los proveedores estén dispuestos a proporcionar, lo cual indirectamente beneficia al usuario. En este caso, la interoperabilidad tendrá especial relevancia en el mercado secundario o de productos complementarios.

*B.2). Efectos positivos: mejora de la competitividad y aumento del bienestar del consumidor.*

En una situación donde se produzcan *network effects* directos sobre un programa determinado, la interoperabilidad se verá condicionada por la presencia de interfaces de información abiertas o cerradas. Las primeras van a permitir la creación de nuevos programas interoperables con el programa originario, favoreciendo la interoperabilidad horizontal, y las segundas la obstruirán, favoreciendo la interoperabilidad vertical.

Los usuarios de programas interoperables con el programa originario forman parte de la misma red, pero los usuarios de programas que no puedan interoperar con él se quedarán fuera de la red y tampoco podrán formar una red propia. Por lo tanto, las acciones que favorezcan la interoperabilidad horizontal parecen las más beneficiosas para el consumidor.

En cuanto al fabricante del programa originario, que su software disfrute de efectos directos significa que cuanto más demanda del mismo haya, más crecerá también la oferta. A la inversa, un programa que no puede interoperar con los demás y cuyo grado de penetración en el mercado es escaso, formará una red que se irá

---

<sup>41</sup> *Ibidem*, pág. 71.

empequeñeciendo a cada paso (a menor demanda, menor oferta). El punto de inflexión en una situación como cualquiera de las dos expuestas, se produce cuando el fabricante del programa que sufre el efecto directo negativo, pretende engancharse a la red de aquel que disfruta de un efecto directo positivo, accediendo a la información interoperable de aquél.

En una situación como esta, el derecho de autor puede usarse para controlar el uso y acceso a las especificaciones de las interfaces, jugando un papel fundamental. Si el fabricante del programa que está fuera de la red positiva decide emplear un proceso de ingeniería inversa para lograr interoperabilidad con el programa del primer fabricante, y así no salir del mercado, que este hecho se califique o no como infracción de los derechos de autor es determinante<sup>42</sup>.

Cuando el fabricante del programa que disfruta de los efectos red positivos decide publicar o poner las especificaciones de la interfaz a disposición de otros operadores a través de licenciamiento, convirtiendo las técnicas de ingeniería inversa en innecesarias, facilitando que las aplicaciones existentes y las nuevas puedan interoperar con su software, puede ser beneficioso para los consumidores, pero arriesgado para el operador que ocupa una mejor posición en el mercado. Por ejemplo, Hewlett-Packard y Dell son operadores que se dedican, entre otras cosas, a fabricar PCs compatibles con IBM, y cuando IBM decidió publicar su arquitectura de sistema, ambos operadores aumentaron su cuota de mercado e IBM perdió su mejor posición respecto a ellas<sup>43</sup>. Los grandes beneficiados de una situación como la descrita son los consumidores, porque la mayor competencia entre los operadores y la interoperabilidad entre sus programas conllevan, en primer lugar, un aumento del número de usuarios de esa clase de programas, y de ese aumento puede nacer un estándar en el mercado, que impida el efecto de exclusión o lock-in del consumidor.

En ausencia de interoperabilidad, un consumidor que utilice un sistema de software, una plataforma, y quiera migrar a otra distinta, tendrá que asumir todos los costes de cambio: la nueva infraestructura de software, la formación personal para manejar el programa y la sustitución de las aplicaciones complementarias del programa,

---

<sup>42</sup> Vid. P., SAMUELSON, S., SCOTCHMER, "The Law and Economics...", *op. cit.*, , pág. 1624 (1624); A. VAN, ROOIJEN, *The software interface between Copyright and Competition Law, A legal analysis of Interoperability in Computer Programs*, Alphen aan den Rijn, Kluwer Law International, 2010, pág. 24.

<sup>43</sup> P. SAMUELSON, S., SCOTCHMER, "The Law and Economics...", *op. cit.*, pág. 1617.

pudiendo extenderse incluso al hardware, si necesita otro de distintas características. A ello se suman también los costes de ajuste, es decir, el coste derivado de migrar todos los datos del primer programa al segundo. Al no haber interoperabilidad entre los sistemas, nada impide que el proveedor del segundo software intente incrementar los costes de ajuste no facilitando apoyo para la portabilidad de sus aplicaciones de software, de forma que crea una barrera que dificulta la transición de una plataforma a otra. En una situación de lock-in o encierro como esta, las técnicas de ingeniería inversa pueden ser muy valiosas<sup>44</sup>.

Claros ejemplos de este efecto de encierro o lock-in, son el caso del teclado QWERTY<sup>45</sup> o del formato de vídeo VHS. Ni el teclado diseñado por Sholes en 1868 era el más eficiente ni el formato de vídeo VHS era el mejor a nivel técnico, pero ambos fueron socialmente aceptados, convirtiéndose en estándares de facto, haciendo que la demás competencia desapareciese. Por ejemplo, en el caso de QWERTY, el teclado DVORAK, no obtuvo apenas difusión, y en el caso de VHS, el formato BETAMAX apenas tuvo repercusión en el mercado, debido al alto coste de cambio que tendría que asumir un consumidor para cambiar de tecnología<sup>46</sup>.

Otro ejemplo más reciente en el ámbito de los programas fue el caso Microsoft, en el cual, un competidor, Sun Microsystems, para evitar los costes de cambio del sistema operativo Windows solicitó a Microsoft la información interoperable de su sistema operativo para desarrollar una nueva plataforma que funcionase como ese sistema operativo pero que en realidad sólo sería una capa adicional colocada sobre él, con interfaces claramente definidas. Una máquina virtual que permitiría a usuarios de otros sistemas operativos interoperar con el sistema operativo Windows. La postura de

---

<sup>44</sup> Sobre situaciones de encierro e interoperabilidad vid., entre otros: S.N., DUPONT, “The Copyright and Trade Secret Protection of Communication Software: Placing a Lock on Interoperability”, J. Marshall J. Computer & Info. L., vol. 13, 1994-1995, pág. 17; R., SUTOR, “Software Standards, Openness and Interoperability”, en L., DENARDIS, (ed.), *Opening Standards: The Global Politics of Interoperability*, Cambridge, Massachusetts, The MIT Press, 2011, pág. 209.

<sup>45</sup> Una descripción completa del caso puede verse en P., DAVID, “Clio and the Economics ...”, *op.cit.*, págs. 332; T., VINJE, “El desarrollo de productos interoperables bajo la directiva sobre software de la CE”, RGD, año L, núm. 589-90, octubre-noviembre, 1993, pág.10133.

<sup>46</sup> Sobre la competencia entre los formatos de vídeo VHS y BETAMAX, vid. <http://www.mediacollege.com/video/format/compare/betamax-vhs.html> (Última consulta 1 de junio, 2013).

Microsoft fue rechazar ofrecer esa información, a fin de mantener sus costes de cambio elevados<sup>47</sup>.

Los costes de cambio son la regla general en los mercados de software, no la excepción, y son un arma de doble filo<sup>48</sup>. La duda radica en si la recompensa que obtiene el titular del programa o de la tecnología que se convierte en estándar a consecuencia de un desajuste de esta clase es excesiva, porque esa recompensa procede de una explotación de los *network effects* unido a la no permisividad de interoperabilidad a los demás operadores del mercado, más que del valor real que esa tecnología, ese programa, aporte a la comunidad<sup>49</sup>.

Quizás la cuestión no estribe tanto en la magnitud de la recompensa como en si se llega a obtener igual recompensa en una situación de igualdad de oportunidades con los demás operadores. Posiblemente el elemento que iguale a todos los competidores sea la posibilidad de lograr la interoperabilidad.

Por otra parte, los *network effects* indirectos se ven afectados también por la presencia o ausencia de interoperabilidad.

Un sistema operativo está formado por varios componentes que necesitan funcionar entre ellos, de manera que su diseño estará condicionado por las especificaciones de la interfaz que hacen posible que todos los componentes trabajen en la red.

Si las especificaciones de las interfaces son abiertas, los proveedores de los distintos componentes tendrán más interés en desarrollar componente nuevos que funcionen con los existentes, antes que dedicarse a fabricar un sistema operativo nuevo<sup>50</sup>, de manera que se estimula la innovación de componentes, o “*mix-and-match compatibility*”.

---

<sup>47</sup> Sobre el caso Microsoft, primer caso que trató directamente la cuestión de la interoperabilidad informática en el derecho comunitario, vid. infra., capítulo 4.

<sup>48</sup> C., SAPHIRO, H.R., VARIAN, *Information Rules: A strategic guide to Network Economy*, Boston, Massachusetts, Harvard Business School Press, 1999, pág. 103.

<sup>49</sup> J., FARRELL, M.L., KATZ, “The Effects of Antitrust and Intellectual Property Law on Compatibility and Innovation”, *Antitrust Bulletin* 43, 1998, pág. 609 (612).

<sup>50</sup> C., SAPHIRO, H.R., VARIAN, *Information Rules...*, *op.cit.*, pág. 232.

Si las especificaciones son cerradas, el estímulo se centra en la innovación de sistemas completos. Puede interesar más al fabricante no competir en cada componente e invertir en el desarrollo del sistema operativo y sus componentes<sup>51</sup>.

A modo de recapitulación, ante los efectos red directos, la presencia de interoperabilidad entre programas redundará en beneficio del consumidor, porque su abanico de elección será mucho mayor que en un mercado de software no interoperable. Además, el consumidor maximizará sus beneficios cuanto mayor sea la red de productos interoperables y cuanto más usuarios se añadan a esa red, y finalmente, en presencia de interoperabilidad, el consumidor estará protegido del efecto de encierro o lock-in.

Los *network effects* indirectos tienen mayor repercusión en los operadores del mercado y la innovación. En situaciones donde se produzca la interoperabilidad influyen en el modo de competir en y por el mercado, ya que la lucha por alcanzar una posición de dominio sobre los demás competidores no es lo más importante. En cuanto a la innovación derivada de los *network effects* indirectos, tendrá más presencia en el mercado de productos complementarios si las interfaces son abiertas, y en la creación de sistemas completos, si son cerradas.

### *B.3). Efectos negativos: estandarización susceptible de frenar la innovación.*

A primera vista, estandarización y diversidad competitiva parecen dos términos antagónicos. Sin embargo, en realidad son dos puntos centrales de una misma elipse alrededor de la cual cualquier competidor debería moverse. La eficiencia competitiva requiere un mínimo grado de estandarización por un lado, y por otro lado, demasiados estándares industriales pueden frenar el desarrollo, evitando la existencia de pluralidad y variedad de ofertas competitivas en el mercado.

El principio de la interoperabilidad se presenta como un plan aceptable para conseguir el equilibrio del mercado de la informática<sup>52</sup>.

---

<sup>51</sup> P., SAMUELSON, S., SCOTCHMER, "The Law and Economics...", *op. cit.*, pág. 1623.

<sup>52</sup> M., LEHMANN, "Standardization and the EC Directive of 14 may 1991 on the Legal Protection of Computer Programs", en W.F., KORTHALS (ed.), *Information Law Towards the 21st Century*, Deventer, Boston, Kluwer Law and Taxation, 1992, pág. 368.

Antes de calificar la estandarización como un efecto negativo derivado de la interoperabilidad, conviene hacer una serie de matizaciones:

1°. Los estándares son útiles: Como en el caso de los enchufes, de manera que un aparato eléctrico pueda usar el mismo enchufe en el mismo cajetín, y en el caso del software permitiendo que un mismo programa pueda funcionar en distintos sistemas operativos.

2°. Los estándares son económicos: las empresas pueden reducir su stock de piezas al emplear el mismo tipo de pieza en distintos productos.

3°. Los estándares pueden ser vitales para nuestra seguridad como ciudadanos, por ejemplo, al garantizar que una luz roja en un semáforo significa parar tanto en España como en cualquier país europeo o del mundo.

No es sorprendente que la estandarización o la normalización sea promovida por un amplio número de organizaciones oficiales. Muchos estándares simplemente surgen de la práctica. Es lo que se denomina “estándar de facto”. El producto de un fabricante puede tener éxito y sus características servir de modelo para los venideros. Este reconocimiento del mercado puede ser un halago para el fabricante en cuestión, pero los halagos por sí solos no reportan mucho, más bien ningún beneficio económico. El desarrollo de un producto requiere inversiones que deben recuperarse, y a ser posible, con un margen de beneficios.

A ello ha de sumarse que algunos operadores del mercado están dispuestos a buscar una vía de elusión a cualquier clase de derecho exclusivo que pueda proteger un estándar determinado con el único fin de desarrollar productos propios complementarios del estándar, con independencia de que no siempre resulte ser una solución viable.

Por lo tanto, en aras de promover la innovación es obvio que los estándares deben ser objeto de protección a través de derechos de propiedad industrial e intelectual. Algunos estándares podrán tener un ámbito de protección mayor, y por tanto, posiblemente, el monopolio de explotación de su titular sea mayor que en el caso de otros estándares cuyo ámbito de protección sea menor, en cuyo caso, el desarrollo de productos compatibles por otros no constituya una infracción de sus derechos de

exclusiva. Es más, en tanto que el ámbito de protección de un derecho de propiedad intelectual sobre un estándar es normalmente indeterminado, los efectos red pueden jugar un papel importante para el tribunal a fin de determinar el ámbito de protección en un caso de infracción del derecho <sup>53</sup>.

Por ejemplo, si consideramos el *iPod* de Apple como un estándar, ello no significa que cada componente del producto lo sea, y a efectos de interoperabilidad, sólo interesa la información contenida en una parte del programa, las interfaces. Si esa parte del programa está protegida por el derecho de autor antes de convertirse en estándar, ¿habría que determinar una clase de protección distinta si el producto se convierte en estándar? No es una solución que ofrezca seguridad jurídica en absoluto.

En una situación de interoperabilidad total en el mercado, el consumidor estará obligado a elegir un estándar entre varios estándares. En esencia esto significa que cuando un programa se crea teniendo que hacerlo interoperable con otros, la interoperabilidad se convierte en un condicionante del diseño del programa.

Dicho de otro modo: Hay situaciones en las cuales para que una expresión en particular cumpla un propósito determinado sólo puede enunciarse de un modo determinado. Si se enuncia de otra forma, ya no cumple ese propósito. Las interfaces de software, como veremos más adelante, son un grupo de expresiones funcionales que deben ser duplicadas palabra por palabra, en las que no cabe la opción de expresar la misma idea de un manera diferente. Aquí es donde radica el problema de la estandarización en relación a la interoperabilidad informática. La interoperabilidad es una forma de normalizar y por tanto requiere de una cierta estandarización de la expresión: los programadores deben utilizar una expresión idéntica de la información de la interfaz con el fin de establecer la interoperabilidad<sup>54</sup>.

---

<sup>53</sup> Sobre la influencia de los *network effects* en los derechos de propiedad intelectual e industrial como elemento coadyudante en la determinación del ámbito de protección vid: G., PESSACH, "Copyright Law as a silencing restriction on non infringing materials: unveiling the scope of copyright's diversity externalities", *Southern California L. Rev.*, vol. 76, 2003, pág. 1067; J., MOENIUS, V., TRINDADE, "Networks, standards and Intellectual Property Rights" en K. E., MASKUS (ed.), *Intellectual Property, Growth and Trade*, Amsterdam, Elsevier, 2008, pág. 157.

<sup>54</sup> A.L., CLAPES et al, "Silicon Epics and Binary Bards: Determining the Proper Scope of Copyright Protection for Computer Programs", *UCLA Law Rev.*, 34, 1987, pág 1493(1562); T., DREIER, "Rechtsschutz von Computerprogrammen: Die Richtlinie des Rates der EG von 14. Mai 1991", *CR* 7, 1991, pág. 577 (583); J., FARRELL, "Standardization and Intellectual...", *op.cit.*, pág. 47; P. GOLDSTEIN, "Infringement of Copyright in Computer Programs", *University of Pittsburgh Law*

El efecto negativo se incrementa cuando, debido a las necesidades de interoperabilidad, la funcionalidad del programa se ve comprometida. En tal caso, el software será menos atractivo para el consumidor, con lo cual, en último extremo, podría causar la desaparición del programa del mercado.

## **2. Los programas de ordenador.**

Como se indicó con antelación, en una sociedad en la que “el software está en todas partes”, el programa de ordenador se convierte en una pieza sin la cual el debate sobre la interoperabilidad quedaría vaciado de contenido. Para entender cómo el derecho de autor protege esta clase de obra, es necesario tener un conocimiento técnico del concepto de programa.

### *2.1. Los programas de ordenador desde una perspectiva técnica.*

#### *A). El concepto de programa de ordenador.*

Todo sistema informático requiere de dos elementos para poder funcionar: el *hardware* y el *software*.

El *hardware* es la denominación que se emplea generalmente para designar a todas las partes físicas y tangibles de una computadora: sus componentes eléctricos, electrónicos, electromecánicos y mecánicos; sus cables, gabinetes o cajas, periféricos de todo tipo y cualquier otro elemento físico involucrado. Por tanto, cuando se construye *hardware*, el proceso creativo humano se traduce en una forma física; los bocetos iniciales, los prototipos de prueba, evolucionan hacia un producto físico (chips, tarjetas de circuitos, fuentes de potencia, etc.)

El *programa de ordenador o software*, por el contrario, se refiere al equipamiento lógico o soporte lógico de un ordenador.

---

Review, 47, 1986, pág. 1119 (1127); M. SUCKER, “The Software Directive – Between the Combat against Piracy and the Preservation of Undistorted Competition”, en M. LEHMANN, C. TAPPER (eds.) *A Handbook of Software European Law*, Oxford, Clarendon Press, (vol. 1) 1993, (vol.2) 1995, pág 13; A. VAN ROOIJEN, *The Software Interface...*, *op.cit.*, pág. 61.



Técnicamente el software es, según el «Institute of Electrical and Electronics Engineers»<sup>55</sup> (IEEE), “una combinación de instrucciones computacionales y definiciones de datos que permiten a los elementos físicos del ordenador realizar funciones computacionales o de control”<sup>56</sup>.

Así pues, el concepto de software va más allá del de programa de ordenador entendido como códigos; también su documentación, los datos a procesar y la información de usuario forman parte del software. Abarca todo lo "no físico" intangible y lo tangible relacionado.

### B) Características y clasificaciones.

Contrapuesto al hardware, el software tiene unas características considerablemente distintas:

1º No se fabrica en un sentido clásico, sino que se desarrolla.

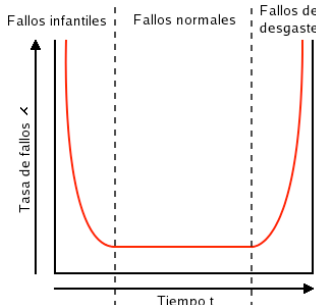
2º El software no se estropea, sino que se deteriora<sup>57</sup>.

---

<sup>55</sup> Institute of Electrical and Electronics Engineers, ( Instituto de Ingenieros Eléctricos y Electrónicos), es la mayor asociación internacional sin ánimo de lucro formada por profesionales de las nuevas tecnologías, como ingenieros eléctricos, ingenieros en electrónica, científicos de la computación, ingenieros en informática, ingenieros en biomédica, ingenieros en telecomunicación e ingenieros en mecatrónica. <http://www.ieee.org/index.html> (Última consulta 1 de junio, 2013).

<sup>56</sup> IEEE Std. 610. 12-1990, en *IEEE Standard Glossary of Software Engineering Terminology*, The Institute of Electrical and Electronics Engineers, New York, IEEE Computer Society, 1990. La definición original en inglés dice: “a combination of computer instructions and data definitions that enable computer hardware to perform computational or control functions”.

<sup>57</sup> Cuando un componente de hardware se estropea, se cambia o se sustituye por otra. Pero no hay piezas de repuesto para el software. Si imagináramos la vida del software en un eje cartesiano, siendo y, el índice de fallos, y x, el tiempo, la imagen que obtendríamos sería la que se muestra en la gráfica:



<http://www.esacademic.com/dic.nsf/eswiki/322029>. (Última consulta 1 de junio, 2013).

Los defectos no detectados harán que falle el programa en las primeras etapas de su vida. Sin embargo una vez que se corrigen (suponiendo que no se introduzcan nuevos errores) la curva se aplana. Por lo tanto se dice que el software no se estropea, no se desgasta, pero sí se deteriora. Durante su vida, el software sufre cambios (mantenimiento). Conforme se hacen los cambios, es probable que se introduzcan nuevos errores, haciendo que la curva tenga picos (como se muestra en la figura). Por cada cambio que se solicite se creará o se originará otro pico, lo cual quiere decir que el software se va deteriorando debido a los cambios. Vid: R., PRESSMAN, *Ingeniería del Software ...*, op.cit., pág. 5.

3º Aunque la industria tiende a ensamblar componentes, la mayoría del software se construye a medida<sup>58</sup>.

La Economía no ha sido tampoco ajena a esta realidad, y por ello, con carácter general desde una perspectiva económica se define al software como “sistema de comandos e instrucciones complejo y re combinado para procesar datos, un bien intangible que sólo puede ser utilizado en unidades diferenciadas”<sup>59</sup>.

Podrían hacerse tantas clasificaciones de software como características nos permita diferenciar a un programa de otro. Pero tal clase de división sería interminable. Como han reconocido distintos autores, debido a la complejidad del software, resulta cada vez más complicado incluir un programa dado en una única categoría predefinida<sup>60</sup>. De tal modo que para comprender los efectos que a nivel jurídico produce la interoperabilidad sobre el software, así como los efectos económicos derivados de la permisividad o prohibición de procedimientos de ingeniería inversa sobre el software, tres son las clasificaciones que hemos de tener en cuenta:

La primera clasificación es la proporcionada por PRIETO ESPINOSA<sup>61</sup>, y atiende a la función que el programa desempeña dentro de un ordenador.

Para este autor el software puede dividirse en tres categorías:

1) *Software de control o sistemas de explotación*: Lo define como “el principal responsable de facilitar a los distintos programas acceso seguro al hardware o en forma más básica, es el encargado de gestionar recursos. Como hay muchos programas y el acceso al hardware es limitado, también se encarga

---

<sup>58</sup> A medida que la industria del software evoluciona se han creado un grupo de componentes estándar, de forma que pueden utilizarse en muchos programas diferentes. La finalidad de disponer de componentes estándar reutilizables radica en que el programador pueda concentrarse en elementos innovadores de un diseño, permitiéndole crear nuevas aplicaciones. Una explicación más detallada de estas características puede verse en: R., PRESSMAN, *Ingeniería...*, op.cit., pág. 5. Otras obras de consulta: R., PRESSMAN, R. HERRON, *Software Shock: The Danger and the Opportunity*, New York, Dorsett House, 1991; M., MINASI, *The Software Conspiracy: Why Companies Put Out Faulty Software, How They Can Hurt You And What You Can Do About It*, McGraw-Hill, 1999, disponible en: <http://www.softwareconspiracy.com/thebook.htm> (Última consulta 1 de junio, 2013).

<sup>59</sup> S., VON ENGELHARDT, “The Economic Properties ...”, op.cit., pág. 3. La definición original en inglés dice: “Software as complex and recombination system of commands and instructions for data processing is an intangible good which can be used only in discrete units”.

<sup>60</sup> A., PRIETO ESPINOSA, A., RUIZ LLORIS, J.C., TORRES CANTERO, *Introducción a la Informática*, Madrid, 3ª ed., McGraw-Hill, 2001, pág. 463; R., PRESSMAN, *Ingeniería del Software...*, op.cit., pág. 6.

<sup>61</sup> A., PRIETO ESPINOSA, *Introducción a la Informática...*, op.cit., pág. 459.

de decidir qué programa podrá hacer uso de un dispositivo de hardware y durante cuánto tiempo”. A este apartado pertenecerían los sistemas operativos, los intérpretes de órdenes<sup>62</sup>, y el software de diagnóstico y mantenimiento<sup>63</sup>.

2) *Software de utilidades*: Hace referencia al conjunto de programas de servicio que, en cierta medida, puede considerarse una ampliación del sistema operativo. Incluye programas para realizar tareas tales como comprensión de datos, respaldo de seguridad, recuperación de archivos erróneamente borrados, antivirus, salvapantallas, etc. También incluyen herramientas generales que facilitan al programador la construcción de aplicaciones para los usuarios, tales como intérpretes, compiladores, editores de texto y cargadores/montadores<sup>64</sup>. Una clase concreta de este tipo de programas con especial relevancia a efectos de la interoperabilidad es el *middleware* o software intermedio. El *middleware* es un software de sistema-distribuido que se encuentra entre las aplicaciones y las plataformas de base (los sistemas operativos, las bases de datos, el hardware), y une las aplicaciones distribuidas, bases de datos o dispositivos. Su función principal es coordinar y posibilitar la comunicación entre las diferentes capas o componentes, a la vez que aísla gran parte de la complejidad del sistema distribuido en un único, probado y comprensible sistema abstracto<sup>65</sup>. En términos más simples, es el “pegamento” entre diferentes elementos necesarios para que el software funcione correctamente en un ordenador electrónico.

---

<sup>62</sup> Parte fundamental de un sistema operativo encargada de ejecutar las órdenes básicas para el manejo del sistema. Suelen incorporar características tales como control de procesos, redirección de entrada/salida y un lenguaje de órdenes para escribir programas por lotes o (scripts). Un ejemplo de esta clase de programa es el Bourne Shell, o el command.com del MS-DOS. Vid.: [www.linux10.com.ar/Glosario](http://www.linux10.com.ar/Glosario) (Última consulta 1 de junio, 2013)

<sup>63</sup> Un programa de diagnóstico es un software que permite monitorizar y en algunos casos controlar la funcionalidad del hardware, como de la computadora, de los servidores y de los periféricos. Este software permite controlar aspectos como la temperatura, el rendimiento o la transferencia de datos. [http://en.wikipedia.org/wiki/Diagnostic\\_program](http://en.wikipedia.org/wiki/Diagnostic_program) (Última consulta 1 de junio, 2013).

<sup>64</sup> Al tratar el proceso de creación o desarrollo de un programa de ordenador se entrará en más detalle en los conceptos aquí mencionados.

<sup>65</sup> Disponible en <http://2011.middleware-conference.org>, (Última consulta 1 de junio, 2013). La definición original en inglés dice: “Middleware is a distributed-system software that resides between applications and underlying platforms (operating systems; databases; hardware), and/or ties together distributed applications, databases or devices. Its primary role is to coordinate and enable communication between different layers or components while isolating much of the complexity of distribution into a single, well tested and well understood system abstraction”. Para un estudio más detallado sobre el middleware ver R., PRESSMAN, *Ingeniería del Software...*, op.cit., pág. 494.

3) *Software de aplicación*: Incluye esta categoría los programas relacionados con aplicaciones específicas, como pueden ser procesadores de texto, bibliotecas de programas para resolver problemas estadísticos o de cálculo numérico, sistemas de bases de datos, programas para gráficas y dibujos, gestores de correo electrónico, visualizadores y navegadores de Internet, videojuegos, etc.

La segunda clasificación a tener en cuenta es la de NORTON, que atiende a la funcionalidad intrínseca del programa. Partiendo de que en una computadora confluyen distintas clases de programas como los antes expuestos, su conjunto forma un sistema operativo, es decir, un conjunto de programas que funcionan de manera unitaria y realiza unas funciones determinadas: despliega en la pantalla los elementos con los que el usuario interactúa; carga los demás programas en la memoria del ordenador para que puedan utilizarse; coordina la forma en que los programas trabajan entre sí y con el hardware del ordenador; y controla la manera en que se almacena y recupera la información de los discos<sup>66</sup>.

Según el número de usuarios y el número de tareas que cada sistema operativo podemos distinguir:

1) *Sistemas operativos (SO) de una sola tarea*. Son aquellos que sólo permiten al ordenador realizar una única tarea o proceso cada vez. El ejemplo más claro de un sistema operativo de esta clase es el MS-DOS, o el sistema Palm que se utiliza en las PDA<sup>67</sup>. Como dice el autor, “a pesar de ser sistemas operativos limitados, existe un uso específico para ellos, porque ocupan muy poco espacio en el disco o en la memoria cuando están ejecutándose y no requieren una computadora poderosa y costosa”<sup>68</sup>.

2) *Sistemas operativos multitarea o plataformas*. Son los que permiten realizar varias tareas o procesos simultáneos. Algunos ejemplos serían, Windows de Microsoft, Mac, Leopard o Lion de Apple, y GNU/Linux o Debian GNU/Linux de GNU Project.

---

<sup>66</sup> P., NORTON, *Introducción a la ...*, *op.cit.*, pág. 265.

<sup>67</sup> Más información en <http://www.palm.com/es> (Última consulta 1 de junio, 2013).

<sup>68</sup> P., NORTON, *Introducción a la ...*, *op.cit.*, pág. 266.

La tercera clasificación que habrá que tener en cuenta, no es de carácter técnico sino económico, y la propone KEMPER<sup>69</sup>.

Según la magnitud de los efectos económicos que tienen lugar en relación a un tipo u otro de programas, KEMPER divide el software en:

1) *Software de gestión y estrategia para empresas*. Esta clase de software sirve para integrar en un solo sistema todos los datos y procesos informáticos que tengan lugar en una gran empresa. Por ejemplo, dos de los programas de esta clase con más presencia en el mercado son EPR (Enterprise Resource Planing) y SAP (Systems, Applications, Products in Data Processing).

2) *Software de intercambio electrónico de datos*. El intercambio electrónico de datos consiste en la transferencia estructurada de datos entre organizaciones por medios electrónicos. La idea que fundamenta esta clase de programas es la transmisión estructurada de datos basada en unos estándares acordados, de un sistema computacional a otro sin la intervención del ser humano<sup>70</sup>. Se usa no sólo para transferir documentos o datos electrónicos de un sistema computacional a otro, sino también para implementar dichos procesos. Debido a innovaciones tecnológicas como los servicios de web XML, o Internet y la World Wide Web, esta clase de software tiene un papel fundamental en el comercio electrónico así como en la administración electrónica, agilizando las relaciones con agentes externos.

3) *Software para suites informáticas*. Esta clase de programas es un conjunto de programas de aplicación que permiten realizar distintas tareas como crear, modificar, organizar, escanear, imprimir, archivos y documentos. Normalmente esta clase de software se distribuye en un solo paquete, suele tener una interfaz de usuario sistemática, que permite que las aplicaciones que lo forman puedan interoperar entre ellas. Microsoft Office o Open Office son dos ejemplos de esta clase de software.

---

<sup>69</sup> A., KEMPER, *Valuation of Network Effects... op.cit.*, pág. 80-81.

<sup>70</sup> *Ibidem*, pág. 81.

4) *Software para servicios de redes sociales*. Su objetivo principal es la configuración y el mantenimiento de las redes sociales. Se basan en las soluciones web o software en línea y proporcionan un abanico de canales de interacción a los usuarios de la red (chats, mensajería instantánea, compartir archivos, grupos de discusión). Ejemplos de esta clase de plataformas de software son Facebook, Twitter o LinkedIn.

### C) *La creación de un programa de ordenador: de la programación a la ingeniería informática.*

Algunos autores, entre los que destaca KNUTH, consideran al programa de ordenador como una obra de arte porque cree que es una experiencia estética semejante a escribir un poema o componer música<sup>71</sup>. Para otros como DENNING sin embargo, el programa de ordenador es una obra científica, entendida como el resultado del estudio sistemático de procedimientos algorítmicos que describen y transforman información<sup>72</sup>.

A pesar de que los debates sobre la naturaleza del programa de ordenador continúan, la cada vez mayor complejidad y sofisticación del software ha culminado en la calificación de la Informática (*Computing* en inglés) como ciencia. La prueba más evidente la encontramos en la Ingeniería Informática (también denominada Ingeniería del Software), cuyo objeto es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadora y la documentación asociada requerida para desarrollar, operar (funcionar) y mantenerlos<sup>73</sup>.

A ello hay que añadir que el proceso de creación de un programa lo acerca más a una obra científica que a una obra literaria. El presupuesto básico del que parte la concepción de un programa de ordenador comienza con el planteamiento del problema que éste ha de resolver. Cuando se ha adquirido una idea clara y precisa del problema, el programador debe escoger una representación para los diferentes datos que intervienen en él, y a continuación, se puede pensar en redactar un algoritmo que resuelva aquel

---

<sup>71</sup> D.E., KNUTH, *The Art of Computing Programming*, Barcelona, 1ª ed., Reverté, 1980, prólogo.

<sup>72</sup> P.J., DENNING, "Is Computer Science Science?", *Communications of the ACM*, vol. 48, n.4, Abril 2005, pág. 27 (31).

<sup>73</sup> B.W., BOEHM, "Software engineering", *IEEE Transaction on Computers*, C-25, n.12, Diciembre, 1976, pág. 1226 (1230).

problema. Esta sería una forma sencilla de definir lo que los ingenieros informáticos denominan “proceso de desarrollo de sistemas de software”<sup>74</sup>.

En términos generales, el proceso de creación de un programa puede dividirse en tres grandes etapas: la definición, el desarrollo y el mantenimiento<sup>75</sup>.

La primera es denominada “fase de definición”, y consiste en el planteamiento del problema. La meta de esta fase es responder a la preguntas: ¿Cuál es el problema a resolver?, y ¿cuáles son las características de la entidad que se utiliza para resolver ese problema? Es en esta fase donde han de identificarse los requisitos clave del sistema y del software. Como indica NORTON, “una vez que el programador ha determinado el punto de inicio y el punto final del programa, puede comenzar a diseñar el código”<sup>76</sup>.

La segunda fase, “el desarrollo”, se centraría en el cómo. Durante esta etapa el programador intenta definir cómo ha de diseñarse el programa (las estructuras de datos, las interfaces, los procedimientos, ...) y cómo ha de traducirse el diseño en un lenguaje comprensible por el ser humano – el código fuente – y en un lenguaje comprensible por la máquina – código objeto -, para luego realizarse la prueba del software y la corrección de errores.

La tercera fase, “mantenimiento” comienza desde el momento en que el programa está elaborado y se ejecuta en un sistema. Como indica PRIETO ESPINOSA, coincide con la vida útil del programa. En esta fase, que es la más larga del ciclo de vida del programa, habrá que ir adaptando el programa a las necesidades del usuario y a los avances que se produzcan en el entorno del software. En este sentido PRESSMAN enumera cuatro tipos de cambios que normalmente suceden en esta etapa: *Cambios correctivos*, en relación a los errores menores que no se hayan reparado durante la fase anterior y que sólo puedan detectarse a través del uso del programa. *Adaptaciones*, entendiéndose por este clase de cambios los avances que se produzcan en el entorno original para el cual se “definió” el software. *Mejoras*: Para PRESSMAN, según se emplee el software, el usuario puede descubrir funciones adicionales que van a

---

<sup>74</sup> Vid. R., PRESSMAN, *Ingeniería del Software...*, *op.cit.*, pág 16; P., NORTON, *Introducción...*, *op.cit.*, pág. 525; A., PRIETO ESPINOSA, *Introducción...*, *op.cit.*, pág. 573.

<sup>75</sup> R., PRESSMAN, *Ingeniería del Software...*, *op.cit.*, pág. 13. Esta autor inicialmente habla de estas tres grandes etapas, si bien, a continuación, subdivide cada una de ellas, estableciendo el estándar del llamado “ciclo de vida del software” que consta de cinco fases: análisis de las necesidades, diseño del programa, desarrollo, implementación y mantenimiento.

<sup>76</sup> P., NORTON, *Introducción...*, *op.cit.*, pág. 525.

producirle beneficios. En cuarto lugar, los *cambios preventivos* o *reingeniería*. Como ya dijimos, el software puede volverse inestable o quedar obsoleto tanto por el paso del tiempo como por el avance de la tecnología. Para evitarlo, los programas pueden ser sometidos a procesos de reingeniería o reconstrucción.

La reingeniería del software ha sido definida como “la modificación de un producto software, o de ciertos componentes, usando para el análisis del sistema existente técnicas de ingeniería inversa y, para la etapa de reconstrucción, herramientas de ingeniería directa, de tal manera que se oriente este cambio hacia mayores niveles de facilidad en cuanto a mantenimiento, reutilización, comprensión o evaluación”<sup>77</sup>. En resumen, se trata de crear una versión de un programa existente superior en calidad al original e inferior en costes de mantenimiento.

Si comparamos estas tres etapas descritas con las fases del proceso creativo de WALLAS<sup>78</sup>, veremos que existen bastantes similitudes. Para este filósofo inglés, los enfoques creativos e iluminaciones eran explicados por un proceso de cinco etapas: **Preparación** (planteamiento de un problema donde se centra la mente y se exploran sus dimensiones); **incubación** (el problema es interiorizado en el hemisferio derecho y parece que nada pasa externamente); **intimación** (la persona creativa "presiente" que una solución está próxima); **iluminación** o *insight* (la idea creativa salta del inconsciente al consciente), y **verificación** (la idea es conscientemente verificada, elaborada y luego aplicada).

La elaboración del software es un proceso creativo que no es en absoluto sencillo y además requiere grandes inversiones de tiempo y dinero. Por ello, uno de los objetivos de la Ingeniería Informática desde hace algunas décadas se centra en encontrar procesos y metodologías de desarrollo de software que sean sistemáticas, predecibles y repetibles, a fin de mejorar la productividad en el desarrollo y la calidad del software<sup>79</sup>.

Asimismo, en todo este proceso creativo, suelen participar muchas personas, ya que es difícil que una única persona conozca todos los detalles de una aplicación, y el volumen del proyecto (decenas o cientos de miles de líneas de código) suele ser cada

---

<sup>77</sup> M.A., SILICIA, V., DE LA MORENA, “Qué es reingeniería de software”, artículo en línea, 2009, disponible en: <http://cnx.org/content/m17438/latest>, (última consulta 1 de junio, 2013).

<sup>78</sup> G., WALLAS, *The Art of Thought*, New York, Harcourt, Brace and Company, 1926.

<sup>79</sup> R., PRESSMAN, *Ingeniería del Software...*, *op.cit.*, pág. 13-15.



vez más grande en la práctica. Desde una perspectiva comercial y competitiva, si el proceso de desarrollo es débil, el producto final va a sufrir sin lugar a dudas. Dicho de otro modo, a la hora de elaborar un software se produce una dualidad producto-proceso que conlleva una actividad creativa y cuyo núcleo central son los códigos<sup>80</sup>.

Hablar de códigos nos lleva a hablar de lenguajes de programación. Ambos son conceptos clave para comprender no sólo cómo se elabora un programa sino también en qué consiste la ingeniería inversa del software.

En primer lugar, un lenguaje de programación es un conjunto de símbolos y de reglas que combinados se usan para expresar algoritmos. Los lenguajes de programación, al igual que los lenguajes que usamos para comunicarnos, poseen un léxico, una sintaxis y una semántica<sup>81</sup>.

Cuando el programador ha definido el problema que pretende resolver, el resultado al que llegará será un algoritmo. Este algoritmo se redacta en un lenguaje de programación específico, comprensible por el ser humano pero no por la máquina. Es lo que se denomina “lenguaje de alto nivel” y en este lenguaje se escribe el código fuente del programa.

Los lenguajes de alto nivel tienen una sintaxis más parecida a los idiomas humanos que el código de lenguaje máquina. Hoy día, el principal problema que presentan los lenguajes de alto nivel es su gran variedad: FORTRAN, LISP, ALGOL, COBOL, APL, SNOBOL, PROLOG, MODULA2, ALGOL68, PASCAL, SIMULA67,

---

<sup>80</sup> M. DAVIS comenta a propósito de esta dualidad producto y proceso: cada diez años o cinco aproximadamente, la comunidad del software vuelve a definir «el problema» cambiando el foco de los aspectos de producto a los aspectos de proceso. Por consiguiente, se han abarcado lenguajes de programación estructurados (producto) seguidos por métodos de análisis estructurados (proceso) seguidos a su vez por encapsulación de datos (producto) y después por el énfasis actual en el Modelo de Madurez de Capacidad de Desarrollo del software del Instituto de ingeniería de software (proceso). En la comunidad científica hay un precedente que se adelanta a las nociones de dualidad cuando contradicciones en observaciones no se pueden explicar completamente con una teoría competitiva u otra. (...). Creo que las observaciones que se hacen sobre los mecanismos del software y su desarrollo demuestran una dualidad fundamental entre producto y proceso. Nunca se puede comprender el mecanismo completo, su contexto, su uso, significado y valor si se observa sólo como un proceso o sólo como un producto. Toda actividad humana puede ser un proceso, pero cada uno de nosotros obtiene el sentido de autoestima ante esas actividades que producen una representación o ejemplo que más de una persona puede utilizar o apreciar, una u otra vez, o en algún otro contexto no tenido en cuenta. Es decir, los sentimientos de satisfacción se obtienen por volver a utilizar nuestros productos por nosotros mismos o por nosotros. En: M., DAVIS, “Process and Product: Dichotomy or Duality”, Software Engineering Notes, ACM Press, vol. 20, nº 2, Abril 1995, pág. 17 (18).

<sup>81</sup> A., PRIETO ESPINOSA, *Introducción...*, *op.cit.*, pág. 519.

ADA, C++, LIS, EUCLID, BASIC, PYTHON, etc., además de las diferentes versiones o dialectos que se han desarrollado de algunos de ellos.

Como ha indicado NORTON<sup>82</sup>, con esta clase de lenguaje de programación, se pretenden alcanzar una serie de objetivos cuyo fin común es normalizar y simplificar, de manera que el proceso de libre creación (o dicho en términos de derechos de autor, la originalidad) se ve limitado de cara a:

1. Lograr la interoperabilidad del programa, pudiendo utilizar un mismo programa en diferentes equipos con la única condición de disponer de un programa traductor o compilador, para obtener el programa ejecutable en lenguaje binario de la máquina que se trate.
2. Aproximarse al lenguaje humano para evitar la comisión de errores que sí se dan en lenguaje máquina. Cuanto más fácil se pueda escribir y leer el programa, utilizando palabras (normalmente en lengua inglesa) en lugar de cadenas de símbolos sin ningún significado aparente, menor el número de errores y por tanto, menor el número de correcciones o “debugging” que necesitará el proceso.
3. Incluir rutinas de uso frecuente (estándares) como son las de entrada/salida, funciones matemáticas, manejo de tablas, etc., que figuran en una especie de librería del lenguaje, de tal manera que se puedan utilizar siempre que se quieran sin necesidad de tener que programarlas cada vez que se accede al código fuente del programa para realizar tareas de mantenimiento, nuevos desarrollos, correcciones de errores, etc.

En cualquier caso, para que la máquina entienda las instrucciones del programador debe usarse un lenguaje específico conocido como “lenguaje máquina”. Lenguaje que el ordenador electrónico comprende fácilmente y que resulta ilegible para las personas. De hecho sólo consiste en cadenas extensas de números 0 y 1. Para acelerar el trabajo y facilitar la necesidad de recordar secuencias de programación para acciones habituales, los primeros programadores comenzaron a denominarlas con nombres fáciles de memorizar y asociar: ADD (sumar), SUB (restar), MUL (multiplicar), etc. A esta secuencia de posiciones se las denominó “instrucciones”, y a

---

<sup>82</sup> P., NORTON, *Introducción...*, op.cit., pág. 516.

este conjunto de instrucciones que traducían los 0 y 1 por palabras, “lenguaje ensamblador”. Prácticamente la única ocasión en que los programadores escriben programas en un lenguaje ensamblador es cuando les preocupa que el código sea eficiente y rápido (un ejemplo son los videojuegos de acción, donde la velocidad del programa es muy importante). De otra forma, los programadores utilizan esta clase de lenguaje para afinar partes importantes de programas que están escritos en un lenguaje de nivel más alto.<sup>83</sup>

Es en la fase de desarrollo donde se va llevar a cabo la traducción de los códigos, denominada compilación. Compilar un programa de ordenador significa por tanto, convertir un lenguaje de alto nivel - el código fuente - en código objeto, para que pueda ser ejecutado por la computadora. Cuanto más complejo sea el código fuente que debe ser traducido, más delicada será la labor de compilación del mismo. Por ello, para efectuar esta traducción se emplea un compilador, que no deja de ser otro programa de ordenador. Pero como cada lenguaje de programación tiene unas reglas especiales (sintaxis) debe existir un compilador específico para cada lenguaje de programación<sup>84</sup>.

Si el código fuente es sintácticamente correcto, el compilador generará un código objeto, constituyendo un programa que todavía no será ejecutable por el ordenador. En caso contrario, no se generará el código objeto y se mostrará una lista de errores que habrán de depurar, utilizando para ello otra aplicación informática denominada en jerga informática “debugger”. Conseguido el código objeto, la fase de compilación aún no ha terminado, sino que se procede a lo que denominan fase de *linkado* o enlazado. En esta etapa, se enlaza el código objeto generado en todos los ficheros y subprogramas que se han compilado y se añade a éste el código de las funciones que hay en las bibliotecas del compilador, dando como resultado un programa que puede ser ejecutado por el ordenador. Llegado a este punto, se realizaría la prueba del software, y si funciona, el siguiente paso es situarlo en el mercado.

Todo este proceso no es en absoluto sencillo ni rápido y además su coste es elevado. Por ello, los proveedores de software ponen un especial énfasis en proteger sus códigos fuentes y sus códigos objetos. Ahora bien, hacer el trabajo inverso, convertir el

---

<sup>83</sup> *Ibidem*, pág. 518.

<sup>84</sup> A., PRIETO ESPINOSA, *Introducción...*, *op.cit.*, pág. 528.

código objeto en un lenguaje de alto nivel como es el código fuente, es tanto o más complicado que el propio proceso de compilación.

## *2.2. Los programas de ordenador desde la perspectiva jurídica*

### *A) Marco normativo internacional.*

El primer instrumento normativo supranacional que declaró la protección de los programas de ordenador como obras literarias en el marco del Convenio de Berna, y por tanto objeto de protección por el derecho de autor, fue la Directiva 91/250/CEE de 14 de mayo de 1991 sobre protección jurídica de los programas de ordenador<sup>85</sup>. Quizás esta decisión en el ámbito comunitario condicionó que poco después esta postura se trasladase al ámbito internacional, con el consiguiente efecto armonizador.

A nivel internacional la configuración de la protección de los programas de ordenador como obras literarias en el marco del Convenio de Berna<sup>86</sup> se fundamenta en los siguientes Tratados:

- 1.El primer lugar corresponde al Acuerdo de 1994 sobre los aspectos de los derechos de la propiedad intelectual relacionados con el comercio<sup>87</sup>, (Acuerdo sobre los ADPIC o, por sus siglas en inglés, TRIPs) que en su artículo 10 declara que los programas de ordenador, sean programas fuente o programas objeto, serán protegidos como obras literarias en virtud del Convenio de Berna<sup>88</sup>.

---

<sup>85</sup> Artículo 1.1 de la Directiva Software.

<sup>86</sup> Convenio de Berna para la protección de las Obras Literarias y Artísticas del 9 de septiembre de 1886, completado en París el 4 de mayo de 1896, revisado en Berlín el 13 de noviembre de 1908, completado en Berna el 20 de marzo de 1914 y revisado en Roma el 2 de junio de 1928, en Bruselas el 26 de junio de 1948, en Estocolmo el 14 de julio de 1967, en París el 24 de julio de 1971 y enmendado el 28 de septiembre de 1979; BOE núm. 235, de 1 de octubre 1986, pág. 33547.

<sup>87</sup> Acuerdo de 1994 sobre los aspectos de los derechos de la propiedad intelectual relacionados con el comercio. Anexo 1 C del Acuerdo por el que se establece la Organización Mundial del Comercio, BOE n. 20, supl. de 24 de Enero 1995; BOE núm. 33, de 8 de febrero 1995, pág. 192, (en adelante Acuerdo sobre los ADPIC)

<sup>88</sup> El texto original del artículo en inglés dice "computer programs, whether in source or object code, shall be protected as literary works under the Berne Convention", y su traducción al español no parece demasiado afortunada, porque como se verá, código de un programa y programa, no son lo mismo.

2.El segundo instrumento internacional es el Tratado de la OMPI sobre Derecho de Autor<sup>89</sup>, cuyo artículo 4, expresamente declara que "*los programas de ordenador están protegidos como obras literarias en el marco de lo dispuesto en el artículo 2 del Convenio de Berna. Dicha protección se aplica a los programas de ordenador, cualquiera que sea su modo o forma de expresión*".

La trascendencia de este Tratado en las legislaciones de los Estados miembros, en especial en el derecho norteamericano, merece ahondar un poco más en cómo se llegó a la redacción del mencionado artículo 4.

En los documentos preparatorios elaborados por la Oficina Internacional de la OMPI para los comités de expertos, se proponía un articulado a través del cual los países parte del Tratado estarían obligados a conceder una protección por Derecho de autor a los programas de ordenador. Dicha protección debía ser la misma que la prevista en el Convenio de Berna para las obras literarias, y el concepto de programa de ordenador abarcaría tanto los programas consistentes en sistemas operativos como los programas de aplicación, y tanto en forma de código fuente como código objeto. Además, el Memorándum para la primera sesión del Comité proponía que el mismo texto normativo aclarase que no se permitía la reproducción de un programa para fines privados, dejando a la Ley nacional la posibilidad de autorizar ciertas limitaciones sobre el derecho de exclusiva, como la realización de una copia o la adaptación que fuese necesaria para utilizar el programa en un dispositivo físico, para cuya finalidad se hubiese adquirido de forma legítima; o para fines de seguridad o salvaguarda; o para su descompilación, sometida a un conjunto de condiciones similares a las previstas en el artículo 6 de la Directiva Europea del 14 de mayo sobre la protección jurídica de los programas de ordenador.

La discusión fue complicada porque, de una lado, muchas delegaciones opinaban que los programas de ordenador ya eran objetos protegidos por el Derecho de autor de acuerdo a la normativa vigente en los ordenamientos nacionales y esta opción también había sido respaldada por la jurisprudencia de diversos países. A ello añadieron que, tal como estaba redactada la propuesta, podría interpretarse que esta clase de

---

<sup>89</sup> Tratado de la OMPI sobre Derecho de Autor ..., *op.cit.*

protección solamente se concedía a los programas creados con posterioridad a la aprobación del Tratado y no a los anteriores, o que igualmente cabría interpretar el texto de la propuesta como que la obligación de tutelar los programas de ordenador a través del derecho de autor sólo comprometería a los países miembro del Convenio de Berna que se adhirieran al nuevo instrumento normativo. El tercer argumento en contra de la propuesta era que las limitaciones al derecho exclusivo del titular debían dejarse en manos de los ordenamientos jurídicos nacionales, cuyos legisladores tendrían que aplicar el principio de los “usos honrados” contemplado en el artículo 9.2 del Convenio de Berna<sup>90</sup>.

Al final, el nuevo Tratado se conformó con unas expectativas menores. El texto del artículo 4 es bastante similar en su redacción y contenido al artículo 10.1 del Acuerdo sobre los ADPIC (o TRIPs). La única salvedad destacable es que fue aprobada una declaración concertada según la cual el artículo 4 del Tratado debe ser leído en conjunto con el artículo 2 del mismo cuerpo legal. En segundo lugar, se declara que el artículo 4 del Tratado está en conformidad con el artículo 2 del Convenio de Berna y con el artículo 10 del Acuerdo sobre los ADPIC<sup>91</sup>.

*B) El programa de ordenador como obra de derecho de autor asimilada a una obra literaria.*

La historia de los programas de ordenador como obras protegida por el derecho de autor comienza en el mismo lugar que el fenómeno tecnológico de la informática nació y maduró: en Estados Unidos. Fue aquí donde la demanda de una protección exclusiva para el programa de ordenador tuvo su origen, coincidiendo con el desarrollo gradual y la comercialización de los programas como productos independientes del

---

<sup>90</sup> Vid. *Memorandum de la Oficina Internacional de la OMPI a la primera sesión del Comité de Expertos sobre un eventual Protocolo relativo al Convenio de Berna*, Derecho de Autor, año IV, núm. 2, OMPI, Ginebra, 1992, pág. 24 y 29.

<sup>91</sup> La Declaración concertada relativas al Tratado de la OMPI sobre Derecho de Autor, adoptadas por la Conferencia Diplomática el 20 de diciembre de 1996, respecto del artículo 4 dispone: “*El ámbito de la protección de los programas de ordenador en virtud del Artículo 4 del presente Tratado, leído junto con el Artículo 2, está en conformidad con el Artículo 2 del Convenio de Berna y a la par con las disposiciones pertinentes del Acuerdo sobre los ADPIC*”.

El artículo 2 del Tratado WCT establece el ámbito de protección del derecho de exclusiva, y dispone: “*La protección del derecho de autor abarcará las expresiones pero no las ideas, procedimientos, métodos de operación o conceptos matemáticos en sí*”.

hardware o computadora<sup>92</sup>. En consecuencia, el proceso progresivo de integración vertical en el mercado de los ordenadores personales<sup>93</sup> y la aparición simultánea de un mercado independiente para los programas, diferenciado del mercado de componentes físicos (hardware), provocó una doble necesidad. En primer lugar, la necesidad de contar con una defensa más fuerte contra los competidores; los secretos industriales en combinación con la protección contractual ofrecían una protección adecuada entre las partes del contrato, pero no garantizaban la protección contra posibles prácticas parasitarias de terceros. En segundo lugar, era necesario incentivar la creación de programas en un sector del mercado emergente. Estas necesidades llevaron a los operadores del mercado a poner en el punto de mira el derecho de autor como paradigma de protección capaz de proporcionar una protección exclusiva erga omnes, pero que a diferencia de las patentes, tenía unos costes de acceso mucho menores ya que los requisitos de protección eran más laxos.

El enfoque pro-derecho de autor fue impulsado por el curioso acogimiento del mismo por los tribunales estadounidenses y por la Oficina de Marcas y Patentes (USPTO)<sup>94</sup>. El último paso lo dio el legislador, estimulado por el Director General de IBM y respaldado por un informe de la Comisión Nacional titulado “Nuevos Usos Tecnológicos sobre obras protegidas por derecho de autor” (CONTU)<sup>95</sup>, que culminó en la modificación de la Copyright Act en 1980. En dicha modificación fue introducida la mención explícita de los programas de ordenador entre la lista de obras objeto de protección por el derecho de autor<sup>96</sup>.

---

<sup>92</sup> Nos referimos al desarrollo del mercado de “paquetes de software” y la entrada gradual de vendedores independientes de software, que comenzó a finales de la década de 1960 y tuvo su mayor crecimiento entre el final de 1970 hasta la mitad de la década de 1990, gracias a la propagación de los ordenadores personales.

<sup>93</sup> Vid. T., BRESNAHAN, “New Modes of Competition: Implication for the Future Structure of the Computer Industry”, 1998, disponible en: <http://siepr.stanford.edu/publicationsprofile/1885> (Última consulta 1 de junio, 2013).

<sup>94</sup> Las Directrices de 1968 de la USPTO (33 Fed. Reg. 15.609, 15.610) declaraban que: “a computer programming process which produces no more than a numerical, statistical or other informational result is not directed to patentable subject matter”.

<sup>95</sup> El Final Report of the National Commission on New Technological Uses of Copyrighted Works es un informe que el Congreso encargó a esta comisión especial a finales de 1974 sobre los programas de ordenador y las fotocopias. Vid. infra., capítulo 2.1

<sup>96</sup> Siguiendo la recomendación del Informe CONTU, el Congreso de los Estados Unidos modificó el artículo 101 de la Copyright Act, e incluyó la definición de “computer program” como “a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result”.

En un sector global como el de las TICs, el enfoque adoptado por los Estados Unidos se convirtió también en global de una forma natural, y en parte también ayudado por la presión diplomática internacional. Aunque el intento de incluir de forma expresa el programa de ordenador como obra protegida en el Convenio de Berna falló, la Unión Europea siguió el camino iniciado por Estados Unidos, plasmado en la Directiva 91/250/CEE de 14 de mayo de 1991 sobre protección jurídica de los programas de ordenador, junto con la introducción de la ficción legal de considerar al programa de ordenador como obra literaria.

Encajar el programa de ordenador como obra literaria en el seno de los derechos de autor, donde la obra es expresión de la personalidad de su autor<sup>97</sup>, es más que un reto, es imposible. Cualquier obra literaria tradicional puede contener en mayor o menor medida (desde una novela hasta un mapa cartográfico) su aportación personal. Sin embargo, un programa de ordenador no pretende ser el reflejo de la personalidad de su autor, sino resolver un problema. La forma de expresión de un programa siempre será funcional y no tiene ninguna relevancia las variaciones que puedan realizarse de esa forma de expresión, porque las mismas siempre se corresponderán con conceptos funcionales y no con ninguna clase de “libertad estética” o incluso capricho del autor<sup>98</sup>.

A ello se añade que, si bien los derechos de autor otorgan una serie de facultades exclusivas al titular de los derechos sobre la obra, de igual modo permiten a terceros estudiar, analizar o examinar libremente dicha obra, contribuyendo así al aumento del acervo cultural y al progreso de la humanidad. Pero esto no sucede con los programas de ordenador<sup>99</sup>.

La ficción de tratar a los programas de ordenador como obras literarias sigue suscitando dificultades tanto a la doctrina como a los tribunales. Como han indicado HILTY y GEIGER, aunque inicialmente los programas fueron, de una manera un tanto arbitraria, comparados a obras literarias para poder incluirlos dentro del ámbito de

---

<sup>97</sup> La denominada concepción naturalista que Hegel, y posteriormente von Gierke y Köhler, tomaron prestada de Kant. Vid. G., DAVIES, *Copyright and the Public Interest*, IIC Studies, vol. 14, Weinheim, New York, VCH, 1994, pág. 10.

<sup>98</sup> G., GHIDINI, *Innovation, competition and consumer welfare in international property law*, Cheltenham, Edward Elgar Publishing, 2010, pág. 133; J.A., GÓMEZ SEGADÉ, “La protección jurídica de los programas de los ordenadores electrónicos”, RDM, n° 113, Julio-Septiembre, 1969, pág. 413 (428).

<sup>99</sup> G.R., IGNATIN, “Let the hackers hack: Allowing the Reverse Engineering of Copyrighted Computer Programs to Achieve Interoperability”, UPLR 140(5), 1992, pág. 1999 (2009).



protección del derecho de autor, sin embargo, aparte del hecho de la difícil comparación entre una novela y una serie de dígitos o números, el problema se sitúa en que los requisitos para considerar una obra protegida por derecho de autor no parecen muy apropiados para otorgar dicha protección a la mayoría de los programas de ordenador<sup>100</sup>.

El programa, el software, es intrínsecamente tecnología: es el paradigma de las tecnologías de la información. La forma del programa es siempre funcional y no tienen importancia las variaciones de su forma, ya que estas posibles variaciones suelen deberse a razones de utilidad y no a “libertad estética” ni mucho menos al capricho del programador. Se produce una fusión entre la expresión y la función<sup>101</sup>.

Desde la perspectiva tradicional del derecho de autor, para cualquier obra literaria, como un artículo de opinión, un ensayo o un poema, este derecho exclusivo sobre la obra representa un acuerdo entre el autor y el público: el autor tiene una serie de facultades exclusivas sobre la obra, representadas por el *ius prohibendi*, y el público tiene derecho a acceder a la obra completa a cambio de una compensación. Ahora bien, una vez se ha accedido a la obra de manera legítima, identificar uno por uno, cuáles son los elementos protegidos y no protegidos de ella a través de su lectura resulta imposible. El acceso es completo.

En un programa de ordenador, al que el público normalmente tiene acceso solo en un lenguaje incomprensible para un ser humano (código objeto), este acuerdo es unilateral: el autor además de las facultades patrimoniales y morales, tiene derecho prohibir el acceso del público a elementos que se encuentran fuera del ámbito de protección de este derecho, es decir, a las ideas en que se inspira el código del programa<sup>102</sup>.

Los programas de ordenador poseen unas características que los convierten en una clase de obras únicas en el sistema del derecho de autor (tanto en el *copyright* como

---

<sup>100</sup> R., HILTY, C., GEIGER, “Towards a new instrument of protection for software in the EU? Learning the lessons from the harmonization failure of software patentability”, MPI for Intellectual Property and Competition Law Research Paper No. 11-01, 2011, pág. 3, disponible en: <http://ssrn.com/abstract=1746846> (Última consulta 1 de junio, 2013).

<sup>101</sup> G., GHIDINI, *Innovation, ...*, *op.cit.*, pág. 133.

<sup>102</sup> P., SAMUELSON, R., RANDALL, D., MITCHELL et al., “A Manifesto concerning the Legal Protection of Computer Programs”, *Columbia Law Review* 94(8), 1994, pág. 2308 (2430). En igual sentido la sentencia del caso *Sega Enterprises Ltd. v. Accolade, Inc.*, 977 F.2d (9th Cir. 1992), pág. 1510 (1520), al referirse a “the unique nature of computer object code” as reason for considering reverse engineering cases on a case-by-case basis under section 107 of the 1976 Copyright Act.

en el *droit de auteur* continental). Estas características son, en primer lugar, la funcionalidad, y en segundo, la ausencia de finalidad comunicativa. Son además estos rasgos, motivo de fricción con alguno de los requisitos que una obra ha de cumplir para considerarlos objeto protegible, como son la originalidad y la expresión de la obra.

En todo caso al analizar el programa de ordenador desde la perspectiva jurídica, como ha apuntado la Prof. LEWINSKI, no debe perderse de vista que los sistemas de protección en general son normalmente la expresión de las industrias y las culturas para las que son articulados más que una herramienta neutral que regula una situación jurídica<sup>103</sup>. Ejemplo de ello es que desde la entrada de Estados Unidos en el Convenio de Berna en el año 1989, este país, donde reside la mayor parte de la industria informática, no ha cesado en la lucha por el reconocimiento de elementos exclusivos del sistema del copyright a nivel internacional, lo que ha acentuado una especie de competición entre el sistema del *droit de auteur* y el del *copyright*.

### C). *Las características especiales de los programas de ordenador.*

#### a) *La funcionalidad.*

La primera dificultad que surge al comparar un programa de ordenador con una obra literaria radica en que el software es una obra funcional. Los programas de ordenador son tecnología intrínseca, es decir, son el paradigma de las tecnologías de la información<sup>104</sup>. Son “máquinas cuyo medio de construcción resulta ser texto<sup>105</sup>”.

La funcionalidad de un programa es todo lo que sucede desde su ejecución en el sistema de destino, normalmente, un ordenador<sup>106</sup>. Ahora bien, los aspectos funcionales de las obras literarias, no están incluidos dentro del ámbito de protección de los derechos de autor: Si en una misma obra confluyen ideas y expresiones susceptibles de

---

<sup>103</sup> S. VON, LEWINSKI, *International Copyright Law and Policy*, Oxford, New York, Oxford University Press, 2008, pág. 33-34. Incluso algún autor como CORNISH, ha apuntado que la opción de proteger los programas de ordenador como obras literarias en el Convenio de Berna, fue una de las razones por las que Estados Unidos decidió suscribir el Convenio en 1989. Vid., W.R., CORNISH, “Computer Programs Copyright and the Berne Convention” en M., LEHMMAN, C. TAPPER (eds.), *A Handbook of...*, *op.cit.*, pág. 185.

<sup>104</sup> G., GHIDINI, *Innovation, ...*, *op.cit.*, pág 133.

<sup>105</sup> P., SAMUELSON, *et al*, “A Manifesto ...” *op.cit.*, pág. 2341.

<sup>106</sup> D., CARLETON, “A Behaviour-Based Model for Determining Software Copyright Infringement”, *BTLJ* 10, nº 2, 1995, pág. 405 (409).

protección a través de alguno de los demás sistemas de protección existentes para los bienes intangibles, los derechos de autor no impiden que tales ideas puedan protegerse a través de la patente, la marca comercial, etc., pero no por los derechos de autor. Por ejemplo, en una obra literaria tradicional como una novela o un ensayo, resulta a priori sencillo deslindar qué son ideas y qué son expresiones. Las segundas comprenden las palabras, las frases y los párrafos tal y como componen la obra, y las primeras, estarían formadas por el argumento, la secuenciación de la historia, los personajes y las escenas<sup>107</sup>. Pero cuando esta separación entre las partes que constituyen ideas de las que constituyen expresión de la obra no es tan obvia, surgen problemas, y en este escenario se sitúan los programas de ordenador.

Cierto es que los programas de ordenador no son las únicas clases de obras consideradas literarias que poseen características funcionales. Las coreografías, las obras musicales o las representaciones artísticas son obras de carácter funcional<sup>108</sup>. Sin embargo, para la mayoría de estas clases de obras, el carácter utilitario o funcional reside en la ejecución, y ésta depende directamente de la expresión literal de la obra<sup>109</sup>. En los programas de ordenador esta afirmación no es válida. Un programa de ordenador no deja de ser una herramienta empleada con la intención de realizar un determinado trabajo, y sólo si alcanza su finalidad, tanto en el sentido de dirigir la máquina que implementa como si satisface los intereses de la persona que lo utiliza, tendrá valor<sup>110</sup>.

Esta afirmación queda corroborada por las definiciones técnicas de programa de ordenador. A modo de ejemplo, en el glosario de estándares del *Institute of Electrical and Electronics Engineers* (IEEE), la voz software es definida como “una combinación de instrucciones computacionales y definiciones de datos que permiten a los elementos

---

<sup>107</sup> D., BAINBRIDGE, *Intellectual property*, Harlow, Munich, 9ª ed., Pearson, 2012, pág. 245 – 250.

<sup>108</sup> J., GINSBURG, “Four Reasons and a Paradox: the Manifest Superiority of Copyright Over Sui Generis Protection of Computer Software”, *Columbia Law Review*, vol. 8, nº 94, 1995, pág. 2559 (2567).

<sup>109</sup> D., CARLETON, “A Behaviour-Based Model...”, *op.cit.*, pág. 410.

<sup>110</sup> En este sentido se ha pronunciado la mayoría de la doctrina, tanto española como internacional. Vid, entre otros, J.A., GÓMEZ SEGADÉ, “La protección jurídica ...”, *op.cit.*, pág. 428; F., TOUBOL, (traducción de M., ESPANÉS) *El software : análisis jurídico*, Buenos Aires, Zavalia, 1990, pág. 24; J., DREXL, *What is protected in a computer program? Copyright protection in the United States and Europe*. IIC Studies, vol. 15, Weinheim, VCH, 1994, pág. 10; P., SAMUELSON *et al*, “A Manifesto ...”, *op.cit.*, pág. 2320; AIPPI, *Annuaire 1988/N*, Congrès d'Amsterdam 1989 (4-10 juin 1988), *Rapports des Groupes Q57*, *Protection du logiciel*, AIPPI Zurich, 1988, pág. 83, 111, 175.

físicos del ordenador realizar funciones computacionales o de control”<sup>111</sup>. En un sentido similar, la *Association for Computing Machinery* (ACM), definió el software como “an ordered set of data that are coded instructions or statements that when executed by the computer cause the computer to process data”<sup>112</sup>.

Desde el ámbito jurídico, si bien es cierto que la postura tomada por del legislador ha sido no definir detalladamente el concepto de programa de ordenador<sup>113</sup>, las Disposiciones Tipo de la OMPI para la protección del soporte lógico<sup>114</sup>, declaraban que un programa de ordenador es “el conjunto de instrucciones capaces, cuando se incorporan en un medio técnico legible por el ordenador, de causar que una máquina con capacidad de procesar la información, indique, actúe y obtenga una función, tarea o resultados concretos”.

De las anteriores definiciones queda bastante claro que un programa de ordenador no es un objeto industrial, sino una información intelectual que puede expresarse de dos formas: en código fuente o en código objeto<sup>115</sup>. El hecho de que el

---

<sup>111</sup> IEEE Standard 610. 12-1990, Standard Glossary of Software Engineering Terminology, 1990, New York.

<sup>112</sup> La Association for Computing Machinery (ACM), fue la primera sociedad científica y educativa fundada acerca de la Computación. Vid: [www.acm.org](http://www.acm.org) (última consulta 1 de junio, 2013).

<sup>113</sup> Ejemplos se encuentran en la Copyright Act de Estados Unidos, que define programa de ordenador como “a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result” Otro ejemplo, en la Directiva Software, que directamente no incluye una definición de programas de ordenador.

<sup>114</sup> Las Disposiciones Tipo de la Organización Mundial de la Propiedad Intelectual (OMPI) fueron una iniciativa legislativa de esta organización, instada por la Organización de Naciones Unidas (ONU) que tuvo lugar en la década de los años 70 del siglo XX. Con dicha propuesta se pretendía elaborar un texto normativo internacional para proteger a los programas de ordenador a través de un sistema sui generis. Inicialmente las Disposiciones Tipo no eran un borrador de convenio internacional, sino que su intención era servir de ayuda para completar las legislaciones nacionales; sin embargo, en una segunda fase de este proyecto, la intención de la OMPI se encaminó a la elaboración de un Tratado Internacional en donde se configurase un sistema de protección sui generis para los programas de ordenador. La iniciativa no tuvo éxito y fue finalmente rechazada en 1985 (Documento UNESCO/OMPI/GE/CCS/1-3, ANNEX A-B, en Copyright, abril 1985, pág. 146-149). Más información sobre este proyecto puede encontrarse en: E., FERNANDEZ MASIÁ, *La protección de los programas de ordenador en España*, Tirant Monografías nº 52, Valencia, Tirant lo Blanch, 1996, pág. 71; J.P., APARICIO VAQUERO, J., DELGADO ECHEVARRÍA, “Comentario al Libro Primero-Título II” en R., BERCOVITZ RODRIGUEZ-CANO, (coord.), *Comentarios a la Ley de Propiedad Intelectual*, Madrid, 3º ed., Tecnos, 2007, pág. 1232.

<sup>115</sup> Dichas instrucciones son lo que la doctrina alemana ha denominado como “Anweisungen an der menschlichen Geist” o indicaciones del espíritu humano. Vid., E., ULMER, *Der Urheberrechtsschutz wissenschaftlicher Werke unter besonderer Berücksichtigung des Programme elektronischer Rechenanlagen*, München, Bayerische Akademie der Wissenschaften, Philosophisch-Historische Klasse, Sitzungsberichte, 1967, H.1, pág. 9; P., SIDLER, *Der Schutz von Computerprogrammen im Urheber- und Wettbewerbsrecht*, Basel, Suiza, AG, 1969. En palabras de GÓMEZ SEGADE, “la esencia del programa consiste en una pura actividad intelectual sin aplicación directa al campo de la técnica industrial”, en J.A., GÓMEZ SEGADE, “La protección jurídica...”, *op.cit.*, pág. 428.

código fuente pueda redactarse por escrito en un lenguaje comprensible por un ser humano, fue una de las razones que llevaron al consenso internacional en cuanto a tratar a un programa de ordenador como una obra literaria<sup>116</sup>. Sin embargo las líneas del código fuente de un programa tienen una función que no depende de su construcción gramatical. Esto significa que el código fuente de un programa de ordenador puede ser completamente diferente al de otro programa y, con todo, cumplir la misma función porque el conjunto de instrucciones que producen un resultado final similar. Es decir, la aportación intelectual del autor radica precisamente en el modo en que el programador llega del problema a la solución, más que en cómo lo expresa, y si bien en alguna ocasión, la funcionalidad de un programa ha sido considerada lo suficientemente “expresiva” como para que los Tribunales la considerasen incluida en el ámbito de protección de los derechos de autor<sup>117</sup>, delimitar los contornos de qué ha de entenderse por funcionalidad expresiva y mera funcionalidad en un programa, es más que un reto. Por este motivo, en alguna ocasión la protección de los derechos de autor fue extendida a elementos funcionales del programa, ideas o utilidades, en una manera más propia del sistema de patentes que de los derechos de autor. Las dificultades surgen, en concreto, en el examen de la copia no literal de determinados aspectos, elementos o niveles entre dos programas de ordenador<sup>118</sup>.

Por lo tanto la concepción de un programa de ordenador como una obra textual, si bien no es totalmente incorrecta sí es incompleta. La característica más importante de un programa reside en que el programa funcione<sup>119</sup>, y esta característica del programa tiene difícil encaje dentro del ámbito de protección de los derechos de autor.

*b) La ausencia de finalidad comunicativa.*

Las obras literarias tienen como finalidad comunicar una serie de ideas del autor al lector, quien, por regla general, es una persona física. Un programa de ordenador

---

<sup>116</sup> vid. Art. 4 Tratado WCT, art. 10 Acuerdo sobre los ADPIC, art. 102(a) Copyright Act de Estados Unidos; art. 1 Directiva Software.

<sup>117</sup> Ejemplos de estos criterios pueden verse en: *Whelan Associates, Inc., v. Jaslow Dental Laboratory, Inc.*, 797 F. 2d. (3d. Cir., 1986), pág. 1222 ; *Computer Assoc. Int’l, Inc. c. Altai, Inc.*, 982F 2.d (2d Cir., 1992), pág. 693.

<sup>118</sup> G., LUNNEY, “Lotus v. Borland: Copyright and Computer Problem”, *Tul. L. Rev.* Nº 70, 6 Part B, 1996, pág. 2397.

<sup>119</sup> “The most important characteristic of computer programs is the fact that they behave: the importance of programs lies in that they provide instructions to make hardware perform certain tasks”. En: P., SAMUELSON *et al*, “A Manifesto ...”, *op.cit.*, pág. 2316.

comunica ideas a una persona experta si está escrito en código fuente. Pero la finalidad principal del programa no es esa, sino la de dirigir el funcionamiento de una máquina. El código fuente no es más que uno de los pasos en el proceso de creación de un programa de ordenador, pero es el código objeto el único que contiene las instrucciones necesarias para hacer funcionar la máquina.

En segundo lugar, el uso y acceso a una obra literaria suelen ser actos coincidentes. Por ejemplo, para acceder a la forma de expresión de una novela es necesario usarla, es decir, leerla. De tal modo que el titular de los derechos de autor sobre la obra enriquece el acervo cultural a través de la revelación de la expresión de su obra. Por tanto, el acceso a la forma de expresión de una obra protegida por derechos de autor no suele ser una cuestión problemática, ya que el objeto de protección de este derecho de propiedad especial es el contenido de la obra en la forma en que su autor ha decidido exponerla a la audiencia, al público<sup>120</sup>.

Sin embargo, la anterior afirmación no puede predicarse de los programas de ordenador, ya que, a diferencia de las obras literarias tradicionales, el usuario de un programa protegido por derechos de autor, para disfrutar de dicho programa sólo podrá reproducirlo en una máquina y acceder a la forma de expresión que no es legible por un ser humano, al código objeto.

Dando un paso más, algún autor ha manifestado que considerar el código objeto como una forma de expresión del programa supone ir contra la propia naturaleza de los derechos de autor, ya que el código objeto no es la manifestación de una idea original sino que es una manifestación funcional, lo cual se parece más a una regla técnica, quizás susceptible de protección a través del sistema de patentes pero no por derechos de autor. Es en el código fuente donde reside la originalidad del creador del programa, y éste, si bien está protegido por los derechos de autor, en la práctica los titulares de los derechos sobre los programas, suelen mantenerlo como secreto industrial, porque no es necesario hacerlo público para que el programa esté protegido. Esta práctica, si bien lícita, restringe de un modo severo la capacidad de los competidores para desarrollar

---

<sup>120</sup> W.R., CORNISH, "Computer Program Copyright and the Berne Convention" en M., LEHMANN, C. TAPPER, (eds.), *A Handbook of ...*, *op.cit.*, pág. 197.

programas interoperables, porque tanto el Know-how como la información necesaria para alcanzar la interoperabilidad se encuentran en el código fuente<sup>121</sup>.

En 1994, DREXL ya había anticipado este problema, al decir que, en muchos casos la ausencia de finalidad comunicativa es de importancia fundamental para trazar la frontera entre protección y competencia. Esta característica limita el acceso público a la idea del programa porque los programas de ordenador son distribuidos casi exclusivamente en código objeto<sup>122</sup>. El problema vaticinado por DREXL se confirmó en el tratamiento dado en el Acuerdo sobre los ADPIC a los programas de ordenador y posteriormente, en el artículo 4 del Tratado WCT.

Según ha indicado COTTIER, la referencia expresa que hace el artículo 10 de TRIPS a ambos, código fuente y código objeto, deja claro que el programa de ordenador es susceptible de protección no sólo cuando sea expresado en una forma humanamente comprensible (y por tanto con finalidad comunicativa) como es el código fuente, sino también es susceptible de protección si se expresa en una forma no comprensible para el ser humano, (carente de finalidad comunicativa) como es el código objeto<sup>123</sup>.

Pero como veremos en los siguientes capítulos, ni los tribunales ni los legisladores han entendido que la ausencia de finalidad comunicativa sea óbice para proteger los programas de ordenador a través de los derechos de autor.

Aún así, no hemos de olvidar que por ejemplo en Estados Unidos, la Copyright Act tiene entre sus objetivos de protección no sólo incentivar la creación, sino también la publicación y distribución de la obra. La Ciencia y las Artes progresan sólo si el público puede tener acceso a las obras. El titular de los derechos de copyright, normalmente hará pública la obra ya que es la única forma de que reciba una recompensa económica a su esfuerzo. Es la divulgación de una obra lo que permite al

---

<sup>121</sup> J., GIBSON, "Once and Future Copyright", *Notre Dame Law Review*, 81, 2005-2006, pág. 167 (190); J., BARTMAN, *Grenzen der Monopolisierung durch Urheberrechte am Beispiel von Datenbanken und Computerprogrammen: Eine rechtsvergleichende Studie des europäischen, deutschen und US-amerikanischen Rechts*, Cologne, Berlin, Munich, Carl Heymanns, 2005, pág. 122; J-K., JEONG, "Comparative Study on Legal Protection in the USA, EU, Japan and Korea for Computer Programs: Focus on Program Reverse Engineering", *Institute of Intellectual Property Bulletin*, 2006, pág. 140 (141).

<sup>122</sup> J., DREXL, *What is protected in ...*, *op.cit.*, pág. 11.

<sup>123</sup> T., COTTIER, *Concise international and European IP law: TRIPS, Paris Convention, European enforcement and transfer of technology*, Alphen aan den Rijn, 2ª ed., Kluwer Law International, 2011, pág. 199. En igual sentido se pronunció J., REICHMAN, "Legal Hybrids between the Patent and Copyright Paradigms", *Columbia Law Review* 94 (8), 1994, pág. 2432 (2486).

público acceder a su expresión y a las ideas subyacentes. Ambos elementos pueden “inducir” nuevas creaciones, aunque sus ideas y expresiones no puedan utilizarse de igual modo. Pero los programas de ordenador al ser comercializados en código objeto, solo revelan su función (lo cual por supuesto es parte de la idea), y sus interfaces gráficas, nada más. El acceso a otras partes del programa no es posible sin acceder al código fuente, que al no haberse divulgado, no es directamente accesible para quien legítimamente obtenga y use el programa. Sólo a través de técnicas de ingeniería inversa puede accederse al código fuente.

Por lo tanto, a diferencia de las obras literarias en general, los usuarios legítimos de un programa de ordenador no podrán acceder a una forma de la obra que permita ni su disfrute, ni su apreciación, ni incluso la mera contemplación de la expresión original de la obra. Sólo podrán contemplar cómo funciona al reproducir el programa en una máquina, de modo que la obra no comunica nada al usuario.

Como ha indicado GIBSON, en el caso del software nos encontramos ante una obra protegida por copyright cuya arquitectura única, permite a su autor beneficiarse de su creación sin tener que divulgar ni la expresión creativa ni las ideas de la obra a su comprador. El titular de los derechos por tanto, recibe una recompensa de los derechos de autor sin contribuir al beneficio de la comunidad. Cualesquiera sean las ideas que inspiren el código fuente del programa, se quedan para el titular; para cualquier miembro de la comunidad el programa no es más que un impenetrable e indescifrable conjunto de unos y ceros<sup>124</sup>.

*D) La ficción jurídica de tratar al programa como obra literaria en el marco del Convenio de Berna.*

Como se ha expuesto en los epígrafes anteriores, el encaje del programa de ordenador como obra literaria, debido a las especiales características de funcionalidad y

---

<sup>124</sup> J., GIBSON, "Once and ...", *op.cit.*, pág. 181. En igual sentido se había pronunciado también L., LESSIG, *The future of Ideas: The Fate of the Commons in a connected world*, New York, Random House, 2001, pág. 252-253; disponible en: [http://www.lessig.org/blog/2008/01/the\\_future\\_of\\_ideas\\_is\\_now\\_fre\\_1.html](http://www.lessig.org/blog/2008/01/the_future_of_ideas_is_now_fre_1.html) (última consulta 1 de junio, 2013).



ausencia de comunicación del programa, resulta bastante forzado para algunos de sus elementos, en concreto para los códigos, que son el corazón del programa<sup>125</sup>.

La cuestión radica en si desde una interpretación estrictamente jurídica del texto del Convenio de Berna, puede mantenerse la consideración del programa como obra literaria o no.

El artículo a considerar es el 2.1, donde se define el concepto de “obra literaria y artística”.

Partiendo de una interpretación literal, un programa de ordenador estará dentro del ámbito de protección de la cláusula general del citado artículo, si el mismo constituye “una producción en el campo literario, científico y artístico, cualquiera que sea el modo o forma de expresión”. Un programa puede describirse a través de una serie de instrucciones (los códigos). El código fuente no es más que la expresión de una serie de pensamientos humanos, en concreto, las instrucciones para la máquina, escritas en un lenguaje de programación<sup>126</sup> por esa persona y que por tanto constituye una producción en el sentido del Convenio de Berna. Las obras literarias pueden estar expresadas en un lenguaje de cualquier clase, con independencia de que dicho lenguaje sea comprensible por una sola persona o por millones.

DREIER ha argumentado que el Convenio no contiene una verdadera definición de lo que ha de entenderse por obra literaria. Antes bien solo describe las obras protegidas por el texto como “obras literarias y artísticas”, de lo cual podría inferirse literalmente que el Convenio solo se ocupa de la protección de creaciones del ámbito literario y artístico. Si la expresión citada incluyese también el adjetivo “científicas” quedaría claro que la intención del Convenio era la de incluir en su marco la protección de obras de carácter creativo que versen sobre asuntos científicos, a pesar de que en última instancia esta clase de obras están protegidas sobre todo por la Convención de

---

<sup>125</sup> Si el programa de ordenador es considerado como el conjunto de los códigos y el material preparatorio y/o complementario, la asimilación a una obra literaria de estos últimos elementos no presenta ningún problema.

<sup>126</sup> Los lenguajes de programación, al igual que los lenguajes que usamos para comunicarnos entre las personas, poseen un léxico, una sintaxis y una semántica. A., PRIETO ESPINOSA, *Introducción...*, *op.cit.*, pág. 519.

París y otros instrumentos internacionales para la protección de la propiedad industrial<sup>127</sup>.

Quizás la elección de la expresión “obras literarias y artísticas” no fue la más acertada, pero en la frase siguiente el texto indica claramente que bajo tal expresión se incluyen producciones del “*campo literario, científico y artístico, cualquiera que sea el modo o forma de expresión*”, con lo cual, no podemos compartir esta postura.

En relación a la funcionalidad y ausencia de finalidad comunicativa del programa, algún jurista español como ABADÍA VICENTE, ha argumentado que si la diferencia entre la propiedad intelectual y la propiedad industrial es que la esencia de la primera es comunicar ideas, belleza y sentimientos al género; en cambio en la segunda predomina intensamente el sentido práctico y transformador de la materia, no cabe plantearse la comparación entre la lectura del Quijote de Miguel de Cervantes con el Software. Si el programa de ordenador no puede transmitir la belleza, sensibilidad y plenitud espiritual que deposita en el ser humano la lectura de un poema, la respuesta, además de ser negativa, lleva anexa otra afirmación: no tienen nada que ver la obra literaria con el programa de ordenador<sup>128</sup>.

Desde una interpretación sistemática, sin embargo, la cláusula del artículo 2.1 del Convenio de Berna no establece entre los requisitos excluyentes para considerar a una obra protegida que no presente atractivo estético al ojo humano, ni que sea funcional. Esta afirmación se ve confirmada en la lista de ejemplos que incluye el propio artículo, en la que podemos encontrar tanto obras sin atractivo estético, como pueden ser las cartas, los mapas geográficos o los dibujos técnicos, y también encontramos ejemplos de obras funcionales opuestas a las obras de “*l’art pour l’art*”, como son obras arquitectónicas, obras de arte aplicado o mapas.

Una interpretación teleológica de la norma conduce a concluir que es posible incluir a los programas de ordenador dentro del ámbito de protección del Convenio, ya que el objetivo del mismo, la protección de los autores y sus obras, como indican el

---

<sup>127</sup> T., DREIER, “Comentario a la Convención de Berna” en T., DREIER, P.B., HUGENHOLTZ, (eds.), *Concise European Copyright Law*, Alphen aan den Rijn, The Netherlands, Kluwer Law International, 2006, pág. 12.

<sup>128</sup> M., ABADÍA VICENTE, “Cuestiones complejas que suscitan la propiedad intelectual e industrial. Especial consideración de los programas de ordenador” en M., ABADÍA, J., CARDENAL, *et al*, *Patentes, marcas, software*, Murcia, Laborum, 2001, pág. 31.

Preámbulo y el artículo 1, justifica proteger los programas de ordenador que sean creados por personas físicas. Otros argumentos que favorecen esta interpretación son que el listado del artículo 2.1 no sea exhaustivo, lo cual manifiesta la intención de los redactores de proteger el mayor número de obras posibles, de manera que cualquier nueva clase de producción pudiera quedar cubierta por la amplia definición de la expresión “obras literarias y artísticas”.

En último lugar, podría argumentarse en el sentido mostrado por DREIER, que ningún país ni grupo de países está capacitado para imponer un criterio interpretativo del Convenio de Berna que vincule a los demás países miembros del acuerdo. Por lo tanto, podría concluirse que las nuevas producciones no estarán protegidas en el marco del Convenio en tanto no se hubiesen incorporado de manera unánime al texto sea a través de un proceso de revisión o por la práctica común de los Estados (en consonancia con lo dispuesto en el artículo 31 y siguientes de la Convención de Viena)<sup>129</sup>. Esta argumentación tampoco cabe en relación a los programas de ordenador, ya que como se ha indicado al inicio del epígrafe, su inclusión como obra literaria en el marco del Convenio de Berna está refrendado por dos Tratados Internacionales.

En resumen, parece que ni la funcionalidad ni la ausencia de finalidad comunicativa son obstáculo suficiente para que el programa pueda considerarse como obra literaria en el marco del Convenio de Berna. Sin embargo, el hecho de elegir esta opción como la más adecuada para proteger el software, trae consigo una serie de problemas que afectan sobre todo a la delimitación del ámbito de protección del programa y a la calificación de una infracción sobre los derechos exclusivos del titular del programa. Dos cuestiones directamente implicadas en el debate de la interoperabilidad.

### **3. La ingeniería inversa del software en general y la descompilación en particular.**

El Diccionario de la Real Academia Española (DRAE) define “ingeniería” como “el estudio y aplicación, por especialistas, de las diversas ramas de la tecnología”. Una definición más detallada la encontramos en el American Engineers’ Council for Professional Development (ECPD), que define “ingeniería” como “la aplicación

---

<sup>129</sup> T., DREIER, “Comentario al Convenio de Berna...”, *op.cit.*, pág. 13.

creativa de principios científicos en el diseño y desarrollo de estructuras, máquinas, aparatos o procesos de fabricación, o de obras que son utilizadas por sí mismas o en combinación con otras”<sup>130</sup>. Es por tanto un proceso que parte de unos principios y cuya meta es lograr un producto elaborado, como es el caso de la Ingeniería del software.

La ingeniería inversa es justo lo contrario: es un proceso que parte de un producto conocido y trabaja sobre él para conseguir el principio técnico en el que se basa. En cierto modo consiste en hacer ciencia sobre la tecnología, en vez de sobre el mundo natural.

La jurisprudencia estadounidense ha definido la ingeniería inversa como “*a fair and honest means of starting with the known product and working backwards to divine the process which aided in its development or manufacture*”<sup>131</sup>.

No hace demasiado tiempo, la ingeniería inversa se consideraba no tanto una tecnología sino una actividad lúdica, practicada por mucha gente. Por ejemplo, en los comienzos de la era electrónica, cuando se inventaron la radio y más tarde la televisión, cuánta gente no se aventuró a ver qué había dentro de esos aparatos. Esto es un acto de ingeniería inversa. Hoy día, sin embargo, los componentes electrónicos digitales son tan diminutos que una persona sin conocimientos específicos y sin los medios adecuados no sería capaz de ver qué hay dentro y descubrir, por ejemplo, cómo funciona algo tan habitual en nuestros tiempos como un teléfono móvil.

La ingeniería inversa del software es una de las tecnologías más complejas y complicadas que existen a día de hoy. PRESSMAN la define como “el proceso de análisis de un programa con el fin de crear una representación del programa con un nivel de abstracción más elevado que el código fuente. La ingeniería inversa es un proceso de recuperación de diseño”<sup>132</sup>.

En sí misma, la ingeniería inversa del software está compuesta por un conjunto de artes o habilidades: ruptura de códigos, solución de puzzles, programación, y análisis

---

<sup>130</sup> La definición original está en inglés: “The creative application of scientific principles to design or develop structures, machines, apparatus, or manufacturing processes, or works utilizing them singly or in combination”. Disponible en: <http://en.wikipedia.org/wiki/Engineering>. (Última consulta 1 de junio, 2013).

<sup>131</sup> *Secure Services Technology, Inc. v. Time & Space Processing*, 722 F. Supp., n.16 (E.D. Va. 1989) pág. 1354.

<sup>132</sup> R., PRESSMAN, *Ingeniería del Software...*, op.cit., pág. 546.

lógico<sup>133</sup>. Aunque sus orígenes parecen situarse en el escenario de la Guerra Fría, tras la Segunda Guerra Mundial, lo cierto es que la cuestión de la licitud de su utilización en los programas de ordenador no se plantea hasta principios de los años noventa, en Estados Unidos. Y no es hasta mediados de los años noventa que también en Estados Unidos se publica el primer trabajo de investigación donde se presentan por primera vez técnicas de descompilación, una de las modalidades de ingeniería inversa, hasta entonces desconocidas<sup>134</sup>. Desde entonces, tan rápido ha avanzado la tecnología informática, que la ingeniería inversa del software se ha convertido en un proceso puramente virtual, que requiere dos elementos esenciales: un procesador o CPU, y una mente humana.

En cuanto a los usos de estas técnicas en la industria informática, se distinguen tres clases de aplicaciones prácticas: las relacionadas con la seguridad del software, las destinadas a su mejora, y las destinadas a la innovación y la investigación.

Respecto de la seguridad del software, la ingeniería inversa se ha venido empleando con los virus informáticos, en un doble sentido: por una parte, los programadores de software malicioso, para crear virus más dañinos, y por otra parte, las empresas desarrolladoras de software, para crear antídotos más eficaces.

La ingeniería inversa tiene también gran importancia en el terreno de los estudios de codificación y encriptado. El objetivo de las empresas informáticas es hacer que el acceso a un programa codificado esté reservado sólo para quien posea una determinada clave (*key*). Esta llave se conoce como algoritmo criptográfico o *cipher*.

---

<sup>133</sup> E., EILAM, *Reversing: Secrets of reverse engineering*, Indianapolis, Indiana, Wiley Publishing Inc. 2005, pág. 4.

<sup>134</sup> C., CIFUENTES, *Reverse Compilation Techniques*. [tesis doctoral] Queensland University of Technology (US), Julio 1994, disponible en:

<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.105.6048>. (Última consulta 1 de junio, 2013)

En el resumen declara la autora: "Techniques for writing reverse compilers or decompilers are presented in this thesis. These techniques are based on compiler and optimization theory, and are applied to decompilation in a unique way; these techniques have never before been published". En una entrevista publicada en el periódico de la Queensland University of Technology, Inside QUT, n.132, Mayo 16-29, 1995, decía CIFUENTES: "Reverse compilation is very difficult because you're going from a low-level program to a high-level one and so there is a lot of information missing which you have to create. A lot of people have attempted it but they haven't been able to generate the final, high-level code". Unos años antes, en 1990, ELLIOT J. CHIKOFISKY, ya había apuntado en un artículo publicado en la IEEE Magazine: "Reverse Engineering is evolving as a major link in the software life cycle, but its growth is hampered by confusion over technology". Vid., E.J., CHIKOFISKY, "Reverse Engineering and Design Recovery: A Taxonomy", IEEE Magazine, January 1990, pág. 13 (14).

Por medio de la ingeniería inversa puede determinarse el nivel de seguridad proporcionado por un determinado sistema de acceso condicionado.

Como ejemplo de este primer uso, encontramos las conocidas como medidas tecnológicas de protección (MTP o DRM en inglés), que se basan en su práctica totalidad en estas técnicas de encriptación, y que constituyen mecanismos insertos en obras protegidas por derechos de autor<sup>135</sup>. La ingeniería inversa del software desde la perspectiva de las MTP, como se verá más adelante, reviste una gran importancia tanto para los titulares de los derechos de autor sobre las obras en formato digital, como para la comunidad de usuarios de esta clase de obras. Para los primeros, la ingeniería inversa se emplea para mejorar las técnicas de codificación y encriptado de sus programas, porque las técnicas de ingeniería inversa permiten detectar las vulnerabilidades o brechas de seguridad del programa. Desde el punto de vista del usuario legítimo, las MTP pueden limitar el uso de la obra más allá de lo que el ordenamiento jurídico permite o incluso, hacer la obra inutilizable. En estos casos la ingeniería inversa es útil para restablecer la vigencia de usos legítimos.

Respecto de la segunda aplicación mencionada, la mejora del software, las técnicas de ingeniería inversa permiten alcanzar la interoperabilidad entre programas no compatibles. Ciertos sistemas operativos se caracterizan por encontrarse insertos en un mercado que tiende hacia la estandarización alrededor de un solo producto, en atención a que los consumidores se benefician de trabajar en la misma plataforma o multiplataforma que todo el mundo utiliza. Esta situación confiere de hecho, a los fabricantes de determinados sistemas operativos, una posición de dominio en el mercado que puede crear una barrera de entrada a nuevos competidores.

En tercer lugar, el fomento de la investigación y la innovación, se traduce en dos aspectos. Por un lado, el uso de la ingeniería inversa para lograr la interoperabilidad entre programas no por razones de estandarización alrededor de un solo producto, sino con el fin de crear nuevos programas capaces de intercambiar información con los ya existentes en el mercado. Además, la utilización de técnicas de ingeniería inversa para auditar un programa, para evaluar su calidad y sus vulnerabilidades ayuda a los

---

<sup>135</sup> J.L., BARO RÍOS, “Derribando algunos mitos en torno a la ingeniería inversa del software”, en A., CERDA SILVA (ed.), *Acceso a la cultura y Derechos de Autor. Excepciones y Limitaciones al Derecho de Autor*, Santiago de Chile, Ford Foundation, 2008, pág. 142, disponible en: <http://www.derechosdigitales.org/a2k/wp-content/uploads/baro.pdf>, (última consulta 1 de junio, 2013).

ingenieros, especialmente a los que se encuentran en formación, a mejorar sus técnicas de programación<sup>136</sup>.

En relación a las clases de técnicas de ingeniería inversa en particular, se habla de tres tipos de análisis, el “análisis de caja negra”, el “desensamblaje” y la “descompilación”.

La primera técnica de ingeniería de inversa, el análisis de caja negra o *black-box* tiene como objetivo la inversión de sistema (*system-level reversing*). Sin acceder directamente a los códigos del programa, pero con la ayuda de la documentación preparatoria del programa y a través de observar su funcionamiento en el computador, el o los programadores han de intentar visualizar cada función concreta del programa como una caja negra independiente. A continuación, con la información adquirida a través de la documentación preparatoria y la observación, cada programador ha de intentar implementar su caja negra, responsable de una tarea específica, probar si funciona y en caso afirmativo, integrar su caja con los demás componentes del programa. La idea general es pues visualizar el programa como un conjunto de cajas negras para determinar la estructura general del programa<sup>137</sup>.

A un nivel más profundo se sitúan las técnicas de conversión del código (*code-level reversing*), tanto el desensamblaje como la descompilación, cuya meta es la extracción de los conceptos nucleares de diseño del programa e incluso los algoritmos que lo componen.

En primer lugar, el desensamblaje (*disassembly*) es un proceso de traducción del código objeto del programa al lenguaje ensamblador<sup>138</sup> de la computadora, que permite determinar las interfaces del programa que intercambian información con el

---

<sup>136</sup> Vid., P., SAMUELSON, S., SCOTCHMER, “The Law and Economics...”, *op.cit.*, pág. 1604; J., ABBOT, “Reverse Engineering of Software: Copyright and Interoperability”, *J.L. & Inf. Sci.*, vol. 14, 2003, pág. 7 (12); E., EILAM, *Reversing ...*, *op.cit.*, pág. 4.

<sup>137</sup> E., EILAM, *Reversing ...*, *op.cit.*, pág. 27.

<sup>138</sup> B., FITZGERALD, C., CIFUENTES, M., LEHMAN, “Innovation, Software and Reverse Engineering”, *Santa Clara Computer & High. Tech. L.J.*, vol. 18, 2001-2002, pág. 121 (125).

El lenguaje ensamblador es un lenguaje de programación de bajo nivel para los computadores, microprocesadores, microcontroladores y otros circuitos integrados programables. Implementa una representación simbólica de los códigos de máquina binarios y otras constantes necesarias para programar una arquitectura dada de CPU y constituye la representación más directa del código máquina específico para cada arquitectura legible por un programador. Esta representación es usualmente definida por el fabricante de hardware, no por el fabricante del programa. (Vid. E., EILAM, *Reversing...*, *op.cit.*, pág. 15-16).

componente físico donde el programa ha de funcionar. Esta clase de técnicas fueron muy utilizadas en los años 80 y 90 del siglo pasado para entender el funcionamiento de los antivirus y neutralizarlos. Hoy día, debido a la sofisticación de los virus informáticos, ha dejado de ser una herramienta eficaz para esa finalidad. Se sigue utilizando esta clase de técnica de ingeniería inversa sobre todo en ambientes académicos y de investigación, cuando se requiere la interoperabilidad del programa con un hardware determinado, pero no hay acceso directo al código fuente del programa.

La otra técnica de ingeniería inversa de carácter profundo es la descompilación y es el proceso más complejo de los tres mencionados. A través de ella se pretende traducir el código objeto del programa en código fuente; es justo lo contrario al proceso de compilación del programa que tiene lugar cuando es creado. El objetivo de la descompilación es descifrar, a partir del lenguaje máquina, las ideas en que se basa el código fuente del programa, escrito en lenguaje de alto nivel<sup>139</sup>. Con carácter general, el proceso de descompilación incluye operaciones de desensamblaje del código objeto en lenguaje ensamblador, analizando a continuación el código ensamblador, tratando de recuperar información del lenguaje de alto nivel en que está escrito el código fuente, y generando una especie de “código fuente” final. Durante los procesos de ingeniería inversa de descompilación, la transformación o traducción del código objeto en código fuente se realiza empleando otro programa de ordenador muy especial denominado compilador. Con este programa no sólo se traduce el código fuente en código objeto y viceversa, sino que el mismo optimiza al programa que compila, arreglando instrucciones escritas en lenguaje de bajo nivel que impiden al programa ejecutarse más rápido<sup>140</sup>. En el proceso de compilación, el programa produce dos efectos colaterales sobre el código importantes: En primer lugar, degrada su legibilidad por un ser humano. En segundo lugar, el resultado de la compilación puede haber sido reestructurado de una forma distinta a como fue introducido en este programa, precisamente por razones de optimización<sup>141</sup>.

---

<sup>139</sup> *Ibidem*, pág. 14.

<sup>140</sup> A., JOHNSON-LAIRD, "Software Reverse Engineering in the Real World", U. Dayton L. Rev, 19, 1993-1994, pág. 843 (859).

<sup>141</sup> E., EILAM, *Reversing ...*, *op.cit.*, pág. 54.



Es por ello un proceso incompleto, porque durante la creación del programa original, al compilarlo desde código fuente a código objeto, ya se ha perdido mucha información. De modo que al descompilar el programa se parte de una situación de clara desventaja. Es cierto que la habilidad del programador o programadores que llevan a cabo un proceso de descompilación puede, de algún modo suplir esta desventaja inicial. Pero la calidad del producto final originado tras un proceso de descompilación es muy inferior al código fuente del programa original<sup>142</sup>.

### *3.1. Ingeniería inversa e interoperabilidad.*

Para un programador informático, lograr la interoperabilidad puede convertirse en su peor pesadilla. Desde el punto de vista de un programador, interoperabilidad significa que el programa que ha creado pueda comunicarse con una programa escrito por otra persona. Esta otra persona puede ser alguien que está trabajando en su misma empresa y sobre el mismo programa, o puede ser el programador de un programa diferente en su totalidad. Como se indicó antes, en la Sociedad de la Información, el software está en todas partes y todas estas partes necesitan comunicarse entre ellas, ya sean programa de aplicación con sistemas operativos, aplicaciones con librerías de otros programas, o programas de aplicación con otros programas de aplicación.

En la mayoría de los casos, el programador que quiere alcanzar la interoperabilidad de su programa con el de un tercero solo va a poder acceder a la librería del programa de ese tercero, no a la parte del código fuente que contiene la información interoperable respecto del componente con el que quiere alcanzar interoperabilidad. En tales casos, el programador está obligado a confiar en la documentación preparatoria que el fabricante del primer programa ponga a su disposición en el mercado. Pero esta documentación con frecuencia no hace mención ni a determinadas funciones del programa, ni a sus parámetros o la serie completa de funcionalidades del programa, necesarias a efectos de lograr la interoperabilidad<sup>143</sup>.

---

<sup>142</sup> Sobre las técnicas de descompilación vid. E., EILAM, *Reversing...*, *op.cit.*, págs. 457-478.

<sup>143</sup> Más razones que justifican la ingeniería inversa del software a efectos de interoperabilidad en: H., HARTE-BAVENDAMM, "Wettbewersrechtliche Aspekte des Reverse Engineering von Computerprogrammen", GRUR 1990, pág. 657 (659); A., OHLY, "Reverse Engineering: Unfair Competition or Catalyst for Innovation?" en J., STRAUS, (ed.), *Patents and Technological Progress in a*

Un ejemplo ilustrativo de esta situación es el sistema operativo Windows, que históricamente es conocido entre la industria informática por contener cientos de interfaces de aplicación de programación (APIs) vacías. Estos componentes están vacíos por muy diversas razones; entre ellas el garantizar la compatibilidad del sistema operativo con otros programas basados en Windows. De hecho, mucha gente del sector ha afirmado que la razón de Microsoft para mantener las APIs vacías es disponer de una ventaja competitiva ante otros fabricantes de software. Cualquier producto de Microsoft puede aprovecharse de una APIs vacía para proporcionar una mejor interoperabilidad vertical entre sus productos, pero el creador de un programa que no sea Microsoft no sabrá cómo sacar partido de este componente vacío. En estos casos, el acceso a la información interoperable del código fuente suele ser una medida más rápida y eficiente que contactar con la empresa fabricante del software con el que el nuevo programa necesita comunicarse, explicar un problema muy técnico y de carácter urgente, esperar su respuesta y comprobar si efectivamente funciona, puede acarrear un problema debido al dinamismo del mercado informático. Incluso si se acude a la vía judicial, en casos donde el fabricante de software se niega a publicar las interfaces o las retiene para controlar la competencia, la solución puede llegar demasiado tarde.

Sin embargo, si se opta por la opción de intentar acceder al código fuente a través de la descompilación, el programador se encontrará con una barrera jurídica: la protección conferida por el derecho de autor sobre el programa, que posiblemente califique su comportamiento como una infracción.

### *3.2. El carácter infractor de la descompilación para el derecho de autor.*

En el mundo analógico, el derecho de autor no planteaba ningún problema a la hora de realizar un proceso de ingeniería inversa de una obra literaria en un sentido totalmente funcional, es decir, de la lectura de un libro cabe extraer las ideas en que se fundamenta esta obra, y para ello no es necesario hacer copia alguna del libro, con lo cual, no hay riesgo de infracción de los derechos de autor. Sin embargo, en el mundo digital, cada uso de una obra literaria supone al menos la realización de una copia, para

---

*Globalized World: Liber Amicorum*, Berlin, Springer, 2009, pág. 535 (537); disponible en SSRN: <http://ssrn.com/abstract=1523649> (última consulta 1 de junio, 2013)

la que a priori, será necesaria la autorización de su propietario. En relación a los procesos de ingeniería inversa se traduce del siguiente modo:

En circunstancias normales, para realizar un análisis de caja negra (*black-box testing*), el informático debe realizar una serie de reproducciones del programa para observar cómo es ejecutado por el ordenador, para observar su comportamiento en la máquina. Para ello, en primer lugar, la máquina realizará una copia temporal en la memoria del ordenador (RAM). A continuación, será necesario realizar más copias temporales del programa durante el proceso de lectura y creación de las partes del código fuente que los programadores transcriban del código objeto<sup>144</sup>. Al final de proceso, la transcripción de los códigos obtenidos constituye una obra derivada no autorizada. Si esos resultados obtenidos se fijaran en algún soporte, tangible o intangible, constituirían una reproducción no autorizada, y de divulgarse a terceros, podrían constituir un acto de comunicación pública no autorizado<sup>145</sup>.

La descompilación tal como se indicó páginas atrás, es un procedimiento mucho más complejo a través del cual se intenta acceder a “la esencia”, a las ideas y los elementos funcionales que están en el código fuente del programa<sup>146</sup>. El resultado final de este proceso es una copia de poca calidad pero funcionalmente exacta a la del código fuente original, que permite a otro programador entender, sino toda, una gran parte de la estructura y el modo de operar del programa descompilado<sup>147</sup>.

Veamos uno por uno dónde residen los principales puntos de fricción entre este procedimiento y la protección que el derecho de autor otorga al autor y al titular del programa.

---

<sup>144</sup> O., STAFFELFACH, *Die Dekompilierung von Computerprogrammen gemäss Art. 21 URG*, Bern, Stämpfli, 2003, pág. 88; E., EILAM, *Reversing...*, *op.cit.*, pág. 19.

<sup>145</sup> Vid. R.V., DONAHOE, “Does Intermediate Copying of Computer Software for the Purpose of Reverse Engineering a Non-Infringing Product Infringe the Copyright in the Software?”, B.C. Intell. Prop. & Tech. F., n° 16, 2001, pág. 1 (2); A., OHLY, “Reverse Engineering: ...”, *op.cit.*, pág. 536.

<sup>146</sup> D., BAINBRIDGE, *Legal protection of computer software*, Haywards Heath, 5° ed., Tottel publ., 2008, pág. 113. La definición original en inglés “Decompilation is the process by which ideas and techniques as functional elements embodied in computer program code are made available by means of an operation”.

<sup>147</sup> M., MELVILLE, D., NIMMER, *Nimmer on Copyright*, vol. 4, New York, Bender, 2009, §13.05[D][4], pág. 13-230.18; vid. también la sentencia del caso E.F. Johnson Co. v. Uniden Corp., 623 F. Supp, D. Minn. 1985, pág. 1485 (1488-90), en donde el juez indica que “decompilation does not yield a perfect copy of the source code, but can afford an understanding of much of the structure and operation of a program”.

*A). El derecho de reproducción.*

La facultad que el derecho de autor concede al titular de la obra de controlar su copia o fijación por terceras personas, es lo que conocemos como derecho de reproducción con carácter general.

De entre todas las facultades de carácter patrimonial que el derecho de autor concede al autor, el derecho de reproducción es el eje central del derecho de autor y la facultad exclusiva de mayor repercusión económica<sup>148</sup>.

La descompilación de un programa de ordenador va a originar distintas copias o fijaciones del programa mientras se lleva a cabo, y el resultado final de este proceso de ingeniería inversa, de ser satisfactorio será una copia del programa. Ciertamente que como indicamos se trata de una copia funcionalmente igual y de poca calidad, pero a efectos de la normativa de derecho de autor, la calidad de la copia no será una circunstancia a tener en cuenta. La existencia de una copia o la fijación de la obra no autorizada por el titular es lo que importa a efectos del derecho de exclusiva.

Aquí reside el punto de fricción entre la descompilación y derecho de reproducción sobre el programa: si las copias generadas durante la descompilación infringen el derecho de exclusiva del titular sobre el programa.

La primera pista sobre si esa copia o fijación de los códigos del programa constituyen una infracción nos la proporciona el ordenamiento europeo, cuando la Directiva de 1991 sobre protección jurídica de los programas de ordenador regula la descompilación a través de una excepción al derecho de exclusiva. El derecho norteamericano, como se verá en el siguiente capítulo, aun siendo paradigma de un sistema de limitaciones abierto, introdujo una limitación de clara inspiración en la de la descompilación de la Directiva europea en la Digital Millennium Copyright Act<sup>149</sup>, lo

---

<sup>148</sup> Consecuencia de la importancia histórica que la propia reproducción ha tenido en la formación del derecho de autor. Vid. E., SANTORO, "Cenni sul diritto di riproduzione delle dell'ingegno", Riv. Dir. Comm. I, 1966, pág. 65; M., BOTANA AGRA, "Los derechos de explotación de la obra de autor en la Ley española de propiedad intelectual", ADI, 1987, pág. 37 (45); Vid. El Libro Verde de la Comisión "Derecho de Autor y Derechos Afines en la Sociedad de la Información", COM(95) 382 final, 1995, pág. 49: "El derecho de reproducción es la esencia del derecho de autor y los derechos afines".

<sup>149</sup> Digital Millennium Copyright Act of 1998, Pub. L. No. 105-304, 112 Stat. 2860 (Oct. 28, 1998). Esta norma de derecho norteamericano incorporó a su ordenamiento interno los Tratados de la OMPI de 1996 sobre derechos de autor. Vid. infra., capítulo 2.3.

cual también es indicativo de la postura del legislador norteamericano sobre esta cuestión.

La segunda clave, se encuentra en el alcance del derecho de reproducción sobre el programa. A este respecto hay que hacer varias puntualizaciones:

- En primer lugar, en el marco normativo internacional, el Convenio de Berna en su artículo 9, dispone sobre el derecho de reproducción que es “el derecho exclusivo de autorizar la reproducción de sus obras por cualquier procedimiento y bajo cualquier forma.” La aplicación de la expresión “cualquier procedimiento y bajo cualquier forma” a las reproducciones antes comentadas que se producen durante la descompilación, implica que ni el carácter temporal o permanente de las copias del código, ni su calidad, ni si son sobre el código objeto o el código fuente se han de tener en cuenta para determinar la existencia de reproducción sujeta a autorización en las obras literarias.
- En el derecho comunitario, el artículo 4 de la Directiva declara que los actos de reproducción total o parcial, tanto permanentes como temporales, están sujetos a autorización del titular<sup>150</sup>. A continuación detalla una serie de actos que siempre estarán sujetos a autorización del titular: carga, presentación, ejecución, transmisión y almacenamiento<sup>151</sup>. Si son o no reproducción es una cuestión que deberá determinarse caso por caso, siendo los tribunales nacionales quienes interpreten el texto y especifiquen que actos constituyen o no una reproducción.

---

<sup>150</sup> En relación a las copias temporales (copias RAM, efímeras, caché, temporales en sentido estricto), inicialmente parte de la doctrina, en especial la alemana, cuestionó que pudiesen ser consideradas como acto de reproducción, porque como indicó LEHMANN, el término reproducción es un concepto legal que debiera interpretarse sobre una base normativa, y que por lo tanto, no debiera depender del estado de la técnica o del particular modo en que funcione un ordenador. Vid. M., LEHMANN, "Erwiderung. Reverse Engineering ist kein Vervielfältigung i.S.d. §§ 16, UrhG 53", CR 6, 1990, pág. 94 (95). Vid también, T., DREIER, "The Council Directive of 14 May 1991 on Legal Protection of Computer Programs", EIPR 13, 1991, pág. 319 (323).

<sup>151</sup> En opinión de la doctrina española esta enumeración no se hizo a efectos de determinar si son o no actos de reproducción, sino de someterlos al ámbito del *ius prohibendi* del titular del derecho en los casos que impliquen reproducción. Vid. J., MASSAGUER FUENTES, "La adaptación de la Ley de propiedad intelectual a la Directiva CEE relativa a la protección jurídica de los programas de ordenador", RDM, núm. 199-200, 1991, pág. 39 (51); E., FERNÁNDEZ MASIÁ, *La protección ...*, *op.cit.*, pág. 152.

- En el derecho estadounidense, la respuesta de la jurisprudencia es similar a la europea, declarando al respecto de los programas de ordenador que "*the making of a hard copy, a translated copy or another disk copy of the original target code*" y que "*any intermediate copy of the object code made in order for a programmer to read and comprehend it suffices to meet the fixation threshold of the Copyright Act*"<sup>152</sup>. Más muestras de esta concepción amplia del derecho de reproducción sobre el programa de ordenador por los tribunales norteamericanos son: "*an automatic loading of copyrighted software into former's licensee's computers' random access memory (RAM) created copies of software that were sufficiently fixed for purposes of copyright infringement*"<sup>153</sup>; o la afirmación de que "*unauthorized loading of a software program onto the random access memory (RAM) of a computer constitutes an act of copying and thus of copyright infringement*"<sup>154</sup>.

A la vista de estos datos, queda claro que todos los actos de reproducción del programa que tienen lugar durante un proceso de descompilación de software están sujetos a la autorización del titular.

El alcance del derecho de reproducción sobre el programa de ordenador se ha erigido en estandarte de lo que varios autores han denominado el efecto expansivo del derecho de reproducción<sup>155</sup>. Ello se debe a una de las características específicas de esta clase de obras: su funcionalidad.

La funcionalidad intrínseca del programa produce una anomalía en el sistema de derecho de autor si la comparamos con otras obras literarias: En las obras literarias

---

<sup>152</sup> MAI Sys. Corp. v. Peak Computer, Inc., 991 F.2d 511, 26 U.S.P.Q.2d (BNA) 1458 (9th Cir. 1993), cert. dismissed, 510 U.S. (1994), pág. 1033. Un comentario sobre esta sentencia puede verse en A., JOHNSON-LAIRD, "Software Reverse Engineering ...", op.cit., pág. 892.

<sup>153</sup> Quantum Systems Integrators, Inc. v Sprint Nextel Corp., 338 Fed. Appx., 4th Cir. 2009, pág. 329.

<sup>154</sup> DocMagic, Inc. v. Ellie Mae, Inc., 745 F. Supp., 2d (N.D. Cal. 2010), pág. 1119.

<sup>155</sup> L., GUIBAULT, O., VAN DAALEN, *Unravelling the Myth around Open Source Licenses*, The Hague, TMC Asser Press, 2006, pág. 100; P., SAMUELSON, "Preliminary Thoughts on Copyright Reform", Utah Law Review, 2007, pág. 551 (563); J., SPOOR et al., *Auteursrecht: auteursrecht, naburige rechten en databankenrecht*, Deventer, Kluwer, 2005, pág. 594; M., SUCKER, "The Software Directive Between...", op.cit., pág. 15; H., HABERSTUMPF, "Der urheberrechtliche Schutz von Computerprogrammen" en M., LEHMANN, (ed.), *Rechtsschutz und Verwertung von Computerprogrammen*, Colonia, Otto Schmidt, 1993, pág. 133; S., LÓPEZ MAZA, *Límites del derecho de reproducción en el entorno digital*, Granada, Comares, 2009, pág. 10.

“tradicionales” el uso y disfrute de la obra no conlleva la realización de ningún acto de reproducción sujeto a autorización. En los programas de ordenador, sí<sup>156</sup>.

Por otra parte, esta amplitud del derecho de reproducción en los programas de ordenador sirve para compensar un fallo del sistema de derecho de autor sobre la protección de esta obra: la funcionalidad o el know-how contenido en el código. Este know-how solo puede conocerse si se analiza el código fuente del programa, y al distribuir los programas en código objeto, la descompilación se convierte en una valiosa herramienta para acceder a ese know-how del código fuente<sup>157</sup>.

Es decir, al impedir la descompilación del programa a través de la concesión de una facultad de reproducción tan amplia a sus titulares, el derecho de autor protege el corazón del programa de ordenador estableciendo una barrera de acceso a sus ideas, principios o know-how<sup>158</sup>.

Habrá que examinar si esta concepción del derecho de reproducción en sede de programas de ordenador tiene connotaciones más políticas que teleológicas, lo cual ha permitido que los derechohabientes de los programas tengan la facultad exclusiva de controlar su uso casi hasta el mínimo detalle<sup>159</sup>.

#### *B). El derecho de transformación.*

El derecho de transformación constituye otro de los derechos de explotación que el derecho de autor concede al derechohabiente del programa. Este derecho mantiene un estrecho vínculo con los derechos morales, porque la transformación de una obra tiene

---

<sup>156</sup> En el caso de las obras musicales o audiovisuales en formato digital sí es necesario realizar copias intermedias para disfrutar de la obra, sin embargo estos actos de reproducción están excepcionados a través de la Directiva 2001/29/CE del Parlamento Europeo y del Consejo, de 22 de mayo de 2001, relativa a la armonización de determinados aspectos de los derechos de autor y derechos afines a los derechos de autor en la sociedad de la información, DO L 167 de 22 de junio de 2001.

<sup>157</sup> J., BARTMANN, *Grenzen der Monopolisierung ...*, *op.cit.*, pág. 123; P., SAMUELSON *et al*, “A Manifiesto...”, *op.cit.*, pág. 2392; J.H., SPOOR, “Copyright Protection and Reverse Engineering of Software: Implementation and Effects of the EC Directive”, *University of Dayton Law Rev.*, vol. 19, 1994, pág. 1063 (1082).

<sup>158</sup> J.H., SPOOR, “Copyright Protection...”, *op.cit.*, pág. 1083.

<sup>159</sup> T., DREIER, “Rechtsschutz von Computerprogrammen: Die Richtlinie des Rates der EG vom 14. Mai 1991”, *CR* 7, 1991, pág. 577 (579); M., LEHMANN, “Freie Schnittstellen (“Interfaces”) und freier Zugang zu den Ideen (“Reverse Engineering”): Schranken des Urheberrechtsschutzes von Software”, *CR* 5, 1989, pág. 1057 (1062); A., VAN ROOIJEN, *The Software Interface ...*, *op.cit.*, pág. 69.

como consecuencia la aparición de una nueva obra, denominada “obra derivada”, susceptible de protección autónoma por el derecho de autor<sup>160</sup>.

Sin embargo, en el entorno digital donde se sitúan los programas de ordenador, la codificación o decodificación del software no altera su composición o su forma de expresión, por lo tanto, el resultado de la conversión es una reproducción, no una adaptación, traducción o transformación de la obra<sup>161</sup>.

Con lo cual, dependiendo de la posición que se adopte, las fricciones entre el derecho de transformación del programa y la descompilación variarán de manera radical.

El punto de partida de la problemática comienza con la exposición de qué ha de entenderse por derecho de transformación.

En la Unión Europea todos los ordenamientos nacionales de sus Estados Miembros reconocen el derecho de transformación, pero su emplazamiento varía. Para algunos ordenamientos, esta facultad es considerada como un derecho exclusivo único, y para otros, el derecho de transformación se encuentra dentro del derecho de reproducción<sup>162</sup>. En el caso de los ordenamientos objeto de estudio en este trabajo, todos se sitúan en el primer caso.

La Directiva sobre protección jurídica de los programas de ordenador, primer instrumento que armoniza el derecho de transformación a nivel comunitario sobre esta obra en particular, dispone que por transformación de un programa ha de entenderse “la traducción, adaptación, arreglo y cualquier otra transformación de un programa de ordenador y la reproducción de los resultados de tales actos<sup>163</sup>”.

En el caso de Estados Unidos, el apartado 2 del artículo 106 de la Copyright Act se refiere al derecho de transformación sobre la obra como “ (...) la facultad del derechohabiente a realizar y a autorizar la realización de obras derivadas basadas en la

---

<sup>160</sup> Sobre el acto y el derecho de transformación vid. J.C., ERDOZAIN LÓPEZ, “Comentarios al Libro Primero-Título II” en R., BERCOVITZ RODRÍGUEZ-CANO (coord.), *Comentarios a la Ley de Propiedad Intelectual*, Madrid, 3º ed., Tecnos, 2007, pág. 378.

<sup>161</sup> En P.B., HUGENHOLTZ, “Adapting Copyright to the Information Superhighway”, paper based on studies prepared for the European Commission (DG XIII and DG XV) in 1994 and 1995, sec. 2.1.

<sup>162</sup> M., VAN EECHOUD, P.B., HUGENHOLT, S. VAN, GOMPEL *et al*, *Harmonizing European Copyright Law. The Challenges of Better Lawmaking*, Alphen aan den Rijn, Kluwer Law International, 2009, pág. 83.

<sup>163</sup> Directiva Software, artículo 4.1, letra b).



obra sujeta a copyright<sup>164</sup>”. La remisión a la definición de la norma de obra derivada es obligatoria. El artículo 101 la define como “una obra que se basa en una o más obras preexistentes, como una traducción, un arreglo musical, una dramatización, “ficcionalización”, versión cinematográfica, grabación sonora, reproducción artística, resumen, condensación, o cualquier otra forma en que una obra puede ser reformulada, transformada o adaptada. Una obra que consiste en revisiones editoriales, anotaciones, elaboraciones u otras modificaciones que, en su conjunto, representan una obra original de un autor, es una obra derivada”<sup>165</sup>.

Aplicado a los programas de ordenador, se podría decir que la norma concede al titular por tanto la facultad de controlar cualquier clase de refundición, transformación o adaptación de los códigos del programa que den lugar a otro programa.

Ante esta amplitud de la facultad de transformación, la descompilación de un programa, cuya finalidad es obtener una “versión” aproximada del código fuente original, descodificarlo para acceder a las ideas del program, dos son las cuestiones que cabe plantearse:

- 1.Si aceptamos la premisa de que la codificación o decodificación de una obra digital no altera su composición o su forma de expresión, la descompilación no se vería afectada por el derecho de transformación sobre la obra, porque no da lugar en ningún caso a una obra derivada. Sin embargo el problema no termina aquí, porque aunque no haya obra derivada, sí habrá copia o fijación que permite la copia o comunicación de la obra, lo que nos remite nuevamente al derecho de reproducción.
- 2.Si no aceptamos la premisa anterior, y entendemos que la traducción y/o adaptación obtenida a través de la descompilación es una obra derivada del programa original, nos encontramos ante una infracción del derecho de autor

---

<sup>164</sup> Copyright Act, section 106: “Subject to sections 107 through 122, the owner of copyright under this title has the exclusive rights to do and to authorize any of the following: (2) to prepare derivative works based upon the copyrighted work”.

<sup>165</sup> Copyright Act, section 101: “A “derivative work” is a work based upon one or more preexisting works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which a work may be recast, transformed, or adapted. A work consisting of editorial revisions, annotations, elaborations, or other modifications, which, as a whole, represent an original work of authorship, is a “derivative work”.

sobre la obra si el titular del programa no ha prestado su consentimiento para realizar la descompilación.

Si seguimos este segundo razonamiento, que parece concordar con la postura tanto del legislador comunitario como del norteamericano, el problema se complica aún más porque habrá que adentrarse en la cuestión de si puede calificarse como obra derivada el resultado de la descompilación.

En el área del derecho de autor continental, aún enumerando la Directiva una serie de actos que están sujetos a la autorización del titular, al ser una norma de derecho derivado, corresponde al derecho nacional de cada Estado miembro establecer qué ha de entenderse por traducción, adaptación o transformación de un programa de ordenador, y conviene resaltar algunas diferencias importantes.

En el caso del Derecho inglés, la descompilación y desensamblamiento de un programa de ordenador constituyen una traducción, que es una modalidad del derecho de transformación (*adaptation right*)<sup>166</sup>.

En el Derecho alemán, para considerar la descompilación de un programa como un acto de transformación, sería necesario que la acción de compilar o descompilar fuese acompañada de un esfuerzo creativo, lo cual no tiene lugar porque tanto el proceso de compilación como de descompilación se realizan a través de otros programas de ordenador, no de personas. Además, la traducción del código objeto en código fuente no tiene como resultado un código fuente idéntico del que procede el código objeto descompilado. Por eso el código fuente resultado de una descompilación es denominado "pseudocódigo fuente"<sup>167</sup>.

En España, los actos de traducción, adaptación, arreglo y cualquier otra transformación de un programa, entran dentro del tenor del artículo 17 de la Ley de Propiedad Intelectual siempre que den lugar a una obra derivada original<sup>168</sup>. Así pues,

---

<sup>166</sup> CDPA, artículo 21: "Making an adaptation includes a version of the program in which it is converted into or out of a computer language or code or into a different computer language or code, otherwise than incidentally in the course of running the program".

<sup>167</sup> H., ULLRICH, E., KÖRNER, *Der International Softwarevertrag*, Heidelberg, Verl. Recht und Wirtschaft, 2006, pág. 71, n. 39.

<sup>168</sup> Vid. J.P., APARICIO VAQUERO, J., DELGADO ECHEVERRÍA, J., "Comentarios al Libro Primero-Título VII", en R., BERCOVITZ RODRÍGUEZ-CANO, (coord.) *Comentarios a la Ley de Propiedad Intelectual*, Madrid, 3º ed., Tecnos, 2007, pág. 1348

en el proceso de descompilación, en sintonía con la postura alemana, parece que ni se crearía una obra derivada ni tampoco se realizaría propiamente una traducción<sup>169</sup>.

Respecto de la postura norteamericana, si bien hay pocos comentarios doctrinales<sup>170</sup> y la jurisprudencia que ha tratado el tema no es muy reciente, la postura mantenida por algún tribunal ha sido la de apreciar infracción del Copyright cuando se han realizado actos de adaptación o transformación del código informático. Alguna de las decisiones es sorprendente porque el tribunal llegó a determinar que existía infracción de los derechos de Copyright por el mero hecho de utilizar un lenguaje de programación diferente al del programa anterior, para crear un programa nuevo, cuando "el mensaje que contienen ambos es el mismo"<sup>171</sup>.

Si recordamos lo dicho al hilo de cómo se produce el proceso de descompilación de un programa, con la participación indispensable de otro programa denominado "compilador", cabría afirmar que no se realiza una traducción del código al modo de una obra literaria del código objeto al código fuente, sino que lo que se obtiene es, en el mejor de los casos, una copia funcionalmente exacta a la del código fuente original, que permite a otro programador entender, una parte de la estructura y el modo de operar del programa descompilado<sup>172</sup>. Considerar como traducción el resultado de la una descompilación, sería como calificar de traducción de una novela que en la obra traducida la mayoría de los párrafos de la original fuesen cambiados de lugar por decisión de un programa de ordenador o robot, no de una persona, porque según los parámetros de optimización del programa, así mejoraría la "funcionalidad" de la novela. Ahora bien, si un lector tratase de leer ese resultado, resultaría incomprensible.

Es decir, el proceso de descompilación no genera una versión del código fuente como originalmente fue escrito, sino más bien una "reconstrucción plausible" de

---

<sup>169</sup> Ibidem, pág. 1349.

<sup>170</sup> Un comentario doctrinal puede verse en P. SAMUELSON, "Modyfing Copyrighted Software: Adjusting Copyright Doctrine to Accomodate Technology", *Jurimetrics Journal*, n°. 28, 1987-1988, pág. 179.

<sup>171</sup> *Pearl Systems, Inc. v. Competition Electronics, Inc.* 8 USPQ2d (BNA), 1988, S.D. Fla, 1988, pág. 1520 (WL 146047). En igual sentido, *Tradescape.com v. Shivaram*, 77 F.Supp.2d (S.D. N.Y. 1999), pág. 408.

<sup>172</sup> M., MELVILLE, D., NIMMER, *Nimmer on...op.cit.*, §13.05[D][4], pág. 13-230.18.

porciones del código original, de la forma en como pudo haberlo expresado su creador<sup>173</sup>.

El estudio y análisis a partir de esa aproximación al código fuente, de sus ideas y principios, es lo que permite al investigador deducir los detalles internos del diseño del programa, los elementos que le dan su funcionalidad única, así como aquellos que permiten la obtención de interoperabilidad con otros programas y otros dispositivos. Si los actos de traducción de un código fuente en un lenguaje de alto nivel a otro, por ejemplo de lenguaje C a lenguaje Python, o la adaptación que realiza un compilador al descompilar el código objeto de un programa, constituye una infracción de los derechos de autor del programa por vulnerar el derecho de transformación, se extiende el ámbito de protección del derecho de autor a las ideas y principios contenidos en el programa<sup>174</sup>.

### *3.3. La ingeniería inversa entre el impulso y el freno a la innovación.*

La gran duda que planea sobre la posibilidad de permitir la ingeniería inversa sobre el software y la descompilación en particular gira en torno a la innovación en el sector.

El resultado de un proceso de descompilación exitoso será la obtención de una copia funcionalmente muy similar a la del programa descompilado. La parte más valiosa de un programa de ordenador es precisamente esa: su comportamiento, que se contiene en las ideas y principios en que se inspira, para los cuales, el derecho de autor no proporciona protección. El derecho de autor protege la expresión de las ideas, no las ideas en sí mismas consideradas.

En relación a la interoperabilidad, el conflicto que legitimar la descompilación supone a los creadores de programas se sitúa en el coste de esta medida: aunque el estímulo de la interoperabilidad en la industria del software puede tener importantes

---

<sup>173</sup> J.E., COHEN, M.A., LEMLEY, "Patent Scope and Innovation in the Software Industry", Calif. L.Rev., nº 89, 2000, pág. 1 (6); P., SAMUELSON, S., SCOTCHMET, "The Law and Economics...", *op.cit.*, pág. 1603.

<sup>174</sup> Para algunos autores como VERSTRYNGE sin embargo, "acts of translation from one language into another - e.g., from Pascal to C- and adaptation of the target program code are not subjected to the exception of Article 6 and thus constitute copyright infringement". Vid. J.F., VERSTRYNGE, "Protecting Intellectual Property Rights within the New Pan-European Framework - Computer Software", en M., LEHMANN, C., TAPPER, *A Handbook of Software European Law*, Oxford, Clarendon Press, (vol. 1) 1993, (vol.2) 1995, pág. 1 (8).

efectos sobre la innovación, su coste pasaría por delimitar el ámbito de protección del derecho de autor sobre los programas. Esta dicotomía sigue vigente a día de hoy, aunque no es exclusiva de los programas de ordenador. La postura de permitir o no la descompilación tiene relación directa según se parta de la premisa de que con la imitación se fomenta la innovación o, por el contrario, se retrae.

Si se legitima la descompilación de un programa distribuido en una forma de expresión que no permite el acceso a sus ideas, como es el código objeto, cualquier competidor podría descompilar el programa y crear un nuevo programa sustituto del anterior que compitiese directamente con él en el mercado. Autorizar la descompilación con carácter general abriría la puerta al abuso de esta práctica y la competencia parasitaria.

Desde el punto de vista del derecho de autor, se estaría legitimando la piratería y con ello las empresas dedicadas a crear software dejarían de tener un aliciente para invertir en el desarrollo de nuevos programas. Por ejemplo, en un programa resultado de un proceso de ingeniería inversa será complicado acreditar la existencia de similitud sustancial con el programa anterior, de modo que ¿cómo se podrían contar el número de códigos duplicados?, y aún en el caso de que este ejercicio fuese posible, ¿qué porcentaje debiera tenerse en cuenta para probar que el segundo es una copia literal del primero en vez de una obra derivada o una obra nueva original? Con la prohibición absoluta de la ingeniería inversa se evita este problema<sup>175</sup>.

Aún en el caso de autorizarla para fines de lograr interoperabilidad llevaría al mismo resultado, porque la mayoría de información interoperable que un nuevo competidor pueda necesitar para desarrollar un nuevo programa se encuentra en los manuales del programa, en lo que el programa muestra en pantalla. Esa información es más que suficiente para crear un programa nuevo e interoperable, sin necesidad de descompilar el código.

---

<sup>175</sup> P., WATERS, P.G., LEONARD, “The lessons of recent EC and US Developments for Protection of Computer Software under Australian Law”, EIPR 13, 1991, pág. 124 (129). En opinión de estos autores, “prohibiting reverse engineering outright avoids these difficulties (literal and substantial copying) by turning the existing law on its head: the act of reverse engineering is the primary copyright offense, and the close similarity of the two programs is nothing more than circumstantial evidence of the act of reverse engineering having taken place.”

En última instancia, iría también en perjuicio de los consumidores, que dispondría de una cada vez menor variedad de programas entre los que elegir.

Estos son los argumentos que parte de la industria del sector informático como Intel, Apple, Digital Equipment Company, Microsoft y otros miembros de la Business Software Alliance (BSA, también conocida por las siglas SAGE – software alliance group), defienden desde principios de los años 90 del siglo pasado, cuando la cuestión de la permisividad de la descompilación por razones de interoperabilidad fue planteada ante tribunales y legisladores.

Sin embargo, las anteriores razones tienen contraargumentos de igual o incluso mayor peso. Otro sector de la industria informática, entre los que se encuentra Sun Microsystems, NCR, Storage Tech o Amadahl, representados por ECIS<sup>176</sup> defienden la legitimidad de la ingeniería inversa del software porque no sólo fomenta la creación de nuevos programas sino que estimulaba la creación de programas compatibles con los ya existentes<sup>177</sup>.

En primer lugar, existen razones que nada tienen que ver con la creación de programas sustitutos o clones para realizar un proceso de descompilación:

- Mejorar la portabilidad de datos,
- Detectar vulnerabilidades del programa que permiten corregir errores y mejorar la seguridad del programa.
- Facilitar la interoperabilidad entre programas.
- Mejorar la investigación y la competencia educativa de futuros ingenieros informáticos.

En contra del argumento de que autorizar la descompilación legitima la piratería, pervirtiendo el propio sistema de derecho de autor, algún autor como KARJALA, cuya postura comparto, defiende que en el ámbito del derecho de autor, al igual que no cabe amenazar los esfuerzos de terceras personas para crear una obra derivada colocando el

---

<sup>176</sup> ECIS, European Committee for Interoperable Solutions, es una asociación internacional fundada en 1989 cuyo objetivo es promover escenarios favorables para alcanzar soluciones interoperable en el sector de las TICs. Más información en: [www.ecis.eu](http://www.ecis.eu)

<sup>177</sup> Vid. J., ABBOT, "Reverse Engineering ...", *op.cit.*, pág. 12; A., JOHNSON-LAIRD, "Software Reverse...", *op.cit.*, pág. 845.

letrero de “piratería”, tampoco se puede calificar como piratería un proceso de duplicación mecánica o funcional<sup>178</sup>.

En relación al coste económico, a diferencia de los piratas, un programador independiente, no podrá permitirse comercializar su programa por debajo del precio de los demás, a diferencia de los piratas, porque para lograr un programa compatible o incluso similar a uno previo es necesaria una inversión considerable en investigación, es decir, tiempo y dinero. No sólo se pueden considerar los costes de realizar el proceso de descompilación, sino los de creación del nuevo programa. Por lo tanto, no cabe hablar de competencia parasitaria.

El creador de un programa interoperable creado con ayuda de un proceso de descompilación, tampoco podrá entrar a competir en el mercado de forma simultánea ni siquiera cercana a la del programa descompilado, porque este proceso de ingeniería inversa y la programación posterior son largos. El titular del programa originario dispone siempre de una ventaja temporal, a lo que se añaden las particularidades del sector de las TICS. La tecnología evoluciona cada día más rápido, y los ciclos de vida útil de los programas son cada vez menores<sup>179</sup>. Por lo tanto, resulta difícil calificar a la descompilación de competencia desleal<sup>180</sup>.

A pesar de todo, es probable que se produzcan situaciones en las que terceros, bajo la justificación de adquirir interoperabilidad, no hagan más que algún cambio meramente estético en el programa previo<sup>181</sup>. Sin embargo, la existencia de esta probabilidad no parece motivo suficiente para situar a la ingeniería inversa entre las causas que fomentan la piratería del software<sup>182</sup>. No ha de perderse de vista que el

---

<sup>178</sup> D., KARJALA, “Copyright, Computer Software, and the New Protectionism”, *Jurimetrics J.*, n° 28, 1987, pág. 33 (78).

<sup>179</sup> Vid. M.C., RILEY, “The Need for Software Innovation Policy”, *Journal of Telecommunications and High Technology Law*, 5(589), 2007, pág. 529; disponible en <http://ssrn.com/abstract=910045> (última consulta 1 de junio, 2013); M.A., LEMLEY, “Intellectual Property Rights and Standard-Setting Organizations”, *California Law Review*, Vol. 90, 2002, pág. 1889 (1897); disponible en: <http://ssrn.com/abstract=310122> y <http://dx.doi.org/10.2139/ssrn.310122> (última consulta 1 de junio, 2013).

<sup>180</sup> M.A., JACOBS, “Copyright and Compatibility, LaST Frontier Conference on Copyright Protection of Computer Software”, *Jurimetrics J.*, n° 30, 1989, pág. 91 (102).

<sup>181</sup> *Apple Computer Inc. v. Franklin Computer Corp.*, 714 F.2d, 1240 (3rd Cir. 1983); cert. Dismissed, 464 US, 1984, pág. 1033, al señalar que “the variations that did exist were minor, consisting merely of such things as deletion of reference to Apple or its copyright notice”.

<sup>182</sup> Según el Informe de 2010 de la BSA, asociación comercial sin ánimo de lucro creada para defender los objetivos del sector de software y de sus socios de hardware, la forma más común de piratería de software es la compra de una única copia de un programa para instalarlo en varios ordenadores, es decir,

término «piratería» designa generalmente la fabricación y la venta no autorizadas de contenidos protegidos por los derechos de autor o los derechos conexos. Esto incluye la producción de bienes falsificados, fabricados con el fin de asemejarse al producto original y confundir al consumidor, así como los productos clandestinos (bootleg) procedentes del registro ilegal de representaciones en directo (live)<sup>183</sup>. La creación de un programa interoperable a través de un proceso de ingeniería inversa, no pretende, a priori, falsificar el programa analizado, ni confundir al consumidor.

En relación al argumento de perjuicio a los consumidores, si se legitima la descompilación por razones de interoperabilidad, las posibilidades de los usuarios de tecnologías de mejorar sus condiciones de interoperabilidad horizontal se verán incrementadas, no disminuídas, porque dispondrán de más opciones entre las que escoger. Consecuencia de lo anterior, el coste de portabilidad entre programas, característico de una situación de intraoperabilidad, también disminuirá.

Cuando la innovación y la mejora del software distribuido en código objeto depende del acceso que terceros tengan a las ideas de programas existentes; cuando para conocer esas ideas no existe otra vía que la de la descompilación, la cual, según se ha visto conlleva la infracción sobre el derecho de autor sobre el programa, parece que nos encontramos ante un desequilibrio del sistema de protección<sup>184</sup>.

#### **4. La información interoperable contenida en las interfaces del programa.**

##### *4.1. El concepto de interfaz desde la perspectiva técnica.*

Desde un punto de vista técnico una interfaz es el límite natural entre un sistema dado y otro sistema, o entre dos sistemas, o entre dos partes de un mismo sistema, a

---

la copia no autorizada. Vid. "2010 Piracy Study", 8th Annual BSA Global Software, mayo 2011, pág. 2, disponible en: <http://portal.bsa.org/globalpiracy2010/>, (última consulta 1 de junio, 2013).

<sup>183</sup> Vid. UNESCO: <http://portal.unesco.org/culture/es/ev>, (última consulta 1 de junio, 2013).

<sup>184</sup> D., BAINBRIDGE, *Software Copyright Law*, London, 4ªed., Butterworlds, 1999, pág. 138. Vid. el caso *Bonito Boats, Inc. v Thunder Craft Boats, Inc.* US 141 (1989), pág. 489, letra (c): "(...) *reverse engineering* (...) [t]his is one of the rights granted to the federal patent holder, but has never been part of state protection under the law of unfair competition or trade secrets. The study and recomposition of unpatented articles available to the public at large may lead to significant advances in technology and design. Moreover, the threat of reverse engineering of unpatented articles creates a significant spur to the achievement of the rigorous standards of patentability established by Congress".



través de las cuales se transmite información<sup>185</sup>. Están presentes en todos los elementos de un ordenador electrónico: entre los elementos estructurales (hardware), entre los elementos estructurales y los elementos funcionales (software), y entre los distintos elementos funcionales o programas que forman parte del ordenador.

Siguiendo lo dispuesto en el estándar internacional de la IEEE número 1003, “Portable Operating System Interface (POSIX)”<sup>186</sup>, las interfaces de software pueden clasificarse en cuatro grupos:

Interfaces de programas de aplicación (Application programming interface o API). Consisten en un conjunto de estándares del programa que incluye procesos de interrupción, llamadas, funciones y formatos de datos que pueden ser utilizados por un programa de aplicación para acceder a servicios de red, dispositivos o sistemas operativos<sup>187</sup>. Son fundamentales en los sistemas operativos ya que permiten a los programadores desarrollar programas de aplicación que funcionen en el entorno del sistema operativo. Pueden ser interfaces de sistema operativo a sistema operativo, interfaces de aplicación a sistema operativo, como el caso del programa Adobe Photoshop, o interfaces de aplicación a aplicación, como sucede con el programa iTunes.

Interfaces de usuario (User interface). Es una interfaz que permite el paso de información entre un usuario humano y los componentes estructurales y lógicos del sistema computacional<sup>188</sup>. Esta clase de interfaces no suponen una interacción de software a software o de software a dispositivo, por lo que, aunque tienen interés desde el punto de vista de los derechos de autor, no afectan a la interoperabilidad.

---

<sup>185</sup> La definición original en inglés dispone: “a common boundary between a considered system and another system, or between parts of a system, through which information is conveyed.” 488.1-2003 IEEE Standard for Higher Performance Protocol for the Standard Digital Interface for Programmable Instrumentation.

<sup>186</sup> POSIX.1-2008 es simultáneamente un estándar del IEEE Std 1003.1™-2008 y del “Open Group Technical Standard Base Specifications, Issue 7”.

<sup>187</sup> La definición original en inglés las define como: “A set of standard software interrupts, calls, functions, and data formats that can be used by an application program to access network services, devices, or operating systems”. Vid. supra., POSIX. 1-2008.

<sup>188</sup> La definición original en inglés: “An interface that enables information to be passed between a human user and hardware or software components of a computer system”. *Ibidem*.

Interfaces de datos (Data interface). Son las responsables de almacenar y recuperar datos en un formato específico<sup>189</sup>. Ejemplos de esta clase de interfaces son lo que se encuentran en los formatos JPEG, HTML, PDF, o MP4.

Interfaces de comunicación (Communications interface). La interfaz de serie de un dispositivo que permite el intercambio de información tanto de carácter físico como lógico, entre los distintos dispositivos de un sistema o de otros sistemas que están organizados jerárquicamente igual a nivel funcional<sup>190</sup>. Dos sencillos ejemplos de esta clase son las interfaces de los programas de correo electrónico (POP3, IMAP o HTTP/SMTP) y la transferencia de sitios en la World Wide Web (WWW) de Internet.

De modo que una interfaz describe la conexión de los enlaces de comunicación, tanto en el caso de conexiones mecánicas como de conexiones basadas en señales de carácter físico y funcional<sup>191</sup>. El modo en que la comunicación se produce depende del contenido de la especificación de cada interfaz. Por lo tanto, es la especificación la que determina la forma de comunicación. Su importancia llega al punto de que sin la información que contienen, un desarrollador o programador informático no podrá crear un programa que sea interoperable o capaz de comunicarse con programas ya existentes en el mercado<sup>192</sup>.

Durante la elaboración de un programa, las interfaces están presentes a distintos niveles de abstracción. Se utilizan a nivel bajo, en componentes físicos, como el conector de la Ethernet y a nivel alto o lógico, como las interfaces de datos entre dos procesadores de textos. Pero para que dos programas sean interoperables, ambos deben adherirse exactamente a la mismas especificaciones de la interfaz. Deben ser idénticas o literales.

La especificación de una interfaz es la descripción de los requisitos esenciales de funcionamiento, comportamiento y diseño y facilita la delimitación entre dos o más

---

<sup>189</sup> La definición original en inglés: "The ones responsible for storing and retrieving data in a specific format".

<sup>190</sup> La definición original en inglés: "Serial interface of a device that allows exchange of (physical and logical) information among devices of the same or different functional levels in a hierarchical system".

<sup>191</sup> C., SEVERANCE, "Posix: A model for future computing", Computer 32, Enero, 1992, pág. 131 (132).

<sup>192</sup> T., VINJE, "Magill: its impact on the information technology industry", EIPR 14, 1992, pág. 397 (401).

elementos del sistema<sup>193</sup>. Esta es la parte de la interfaz que debe mostrarse a otro programador.

Separar e implementar las especificaciones de la interfaz es una forma de ocultar información, lo cual constituye uno de los principios fundamentales de la Ingeniería del software<sup>194</sup>. Esto que a priori puede parecer inverosímil, es fácil de comprender con una sencilla comparación: comparadas a un lenguaje que dos personas estudian a la vez, las especificaciones se comportan como la sintaxis y el vocabulario: aunque ambas personas aprendan el lenguaje de forma simultánea, el uso que hagan de lo aprendido no tiene por qué ser idéntico<sup>195</sup>.

Las interfaces de software son por tanto abstracciones de los servicios para los que están pensados. Pero solo son eficaces en la invocación de esos servicios cuando se aplican en el código fuente y el código objeto. Las interfaces por tanto, condicionan en parte la creatividad del diseñador del programa, porque el programador ha de adaptarse exactamente a lo dispuesto en ellas para que el programa pueda prestar los servicios configurados en la interfaz<sup>196</sup>.

#### *4.2. El concepto de interfaz desde la perspectiva jurídica: ¿idea, expresión o método de operación?*

Algunos de los primeros autores que tomaron parte en el debate jurídico sobre la interoperabilidad, definieron las interfaces informáticas como “aquellas características funcionales de un elemento que le permite interactuar con otros dentro de un sistema

---

<sup>193</sup> La definición original dice: “the description of essential functional, performance, and design requirements and constraints at a common boundary between two or more system elements. 1220-2005 IEEE Standard for Application and Management of the Systems Engineering Process.

<sup>194</sup> Esta práctica se denomina ofuscación del código, y tiene repercusiones importantes a efectos de lograr la interoperabilidad. Por ejemplo, es el caso del estándar OOXML de Microsoft. Los desarrolladores de la suite informática Open Office intentaron durante años desvelar este estándar para que su programa pudiese interoperar con archivos de la suite informática de Microsoft, Microsoft Office. Más información en C., SOUZA et al., “Sometimes you need to see through walls – a field study of application programming interfaces”, Computer Supported Cooperative Work (CSCW), ACM Press 10, (2004), pág. 63; H-H., JONAS, “Obfuscating and Software Reverse Engineering in Practice”, MAS-IP Diploma Papers & Research Reports, nº 13, 2008; disponible en: [www.bepress.com/ndsip/reports/art13](http://www.bepress.com/ndsip/reports/art13) (última consulta 1 de junio, 2013).

<sup>195</sup> H., HOVENKAMP, “Standards Ownership and Competition Policy”, Boston College Law Review, 48, 2007, pág. 87 (88).

<sup>196</sup> P., SAMUELSON, “Are patents on interfaces impeding interoperability?”, Minn. L. Rev. 93, 2008, pág. 1943 (1949); disponible en: <http://scholarship.law.berkeley.edu/facpubs/1712>, (última consulta 1 de junio, 2013)

informático”, y emplearon el siguiente símil para explicar su comportamiento: “An interface is an electrical plug and wall socket. If the plug has a different shape from the socket, the plug simply will not fit into the socket”<sup>197</sup>. Sin embargo, como se ha afirmado posteriormente, nadie hubiese imaginado jamás que la protección de los derechos de autor llegase a los diseños de enchufes y de tomas de corriente, porque ninguno de ellos constituye ni una obra literaria ni una obra artística. Es más, tampoco la protección a través de la figura del secreto industrial tiene sentido sobre las interfaces porque si se pueden comparar por analogía con los enchufes y tomas de corriente existentes, sus diseños no tienen ningún elemento innovador. Sería como los diseños de máquinas que si bien alguna vez pudieron haber constituido materia patentables, de haberse concedido hoy día, la patente habría ya expirado<sup>198</sup>.

El problema que plantean estas características funcionales en el marco del derecho de autor es si las mismas pueden calificarse como ideas o como métodos de operación, lo que significaría que no están protegidas por el derecho de autor, o forman parte de la expresión del programa y por tanto, del objeto de protección del derecho de exclusiva.

Desde el punto de vista de los titulares del derecho de autor sobre el programa, esta calificación es vital para adoptar una estrategia competitiva en el mercado. Normalmente los creadores de programas cuentan con incentivos para hacer públicas las especificaciones de sus interfaces y fomentar su uso de manera que tanto usuarios como otras empresas dedicadas no solo a programación sino también las dedicadas a la elaboración de dispositivos periféricos, puedan producir nuevos programas o dispositivos y adaptar los existentes para interoperar con los programas de aquéllas. Sin embargo, otros creadores prefieren mantener sus interfaces en secreto a fin de intentar

---

<sup>197</sup> J., BAND, M., KATOH, *Interfaces on trial: ..., op.cit.*, pág. 7. BAINBRIDGE, las define como aquellas partes del programa que incluyen la forma y la estructura de los archivos que el programa crea y usa, las que controlan la asignación y gestión de la memoria, su interacción con los dispositivos periféricos y los formatos de entrada y de salida. También el modo en que el programa interactúa con el usuario es otra modalidad de interfaz. Vid. D. BAINBRIDGE, *Software Copyright ..., op.cit.*, pág. 138.

<sup>198</sup> Vid., P., SAMUELSON, “The Strange Odyssey of Software Interfaces and Intellectual Property Law”, UC Berkeley Public Law Research Paper No. 1323818, (2008); disponible en SSRN: <http://ssrn.com/abstract=1323818> ó <http://dx.doi.org/10.2139/ssrn.1323818> (última consulta 31 de enero 2013).

controlar la clase de software y de dispositivos periféricos que puedan salir al mercado para alcanzar la interoperabilidad con sus programas<sup>199</sup>.

En el marco normativo internacional, el artículo 9.2 del Acuerdo sobre los ADPIC y el artículo 2 del Tratado WCT de la OMPI declaran que “la protección del derecho de autor abarcará las expresiones pero no las ideas, procedimientos, métodos de operación o conceptos matemáticos en sí”.

Esta declaración está recogida en la Copyright Act norteamericana en su artículo 102, sin embargo, en el derecho comunitario, la Directiva sobre protección jurídica de programas de ordenador no contiene una declaración de tal clase. A pesar de que en su Considerando 11 dice que “las ideas y principios implícitos en los elementos del programa, incluidas las de sus interfaces, no pueden acogerse a la protección de los derechos de autor con arreglo a la presente Directiva”, esta manifestación no ha sido de gran ayuda para los tribunales, porque la configuración de la protección establecida en la Directiva, como se verá en el capítulo tercero, no impide calificar a las interfaces como expresión del programa por dos razones:

Si los programas de ordenador son protegidos por el derecho autor como obras literarias, entonces, el ámbito primario de esta protección se circunscribe al código escrito del programa, más que a su funcionalidad o comportamiento<sup>200</sup>. Consecuentemente, al ser las interfaces del programa parte de este código escrito podrían constituir también parte del objeto de protección. Asimismo, la protección de la funcionalidad o del comportamiento también podría ser susceptible de protección por patente<sup>201</sup>.

Por otra parte, las interfaces de un programa, al contrario del comportamiento de éste, son unas informaciones muy concretas y detalladas que pueden ser representadas por escrito. La esencia de la propia interfaz informática radica en su concreción, gracias

---

<sup>199</sup> En relación con la industria del videojuego vid. D., SHEFF, *Game Over: how Nintendo zapped an american industry, captured your dollars, and enslaved your children*, New York, Random House, 1993, pág. 286-291.

<sup>200</sup> P., SAMUELSON *et al*, “A Manifesto...”, *op.cit.* pág. 2360.

<sup>201</sup> Vid. R., BAKELS, P.B., HUGENHOLTZ, “The Patentability of Computer Programs: Discussion of European-Level Legislation in the Field of Patents for Software”, European Parliament, 2002. Este aspecto queda claramente recogido en el Considerando Undécimo de la Directiva, afirmando “solo se protege la expresión del programa de ordenador y las ideas y principios implícitos en los elementos del programa, incluidas las de sus interfaces, no pueden acogerse a la protección de los derechos de autor con arreglo a la presente Directiva”.

a la cual la máquina (el ordenador) es capaz de ejecutar la interconexión y el intercambio de información entre distintos programas o entre un programa y un componente físico del mismo. Por lo mismo, son expresiones concretas que forman parte del código del programa. Desde esta perspectiva, podrían calificarse como expresión del programa y no ideas.

La interpretación de la doctrina de la idea y la expresión es la herramienta a la que los tribunales han recurrido para decidir sobre la naturaleza de esta parte del programa, y como se verá en los siguientes capítulos, la jurisprudencia norteamericana ha sido bastante más clara y rotunda que la europea a la hora de posicionarse.

Si consideramos a las interfaces ya no como ideas sino como métodos de operación o estándares, también excluidas de protección por el derecho de autor según los principales tratados internacionales en la materia, nos encontramos con otro escollo. El derecho de autor, como sistema de protección de los programas de ordenador, estimula la diversidad de la expresión a través del requisito de la originalidad. La consideración de las interfaces como métodos de operación o estándares promueve justo lo contrario: la normalización de la expresión.

Diversidad en el sentido de originalidad en la expresión y estandarización de la expresión son mutuamente excluyentes. La necesidad de usar expresiones idénticas en interés de la interoperabilidad y de la estandarización va en contra del epicentro del derecho de autor, la originalidad de la expresión.

Varios comentaristas han argumentado esta razón para defender que la propiedad intelectual no debería proteger los elementos de programa que controlan las interfaces entre los componentes. De esto modo se permitiría un acceso ilimitado a tales componentes por competidores<sup>202</sup>.

---

<sup>202</sup> Vid. J., FARRELL, "Arguments for Weaker IP protection in Network Industries, in Standards Policy for Information Infrastructure", en B., KAHIN, J., ABBATE (eds.), *Standards Policy for Information Infrastructure*, Cambridge, MA and London, The MIT Press, 1995, pág. 369. En igual sentido, P., MENELL, "An analysis of the Scope of Copyright Protection for Application Programs", *Stand. L. Rev.* 41, 1989, pág. 1045, (1066); P., MENELL, "The Challenges of Reforming Intellectual Property Protection for Computer Software", *Colum. L. Rev.* 94, 1994, pág. 2644, (2652).

#### 4.3. Protección sobre las interfaces y derecho a acceder a la información interoperable: ¿un posible nuevo conflicto para el derecho comunitario?

##### A) Planteamiento de la cuestión: el conflicto entre dos derechos fundamentales.

El acceso a la información y al saber son las principales bases de la creación y el desarrollo de la sociedad de la información y de las redes electrónicas. Por supuesto hay que tener presente hasta qué punto la protección de los autores resulta determinante en la circulación del saber y de la cultura y, en consecuencia, habrá que considerar en qué medida el derecho del público a la información pone en jaque dicha protección. Sin embargo, no sólo hay que tener en cuenta los intereses legítimos de los autores, sino también los de los usuarios y la sociedad en su conjunto.

La delimitación de la relación entre derecho de autor y derecho a la libertad de expresión es la que proporciona la clave para ver cuál es conflicto que puede producirse entre estos dos derechos en relación a las interfaces de software.

Desde una perspectiva general, los conceptos de derecho de autor y libertad de expresión en muchas ocasiones han sido percibidos como complementarios o un binomio armónico<sup>203</sup>, pero en otras ocasiones se han visto enfrentados. Cuando el derecho de autor es analizado desde una perspectiva más económica que jurídica, se ha descrito como un medio para asegurar una mejor circulación de las ideas, o como declara el artículo 1 de la Constitución norteamericana, una herramienta para “promover el progreso científico y de las artes útiles”, o incluso, como “el motor de la libertad de expresión”<sup>204</sup>. Pero cuando el derecho de autor como derecho de propiedad especial es observado desde la perspectiva de los derechos fundamentales, surgen problemas. El punto de partida de esta observación pasa por elevar el derecho de autor a rango de derecho fundamental, y no está exenta de dificultades.

Por un lado, son pocos los países que garantizan expresamente el derecho de autor en su Constitución. Es el caso de Estados Unidos, Suecia y México. En este

---

<sup>203</sup> P., SAMUELSON, “Copyright and Freedom of Expression in Historical Perspective”, *J. Intell. Prop. L.* 10, 2002-2003, pág. 319 (319); D., PEEPERKORN, “Malesherbes on the Rights and Duties of Authors”, en J.J.C., KABEL, G.J.H.M, MOM, (eds.), *Intellectual Property and Information Law, Essays in Honour of H. Cohen Jehoram*, Information Law Series 6, The Hague, London, Kluwer Law International 1998, pág. 230.

<sup>204</sup> *Harper & Row Publishers, Inc. v. Nation Enterprises*, 471 U.S., 1985, pág. 539 (558).

último, la inclusión confiere al derecho de autor el carácter de derecho fundamental. En Estados Unidos, sin embargo, la jurisprudencia se ha encargado de aclarar que esta inclusión no significa que el “copyright” sea un derecho fundamental<sup>205</sup>.

En el continente europeo, es Alemania la que cuenta con prolífica jurisprudencia y comentarios al respecto, que pueden resumirse en la idea de que el núcleo del derecho de autor garantizado por la Constitución consiste en la potestad del autor para obtener un aprovechamiento pecuniario de su obra según las reglas del derecho privado, y en la libertad de disponer de ella. También se tiende a admitir, aunque el Tribunal Constitucional no se haya pronunciado expresamente sobre esta cuestión, que el derecho moral (*Urheberpersönlichkeitsrechte*) está amparado por el artículo 1 de la Constitución, el cual garantiza el carácter inviolable de la dignidad de la persona y el derecho de cada individuo al desarrollo de su personalidad<sup>206</sup>.

En el derecho comunitario no existía hasta hace poco una norma de derecho primario que garantizase el derecho de autor a nivel constitucional, por lo que se recurría a algún instrumento internacional que sí lo sitúa entre los derechos fundamentales, como la Carta de los Derechos Fundamentales de la Unión Europea de 18 de diciembre de 2000, en su artículo 17(2)<sup>207</sup>. Esta situación ha cambiado con la entrada en vigor del Tratado de Lisboa el 1 de diciembre de 2009, en virtud de cuyo artículo 6.1<sup>208</sup>, la Carta de Derechos Fundamentales ha adquirido rango de derecho primario de la Unión.

---

<sup>205</sup> Vid. *Darden v. Peters*, 488 F.3d (4th Cir., 2007), pág. 277.

<sup>206</sup> Vid, entre otros: F., LEINEMANN, *Die Sozialbindung des “Geistigen Eigentums”*, Nomos, Baden, 1998, pág. 52-58; F., FECHNER, *Geistiges Eigentum und Verfassung*, Tübingen, Mohr Siebeck, 1999. Entre la jurisprudencia: *Kirchen- und Schulgebrauch*, sentencia del Tribunal Constitucional Federal Alemán (BGH) de 7 de Julio de 1971, GRUR 1972, pág. 481; *Kirchenmusik*, sentencia del Tribunal Constitucional Federal Alemán (BGH) de 25 de Octubre de 1978, GRUR 1980, pág. 44.

<sup>207</sup> Art. 17(2): “(...) se protege la propiedad intelectual”. Carta de los Derechos Fundamentales de la Unión Europea, DOUE de 18 de diciembre de 2000, C364/01. Otros instrumentos internacionales que reconocen el rango de derecho fundamental al derecho de autor son: La Declaración Universal de los Derechos Humanos de 1948 (art. 27.1), el Pacto Internacional de Nueva York, de 19 de diciembre de 1966, sobre los Derechos Económicos, Sociales y Culturales (art. 15.1), el Convenio Europeo para la Protección de los Derechos Humanos y las Libertades Fundamentales, de 4 de noviembre de 1950, porque su Protocolo primero garantiza la propiedad.

<sup>208</sup> El artículo 6.1 del TUE establece: “La Unión reconoce los derechos, libertades y principios enunciados en la Carta de los Derechos Fundamentales de la Unión Europea de 7 de diciembre de 2000, tal como fue adaptada el 12 de diciembre de 2007 en Estrasburgo, la cual tendrá el mismo valor jurídico que los Tratados.” Ni el Tratado de la Unión anterior, ni el Convenio Europeo para la protección de los Derechos Humanos y Libertades Fundamentales, otorgaba a los derechos de autor un fundamento constitucional explícito.



Ahora bien, desde el punto de vista estrictamente jurídico, la inclusión del derecho de autor como derecho fundamental no lleva espontáneamente a la idea de un control sistemático de proporcionalidad, porque nadie sostiene que se deba ponderar, por principio, el derecho de propiedad con los demás derechos fundamentales que podrían paralizar su ejercicio<sup>209</sup>.

El derecho a la libertad de expresión suele estar garantizado por las constituciones y por numerosos instrumentos internacionales, entre los que aquí destaca el Convenio Europeo para la protección de los Derechos Humanos y Libertades Fundamentales, que además de tener efecto directo para los ciudadanos de los Estados signatarios, se ha convertido en derecho primario de la Unión Europea siguiendo lo dispuesto en el artículo 6.2 del Tratado.

El artículo 10 del CEDH configura el derecho fundamental a la libertad de expresión<sup>210</sup>. Esta norma proveniente de un tratado internacional, ha adquirido rango de derecho primario en la Unión Europea desde que, con la entrada en vigor del Tratado de Lisboa el 1 de diciembre de 2009, se abrió la puerta a la adhesión de la Unión al Convenio Europeo para la protección de los Derechos Humanos y Libertades Fundamentales<sup>211</sup>.

---

<sup>209</sup> A., LUCAS, “Derecho de autor y Libertad de Expresión”, en Jornadas de Estudios ALAI 2006, Barcelona, Huygens, 2008, pág. 63.

<sup>210</sup> Artículo 10 CEDH: 1. Toda persona tiene derecho a la libertad de expresión. Este derecho comprende la libertad de opinión y la libertad de recibir o de comunicar informaciones o ideas, sin que pueda haber injerencia de autoridades públicas y sin consideración de fronteras. El presente artículo no impide que los Estados sometan a las empresas de radiodifusión, de cinematografía o de televisión a un régimen de autorización previa.

<sup>211</sup> El artículo 6.2 del TUE establece que: “(...) 2. La Unión se adherirá al Convenio Europeo para la protección de los Derechos Humanos y las Libertades Fundamentales. Esta adhesión no modificará las competencias de la Unión que se definen en los Tratados. 3. Los derechos fundamentales que garantiza el Convenio Europeo para la Protección de los Derechos Humanos y de las Libertades Fundamentales y los que son fruto de las tradiciones constitucionales comunes a los Estados miembros formarán parte del Derecho de la Unión como principios generales”. En julio de 2010, se abrieron las negociaciones de adhesión, que siguen avanzando lentamente; vid. Council of Europe, press release 545(2010) “European Commission and Council of Europe kick off joint talks on EU’s accession to the Convention on Human Rights”, 2010; Europa, IP/10/291, “European Commission acts to bolster the EU’s system of protecting fundamental rights”, 17 de marzo de 2010. Un comentario sobre la adhesión y sus consecuencias en: V., REDING, “Filling the Missing Link in Fundamental Rights Protection: The European Union’s Accession to the European Convention on Human Rights”, Comité de Ministros del Consejo de Europa, Strasbourg, 7 de julio, (2010); disponible en <http://www.coe.int> (última consulta 1 de junio, 2013); Actualmente las negociaciones se encuentran bloqueadas por las exigencias planteadas por el Reino Unido. [http://www.euroefe.efe.com/1311\\_noticias/1442585\\_piden-desbloquear-el-proceso-de-adhesion-de-la-ue-al-convenio-de-derechos-humanos.html](http://www.euroefe.efe.com/1311_noticias/1442585_piden-desbloquear-el-proceso-de-adhesion-de-la-ue-al-convenio-de-derechos-humanos.html) (última consulta 1 de junio, 2013).

El valor jurídico del artículo 10 del CEDH es fundamental ya que en el mismo se define el derecho a la libertad de expresión y de información, el derecho a mantener opiniones, así como a impartir, distribuir y recibir información sin interferencia alguna por parte de los Estados.

Sin embargo, los contornos de este derecho son bastante imprecisos. La libertad de expresión es un principio axiomático de las sociedades abiertas, que incluye cualquier manifestación del lenguaje humano, por tanto, también la libertad de creación literaria y artística<sup>212</sup>.

A la hora de relacionar este derecho con el derecho de autor, es necesario tener en cuenta sus dos facetas, libertad de expresión en sentido estricto, y derecho del público a acceder a la información, porque no pueden colocarse ambas en el mismo plano.

No puede existir conflicto entre la libertad de expresión en sentido estricto y el derecho de autor porque ello supondría que el sistema de derecho de autor se traicionaría a sí mismo. El ejercicio de la libertad de expresión, en su modalidad de creación literaria y artística produce el objeto sobre el que se reconoce el derecho de autor, y es en este plano donde se recoge la afirmación antes citada de que libertad de expresión y derecho de autor son un binomio armónico.

Con el derecho del público a acceder a la información no debería a priori existir ningún problema, porque ambos derechos tienen objetos distintos: para el derecho de autor, es la obra, y para el segundo, la información que contiene. Sin embargo, la consideración de las interfaces del programa como parte de su expresión, y por tanto protegidas por el derecho de autor, sí puede plantear un choque de intereses, debido en parte, a que la protección del derecho de autor sobre el programa no se centra únicamente en la obra en sí misma considerada, sino que se extiende también a la información que contiene.

Es más, si tenemos en cuenta que el Tribunal de Derechos Humanos ha reiterado que la Convención es un instrumento vivo, que ha de ser interpretado a la luz de las

---

<sup>212</sup> Art. 19.2 del Pacto Internacional de Derechos Civiles y Políticos de 1966: “Toda persona tiene derecho a la libertad de expresión; este derecho comprende la libertad de buscar, recibir y difundir informaciones e ideas de toda índole, sin consideración de fronteras, ya sea oralmente, por escrito o en forma impresa o artística, o por cualquier otro procedimiento de su elección”.

condiciones del momento<sup>213</sup>, la consideración del código informático como información al amparo del artículo 10 CEDH no puede descartarse a priori, y ya varios autores se han planteado la cuestión de si el sistema de protección establecido por la Directiva sobre protección de los programas de ordenador colisiona con este artículo<sup>214</sup>.

*B) La propiedad sobre la información interoperable frente al derecho del público a acceder a la información.*

Si el código informático en general y las interfaces en particular son consideradas obras literarias, y por tanto, la expresión humana de una información original, nada hay que objetar a la protección del derecho de autor. El conflicto con la faceta del derecho fundamental a la libertad de expresión, el derecho del público a acceder a la información, surge cuando esa información puede calificarse como materia de interés general porque la misma es indispensable para que se produzca la comunicación e interacción funcional entre distintos programas y dispositivos.

El derecho comunitario nunca había tenido que plantearse la consideración del código informático como información hasta la entrada en vigor del Tratado de Lisboa, porque la tradición jurídica continental no había reflejado consideraciones de tipo utilitario en relación a la política de la información<sup>215</sup>. El fundamento constitucional del derecho de autor europeo descansa sobre principios de derecho natural más que en razones de carácter utilitario<sup>216</sup>.

En Estados Unidos, sin embargo, sí se ha producido un considerable debate doctrinal, debido a la mención al derecho de autor contenida en su Constitución. Dos son las posiciones enfrentadas.

---

<sup>213</sup> Vid. entre otras, *Airey v. Ireland*, sentencia del 9 de octubre de 1979, Serie A, número 32, pág. 14-16, párr. 26; *Loizidou v. Turkey*, sentencia del 23 de marzo de 1995, Serie A, número 310, pág. 26-27, párr. 71.

<sup>214</sup> B., ROTENBERG, “The European Regulation of Communications Software: Building a “Platform” for Freely Interoperable Digital Expression?”, en *International Journal of Communications Law and Policy*, nº 8, Invierno, 2003/2004, pág. 1; R.E., HERR, “The Right to Receive Information Under Article 10 of the ECHR: An Investigation from a Copyright Perspective”, 2011) *Tidskrift utgiven av Juridiska Föreningen i Finland (JFT)*, (prepublicación). Disponible en SSRN: <http://ssrn.com/abstract=1787085> (última consulta 1 de junio, 2013).

<sup>215</sup> P.B., HUGENHOLTZ, “Copyright and Freedom of Expression in Europe”, en R., COOPER DREYFUSS, H., FIRST, D., LEENHEER ZIMMERMAN, (eds.), *Expanding the Boundaries of Intellectual Property*, Oxford University Press, Oxford, 2001, pág. 2.

<sup>216</sup> G., DAVIES, *Copyright and ...*, *op.cit.*, pág. 7 – 13.

Por un lado, los defensores de tratar al código informático como información, alegan que si el software es protegido como obra literaria, es por lo tanto expresión. De manera que el acto de distribuir el software en la forma de código, fuente u objeto, es similar al acto de hablar en un lenguaje hasta el punto de que ambos actos transmiten información con un propósito<sup>217</sup>. Un segundo argumento que apoya esta visión es la analogía. Si las fórmulas matemáticas o las composiciones musicales pueden ampararse bajo la protección constitucional de la libertad de expresión, igualmente el código informático<sup>218</sup>.

La otra postura, defendida sobretod por PLOTKIN, sostiene que, el software como bien funcional es un instrumento, un medio para conseguir un fin, al igual que un dispositivo mecánico o electrónico, y por lo mismo, no merece protección constitucional como información<sup>219</sup>.

La información interoperable contenida en las interfaces del programa puede verse afectada por el artículo 10 del CEDH desde la siguiente perspectiva.

Partiendo de lo dispuesto en el caso *Autronic*<sup>220</sup>, el Tribunal Europeo de Derecho Humanos reconoció que el artículo 10 ha de ser aplicado no solo al contenido de la información sino también a los medios de transmisión o de recepción de ésta, ya que cualquier clase de restricción impuesta a los medios interfiere necesariamente con el derecho a recibir e impartir información. ("art. 10 applies not only to the content of information but also to the means of transmission or reception since any restriction imposed on the means necessarily interferes with the right to receive and impart information".) Si concebimos a las interfaces del programa como los medios necesarios para recibir e intercambiar información y para utilizar la información así intercambiada, siguiendo la definición de interoperabilidad proporcionada por la Directiva de Software,

---

<sup>217</sup> L., TIEN, "Publishing Software as a Speech Act", *Berkely Technology Law Journal*, nº 15, (2000), pág. 629, disponible en: <http://www.law.berkeley.edu/journals/btlj/articles/vol15/tien/tien.html>, (última consulta 1 de junio, 2013); G., COLEMAN, "Code is Speech: Legal Tinkering, Expertise, and Protest among Free and Open Source Software Developers", *Cultural Anthropology*, vol. 24, núm. 3, pág. 420 (420).

<sup>218</sup> L.A., HOLLAAR, *Legal Protection of Digital Information*, 2002, online, disponible en: <http://digital-law-online.info/lpdi1.0/treatise50.html>, (última consulta 1 de junio, 2013).

<sup>219</sup> R., PLOTKIN, "Fighting Keywords: Translating the First Amendment to Protect Software Speech", *Journal of Law, Technology and Policy*, 2003, pág. 329 (337), disponible en: <http://ssrn.com/abstract=631861>, (última consulta 1 de junio, 2013).

<sup>220</sup> *Autronic AG v. Switzerland*, Judgment of 22 May 1990, núm. 178, 12 EHRR 485, para. 47.

entonces, parece existir un conflicto entre la propiedad que confiere el derecho de autor sobre el programa y el derecho a la libre expresión.

Si el valor del programa depende principalmente de su capacidad para comunicarse con otros programas y dispositivos, pero un derecho de propiedad como el derecho de autor puede limitar los medios de comunicación entre los programas y con ello restringir la creación de nuevos programas complementarios y competitivos, porque el acceso a la información contenida en las interfaces del programa es propiedad privada, ¿podría la negativa del titular a proporcionar dicha información a un tercero considerarse una vulneración del derecho fundamental a la libertad de expresión? Por ejemplo, cuando una compañía informática como Microsoft o Apple distribuye sus programas en código objeto y prohíbe el acceso al código fuente del programa, ¿está dando a entender “no leas el programa aunque pudieras leerlo”?<sup>221</sup>.

Cierto que la Directiva indica que “no será necesaria la autorización del titular cuando la reproducción del código y la traducción de su forma sea indispensable para obtener la información necesaria para la interoperabilidad”, pero la ausencia de esa autorización descansa sobre el cumplimiento de una serie de requisitos que, como se verá, dejan muchas incógnitas por despejar.

## **5. Conclusión de capítulo.**

La interoperabilidad como concepto multidimensional, juega un papel fundamental en el mercado del software. A mayor capacidad de interoperabilidad de un programa, mayor será la capacidad funcional de este con un mayor número de computadoras o dispositivos. Por tanto, la interoperabilidad aumenta la funcionalidad del programa en el mercado, aumentando el atractivo del programa tanto para competidores como para consumidores. Pero cuando esta información no está a disposición de los competidores, la consecución de la interoperabilidad choca con el derecho de autor sobre el programa de ordenador.

---

<sup>221</sup> L., TIEN, L., “Publishing Software as ...”, *op.cit.*, pág. 19. En igual sentido, B., ROTENBERG, “The European Regulation of ...”, *op.cit.*, pág. 8: “Interoperability primarily and mainly depends on the copyright-holder’s willingness. It is that player who will ultimately determine whether programs are interoperable”.

La protección de estas partes del programa como expresiones funcionales es causa de fricción entre la protección otorgada por el derecho de autor a los programas de ordenador y los intereses de la interoperabilidad como materia de interés público. El núcleo central del problema es cómo delimitar los contornos de qué ha de entenderse por funcionalidad expresiva y mera funcionalidad en un programa de ordenador.

Esta ausencia de claridad sobre el grado de protección de las partes del programa que contienen la información interoperable, unida a los *network effects* que se producen en el sector de las TICs, tiene también consecuencias en el mantenimiento de la libre competencia en el mercado porque en la práctica, indirectamente, fomenta conductas anticompetitivas cuando el titular del programa haya alcanzado una posición de dominio en el mercado.

Cuando la información interoperable no es proporcionada a los competidores de forma voluntaria, la ingeniería inversa se convierte en una herramienta importante a efectos de lograr la interoperabilidad. Sin embargo, alguna de las técnicas más adecuadas para este fin, como es la descompilación, plantean un importante problema al derecho de reproducción sobre el programa, que en sus últimas consecuencias sitúa a esta técnica ante la dicotomía de constituir una barrera para la innovación por legitimar la piratería o de ser un herramienta de impulso innovador por constituir una poderosa herramienta de investigación.

El examen de los elementos del debate por tanto, sitúa al derecho de autor en relación a la interoperabilidad no sólo como una cuestión jurídica en busca de un punto de equilibrio entre las partes implicadas, sino como un elemento con efectos directos sobre la innovación y el dinamismo del sector de la tecnologías de la información.

## **Capítulo 2**

### **El debate sobre la interoperabilidad informática en la Copyright Act de Estados Unidos.**





## 1. La protección de los programas en la Copyright Act de los Estados Unidos.

Entre las décadas de 1960 y 1970, profesionales de todo el mundo de la Propiedad Intelectual e Industrial comenzaron a debatir acerca de cuál debiera ser (si alguna) el régimen legal de protección que mejor se adaptase a los programas de ordenador. Estados Unidos fue el primer país que se decantó claramente por la opción del derecho de autor como mejor sistema de protección del software. Resultado de esta opción fue la modificación de la Copyright Act en 1980, que incluye y define el programa de ordenador en la enumeración de obras protegidas por el copyright del artículo 101<sup>222</sup>. Sin embargo, la protección del programa como obra autoral no fue tan sencilla como a priori pueda parecer, y, como casi siempre sucede, razones de carácter no estrictamente jurídico influyeron en que el legislador norteamericano se decantase por este sistema de protección exclusiva.

Desde el punto de vista doctrinal, uno de los primeros análisis detallados acerca de la “intelectualidad” de un programa de ordenador fue obra de *Stephen Breyer* en 1970<sup>223</sup>. En su análisis, BREYER destacaba que las razones de carácter económico para extender la protección del derecho de autor a un objeto como el programa de ordenador eran como mínimo dificultosas y poco convincentes<sup>224</sup>. Entre otros, destacaba la futilidad del argumento acerca de la recompensa a los programadores, ya que la mayoría del software en los años sesenta era producido por las empresas que fabricaban los ordenadores, a los que o bien ataban (*bundling*) sus propios programas, o bien creaban los programas por encargo del cliente (instituciones públicas o consumidores privados) que incluían en el mismo paquete que el ordenador. Incluso en el caso de programadores independientes, BREYER argumentaba que los costes de creación de un programa se recuperaban sin necesidad de proteger el programa con el derecho de autor. El programador no sólo vendía el programa al cliente, sino que incluía las labores de actualización, instalación, enseñanza de uso y mantenimiento del programa, de modo

---

<sup>222</sup> Artículo 101 “Definitions”: “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.”

<sup>223</sup> S., BREYER, “The Uneasy Case for Copyright: A Study of Copyright in Books, Photocopies, and Computer Programs”, Harv. L. Rev. 84, 1970, pág. 281.

<sup>224</sup> Ibidem, pág. 281.

que la inversión económica del programador se veía totalmente satisfecha<sup>225</sup>. Por tanto, los argumentos de carácter económico para proteger el programa a través del copyright, eran bastante débiles.

La historia legislativa de la adaptación del copyright al programa de ordenador, tampoco estuvo libre de inquietudes.

El primer paso hacia esta opción de protección se produjo con la aceptación a registro de un programa de ordenador por la Copyright Office en 1964, aunque se hizo “en beneficio de la duda”<sup>226</sup>. Es decir, la Oficina decidió emitir certificados de registro para los programas de ordenador, pero dichos documentos reflejaban las dudas del organismo acerca de la adecuación del régimen del copyright a una obra como el programa. Dejaba de este modo la carga de la prueba sobre el solicitante del registro, de llegar a cuestionarse ante un tribunal la idoneidad del copyright sobre el programa. Además, la Oficina hacía depender el registro del depósito de todo el código fuente del programa<sup>227</sup>.

En 1976, el Congreso revisó la Copyright Act y no consideró necesario incluir una mención explícita a los programas de ordenador. A pesar de ello, en la historia legislativa de la norma hay dos menciones a los programas en relación con el artículo 102(a) que dejan clara la intención del Congreso de proteger al programa de ordenador como obra de copyright y asimilada a una obra literaria<sup>228</sup>.

---

<sup>225</sup> *Ibidem*, pág. 344-345.

<sup>226</sup> “U.S. Copyright Office, Library of Congress, Circular 31D (1965)”, impresa en D.M., DAVIDSON, “Protecting Computer Software: A Comprehensive Analysis”, *Arizona St. L. J.*, n° 72, (1983), pág. 611 (652).

<sup>227</sup> *Ibidem*.

<sup>228</sup> *H.R.Rep.No. 1476, 94th Cong., 2d Sess. 54, reprinted in 1976 U.S.C.C.A.N.*, págs. 5659-5680. En concreto, al definir el término “obra literaria” de la Sección 102(a), el Informe del Congreso decía que “does not connote any criterion of literary merit or qualitative value” y que el mismo incluía “computer programs to the extent that they incorporate authorship in the programmer’s expression of original ideas, as distinguished from the ideas themselves” (pág. 5667). En relación al ámbito de protección del Copyright sobre el programa: “Some concern has been expressed whether copyright in computer programs should extend protection to the methodology or processes adopted by the programmer, rather than merely to the “writing” expressing his ideas. Section 102(b) is intended, among other things, to make clear that the expression adopted by the programmer is the copyrightable element in the computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law” (pág. 5670).

Debido a las dudas que surgieron durante el proceso legislativo de revisión de la Copyright Act en relación a los programas de ordenador<sup>229</sup>, por mandato del Congreso fue constituida a finales de 1974 una comisión especial, denominada *National Commission on New Technological uses of Copyrighted Works* (CONTU)<sup>230</sup>, con el objetivo de elaborar un estudio sobre dos cuestiones en el plazo de tres años: Los programas de ordenador y las fotocopias.

En 1978 fue publicado el informe final de la Comisión, que recomendaba reformar la *Copyright Act* de modo que incluyese la declaración formal de que los programas de ordenador son objeto del copyright. Siguiendo su recomendación, el Congreso reformó la *Copyright Act* en 1980 de dos maneras: Incluyó una definición explícita de “programa de ordenador” en el artículo 101, y modificó el artículo 117 para permitir a los propietarios (*owners*) de una copia de un programa, la copia y adaptación del mismo cuando tales actos “constituyesen un paso esencial para la utilización del programa de ordenador en conjunción con la máquina<sup>231</sup>”.

La importancia de este informe ha sido destacada por la doctrina por dos razones. En primer lugar, porque la *Computer Software Protection Act de 1980* que modifica la Copyright Act de 1976, recoge las recomendaciones de la CONTU casi literalmente. En segundo lugar, los tribunales han venido tratando el contenido del informe al modo de “exposición de motivos” de la Ley de 1980, de modo que se convirtió en el punto inicial de análisis en los casos de infracción de copyright del programa<sup>232</sup>.

---

<sup>229</sup> Durante las sesiones parlamentarias de revisión de la Copyright Act surgieron muchas dudas acerca de la idoneidad del derecho de autor como sistema de protección del programa, que pueden resumirse con el argumento de que los programas son objetos funcionales, fácilmente distinguibles de obras literarias tradicionales porque no guardan la menor relación con ellas. Además, la expresión del programa en código objeto es el proceso funcional del programa, lo cual no permite su asimilación a una obra literaria. De manera que la concesión de protección por copyright a una obra que consiste en un proceso, supondría conceder una protección similar a la de una patente pero sin las garantías que el sistema de patente proporciona. Vid. Copyright Law Revision: Hearing of S. 597 before the Subcomm. on Patents, Trademarks and Copyrights of the S. Comm. on the Judiciary, 90th Cong. 196-197, statement of Arthur R. Miller, 1967.

<sup>230</sup> *Final Report of the National Commission on New Technological Uses of Copyrighted Works, July 31, 1978*, Washington: Library of Congress, 1979. Disponible online: <http://www.digital-law-online.info/CONTU> (Última consulta 1 de junio, 2013).

<sup>231</sup> Computer Software Copyright Act of 12 December 1980, 1980, Pub. L. No. 96-517, 94 Stat. págs. 3007-3028

<sup>232</sup> J., BAND, M., KATOH, *Interfaces on Trial...*, *op.cit.*, pág. 72.

Sin embargo, este informe de la CONTU también ha sido objeto de especiales críticas por parte de otro sector de la doctrina. En su opinión, las recomendaciones de la Comisión tenían un componente mucho más político y económico que estrictamente jurídico. Entre los principales argumentos han indicado la falta de rigor en el análisis de la historia legislativa de la *Copyright Act*, la no adecuación con el principio constitucional de promover la ciencia y las artes, la ambigüedad en el uso del término “programa” a lo largo del informe, o el incorrecto planteamiento jurídico de la cuestión utilitaria de los programas, dejando a la discreción de cada tribunal asuntos tan esenciales como la definición de los requisitos de la obra o la delimitación del ámbito de protección del programa<sup>233</sup>.

La modificación de la *Copyright Act* trajo consigo una elevada actividad litigiosa a lo largo de las dos décadas siguientes, en las que, como veremos a continuación, los criterios de los tribunales fueron evolucionando desde una postura extensiva del ámbito de protección del programa, hacia otra postura más restrictiva. Como había predicho el Informe CONTU, el común denominador de los litigios fue la presunta infracción del copyright por copia del código.

## **2. La garantía de la interoperabilidad en la Copyright Act.**

### *2.1. El ámbito de protección del programa.*

El artículo 102 de la Copyright Act configura el objeto de protección del derecho de exclusiva, es decir, el concepto de obra. Literalmente dispone:

*“(a) Copyright protection subsists, in accordance with this title, in original works of authorship fixed in any tangible medium of expression, now known or later developed, from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device. Works of authorship include the following categories:*

*(1) literary works;*

---

<sup>233</sup> Vid. “Concurring opinion of Commissioner Nimmer”, y “Dissent of Commissioner Hersery”, en Capítulo 3, *Final Report of ...*, *op.cit.*, pág. 27-28; P., SAMUELSON, “CONTU Revisited: The Case Against Copyright Protection for Computer Programs in Machine Readable Form”, *Duke Law Journal*, 4, Septiembre, 1984, pág. 663; R.H., STERN, “Section 117 of the Copyright Act: Charter of the Software Users' Rights or an Illusory Promise”, *W. New Eng. L. Rev.*, 7, 1984-1985, pág. 459; J.K., HALVEY, “Rose by Any Other Name: Computer Programs and the Idea-Expression Distinction”, *Emory L. J.*, 34, 1985, pág. 741.

*(2) musical works, including any accompanying words;*

*(3) dramatic works, including any accompanying music;*

*(4) pantomimes and choreographic works;*

*(5) pictorial, graphic, and sculptural works;*

*(6) motion pictures and other audiovisual works;*

*(7) sound recordings; and*

*(8) architectural works.*

*(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."*

En relación a la interoperabilidad, dos son las cuestiones que se plantearon ante los tribunales. En primer lugar, la consideración de las interfaces del programa como parte del objeto protegido. En caso de que la respuesta sea afirmativa, habrá que determinar si la copia puntual de una parte del código, así como su transformación durante un proceso de ingeniería inversa como la descompilación, infringe el copyright del autor.

Las peculiares características del programa (funcionalidad y ausencia de finalidad comunicativa), encuentran su mayor exponente en las interfaces, las cuales como se vio en el anterior capítulo, tienen un carácter híbrido, es decir, están formados por una serie de elementos interrelacionados, unos funcionales y otros no funcionales. Si las mismas constituyen o no parte del objeto protegido por el copyright, las condiciones para alcanzar la interoperabilidad variarán considerablemente, porque en el primer caso será necesaria la autorización del titular del programa y en el segundo caso no. La aplicación de la doctrina idea - expresión va a ser una herramienta fundamental a la hora de tomar una posición al respecto.

En segundo lugar, si se concluye que las interfaces son parte del objeto de protección, la otra cuestión que afectará a efectos de lograr la interoperabilidad tiene que ver con las posibles utilizaciones que como usuarios legítimos del programa el ordenamiento nos permita realizar. Es decir, si ante la ausencia de la información interoperable por otros medios decidimos descompilar nuestro programa para acceder a

tal información, ¿la copia puntual del código, sea total o parcial, sea temporal o permanente, y la transformación del mismo de código objeto a código fuente constituyen una infracción del copyright? ¿Son una copia ilícita?

El control sobre el acto de copia es la facultad fundamental garantizada por el sistema del copyright. Pero el elemento esencial para considerar si ha habido tal copia es la similitud sustancial entre dos obras. Es decir, aunque se acredite la existencia de copia, ésta no tendrá consecuencias legales, no constituirá infracción, salvo que la copia sea sustancial (y no haya sido autorizada por su autor)<sup>234</sup>. A pesar de la aparente sencillez de estas premisas, los tribunales han encontrado bastantes problemas para determinar qué grado de correspondencia entre dos obras es el necesario para afirmar que exista una similitud sustancial suficiente que constituya infracción del derecho. De nuevo, la interpretación de los tribunales de la doctrina idea-expresión es clave para contestar esta pregunta.

Siguiendo la opinión de GOLDSTEIN, los conceptos de idea y expresión debieran ser entendidos como metáforas de las partes de las obras protegidas que se encuentran fuera y dentro del ámbito de protección del derecho de exclusiva<sup>235</sup>. De tal manera que idea es la metáfora de lo que no puede ser protegido por copyright, incluyendo pero no limitado a las ideas en abstracto, y expresión es la metáfora de lo que queda incluido dentro del ámbito de protección del copyright, incluso cuando se trata de palabras idénticas de un texto, notas musicales de una composición, o líneas de un dibujo sobre las que no cabe propiedad<sup>236</sup>. Por lo tanto, antes de decidir si la descompilación del código para lograr interoperabilidad constituye una infracción de los derechos de exclusiva del titular del copyright, hay que determinar qué parte del programa es idea y cuál expresión de la obra, para lo que es necesario aplicar el artículo 102 de la Copyright Act.

El examen de la jurisprudencia desde comienzos de los años ochenta hasta hoy día, muestra una clara evolución de la doctrina idea-expresión aplicada al objeto de protección del programa de ordenador.

---

<sup>234</sup> M. NIMMER, D., NIMMER, D., *Nimmer on...*, *op.cit.*, pág. 13.03 [A], 13-37.

<sup>235</sup> P., GOLDSTEIN, *Goldstein on Copyright*, Boston, Little, Brown & Co., 1989, §2.3.1

<sup>236</sup> Vid., entre otros, *Nichols v. Universal Pictures Corp.*, 45 F.2d 119, 121 (2d Cir. 1930): “It is of course essential to any protection of literary property...that the right cannot be limited literally to the text, else a plagiarist would escape by immaterial variations”

A continuación se presenta esta evolución estructurada cronológicamente. De su examen cabe hacer una triple clasificación en función de la posición adoptada por los tribunales: En un primer momento, la opción de extender el ámbito de protección del programa de ordenador a las interfaces en general, concediendo una protección muy amplia al titular del derecho. Después, los tribunales han tendido a la restricción del ámbito de protección del programa para favorecer los principios de la libre competencia. La jurisprudencia más reciente muestra la aproximación a la figura de las interfaces a través de un examen analítico de las mismas.

### *2.1.1. La extensión de la protección a las interfaces del programa.*

#### *A). Apple contra Franklin.*

El caso *Apple v. Franklin*<sup>237</sup> fue el primero con importantes repercusiones a efectos de la protección sobre las interfaces del programa<sup>238</sup>.

Este caso ilustra las complicadas políticas que inspiraron la protección del programa de ordenador a través del copyright.

Franklin, una pequeña empresa informática, desarrolló un clon del ordenador Apple II, copiando además, bit a bit, el sistema operativo Apple II-OS. Desde el punto de vista del consumidor esto se traducía en la posibilidad de adquirir un ordenador Franklin, más barato que el fabricado por Apple, y en el que podría utilizar cualquier software de Apple.

Ante la demanda por infracción del copyright, la principal alegación de Franklin fue admitir la copia del sistema operativo, justificando tal acto en su indispensabilidad para garantizar la interoperabilidad entre los programas de aplicación creados por ellos o cualquier otro competidor en un ordenador de Apple.

---

<sup>237</sup> *Apple Computer Inc. v. Franklin Computer Corp.*, 714 F.2d 1240 (3rd Cir. 1983); *cert. dismissed*, 464 US, 1033 (1984).

<sup>238</sup> El caso *Synercom Technology, Inc. v. University Computing Co.*, 42 F. Supp. 1003 (N.D. Tex., 1978), fue el que por primera vez trató de aplicar la dicotomía idea-expresión a los programas de ordenador, pero su doctrina no fue seguida por otros tribunales como sí sucedió en el caso *Apple v. Franklin*. La pregunta que se planteaba el tribunal del caso *Synercom* era: “if sequencing and ordering is expression, what separable idea is expressed?” (pág. 1013). Un comentario sobre la misma puede encontrarse en J.K., HALVEY, J.K., “A rose by any ...”, *op.cit.*, pág. 760.

Los demás argumentos usados por Franklin en su defensa se basaban en que, en primer lugar, un programa de ordenador expresado en código objeto no podía constituir obra para el copyright al resultar ilegible para un ser humano; por otra parte, un programa de ordenador integrado en un chip de memoria ROM (Read-Only Memory), no podía ser protegido por copyright por tratarse de una parte de la máquina; y en tercer lugar, que los sistemas operativos no podían ser protegidos como programas de ordenador a través de copyright porque el sistema operativo de Apple constituía la única forma de expresión posible de la idea en la que se había inspirado.

La cuestión que tuvo que valorar el Tribunal de Apelaciones del Tercer Circuito fue si estas tres clases de obras tenían naturaleza mecánica o literaria, y la conclusión a la que llegó fue la segunda.

En relación al código objeto, el tribunal declaró que “the copyrightability of computer programs is firmly established after the 1980 amendment to the Copyright Act”.

Respecto del programa integrado en un chip de memoria de lectura (ROM), consideró que el mismo constituía una forma de expresión válida del programa y por tanto protegible por copyright.

Sobre la cuestión de los sistemas operativos, el tribunal también rechazó la argumentación de Franklin, citando alguna de las conclusiones del Informe de la CONTU según el cual, que las palabras de un programa sean utilizadas para implementar un proceso no debe afectar a la hora de valorar su adecuación como objeto protegible por copyright. Hasta aquí la decisión no hubiese presentado mayores problemas, sin embargo el tribunal fue un paso más allá en el examen de la dicotomía idea-expresión y añadió: “Franklin may wish to achieve total compatibility with independently developed application programs written for the Apple II, but that is a commercial and competitive objective which does not enter into the somewhat metaphysical issue of whether particular ideas and expressions have merged”<sup>239</sup>.

El análisis del tribunal de apelación parte de tres afirmaciones erróneas:

---

<sup>239</sup> Apple v. Franklin, *op.cit.*, pág. 1253.



- 1) El tribunal identifica la idea de un sistema operativo, con la compilación del código, un paso en el proceso de creación del programa. Al considerar que el código objeto del programa sin más es el programa de ordenador protegido por copyright, sin distinguir entre las distintas partes que componen el código informático, se está protegiendo por igual partes del programa que sí pueden ser originales y otras que no lo son.
- 2) El tribunal tergiversa el objetivo de la doctrina de la merger doctrine. La finalidad de la doctrina de la fusión (merger doctrine) al distinguir entre idea y expresión, es limitar el derecho de copyright para mantener la competitividad entre las ideas<sup>240</sup>, de manera que nadie pueda ser titular de un derecho de propiedad sobre una idea que sólo puede expresarse de una única forma<sup>241</sup>. El tribunal la aplica porque Franklin copió literalmente el sistema operativo de Apple y no tuvo intención de escribir programas alternativos que realizasen las mismas funciones, cuya expresión no podía expresarse de otro modo, y determina que su comportamiento constituye una infracción<sup>242</sup>.
- 3) El Tribunal identifica “interoperabilidad” con “compatibilidad”, a la que además trata como “objetivo competitivo comercial” que no ha de valorarse en la determinación de fusión entre expresión e idea<sup>243</sup>.

*B). Whelan contra Jaslow.*

La doctrina sentada por el caso *Franklin* en 1984 se vio reflejada poco después en el caso *Whelan*, que la llevó a su máximo exponente.

En 1986, el caso de *Whelan v. Jaslow* extendió el ámbito de protección del programa de ordenador no solo a las interfaces que contienen la información sobre interoperabilidad, sino también a las interfaces gráficas de usuario.

---

<sup>240</sup> J., BAND, M., KATOH, *Interfaces on Trial...*, *op.cit.*, pág. 87.

<sup>241</sup> Como indica GOLDSTEIN, esta es la interpretación tradicional de la doctrina de la fusión. Sin embargo, en los primeros años del siglo XXI, ha surgido una nueva interpretación menos ortodoxa por parte del Tribunal de Apelaciones del Noveno Circuito, cuyo fin se centra más en determinar el ámbito de protección del copyright que la existencia del objeto protegible. Vid. P., GOLDSTEIN, *Goldstein on Copyright*, *op.cit.*, §2.3.2, 2:33, 2005 Supplement.

<sup>242</sup> *Apple v. Franklin*, *op.cit.*, pág. 1245 en relación con pág. 1253.

<sup>243</sup> *Apple v. Franklin*, *op.cit.*, pág. 1253.

El punto de partida, una demanda por infracción del derecho de copyright sobre un programa de gestión de un laboratorio dental. A lo largo del juicio quedó claro que los códigos de ambos programas eran diferentes, sin embargo, se habló de que la estructura de los archivos y la presentación de ambos programas en pantalla eran “prácticamente idénticas”<sup>244</sup>. El Tribunal de Apelaciones del Tercer Circuito, para decidir si había infracción de copyright, se planteó si la protección otorgada a un programa incluía la estructura de éste o sólo protegía sus elementos literales (los códigos). Para ello formuló una regla general que diferenciaba entre idea y expresión en una obra utilitaria, y dio una nueva definición de la expresión en un programa de ordenador como la manera en la cual el programa opera, controla y regula que el ordenador reciba, ensamble, calcule, retenga, correlacione y produzca información útil tanto en una pantalla, como en una impresión, como en una comunicación auditiva<sup>245</sup>. En base a ello concluyó que la protección de los programas de ordenador debía extenderse más allá de los elementos literales y proteger su estructura, secuencia y organización.

La posición adoptada por el Tribunal de Apelaciones del Tercer Circuito en el caso *Whelan* fue duramente criticada por la doctrina por dos motivos:

- 1) El tribunal identificó que en un programa de ordenador hay una única idea.
- 2) El Tribunal redujo la doctrina de la dicotomía entre idea y expresión a la doctrina de la fusión<sup>246</sup>. Es más, al otorgar protección al modo en que el programa ejecuta una función, el tribunal estaría empleando el sistema de copyright para proteger materia patentable<sup>247</sup>.

La doctrina expansiva del caso *Whelan* influyó sobre la Oficina de Copyright, que a partir de 1988 estableció que el registro de un programa de ordenador se haría en un asiento único, con el cual se protegían todos los aspectos susceptibles de amparo,

---

<sup>244</sup> *Whelan Associates, Inc., v. Jaslow Dental Laboratory, Inc.*, 797 F. 2d 1222 (3d. Cir., 1986), pág. 1228.

<sup>245</sup> *Ibidem*. pág. 1236

<sup>246</sup> M., NIMMER *et al*, *Cases and materials on Copyright and other aspects of Entertainment litigation including Unfair Competition, Defamation, Privacy*, Lexis cop., 6ª ed., New York, 2000, pág. 253; J., BAND, M. KATOH, *Interfaces on Trial 2.0*, *op.cit.*, pág. 13.

<sup>247</sup> “Patents, not copyrights, protect «the means for carrying the idea out»” en *Kruger v. Whitehead*, 153 F. 2d 238, 239 (9<sup>th</sup> Cir. 1946), *cert. denied*, 332 U.S. (1947), pág. 774. Vid. J. BAND, M. KATOH, *Interfaces on Trial 2.0*, *op.cit.*, pág. 14; J., PINHEIRO, G., LACROIX, “Protecting the “look and feel” of computer programs”, *High Technology Law Journal*, 11, 1987, pág. 1, online: <http://btlj.org/data/articles/voll/pinheiro.pdf> (última consulta 1 de junio, 2013).

incluyendo el código fuente, el código objeto y las presentaciones visuales. Es más, la Oficina llegó a afirmar que el código del programa y las presentaciones visuales están relacionadas de modo integral y forman de manera ordinaria una única obra<sup>248</sup>.

Los tribunales menores siguieron la doctrina *Whelan* como demuestra algún caso como *Broderbund v. Unison World*<sup>249</sup> donde el tribunal parte de esta sentencia para extender la protección hasta la estructura, secuencia y organización de las apariencias en pantalla.

Otro ejemplo fue el caso *Lotus v. Paperback*, que confirmó la postura doctrinal de ampliar el ámbito de protección concedido por el copyright a los elementos no literales de los programas de ordenador como las interfaces. Abrió además una peligrosa senda a la expansión del copyright a los aspectos sonoros (audio y voces) o cualquier otra forma no tradicional de interfaz gráfica de usuario<sup>250</sup>. En teoría, la voz, el audio, el tacto y cualquier otra forma de percepción sensorial, si es original y está “fijada” de alguna manera en una expresión, podría ser protegida por copyright si, en palabras del Tribunal “*the overall structure, the order of commands, the presentation of these symbolic tokens on the screen (i.e., first letter only, abbreviations, full words, full words with one or more letters capitalized or underlined), the type of menu system used (i.e., one-, two-, or three-line moving-cursor menus, pull-down menus, or command-driven interfaces), and the long prompts — could be expressed in a great many if not literally unlimited number of ways.*”<sup>251</sup>

### 2.1.2. La restricción del ámbito de protección sobre algunos elementos del programa en favor de la libre competencia.

#### A) *Pains Cotton contra Goodpasture*.

Hacia finales de los años ochenta, otros tribunales comenzaron a rechazar las teorías extensivas de protección sobre los elementos no literales de los programas.

---

<sup>248</sup> Copyright Office, Notice of Registration Decision: Registration and Deposit of Computer Screen Displays, 53 Fed. Reg. 21.817.21.819 (1988).

<sup>249</sup> *Broderbund v. Unison World*, 648 F. Supp. 1127 (N.D. Cal, 1986), pág. 1127.

<sup>250</sup> *Lotus Dev. Corp. v. Paperback Software Intern.*, 740 F. Supp. 37 (D.C. D. Massachusetts 1990), pág. 37.

<sup>251</sup> *Ibidem* pág. 67

Entre los tribunales de instancia destaca el caso *Digital Communications v. Softklone*. En esta ocasión el tribunal, si bien entiende que el derecho de copyright sobre el programa se extiende hasta los elementos no literales del mismo, siguiendo la doctrina Whelan, matiza que debe diferenciarse entre la protección concedida a los códigos y la protección concedida a las interfaces o apariencia visual del programa, puesto que ambas son partes del mismo pero cada una constituye una obra independiente y por lo tanto requieren dos registros diferentes<sup>252</sup>.

La confirmación de que la protección del programa no podía extenderse por igual a todos sus elementos se produjo en *Plains Cotton Cooperative Association v. Goodpasture Computer Service*<sup>253</sup>.

El Tribunal de Apelaciones del Quinto Circuito consideró que pese a que el programa del apelado tenía una organización muy similar, no existía infracción del copyright. La Corte de Apelaciones recoge la doctrina establecida en 1978 por el caso *Synercom v University Computing*, según la cual los “*input formats*” de un programa de ordenador (la organización y configuración de la información que el programa pasa a la computadora), son ideas, no expresiones y por tanto no están protegidas por copyright<sup>254</sup>.

También rechaza la doctrina *Whelan* al afirmar que la calificación de una interfaz gráfica de usuario como idea o expresión, debe hacerse examinando caso por caso, teniendo en cuenta el contexto de ese programa en la industria y valorando si para ese sector empresarial es imprescindible que las presentaciones en pantalla se muestren de una forma determinada debido a los usos y costumbres del sector.

Por lo tanto, factores como la interoperabilidad de los programas entre sí, la facilidad de su uso y aprendizaje, y la necesidad de establecer estándares en algunos sectores del mercado se configuran como elementos limitativos del ámbito de protección concedido por el copyright sobre el programa de ordenador.

---

<sup>252</sup> *Digital Communications v. Softklone*, 659 F. Supp. 449 (N.D. Ga. 1987), págs. 455-456.

<sup>253</sup> *Plains Cotton Cooperative Association of Lubbock, Texas v. Goodpasture Computer Service, Inc.*, 1 USPQ 2d 1635 (5<sup>th</sup> Cir. 1987), pág. 1635.

<sup>254</sup> *Synercom Technology Inc. v. University Computing Co.*, 42 F. Supp. 1003 (N.D. Texas 1978), pág. 1014.

La doctrina establecida por el Tribunal de Apelaciones del Quinto Circuito constituyó un primer paso hacia la adopción de un examen analítico del programa para determinar qué elementos forman parte de la obra a efectos de copyright y cuáles no.

### *2.1.3. La exclusión de las interfaces del ámbito de protección sobre el programa a través de un análisis analítico.*

#### *A). Computer Associates contra Altai.*

La tercera etapa de la jurisprudencia norteamericana se caracteriza por la aplicación de un test analítico para determinar el ámbito de protección sobre el programa.

Este análisis de elemento por elemento del programa fue desarrollado por el Tribunal de Apelaciones del Segundo Circuito en el caso *Computer Associates v. Altai* y es conocido como “el test de Altai”<sup>255</sup>.

El objeto del test consiste en determinar si una copia no literal de un programa, es decir, una copia que parafrasea la obra, constituye una infracción. Para ello es necesario determinar el ámbito de protección de la obra. A tales efectos, han de efectuarse tres operaciones: abstracción, filtración y comparación.

El tribunal examina en primer lugar, qué son ideas y qué son expresiones en las interfaces en lid (abstracción). En segundo lugar, define el alcance del derecho de copyright del demandante, “examinando los componentes estructurales en cada nivel de abstracción para determinar si su inclusión particular en ese nivel es “idea” o ha sido dictado por consideraciones de eficiencia”<sup>256</sup> (filtración). La etapa final del test consiste en comparar los resultados obtenidos en los pasos anteriores, aplicando un examen subjetivo (comparación de las similitudes sustanciales) para determinar si son lo suficientemente similares como para que exista infracción del copyright. Sólo en el caso de que exista “virtual identity” (identidad) habrá infracción.<sup>257</sup>

La relevancia de la sentencia en el caso *Computer Associates v. Altai*, va más allá de la proposición de este test, porque no sólo marcó un punto de inflexión en

---

<sup>255</sup> *Computer Assoc. Int'l, Inc. v. Altai, Inc.*, 982 F.2d 693 (2d Cir. 1992).

<sup>256</sup> *Computer Assoc. Int'l, Inc. v. Altai, Inc.*, *op.cit.*, pág. 707.

<sup>257</sup> *Brown Bag Software v. Symantec Corp.*, 960 F. 2d 1465 (9<sup>th</sup> Cir. 1992), pág. 1475.

relación al ámbito de protección del copyright sobre los programas de ordenador. El Tribunal de Apelaciones del Segundo Circuito aplica por primera vez la doctrina del Tribunal Supremo de *Feist Publications, Inc. v. Rural Telephone*<sup>258</sup> a los programas de ordenador.

En relación a la primera cuestión, la decisión del Tribunal de apelación descarta por completo la doctrina *Whelan*, argumentando que la misma es demasiado simplista y que se basa en distinciones metafísicas. Para ello propone utilizar un examen basado en consideraciones prácticas, materializado en el expuesto test de *Altai*<sup>259</sup>. Propuesto el test, añade que los elementos de un programa de ordenador pertenecerán al dominio público si la expresión de los mismos es común o estándar para la industria informática<sup>260</sup>, e indica que el copyright solo puede ofrecer una fina protección a los programas, de modo que para obtener una protección más amplia sobre ciertos elementos del programa, el sistema más adecuado sería el derecho de patente, aunque en tal caso, los requisitos de protección son más rigurosos que en el caso del copyright<sup>261</sup>.

En cuanto a la aplicación de la doctrina *Feist* a los programas de ordenador, conviene recordar que en este caso el Tribunal Supremo declara que el objetivo primario del copyright no consiste en recompensar la labor de los autores<sup>262</sup>. El Tribunal de Apelaciones del Segundo Circuito aplica esta doctrina afirmando que un esfuerzo sustancial por sí solo no puede ser causa para proteger una obra a través del copyright, si en ausencia de tal esfuerzo, la obra no pudiera constituir objeto de copyright<sup>263</sup>. Trasladado a los programas de ordenador, continúa el Tribunal, significa que la doctrina *Feist* rechaza la doctrina *Whelan*, porque la misma amplía el ámbito de protección a los elementos no literales del programa de un modo que legitima a los que “llegan primero”

---

<sup>258</sup> *Feist Publications, Inc. v. Rural Telephone Service Co.*, 499 U.S. 340 (1991).

<sup>259</sup> *Computer Assoc. Int'l, Inc. v. Altai, Inc.*, *op.cit.*, pág. 703.

<sup>260</sup> *Ibidem*, pág. 710.

<sup>261</sup> *Ibidem*, pág. 711-712.

<sup>262</sup> *Feist Publications, Inc. v. Rural Telephone Service Co.*, *op.cit.*, pág. 1290. El texto original dice: “the primary objective of copyright is not to reward the *labor* of authors (...)”.

<sup>263</sup> “Feist teaches that substantial effort alone cannot confer copyright status on an otherwise uncopyrightable work” en *Computer Assoc. Int'l, Inc. c. Altai, Inc.*, *op.cit.*, pág. 712.

bloquear técnicas de programación básicas en la forma que eran implementadas para realizar determinadas funciones<sup>264</sup>.

La elaboración de un test como el de *Altai* demuestra la comprensión por parte del Tribunal de que los retos que la protección del software plantea al copyright no se producen porque sí, sino que tienen lugar en un escenario comercial muy concreto, el cual limita el abanico de posibles expresiones que un programador puede utilizar para escribir el programa. Por otro lado, la declaración del Tribunal de que el copyright no puede proteger elementos que vengan dados por razones de eficiencia o factores externos, manifiesta la importancia que la interoperabilidad tiene para la industria informática.

Entre estos elementos que no pueden protegerse por el copyright, enumera el Tribunal<sup>265</sup>: Las especificaciones mecánicas de un ordenador donde un programa concreto haya de ejecutarse; los requisitos de interoperabilidad con otros programas; los estándares de diseño de los fabricantes de ordenadores; las necesidades de la industria y la prácticas comúnmente aceptadas por la industria informática.

Una última consideración a destacar de la sentencia del caso *Altai* es que el Tribunal recomienda a los tribunales de instancia que soliciten la guía de expertos independientes en casos donde las cuestiones tecnológicas sean complejas<sup>266</sup>.

#### *B). Apple Computer contra Microsoft y Hewlett-Packard.*

La decisión del Tribunal de Apelaciones del Noveno Circuito en el caso *Apple Computer v. Microsoft y Hewlett-Packard*<sup>267</sup> siguió la doctrina sentada por el Tribunal de Apelaciones del Segundo Circuito en *Computer Associates* en relación a las interfaces gráficas. En el litigio, Apple alegaba que las demandadas habían copiado la interfaz de usuario de su sistema operativo Lisa, en especial la metáfora del escritorio denominado *Finder*. Sin embargo, el Tribunal de Apelaciones del Noveno Circuito, tras aplicar el test, sostuvo que no puede otorgarse un derecho de copyright al uso de ventanas para

---

<sup>264</sup> “(...) the Whelan rule, (...) enables first comers to “lock up” basic programming techniques as implemented in programs to perform particular tasks” en *Computer Assoc. Int’l, Inc. c. Altai, Inc., op.cit.*, pág. 713.

<sup>265</sup> *Ibidem*, pág. 708 a 711.

<sup>266</sup> *Ibidem*, pág. 698.

<sup>267</sup> *Apple Computer, Inc. v. Microsoft Corp. and Hewlett-Packard Co.*, 35 F. 3d 1435 (9<sup>th</sup> Cir. 1994).

mostrar varias imágenes en una pantalla de ordenador, a la representación de iconos de un equipo de oficina familiar, a la manipulación de iconos para el control operativo de una computadora, al uso de los menús para almacenar información, o a la apertura y cierre de objetos como medio de recepción, transferencia y almacenamiento de la información. La copia por un competidor de estas ideas contenidas en la interfaz gráfica de usuario, por lo tanto, no infringen su derecho de copyright<sup>268</sup>.

El efecto que produjo esta decisión fue permitir a los competidores incorporar las mismas funciones en sus programas siempre que usasen formas de representación distintas en la pantalla. Esta sentencia, confirmó la tendencia de los tribunales de restringir el alcance de la protección del copyright a elementos no literales del programa, especialmente a las interfaces gráficas de usuario.

*C). Lotus contra Borland (Lotus II).*

Poco después, el caso *Lotus v. Borland (Lotus II)*<sup>269</sup>, planteó una nueva cuestión, la de si el copyright alcanzaba a la jerarquía de comandos de un menú en cuanto tal. El interés del caso radica en que la respuesta a esta pregunta afecta directamente a la protección de elementos necesarios para alcanzar interoperabilidad.

El Tribunal de Apelaciones del Primer Circuito mantuvo el criterio de que los comandos de un menú no pueden ser protegidos por copyright porque son un método de operación, lo cual está expresamente excluido en el art. 102(b) de la Copyright Act<sup>270</sup>. El tribunal justificó su conclusión comparando el menú de comandos de Lotus con las

---

<sup>268</sup> Un comentario extenso de la sentencia puede encontrarse en R., COLE, “Substantial similarity in the Ninth Circuit: a “virtually identical” “look and feel”? *Apple Computer, Inc. v Microsoft Corp.*, 35F 3d 1435 (9<sup>th</sup> Cir. 1994)”, en *Santa Clara J.C.H.T.*, 11, Julio, 1995, pág. 417.

<sup>269</sup> *Lotus Development Corp. v. Borland Intern., Inc.*, 49 F. 3d 807 (1st Circuit 1995), *aff'd by an equally divided Court*, 516 U.S. 233 (1996). Comentario a esta sentencia puede encontrarse en M., BOUZA LOPEZ, “Idea/expresión en los programas de ordenador. (Comentario a la Sentencia *Lotus c. Borland* del Primer Circuito de los Estados Unidos, de 9 de marzo de 1995)”, *ADI*, 18, (1997), págs. 325-346; J., BAND, M. KATOH, *Interfaces on Trial 2.0...*, *op.cit.*, pág. 24 – 41.

<sup>270</sup> Por otra parte, el juez Boudin, en su “concurrance opinion”, había manifestado que los comando de menú de un programa de ordenador no podían calificarse como métodos de operación. A pesar de ello, alegaba que su uso debía permitirse a través de la doctrina del fair use, abriendo la puerta a posible aplicación de la doctrina del “privileged use” en situaciones de interoperabilidad de software. Vid. *Lotus v. Borland*, (1st Circuit 1995), *op.cit.*, pág. 820.



teclas de una grabadora y aplicando el test de *Altai*, el cual, en su opinión era de poca ayuda cuando se trataba de considerar la copia de elementos literales<sup>271</sup>.

El Tribunal Supremo revisó la decisión del Tribunal de Apelaciones del Primer Circuito<sup>272</sup> a través de una petición de *certiorari*. Fue confirmada a pesar de producirse un empate técnico (cuatro votos a favor y cuatro en contra). Lamentablemente, el Tribunal Supremo no dio ninguna explicación de su decisión, y como resultado no sentó un precedente federal uniforme para regular un tema tan controvertido como éste.

La enseñanza que puede extraerse de la decisión tomada por el Supremo en el caso *Lotus II* es que el péndulo del copyright se balancea constantemente desde una postura extensiva, concediendo una amplia protección para los programas de ordenador, hasta una protección más moderada, tendente a limitar el alcance de la protección concedida. Esta constante oscilación sigue vigente hoy día, ya que desde el caso *Lotus II* hasta hoy, a falta de doctrina del Tribunal Supremo, las posiciones de los tribunales norteamericanos muestran poca uniformidad de criterio.

#### *2.1.4. El aparente retorno hacia la doctrina extensiva.*

##### *A). Bateman contra Mnemonics.*

Muestra de que la oportunidad dejada pasar en el caso *Lotus II* fue una operación de alto riesgo, fue la decisión del Tribunal de Apelaciones Undécimo del Circuito en el caso *Bateman v. Mnemonics*.

El Tribunal declaró que los programas son obras literarias utilitarias que deben recibir una protección más estrecha que obras literarias más expresivas como novelas y obras teatrales<sup>273</sup>. Aunque el tribunal no se pronuncia sobre si las interfaces deberían excluirse de la protección del copyright, afirma que la argumentación de *Mnemonics*, parte que defendía la inclusión de las interfaces en el ámbito de protección del programa, es incorrecta porque, que algo pueda constituir objeto de copyright no significa que vaya a ser protegido por copyright y, porque la expresión de ese algo ha de

---

<sup>271</sup> *Lotus v. Borland*, (1st Circuit 1995), *op.cit.*, pág. 816.

<sup>272</sup> *Lotus Development Corp. v. Borland International, Inc.*, 516 U.S. 233 (1996).

<sup>273</sup> *Bateman v. Mnemonics*, 79F. 3d 1532 (11th Circuit 1996), pág. 1541, y nota 21.

superar la aplicación de la doctrina de la fusión establecida en el artículo 102(b) de la Copyright Act<sup>274</sup>.

La renuencia del tribunal a pronunciarse sobre este tema queda más clara cuando afirma que “especificación de interfaz” es un término informático, y a lo largo del fallo emplea como sinónimas otras expresiones como “*interface commands*”<sup>275</sup> o “operating system interface”<sup>276</sup>, cuando a continuación, pero en una nota al pie, recuerda el tribunal que en ningún caso la protección del copyright debe extenderse a resultados funcionales<sup>277</sup>.

#### *B). Dun & Bradstreet contra Grace Consulting.*

En 2002, la decisión del Tribunal de Apelaciones del Tercer Circuito en el caso *Dun & Bradstreet v. Grace Consulting*<sup>278</sup> llama poderosamente la atención porque la misma parece un retorno a la doctrina extensiva del ámbito de protección del programa, aparentemente superado desde los casos *Computer Associates* y *Lotus*.

Dos son las razones que llevan a concluir este retorno: En primer lugar, porque el tribunal rechaza expresamente el examen de los factores externos establecido en el test *Altai*<sup>279</sup>, y en segundo lugar porque cita la sentencia del caso *Apple* contra *Franklin*, con más de veinticinco años de antigüedad, recuperando la concepción de la “compatibilidad” mantenida por el tribunal como objetivo competitivo comercial que no ha de valorarse en la determinación de fusión entre expresión e idea al delimitar el ámbito de protección sobre el programa<sup>280</sup>.

#### *C). Oracle contra Google.*

En mayo de 2012, el Juzgado del Distrito de San Francisco, en un juicio que ha tenido bastante repercusión mediática por la cuestión que se debatió, deja claro que la

---

<sup>274</sup> *Ibidem*, pág. 1547 y nota 32. “It is an incorrect statement of the law that interface specifications are not copyrightable as a matter of law”.

<sup>275</sup> *Ibidem*, pág. 1547.

<sup>276</sup> *Ibidem*, nota 30.

<sup>277</sup> *Ibidem*, nota 33: “In no case, however, should copyright protection be extended to functional results obtained when program instructions are executed and such results are processes of the type better left to patent and trade secret protection.”

<sup>278</sup> *Dun & Bradstreet Software v. Grace Consulting*, 307 F.3d 197 (3rd Cir. 2002).

<sup>279</sup> *Ibidem*, pág. 216.

<sup>280</sup> *Apple v. Franklin*, *op.cit.*, pág. 1253. Para un comentario detallado sobre la decisión del caso *Dun & Bradstreet* contra *Grace Consulting*, vid. J., BAND, M., KATOH, *Interfaces on Trial 2.0...*, *op.cit.*, pág. 51 y ss.

postura de la jurisprudencia norteamericana en relación a las interfaces del programa es que no forman parte del objeto de protección del copyright<sup>281</sup>.

El origen de la disputa se encuentra en un comportamiento de Google sobre determinadas partes del software de Oracle y de un lenguaje de programación creado por Oracle, Java.

En teoría, Google era totalmente libre para usar el lenguaje de programación Java para crear su sistema operativo. Pero la empresa no hizo solamente esto; Google también usó algunas de las interfaces de aplicación (APIs) que Sun Microsystems había creado para Java.

El sistema operativo que creó Google utilizando Java es el denominado Android, y ha resultado un éxito rotundo. Desde una perspectiva comercial, hoy en día hay más *smartphones* con sistemas Android que con el iOS de Apple, característico de los *Ip hones*. Mientras Android se convertía en el sistema operativo móvil más popular del mundo, Oracle compró Sun Microsystems y, paso seguido, demandó a Google por usar las APIs de Java, insinuando que podría cobrar al gigante informático cientos de millones de dólares por usar tecnología que ahora le pertenecía.

La cuestión que se debatió ante el juzgado de San Francisco, fue si esas interfaces de aplicación o APIs estaban protegidas por el copyright.

En su fallo, el juez Alsup recurre al test *Altai* para examinar estos elementos y determinar si ha habido copia no literal del programa. Aplicando los tres pasos del test, el juez analiza si las APIs constituyen idea o expresión del programa, y declara que “*as long as the specific code writer to implement a method is different, anyone is free under the Copyright Act to write his or her own method to carry out exactly the same function or specification of any and all methods used in the Java API. (...) copyright law does not confer ownership over any and all ways to implement a function or specification, no matter how creative the copyrighted implementation or specification may be. The Act confers ownership only over the specific way in which the author wrote out his version.*

---

<sup>281</sup> Oracle America Inc. v. Google Inc., US District Court (31 Mayo 2012, C10-03561 WHA). A fecha de cierre de este trabajo, la apelación presentada por Oracle, en octubre de 2012, sigue a la espera de la decisión por el Tribunal de Apelación del Circuito Federal. (Oracle America Inc. v Google Inc. Appeal from the United States District Court for the Northern District of California in case no. 10-CV-3561, Judge William H. Alsup).

*Others are free to write their own implementation to accomplish the identical function, for, importantly, ideas, concepts and functions cannot be monopolized by copyright*<sup>282</sup>.

Es más, el juez *Alsup*, en consonancia con parte de la doctrina<sup>283</sup> y con la postura defendida en este trabajo, declara que las interfaces son métodos de operación, y por tanto no pueden ser protegidas por el copyright. Aún reconociendo que sean expresiones y que cumplan con el requisito de originalidad, son métodos de operación, “una lista de más de seis mil comandos para llevar a cabo unas funciones asignadas previamente”<sup>284</sup> y no pueden ser protegidas por el copyright, quizás sí puedan protegerse a través del derecho de patente, pero jamás a través de copyright.

A pesar de esta importante sentencia, la cuestión de la protección de las interfaces, consideradas como elementos no literales del programa de ordenador, sigue sin un criterio único, y así, algunos tribunales se han mostrado a favor de incluir las interfaces en el ámbito de protección del copyright del programa siempre que formen parte de una compilación en una manera única o creativa<sup>285</sup>, o de considerar que tanto los elementos literales del programa, los códigos fuente y objeto, como los elementos no literales, que incluyen las visualizaciones en pantalla, los menús principales, las estructuras de comandos de los submenús, las listas de parámetros, las macros, y los diagramas generales, pueden recibir protección por copyright<sup>286</sup>.

Quizás, si la apelación presentada por Oracle contra la decisión del Juez *Alsup* es confirmada por el Tribunal de Apelación del Circuito Federal, y Oracle apelase al Tribunal Supremo, éste tendría la oportunidad de establecer un criterio que pusiera fin a este debate, con grandes repercusiones para la innovación en la industria de las TICs<sup>287</sup>.

---

<sup>282</sup> *Ibidem*, pág. 34.

<sup>283</sup> P., SAMUELSON, T., VINJE, W., CORNISH, “Does Copyright Protection Under the EU Software Directive Extend to Computer Program Behaviour, Languages and Interfaces?”, *EIPR* 2, (2012), pág. 158; disponible en SSRN: <http://ssrn.com/abstract=1974890> or <http://dx.doi.org/10.2139/ssrn.1974890> (Última consulta 1 de junio, 2013); S., WESTON, “Software Interfaces – Stuck in the Middle: The Relationship Between the Law and Software Interfaces in Regulating and Encouraging Interoperability”, *IIC* 43(4), 2012, pág. 427 (449); S., VEZZOSO, “Copyright, Interfaces and a Possible Atlantic Divide”, *JIPITEC* 2, 2012, pág. 153 (158).

<sup>284</sup> Oracle America, Inc. v. Google Inc., *op.cit.*, pág. 38.

<sup>285</sup> SecureInfo Corp. v. Telos Corp., 387 F. Supp. 2d 593 (E.D. Va. 2005).

<sup>286</sup> Goldman v. Healthcare Management Systems, Inc., 628 F. Supp. 2d 748 (W.D. Mich. 2008).

<sup>287</sup> En relación a la repercusión de la interoperabilidad a través de las interfaces de aplicación (APIs) para la innovación en el sector de las TICs, vid. *Amici Curiae in Oracle v. Google*, presentado el 30 de mayo de 2013; disponible en: <http://www.groklaw.net/pdf4/OraclevGoogleAppealEFFamicus.pdf> (última consulta 1 de junio, 2013).

## 2.2. *Las limitaciones de la Copyright Act.*

Con carácter general, las limitaciones son la clave para asegurar el equilibrio de intereses entre los titulares de los derechos de autor y el público, a pesar de que las nuevas tecnologías y la continua aparición de nuevas prácticas sociales y culturales desafíen constantemente este equilibrio legal<sup>288</sup>. Como ha recordado GEIGER, si bien las discusiones doctrinales sobre este tema hace unos años se centraban más en cómo expandir el sistema de los derechos de autor y en cómo adaptarlo a las nuevas tecnologías, hoy por hoy, han girado hacia cómo garantizar el justo equilibrio entre protección y usos libres<sup>289</sup>.

En este dilema se sitúan las limitaciones y excepciones que afectan al copyright sobre el programa cuando la interoperabilidad se cruza entre las facultades del autor del programa y los usuarios legítimos del mismo. Cuando la información puesta a disposición por el autor del programa no es suficiente para alcanzar la interoperabilidad con otro, el usuario legítimo puede encontrar en las técnicas de ingeniería inversa, y en concreto en la descompilación una herramienta valiosa.

Pero un proceso de descompilación, como se ha visto en el capítulo anterior, conlleva la copia del programa como paso necesario para acceder a la información contenida en sus interfaces. Los actos de reproducción que tienen lugar durante la descompilación constituirán, a priori, una infracción de los derechos del titular salvo que los mismos puedan ampararse bajo una excepción legal.

Los tribunales norteamericanos, también han sido pioneros también en el tratamiento de esta cuestión, en la cual han sido dos los artículos cuya aplicación tuvo cabida: Por un lado, la excepción general del fair use establecida en el artículo 107. Sin embargo, en materia de programas de ordenador, la Copyright Act contiene una

---

<sup>288</sup> Libro Verde de la Comisión, de 16 de julio de 2008, sobre los derechos de autor en la economía del conocimiento (COM (2008) 466 final – No publicado en el Diario Oficial), pág. 4; A., KUR, “Of Oceans, Islands and Inland Water – How much Room for Exceptions and Limitations under the three step test”, Max Planck Institute for Intellectual Property, Competition & Tax Law Research Paper Series No. 08-04, 2008; disponible en: <http://ssrn.com/abstract=1317707> ó <http://dx.doi.org/10.2139/ssrn.1317707> (última consulta 1 de junio, 2013).

<sup>289</sup> G., GEIGER, “Exclusivity in Copyright Law”, JET Law, n.º. 12, 2010, pág. 515, (517).

excepción específica para esta clase de obras, contenida en el artículo 117, cuya finalidad inicial, como veremos, no era garantizar la interoperabilidad entre programas.

### *2.2.1. La excepción de uso limitado del artículo 117.*

#### *A). Fundamento de la excepción.*

El artículo 117 de la Copyright Act fue introducido a través de la Computer Software Copyright Act de 1980<sup>290</sup>, siguiendo las recomendaciones del Informe de la CONTU de limitar determinadas facultades del titular sobre el programa, para permitir a los propietarios de una copia del programa realizar determinados actos de reproducción sin la autorización del titular<sup>291</sup>.

El texto de la norma disponía:

*“Notwithstanding the provisions of section 106, it is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:*

*(1) that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner, or*

*(2) that such new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.*

*Any exact copies prepared in accordance with the provisions of this section may be leased, sold, or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale, or other transfer of all rights in the program.*

---

<sup>290</sup> Computer Software Copyright Act of 1980, Pub. L. No. 96-517, 94 Stat. (1980), pág. 3015.

<sup>291</sup> Sobre la historia legislativa del artículo 117, vid. R.A., KREISS, “Section 117 of the Copyright Act”, *BYU L. Rev.* 4, 1991, pág. 1497 (1508) y M., LOGAN, “The Section 117 adaptation defense and *Krause v. Titleserv. Inc.*: An analysis of the various circuit’s approaches”, *Cardozo Arts & Ent. L.J.* 26, 2008, pág. 269.

*Adaptations so prepared may be transferred only with the authorization of the copyright owner*<sup>292</sup>.

La introducción de esta limitación sobre las facultades exclusivas del titular se fundamentan en la naturaleza funcional del programa. Cuando adquirimos un programa, éste se encuentra fijado en un soporte tangible como puede ser un disco compacto, una memoria USB o cualquier clase de dispositivo que permite a la memoria RAM del ordenador leer el programa para ejecutarlo. Este acto de lectura de la copia desde el soporte tangible a la memoria del ordenador es un acto de reproducción en el sentido de la Copyright Act<sup>293</sup>. Si adquirido el programa, tuviéramos que solicitar permiso al autor o titular para hacerlo funcionar, nos encontraríamos en una situación absurda.

---

<sup>292</sup> Debe tenerse en cuenta que el artículo 117 fue modificado por la Digital Millennium Copyright Act de 1998, de manera que en aquellos casos posteriores a la entrada en vigor de esta norma, el texto objeto de estudio es el que se indica a continuación:

(a) *“Making of Additional Copy or Adaptation by Owner of Copy”*: Notwithstanding the provisions of section 106, it is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:

(1) that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner, or

(2) that such new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.

(b) *(b) Lease, Sale, or Other Transfer of Additional Copy or Adaptation.* — Any exact copies prepared in accordance with the provisions of this section may be leased, sold, or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale, or other transfer of all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner.

(c) *Machine Maintenance or Repair.* — Notwithstanding the provisions of section 106, it is not an infringement for the owner or lessee of a machine to make or authorize the making of a copy of a computer program if such copy is made solely by virtue of the activation of a machine that lawfully contains an authorized copy of the computer program, for purposes only of maintenance or repair of that machine, if—

(1) such new copy is used in no other manner and is destroyed immediately after the maintenance or repair is completed; and

(2) with respect to any computer program or part thereof that is not necessary for that machine to be activated, such program or part thereof is not accessed or used other than to make such new copy by virtue of the activation of the machine.

(d) *Definitions.* — For purposes of this section —

(1) the “maintenance” of a machine is the servicing of the machine in order to make it work in accordance with its original specifications and any changes to those specifications authorized for that machine; and

(2) the “repair” of a machine is the restoring of the machine to the state of working in accordance with its original specifications and any changes to those specifications authorized for that machine.

<sup>293</sup> Así lo confirma la jurisprudencia: “The language of the Copyright Act, case law, and common sense support the proposition that the installation of software onto a computer results in copying with the meaning of the Copyright Act”, en *Stenograph L.L.C. v Bossard Associates, Inc*, 144 F.3d 96, 46 U.S.P.Q.2d (BNA) 1936 (D.C. Cir. 1998), pág. 1936.

Dos son las condiciones que, de forma alternativa, constituyen el presupuesto objetivo de la limitación:

- 1º La copia o adaptación ha de constituir un paso esencial para la utilización del programa en conjunción con la máquina y no ser usada de ningún otro modo,
- 2º O bien, la copia o adaptación se utilizará únicamente para fines de archivo y todas las copias a estos efectos serán destruidas en el caso de que la posesión de la copia sobre el programa deje de ser legítima.

En cuanto al requisito subjetivo, la persona autorizada por la limitación es el “propietario” (“owner”) de la copia del programa.

*B) Requisitos subjetivos: la persona autorizada.*

Según el texto del artículo, la persona autorizada para realizar los actos excepcionados es el propietario “owner” de la copia. La primera incógnita a despejar es si se refiere a propietario en el sentido real del término o no.

Lo cierto es que la recomendación del Informe de la CONTU no utilizaba el término propietario sino poseedor legítimo<sup>294</sup>, sin embargo el Congreso empleó el término “owner”, lo cual ha dado lugar a bastantes ambigüedades interpretativas por parte de los tribunales.

Algún autor como KREISS ha indicado que la diferencia de criterios interpretativos planteaba dos importantes cuestiones:

- 1º Si el término “owner” ha de circunscribirse a quienes adquieren la copia física del programa a través de un título formal o también incluye a los licenciatarios y usuarios de una copia del programa.
- 2º Si cualquier persona en la cadena de distribución del programa es “owner”.

Para KREISS, nada en la Copyright Act indica que el Congreso, cuando modificó el artículo 117, al utilizar el término “owner”, tuviera la intención de incluir en su acepción a los licenciatarios, usuarios o otra clase de poseedores de una copia del

---

<sup>294</sup> Final Report of (...), Chapter 3, 1978, op.cit. págs. 12-13



programa, decantándose por una interpretación de “owner” propia de los derechos reales<sup>295</sup>.

A nuestro juicio, la utilización del término “owner” en la excepción es una desafortunada elección del legislador norteamericano. Se parta de una concepción más europeísta o más anglosajona del concepto de copyright, la conclusión es que se trata de un derecho de propiedad, por lo tanto, no tiene ningún sentido limitar legalmente las facultades de este derecho de propiedad en beneficio de la generalidad, si dicha limitación no limita al propietario, sino que le concede más facultades. Es más, el Tribunal Supremo ha indicado respecto de la interpretación de la Copyright Act, que cuando el Congreso haga uso de términos que puedan haber adquirido un significado determinado en el ámbito del common law, salvo que la norma que contenga dicho término lo indique de otro modo, el juez deberá interpretar tal término en el sentido de los usos del sistema.<sup>296</sup> Por lo tanto, el término “owner” del artículo 117 debiera entenderse y sustituirse por el de “legítimo usuario”.

Dos fueron los casos que en 1983, al poco de entrar en vigor la norma trataron esta cuestión.

Para el Tribunal de Distrito de Idaho, en el caso *Hubco Data Products Corp. v. Management Assistance, Inc.*, el término “owner” había de ser interpretado de manera literal. Por lo tanto, la limitación del artículo 117 no tiene cabida porque “*Hubco is not the “owner” of the higher-level object code copy contained in the software package*<sup>297</sup>.”

Esta interpretación conduciría a la imposibilidad de aplicación de la excepción, porque el único “owner” en el sentido real del término será el titular del copyright o el autor del programa. Así, poco después, otro Tribunal de Distrito, en Illinois, interpretó el término “owner” en sentido amplio, declarando que “ (*...*) *lawful owners of copyrighted computer programs are permitted to adapt their programs for use in conjunction with a machine. 17. U.S.C. § 117. This necessary limitation on the right of copyright holders is intended to permit owner of copies to make the programs*

---

<sup>295</sup> R.A., KREISS, “Section 117...”, *op.cit.*, pág. 1536.

<sup>296</sup> “Where Congress uses terms that have accumulate settled meaning under the common law, a court must infer, unless the statute otherwise dictates, that Congress means to incorporate the established meaning of these terms”. *Community for Creative Non-Violence v. Reid*, 490 U.S. 730 (1989) pág. 739.

<sup>297</sup> *Hubco Data Products Corp. v. Management Assistance, Inc.* (219 U.S.P.Q. (BNA) 450 (D. Idaho, 1983), pág. 450.

*compatible with their computer hardware or systems*”. Es decir, entendiendo por propietario de la copia, a efectos de la excepción, a la persona que legítimamente hubiese adquirido una copia del programa<sup>298</sup>.

Sin embargo, la jurisprudencia norteamericana consolidó la postura del tribunal de Idaho en el caso *Mai v. Peak*<sup>299</sup>. El Tribunal de Apelaciones del Noveno Circuito declaró que el ámbito subjetivo del artículo 117(a) solo alcanzaba al propietario de la copia, pero no al licenciatario de la misma<sup>300</sup>. Aunque el caso *Mai v. Peak* fue especialmente comentado por la cuestión que se planteó ante el Tribunal. Esta versaba sobre la determinación de si la carga de un programa de ordenador en la memoria RAM por el equipo técnico de reparación constituía una copia que infringía el copyright sobre el programa. Posiblemente por eso, la declaración sobre la persona autorizada en la limitación del artículo 117 quedó en un segundo plano, pero su relevancia fue de más calado que la propia cuestión de fondo.

Tanto fue así, que la reforma del artículo 117 en 1998 con el Título III de la Digital Millennium Copyright Act (DMCA)<sup>301</sup> tuvo entre sus razones rechazar la estricta interpretación efectuada por el Noveno Circuito del término “owner” en el caso *Mai v. Peak*.

No fue hasta 2005 y con el nuevo texto del artículo 117 vigente, que el caso *Krause v. Titleserv.*<sup>302</sup> inició el cambio del estricto criterio de interpretación establecido por el Tribunal de Apelaciones del Noveno Circuito en el caso *Mai v. Peak*.

En *Krause v. Titleserv.*, el Tribunal de Apelaciones del Segundo Circuito afirmó que la tenencia de un título formal no puede ser un requisito necesario para ser considerado “owner” de una copia del programa y poder acogerse a la limitación del artículo 117.

---

<sup>298</sup> *Midway Mfg. Co. v. Strohon* 564 F.Supp. 741 (D.C.Ill.,1983).

<sup>299</sup> *MAI Sys. Corp. v. Peak Computer, Inc.*, 991 F.2d 511, 26 U.S.P.Q.2d (BNA) 1458 (9th Cir. 1993), *cert. dismissed*, 510 U.S. (1994), pág. 1033.

<sup>300</sup> *Ibidem*. El apartado 32 de la sentencia disponía: “(...) In the absence of ownership of the copyright or express permission by license, such acts (refiriéndose a la copia en la memoria RAM del dispositivo electrónico) constitute copyright infringement”.

<sup>301</sup> Sobre el texto resultante de la modificación, vid. supra. epígrafe 2.2.1 Para un comentario sobre esta cuestión vid. J., BAND, “The Digital Millennium Copyright Act”, ALA Washington Office, 1998, pág. 11, disponible en: <http://www.ala.org/advocacy/copyright/dmca> (última consulta 1 de junio, 2013).

<sup>302</sup> *William KRAUSE dba Special-T software v. TITLESERV, Inc. et al.*, 402 F.3d 119 (2d Cir. 2005); (*cert. denied*), S.Ct. 622 (2005), pág. 126.

El Tribunal justifica su interpretación argumentando las razones por las cuales el Congreso no empleó el término “rightful possessor” contenido en el Informe de la CONTU: “(...) Congress’s decision to reject “rightful possessor” in favour of “owner” does not indicate an intention to limit the protection of the statute to those possessing formal title. The term “rightful possessor” is quite broad. Had that term been used, the authority granted by the statute would benefit a messenger delivering a program, a bailee, or countless others temporarily in lawful possession of a copy. Congress easily could have intended to reject so broad a category of beneficiaries without intending a narrow, formalistic definition of ownership dependent on title”.

Además, añade que la interpretación del término “owner” en un sentido real iría directamente en contra de la doctrina del Tribunal Supremo por dos razones:

1ºLa determinación de si existe un título formal no es objeto del derecho federal sino del derecho estatal. Por tanto, exigir tal título para considerar a una persona “owner” de la copia de un programa iría en contra de objetivo de la Copyright Act de crear un derecho nacional uniforme, establecido en la doctrina del Tribunal Supremo<sup>303</sup>.

2ºSería inverosímil que un usuario cuyo grado de propiedad sobre una copia del programa le permite legítimamente usarla o guardarla para siempre, o si prefiere, tirarla a la basura, no pudiera estar autorizado para arreglarla si contiene un error, o para hacer una copia de seguridad.

Desde 2010 el Tribunal de Apelaciones del Noveno Circuito nuevamente, con el caso *Verno v. Autodesk*. A juicio del Tribunal, ha vuelto a arrojar sombras sobre el significado de “owner” a efectos del artículo 117.

Según el Tribunal, antes de decidir si una persona puede calificarse como “owner” en el sentido de la excepción, deben examinarse tres factores:

1ºSi el titular del copyright especifica que al usuario del programa se le concede una licencia de uso.

2ºSi el titular del copyright restringe o no al usuario la capacidad para transferir la copia del programa.

---

<sup>303</sup> “express objective of creating national, uniform copyright law by broadly preempting state statutory and common-law copyright regulation”. *Community for Creative Non-Violence v. Reid*, 490 U.S. 730 (1989) pág. 740.

3º Si el titular del copyright impone restricciones de uso importantes o banales.

Si el resultado general de este examen concluye que el usuario tiene restringidas sus facultades de uso y disfrute del programa, no podrá considerarse “owner” en el sentido del artículo 117<sup>304</sup>.

A la vista de los anteriores criterios jurisprudenciales, parece que el único punto donde existe unanimidad es respecto a la innecesariedad de contar con un título formal para ser calificado como “owner”. En cuanto al significado del término, sin embargo, no hay criterio unánime.

En mi opinión, la doctrina del Tribunal Supremo sobre interpretación de los términos de la Copyright Act, unida a la historia legislativa del artículo 117 de la Copyright Act debiera conducirnos a concebir “owner” en sentido amplio, entendiendo por tal a cualquier usuario legítimo de una copia del programa. Pero la doctrina de alguno de los Tribunales de Apelación del Circuito ha optado por una interpretación mucho más restrictiva del término, de manera que la condición de “owner” habrá de inferirse de las cláusulas del contrato de adquisición del programa.

A falta de un pronunciamiento por parte del Tribunal Supremo, el criterio del Tribunal del Noveno Circuito es el que impera y el que ya ha aplicado algún tribunal de distrito<sup>305</sup>.

*C) Requisitos objetivos: la conducta permitida.*

El artículo 117 menciona dos clases de actos que están amparados por la limitación: la copia y la adaptación. Tales actos pueden realizarse con dos fines, porque sean necesarios para la utilización del dispositivo electrónico con el programa, o para disponer de una copia de seguridad del programa.

A efectos de la descompilación es el primero de los fines permitidos el que interesa, es decir, la realización de copia o adaptación como paso esencial para la utilización del programa en conjunción con la computadora. El texto literal del apartado (1) del artículo 117 dispone: “*that such a new copy or adaptation is created as an essential step in the utilization of the computer program in the utilization of the*

---

<sup>304</sup> Verno v. Autodesk, Inc. 621 F.3d 1102 (9th Cir. 2010), pág. 1111.

<sup>305</sup> Burroughs Payment Systems, Inc. v. Symco Group, Inc. Case C-11-06268 JCS (2012 U.S. Dist.) Lexis 67198.

*computer program in conjunction with a machine and that it is used in no other manner”.*

Las dudas interpretativas se han centrado en dos aspectos: por un lado qué constituye un paso esencial para la utilización del programa en conjunción con la máquina, y por otro, el alcance de la expresión, “*que no se use de ningún otro modo*”.

*a). El acto constituye un paso esencial.*

El primer caso donde, debido a la descompilación del código, el tribunal examinó el significado de “paso esencial” fue *Vault Corp. v. Quaid Software Ltd.*<sup>306</sup>, en 1988.

El Tribunal de Apelaciones del Quinto Circuito tuvo que valorar si la carga del programa en la memoria, cuya finalidad no era solamente usar el programa de la manera ideada por su autor, sino analizarlo, era un acto amparado por la limitación del artículo 117(a)(1).

En esta ocasión, el Tribunal, para examinar si tal acto constituía “un paso esencial”, partió de la indicaciones del Informe de la CONTU sobre la autorización de actos de reproducción sobre el programa. Según el Informe, debiera articularse un derecho que permitiese realizar los cambios necesarios para usar un programa que hubiese sido vendido y comprado. La conversión de un programa desde un lenguaje de alto nivel a otro a fin de facilitar su uso, debiera entrar en el ámbito de tal derecho, así como la facultad de añadir funciones al programa que no existiesen al tiempo de su adquisición. El ejercicio de estas facultades debiera permitirse siempre y cuando no dañase los intereses del titular del copyright<sup>307</sup>.

A la vista de lo dispuesto en el Informe, parece que la consideración de lo que constituya un paso esencial es bastante amplia, y en esa línea se mantuvo el Tribunal de Apelaciones del Quinto Circuito al declarar que la copia realizada por Quaid del programa de Vault, para el concreto fin de desarrollar un modo de evadir las medidas

---

<sup>306</sup> *Vault Corp. v. Quaid Software Ltd.*, 655 F. Supp. 750 (E.D. La. 1987), *aff'd*, 847 F.2d, 7 U.S.P.Q.2d (BNA) 1281 (5th Cir. 1988).

<sup>307</sup> *Final Report of ...*, *op.cit.* pág. 13.

de protección del mismo<sup>308</sup>, constituía un paso esencial que estaba permitido por el artículo 117(a)(1) de la Copyright Act.

Sin embargo, esta exégesis del tribunal parece excesiva para una limitación, que a priori debiera ser interpretada de forma restrictiva. Así se pronunció en 1992 el Tribunal de Apelaciones del Noveno Circuito en el caso *Sega v. Accolade*<sup>309</sup>, cuando declaró que el artículo 117(a)(1) no protege a un usuario que descompila el código objeto, lo convierte en código fuente, e imprime y fotocopia la versión descompilada del código fuente<sup>310</sup>, sino que este artículo define una concreta clase de copia que es legítima per se<sup>311</sup>.

Poco duró la doctrina del Noveno Circuito, que fue refutada por el Segundo Circuito en 1995 al afirmar que los actos modificativos del programa (los cuales conllevaban la realización de copia en memoria, análisis y adaptación del código) realizados por el demandado no vulneraban el copyright del titular del programa porque los mismos constituían un paso esencial para la utilización del programa<sup>312</sup>.

Este criterio es el que sigue manteniendo el Tribunal de Apelaciones del Segundo Circuito, confirmado en el caso *Krause v. Titleserv*<sup>313</sup>. Lo más interesante de la resolución es el análisis que el Tribunal realiza de las cuatro clases de modificaciones realizadas por el demandado a la copia del programa: Corrección de errores (debugging); cambios en el código fuente; adaptación del programa a un sistema operativo para el que no fue diseñado originalmente; y adición de funciones.

En relación a la corrección de errores, el Tribunal aplicando la doctrina mantenida de la Sala, afirma que este acto se hace para que el programa pueda funcionar, por tanto, constituye un paso esencial en el sentido del artículo 117(a)(1); en relación a los cambios en el código fuente que tienen como objetivo añadir nuevos clientes al entorno del programa, el argumento es idéntico, y por tanto, el acto de adaptación constituye un paso esencial.

---

<sup>308</sup> “express purpose of devising a means of defeating its protective function”. Vid. *Vault v. Quaid*, *op.cit.*, pág. 261.

<sup>309</sup> *Sega Enterprises Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992).

<sup>310</sup> *Ibidem*, pág. 1520.

<sup>311</sup> *Ibidem*, pág. 1521.

<sup>312</sup> *Aymes v. Bonelli*, 47 F.3d 23, (2d. Cir. 1995), 980 F.2d 857 (2d Cir. 1992).

<sup>313</sup> *William KRAUSE dba Special-T software v. TITLESERV, Inc. et al*, *op.cit.*

Respecto del tercer tipo de modificación, la adaptación a otro sistema operativo, el tribunal la acepta como paso esencial, pero no en base a un análisis del acto en sí mismo, sino por aplicación de una presunción procesal de prueba.

Sobre la adición de nuevas funciones, el Segundo Circuito afirma categóricamente que esta clase de cambios en el programa no son estrictamente necesarios para mantener la funcionalidad del programa, sin embargo, a continuación matiza esta afirmación en base a que la normativa es ambigua, ya que, por un lado, el término “esencial” se utiliza en ocasiones como sinónimo de “necesario”, y por otro lado, porque la frase “an essential step in the utilization of the computer program” depende del significado del término “utilization”. Este término es definido por el diccionario como “to make useful”, “turn to profitable account or use”, “make use of”, y “conver to use”. Por lo tanto, aunque el término “essential” sea objeto de una interpretación gramatical, el artículo 117(a)(1) es ambiguo porque no indica claramente *para qué fin* la modificaciones deben considerarse absolutamente necesarias.

El otro argumento que utiliza para justificar su opinión el Tribunal de Apelaciones del Segundo Circuito son las recomendaciones del Informe de la CONTU. Lo llamativo es que el Tribunal cita un pasaje del Informe donde la Comisión, en 1978, mencionaba que la ausencia de estandarización entre los lenguajes de programación y el hardware de la industria informática son razones más que sobradas para permitir que un legítimo adquirente de una copia del programa tenga que adaptar el programa para que pueda funcionar<sup>314</sup>.

Lo que queda claro a la vista de la evolución jurisprudencial, es que, hoy por hoy, no existe un criterio claro para determinar qué puede considerarse un paso esencial en el sentido de la limitación, con lo cual parece que los actos de ingeniería inversa a efectos de interoperabilidad, si consisten en actos de copia o adaptación del programa, pueden ampararse en esta limitación siempre que pueda demostrarse ante un tribunal que los mismos constituyen un paso esencial para la utilización del programa en conjunción con la máquina.

---

<sup>314</sup> *Final Report of ...*, *op.cit.*, pág. 13.

b). La expresión “que no se use de ningún otro modo”.

El otro tema controvertido en relación a los requisitos objetivos de esta limitación, es el significado de la expresión “*it is used in no other manner*”, es decir que la copia no se use de ningún otro modo al prescrito en epígrafe 1 del artículo 117(a).

Sin embargo a la vista del laxo criterio jurisprudencial para valorar qué constituye un paso esencial en el contexto de la limitación poco se puede añadir. Es más, en algunos de los casos más representativos antes citados, como *Vault v. Quaid*, o *Mai v. Peak*, las salas ni siquiera entraron a examinar este requisito. El Tribunal de Apelaciones del Segundo Circuito, en el caso *Krause*, sí lo hace y concluye que “*el uso de ningún otro modo*” depende del tipo de uso previsto en la creación del programa<sup>315</sup>.

En opinión de algún autor, esta interpretación del Segundo Circuito supone volver a valorar la esencialidad del acto, lo cual iría contra el propio espíritu de la norma. Si se atiende a una interpretación gramatical del texto de la norma, el pronombre “it”, en la frase “*it is used in no other manner*”, se refiere a la copia o adaptación del programa<sup>316</sup>. Por lo tanto, el requisito parece referirse a que ninguno de estos dos actos se usen en otro modo más allá de la utilización del programa en conjunción con la máquina, lo cual poco parece tener que ver con la interpretación realizada por el Tribunal Apelaciones en este caso.

A la vista de los criterios jurisprudenciales hasta aquí examinados, parece que la redacción del artículo 117(a)(1) permite una interpretación que va más allá del objetivo para el que fue concebida esta excepción. Como indicó el Tribunal de Apelaciones del Noveno Circuito en el caso *Sega v. Accolade*, el objetivo del artículo 117 es definir una concreta clase de copia como legítima per sé. Es en el artículo 107, donde se establece una excepción a actos que a priori constituirían una infracción de copyright<sup>317</sup>.

Por lo tanto, los actos de descompilación, incluso a efectos de interoperabilidad no debieran poder ampararse en esta norma de la Copyright Act, sino únicamente en la excepción del *fair use*.

---

<sup>315</sup> William KRAUSE dba Special-T software v. TITLESERV, Inc. et al, *op.cit.*, pág. 129.

<sup>316</sup> M., LOGAN, “The Section 117 adaptation...”, *op.cit.*, pág. 291.

<sup>317</sup> *Sega Enterprises Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992), pág. 1521.



### 2.2.2. La excepción del fair use del artículo 107.

El artículo 107 de la Copyright Act configura el uso leal o fair use como excepción a los derechos del titular del copyright<sup>318</sup>.

Literalmente dispone:

*“Notwithstanding the provisions of sections 106 and 106A, the fair use of a copyrighted work, including such use by reproduction in copies or phonorecords or by any other means specified by that section, for purposes such as criticism, comment, news reporting, teaching (including multiple copies for classroom use), scholarship, or research, is not an infringement of copyright. In determining whether the use made of a work in any particular case is a fair use the factors to be considered shall include—*

*(1) the purpose and character of the use, including whether such use is of a commercial nature or is for non-profit” educational purposes;*

*(2) the nature of the copyrighted work;*

*(3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and*

*(4) the effect of the use upon the potential market for or value of the copyrighted work.*

*The fact that a work is unpublished shall not itself bar a finding of fair use if such finding is made upon consideration of all the above factors.*

En la norma se establecen cuatro requisitos que deben ser valorados por el tribunal en conjunto a fin de determinar si un acto a priori constitutivo de infracción puede considerarse un uso leal de la obra y por tanto debe ser soportado por el titular del derecho. Estos requisitos son:

- 1.El propósito y el carácter del uso, incluyendo si tal uso tiene finalidad comercial u otra finalidad no comercial.
- 2.La naturaleza de la obra protegida por copyright.

---

<sup>318</sup> R., OKEDIJI, “Toward an International Fair Use Doctrine”, Columbia Journal of Transnational Law, vol. 39, 2000, pág. 75 (89). En palabras de COLOMBET, “un sistema abierto de limitaciones se basa en un enunciado general y no en una lista exhaustiva de actos no constitutivos de violación de derechos. No existe una lista de limitaciones, sino una cláusula general de la cual se derivan los límites al derecho de autor”, en C. COLOMBET, traducido del francés por P. ALMEIDA *Grandes principios del Derecho de Autor y los Derechos Conexos en el Mundo. Estudio de Derecho Comparado*, UNESCO/CINDOC, 1997, pág. 50.

3.La cantidad y sustancialidad de la parte de la obra utilizada en relación con la obra en total.

4.El efecto del uso en el mercado potencial sobre el valor de la obra protegida.

Hasta la modificación del artículo 107 en Octubre de 1992<sup>319</sup> la defensa del *fair use* no se había aplicado a los actos de ingeniería inversa sobre programas de ordenador. Pero la modificación operada por la ley de 1992 añadió un párrafo al final del artículo 107 a través del cual se permitía a las obras no publicadas, como podría predicarse del código objeto de un programa de ordenador, ampararse bajo la excepción del fair use. El párrafo establecía: “*The fact that a work is unpublished shall not itself bar a finding of fair use if such finding is made upon consideration of all the above factors*”.

Ahora bien, el espíritu de la modificación no era abrir una puerta a la utilización del fair use como defensa ante actos de ingeniería inversa sobre programas de ordenador, tal como declaró el Senador Simon, proponente de la misma: “*this bill is not intended to provide a fair use access to invisible and thus unpublished computer code through decompilation*”<sup>320</sup>. El informe del Comité del Senado para Asuntos Judiciales confirmaba esta postura al declarar “nothing in the bill is intended in any way to broaden fair use of unpublished computer programs”<sup>321</sup>. Es más, como han indicado varios autores, el Congreso se preocupó por recordar el fundamento sobre el que descansa la doctrina del fair use al aprobar la modificación de 1992, al afirmar que esta excepción depende de las circunstancias particulares de cada caso. No puede definirse el fair use en una única regla, ni puede determinarse a través de un hecho aislado. Además para decidir si una situación constituye fair use, han de valorarse todos los factores de la norma en conjunto<sup>322</sup>.

---

<sup>319</sup> Sección 107 de la Copyright Act, modificada el 24 de Octubre de 1992, Pub.L. 102-492, 106 Stat. 3145. La modificación llevada a cabo en 1992 iba dirigida a las obras no publicadas, de manera que “shall not itself bar a finding of fair use if such finding is made upon consideration of all the above factors”.

<sup>320</sup> Vid. 137 CONG.REC.S5648 (daily ed. may 9, 1991)

<sup>321</sup> Vid. 137 CONG.REC.S13925 (daily ed. Oct. 27, 1991)

<sup>322</sup> K.D., CREWS, “Fair Use of Unpublished Works: Burdens of Proof and the Integrity of Copyright”, Arizona State Law Journal, vol. 31, núm.1, 1999, pág. 1 (53); J., ADKINS, “Fair Use and Unpublished Works: Public Law 102-492-An Amendment to the Fair Use Statute”, St. Louis U.L.J., vol. 38, (1993), pág. 231; J., LEMAN, “The Future of Unpublished Works in Copyright Law After the Fair Use Amendment”, Iowa J.Corp.L., 18, 1993, pág. 619.

Sin embargo, un mes después de la entrada en vigor de la modificación comentada, un tribunal aplicó la excepción del fair use a una situación de ingeniería inversa de software, lo cual supuso una revolución para el sector de la informática.

Fue en el caso Atari v. Nintendo en 1992<sup>323</sup> ante el Tribunal de Apelación del Circuito Federal. La cuestión planteada consistía en determinar si la copia del código de un videojuego durante un procedimiento de ingeniería inversa cuya finalidad era aprender cómo crear un nuevo programa que pudiese funcionar con el descompilado constituía una infracción del copyright o podía considerarse un uso leal de la obra.

Las dos partes enfrentadas eran competidores en el sector de los videojuegos informáticos. Nintendo empleaba un programa de ordenador (10NES) embebido en un microchip de su sistema de juego para bloquear el uso de su videoconsola a los competidores. Para obtener una copia de ese programa contenido en el microchip, Atari negoció una licencia con Nintendo, pero los términos del acuerdo eran lo suficientemente estrictos como para no permitir el acceso de Atari al código del programa de seguridad de Nintendo. El interés por obtener ese código era tal que Atari llegó a mentir ante la Oficina de Copyright norteamericana para obtener una copia del código fuente del programa. Con esa información, Atari desarrolló su propio programa, "Rabbit", que permitía desbloquear la consola de Nintendo. Aunque "Rabbit" fue elaborado usando un microprocesador diferente y un lenguaje de programación distinto, el flujo de datos a la consola de Nintendo se producía de la misma forma que en los videojuegos creados por Nintendo. De este modo, Atari podía vender videojuegos compatibles con la consola de Nintendo sin violar las condiciones de su acuerdo de licencia. Nintendo sin embargo demandó a Atari por infringir su derecho de copyright sobre su programa de ordenador, y Atari contestó que las copias realizadas del programa constituían fair use.

En apelación, el Tribunal de Apelación del Circuito Federal siguió la doctrina mantenida por el Tribunal de Apelaciones del Noveno Circuito respecto del ámbito de protección del programa<sup>324</sup>. En primer lugar estableció la parte del programa objeto del

---

<sup>323</sup> Atari Games Corp. v. Nintendo of America Inc., 975 F.2d 832 (Federal Cir., 1992). Un comentario detallado de la sentencia puede encontrarse en J., BAND, M., KATOH, *Interfaces on Trial...*, *op.cit.*, pág. 212 y ss.

<sup>324</sup> Atari Games Corp. v. Nintendo of America Inc, *op.cit.*, pág. 840. La doctrina del Noveno Circuito a este respecto ya estaba en consonancia con la establecida poco después por el Segundo Circuito a través

litigio que constituía idea y la parte que constituía su expresión. Del examen concluyó que la reproducción de la copia adquirida de forma ilegítima por el demandado constituía infracción del copyright. Sin embargo, tras esta afirmación, pasó a considerar la circunstancia de la ingeniería inversa como condicionante para tomar en cuenta la excepción del fair use. Para ello, el tribunal, en lugar de realizar un examen detallado de cada uno de los cuatro requisitos cumulativos contenidos en el artículo 107 de la Copyright Act, se limitó a valorar los principios generales de la doctrina del fair use, haciendo un breve análisis del segundo requisito, referido a la naturaleza de la obra. A este respecto afirmó que los programadores no podían entender el código objeto de un programa sin descompilarlo, por lo que este proceso era indispensable para facilitar el entendimiento de las ideas y procesos del programa y constituía fair use. Ahora bien, la sala matizó que el fair use de la descompilación se limitaba a entender los elementos no protegidos de la obra y recordó que consentir los actos de reproducción del código no era una invitación a la piratería<sup>325</sup>.

Poco tiempo después, el Tribunal de Apelaciones del Noveno Circuito, en el caso *Sega v. Accolade*<sup>326</sup>, tras un examen detallado de cada una de las condiciones de la excepción, confirmó que constituye fair use la actuación de una persona que no es titular del copyright ni licenciataria del mismo pero que tiene un interés legítimo, como es entender el funcionamiento del programa descompilando el código, siempre y cuando no exista otro medio para acceder a los elementos funcionales del programa (que no son objeto de protección por copyright). En este caso, la empresa Accolade había adquirido legalmente una consola de Sega (Genesis) y tres videojuegos. Instalaron un descompilador en la consola y al reproducir los videojuegos, imprimieron los resultados producidos por el descompilador para estudiar y analizar las especificaciones de las interfaces de la consola. Tras este estudio, elaboraron un manual donde exponían las descripciones funcionales de las interfaces, pero no el código fuente del programa de Sega.

---

del test *Altai*, es decir, una postura limitativa del ámbito de protección basada en el análisis de elemento por elemento en el caso concreto.

<sup>325</sup> *Ibidem*, pág. 843: “Fair use to discern a work’s ideas...does not justify extensive efforts to profit from replicating protected expression...The fair use reproductions of a computer program must not exceed what is necessary to understand the unprotected elements of the work. This limited exception is not an invitation to misappropriate protectable expression”.

<sup>326</sup> *Sega Enterprises Ltd. v. Accolade, Inc.*, *op.cit.*

Dos condiciones son las que han de cumplirse para que la copia no autorizada del código en los actos de ingeniería inversa se considere fair use:

1º Debe constituir el único modo de tener acceso a las ideas y elementos funcionales embebidos en el programa. (Regla de la necesidad del método o de la indispensabilidad)

2º Ha de existir una causa legítima para acceder a esa información funcional.

Esta regla establecida por el Noveno Circuito sólo protege los actos de descompilación en sí mismos considerados, siguiendo la línea jurisprudencial iniciada en el caso Atari. Es decir, si quien realiza la copia y descompila el código, pone después en el mercado un programa nuevo que contiene la expresión del programa descompilado, su actuación no estaría amparada por la doctrina del fair use y constituiría una infracción del copyright<sup>327</sup>.

El tercer caso que aporta claves en el examen de la doctrina del fair use como garante de la interoperabilidad informática fue *Bateman v. Mnemonics*<sup>328</sup>. El asunto de fondo no versaba sobre videojuegos, sino sobre un sistema operativo utilizado en sistemas de aparcamiento automáticos.

*Bateman*, empresa dedicada a fabricar sistemas de aparcamiento, había desarrollado un dispositivo electrónico y un sistema operativo y los comercializaba juntos. *Mnemonics* quería desarrollar programas interoperables con el dispositivo propiedad de *Bateman*, y para ello quiso negociar un acuerdo de licencia con *Bateman* para acceder al código fuente del programa que controlaba dicho dispositivo. *Bateman* se negó y *Mnemonics* adquirió un ejemplar en el mercado, y descompiló el software para estudiar sus interfaces y poder desarrollar programas interoperables. Este fue el detonante para que *Bateman* demandase a *Mnemonics* por infracción de copyright.

En relación a la infracción del copyright, en primera instancia, el veredicto del jurado dio la razón a *Bateman*, pero *Mnemonics* apeló, y el Tribunal de Apelaciones del Undécimo Circuito reexaminó las dos cuestiones fundamentales que se habían

---

<sup>327</sup> NIMMER citando la doctrina del Tribunal Supremo “Every commercial use of copyrighted material is presumptively an unfair exploitation of the monopoly privilege that belongs to the owner of the copyright”. *Sony Corp. v. Universal City Studios, Inc.*, 464 U.S. 417, 451 (1984) en *M., NIMMER, D., NIMMER, Nimmer on ..., op.cit.*, § 13.05 [D][4], pág.13-232.

<sup>328</sup> *Bateman v. Mnemonics*, 79 F. 3d 1532 (11th Cir. 1996).

planteado ante el Tribunal de Distrito: La adecuación de la metodología empleada para examinar las reivindicaciones de infracción de copyright, y si las especificaciones de las interfaces eran elementos protegidos por el copyright sobre el programa.

Para hacer dicho examen, el tribunal de apelación se centró en como se había llevado a cabo la apreciación de similitud sustancial entre el programa desarrollado por el Mnemonics y los elementos copiados del programa de Bateman. En el centro del análisis de esta cuestión se encontraban las instrucciones que el juez de instancia había dado al jurado en relación al ámbito de protección del copyright sobre un programa de ordenador.

El tribunal del apelación concluyó que las instrucciones dadas al jurado habían sido erróneas porque no se les había indicado que tuvieran en cuenta la conexión entre el test de Altai y los retos al requisito de la originalidad, como las doctrinas de la fusión, *scenes a faire*, la existencia de técnicas o prácticas estandarizadas, consideraciones sobre la eficiencia, requisitos de compatibilidad, y similares<sup>329</sup>.

Declarado esto, el tribunal se toma la libertad de valorar si el uso hecho por el demandado de la información contenida en las interfaces constituía *fair use*, aunque esta cuestión no formaba parte de la apelación. Por ello, el Undécimo Circuito lleva este examen a una nota al pie cuyo análisis se hace obligado.

La razón que alega para justificar su actuación es que se trataba del primer caso sobre ingeniería inversa a efectos de interoperabilidad que no versaba sobre un videojuego y existía bastante incertidumbre sobre la cuestión. Por lo tanto, a fin de establecer una guía para los tribunales de distrito, el tribunal entra a analizar la legitimidad de la descompilación realizada por Mnemonics. En realidad se limita a

---

<sup>329</sup> *Bateman v. Mnemonics*, *op.cit.*, pág. 1547: “Where the district court erred, however, was in not requiring the jury to consider the connection between the Altai test and the challenges to originality, such as merger, *scenes a faire*, standard technique or practice, considerations of efficiency, compatibility requirements, and the like. Since the district court instructed the jury that Altai filtration was limited to nonliteral copying, the jury must have concluded that any instances of literal copying of Bateman's code by PAC were by definition acts of copyright infringement. This conclusion is a manifest distortion and misstatement of the law. There was substantial evidence presented of the literal copying of code in this case. Much of the testimony of the expert witnesses dealt with the Nimmer test and its role as the proper methodology to employ in analyzing the copyright infringement claims of Bateman and Fricker. When this testimony was combined with the jury instruction limiting filtration analysis to nonliteral copying, it is clear that the jury was misled and did not fully understand the law of copyright relevant to this case. Under the standard established in *Miller*, we are “left with a substantial and ineradicable doubt as to whether the jury was properly guided in its deliberations. (650 F.2d at 1372)”.

recoger la doctrina establecida por el Noveno Circuito en los casos Sega y Atari, declarando que “*we find the Sega opinion persuasive in view of the principal purpose of copyright – the advancement of science and the arts*”<sup>330</sup>.

De este modo, el tribunal devuelve el caso al tribunal de instancia indicando de forma directa que un uso de la ingeniería inversa para conseguir interoperabilidad, como el realizado por Mnemonics, ha de considerarse fair use.

En el año 2000 fue el Tribunal de Apelaciones del Noveno Circuito quien reexaminó la cuestión de la licitud de la ingeniería inversa en el caso Sony Computer Entertainment, Inc. v. Connectix Corp.<sup>331</sup>, de nuevo en el sector de los videojuegos.

---

<sup>330</sup> Por la relevancia de esta nota al pie, se incluye completa a continuación. “Note 18: As noted in note 17, supra, there is an important technical distinction between the source code and the object code. Generally, the object code is readable only by machines. But, as Professor Nimmer notes: The skilled programmer, however, may reverse engineer the program by using the techniques of “decompilation” and “disassembly” to work backwards from object code and to produce a facsimile of the original source code. Although decompilation techniques generally will not yield a perfect copy of the original source code, decompilation can afford an understanding of much of the structure and operation of a program.3 Nimmer § 13.03[F] n. 271 (citation omitted). PAC utilized the techniques of disassembly and decompilation on Bateman's SBCOS. PAC claims that such reverse engineering was necessary to develop an operating system compatible with its application program, which in turn was written to be compatible with Bateman's SBCOS. Although the issue is not directly before us on appeal, it will likely appear on remand, and thus we think it proper to address it and thereby provide some guidance to the district court on remand. See United States v. Costa, 31 F.3d 1073, 1080 (11th Cir.1994). And although there has been some uncertainty as to whether reverse engineering constitutes copyright infringement, the one federal circuit court that has squarely addressed the issue has concluded that reverse engineering may be a fair use. See Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1527-28 (9th Cir.1992) (holding that “where disassembly is the only way to gain access to the ideas and functional elements embodied in a copyrighted computer program and where there is a legitimate reason for seeking such access, disassembly is a fair use of the copyrighted work, as a matter of law”). Another federal circuit court has addressed the issue, although its highly unusual factual posture seems to limit its precedential value. See Atari Games Corp. v. Nintendo of America Inc., 975 F.2d 832, 843 (Fed.Cir.1992) (concluding that “reverse engineering object code to discern the unprotectable ideas in a computer program is a fair use,” although denying the defendant's fair use claim, based on the fact that it was wrongfully in possession of the source code). We find the Sega opinion persuasive in view of the principal purpose of copyright—the advancement of science and the arts. See Feist Publications, Inc. v. Rural Tel. Serv. Co., 499 U.S. 340, 348, 111 S.Ct. 1282, 1290, 113 L.Ed.2d 358 (1991) (noting that “[t]he primary objective of copyright is not to reward the labor of authors, but ‘[t]o promote the Progress of Science and useful Arts’”) (quoting U.S. Const. art. I, § 8, cl. 8). We note that we are in good company in endorsing the approach of the Sega court. Eleven copyright law professors filed an amicus curiae brief in the Sega case in which they endorsed this approach. See Kreiss, at 4 n. 9 (citing Brief Amicus Curiae [sic] of Eleven Copyright Law Professors in Sega Enterprises Ltd. v. Accolade, Inc., 33 Jurimetrics J. 147, 155 (1992)). For an extensive list of law review articles discussing the Sega and/or Atari cases, see Kreiss, at 61 n. 226. As discussed subsequently in this opinion, an important distinction to be emphasized is that reverse engineering, in addition to accessing nonprotectable ideas and functional elements, may also yield access to original expression. However, while the former may be unoriginal and not worthy of copyright protection, the latter, though original and hence copyrightable, may also be denied protection where its use is found to be “fair” under 17 U.S.C. § 107”.

<sup>331</sup> Sony Computer Entertainment v. Connectix Corp., 203 F.3d 596 (9th Cir. 2000).

Este caso presentaba una diferencia sustancial en relación a los dos anteriores. Tanto en los casos de Atari como de Sega, los demandados que habían realizado los procesos de ingeniería inversa sobre los programas eran competidores directos de los demandantes, fabricantes de videojuegos. En esta ocasión, la parte demandada no era un competidor directo de la demandante. Con la información adquirida consecuencia del proceso de ingeniería inversa, Connectix elaboró un programa independiente que permitía a usuarios de sistemas operativos diferentes al de Sony reproducir en sus plataformas videojuegos fabricados por Sony<sup>332</sup>. Para elaborar dicho programa, Connectix había extraído la BIOS<sup>333</sup> de una videoconsola Sony adquirido de forma legal<sup>334</sup>. El Noveno Circuito, siguiendo la doctrina establecida en el caso Sega, declaró que el comportamiento de Connectix tanto al copiar el programa de Sony como al usar su información, cumplía los requisitos del fair use porque la finalidad de sus actos era conseguir acceder a elementos no protegidos del programa, y tales actos eran necesarios por ser la única forma de acceder a la información contenida en ellos de la manera menos intrusiva y más eficiente<sup>335</sup>.

En relación al análisis de los cuatro requisitos del artículo 107, el Noveno Circuito parece centrarse más en el segundo (la naturaleza de la obra protegida por copyright), cuando afirma que la copias intermedias realizadas durante el proceso de descompilación del código de la BIOS de Sony eran “necesarias” para acceder a la información contenida en los elementos funcionales no protegidos por el derecho de exclusiva sobre el programa. Sin embargo, comparado con la jurisprudencia precedente, el examen de la necesidad llevado a cabo por el Noveno Circuito en este caso presenta una importante diferencia. En esta ocasión el Tribunal no se basa solo en si el acto de ingeniería inversa es el único medio técnico para acceder a los elementos funcionales

---

<sup>332</sup> Específicamente, los ingenieros de Connectix desarrollaron un programa que emulaba a la videoconsola de Sony permitiendo que los componentes de un ordenador Macintosh, como la tarjeta gráfica, imitasen los componentes de la videoconsola de Sony.

<sup>333</sup> BIOS: basic input-output system, es un software cuya función es iniciar y coordinar las funciones de la máquina.

<sup>334</sup> Sony v. Connectix, *op.cit.*, pág. 602.

<sup>335</sup> *Ibidem*, págs. 604 y 605. El Tribunal, al igual que en el caso Sega, realiza un análisis de cada uno de los requisitos del artículo 107 de la Copyright Act, rechazando las conclusiones del juez de instancia para quien los actos de ingeniería inversa sí habían constituido una infracción del copyright. Un comentario sobre la sentencia puede verse en P., HEINDL, *A Status Report from the Software Decompilation Battle: A Source of Sore for Software Copyright Owners in the United States and the European Union?*, TTLF Working Papers, n.1, [http://www.law.stanford.edu/program/centers/ttlf/papers/heindl\\_wpl.pdf](http://www.law.stanford.edu/program/centers/ttlf/papers/heindl_wpl.pdf), pág. 97-107, (última consulta 31 de enero 2013).



del programa, sino que tiene en cuenta si este medio (la ingeniería inversa) es también la solución técnica más eficiente y la menos intrusiva en los derechos de exclusiva del demandante<sup>336</sup>.

En cuanto a los elementos funcionales de un programa de ordenador, subraya que no pueden ser objeto de protección del copyright, y añade que el medio apropiado de proteger esta clase de elementos es el derecho de patente, cuyos requisitos de protección son más estrictos<sup>337</sup>.

Finalmente, al examinar la tercera condición (cantidad y sustancialidad de la parte de la obra empleada en relación a la obra en su totalidad) declara que, si bien Connectix había copiado el programa protegido en su totalidad y varias veces, al no copiar el nuevo programa el código del programa descompilado, la infracción del derecho de reproducción pasaba a tener “a very little weight”.

Expuestos los casos más relevantes que han considerado la descompilación a efectos de interoperabilidad como un uso leal, amparado por tanto por el artículo 107 de la Copyright Act, es posible ver algunos puntos en común en los análisis de los distintos tribunales.

1º Tanto en el caso de Sega como en el de Sony, los actos de descompilación del código amparados por la excepción no tenían como finalidad crear un nuevo programa que compitiese directamente con el descompilado. En el caso de Sega, el Tribunal declara que las copias realizadas en el curso de procesos de ingeniería inversa para propósitos de no explotación directa han de ser sustanciales para considerarlas como infracción<sup>338</sup>. En el caso de Sony, el nuevo programa tenía como finalidad un uso comercial indirecto como era la creación de un programa compatible<sup>339</sup>.

En cuanto a los efectos sobre el mercado potencial, quizás sea esta la cuestión más delicada y que muestra una estrecha relación con el requisito de la naturaleza de la obra. Si bien los tribunales han dejado claro que el hecho de permitir la realización de procesos de ingeniería inversa por razones de

---

<sup>336</sup> Sony v. Connectix, *op.cit.*, pág. 605.

<sup>337</sup> Ibidem, pág. 606.

<sup>338</sup> Sega v. Accolade, *op.cit.*, pág. 1522-1523.

<sup>339</sup> Sony v. Connectix, *op. cit.*, pág. 606-607.

interoperabilidad no es una invitación a la piratería<sup>340</sup>, por otro lado valorar el papel que juega la interoperabilidad en el mercado de la informática es tarea sensible. En los cuatro casos examinados los tribunales han permitido la realización de ingeniería inversa cuando su finalidad era la elaboración de un nuevo producto competitivo con el descompilado, porque como indica el Noveno Circuito en el caso Sony “*the copyright did not give a right for Sony to control the device markets for the copyrighted work*”<sup>341</sup>. Es decir, si la interoperabilidad puede actuar como factor de apertura a una situación de lock-in, permitiendo por tanto la entrada de competencia en el mercado, la ingeniería inversa puede considerarse legítima.

2º En los cuatro casos anteriores, la realización de copias temporales y de actos de adaptación y transformación del código de un programa sin autorización, en un proceso de descompilación, no constituyeron infracción de los derechos exclusivos del titular, porque se daban dos condiciones:

1. Constituían un medio indispensable.

Si bien podemos distinguir dos posiciones ligeramente diferentes en la jurisprudencia norteamericana.

1º Por un lado la que podría denominarse el criterio del caso Sega, que requiere que la ingeniería inversa constituya el único medio (*the only way*) de acceso a las ideas y a los elementos funcionales del programa.

2º Por otro lado, el criterio del caso Sony, según el cual un medio indispensable es aquel que sea el menos intrusivo para los derechos del titular del copyright y al mismo tiempo el más eficiente en términos técnicos.

2. Se fundamentaban en una razón legítima: la interoperabilidad.

Aunque los argumentos para justificar la interoperabilidad varían, todos los tribunales finalmente la aceptan. Así, para el Tribunal de Apelación del Circuito Federal en el caso Atari, la razón que legitima la

---

<sup>340</sup> Atari v. Nintendo, *op.cit.*, pág. 843.

<sup>341</sup> Sony v. Connectix, *op.cit.*, pág. 607.

excepción es la descompilación realizada para entender y copiar ideas que no forman parte del objeto de protección, y en igual sentido el Tribunal de Apelaciones del Noveno Circuito en el caso Sony. Podríamos decir que justifica la investigación a efectos de interoperabilidad.

El Tribunal de Apelaciones del Noveno Circuito en el caso Sega y en igual sentido el del Undécimo en el caso Bateman, restringen el significado de razón legítima únicamente a la consecución de compatibilidad con otro programa. Podría hablarse de la justificación de la descompilación a efectos de interoperabilidad que mejora la competitividad del mercado.

### **3. La Digital Millennium Copyright Act (DMCA) como garante de la interoperabilidad informática.**

#### *3.1. Introducción: La DMCA y su relación con la interoperabilidad informática.*

La Digital Millennium Copyright Act (DMCA) fue promulgada por el Presidente Clinton el 28 de octubre de 1998<sup>342</sup>, incorporando los dos Tratados de la OMPI de 1996 sobre derechos de autor al derecho norteamericano: El Tratado de la OMPI sobre Derecho de Autor (WCT) y el Tratado de la OMPI sobre Interpretación o Ejecución y Fonogramas (WPPT).

En relación a la interoperabilidad informática, el artículo 1201 contenido en el Título I de la norma, prohíbe por un lado el desarrollo, la distribución y el uso de tecnologías que inutilicen mecanismos tecnológicos de protección de una obra, y a continuación establece un catálogo cerrado de excepciones legales, entre las que se encuentra la ingeniería inversa. Como se analizará en los siguientes epígrafes, el vocabulario y el estilo de la letra (f) del artículo 1201 que establece la excepción de la retro ingeniería, guarda un parecido considerable con el texto del artículo 6 de la Directiva de la Unión Europea sobre protección de programas de ordenador de 1991. Sin embargo, como se verá también, desde su entrada en vigor los tribunales norteamericanos se han mostrado reacios a la aplicación de algunas de las normas establecidas por la DMCA en cualquier forma que pudiese impedir la interoperabilidad

---

<sup>342</sup> Digital Millennium Copyright Act of 1998, Pub.L. No. 105-304, 112 Stat. 2860 (Oct. 28, 1998).

informática. Para entender esta especie de animadversión de los tribunales a la aplicación del artículo 1201, es necesario comprender el complejo escenario en que se elaboró la DMCA, el espíritu de los legisladores y las cuestiones que planteó a la doctrina establecida por los tribunales garantista de la interoperabilidad, a través de la aplicación de la excepción del fair use.

### 3.2. *La Digital Millennium Copyright Act (DMCA).*

Desde una perspectiva cronológica el modo en que se llegó a la aprobación del a DMCA es ligeramente complicado. El punto de salida podría situarse en la esfera internacional, en 1996, con la aprobación de dos Tratados en la Conferencia Diplomática de la OMPI celebrada en Ginebra en diciembre de ese año: El Tratado de la OMPI sobre Derecho de Autor (WCT) y el Tratado de la OMPI sobre Interpretación o Ejecución y Fonogramas (WPPT).

Esta Conferencia Diplomática había sido la culminación de un proceso que formalmente había dado comienzo en 1991 cuando la OMPI convocó a un “Comité de Expertos” para discutir la conveniencia de un Protocolo para el Convenio de Berna<sup>343</sup>. La Oficina Internacional de la OMPI elaboró varios documentos preparatorios para ser analizados por ese Comité, cuya primera sesión se celebró del 4 al 8 de noviembre de 1991. La propuesta de la Oficina en cuanto a incorporar en el marco de un Protocolo al Convenio de Berna, un mecanismo que reconociera derechos a los productores de fonogramas, coincidentes unos y adicionales otros a los contenidos en la Convención de Roma para la protección de los intérpretes o ejecutantes, los productores de fonogramas y los organismos de radiodifusión, generó un ardiente debate durante la primera sesión del Comité en el cual, si bien hubo consenso entre todas las delegaciones sobre la necesidad de fortalecer los derechos de los productores de grabaciones sonoras, también se expresó una opinión mayoritaria acerca de que la modernización de esa protección debía tener lugar en el contexto de la Convención de Roma y “otros instrumentos” y no

---

<sup>343</sup> Vid. *Digital Millennium Copyright Act (DMCA) Section 104 Report, A Report of the Register of Copyrights to §104 of the Digital Millennium Copyright Act*, Library of Congress, Copyright Office, 2001, pág. 5.

bajo el marco del Convenio de Berna. Cinco años después, lo que iba a ser un Protocolo del Convenio de Berna acabó constituyendo dos nuevos Tratados Internacionales<sup>344</sup>.

Poco después de la primera sesión del Comité de la OMPI, en Estados Unidos se estaba produciendo importantes movimientos relacionados con la regulación de la información en la era digital. En 1993, por orden de la Casablanca se constituyó un comité denominado “Information Infrastructure Task Force” (IITF) cuya misión era implementar las políticas de la administración Clinton acerca de crear una “Infraestructura de Información Nacional”<sup>345</sup>. Dentro de ese comité se creó un grupo de trabajo específico sobre derechos de propiedad intelectual. El objetivo de tal grupo era investigar los efectos de la tecnología digital sobre los derechos de propiedad intelectual y efectuar recomendaciones acerca de la conveniencia de modificar la legislación existente. El resultado fue la publicación de un Libro Verde en 1994 y de un Libro Blanco en 1995.

El Libro Verde, publicado en julio de 1994, proponía el establecimiento de una prohibición para la producción y distribución de dispositivos que eludiesen tecnología anticopia<sup>346</sup>. Curiosamente, por petición del gobierno norteamericano el borrador presentado en octubre de 1994 para el Protocolo del Convenio de Berna que estaba siendo discutido en la OMPI, contenía normas de carácter extensivo en relación a la elusión de medidas anticopia<sup>347</sup>.

Un año más tarde, se publicó el Libro Blanco, el cual seguía la línea iniciada por el Libro Verde, contestando incluso alegaciones hechas por el American Committee for Interoperable Systems (ACIS)<sup>348</sup>, al señalar que la doctrina del fair use no requiere que

---

<sup>344</sup> Para una versión detallada de la elaboración de los Tratados de la OMPI de 1996 vid. *World Intellectual Property Organization: WIPO Copyright Treaty (WCT) (1996) with the agreed statements of the Diplomatic Conference that adopted the Treaty and the provisions of the Berne Convention (1971) referred to the Treaty*, publicación OMPI 226, Ginebra, OMPI, 1997.

<sup>345</sup> National Information Infrastructure (NII) es un término que acuñó la administración Clinton, definido como “a seamless web of communications networks, computers, databases, and consumer electronics that will put vast amounts of information at users fingertips” disponible en: <http://www.ibiblio.org/nii/> (última consulta 1 de junio, 2013).

<sup>346</sup> Green Paper, “Information Infrastructure Task Force, Working Group on Intellectual Property Rights, Intellectual Property and the National Information Infrastructure: A Preliminary Draft of the Report of the Working Group on Intellectual Property Rights” (July 1994).; disponible en: <http://cool.conservation-us.org/bytopic/intprop/ipwg/> (última consulta 1 de junio, 2013).

<sup>347</sup> Vid. Memorandum de la OMPI, Oficina Internacional, 98(a)(i), 5 de octubre de 1994.

<sup>348</sup> Esta asociación había mostrado su desacuerdo a las propuestas normativas del Libro Verde sobre medidas tecnológicas de protección, argumentando entre otras razones que las propuestas del gobierno iban en contra del artículo 117 de la Copyright Act o de las copias realizadas en el curso de un proceso de

el titular de un derecho de copyright permita o facilite el acceso no autorizado o el uso de la obra. Si así fuese (...) los museos no podría exigir el pago de una entrada o prohibir hacer fotografías (...)»<sup>349</sup>.

La postura que muestra el Libro Blanco es igual de tajante que su predecesor, recomendando la elaboración de una norma que prohibiese la producción y la distribución de dispositivos o servicios capaces de eludir mecanismos tecnológicos de protección sobre obras protegidas por copyright, lo cual se tradujo en el artículo 1201 de la DMCA, que junto a las demás propuestas normativas contenidas en el Libro fueron presentadas ante las cámaras legislativas a finales de 1995<sup>350</sup>.

En medio de este proceso legislativo interno y tras la aprobación de los Tratados de la OMPI en 1996, la Oficina de Patentes y Marcas estadounidense (PTO) presentó una propuesta a la Administración para incorporar los citados Tratados a la legislación nacional. Dicha propuesta fue modificada por la Oficina del Asesor General del Departamento de Comercio, que luego la envió al Congreso en Julio de 1997. Según indicaba, la protección contra medidas elusivas de mecanismos tecnológicos de protección existente en Estados Unidos no cumplía con las obligaciones dispuestas en los artículos 11 del WCT y del artículo 18 WPPT respectivamente<sup>351</sup>. Como recoge el informe de la Oficina de Copyright publicado en 2001, a pesar de que algunas medidas de carácter elusivo se encontraban reguladas y prohibidas por la ley federal antes de

---

ingeniería inversa y consideradas fair use por los tribunales. Vid. ACIS Comments on Intellectual Property Issues Involved in the National Information Infrastructure Initiative, presentadas el 10 de diciembre de 1993, pág. 5; disponible en: [http://w2.eff.org/IP/ipwg\\_nii\\_ip\\_report\\_acis.comments.txt](http://w2.eff.org/IP/ipwg_nii_ip_report_acis.comments.txt) (última consulta 1 de junio, 2013)

<sup>349</sup> Information Infrastructure Task Force, *Intellectual Property and the National Information Infrastructure: The Report of the Working Group on Intellectual Property Rights*, 1995, pág. 231; disponible en: <http://www.uspto.gov/web/offices/com/doc/ipnii/> (última consulta 1 de junio, 2013).

<sup>350</sup> Joint Hearing on H.R. 2441 and S. 1284 Before the House Sucomm. on Courts, Intellectual Property, and the Administration of Justice of the House Comm. on the Judiciary, and the State Judiciary Committee, 104th Cong., 1st. Sess. 24 (Nov. 15, 1995).

<sup>351</sup> Artículo 11 WCT: Las Partes Contratantes proporcionarán protección jurídica adecuada y recursos jurídicos efectivos contra la acción de eludir las medidas tecnológicas efectivas que sean utilizadas por los autores en relación con el ejercicio de sus derechos en virtud del presente Tratado o del Convenio de Berna y que, respecto de sus obras, restrinjan actos que no estén autorizados por los autores concernidos o permitidos por la Ley.

Artículo 18 WPPT: Las Partes Contratantes proporcionarán protección jurídica adecuada y recursos jurídicos efectivos contra la acción de eludir medidas tecnológicas efectivas que sean utilizadas por artistas intérpretes o ejecutantes o productores de fonogramas en relación con el ejercicio de sus derechos en virtud del presente Tratado y que, respecto de sus interpretaciones o ejecuciones o fonogramas, restrinjan actos que no estén autorizados por los artistas intérpretes o ejecutantes o los productores de fonogramas concernidos o permitidos por la Ley.

1998, su alcance no llegaba a los derechos de propiedad intelectual. Si para llevar a cabo la inutilización de la medida tecnológica de protección es necesaria la reproducción de la obra protegida por ella, un acto de esta clase constituye una infracción del copyright. A ello se añade que la puesta a disposición de dispositivos que permitan eludir tales medidas podría constituir una infracción indirecta del copyright, aunque únicamente en situaciones muy determinadas y concretas<sup>352</sup>.

Finalmente el 28 de octubre de 1998 el presidente Clinton promulgó la Digital Millennium Copyright Act (DMCA)<sup>353</sup>, con la cual la Copyright Act experimentó los cambios más importantes desde que en 1989 se había producido la adhesión de los Estados Unidos al Convenio de Berna. La DMCA consta de cinco títulos<sup>354</sup>, pero es el Título I, y en particular los apartados (a), (b), (c) y (f) del artículo 1201<sup>355</sup> el cuerpo normativo que interesa en este trabajo.

El líneas generales el Título I, para cumplir con las exigencias de los Tratados de la OMPI, añade un nuevo capítulo 12 a la Copyright Act, donde establece con bastante amplitud una prohibición contra dispositivos y servicios que permitan inutilizar mecanismos de protección tecnológica (TPM) sobre obras en formato digital o que permitan alterar los datos informativos necesarios para la administración del derecho de autor. Entre el catálogo de excepciones establecidas a la prohibición, se encuentra la ingeniería inversa para conseguir la interoperabilidad de un programa de ordenador en el artículo 1201(f).

---

<sup>352</sup> DMCA Section 104 Report: A Report of the Register of Copyrights Pursuant to §104 of the Digital Millennium Copyright Act.; U.S. Copyright Office, August 2001, pág. 9.

<sup>353</sup> Digital Millenium Copyright Act, Pub. L. No. 105-304, 112 Stat. 2860 (Oct. 28, 1998), *codified at* 17 U.S.C. 512, 1201-05, 1301-22; 28 U.S.C. 4001.

<sup>354</sup> De manera sucinta el contenido de los cinco títulos de la DMCA es el siguiente: el primero incorpora al derecho norteamericano los preceptos de los tratados de la OMPI sobre derecho de autor de diciembre de 1996; el segundo establece determinados mecanismos de salvaguardia para limitar la responsabilidad de los proveedores de servicios «en línea»; el tercero permite la ejecución de copias temporales durante la realización de trabajos de mantenimiento de un ordenador; el cuarto introduce diversas notificaciones en la Ley de derecho de autor norteamericana, incluyendo algunos que facilitan la transmisión mediante Internet, y finalmente el quinto introduce una protección especial para el diseño de cascos de buques, peculiaridad del Derecho norteamericano a consecuencia de la Sentencia del Tribunal Supremo en el caso «Bonito Boats».

<sup>355</sup> El texto completo del artículo 1201 puede consultarse en: <http://www.copyright.gov/title17/92chap12.html> (última consulta 31 de enero 2013). Debido a la extensión de la norma, en este trabajo sólo reproducimos aquellos apartados que afectan a la ingeniería inversa de programas de ordenador.

Antes de pasar al examen de la excepción en detalle, una última peculiaridad que habrá que tener en cuenta al analizar la DMCA es que la misma introdujo un sistema de revisión trienal de la Copyright Act en el artículo 1201(a)(1)(C)<sup>356</sup>, con la que se faculta al Bibliotecario del Congreso para iniciar, cumpliendo una serie de formalidades, un proceso de revisión con objeto de determinar si existe alguna clase de obra sobre la que la prohibición de inutilizar medidas tecnológicas de protección produzca efectos especialmente adversos para los usuarios y por tanto los actos descritos por la prohibición debieran quedar eximidos para esa clase de obra.

La última revisión del artículo 1201 se inició a finales de 2011<sup>357</sup>, y de entre los dieciocho informes presentados, destaca la petición coincidente en seis de ellos<sup>358</sup> solicitando la exención de determinadas clases de programas de ordenador, muchos de ellos contenidos en dispositivos móviles (smartphones, tablets), por razones de interoperabilidad, lo cual pone de manifiesto el cada vez mayor interés de los operadores del mercado en esta materia.

---

<sup>356</sup> Artículo 1201(a)(1)(C): *During the 2-year period described in subparagraph (A), and during each succeeding 3-year period, the Librarian of Congress, upon the recommendation of the Register of Copyrights, who shall consult with the Assistant Secretary for Communications and Information of the Department of Commerce and report and comment on his or her views in making such recommendation, shall make the determination in a rulemaking proceeding for purposes of subparagraph (B) of whether persons who are users of a copyrighted work are, or are likely to be in the succeeding 3-year period, adversely affected by the prohibition under subparagraph (A) in their ability to make noninfringing uses under this title of a particular class of copyrighted works. In conducting such rulemaking, the Librarian shall examine —*

*(i) the availability for use of copyrighted works;*

*(ii) the availability for use of works for nonprofit archival, preservation, and educational purposes;*

*(iii) the impact that the prohibition on the circumvention of technological measures applied to copyrighted works has on criticism, comment, news reporting, teaching, scholarship, or research;*

*(iv) the effect of circumvention of technological measures on the market for or value of copyrighted works; and*

*(v) such other factors as the Librarian considers appropriate.*

Artículo 1201(a)(1)(D): *The Librarian shall publish any class of copyrighted works for which the Librarian has determined, pursuant to the rulemaking conducted under subparagraph (C), that noninfringing uses by persons who are users of a copyrighted work are, or are likely to be, adversely affected, and the prohibition contained in subparagraph (A) shall not apply to such users with respect to such class of works for the ensuing 3-year period.*

<sup>357</sup> Federal Register, Vol. 76, No 189, Thursday September 29, 2011, pág. 60398.

<sup>358</sup> Los seis informes los firman: Electronic Frontier Education, Software Freedom Foundation, Consumers Union, Youghioghenny Communications LLC, MetroPCS Communications, Inc. y RCA – The Competitive Carriers ASS. Más información disponible en: <http://www.copyright.gov/1201/> (última consulta 1 de junio, 2013).



### 3.3. Las conductas prohibidas por el artículo 1201 DMCA.

Una de las formas de minimizar el riesgo de infracción en la era digital es que el ordenamiento proporcione herramientas a los titulares de los derechos de autor, por ejemplo legitimando el uso de mecanismos tecnológicos para proteger sus obras en formato digital. El artículo 1201 de la DMCA confirma esta premisa al introducir una prohibición legal contra quienes inutilicen dichas mecanismos. Pero con el establecimiento de esta prohibición, en opinión de algunos autores, no sólo se incorporan los Tratados OMPI de 1996 al derecho norteamericano<sup>359</sup>, sino que el texto de la norma va más allá de esta obligación, y otorga un derecho de acceso ex-novo al propietario de la medida tecnológica de protección, a través del cual se le permite controlar el acceso a la obra<sup>360</sup>. Sin entrar en un examen exhaustivo de la norma, cuyo lugar no corresponde aquí, sí es transcendental comprender cuáles son las medidas prohibidas por la norma antes de examinar el ámbito de la excepción de ingeniería inversa y la interpretación jurisprudencial de la misma.

El artículo 1201(a) distingue dos categorías de medidas tecnológicas. En primer lugar, las que evitan el acceso no autorizado a la obra protegida, y en segundo lugar, las medidas que impidan la copia no autorizada de la obra.

En relación con estas medidas, diferencia dos tipos de actividades que se califican como ilícitas<sup>361</sup>:

- 1º La fabricación, venta y tráfico de mecanismos o servicios que se utilicen para eludir cualquiera de ambas categorías de medidas tecnológicas (de acceso y anticopia).

---

<sup>359</sup> Vid. supra., artículos 11 WCT y 18 WPPT respectivamente.

<sup>360</sup> P., SAMUELSON, "Intellectual Property and the Digital Economy: Why the Anticircumvention Regulations Need to be Revised", BTLJ 14, 1999, pág. 519 (539); J.C., GINSBURG, "Copyright Use and Excuse on the Internet", Columbia VLA Journal of Law & the Arts, Vol. 24, 2000, pág. 1; disponible en SSRN: <http://ssrn.com/abstract=239747> (última consulta 1 de junio, 2013); B., FITZGERALD, C., CIFUENTES, M. LEHMANN, "Innovation, Software ...", *op.cit.*, pág. 140.

<sup>361</sup> Art. 1201(a) (1)(A): *No person shall circumvent a technological measure that effectively controls access to a work protected under this title.* Art. 1201(a)(3)(A): *To "circumvent a technological measure" means to descramble a scrambled work, to decrypt an encrypted work, or otherwise to avoid, bypass, remove, deactivate, or impair a technological measure, without the authority of the copyright owner; and (B) a technological measure "effectively controls access to a work" if the measure, in the ordinary course of its operation, requires the application of information, or a process or a treatment, with the authority of the copyright owner, to gain access to the work.*

2ºEl acto de elusión propiamente de tales medidas, el cual sólo es ilícito cuando se refiere a medidas tecnológicas de acceso; la elusión de medidas tecnológicas anticopia no es ilícita – o cuanto menos, podría decirse que la elusión de medidas anticopia por parte de los usuarios no queda prohibida por la DMCA, pero fabricar y vender mecanismos o prestar servicios para hacer posible la elusión de tales medidas anticopias, sí.

Dentro de las actividades descritas, la norma especifica qué clase de dispositivos están afectados por la prohibición, limitándolo a tres modalidades:

1ºDiseñados con el objetivo principal de inutilizar una medida tecnológica de protección.

2ºCon un escaso valor comercial fuera de la finalidad de inutilización, y,

3ºComercializados solo para usos elusivos de medidas tecnológicas.

A ello se añade una cláusula de salvaguarda contenida en el artículo 1201(c)<sup>362</sup>, a través de la cual se restringe más la consideración como conductas prohibidas de la elusión de controles de acceso para finalidades legítimas, así como la fabricación y distribución de tecnologías que permitan la inutilización de mecanismos tecnológicos para fines legítimos.

En cualquier caso, no debe perderse de vista que una excepción a la infracción del copyright no es una excepción a la específica prohibición contenida en el Capítulo 12 de la Copyright Act.

---

<sup>362</sup> Art. 1201(c) *Other Rights, Etc., Not Affected.* — (1) *Nothing in this section shall affect rights, remedies, limitations, or defenses to copyright infringement, including fair use, under this title.*

(2) *Nothing in this section shall enlarge or diminish vicarious or contributory liability for copyright infringement in connection with any technology, product, service, device, component, or part thereof.*

(3) *Nothing in this section shall require that the design of, or design and selection of parts and components for, a consumer electronics, telecommunications, or computing product provide for a response to any particular technological measure, so long as such part or component, or the product in which such part or component is integrated, does not otherwise fall within the prohibitions of subsection (a)(2) or (b)(1).*

(4) *Nothing in this section shall enlarge or diminish any rights of free speech or the press for activities using consumer electronics, telecommunications, or computing products.*

### 3.4. La excepción de la ingeniería inversa del artículo 1201(f).

Desde una perspectiva general, la DMCA tiene repercusión en tres clases de procesos de ingeniería inversa del software a efectos de copyright.

1º En primer lugar, durante el proceso de descompilación, en dos momentos: primero sobre el propio mecanismo tecnológico de protección (TPM), al analizar el funcionamiento del código informático y determinar los mecanismos necesarios para desbloquearlo; en segundo lugar, al analizar y determinar los requisitos de interoperabilidad con el programa protegido por el mecanismo tecnológico de protección.

2º La segunda ocasión en que la DMCA afecta la ingeniería inversa tiene lugar cuando el programador elabora su propia llave digital que le permite desbloquear el mecanismo tecnológico (que es un programa de ordenador).

3º En tercer lugar, cuando introduce su llave digital en la obra protegida, transformando la obra original.

El artículo 1201(f)<sup>363</sup> establece una excepción a la ingeniería inversa sobre los actos prohibidos por el artículo 1201 en aras de la interoperabilidad, tanto de las medidas de inutilización de control de acceso como de medidas anticopia, en cualquiera de los comportamientos antes descritos.

---

<sup>363</sup> “1) *Notwithstanding the provisions of subsection (a)(1)(A), a person who has lawfully obtained the right to use a copy of a computer program may circumvent a technological measure that effectively controls access to a particular portion of that program for the sole purpose of identifying and analyzing those elements of the program that are necessary to achieve interoperability of an independently created computer program with other programs, and that have not previously been readily available to the person engaging in the circumvention, to the extent any such acts of identification and analysis do not constitute infringement under this title.*

*(2) Notwithstanding the provisions of subsections (a)(2) and (b), a person may develop and employ technological means to circumvent a technological measure, or to circumvent protection afforded by a technological measure, in order to enable the identification and analysis under paragraph (1), or for the purpose of enabling interoperability of an independently created computer program with other programs, if such means are necessary to achieve such interoperability, to the extent that doing so does not constitute infringement under this title.*

*(3) The information acquired through the acts permitted under paragraph (1), and the means permitted under paragraph (2), may be made available to others if the person referred to in paragraph (1) or (2), as the case may be, provides such information or means solely for the purpose of enabling interoperability of an independently created computer program with other programs, and to the extent that doing so does not constitute infringement under this title or violate applicable law other than this section.*

*(4) For purposes of this subsection, the term “interoperability” means the ability of computer programs to exchange information, and of such programs mutually to use the information which has been exchanged.” (el subrayado es propio).*

Es decir, la excepción permite a los desarrolladores de software inutilizar los TPMs de un programa de ordenador adquirido de forma legítima con el objeto de identificar los elementos necesarios para alcanzar interoperabilidad con un nuevo programa creado de manera independiente. Sólo si los elementos necesarios para alcanzar interoperabilidad están disponibles no entra en juego la excepción. Además, la persona cuyo comportamiento está exento de infracción podrá elaborar las herramientas necesarias para llevar a cabo la inutilización de la medida tecnológica, tanto para uso personal como uso de terceros. Es más, podrá elaborar medidas que inutilicen mecanismos tecnológicos de protección sobre un programa y ponerlas a disposición de terceros, si la finalidad del comportamiento es alcanzar interoperabilidad.

#### *3.4.1. Examen de la excepción.*

##### *a). Requisitos subjetivos: La persona autorizada.*

Comparada con la Directiva europea sobre protección de programas de ordenador, es interesante subrayar que la expresión empleada en el texto de la excepción del artículo 1201(f)(1) se parece bastante a la del artículo 6 de la Directiva<sup>364</sup>. La Directiva habla de actos realizados “por el licenciataro o por cualquier otra persona facultada para utilizar una copia del programa”, mientras el artículo 1201(f)(1) de la persona que legítimamente haya obtenido derecho a usar una copia del programa<sup>365</sup>.

Si acudimos a la historia legislativa de la norma, el informe indica que este requisito subjetivo se verá satisfecho si el programa de ordenador se ha adquirido de una fuente legítima, junto con los códigos de serie necesarios, claves u otros medios que sean necesarios para posibilitar el uso del programa con otros programas<sup>366</sup>.

La primera conclusión que puede extraerse es que quienes tengan una copia pirata del programa o lo hayan pirateado no cumplirán con el requisito subjetivo de la excepción porque la obtención del derecho a usar no ha sido legítima.

---

<sup>364</sup> El artículo 6 de la Directiva Software habla de “a *person having a right to use a copy of a program*” o “persona facultada para utilizar la copia”; vid. infra. Cap. 3.2.4.D.2).

<sup>365</sup> “*has lawfully obtained the right to use a copy of a computer program*”

<sup>366</sup> S. Rep. No. 105-190, 1998, pág. 33; disponible en: [http://thomas.loc.gov/cgi-bin/cpquery/?&sid=cp105VABCI&r\\_n=sr190.105&dbname=cp105&&sel=TOC\\_70441&](http://thomas.loc.gov/cgi-bin/cpquery/?&sid=cp105VABCI&r_n=sr190.105&dbname=cp105&&sel=TOC_70441&) (última consulta 1 de junio, 2013).

La jurisprudencia ha mantenido esta concepción que podríamos denominar de “usuario legítimo” desde el caso *Atari v Nintendo*, en el cual el Tribunal de Apelación del Circuito Federal declaró que una persona que obtenga una copia de un programa de ordenador de forma de ilegal, no podrá alegar la excepción de fair use para defender un acto de ingeniería inversa de un programa<sup>367</sup>.

En igual sentido se ha pronunciado la jurisprudencia al examinar la procedencia de la excepción aquí comentada, cuando en 2005, en el caso *Davidson & Associates DBA Blizzard Entertainment, Inc.; Vivendi Universal Inc. v. Jung et al.*<sup>368</sup>, declaró que las acciones de los demandados no podían acogerse a la excepción del artículo 1201(f) porque no podían ser considerados personas con un derecho de uso legítimo sobre la copia del programa.

Para entender esta afirmación del tribunal ayudará una breve descripción de los hechos: Los demandados eran un grupo de usuarios de un videojuego propiedad de los demandantes, los cuales habían formado un grupo llamando “the bnetd project”. El videojuego objeto del litigio tenía la peculiaridad de ofrecer un servicio en línea, “Battle.net”, que permitía a los adquirentes de copias legítimas del programa jugar online. A causa de algunos problemas con la prestación del servicio sufridos por los usuarios de “Battle.net”, este grupo altruista desarrolló un programa de código abierto que permitía a los usuarios de varios videojuegos de la demandantes, jugar on line en servidores distintos de “Battle.net”. Para que su programa fuese funcionalmente operativo, tuvieron que descompilar los videojuegos propiedad de Blizzard para hacerlos interoperables con su servicio online. Sin embargo, todos los miembros del grupo habían aceptado las licencias de usuario final y las licencias de uso tanto de los videojuegos de Blizzard como del servicio de juego on line Battle.net. Estas licencias prohibían expresamente tanto la ingeniería inversa sobre las obras como la utilización de los videojuegos en otro servicio on line. El Tribunal de Apelaciones del Octavo Circuito entendió que las licencias de usuario tenían el valor de contratos válidos y por tanto, los programadores habían violado los mismos al realizar la descompilación de los

---

<sup>367</sup> *Atari v. Nintendo*, vid. supra, apartado 2.2.2.

<sup>368</sup> *Davidson & Associates DBA Blizzard Entertainment, Inc.; Vivendi Universal Inc. v. Jung et al.*, 422 F.3d 630 (8th Cir. 2005).

videojuegos, con lo cual, no podían ser considerados usuarios legítimos de la copia del programa<sup>369</sup>.

A pesar de que esta sentencia ha sido muy criticada por la doctrina<sup>370</sup>, la segunda conclusión que puede extraerse a efectos de considerar si una persona tiene derecho legítimo a usar una copia del programa de ordenador, es que habrá que tener en cuenta la existencia de licencias y de contratos.

*b). Requisitos objetivos de la conducta permitida.*

La norma permite la inutilización de la medida tecnológica que con carácter efectivo controle el acceso a una concreta parte de la obra con el único propósito de identificar y analizar aquellos elementos del programa necesarios para alcanzar la interoperabilidad de un programa creado independientemente con otros programas, y siempre que esos elementos no se encontrasen disponibles de otro modo.

Por tanto, en primer lugar, la norma es muy clara al determinar que el único fin con que puede realizarse la ingeniería inversa es para lograr la interoperabilidad. Además la clase de interoperabilidad permitida por la excepción es la de programa con programa y no otra (software-hardware por ejemplo). Qué haya de entenderse por interoperabilidad también queda palmariamente expuesto en el apartado 4 del artículo<sup>371</sup>.

---

<sup>369</sup> Aunque finalmente el Tribunal concluyó que no debía aplicar la excepción de la interoperabilidad porque la actuación de los apelantes había constituido una infracción sobre el copyright de la obra, a los efectos que aquí interesa, destaca la siguiente argumentación: “*Battle.net’s control measure was not freely available. Appellants could not have obtained a copy of Battle.net or made use of the literal elements of Battle.net mode without acts of reverse engineering, which allowed for a circumvention of Battle.net and Battle.net mode. Unlike in Lexmark case, Battle.net mode codes were not accessible by simply purchasing a Blizzard game or logging onto Battle.net, nor could data from the program be translated into readable source code after which copies were freely available without some type of circumvention*”. Davidson & Associates DBA Blizzard Entertainment, Inc., *op.cit.*, pág. 641.

<sup>370</sup> Vid. entre otros: E., WANNER, “Navigating the Nexus: DMCA Anti-Circumvention Protection of Computer Software”, *Ariz. St. L.J.* 43, 2011, pág. 1081 (1096-1097); A.K., PERZANOWSKI, “Rethinking Anticircumvention’s Interoperable Policy”, *U.C. Davis L. Rev.* 42, 2008-2009, pág. 1549 (1585-1589); A.H., RAJANI, “Davidson & Associates v. Jung: (Re)interpreting Access Controls”, *BTLJ* 21, 2006, pág. 365.

<sup>371</sup> Artículo 1201(f), ap. 4: “(...) *the ability of computer programs to exchange information, and of such programs mutually to use the information which has been exchanged*”. Esta definición es idéntica a la contenida en el Considerando Décimosegundo de la Directiva Software: “*la capacidad de los programas de ordenador para intercambiar información y utilizar mutuamente la información así intercambiada*”.

La interoperabilidad adquirida a través del acto de ingeniería inversa sólo puede tener lugar con un programa creado independientemente. Tal como indicaba el Informe del Senado sobre la DMCA, el programa resultante debe ser una obra nueva y original, en la medida que no viole el programa de ordenador descompilado<sup>372</sup>. Por lo tanto, el programa creado consecuencia del acto permitido por la excepción ha de cumplir con los requisitos generales de una obra de la Copyright Act<sup>373</sup>.

Sin embargo, hay otros conceptos que presentan mayores problemas de interpretación y que afectan al ámbito de la excepción:

*1. El control efectivo al acceso de la obra protegida.*

El apartado (1) indica que la medida tecnológica ha de controlar efectivamente el acceso a la obra protegida. Qué ha de entenderse por *control efectivo del acceso a una parte concreta del programa* es un aspecto que el artículo 1201(f) no entra a explicar. La referencia que contiene el apartado (2) del artículo a que una persona puede desarrollar y emplear (*a person may develop and employ*) dichas medidas tecnológicas, parece sugerir que el ámbito de la excepción está limitado a aquellos mecanismos desarrollados por la persona cuya finalidad es inutilizar la TPM, en oposición a medidas de inutilización de TPMs que ya estuvieran disponibles en el mercado. Sin embargo, la historia legislativa de la norma indica al respecto del 1201(f)(2) que los derechos contemplados en este apartado deben ejercerse tanto a través de herramientas ya existentes como de herramientas desarrolladas para ese específico fin<sup>374</sup>, con lo cual

---

<sup>372</sup> S. Rep. No. 105-190, 1998, pág. 32 (“the resulting program must be a new and original work, in that it may not infringe the original computer program”) disponible en: [http://thomas.loc.gov/cgi-bin/cpquery/?&sid=cp105VABCI&r\\_n=sr190.105&dbname=cp105&&sel=TOC\\_70441&](http://thomas.loc.gov/cgi-bin/cpquery/?&sid=cp105VABCI&r_n=sr190.105&dbname=cp105&&sel=TOC_70441&) (última consulta 1 de junio, 2013). Por otra parte, el término “independently created” es un concepto legal del copyright law, cuyo significado es que la expresión de la obra posea un nivel de originalidad suficiente para obtener protección bajo la Copyright Act. Vid., entre otros, *Strombarck v. New Line Cinema*, 384 F.3d (6th Cir. 2004) pág. 283, (294) donde se debatían los criterios que el tribunal debía usar para diferenciar los elementos no originales y no protegidos por el derecho de exclusiva, que define como “elements that were not independently created by the inventor”, antes de comparar la obra con la presuntamente obra infractora.

<sup>373</sup> Al establecer que el programa interoperable derivado del acto de ingeniería inversa debe ser un programa creado de forma independiente, se planteó la duda de si el tribunal debe examinar primero si se dan los requisitos de la excepción, o si el programa derivado es una obra original independiente. Esta cuestión no se producía al aplicar la doctrina del fair use a la ingeniería inversa en programas de ordenador, y se verá con más detalle en el epígrafe siguiente.

<sup>374</sup> S.Rep. No. 105-190, 1998, pág. 33: “Section 1201(f)(2) recognizes that to accomplish the acts permitted under Section 1201(f)(1) a person may, in some instances, have to make and use certain tools. In most instances these will be generally available tools that programmers use in developing computer programs, such as compilers, trace analyzers and disassemblers, which are not prohibited by this section.

parece que el legislador al decir “may develop and employ”, quería en realidad decir “may develop *and/or* employ”.

Si aplicamos una interpretación analógica de esta expresión a la contenida en el artículo 1201(a)(1)(A), donde se contiene la prohibición que exceptiona el artículo 1201(f), es posible aclarar un poco más el significado de esta expresión:

En primer lugar, el sub-apartado (3) define qué ha de entenderse por control efectivo de acceso a la obra como un mecanismo que para operar con carácter ordinario solicite la introducción de información, o bien requiera de un proceso o un sistema, permitido por el titular del copyright, para acceder a la obra<sup>375</sup>.

A ello ha de añadirse las matizaciones realizadas por la jurisprudencia:

1ºLa efectividad del control de acceso a la obra ha de valorarse en el caso concreto y,

2ºDebe existir algún tipo de mecanismo de control ya que la ausencia de permiso no puede considerarse inutilización en el sentido de la prohibición<sup>376</sup>.

Aclarado el significado de control efectivo, el siguiente concepto a examen es el de acceso a la obra protegida. Los tribunales se han pronunciado en varias ocasiones sobre esta cuestión, aunque los argumentos varían:

Para el Tribunal de Apelaciones del Sexto Circuito, es necesario distinguir entre acceso a la funcionalidad de la obra del acceso a la obra en sí, ya que el artículo 1201(a), donde se recoge la prohibición, no incluye entre los mecanismos de inutilización prohibidos los que simplemente restrinjan el acceso a funcionalidades de la

---

In certain instances, it is possible that a person may have to develop special tools to achieve the permitted purpose of interoperability. Thus this provision creates an exception to the prohibition on making circumvention tools contained in subsections 1201(a)(2) and (b). These tools can be either software or hardware”.

<sup>375</sup> El texto original: “if the measure, in the ordinary course of its operation, requires the application of information, or a process or a treatment, with the authority of the copyright owner, to gain access to the work.”

<sup>376</sup> “(...) *lack of permission is not circumvention under the DMCA*” en *Healthcare Advocates, Inc. v. Harding, Earley, Follmer & Frailey*, 2007 U.S. Dist. (E.D. Pa. July 20, 2007) Lexis 52544, pág. 51. En los casos *Auto Inspection Services v Flint Auto Auction*, 2006 U.S. Dist. (E.D. Mich. Dec. 4, 2006) Lexis 87366, pág. 22 (*the defendant had violated the anti-circumvention provisions of the DMCA by circumventing the quality control feature to gain access to the plaintiff's source code to copy it*), y *Apple, Inc. v. Psystar Corp.*, 673 F.Supp.2d 931 (N.D. Cal. 2009), pág. 942 (*The fact that circumvention devices may be widely available does not mean that a technological measure is not, as the DMCA provides, effectively protecting the rights of copyright owners in the ordinary course of its operations*)



obra<sup>377</sup>. Es importante destacar que siguiendo esta interpretación, la mayoría de los programas de inutilización de TPM quedarían fuera del ámbito de aplicación de la DMCA, porque la mayoría de programas de este tipo protegen el acceso a aspectos funcionales de la obra<sup>378</sup>.

Para el Tribunal de Apelación del Circuito Federal, el acceso existe si se demuestra que hay un nexo entre acceso y protección de un derecho de una obra protegida por copyright<sup>379</sup>. Con esta doctrina se previno que la DMCA pudiera ser usada como una herramienta limitativa de la competencia en los mercados de postventa y repuestos<sup>380</sup>.

El Tribunal de Apelaciones del Noveno Circuito, por el contrario, ha rechazado la interpretación del Tribunal de Apelación del Circuito Federal, afirmando que la DMCA establece un nuevo derecho, distinto a los derechos conferidos por el copyright sobre la obra, que es el derecho de acceso a la obra<sup>381</sup>.

## *2. Los elementos necesarios para alcanzar la interoperabilidad.*

La norma dispone que a través del acto de ingeniería inversa para inutilizar el mecanismo tecnológico de protección sobre la obra, solo se pueden identificar y analizar los elementos del programa que sean necesarios para alcanzar la interoperabilidad. Es bastante improbable que un ingeniero informático sepa, antes de llevar a cabo el acto de descompilación, a qué parte concreta del código objeto ha de acceder para obtener la información de las interfaces. Incluso tras realizar la descompilación del código de un programa completo, no resultará sencillo seleccionar cuáles son los concretos elementos del programa necesarios a efectos de

---

<sup>377</sup> Lexmark International, Inc. v. Static Control Components, Inc, 253 F. Supp. 2d 943, 948-949 (E.D. Ky. 2003), *rev'd*, 387 F.3d (6th Cir. 2004), *reh'g denied*, 2004 U.S. App. (Dec. 29, 2004), *reh'g denied*, 2005 U.S. App. 522 (6th Cir. Feb. 15, 2005), pág. 548: "Because Lexmark's authentication sequence does not restrict access to this literal code, the DMCA does not apply". En sintonía con esta interpretación, en un caso más reciente el Tribunal del Quinto Circuito ha mantenido que la inutilización de la MTP sobre un adaptador (dongle) no viola la prohibición contenida en el artículo 1201(a)(1)(A), porque el adaptador no controla el acceso a la obra protegida sino a las funcionalidades del programa: MGE UPS Systems, Inc. v GE Consumer & Industrial, Inc., 612 F.3d 760 (5<sup>th</sup> Cir. 2010), *withdrawn and superseded on reh'g*, 622 F. 3d 361 (5<sup>th</sup> Cir. 2010) pág. 365.

<sup>378</sup> Un comentario sobre esta cuestión puede verse en E., WANNER, "Navigating the Nexus: ..., *op.cit.*, pág. 1086.

<sup>379</sup> Chamberlain Group, Inc. v. Skyling Technologies, 381 F.3d 1178 (Fed. Cir. 2004), pág. 1204.

<sup>380</sup> Vid. J., BAND, M., KATOH, *Interfaces on Trial 2.0...*, *op.cit.*, pág. 119.

<sup>381</sup> MDY Indus., LLC v. Blizzard Entm't, Inc., 629 F.3d 928 (9th Cir. 2011), pág. 950: "While we appreciate the policy considerations expressed by the Federal Circuit in Chamberlain, we are unable to follow its approach because it is contrary to the plain language of the statute".

interoperabilidad. Incluso cuando las especificaciones de las interfaces son publicadas, y por lo tanto, se ponen a disposición del público, el programador puede considerar necesaria la descompilación del código<sup>382</sup>.

Por lo tanto, solo podrá determinarse si se ha cumplido este requisito a posteriori ante un tribunal, si el titular del programa considera que la persona que ha llevado a cabo la inutilización del mecanismo tecnológico de protección para analizar y estudiar los elementos necesarios del programa para alcanzar la interoperabilidad, ha utilizado la información obtenida de un modo ilícito.

### *3. Los mecanismos tecnológicos de inutilización.*

¿Cualquier mecanismo tecnológico de inutilización puede emplearse al amparo de la excepción? Aunque en el apartado (1) el legislador declara que la clase de ingeniería inversa amparada por la excepción es la de software con software, sin embargo, no indica si los mecanismos de inutilización han de ser de este tipo u otro. En la historia legislativa de la norma, el legislador indica en el Informe que si la finalidad es conseguir interoperabilidad, las herramientas amparadas por la excepción pueden consistir en software o en hardware<sup>383</sup>, pero no si las medidas tecnológicas a que se refiere han de tener por fin exclusivo la inutilización de la MTP o no.

En la jurisprudencia interpretativa de esta norma como el caso *Lexmark International, Inc. v. Static Control Components, Inc.*<sup>384</sup>, el Tribunal de Distrito decidió sobre una acción de cesación en la que se alegaba que Static Control (demandados) había violado la prohibición del artículo 1201(a)(2) por distribuir microchips que eran empleados para reemplazar el microchip de los cartuchos de tóner del demandante (Lexmark). El microchip de Static Control contenía un programa de ordenador que inutilizaba la secuencia de autenticación de Lexmark, cuya función era que el software de sus impresoras no permitiese a la máquina imprimir con un cartucho de tóner que no fuera nuevo y original de Lexmark. El Tribunal de Distrito entendió que en ese caso no cabía aplicar la excepción del artículo 1201(f) porque el microchip de Static Control no

---

<sup>382</sup> D.L., LEE, “Reverse Engineering of Computer Programs under the DMCA: Recognizing a “Fair Access” Defense”, *Marq. Intell. Prop. L. Rev.*, vol. 10, 2006, pág. 537 (555).

<sup>383</sup> S.Rep. No. 105-190, 1998, *op.cit.*, pág. 33.

<sup>384</sup> *Lexmark International, Inc. v. Static Control Components, Inc.*, 253 F. Supp. 2d 943, 948-949 (E.D. Ky. 2003), rev'd, 387 F.3d (6th Cir. 2004), reh'g denied, 2004 U.S. App. (Dec. 29, 2004), reh'g denied, 2005 U.S. App. 522 (6th Cir. Feb. 15, 2005).

podía ser considerado un programa de ordenador de creación independiente, ya que el programa embebido en el microchip era una copia del programa embebido en el microchip de Lexmark. En apelación, el Tribunal de Apelaciones del Sexto Circuito sin embargo, rechazó esta interpretación poniendo en duda si el programa embebido en el microchip de Lexmark estaba protegido como obra por copyright<sup>385</sup>. Al respecto de la creación independiente, declaró el Sexto Circuito que, aún el caso de que el programa de Static Control no fuese creado de forma independiente, el microchip en cuestión contenía otros programas de ordenador embebidos que sí cumplían ese requisito y esos programas interoperaban con las impresoras de Lexmark<sup>386</sup>.

En conclusión, la cuestión de si los mecanismos tecnológicos a que se refiere el apartado (2) del artículo 1201(f) se refieren únicamente a herramientas específicas de ingeniería inversa empleadas para desarrollar un programa de ordenador independiente o pueden incluir un programa de ordenador cuya finalidad principal no sea la de inutilizar un MTP, sigue sin estar clara tras el examen de la jurisprudencia.

El último elemento a tener en cuenta es la postura de la Copyright Office, en cuyo informe de 2003 al hilo del proceso de revisión normativa establecido en el artículo 1201(a)(1)(C)<sup>387</sup>. En tal proceso, Static Control había solicitado que los programas de ordenador embebidos en una máquina o un producto que además de controlar el funcionamiento del aparato, pudieran de algún modo controlar la ejecución o reproducción de obras protegidas por copyright y que por tanto tengan una significación económica independiente, fuesen excluidos de la prohibición del artículo 1201(a)(1). En su informe final la Copyright Office declaraba que el Congreso es muy consciente de la importancia de la interoperabilidad para la competencia y funcionalidad

---

<sup>385</sup> Lexmark v Static Control, *op.cit.*, pág. 536-541.

<sup>386</sup> *Ibidem*, pág. 550.

<sup>387</sup> A través del proceso de revisión normativa introducido por el artículo 1201, en 2002, Static Control solicitó la excepción de tres clases de obras de la prohibición del artículo 1201(a)(1): “Computer programs embedded in computer printer and toner cartridges and that control the interoperation and functions of the printer and toner cartridge; computer programs embedded in a machine or product and which cannot be copied during the ordinary operation or use of the machine or product; computer programs embedded in a machine or product and that control the operation of a machine or product connected thereto, but that do not otherwise control the performance, display or reproduction of copyrighted works that have an independent economic significance”, en *Memorandum from Marybeth Peters, Register of Copyrights, to James H. Billington, Librarian of Congress “Recommendation of the Register of Copyrights in RM 2002-4; Rulemaking on Exemption from Prohibition on Circumvention of Copyright Protection Systems for Access Control Technologies”*, 27 de Octubre de 2003, págs. 172 a 183; disponible en: <http://www.copyright.gov/1201/docs/registers-recommendation.pdf> (última consulta 1 de junio, 2013).

dentro de la propia excepción legal del artículo 1201(f) y por tanto, no era necesario introducir nuevas clases de obras como la señalada.

Por lo tanto, parece que los mecanismos tecnológicos de inutilización que pueden emplearse al amparo de la excepción abarcan todos los que tengan como finalidad alcanzar la interoperabilidad.

*4. El uso de la información adquirida: la posibilidad de proporcionar información o medidas para alcanzar la interoperabilidad a terceros.*

El artículo 1201(f)(3) regula el destino de la información adquirida a través del proceso de ingeniería inversa. Según la redacción del mismo, si se cumplen las condiciones establecidas en los apartados 1 y 2, dicha información puede ponerse a disposición de terceras personas, pero solo si se cumplen las dos condiciones que establece a continuación.

Estas dos condiciones consisten en que la información solo podrá usarse para adquirir la interoperabilidad de un programa de creación independiente con otros programas, y hasta el punto de que la puesta a disposición de tal información a terceros no viole la prohibición contenida en el artículo 1201 o constituya una violación del derecho aplicable al caso concreto.

Partiendo de la historia legislativa de la norma, el informe del Senado declaraba al respecto que este apartado reconoce que el desarrollo de un programa de ordenador es una tarea compleja, cuyo fruto es normalmente el esfuerzo combinado de varias personas. Por tanto, con esta norma se permite a los desarrolladores de un programa de ordenador creado independientemente, confiar en terceras personas tanto para desarrollar las herramientas de inutilización necesarias como para identificar cual es la información necesaria para adquirir interoperabilidad. La posibilidad de contar con terceros es especialmente importante para pequeñas empresas de ingeniería informática que no tienen capacidad para llevar a cabo las tareas descritas por sí solas. La finalidad de esta norma es por tanto permitir el intercambio de información y de herramientas<sup>388</sup>.

---

<sup>388</sup> S. Rep. No. 105-190, 1998, *op.cit.*, pág. 33.

Parece que el espíritu de la norma aboga por una interpretación restrictiva de quiénes son los terceros que pueden tener acceso a la información obtenida (programadores que formen parte del proceso de desarrollo del programa de ordenador de creación independiente) y de cuál es el uso que puede darse a esa información (alcanzar la interoperabilidad con otros programas).

La Copyright Office, sin embargo parece entender que el uso de la información que permite el apartado 3 del artículo 1201(f) es más amplio, cuando en su informe de 2003 “Rulemaking on Exemptions from Prohibition on Circumvention of Copyright Protection Systems for Access Control Technologies”, declaró que su función en este proceso de revisión normativa estaba limitada a decidir sobre los actos individuales exentos de la prohibición del artículo 1201(a)(1), mientras que la excepción legal del artículo 1201(f) incluía actos de tráfico comercial de actos prohibidos por los artículos 1201(a)(2) y 1201(b)<sup>389</sup>.

Por el momento la jurisprudencia no se ha visto obligada a interpretar esta cuestión, pero sí ha aclarado alguna cuestión como la quién puede hacer uso de esta información. En el caso de *Universal v Reimerdes*<sup>390</sup>, el Tribunal utilizó entre los argumentos para negar la procedencia de la excepción el hecho de que los demandados habían puesto a disposición de terceros la información obtenida. En su razonamiento indicaba la Sala que el artículo 1201(f)(3) solo autoriza a poner a disposición de terceros la información adquirida a través del proceso de ingeniería inversa a la persona que haya adquirido directamente esa información<sup>391</sup>.

Por tanto, sólo la persona legitimada por la excepción puede hacer uso de esta facultad, y tal como indica la Copyright Office, el uso que puede hacer de la

---

<sup>389</sup> “The statutory exemption found in 1201(f) not only permits circumvention of technological measures to analyze and identify interoperable elements of a protected computer program, but also provides **exemptions to the trafficking provisions in 1201(a)(2) and 1201(b)**” (la negrita es nuestra) en *Memorandum from Marybeth Peters, Register of Copyrights, to James H. Billington, Librarian of Congress “Recommendation of the Register of Copyrights in RM 2002-4; Rulemaking on Exemption from Prohibition on Circumvention of Copyright Protection Systems for Access Control Technologies”*, 27 de Octubre de 2003, pág. 180; disponible en: <http://www.copyright.gov/1201/docs/register-recommendation.pdf> (última consulta 1 de junio, 2013).

<sup>390</sup> *Universal City Studios, Inc. v. Reimerdes*, 111 F. Supp. 2d 294, 55 U.S.P.Q.2d (BNA) 1873 (S.D.N.Y. 2000) judgment entered, 111 F.Supp. 2d 346 (S.D.N.Y. 2000), aff’d on other grounds, 273 F.3d 429, 60 U.S.P.Q.2d (BNA) 1953 (2d Cir. 2001).

<sup>391</sup> *Ibidem*, pág. 320.

información abarca todos los actos de tráfico comercial indicados en los artículos 1201 (a)(2) y 1201 (b)<sup>392</sup>, siempre que la finalidad de tal uso sea lograr la interoperabilidad.

### 3.4.2. Relación entre el artículo 1201(f) y la doctrina del fair use.

Cuando la administración Clinton inició el proceso legislativo que culminó en la DMCA en 1998, los legisladores norteamericanos tenían en mente el reciente caso *Sega*, tal como indican las afirmaciones recogidas en la historia legislativa de la norma como que “su objetivo es asegurar que los efectos de la jurisprudencia interpretativa de la Copyright Act vigentes no sufran cambios derivados de la aprobación de esta ley”<sup>393</sup>. Esta afirmación tuvo su reflejo en la letra (c) del artículo 1201, rubricado “other rights, etc., not affected”. En su apartado (1) declara que nada de lo dispuesto en el artículo 1201 debe afectar a derechos, recursos, limitaciones o excepciones a infracciones de copyright, incluido *el fair use*<sup>394</sup>.

Sin embargo, donde el Noveno Circuito había declarado que la ingeniería inversa de un programa realizada por una razón legítima y cuya finalidad no se limitaba exclusivamente a alcanzar la interoperabilidad, seis años más tarde, con la aplicación de la DMCA se restringió esta doctrina de fair use al único propósito de alcanzar interoperabilidad. Es decir, la jurisprudencia interpretativa del artículo 1201(f) ha venido negando la posibilidad de emplear la defensa del fair use en situaciones donde el demandando había inutilizado un mecanismo tecnológico de protección para obtener acceso a una obra protegida por copyright con el objeto de llevar a cabo algún uso leal de la obra (como sería los actos de ingeniería inversa en el sentido definido por el caso *Sega*).

El primer caso donde se rechazó la aplicación del artículo 107 de la Copyright Act por ser incompatible con la prohibición de inutilización de mecanismos tecnológicos fue *RealNetworks v. Streambox*<sup>395</sup>. En esta ocasión, *RealNetworks*,

---

<sup>392</sup> “(...) manufacture, import, offer to the public, provide or otherwise traffic (...)” en ambos artículos.

<sup>393</sup> *Vid.* S. REP. No. 105-190, 1998, *op.cit.*, págs. 13 y 32; H.R. REP. No. 105-796, (1998) (Conf. Rep), *op.cit.*, pág. 42.

<sup>394</sup> Artículo 1201(c) (1): “Nothing in this section shall affect rights, remedies, limitations, or defenses to copyright infringement, including fair use, under this title”.

<sup>395</sup> *RealNetworks, Inc. v. Streambox, Inc.*, Case C99-2070Z (W.D. Wa. Dec. 20, 1999). El litigio no llegó a los Tribunales del Circuito, porque tras la sentencia del Tribunal de Distrito, las partes alcanzaron un

elaboraba y distribuía distintas versiones de un programa de ordenador que permitían reproducir vídeo, audio y otros contenidos multimedia, denominado “RealPlayer”. El RealPlayer estaba protegido con una MTP, denominada “RealProducer”. Esta MTP consistía en un programa de ordenador que convertía archivos de audio y video al formato especial del RealPlayer (RealMedia), de forma que sólo podían reproducirse en su programa. Además, esta empresa ofrecía otro programa, “RealServer”, que sirve para crear servidores multimedia, dándole a un sitio web la capacidad de ofrecer streaming de audio y de video. Ahora bien, los archivos multimedia en formato RealMedia podían compartirse a través de Internet utilizando otros servidores web, pero sólo podían reproducirse en el programa RealPlayer.

Estos productos estaban protegidos por dos medidas tecnológicas:

La primera, un programa llamado “Secret Handshake”, aseguraba que los archivos alojados en RealServer solo fuesen enviados a un RealPlayer.

La segunda, “Copy Switch”, otro programa anticopia incrustado en los archivos RealMedia, donde se contenía la preferencia del titular de la obra permitiendo o no su copia. De este modo, el RealPlayer estaba diseñado para leer el Copy Switch y comportarse según los deseos del titular de los derechos, permitiendo al usuario final copiar o no el archivo multimedia cuando era reproducido mediante el servicio de streaming.

La demandada era Streambox, una empresa que también ofrecía una serie de productos basados en programas de ordenador, denominados “Streambox VCR”, “the Ripper” y “the Ferret”. El primero de ellos permitía a sus usuarios acceder y descargar copias de archivos en formato RealMedia alojados en RealServer. Para acceder al servidor, el Streambox VCR imitaba el RealPlayer e inutilizaba el proceso de autenticación que el RealServer requería antes de acceder al contenido alojado.

Los otros dos programas servían, el primero, “the Ripper”, para convertir archivos en formato RealMedia a otros formatos, haciéndolos susceptibles de reproducción en cualquier programa distinto al RealPlayer y por extensión, permitiendo

---

acuerdo privado. Sin embargo, su doctrina fue respaldada por el Tribunal del Séptimo Circuito en el caso Chamberlain v. Skylink, aunque posteriormente rechazada por el Tribunal de Apelaciones del Circuito Federal; vid. Chamberlain Group, Inc. v. Skylink Technologies, 381 F.3d 1178 (Fed. Cir. 2004), pág. 1225.

su copia y reproducción en dispositivos que no eran ordenadores personales. El segundo, “the Ferret”, era un programa de ordenador cuya finalidad era ser instalado como “plug-in” en el RealPlayer, alterando la interfaz gráfica de usuario de éste.

La demanda se fundamentó en la violación de la prohibición del artículo 1201(a) de fabricar, distribuir y comercializar mecanismos o servicios que se utilicen para eludir mecanismos tecnológicos de protección de acceso y anticopia.

Streambox alegó en su defensa que su programa Streambox VCR permitía a los consumidores hacer “fair use” de las copias de archivos en formato RealMedia, siguiendo la doctrina establecida por el Tribunal Supremo en un caso de 1984<sup>396</sup>. El Tribunal rechazó esta argumentación justificando su decisión en que en el caso de 1984, el Tribunal Supremo se había basado en que los reproductores de video cassettes era utilizados mayoritariamente por los consumidores para ver los programas en diferido, no para redistribuir copias digitales de las obras. A ello añadieron que en el caso de Sony la práctica totalidad de los titulares de copyright habían autorizado o no se habían mostrado en contra de que sus obras fuesen grabadas para ser visionadas en diferido. Sin embargo, en el caso de Streambox, los titulares del copyright habían elegido impedir la copia de sus obras al alojarlas en un servidor como RealServer<sup>397</sup>.

El segundo caso negando la aplicación de la doctrina del fair use a un acto de ingeniería inversa sobre un programa de ordenador que inutilizaba un mecanismo tecnológico de protección, fue el caso Universal v. Reimerdes<sup>398</sup>. En esta ocasión, los demandantes eran ocho estudios de cine, que distribuían sus obras cinematográficas en DVDs protegidos por una MTP, un programa de ordenador denominado CSS que evitaba la copia. Los demandados habían elaborado un programa, DeCSS, que inutilizaba la MTP de los DVDs.

En su defensa, los demandados alegaron la excepción del artículo 1201(f), argumentando que su programa era necesario para lograr la interoperabilidad entre ordenadores con sistemas operativos Linux y los DVDs, y la doctrina del fair use, ya

---

<sup>396</sup> Sony Corp. v. Universal City Studios, Inc., *op.cit.*, pág. 417.

<sup>397</sup> RealNetworks, Inc. v. Streambox, Inc., *op.cit.*, pág. 13, aps. 14 y 15. Además añadió, citando a NIMMER, que la finalidad de la DMCA era precisamente contrarrestar el efecto causado por la doctrina del caso Sony. (ap. 16).

<sup>398</sup> Universal v. Reimerdes, *op.cit.*, pág. 273. Un comentario sobre la sentencia completa puede encontrarse en B., FITZGERALD, C. CIFUENTES, M. LEHMANN, “Innovation, Software and ...”, *op.cit.*, pág. 121.



que su programa permitía a los legítimos usuarios de un DVD protegido, inutilizar el mecanismo anticopia no para piratear la obra que contenía, sino para usos leales, como realizar una copia de seguridad.

Sin embargo, el Tribunal de Apelaciones del Segundo Circuito desestimó ambas.

En relación a la excepción del artículo 1201(f), los demandados no habían probado el requisito de la interoperabilidad, y aún asumiendo que el programa en lid hacía posible la reproducción de los DVDs en equipos con sistema operativo Linux, no podía aceptar la afirmación de que el programa se había creado con el único propósito de adquirir interoperabilidad entre un programa de ordenador y los DVDs porque el ámbito de la excepción solo amparaba la interoperabilidad entre programas de ordenador, no entre un programa de ordenador y otra clase de obra, protegida por una MTP consistente en un programa de ordenador<sup>399</sup>.

En cuanto a la doctrina del fair use, el Segundo Circuito recordó que la finalidad del artículo 107 de la Copyright Act es eximir de infracción del copyright determinados usos de obras protegidas que, de otro modo la constituirían. Sin embargo, la demanda encabezada por Universal se sustentaba en la violación de la prohibición del artículo 1201(a)(2) de la DMCA, al ofrecer y proporcionar al público una medida para inutilizar un mecanismo tecnológico de protección de acceso a la obra, lo cual no constituye una infracción del copyright. Si el Congreso hubiese querido que el fair use pudiera emplearse como excepción en estos casos, añadir, lo habría indicado expresamente<sup>400</sup>.

De este modo, el Tribunal de Apelaciones del Segundo Circuito, a través de una motivación que ha sido bastante criticada por la doctrina<sup>401</sup>, declara que la doctrina del fair use no es compatible con las normas que regulan la prohibición de inutilización de mecanismos tecnológicos de protección, entre las que se encuentra la excepción

---

<sup>399</sup> Vid. *Universal v. Reimerdes*, *op.cit.*, pág. 320.

<sup>400</sup> *Ibidem*, pág. 322.

<sup>401</sup> E.W., YOUNG, “Universal City Studios, Inc. v. Reimerdes: Promoting the Progress of Science and the Useful Arts by Demoting the Progress of Science and the Useful Arts [notes]”, *Northern Kentucky Law Review*, Vol. 28, n° 4, 2001, pág. 847; D.A., PETTEYS, “The Freedom to Link: The Digital Millennium Copyright Act Implicates the First Amendment in Universal City Studios, Inc. v. Reimerdes”, *Seattle University Law Review*, Vol. 25, n° 1, 2001, pág. 287; T.J., SADD, “Fair Use as a Defense under the Digital Millennium Copyright Act’s Anti-Circumvention Provisions”, *George Mason Law Review*, Vol. 10, n° 2, 2002, pág. 321; P., SAMUELSON, “Generativity of Sony v. Universal: The Intellectual Property Legacy of Justice Stevens”, *Fordham L. Rev.*, vol. 74, 2005, pág. 1831 (1860).

establecida en el artículo 1201(f) de la DMCA, porque en el caso concreto el fondo del asunto no giraba en torno la posible infracción del copyright sino sobre lo dispuesto en el artículo 1201(a)(2).

En los últimos años, los Tribunales de Distrito no se han apartado sustancialmente de estas doctrinas, pero sí las han matizado, regresando a una interpretación quizás más acorde con lo dispuesto en el artículo 1201(c)(1) de la DMCA. Esta corriente fue iniciada en 2009 por el Tribunal de Distrito de California, en el caso *Realnetworks, Inc. v. DVD Copy Control Ass'n*<sup>402</sup>. En la sentencia, confirmada en apelación, el juez declara que la doctrina del fair use no constituye una excepción legal al tráfico comercial de productos usados para inutilizar mecanismos tecnológicos de protección tanto de acceso como de copia establecidos en el artículo 1201(a) y (b). Sin embargo, el fair use entra en juego en el contexto del propio acto de inutilización.

El juez distingue entre el carácter de los mecanismos tecnológicos a inutilizar a la hora de considerar la aplicación del fair use.

En primer lugar, y manteniendo la postura de la doctrina *Universal v Reimerdes*, afirma que no cabe aplicación del fair use a las medidas tecnológicas de protección del acceso-control de la obra.

En cuanto a las medidas de protección anticopia, sin embargo, entiende que sí cabe aplicar la doctrina del fair use porque el texto del artículo 1201(b) no las menciona; ergo, aplicando el principio general de permisión, no prohíbe usos individuales concretos de inutilización. La prohibición de conductas de inutilización individuales solo entra en juego respecto a las tecnologías que protegen el acceso, no respecto de tecnologías anticopia, por lo tanto, la DMCA recoge implícitamente una excepción de fair use al usuario legítimo de la obra<sup>403</sup>.

A la vista de lo anterior, sobre la relación entre la excepción del artículo 1201(f) y la doctrina del fair use, cabe extraer las siguientes conclusiones:

1ºLa excepción del artículo 1201(f) tiene un ámbito más restringido que la doctrina del caso *Sega* basada en el fair use, a efectos de permitir la

---

<sup>402</sup> *Realnetworks, Inc. v. DVD Copy Control Ass'n*, No.C 08-04548, 641 F. Supp. 2d; 2009 U.S. Dist. Motion granted by, Claim dismissed by *Realnetworks, Inc. v. DVD Copy Control Ass'n*, 2010 U.S. Dist. LEXIS 1433 (N.D. Cal., Jan. 6, 2010), pág. 913.

<sup>403</sup> *Ibidem*, pág. 942.

ingeniería inversa para garantizar la interoperabilidad. La doctrina Segal permite la ingeniería inversa para cualquier razón legítima, entre ellas la interoperabilidad, con lo que deja la puerta abierta a la aparición de nuevas razones legítimas.

2º La doctrina del fair use evita tener que decidir sobre el carácter infractor del programa resultante de un acto de ingeniería inversa de manera simultánea al examen de la procedencia de la excepción. Siguiendo la doctrina Segal, el carácter infractor de la nueva obra es examinado una vez se ha establecido la procedencia o improcedencia del fair use. Según la norma establecida en el artículo 1201(f), cuándo ha de examinarse este aspecto no queda claro, porque por un lado la norma menciona de que el programa resultante ha de ser una obra de creación independiente en los tres apartados del artículo, sin dar ninguna indicación al respecto. Por otra parte, en el apartado tercero, al referirse a la información adquirida a través del proceso de ingeniería inversa para crear un programa interoperable afirma que si el uso que se le dé constituye infracción del copyright, quedará fuera del ámbito de la excepción. Da la impresión de que la excepción de la DMCA une las dos partes del examen elaborado en la doctrina Segal en un único paso, invitando a la ambigüedad y a la incongruencia. Muestra de ello es el caso Davidson donde el tribunal no entra a valorar si se había infringido la prohibición del artículo 1201(a)(1) en el examen de la excepción, sino que en su lugar, valora la existencia de infracción de copyright por parte de los demandados y a continuación al examinar la procedencia de la excepción establece que no procede no porque se haya contravenido la prohibición, sino porque los demandados han infringido el copyright sobre la obra<sup>404</sup>.

3º El artículo 1201(f) ha demostrado ser una herramienta poco eficaz para garantizar la interoperabilidad informática. Si bien las decisiones del caso Chamberlain como Lexmark proporcionan importantes criterios delimitativos del ámbito de protección de la DMCA, no ofrecen a los

---

<sup>404</sup> Davidson & Associates v. Jung et al., *op.cit.*, pág. 641 y 642.

personas que desarrollan tecnologías interoperables seguridad jurídica suficiente. Muestra de ello es que tanto programadores como aquellos que abogan por el desarrollo de tecnologías interoperables han buscado soporte legal en otras áreas del derecho para contrarrestar las restricciones a la interoperabilidad que legitima la DMCA, especialmente el derecho de defensa de la competencia.

#### **4. Conclusión de capítulo.**

El examen de la normativa y de la jurisprudencia norteamericana en relación a garantía de la interoperabilidad informática en la Copyright Act ponen de manifiesto la existencia de un debate para el que todavía no hay una solución que satisfaga a las dos partes enfrentadas.

Dos son las cuestiones que este debate plantea a la protección otorgada por el copyright al programa de ordenador. La primera se circunscribe al ámbito de protección sobre el programa, en concreto a si éste alcanza un elemento del programa denominado “interfaces”; la segunda, a la licitud de los actos que tienen lugar durante un proceso de descompilación cuya finalidad es acceder a la información interoperable del programa, que se contiene en las mencionadas interfaces.

Respecto de la primera cuestión, aunque la más reciente jurisprudencia ha declarado que un tipo determinada de interfaces, las APIs, no pueden estar protegidas por el copyright porque las mismas son métodos de operación, sigue sin existir un criterio único sobre la cuestión. La importancia de esta materia para el sector de las tecnologías de la información y de la comunicación, no debiera pasar desapercibida para el legislador, ni para el Tribunal Supremo de presentársele la oportunidad de establecer un criterio único que proporcione seguridad jurídica a los operados del mercado y a los consumidores.

Respecto de la segunda cuestión, es cierto que desde el caso *Sega* la doctrina seguida por la práctica totalidad de los tribunales estadounidenses es la de considerar que “la copia del código cuando se realiza un proceso de ingeniería inversa con la finalidad de extraer la información de las interfaces para alcanzar interoperabilidad se considera fair use y por tanto no infringe el copyright del programa<sup>405</sup>”.

---

<sup>405</sup> *Sega v. Accolade*, *op.cit.*, pág. 1527-28.

Sin embargo, desde la entrada en vigor de la DMCA la dificultad de gestionar políticas equilibradas sobre la interoperabilidad se ha vuelto a poner de manifiesto. Este hecho, unido a que desde el caso Segal el incremento de solicitudes de patentes para las interfaces de software ha sido llamativo<sup>406</sup>, reflejan que la cuestión de la interoperabilidad solo encontrará una solución de equidad cuando los legisladores rechacen las políticas legislativas que permiten la concesión de derechos de propiedad intelectual generadores de grandes desequilibrios y se comprometan con una política de innovación abierta y multifacética.

---

<sup>406</sup> P. SAMUELSON, "Are Patents on ..., *op.cit.*, pág. 1956.



### **Capítulo 3**

**El debate sobre la interoperabilidad informática en el derecho de autor europeo.**





## **1. La protección de los programas de ordenador en el derecho europeo antes de la Directiva 91/250/CEE de 14 de mayo de 1991.**

La búsqueda del sistema de protección idóneo para los programas de ordenador se planteó en territorio europeo, al igual que en los Estados Unidos, durante la década de los años sesenta del siglo XX. Ciertas instituciones al igual que jueces y académicos aportaron sus opiniones acerca de la mejor o peor adecuación de los distintos derechos de propiedad especial a esta clase de obra. Aunque el derecho de patente parecía a priori el sistema de protección más adecuada para los programas, razones políticas y económicas llevaron a que la práctica totalidad de los países europeos la rechazaran y optaran por la proporcionada por el derecho de autor<sup>407</sup>.

El primer país europeo que rechazó la protección de los programas de ordenador a través del derecho de patente fue Reino Unido. En mayo de 1962 se presentó ante la Oficina de Patentes británica la primera solicitud para un programa de ordenador. En 1965, fue rechazada por considerar los examinadores que el programa era una información intelectual<sup>408</sup>.

Poco tiempo después, el legislador francés excluyó a los programas de ordenador de la materia patentable en el artículo 7 de la Ley de 2 de enero de 1968 que modificaba la Ley de patentes<sup>409</sup>.

Este rechazo al sistema de patente se trasladó al ámbito internacional a principios de los años setenta.

El primer paso se dio con el Reglamento de Tratado de Cooperación en materia de Patentes (PCT) adoptado el 19 de junio de 1970<sup>410</sup>. El objeto del Tratado, como

---

<sup>407</sup> Vid. supra, capítulo 1.2.2.

<sup>408</sup> *Slee and Harris' Applications*, de 4 de octubre de 1965 y de 25 de noviembre de 1965, RPC 1966, pág. 197. En concreto, la decisión declaraba: "a product comprising merely intellectual information is not within the meaning given the Word «product»". El mismo criterio fue seguido en Austria: Decisión de 28 de marzo de 1968 "Ex parte Hufnagel", GRUR Int., 1968, pág. 381

<sup>409</sup> Loi n° 68-1 du 2 janvier 1968 tendant a valoriser l'activite inventive et a modifier le regime des brevets d'invention; JO du 03/01/1968, pág. 13. Art. 7: *Ne constituent pas, en particulier, des inventions industrielles: (...) 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.* (No constituyen en particular, invenciones patentables (...) 3. Los métodos financieros o contables, las reglas de juego, cualquiera otros sistemas de carácter abstracto y en especial, los programas o series de instrucciones para el desarrollo de operaciones de una máquina calculadora [El subrayado es nuestro]).

indica su Preámbulo, es evitar la dispersión de esfuerzos. En el artículo 39.1, apartado vi), exime a las administraciones encargadas de la obligación de realizar la búsqueda internacional en una solicitud de patente cuyo objeto sea un programa de ordenador. Con esta exención implícitamente la norma internacional sugiere que realizar una búsqueda internacional en una solicitud de patente de programa de ordenador, es un esfuerzo fútil.

El paso definitivo se dio en 1973 con la aprobación del Convenio de Múnich para la Concesión de la Patente Europea. Su artículo 52.2, letra c) establece de una manera clara y rotunda que el programa de ordenador, en sí mismo considerado, no es materia patentable<sup>411</sup>.

Al tiempo que se rechazaba la patente como sistema de protección del programa de ordenador, el derecho de autor se erigió como la mejor opción para muchos legisladores nacionales europeos. 1985 fue un año clave a estos efectos.

En Alemania, a pesar del silencio de la Ley de Derecho de Autor de 1965 (*Urheberrechtsgesetz*) sobre los programas de ordenador, su doctrina ya había valorado la idoneidad de los distintos sistemas de propiedad intelectual e industrial para proteger los programas de ordenador a principios de los años setenta<sup>412</sup>. Fue la modificación de la Ley de Derecho de Autor de 1965 a través de la Ley de 24 de junio de 1985 la que

---

<sup>410</sup> Reglamento del Tratado de Cooperación en materia de patentes (PCT), hecho en Washington el 19 de junio 1970. BOE núm. 267, de 7 noviembre 1989, pág. 26066; disponible en: <http://www.wipo.int/treaties/es/registration/pct/index.html> (última consulta 1 de junio, 2013).

<sup>411</sup> Artículo 52.2, letra c) del Convenio sobre Concesión de Patentes Europeas (CPE), hecho en Múnich el 5 de Octubre de 1973. España, Instrumento de Adhesión 10 de julio de 1986, BOE núm. 234, 30 septiembre 1986, pág. 2994. No corresponde al ámbito de este trabajo el examen de las denominadas “patentes de software”, sin embargo, sí ha de matizarse que el artículo 52.2 del Convenio sólo excluye la patentabilidad del programa como tal considerado. Sobre la patentabilidad de los programas de ordenador vid: R., NACK, *Die patentierbare Erfindung unter den sich wandelnden Bedingungen von Wissenschaft und Technologie*, Max-Planck-Institut für Geistiges Eigentum, Wettbewerbs- und Steuerrecht, Band 121, Köln, Berlin, Bonn, München, Carl Heymanns, 2002; R., BAKELS, “Are software patents something special?” en E., AREZZO, G., GHIDINI, (eds.), *Biotechnology and Software Patent Law. A comparative review of new developments*, Edward Elgar, Cheltenham, UK, Northampton, USA, 2011, pág. 131.

<sup>412</sup> E., ULMER, *Der Urheberschutz wissenschaftlicher...*, *op.cit.*, pág. 9; P., MÖHRING, “Die Schutzfähigkeit von Programmen für Datenverarbeitungsmaschinen”, GRUR (1967), pág. 269; P., SIDLER, *Der Schutz von Computerprogrammen im Urheber- und Wettbewerbsrecht*, Basel, Suiza, Verlag für Recht und Gesellschaft AG, 1969; M. KINDERMANN, “Special Protection Systems for Computer Programs – A Comparative Study”, IIC 7(3), (1976), pág. 301. Para consultar el texto de 1965 vid. F., BEIER, (ed.) asistido por D., LLEWELYN, D., STAUDER, *German industrial property, copyright and antitrust laws: legal texts with introd.*, IIC-studies vol. 6, Chemie, Weinheim, 1983.

añadió al artículo § 2, donde se enumeran entre las obras protegidas, los “*Computerprogramme*” en su apartado (1)<sup>413</sup>.

Hasta 1985 los tribunales alemanes habían tenido que lidiar ya en más de una ocasión con la cuestión de otorgar protección a través del derecho de autor a los programas de ordenador. El primer caso tuvo lugar en 1981, en el que el tribunal de distrito de Kassel concedió protección como obra de derecho de autor a un programa de ordenador<sup>414</sup>. Ese mismo año, el tribunal de distrito de Mannheim decidía en sentido contrario, motivando su decisión en que el programa de ordenador carecía de contenido estético-intelectual. Sin embargo esta decisión fue rechazada en apelación y confirmada por el BGH (Tribunal Supremo alemán) en 1985, más conocida como caso «*Inkasso-Programm*»<sup>415</sup>.

En 1985 fue aprobada en Francia la Ley 85-660 de 3 de Julio que modificaba parcialmente la Ley 57-298 de Propiedad Literaria y Artística de 11 de marzo de 1957<sup>416</sup>. Su artículo 3 declaraba que los programas de ordenador eran obras protegidas por el derecho de autor. Pero el legislador francés dio un paso más y estableció un régimen especial para los programas de ordenador, limitando su protección a lo dispuesto en el título quinto<sup>417</sup>.

En Reino Unido, fue aprobada la “*Copyright Computer Software Amendment Act 1985*”, que permitía aplicar la *Copyright Act* de 1956 a los programas de ordenador, protegiéndolos como obras literarias. Fue una medida temporal hasta la aprobación de la

---

<sup>413</sup> § 2 (1) *Zu den geschützten Werken der Literatur, Wissenschaft und Kunst gehören insbesondere: 1. Sprachwerke, wie Schriftwerke, Reden und Computerprogramme.* (Entre las obras literarias, científicas y artísticas protegidas se encuentran en particular: 1. Obras literarias como escritos, discursos y programas de ordenador).

<sup>414</sup> LG de Kassel, en sentencia de 12 de mayo de 1981 (BB 1983, pág. 992) confirmada por OLG Frankfurt en sentencia de 6 de noviembre de 1984 (BB 1985, pág. 139 y CR 1986, pág. 13).

<sup>415</sup> El tribunal LG Mannheim, en su sentencia de 12 de junio de 1981 (BB 1981, pág. 1543) había rechazado la protección del programa a través del derecho de autor por carecer de contenido estético-intelectual. Sin embargo, como ha indicado algún autor, la Ley de Derecho de Autor no exige ningún componente estético para conceder protección a una obra. Vid. U., Loewenheim, “Legal Protection of Computer Programs in West Germany”, High Tec. L.J., 1989, vol. 4, issue 2, pág. 187 (205). La sentencia fue rechazada en apelación (OLG Frankfurt sentencia de 9 de febrero de 1983 – GRUR 1983, pág. 300) y confirmada por el BGH en la sentencia de 9 de mayo de 1985, más conocida como *Inkasso-Programm* (Sentencia del 9 de mayo de 1985, BGHZ 94, pág. 276 y GRUR 1985, pág. 1041).

<sup>416</sup> Loi n° 85-660 du 3 juillet 1985 relative aux droits d'auteur et aux droits des artistes-interprètes, des producteurs de phonogrammes et de vidéogrammes et des entreprises de communication audiovisuelle, modifie Loi n° 57-298 du 11 mars 1957; JO du 04/07/1985.

<sup>417</sup> *Titre V : Des logiciels. Articles 45-51 de la Loi 85-660.* (Título V: De los programas de ordenador. Artículos 45 a 51 de la Ley 85-660).

“*Copyright, Designs and Patents Act 1988*”. Esta última ley supuso un importante acercamiento entre el sistema del copyright y el sistema continental de *droit d’auteur*. En la misma se incluían expresamente los programas de ordenador entre las obras del art. 3(1) (b), manteniendo el mismo régimen que el de la ley de 1985<sup>418</sup>.

En España, no fue hasta la Ley de Propiedad Intelectual 22/1987 de 11 de noviembre que se declaró expresamente la protección del programa de ordenador como obra protegida por derecho de autor<sup>419</sup>. Esta norma dedicaba un título especial a los programas, al igual que el derecho francés, que establecía el régimen de protección específico para los programas<sup>420</sup>.

A pesar de que el legislador español tardó más en declarar expresamente la protección del programa como obra de derecho de autor, sin embargo la doctrina se había situado a la par que la alemana cuando, en 1969 GÓMEZ SEGADÉ planteó la problemática de la protección de los programas para los derechos sobre bienes inmateriales<sup>421</sup>. Por desgracia, el resto de la doctrina española no se manifestó sobre esta cuestión hasta mediados de la década de los ochenta, coincidiendo con los movimientos de ámbito europeo e internacionales expuestos<sup>422</sup>.

---

<sup>418</sup> *United Kingdom Copyright Act 1956, 4&5 Eliz. 2, ch. 74; Copyright Computer Software Amendment Act 1985, ch.41; Copyright, Designs and Patents Act 1988, ch.48* (disponible en: <http://www.legislation.gov.uk/browse> (última consulta 1 de junio, 2013)). Un comentario sobre la protección de los programas como obras de derecho de autor en la leyes de 1985 y 1998 vid. D., BAINBRIDGE, *Legal Protection of Computer Software*, Sussex, 5ª ed., Tottel Publishing, 2008, pág. 54; R., DURIE, “UK Copyright, Designs and Patents Act 1988” IIC 20(5), (1989), pág. 637.

<sup>419</sup> Ley 22/1987, de 11 de noviembre, de Propiedad Intelectual. Boletín Oficial del Estado, núm. 275 de 17 de noviembre de 1987, pág. 34163 a 34176. Conviene destacar que algún autor como PINO ABAD ha respaldado que bajo la vigencia de la LPI de 1879 y su Reglamento de desarrollo era perfectamente defendible considerar que los programas de ordenador estaban protegidos por el régimen de propiedad intelectual. (Vid. M., PINO ABAD, “La protección de los programas de ordenador en la Ley de Propiedad Intelectual de 11 de noviembre de 1987”. *Derecho de los Negocios* 8, (1988), pág. 7). En contra de su postura algunos autores han argumentado que la ausencia de una declaración expresa por parte del Tribunal Supremo confirmando esa posibilidad, impide defender esa interpretación de la Ley. (Vid. M., HEREDERO HIGUERAS, “La protección de los programas de ordenador en el proyecto de Ley de Propiedad Intelectual”, RCDI 63, nº 581 (1987), pág.1021 (1023); J.P., APARICIO VAQUERO, J., DELGADO ECHEVERRÍA, “Comentarios al Libro Primero, Título VII” ..., *op.cit.* pág. 1243).

<sup>420</sup> Un comentario general sobre estos artículos en J.P., APARICIO VAQUERO, J., DELGADO ECHEVERRÍA, “Comentarios al Libro Primero, Título VII” ..., *op.cit.* pág. 1229.

<sup>421</sup> J.A., GÓMEZ SEGADÉ, “La protección jurídica de ...”, *op.cit.*, pág. 413.

<sup>422</sup> E., GALÁN CORONA, “En torno a la protección jurídica del soporte lógico”, en *Revista de Derecho Comercial y Obligaciones* 93, (1983), pág. 329; M., PINO ABAD “La protección de...”, *op.cit.*, pág. 7; Anuario AIPPI, III, 1984, pág. 182 (Respuesta a la cuestión Q 57 para el Consejo de Presidentes en Atenas, noviembre 1983), disponible en: [www.aippi.org](http://www.aippi.org), (última consulta 1 de junio, 2013).

Una vez rechazado de forma cuasi unánime el derecho de patente como sistema de protección del programa de ordenador y adoptado el derecho de autor como idóneo, sin embargo, las diferencias entre los ordenamientos nacionales europeos no hicieron sino seguir demostrando la necesidad de armonización en esta materia.

### *1.1. El camino hacia la Directiva sobre protección jurídica de los programas de ordenador.*

El punto de partida del camino hacia la Directiva se sitúa también en 1985, año en el que la Comisión Europea (EC) presentó el Libro Blanco “La realización del mercado interior”<sup>423</sup>.

Entre las áreas identificadas por la Comisión como necesitadas de armonización para conseguir un mercado interior, se encontraban los derechos de propiedad intelectual e industrial. En concreto, el Libro Blanco indicaba la urgente necesidad de armonización de los derechos de propiedad intelectual e industrial sobre tres tecnologías en particular. Una de ellas eran los programas de ordenador<sup>424</sup>.

Las ideas germinales del Libro Blanco fueron plasmadas en el Libro Verde de 1988, titulado “Libro Verde sobre los derechos de autor y el desafío tecnológico, problemas de derechos de autor que requieren una acción inmediata”<sup>425</sup>. El capítulo quinto estaba dedicado en exclusiva a los programas de ordenador, y entre otras cuestiones, la Comisión manifestó su preocupación acerca de la armonización en las siguientes materias: el objeto y el ámbito de protección del derecho de exclusiva, la condición de autor del programa, la infracción de los derechos del titular y las excepciones al derecho exclusivo del titular.

Todas estas materias identificadas por la Comisión dieron lugar a un acalorado debate en el proceso de elaboración de la Directiva cuando se trató la cuestión de garantizar la consecución de interoperabilidad informática a través de excepciones legales al derecho de autor del titular del programa.

---

<sup>423</sup> *Completing the Internal Market. White Paper from the Commission to the European Council* (Milan, 28-29 June 1985). Doc. COM (85) 310 final, de 14 de junio de 1985.

<sup>424</sup> Chips semiconductores e invenciones biotecnológicas eran las otras dos.

<sup>425</sup> Libro Verde sobre los derechos de autor y el desafío tecnológico, problemas de derechos de autor que requieren una acción inmediata, Doc. COM(88) 172 final, de 23 de agosto de 1988.

Para comprender las claves y consecuencias de este debate, cuyo resultado final fue el artículo 6 de la Directiva, es preciso analizar las diferencias entre los ordenamientos nacionales europeos sobre conceptos clave como originalidad de la obra y el contenido de algunos derechos de exclusiva como la reproducción y transformación de la obra.

### *1.2. Principales diferencias entre los regímenes nacionales de derecho de autor con efectos sobre la interoperabilidad.*

Las principales diferencias entre los regímenes nacionales que afectan directamente al debate sobre la interoperabilidad giran en torno a:

1.2.A. La originalidad del programa.

1.2.B. Las facultades exclusivas de reproducción y de transformación de la obra.

#### *1.2.A) La originalidad del programa.*

##### *A.1) Alemania.*

En Alemania, la originalidad de un programa de ordenador a efectos del derecho de autor ya había sido abordada por algunos tribunales de instancia a principios de los años ochenta<sup>426</sup>. Curiosamente, el primer alto tribunal en tratar la cuestión no fue el BGH, sino el Tribunal Federal Laboral (Bundesarbeitsgericht) en 1983. En este caso la corte no solo afirmó que los programas de ordenador eran obras protegibles por derecho de autor con carácter general, sino que añadió que cualquier programa podría cumplir con el requisito de la originalidad de ser “un creación personal intelectual”<sup>427</sup>.

Al mismo tiempo que la Ley de Derecho de Autor alemana era modificada en 1985, el Tribunal Supremo Alemán (Bundesgerichtshof) abordaba esta cuestión en famoso caso *Inkasso-Programm*<sup>428</sup>.

---

<sup>426</sup> Vid. infra, cap. 3.1.

<sup>427</sup> Sentencia de 13 de septiembre de 1983 (GRUR 1984, pág. 429).

<sup>428</sup> Sentencia del 9 de mayo de 1985, BGHZ 94, pág. 276 y GRUR 1985, pág. 1041.

El BGH aplicó un test de dos fases para determinar si el programa era original en el sentido indicado por el artículo 2(2) de la Ley de Propiedad Intelectual, es decir, si el programa podría considerarse una “creación intelectual personal”<sup>429</sup>.

En primer lugar, el Tribunal examinó el proceso de elaboración de un programa de ordenador distinguiendo tres etapas: Una inicial de planteamiento general del problema a resolver, en la que el programador confecciona un estudio, denominado especificaciones, que es expresado a través de signos matemáticos, números y lenguaje convencional. Este estudio puede considerarse una obra literaria<sup>430</sup>. En la segunda fase de creación, se elabora un diagrama de datos, que podría calificarse como representación de carácter técnico o científico. En la tercera etapa de creación, el programa es codificado, y tanto sea en código fuente como en código objeto, el resultado final puede considerarse una obra literaria<sup>431</sup>.

Por tanto, de acuerdo al análisis del Tribunal, el programa de ordenador podía considerarse tanto una obra literaria como una representación de carácter científico o técnico<sup>432</sup>.

La segunda parte del examen partía del artículo 2(2), es decir, si bien el programa podía calificarse tanto como obra literaria como científica, dependiendo de en qué fase de elaboración del programa se centrara el análisis, sólo aquellos programas que cumplieran con el requisito de la originalidad serían merecedores de protección. Tendrían que constituir una creación personal intelectual.

Al examinar el significado de “intelectual”, el BGH mantuvo que los procesos del intelecto pueden “*cristalizar y ser expresados mediante la elaboración y desarrollo de ideas en el contenido y/o a través de maneras particularmente ingeniosas y las características de la recopilación, organización y disposición del material presentado*”<sup>433</sup>. Sin embargo, el componente intelectual está limitado en las obras de carácter científico o técnico (utilitario) porque “*las enseñanzas y composiciones científicas forman parte del dominio público y por tanto son accesibles por cualquiera*”

---

<sup>429</sup> § 2(2) “(...) persönliche geistige Schöpfungen”

<sup>430</sup> Sentencia de 9 de mayo de 1985, GRUR 1985, pág. 1046-1047.

<sup>431</sup> Ibidem, pág. 1046-1047.

<sup>432</sup> Ibidem, pág. 1046-1047.

<sup>433</sup> Ibidem pág. 1047. El texto original dice: “*Der geistige Gedankeninhalt findet seinen Niederschlag und Ausdruck in der Gedankenformung und -führung des dargestellten Inhalts und/oder der besonders geistvollen Form und Art der Sammlung, Einteilung und Anordnung des dargebotenen Stoffs (vid. BGH, Urt. V 29.3.1984 –I ZR 32/82 en GRUR 1984, pág. 659, (660) Ausschreibungsunterlagen*”.

y “su diseño y representación, en la medida que sean necesarias por razones científicas, son expresados habitualmente en el lenguaje técnico propio de cada área científica, que por lo mismo carecen de la necesaria originalidad y creatividad”<sup>434</sup>. Por lo tanto, en un programa de ordenador, la regla computacional específica, el algoritmo, no está protegido por el derecho de autor<sup>435</sup>.

En resumen, la protección del derecho de autor para un programa de ordenador requiere en primer lugar que el programa presente individualidad creativa suficiente. Este requisito se satisface cuando el programa, en relación a la impresión general que muestra, puede distinguirse claramente de programas anteriores y no representa únicamente un progreso mecánico o técnico de algo ya conocido. En segundo lugar, la elaboración del nuevo programa debe implicar algo más que la sola labor realizada con las manos, esto es, más que una secuencia y combinación mecánica o técnica. La actividad de estructuración realizada por el creador del programa, en lo que respecta a la selección, reunión, ordenación y clasificación de la información y comandos, debe elevarse por encima de la capacidad media común. Solo cuando se cumplan ambas condiciones, un programa de ordenador podrá gozar de la protección del derecho de autor<sup>436</sup>.

La originalidad de un programa de ordenador según el BGH radicaba en la existencia de cierto grado de particularidad o peculiaridad creativa, que debe contraponerse a la actuación de un programador medio, cuyas realizaciones puramente artesanales, de ordenación y unión según criterios mecánico-técnicos, quedan fuera de toda posibilidad de protección.

Este criterio de originalidad mantenido por el BGH fue duramente criticado por la doctrina, por acercarse peligrosamente al concepto de “originalidad objetiva” tradicionalmente aplicado al derecho de patente<sup>437</sup>.

---

<sup>434</sup> Ibidem pág. 1047. El texto original dice: “(...) die wissenschaftliche Lehrer und das wissenschaftliche Ergebnis sind urheberrechtlich frei und jedermann zugänglich” und “ihrer Darstellung und Gestaltung fehlt, soweit diese aus wissenschaftlichen Gründen in der gebotenen Form notwendig und durch die Verwendung der im fraglichen technischen Bereich üblichen Ausdrucksweise üblich sind, die erforderliche eigenschöpferische Prägung (BGH in GRUR 1981, 352, 353 Staatsexamensarbeit)”.

<sup>435</sup> Ibidem, pág. 1048.

<sup>436</sup> Ibidem, pág. 1048.

<sup>437</sup> G. SCHULZE, “Urheberrechtsschutz von Computerprogrammen – geklärte Rechtsfragen oder blosse Illusion”, GRUR 1985, pág. 997; K.A., BAUER, “Rechtsschutz von Computerprogrammen in der Bundesrepublik Deutschland - eine Bestandsaufnahme nach dem Urteil des Bundesgerichtshof von 9. Mai 1985”, CR 1, 1985, pág. 5; H., HABERSTUMPF, “Grundsätzliches zum Urheberrechtsschutz von Computerprogrammen nach dem Urteil des Bundesgerichtshofs vom 9. Mai 1985”, GRUR 3, 1986, pág.



En primer lugar, el test de originalidad aplicado por el BGH era más similar al aplicado para determinar la patentabilidad que los que se habían venido usando para determinar la protección por derecho de autor. Comparar el programa en cuestión con los programas preexistentes es, en realidad un test de novedad, incluso aunque el Tribunal explícitamente lo niegue<sup>438</sup>. La premisa de si “determinados componentes se encuentran presentes en obra anteriores” es idéntica a la de determinar la novedad de esos componentes. La novedad, sin embargo, es un requisito de la patente pero no lo es para el derecho de autor<sup>439</sup>. Además, una invención será patentable si cumple con el requisito de la actividad inventiva, y esta no se cumple si la actividad es obvia o se deriva evidentemente del estado de la técnica. Por el contrario, el test de originalidad de derecho de autor no considera si hay actividad inventiva.

En segundo lugar, la decisión del BGH limita soberamente la protección de los programas de ordenador. En el momento que se dictó esta sentencia los programas de ordenador necesitaban ser protegidos contra la copia ilícita porque no existía un sistema protección eficaz para frenar la piratería y dicha ineficacia se vería incrementada con un criterio de originalidad tan estricto para los programas de ordenador. En la práctica, argumentaba la doctrina, la aplicación de este criterio de originalidad supondría que alrededor del 90% de los programas existentes en el mercado alemán no podrían ser protegidos por derechos de autor<sup>440</sup>.

En tercer lugar, el análisis del BGH requiere una aproximación extremadamente teórica y conlleva un montón de problemas en su aplicación práctica. Muchas cuestiones implícitas en el test del Tribunal son difíciles de responder. ¿Cuál es el nivel de las habilidades medias de un programador? ¿Cuándo un programa “claramente sobrepasa” ese nivel? ¿Qué elementos de un programa deben superar dicho nivel? Y,

---

222; M., RÖTTINGER, “Abkehr vom Urheberrechtsschutz für Computerprogramme?”, IuR 1, 1986, pág. 12; H.W., MORITZ, B., TYBUSSECK, *Computersoftware : Rechtsschutz und Vertragsgestaltung : eine fächerübergreifende Darstellung nach deutschem und EG-Recht*, München, 2ªed., Beck, 1992, pág. 134-146.

<sup>438</sup> Ibidem, pág. 1047: “Dieser Vergleich enthält keine – für die Urheberrechtsschutzfähigkeit unerhebliche – Neuheitsprüfung (...)”.

<sup>439</sup> U., LOEWENHEIM, *Geschützte Werke*, en G. SCHRICKER (ed.), *Urheberrecht: Kommentar*, München, Beck, 1987, §2 (20).

<sup>440</sup> J. DREXL, *What is protected in...*, *op.cit.*, pág. 47; T., HOEREN, “The protection of software in the Federal Republic of Germany – recent developments”, CLP, March/April, 1990, pág. 134; H., HABERSTUMPF, “Der urheberrechtliche Schutz ...”, *op.cit.*, pág. 73; U., LOEWENHEIM, “Abschnitt 2. Das Werk”..., *op.cit.*, §2 (80); E., SCHULZE, “Urheberrechtsschutz von Computerprogrammen - geklärte Rechtsfragen oder blosse Illusion”, GRUR 1985, pág. 997 (1002).

¿quién decide sobre las anteriores cuestiones? Nadie consiguió responder tales respuestas. Si bien algún autor propuso que la extensión y complejidad del programa podría ser considerada como una indicación de su originalidad<sup>441</sup>, en realidad esta propuesta contraría los principios fundamentales del derecho de autor según los cuales, la originalidad no puede medirse por el tamaño de la obra<sup>442</sup>.

Lo cierto es que después del caso *Inkasso-Programm*, los tribunales alemanes mostraron una postura más flexible que la mantenida por el Tribunal Supremo a la hora de valorar la originalidad. A estos efectos, algunas salas declararon que los programas de ordenador podían ser protegidos como obras audiovisuales (*Laufbildschutz*) en el sentido del artículo 95 de la Ley de Propiedad Intelectual, aunque no cumpliesen el requisito de originalidad específico establecido por *Inkasso-Programm*<sup>443</sup>.

A pesar de estos intentos, el criterio objetivo de originalidad establecido en *Inkasso-Programm* fue confirmado, clarificado y desarrollado en 1990 por el Tribunal Supremo en el caso *Betriebssystem*<sup>444</sup>.

El Tribunal Supremo reformuló el concepto de originalidad de un programa de ordenador del siguiente modo:

1ºEl nivel mínimo de creatividad para que la obra sea susceptible de protección por derechos de autor sólo se alcanza si se excede de manera evidente el estándar de los resultados alcanzados por un programador en sus tareas diarias, las cuales se basan en unos procedimientos generalmente rutinarios, manuales, que parten de una secuenciación mecánica o técnica<sup>445</sup>.

---

<sup>441</sup> G., GRAVENREUTH, "Juristische relevante technische Fragen zur Beurteilung von Computer-Programmen", GRURInt. 1986, pág. 720, (725).

<sup>442</sup> U., LOEWENHEIM, "*Geschützte Werke*"..., *op.cit.*, §2 (23).

<sup>443</sup> Vid. OLG Karlsruhe, sentencia de 9 de septiembre de 1986, CR 1986, pág. 723; LG Hannover, sentencia de 28 de octubre de 1987 (18 O 58/87), CR 1988, pág. 826. Comentarios sobre esta tendencia pueden encontrarse en B., SYNDIKUS, "Computerspiele und Urheberrecht", CR 1988, pág. 819; G., GRAVENREUTH, "Computerspiele und Urheberrecht", CR 1987, pág. 161.

<sup>444</sup> BGH sentencia de 4 de octubre de 1990, I ZR 139/89 "Betriebssystem", GRUR 1991, pág. 449. (versión en inglés en IIC 22, 1991, pág. 723).

<sup>445</sup> *Ibidem*, pág. 451: "*Die Frage der schöpferischen Eigentümlichkeit bemißt sich nach dem Gesamtvergleich gegenüber vorbestehenden Gestaltungen. Lassen sich nach Maßgabe dieses Gesamtvergleichs schöpferische Eigenheiten feststellen, so sind diese dem durchschnittlichen Schaffen bei der Programmerstellung gegenüberzustellen. Eine für die Urheberrechtsschutzfähigkeit hinreichende Gestaltungshöhe wird erst erreicht, wenn das alltägliche, durchschnittliche Programmiererschaffen, das auf einer mehr oder weniger routinemäßigen, handwerkmäßigen, mechanisch-technischen Aneinanderreihung und Zusammenfügung des Materials beruht, deutlich überstiegen wird*".

2º Un algoritmo como tal no es susceptible de protección por no ser original, sin perjuicio de que la originalidad en el programa pueda encontrarse en la forma de conjugar y hacer interactuar diferentes algoritmos<sup>446</sup>.

3º La carga de la prueba de la falta de originalidad corresponde a la parte que alegue que el programa no es original<sup>447</sup>.

Con esta reestructuración del criterio de la originalidad se favorecía, al menos en parte, la eficacia del derecho de autor, principalmente al trasladar la carga de la prueba de la falta de originalidad a quien la utilizaba para defender una actuación, que de otro modo, constituiría una infracción.

#### A.2) Francia.

Según las disposiciones de la Ley de Propiedad Intelectual francesa de 1957, las obras protegidas son aquellas denominadas “oeuvres de l’esprit”<sup>448</sup>. La norma no aclara el significado de esta expresión. Es en el artículo siguiente, donde de manera indirecta se explica en parte el significado de esta expresión. En cualquier caso, la norma no detalla que características ha de reunir una obra para ser considerada “oeuvre de l’esprit”<sup>449</sup>. Por lo tanto, son la doctrina y la jurisprudencia las que determinan cuál ha de ser la originalidad requerida a los programas de ordenador<sup>450</sup>.

Para la doctrina francesa, la originalidad que debía mostrar un programa de ordenador era un tema controvertido. Si bien todas las corrientes académicas impulsaban la protección del programa en el terreno del derecho de autor, los criterios variaban. Para un sector liderado por GOUTAL, un programa sería original si

---

<sup>446</sup> *Ibidem* pág. 453: “dazu gehört auch die Verwendung von Algorithmen, die zwar als solche einem Urheberrechtsschutz nicht zugänglich sind (...), wohl aber in der Art und Weise der Implementierung und Zuordnung zueinander urheberrechtsschutzfähig sein können”.

<sup>447</sup> *Ibidem* págs. 450 y 451.

<sup>448</sup> Art. 2 Loi n° 57-298 du 11 mars 1957: *Les dispositions de la présente loi protègent les droits des auteurs sur toutes les oeuvres de l’esprit, quels qu’en soient le genre, la forme d’expression, le mérite ou la destination.*

<sup>449</sup> El artículo 3 enumera una lista ejemplificativa de obras consideradas “oeuvres de l’esprit”.

<sup>450</sup> Tradicionalmente el requisito de la originalidad puede ser entendido desde una concepción subjetiva, en tanto que traduzca la personalidad de su autor y no haya copiado una obra ajena, o desde una concepción objetiva, en cuanto a haber creado algo nuevo, no existente previamente, lo cual sería equiparable a la novedad de la patente. Sobre las posiciones doctrinales a favor de cada postura, vid: A., LUCAS, *La Protection des Créations Industrielles Abstraites*, Centre D’Études Internationales de la Propriété Industrielle, Paris, Libraries Techniques (LITEC), 1975, pág. 41; C., COLOMBET, *Propriété littéraire et artistique et droits voisins*, Paris, 9ª ed., Dalloz, 1999, pág. 85; U., LOEWENHEIM, “Artikel §2”, en U., LOEWENHEIM, A., DIETZ, G., SCHRICKER, *Urheberrecht : Kommentar*, München, 6ª ed., Beck, 2010, pág. 59; R., BERCOVITZ, “Artículo 10”, en R., BERCOVITZ RODRIGUEZ-CANO (coord.), *Comentarios...*, *op.cit.*, pág. 153.

manifestaba suficientemente la personalidad de su autor<sup>451</sup>. Para VIVANT sin embargo, lo definitivo era la originalidad en la forma, porque resultaría embarazoso calificar de original las elecciones que realizan los informáticos cuando programan si se comparan con la actividad realizada por un artista<sup>452</sup>. Una tercera corriente impulsada por DESJEUX y PLAISANT, consideraban que la originalidad en un programa debía ser necesariamente de carácter estético<sup>453</sup>. Es decir, la originalidad de una obra utilitaria como es un programa debía ser de carácter artístico, para no confundirla con la simple habilidad o la destreza profesional.

La jurisprudencia francesa, sin embargo, se situó en la línea de GOUTAL cuando tuvo ocasión de pronunciarse sobre la originalidad exigida a un programa de ordenador. Fue en la sentencia de 7 de marzo de 1986 del Tribunal Supremo (Cour de cassation)<sup>454</sup> la que declaró que el requisito de la originalidad se cumple cuando el programa lleva la impronta del aporte intelectual de su autor. El pasaje que define en este sentido la originalidad, dice: *“Considerando, en segundo lugar, que habiendo investigado como se solicitaba, si los programas elaborados por M. PACHOT eran originales, los jueces han estimado soberanamente sobre el fondo de la cuestión que su autor había dado muestras de un esfuerzo personalizado, que iba más allá de la simple elaboración de una lógica automática y forzosa, y una estructura individualizada; que frente a estas enunciaciones y comprobaciones, y abstracción hecha de los motivos más arriba citados, criticado por el recurso, la Corte de Apelaciones al resolver que los programas concebidos por M. PACHOT tenían el sello de su aporte intelectual, ha justificado legalmente su decisión sobre ese punto”*<sup>455</sup>.

---

<sup>451</sup> J.L., GOUTAL, “La protection juridique du logiciel”, D.Chron.197, no 9, 1984, pág. 197

<sup>452</sup> M., VIVANT, “Table ronde sur la protection juridique des logiciels”, JCP Les cahiers de droit de l’entreprise 3, 1983, pág. 2.

<sup>453</sup> X., DESJEUX, “Le droit d’auteur dans la vie industrielle”, RIDA julio 1975, pág. 137; R., PLAISANT, “La protection du logiciel par le droit d’auteur”, Gaz. Pal., n° 7, 1983, pág. 348.

<sup>454</sup> Asunto Babolat v. Pachot, sentencia de 7 de marzo de 1986 de la Corte de Casación, Asamblea Plenaria (Cour de cassation, AP), RIDA 129, 1986, pág. 130. Versión en inglés en IIC 1987, pág. 288.

<sup>455</sup> F., TOUBOL, (traducción de M., ESPANÉS) *El software : ..., op.cit.*, pág. 91. Texto original: *« Attendu, en second lieu, qu’ayant recherché, comme ils y étaient tenus, si les logiciels élaborés par Monsieur Pachot étaient originaux, les juges du fond ont souverainement estimé que leur auteur avait fait preuve d’un effort personnalisé allant au-delà de la simple mise en oeuvre d’une logique automatique et contraignante et que la matérialisation de cet effort résidait dans une structure individualisée ; qu’en l’état de ces énonciations et constatations, et abstraction faite des motifs ci-dessus cités, critiqués par le pourvoi, la Cour d’appel, qui a ainsi retenu que les logiciels conçus par Monsieur Pachot portaient la marque de son apport intellectuel, a légalement justifié sa décision de ce chef »*

Esta fórmula para determinar la originalidad fue recibida con asombro y alguna que otra crítica<sup>456</sup>. El Alto Tribunal francés había dejado claro que el concepto clásico de originalidad no era válido para esta clase de obras, sino que el programa, para ser original debía manifestar la impronta o sello personal de su autor.

El mismo día del fallo sobre el caso Pachot v. Babolar, el Tribunal Supremo juzgó la causa de Atari v. Williams, cuya cuestión de fondo versaba sobre la protección de derecho de autor sobre un videojuego. En relación a la originalidad el Tribunal declaró que las características estéticas no habían de tenerse en cuenta al examinar la originalidad del programa de ordenador, cuya esencia radicaba en la impronta intelectual del autor<sup>457</sup>.

Siguiendo los criterios del Tribunal Supremo francés, TOUBOL<sup>458</sup> elaboró una serie de conclusiones, de las que destacan las que se refieren a los efectos de adoptar esta concepción de originalidad en un programa de ordenador:

- 1º La originalidad debe apreciarse en el nivel de composición y de expresión y no en la fase del método empleado (en relación a las fases de creación del programa), porque “el programa de ordenador no constituye un simple método y su protección debe examinarse con relación al conjunto”.
- 2º Al admitir la originalidad en la composición, la protección al programa no se limita a su forma codificada, sino que se extiende en parte a las ideas, porque la frontera entre lo que es propio de un método y lo que corresponde a la composición es muy tenue, sobre todo considerada a nivel de organigrama.
- 3º Consecuencia de lo anterior, cualquier doctrina que abogue por una concepción de la originalidad basada en criterios estéticos debería ser desestimada.

Pero con la entrada en vigor de las modificaciones operadas por la Ley 85-660 de 3 de Julio de 1985 que modifica parcialmente la Ley 57-298 de Propiedad Literaria y Artística de 11 de marzo de 1957, no se añadió nada nuevo al requisito de la

---

<sup>456</sup> En este sentido, A., LUCAS y P. KAMINA, indican sobre la decisión del caso Pachot: “Unfortunately the coger courts were left with only, or at the best open ended, language that does not allow them to declinate any clear criterion of originality”. Vid. “France” en P.E., GELLER (ed.), *International Copyright Law and Practice*, Matthew Bender, New York, 2006, §2 [1][b][iii][B]

<sup>457</sup> Asunto Atari v. Williams *Electronics*, sentencia de 7 de marzo de 1986 del Tribunal Supremo, Asamblea Plenaria (*Cour de Cassation*, AP), RIDA 1986, pág. 126 (136); versión en inglés en IIC 18, 1987, pág. 550.

<sup>458</sup> F., TOUBOL, (Traducción de M., ESPANÉS), *El software...*, *op.cit.*, pág. 90 – 91.

originalidad de un programa de ordenador, manteniendo el criterio jurisprudencial establecido por las sentencias del Tribunal Supremo francés, que habían aplicado la Ley de Propiedad Intelectual de 1957.

Comparado con el criterio de originalidad del programa alemán, el criterio francés es subjetivo y basado en razones más de composición o gusto personal del autor (estética) que en criterios de calidad. Esto significaba que el argumento de que existiendo distintas opciones para resolver el mismo problema (que se quiere solucionar con el programa), el hecho de que el programado haya optado por una, imprimiendo su personalidad a la obra a través de esa decisión, acredita su originalidad, porque dicho programa será el resultado de su labor intelectual.

Como han indicado algunos autores, este criterio de originalidad francés, se acerca más a la concepción de originalidad para programas de ordenador norteamericana que a la alemana<sup>459</sup>, y suponía un problema a la hora de garantizar la protección efectiva de los programas en el espacio del mercado comunitario.

#### *A. 3) Reino Unido.*

En Reino Unido, la Computer Software Act de 1985 que modificó la Copyright Act de 1956, protegió los programas de ordenador como obras literarias. La “Copyright, Designs and Patent Act 1988” mantuvo intacto el régimen establecido en 1985.

Al igual que en Alemania, ya con anterioridad a la Ley de 1985, la jurisprudencia había tenido que lidiar con la cuestión de proteger o no los programas de ordenador a través del derecho de autor. Los jueces declararon que esta clase de obras podían protegerse como obras literarias en el sentido del Convenio de Berna<sup>460</sup>. Por lo tanto, para examinar la originalidad de un programa de ordenador en Reino Unido hay que acudir a la jurisprudencia que trate este requisito en relación con las obras literarias en general.

La primera interpretación clara sobre la originalidad de una obra literaria por el Tribunal Supremo británico (Chancery Division) tuvo lugar a principios del siglo XX,

---

<sup>459</sup> Vid., T., SHUSTER, “Originality in Computer Programs and Expert Systems”, *Transnat'l Law*, n° 5, (1992), pág. 13 (14); J., DREXL, *What is protected in...*, *op.cit.*, pág. 61.; M., NIMMER, D., NIMMER, *Nimmer on Copyright...*, *op.cit.*, pág. 2-15.

<sup>460</sup> Vid. *Gates v. Swift*, RPC 13, 1982, p. 339 (Ch. D.) sobre la protección de un programa como obra literaria bajo el Convenio de Berna; *Sega Enterprises Ltd. v. Richards*, FSR 73, 1983, (Ch. D.), sobre la protección por derechos de autor de un video juego; *Thrustcode Ltd v. W.W. Computing Ltd*, FSR 1983, 502 (Ch.D.), corresponde al demandante reproducir el código fuente para determinar si existe infracción.

definiéndola como la expresión de una idea por parte de su autor que no constituyese la copia de una obra previa. Este criterio pasó a denominarse la doctrina de “*the sweat-of-the-brow*”<sup>461</sup>:

*The word “original” does not in this connection mean that the work must be the expression of original or inventive thought. Copyright Acts are not concerned with the originality of ideas, but with the expression of thought, and, in the case of “literary works” with the expression of the thought in print or writing. The originality which is required relates to the expression of thought. But the Act does not require that the expression must be an original or novel form, but the work must not be copied from another work – that is should originate from the author.*<sup>462</sup>

Esta doctrina establecida por el Tribunal Supremo en 1916, suponía que una obra literaria, para ser original no requería ni novedad ni mérito intelectual, simplemente la obra debía haber sido creada por su autor y no ser la copia de otra anterior.

La doctrina de “*the sweat-of-the-brow*” fue desarrollada y matizada años más tarde en el conocido como caso Lego, en el cual el Tribunal Supremo añadió que para que una obra literaria fuese original, debía ser el resultado de las habilidades, esfuerzo y juicio del autor, convirtiéndose en la doctrina “*the skill and labour*”<sup>463</sup>. El principal problema que presentaba un estándar de originalidad tan minimalista es que, llevado a sus últimas consecuencias, una obra “collage” formada por la copia de distintas partes de otras obras, constituiría una obra original ya que la particular combinación de los diferentes elementos, serían el resultado de las habilidades, esfuerzo y juicio de su autor<sup>464</sup>.

---

<sup>461</sup> Para un comentario detallado sobre la doctrina “the sweat-of-the-brow” vid. J., GINSBURG, “Creation and Commercial Value: Copyright Protection of Works of Information”, Columbia L.R. 90, 7, (1990), pág. 1865 (1868).

<sup>462</sup> University of London Press Ltd v. The University Tutorial Press Ltd, sentencia del Tribunal Supremo (Chancery Division) de 26 de julio de 1916, ([1916] 2 Ch 601), pág. 608-09.

<sup>463</sup> “(...) result of skill, labour, or judgement” en Interlego AG v. Tyco International Inc, sentencia de 5 de mayo de 1989, Tribunal de Apelaciones de Hong-Kong ([1989] AC. 217) pág. 259.

<sup>464</sup> Vid. C., TAPPER, “The Software Directive: A UK Perspective”, en M., LEHMANN, C., TAPPER, C (eds.), *A Handbook of Software European Law...*, op.cit., pág. 147.

Aplicado a los programas de ordenador, este criterio admite la originalidad de un programa que sea creación de su autor, si las variaciones realizadas sobre programas anteriores son sustanciales, en términos cualitativos<sup>465</sup>.

En conclusión, el programa de ordenador es original aunque incluya material tomado de programas anteriores siempre que su autor haya incorporado mejoras que pueden ser tanto reformas que afecten al diseño, a la organización o al texto.

Esta interpretación jurisprudencial sitúa el concepto de originalidad requerido por los tribunales británicos en un nivel más bajo que el derecho francés y el derecho estadounidense<sup>466</sup>, y muy alejado del criterio alemán. Este detalle es importante, porque como se mostrará a continuación y han indicado algunos autores, la concepción de originalidad británica tuvo gran influencia en el texto de la Directiva europea de 1991<sup>467</sup>.

#### A.4) España.

El artículo 10 de la LPI de 1987 en su apartado 1, letra i) mencionaba expresamente los programas de ordenador como objeto protegible por medio de derecho de autor<sup>468</sup>. Dicha mención junto con el Título VII, cuyos artículos 95 a 100 regulaban específicamente esta clase de obras, constituyeron una novedad en España.

La Ley de Propiedad Intelectual de 1987 establecía un régimen específico para los programas de ordenador. Para algunos autores se trataba de un régimen de *ius singulare*<sup>469</sup>. Sin embargo, como ha afirmado GÓMEZ SEGADE, del enunciado del artículo 95 también cabe inferir que el régimen que establece la Ley de 1987 es en realidad un derecho especial, ya que no existe ninguna disposición de *ius singulare* en

---

<sup>465</sup> H., CARR, R., ARNOLD, *Computer Software: Legal Protection in the United Kingdom*, Sweet & Maxwell, 2º ed., Londres, 1992, pág. 54.

<sup>466</sup> La jurisprudencia norteamericana rechazó la doctrina “the sweat-of-the-brow” en el caso *Feist Publications, Inc. v. Rural Telephone Service Co.*, 499 U.S. 340 (1991); vid. supra, capítulo 2.1.3.A).

<sup>467</sup> J.P., APARICIO VAQUERO, J., DELGADO ECHEVERRÍA, “Comentarios al Libro Primero, Título VII” ..., *op.cit.*, pág. 1269; M., HEREDERO HIGUERAS, “Observaciones sobre el borrador de directiva comunitaria relativa a la protección jurídica de los programas de ordenador. Documento COM (88) 816 de 21 de diciembre de 1988”, *Tecnolegis* 1, octubre, 1989, pág. 14.

<sup>468</sup> Ley 22/1987 de 11 de noviembre de 1987, de Propiedad Intelectual, BOE núm. 275, 17 noviembre 1987, pág. 34163.

<sup>469</sup> J., DELGADO ECHEVERRÍA, J. “Comentarios al Libro Primero, Título VII”, en BERCOVITZ RODRÍGUEZ-CANO, R. (ed.) *Comentarios a la Ley de Propiedad Intelectual*, Madrid, 2ª ed., Tecnos, 1997, pág. 1244 y 1252; E., GALÁN CORONA, *Comentarios EDESA*, T.V., vol. 4ºB, Madrid, 1995, pág. 232.



los artículos 95 a 100, con lo cual las disposiciones generales del Libro I de la Ley de 1987 se aplican subsidiariamente al régimen del Título VII<sup>470</sup>.

En lo que respecta a la originalidad, la Ley de 1987 no establecía ningún criterio específico para valorar este requisito. Había que acudir al enunciado general del artículo 10 donde se afirmaba que “*son objeto de propiedad intelectual todas las creaciones originales literarias, artísticas o científicas (...)*”.

Por desgracia no hay pronunciamientos jurisprudenciales de la época anterior a la Directiva europea que traten la cuestión de la originalidad de los programas de ordenador<sup>471</sup>.

No obstante, la doctrina española sí se manifestó sobre esta cuestión, cuya postura puede resumirse en que la originalidad requerida a la obra para su protección jurídica a través del derecho de autor se entiende, frente a teorías subjetivas, como una novedad objetiva de la obra en relación con su concepción y/o ejecución que es el fruto del esfuerzo creador del autor<sup>472</sup>.

### *1.2.B) Las facultades exclusivas de reproducción y transformación.*

#### *B.1) El concepto de reproducción del programa.*

El otro punto de fricción entre los ordenamientos nacionales europeos se situaba en torno a la interpretación del alcance del derecho de reproducción y al derecho de transformación.

En relación al primero, las diferencias residían en si cualquier utilización de un programa de ordenador debía considerarse una acto de reproducción sujeto a la autorización del titular o no.

---

<sup>470</sup> J.A., GÓMEZ SEGADÉ, “Spanish Software Law” en *Tecnología y Derecho: Estudios jurídicos del Prof. Dr. h.c. José Antonio Gómez Segade recopilados con ocasión de la conmemoración de los XXV años de cátedra*, Madrid, Marcial Pons, 2001 pág. 848. También, J.P., APARICIO VAQUERO, J., DELGADO ECHEVERRÍA, “Comentarios al Libro Primero, Título VII” ..., *op.cit.*, pág. 1252.

<sup>471</sup> Como ha señalado algún autor, antes de la Directiva de 1991 la jurisprudencia sobre derechos de autor en general, tanto por parte del Tribunal Supremo como de tribunales inferiores, era muy escasa en España. (Vid. J.A., GÓMEZ SEGADÉ, “Spanish Software Law”..., *op.cit.*, pág. 849).

<sup>472</sup> R. BERCOVITZ RODRIGUEZ-CANO, (ed.), *Comentarios a la Ley de Propiedad Intelectual*, Madrid, Tecnos, pág. 208-211; J., MASSAGUER FUENTES, “La adaptación de Ley de Propiedad Intelectual a la Directiva CEE relativa a la protección jurídica de los programas de ordenador”, RDM, nº 199-200, 1991, pág. 39 (46). Con matices, acoge también esta concepción J., DELGADO ECHEVERRÍA para los programas de ordenador en J., DELGADO ECHEVERRÍA, “Comentario a los artículos 95 a 100 de la LPI”, en R. BERCOVITZ RODRÍGUEZ-CANO (ed.), *Comentarios a la Ley de Propiedad Intelectual*, Madrid, Tecnos, 1989, pág. 1352-1354.

Como ha reflejado DELGADO ECHEVERRÍA<sup>473</sup>, la doctrina europea trató de subsumir la necesidad de controlar el uso en el derecho de reproducción. El argumento empleado partió de argumentos técnicos más que jurídicos, ya que si bien es cierto que toda utilización de un programa conlleva que se realice una copia de las instrucciones en la memoria interna de la máquina, que dicha copia constituya una reproducción en el sentido del derecho de autor, y que de serlo, constituya una infracción, es más discutible y puede variar de un ordenamiento a otro. Dependerá de si es exigida cierta duración o perdurabilidad, de qué consistencia física se requiera (las instrucciones de un programa son impulsos eléctricos); de si existe perceptibilidad y comprensión por seres humanos, y en última instancia de qué se entienda por programa como objeto protegido.

En Alemania, la cuestión se planteó ante los tribunales en el caso *Betriebssystem*<sup>474</sup>. El Tribunal Supremo alemán declaró que si al ejecutar el programa, la copia de la obra, en el sentido técnico del término, no resulta en una multiplicación de posibilidades (simultáneas) del uso del programa, los intereses del autor no se ven comprometidos y, por lo tanto, ese acto no ha de considerarse incluido en el derecho de reproducción. En consecuencia, la mera ejecución de un programa no se considera reproducción; a lo sumo, la visualización de los resultados del programa en pantalla podrían constituir un acto de explotación entendido como un acto de comunicación, como un recital o una representación artística; sin embargo estos actos no infringirían ninguna facultad del titular al no realizarse para el público. En tanto que la ejecución del programa fuese considerada una mera utilización más que un uso en el sentido de los derechos de autor, no era necesaria la autorización del titular<sup>475</sup>.

La posición adoptada por el BGH coincidía con la opinión de la doctrina mayoritaria, ilustrada por HABERSTUMPF, para quien la realización de una copia del programa en el disco duro sin autorización constituía una infracción del derecho de reproducción, pero hacer funcionar el programa en un dispositivo es un acto libre

---

<sup>473</sup> J.P., APARICIO VAQUERO, J. DELGADO ECHEVERRÍA, "Comentarios al Libro Primero, Título VII" ..., *op.cit.*, pág. 1344.

<sup>474</sup> BGH sentencia de 04 de octubre de 1990, I ZR 139/89 "Betriebssystem", GRUR 1991, pág. 449. (versión en inglés en IIC 22, (1991), pág. 723).

<sup>475</sup> W., BLOCHER, M., WALTER, "Article 4, Restricted Acts", en M., WALTER, S. VON, LEWINSKI, (eds.) *European copyright law: a commentary* (rev. ed. and translation of the German ed. 2001), Oxford, Oxford Uni. Press, 2010, pág. 127.

porque de otro modo habría que recabar autorización para usar el programa, lo cual solo tiene cabida en el derecho de patente<sup>476</sup>.

Dando un paso más, el Tribunal Supremo, en el caso *Holzhandelsprogramm*, matizó el criterio establecido en *Betriebssystem*, a través de un razonamiento que podría resumirse como que un acto de reproducción a efectos del derecho de autor tiene lugar cuando la reproducción es necesaria para poder disfrutar de la obra<sup>477</sup>.

En el Reino Unido, el primer escollo con que tropezó el concepto de reproducción sobre un programa de ordenador fue terminológico. Si bien la Copyright Act de 1956 hablaba de la facultad exclusiva del titular para controlar cualquier reproducción del obra en cualquier forma material (“*reproducing the work in any material form*”) la modificación llevada a cabo en 1988, identificó esta facultad con la copia de la obra. El artículo 17, en su apartado segundo, declaraba que en relación con las obras literarias, copia significa la reproducción de la obra en cualquier forma material<sup>478</sup>. A continuación, matiza qué ha de entenderse por formas materiales al añadir que la reproducción de la obra incluye su almacenamiento a través de cualquier medio electrónico. Finalmente, el apartado sexto del artículo detalla las copias que han de considerarse incluidas bajo la facultad exclusiva del titular: Toda clase de copia, con independencia de su carácter temporal o permanente<sup>479</sup>.

Por lo tanto, el concepto de reproducción de un programa establecido por la ley inglesa de 1988 es mucho más amplio que el de la ley alemana, ya que el criterio de la necesidad no ha de tomarse en cuenta para determinar si la utilización del programa constituye una reproducción de la obra sujeta a autorización. Esto se traduce en que por

---

<sup>476</sup> Vid. H., HABERSTUMPF, “Der urheberrechtlich Schutz...”, *op.cit.*, pág. 59-63. Ha de tenerse en cuenta que según lo prescrito en el artículo 16 UrhG, basta la realización de una copia para que haya reproducción, y así lo había interpretado la jurisprudencia del Tribunal Supremo (BGH 24 de junio de 1955, BGHZ, 18, pág. 46)

<sup>477</sup> BGH sentencia de 20 de enero de 1994, Caso No. I ZR 267/91 *Holzhandesprogramm*, GRUR 1994, pág. 363 (365). Este caso, a pesar de ser de fecha posterior a la implementación de la Directiva sobre protección de programas de ordenador de 1991, fue juzgado acorde a la normativa anterior. Un comentario sobre el mismo puede verse en M., LEHMANN, “Federal Supreme Court (Bundesgerichtshof) 20.01.1994 Case No. I ZR 267/91 “Lumberyard Program””, IIC 1995, pág. 720.

<sup>478</sup> Artículo 17, apartado 2 de la Copyright Act de 1956: “(...) *copying in relation to a literary work means reproducing the work in any material form*”.

<sup>479</sup> Artículo 17, apartado 6 de la Copyright Act 1956: “(...) *copying in relation to any description of work includes the making of copies which are transient or are incidental to some other use of the work*”. Vid. H., LADDIE, P. PRESCOTT, M., VITORIA, *The Modern Law of Copyright and Designs*, Vol. II, Londres, 4ªed., Butterworths, 2011, pág. 1592; C.J., MILLARD, “United Kingdom” en H., JONGEN (coord.), *Copyright Software Protection in the EC*, Computer Law Series, Deventer, Kluwer Law and Taxation Publ., 1994, pág. 224.

ejemplo, la realización de una copia temporal en la memoria interna del dispositivo que ejecute el programa no sería un acto de reproducción sujeto a autorización para el derecho alemán, pero sí para el derecho inglés.

El derecho francés, partidario de una concepción amplia del derecho de reproducción, lo definía en el artículo 28 de la Ley de Propiedad Intelectual de 1957<sup>480</sup>, como “la fijación material de la obra por cualquier procedimiento que permita comunicarla al público de una manera indirecta”<sup>481</sup>.

Como indica RIVERO HERNÁNDEZ, la identificación de la reproducción con la comunicación “indirecta” al público, es una medida adoptada por el legislador francés para distinguir este acto de la representación, que es una forma de comunicación “directa” al público<sup>482</sup>. Sin embargo, en relación al significado de reproducción para los programas de ordenador, autores como LUCAS habían afirmado ya en 1975 la dificultad de considerar la existencia de reproducción en esta clase de obras, porque las instrucciones se manifiestan de diferente forma a lo largo del proceso en que son tratadas para producir una comunicación indirecta<sup>483</sup>.

El legislador francés zanjó la cuestión en la Ley de Propiedad Intelectual de 1985<sup>484</sup> al establecer directamente en el artículo 47 que “toda utilización” de un programa no expresamente autorizada por el autor o sus causahabientes, será castigado por la ley<sup>485</sup>.

Consagró así una facultad exclusiva sobre el uso o dicho de otro modo, un derecho exclusivo de uso. De manera que, para el derecho de autor francés, cualquier tipo de utilización del programa estaba sujeta a autorización del autor, ampliando el ámbito del *ius prohibendi*.

---

<sup>480</sup> Loi n°57-298 du 11 mars 1957 sur la propriété littéraire et artistique, JORF 14/31957, pág. 2723.

<sup>481</sup> Artículo 28 de la Ley de Propiedad Intelectual de 1957: “*La reproduction consiste dans la fixation matérielle de l'oeuvre par tous procédés qui permettent de la communiquer au public d'une manière indirecte.*”

<sup>482</sup> F., RIVERO HERNÁNDEZ, “Comentario al artículo 18 del Libro Primero, Título II” en R. BERCOVITZ RODRIGUEZ-CANO, R. (coord.), *Comentarios...*, *op.cit.*, pág. 284. Para un comentario detallado sobre el derecho de reproducción en Francia vid. C. COLOMBET, *Propriété littéraire et...*, *op.cit.*, pág. 161.

<sup>483</sup> A., LUCAS, *La Protection des Créations Industrielles Abstraites, Centre D'Études Internationales de la Propriété Industrielle*, Paris, Libraries Techniques (LITEC), 1975, pág. 186

<sup>484</sup> Loi n° 85-660, du 3 juillet 1985 relative aux droits d'auteur et aux droits des artistes-interprètes, des producteurs de phonogrammes et de vidéogrammes et des entreprises de communication audiovisuelle, JORF 4/7/1985 pág. 7495.

<sup>485</sup> Artículo 47 de la Ley de Propiedad Intelectual de 1985: “*(...) toute reproduction autre que l'établissement d'une copie de sauvegarde par l'utilisateur ainsi que toute utilisation d'un logiciel non expressément autorisée par l'auteur ou ses ayants droit, est passible des sanctions prévues par ladite loi.*”

La declaración del artículo 47 suscitó cuestiones para los usuarios como: ¿en qué circunstancias puede un usuario realizar una copia de seguridad sin infringir el derecho de reproducción sobre el programa?, o ¿cabe la posibilidad de realizar una copia de seguridad a través de mecanismos de copia disponibles en el mercado sin infringir el derecho de autor sobre el programa? El tribunal de apelaciones de París tuvo la ocasión de interpretar esta norma en 1988<sup>486</sup>, y declaró que la prohibición de reproducción, referida a la copia de seguridad, no puede interpretarse como una prohibición para duplicar la obra.

Por otro lado, la consagración de este derecho de uso en el derecho francés, obliga a traer a colación las Disposiciones Tipo de la OMPI de 1978. Si bien como instrumento normativo internacional fracasaron en su intento de establecer un sistema de protección *sui generis*, el texto de su artículo 5º guarda una clara similitud con el artículo 47 de la Ley de Propiedad Intelectual francesa.

El artículo 5º de las Disposiciones Tipo regulaba los derechos del propietario. En su apartado vi), establecía la facultad de prohibir a terceros la utilización del programa para:

1. Copiarlo por cualquier medio o en cualquier forma,
2. Elaborar otro programa o una descripción del mismo, idéntico o sustancialmente similar.
3. Controlar el funcionamiento de una máquina capaz de procesar información o almacenar una copia del programa en ella.

Es en el contexto de un derecho *sui generis* que las Disposiciones consagran un derecho exclusivo de uso. Este derecho creado *ex-novo*, se había considerado indispensable a lo largo de las sesiones de los comités para garantizar la protección de los intereses de los creadores. En un contexto social donde casi todos los programas de ordenador se elaboran previo contrato y a medida, con los enormes costes económicos y técnicos que conllevaba, y donde la copia no autorizada y la piratería se situaban como grandes amenazas para una industria naciente, el control sobre el uso sobre el programa se había erigido como la opción más adecuada para equilibrar los intereses entre propietarios y usuarios, pero en el contexto de un sistema de protección *sui generis*, no en el derecho de autor.

---

<sup>486</sup> Asunto La Commande Electronique, sentencia de 20 de octubre de 1988, JCP ed. G. 1989, II, pág. 2188.

Tanto si el legislador francés se inspiró en este proyecto de derecho sui generis configurado en las Disposiciones Tipo para crear un derecho de uso *ad hoc* en el derecho de autor, como si decidió añadirlo al contenido del derecho de reproducción, no deja de ser una anomalía dentro del derecho de autor<sup>487</sup>.

En línea con el derecho francés, la Ley de Propiedad Intelectual española de 1987, en el apartado 1 del artículo 99, consagró también el derecho de uso sobre el programa. El artículo define la “cesión del derecho de uso” como “aquel acto en virtud del cual el titular del derecho de explotación de un programa de ordenador autoriza a otro a utilizar el programa, conservando el cedente la propiedad del mismo”. Como han puesto de manifiesto varios autores, para realizar una cesión, es presupuesto necesario la atribución legal de un “derecho de uso” al autor del programa, y la existencia de este derecho sea como parte de la facultad exclusiva de reproducción o como facultad creada *ex-novo* por el legislador, es muy discutible<sup>488</sup>.

La labor de legislador produjo de este modo una especie de distorsión del concepto de reproducción tanto en Francia como en España, posiblemente motivado por la necesidad económica de controlar el uso del programa ante la amenaza de la piratería.

## B.2) El concepto de transformación (*adaptation right*).

El derecho exclusivo de transformación de la obra (internacionalmente denominado *adaptation right*), como ya se adelantó en el primer capítulo, es una de las materias más complejas del derecho de autor.

Desde el punto de vista del derecho de autor con carácter general, lo primero que cabe destacar es que esta facultad exclusiva no ha sido objeto de armonización a nivel comunitario. Un sólido argumento a su favor es el estrecho vínculo del derecho de transformación con los derechos morales y, en segundo lugar, la falta de armonización

---

<sup>487</sup> Como puso de relieve el Tribunal Supremo alemán en el caso *Betriebssystem* (GRUR 1991, p. 449). En igual sentido, para RIVERO HERNÁNDEZ, “no cabe un *ius utendi* o un *ius possidendi* al modo de los derechos reales ordinarios, en consideración a la naturaleza espiritual, inmaterial, del objeto de la propiedad intelectual”. Vid. F., RIVERO HERNÁNDEZ, F., “Comentario al artículo 21” en R., BERCOVITZ RODRÍGUEZ-CANO, R. (ed.) *Comentarios a la Ley de Propiedad Intelectual*, Madrid, 2ª ed., Tecnos, 1997, pág. 299.

<sup>488</sup> J., DELGADO ECHEVERRÍA, en “Comentario al Libro Primero, Título VII”, en R., BERCOVITZ RODRÍGUEZ-CANO, (ed.) *Comentarios a la Ley de Propiedad Intelectual*, Madrid, 2ª ed., Tecnos, 1997, pág. 1466; J. MASSAGUER FUENTES, “La adaptación de...”, *op.cit.*, pág. 51; J.A., GÓMEZ SEGADE, *Tecnología y Derecho...*, *op.cit.*, pág. 873.

de qué constituye el objeto de protección del derecho de autor. Porque, cuáles sean los elementos que determinan una adaptación o transformación de una obra y cuáles una obra nueva e independiente, son cuestiones ligadas al concepto de originalidad que se aplique a la obra.

En segundo lugar, en el entorno digital, la codificación o decodificación de una obra digital no altera su composición o su forma de expresión, por lo tanto, el resultado de la conversión es una reproducción, no una adaptación, traducción o transformación de la obra<sup>489</sup>.

Como han destacado entre otros HUGENHOLTZ, los ordenamientos jurídicos de todos los Estados Miembros de la Unión Europea reconocen el derecho de transformación, pero su emplazamiento varía. Para algunos ordenamientos, esta facultad es considerada como un derecho exclusivo único, y para otros, el derecho de transformación es parte del derecho de reproducción<sup>490</sup>.

Dicho esto, no deja de ser llamativo que la única directiva que ha abordado directamente el derecho de transformación haya sido la Directiva sobre protección de los programas de ordenador.

En consecuencia, además de lo dicho sobre el requisito de originalidad, nada añade a la cuestión de la interoperabilidad entrar a examinar la concepción del derecho de transformación antes de la Directiva.

Ahora bien, sí conviene tener presente la diversidad terminológica, para no perderse al tratar los actos de transformación del programa en la Directiva y en su transposición a los ordenamientos nacionales: El derecho alemán dedica el artículo 23 de la UrhG a la adaptación y otra clase de alteraciones (“Bearbeitungen und andere Umgestaltungen”); el derecho inglés, emplea el término “adaptation” en general en el artículo 21 de la CPDA, para a continuación individualizar cada clase de transformación según la clase de obra a que se refiera; el derecho francés, sin embargo, utiliza los términos “traduction”, “adaptation”, “transformation” y “arrangement” en el artículo 122-4 de la LPI; y en paridad con el derecho francés se sitúa el derecho español, cuyo artículo 21 habla de transformación y adaptación, aunque luego añade una amplia

---

<sup>489</sup> En P.B., HUGENHOLTZ, “Adapting Copyright to the Information Superhighway”, paper based on studies prepared for the European Commission (DG XIII and DG XV) in 1994 and 1995, sec. 2.1

<sup>490</sup> M.VAN, EECHOUD, P.B., HUGENHOLT, S.VAN, GOMPEL *et al*, *Harmonizing European Copyright Law...op.cit.*, pág. 83.

cláusula de cierre, más en sintonía con la técnica empleada por el legislador alemán: “cualquier otra modificación en su forma de la que se derive una obra diferente”.

## **2. La protección de los programas de ordenador en la Directiva 91/250/CEE de 14 de mayo sobre protección jurídica de los programas de ordenador.**

### *2.1. La finalidad protectora de la Directiva 91/250/CEE.*

Aunque en el marco jurídico internacional los programas de ordenador son protegidos como obras literarias en el marco del Convenio de Berna, lo cual, en el ámbito del derecho de autor de tradición jurídica continental supone tener en cuenta conceptos como autor, derechos morales, creatividad, y sobretodo la protección integral del autor, sin embargo, las justificaciones para proteger los programas de ordenador tienen mejor cabida al amparo de la tradición jurídica del *common-law* como incentivo a la creación y a la innovación, desarrollo y crecimiento económico<sup>491</sup>. El derecho de autor continental justifica su finalidad de protección en consideraciones de carácter más personalista y de derecho natural<sup>492</sup>. No obstante, en el caso de los programas de ordenador, a pesar de la ficción jurídica de considerarlos obras literarias, los objetivos de protección en que se basa, no sólo la directiva, sino incluso las leyes de derecho de autor anteriores de algunos Estados miembro, no son los de carácter personalista<sup>493</sup>.

Baste observar el tratamiento dado por los tribunales en Alemania, en Francia, el Reino Unido o incluso España, y la difícil aplicación del criterio de originalidad al programa. Optar por criterios objetivos, como Alemania, más propios del derecho de patente, llevaría a dejar sin protección a casi todos los programas. Pero cuando los tribunales intentaron apoyarse en criterios más subjetivos, como en Francia buscando elementos estéticos en las decisiones que toma el programador, o la doctrina de “the-

---

<sup>491</sup> En Estados Unidos la última ratio del copyright se ha descrito en los siguientes términos: “the court must be faithful to the statutory language and mindful of the ultimate goal of copyright law – the advancement of public welfare - and Congress’ chosen method of achieving this goal – private reward to the individual actor” (Lotus Dev. Corp. v. Paperback Software Intern., 740 F. Supp. 37 (D.C. D. Massachusetts 1990), pág. 53).

<sup>492</sup> Vid., entre otros, GARZA BARBOSA, R., *Derechos de autor y derechos conexos : marco jurídico internacional : aspectos filosóficos, sustantivos y de litigio internacional*, México, Editorial Porrúa, 2009; STROWEL, A., *Droit d’Auteur et Copyright. Divergences et convergences. Etude de droit comparé*, E.E. Bruylant, Bruselas, 1993.

<sup>493</sup> Existe una corriente doctrinal contraria que defiende los fundamentos de carácter personalista para los programas de ordenador: K.E., HIMMA, “The Justification of Intellectual Property: Contemporary Philosophical Disputes”, *Journal of the American Society for Information Science and Technology*, vol. 59, is. 7, mayo 2008, pág. 1143; M., SIPONEN, “A Justification for Software Rights”, *ACM SIGCAS Computers and Society: selection of best papers 2004-2006*, vol. 36, is. 3, septiembre 2006, pág. 11.



sweat-of-the-broe” en Reino Unido, llevaban a proteger prácticamente cualquier programa que no fuera una copia exacta de la anterior<sup>494</sup>.

Como ha indicado DELGADO PORRAS, “la protección del programa de ordenador a través del derecho de autor, permite preguntarse lícitamente qué razón existe en acudir a una normativa cuya finalidad protectora (la del autor-creador) no sólo no se pretende, sino que hay que traicionarla en sus líneas maestras para obtener la tutela adecuada<sup>495</sup>”.

El Considerando Tercero de la Directiva dispone: “Los programas de ordenador están desempeñando un papel de creciente importancia en una amplia gama de sectores y, en consecuencia, cabe considerar la tecnología informática como de capital importancia para el desarrollo industrial de la Comunidad”.

Por si quedase alguna duda, el Considerando Segundo habla de la elevada “inversión de recursos humanos, técnicos y financieros” que “exige el desarrollo de programas”. Otra pista que proporcionan los Considerandos de la Directiva es que resulta casi imposible encontrar una mención directa al “autor” de la obra (se habla del “sujeto de protección”), y sí es fácil encontrar alusiones al “funcionamiento del mercado interior” (Considerandos Cuarto y Quinto).

Por lo tanto, este estudio parte de que la protección jurídica de los programas de ordenador como obras literarias en el Derecho de autor comunitario, se fundamenta en razones de carácter utilitario y no en la protección integral de su autor, teniendo como objetivo la recuperación de la inversión realizada y en última instancia, el fomento del bienestar del consumidor mediante la divulgación de obras<sup>496</sup>.

## 2.2. *El debate sobre la interoperabilidad en la elaboración de la Directiva.*

El Libro Verde publicado en 1988 por la Comisión Europea planteaba dos cuestiones esenciales en relación a la garantía de la interoperabilidad informática. En

---

<sup>494</sup> Vid. supra, capítulo 3.1.2.A)

<sup>495</sup> Vid. A., DELGADO PORRAS, A., “Del optimismo a la perplejidad: Reflexiones impertinentes de un jurista sobre la protección de los programas de cómputo por el Derecho de Autor”, en A., DELGADO PORRAS, (ed.), *Derecho de autor y derechos afines al de autor: Recopilación de artículos*, Madrid, Instituto de Autor, 2007, pág. 67.

<sup>496</sup> J.H., REICHMAN, “Of Green Tulips and Legal Kudzu: Repacking Rights in Subpatentable Innovation”, *Vand. L. Rev.* 53, 2000, pág. 1744 (1749); N., SHEMTOV, *The Legal Regulation of Decompilation of Computer Programs: Excessive, Unjustified and in Need of Reform*, 2012 (no publicada), pág. 29, disponible en: <https://qmro.qmul.ac.uk/xmlui/handle/123456789/3132> (última consulta 1 de junio, 2013).

primer lugar, la necesidad de examinar la naturaleza de la ingeniería inversa en relación a esta clase de obras. La segunda pregunta planteaba si era adecuado proteger con el derecho de autor el acceso a los protocolos y a la información de las interfaces de los programas de ordenador<sup>497</sup>.

Al tiempo que el Libro Verde planteaba estas cuestiones, las recientes normas aprobadas en Alemania, Reino Unido, Francia y España no contenían mención alguna sobre la ingeniería inversa de programas de ordenador<sup>498</sup>.

En la negociación de la Directiva, el debate acerca de la permisibilidad de la ingeniería inversa tuvo enorme protagonismo, porque de ello dependía la mayor o menor dificultad para lograr la interoperabilidad del software. El origen del debate fue el texto propuesto inicialmente para el artículo 4, donde se establecían las facultades exclusivas del titular del programa. La extraordinaria precisión del lenguaje empleado para describir las facultades exclusivas en este borrador inicial, en concreto la de reproducción, condujo a una situación delicada. El artículo 4 dejaba en manos del titular el control sobre toda clase de reproducción, tanto se tratase de una reproducción meramente técnica, temporal, con capacidad de duplicar el programa, etc., sin excepción de ninguna clase para el usuario de la obra<sup>499</sup>. De esta manera, cualquier acto de ingeniería inversa sobre el programa quedaba sujeto a la autorización del titular del programa.

La amplitud de las facultades concedidas al titular en el artículo 4, unido a la inexistencia de excepción o limitación legal que autorizase procesos de ingeniería inversa, provocó el enfrentamiento entre dos grupos de presión:

---

<sup>497</sup> Libro Verde sobre los derechos de autor y el desafío tecnológico, problemas de derechos de autor que requieren una acción inmediata COM(88) 172, junio de 1988, pág. 19.

<sup>498</sup> Tampoco la jurisprudencia norteamericana había tenido ocasión de pronunciarse sobre esta cuestión. Vid. *supra*, capítulo 2.2.

<sup>499</sup> La propuesta inicial establecía entre las facultades exclusivas del artículo 4: “a) *la reproducción total o parcial de un programa de ordenador por cualquier medio y bajo cualquier forma. En la medida en que comporten la reproducción total o parcial del programa, se considerarán actos prohibidos la carga, visualización, ejecución, transmisión o almacenamiento del mismo; b) la adaptación de un programa de ordenador;*”. La versión original en lengua inglesa: “a) *the reproduction of a computer program by any means and in any form, in part or in a whole. In so far as they necessitate a reproduction of the program in part or in a whole, loading, viewing, running, transmission or storage of the computer program shall be considered restricted acts; b) the adaptation of a computer program;*”. Propuesta de Directiva del Consejo sobre protección jurídica de programas informáticos, COM(88) 816 final – SYN 183 (Presentada por la Comisión el 5 de enero de 1989), DOE 89/C 91/05. Comentarios detallados sobre la historia legislativa de la Directiva pueden verse en: T., VINJE, “The History of the EC Software Directive” en M., LEHMANN, C., TAPPER (eds.), *A Handbook of Software European Law...*, *op.cit.*, pág. 49 y ss.; J., BAND, M., KATOH, *Interfaces on Trial...*, *op.cit.*, pág. 230.

Por un lado, la pequeña y mediana industria informática y programadores emprendedores, representados por el Comité Europeo para los Sistemas Interoperables (ECIS), que veían en la redacción de estos dos artículos una gran amenaza a la innovación del sector, capaz de eliminar el desarrollo de nuevos programas interoperables<sup>500</sup>.

Por otro lado, las grandes empresas informáticas, que a finales de los años ochenta representaban el 80% del sector, representadas por el Grupo Europeo de Acción del Software (SAGE), defendían la no permisibilidad de actos de ingeniería inversa del programa<sup>501</sup>. En su opinión, permitir actos de esa clase no era necesario por dos razones: La elaboración de programas interoperables no necesita acceder a la información de las interfaces del código porque toda la información necesaria se encontraba en los manuales de los programas. En segundo lugar, la permisibilidad de ingeniería inversa supondría la pérdida de todo el esfuerzo y la inversión económica realizada por el propietario del programa original.

El debate llegó al Parlamento Europeo, a quien le correspondió jugar un papel decisivo en un enfrentamiento de vital importancia para la industria informática europea. El Parlamento envió la Propuesta de Directiva a tres comités de quienes solicitó opinión: El Comité de Asuntos Jurídicos, el Comité de Energía, Investigación y Tecnología, y el Comité de Asuntos Económicos y Monetarios y Política Industrial.

De los tres informes emitidos hacia finales de 1989, los de los dos primeros comités fueron los más influyentes, defendiendo cada uno, una de las posiciones enfrentadas.

El Informe del Comité de Asuntos Jurídicos recomendaba no modificar el articulado de la Propuesta de Directiva sobre la ingeniería inversa, porque consideraba apresurado pronunciarse sobre esa cuestión. La justificación de su postura era que sería

---

<sup>500</sup> Además, si la Directiva establecía que los actos necesarios para realizar actos de ingeniería inversa sobre el programa pertenecían al *ius prohibendi* del titular, la ECIS también temía, con razón, que no podrían defenderse de los actos contrarios a la libre competencia descritos en los Arts. 85 y 86 de Tratado de la Comunidad. Vid. T., VINJE, "Compliance with Article 85 in Software Licensing", ECLR 4, 1992, pág. 165.

<sup>501</sup> La SAGE fue apoyada por la USTR (United States Trade Representative), quien trató esta cuestión como asunto de política internacional, posiblemente motivado por el miedo a que si la Directiva Europea optase por condonar la ingeniería inversa, los competidores japoneses pudiesen piratear el software desarrollado en Estados Unidos, pero distribuido y comercializado en territorio de la Unión Europea. Su informe sobre la cuestión concluyó que permitir la ingeniería inversa de un programa debería ser considerada como una actividad anticompetitiva. Vid. J. BAND, M. KATOH, *Interfaces on Trial...*, *op.cit.*, pág. 233.

más apropiado esperar el tratamiento de la cuestión por los tribunales de países europeos líderes en el sector informático, antes de establecer una norma europea<sup>502</sup>. En relación a las interfaces, sin embargo, recomendaba modificar el texto del artículo 1.3 de la Propuesta Inicial para dejar claro que las ideas y principios inspiradores del programa no estaban protegidos por derecho de autor<sup>503</sup>.

El Informe de la Comisión de Energía, en cambio, se mostraba favorable a la modificación del texto de la Propuesta en relación a la ingeniería inversa, argumentando que ni en Estados Unidos, ni en Japón la descompilación del programa estaba prohibida, de modo que prohibir este proceso de ingeniería inversa en Europa situaría a las empresas europeas en una clara desventaja competitiva. La permisibilidad de la descompilación no permitiría a un competidor escribir un programa que infringiese el derecho de autor del programa descompilado, porque existían métodos desarrollados por la industria, como el denominado “clean room” (sala blanca), que dejaban claro que no se copiaba la forma de expresión a través de dicho procedimiento<sup>504</sup>.

El tercer Informe, emitido por el Comité de Política Industrial, defendía la postura de la SAGE, y no sólo recomendaba rechazar cualquier excepción legal que permitiese realizar ingeniería inversa, sino que declaraba que su permisividad supondría dejar de proteger legalmente a los programas<sup>505</sup>. Según el Informe, la solución al mantenimiento de la información sobre las interfaces en secreto por parte de quienes las han creado, y que tienen una posición dominante en el mercado informático, se encuentra únicamente en el Derecho de Defensa de la Competencia<sup>506</sup>.

A raíz de las conclusiones de los informes, en especial el elaborado por la Comisión de Energía, el Consejo de Ministros devolvió la Propuesta de Directiva a la Comisión, indicando que fueran reexaminadas con detalle las cuestiones sobre las interfaces y la ingeniería inversa.

Tras la devolución de la Propuesta Inicial, la Comisión centró sus esfuerzos en establecer el marco de una excepción a los derechos exclusivos del titular que

---

<sup>502</sup> Eur. Parl. Doc. 136. 025 (rev.) DOC EN/PR/78020 (1989) pág. 16.

<sup>503</sup> El texto del artículo 1.3 de la Propuesta Inicial declaraba sobre las interfaces: “Cuando las ideas y principios inspiradores del programa estén constituidos por especificaciones de interfaces, no podrán reconocerse derechos de autor sobre dichos principios e ideas”.

<sup>504</sup> Eur. Parl. Doc. 134.239 (rev) (Oct. 1989) pág. 24 y 25.

<sup>505</sup> “legal protection for computer programs would virtually cease to exist”, en Eur. Parl. Doc 134. 405, DOC EN/PA/74903 (Dec. 1989) pág. 5, apartado 10.

<sup>506</sup> *Ibidem*, apartado 7.

permitiesen la descompilación necesaria para alcanzar la interoperabilidad pero que no amparase procedimientos de descompilación cuyo propósito fuese obtener mayor acceso a las ideas inspiradoras del programa<sup>507</sup>. A tal fin, el grupo de trabajo de la Comisión valoró tres posibilidades:

- 1º Renunciar a establecer una norma reguladora de la descompilación, y dejar la cuestión a la decisión discrecional de los tribunales cuando se diese el caso.
- 2º Introducir un cláusula de fair-dealing siguiendo el modelo británico.
- 3º Introducir una cláusula que expresamente codificase en qué condiciones el titular habría de soportar la realización de actos de ingeniería inversa sobre el programa.

Las dos primeras opciones fueron rechazadas. En medio de tan álgido debate entre los defensores de no permitir acto de ingeniería inversa alguno y los valedores de que la ingeniería inversa para acceder a la información interoperable era fundamental, dejar la cuestión en manos de la discrecionalidad de los tribunales nacionales no tenía sentido. Los esfuerzos de la Comisión y el Parlamento por encontrar una solución salomónica hubiesen quedado en nada si finalmente no se regulaba la descompilación.

En cuanto a la segunda posibilidad, la introducción de una cláusula de fair-dealing para regular la permisibilidad de los actos de ingeniería inversa, tampoco fructificó. En la Exposición de Motivos del primer borrador de la Directiva se afirmaba que los Estados Miembro deberían seguir aplicando a los programas de ordenador las excepciones con que contaban sus ordenamientos nacionales para las obras literarias. En concreto, el concepto de “fair-dealing” anglosajón no era excluido en relación a la venta de segunda mano de programas de ordenador. La Exposición de Motivos sugería que, en el caso de programas licenciados, las disposiciones contractuales deberían contemplar las excepciones a los derechos exclusivos<sup>508</sup>.

Si bien la introducción de una cláusula de fair-dealing presentaba considerables ventajas en una materia como la ingeniería inversa, ya que sería susceptible de adaptarse a nuevas circunstancias que surgiesen en el futuro, sobre todo en un ámbito como el informático, cuyo ritmo de innovación es mucho más rápido que el de otra clase de tecnologías de la información, no resultaría nada sencillo encajar una cláusula de esta

---

<sup>507</sup> T., VINJE, “The Legislative History of the EC Software Directive”, en M., LEHMANN, C., TAPPER (eds.), *A Handbook of...*, *op.cit.*, pág. 59.

<sup>508</sup> Apartado 5.1. de la Exposición de Motivos original – Disposiciones Particulares.

naturaleza en los sistemas nacionales de la mayoría de los Estados Miembro, de tradición continental<sup>509</sup>. Por lo tanto, la tercera opción fue la adoptada finalmente.

La Propuesta que se presentó en Julio de 1990 al Parlamento había tratado de identificar en qué circunstancias los tribunales nacionales habría de entender justa y razonable la realización de actos de ingeniería inversa, codificándolas en unas normas que articulaban un sistema de controles y ponderaciones de dichas circunstancias. El texto resultante fue bastante complejo.

Por un lado, la nueva propuesta contenía una nueva redacción del artículo 5. En ella se añadían dos apartados. El apartado primero regulaba una excepción a favor de un procedimiento de ingeniería inversa, conocido técnicamente como análisis de caja negra, y el segundo establecía una excepción que permitía la descompilación del programa por razones de interoperabilidad.

La votación tras la primera lectura en el Parlamento europeo marcó el hito más dramático en la historia de la Directiva<sup>510</sup>. Por primera vez, uno de los organismos legislativos de la Comunidad Europea apoyaba el establecimiento de una norma que permitía la descompilación a efectos de interoperabilidad oficialmente y la aprobaba con una mayoría abrumadora.

Sin embargo, como ha manifestado BLOCHER<sup>511</sup>, aunque el Parlamento modificó varias veces el texto inicial, la Propuesta Final de la Directiva aprobada por el Consejo en mayo de 1991 no siguió dos indicaciones esenciales que el Parlamento había sugerido:

Por un lado, mientras que la propuesta del Parlamento hablaba de “*any act essential to ensure maintenance of the program and the creation or operation of interoperable programs*”, la Propuesta Final aprobada por el Consejo detallaba que la excepción se limitaba a “*la reproducción del código la traducción de su forma (...)*”, que fuesen “*indispensables para alcanzar la creación, mantenimiento or funcionamiento de un programa interoperable de creación independiente*”.

---

<sup>509</sup> B., CZARNOTA, J., HART, *Legal Protection of Computer Programs in Europe. A Guide to the EC Directive*, London, Butterworths, 1991, pág. 75

<sup>510</sup> T., VINJE, “The Legislative History of the EC Software Directive”, en M., LEHMANN, C., TAPPER (eds.), *A Handbook of...*, *op.cit.*, pág. 68, citando prensa.

<sup>511</sup> W. BLOCHER, M. WALTER, “Computer Program Directive”, en M., WALTER, S.VON, LEWINSKI, (eds.) *European copyright law...*, *op.cit.*, pág. 164, nota 5.6.7.

La segunda gran modificación la realizó el Consejo, que reescribió las condiciones para permitir la descompilación. En opinión del Consejo, el texto propuesto por la Comisión podía ser interpretado de forma muy amplia, por lo tanto, cambió las palabras “*indispensable para conseguir la creación, mantenimiento o funcionamiento de un programa interoperable creado de forma independiente*” por el enunciado “*indispensable para obtener la información necesaria para alcanzar la interoperabilidad de un programa creado de forma independiente con otros programas*”.

### 2.3. La bandera blanca: La excepción de la descompilación del artículo 6.

La historia legislativa del artículo 6 de la Directiva, deja claro que el epicentro del debate sobre la permisibilidad de la descompilación informática reside en la clásica tensión entre el derecho de exclusiva del autor y la salvaguarda del interés general de la comunidad<sup>512</sup>. Esta lucha de intereses está recogida en los considerandos de Directiva, donde, por un lado el considerando Tercero declara que la tecnología informática es de capital importancia para el desarrollo de la Comunidad, y el Décimo Cuarto recoge la obligatoria necesidad de mantener un equilibrio entre los derechos exclusivos del autor y los intereses del usuario<sup>513</sup>.

A pesar de tratarse de una dicotomía clásica, no deja de constituir un auténtico reto para el Derecho el establecimiento de un marco regulatorio en que el derecho de autor sobre el programa de ordenador no se erija como elemento inhibitorio de la innovación. La intención de la Comisión al adoptar el derecho de autor como sistema de protección del programa y como garante de la interoperabilidad era que a través de la concesión de derechos exclusivos se estimulase a los fabricantes de programas a invertir sus recursos intelectuales y financieros y con ello promover el progreso tecnológico en el interés público<sup>514</sup>.

---

<sup>512</sup> Como ha reflejado Anne Wilkinson, esta tensión no es nueva para el Derecho de Autor: “attempts to balance or arbitrate between the public interest and the author’s right are reported at least as early as St. Columba’s time”. En A., WILKINSON, “Software Protection: Policies in the EC” en M., LEHMANN, C., TAPPER (eds.), *A Handbook of...*, op.cit., pág. 38.

<sup>513</sup> Considerando Decimocuarto: “(...) no debe impedirse a la persona facultada para utilizar el programa de ordenador que realice los actos necesarios para observar, estudiar o verificar su funcionamiento siempre que dichos actos no supongan infracción de los derechos de autor sobre el programa”.

<sup>514</sup> Vid. “Conclusiones decididas por la Comisión” en “Propuesta de Directiva del Consejo sobre protección jurídica de programas informáticos”, DOCE Núm. C91/04 de 12 de abril de 1989, pág. 16.

Por otro lado, tanto el progreso tecnológico como el bienestar público son objeto de garantía a través del derecho de defensa de la competencia. En este punto la Directiva es precisa en sus considerandos al reconocer la relación entre las normas comunitarias de la competencia y el derecho de autor como garantes de la interoperabilidad informática<sup>515</sup>.

Esta relación está más clara en el texto del Borrador de la Directiva que el propia Directiva. En la última página del Borrador inicial, bajo el título “Conclusiones de la Comisión”, la Comisión declaró que en determinadas circunstancias hacer valer el derecho de autor sobre algunos elementos del programa, los cuales son necesarios para que otros fabricantes puedan escribir programas compatibles, podría convertirse en un abuso de derecho. Es más, afirma que el acceso a la información no está sujeto a derecho de autor<sup>516</sup>.

Finalmente se optó por regular el tema de la interoperabilidad a través de una excepción al derecho autor, configurada en el artículo 6 de la Directiva, para que funcionase a modo de válvula de seguridad en el caso de que la información interoperable no fuese accesible para el innovador-competidor. Se podría decir entonces que el artículo 6 es esa garantía de acceso a la información necesaria para seguir innovando<sup>517</sup>.

La descompilación que excepciona el artículo 6 es, como se dijo en el capítulo primero, una tarea lenta, costosa y no siempre eficaz. Permitir la libre descompilación de un programa para lograr la interoperabilidad con el programa descompilado podría elevar los niveles de competitividad, con el consiguiente descenso de los precios y el impulso a la innovación en las nuevas compañías, con lo cual el consumidor vería

---

<sup>515</sup> Vid. Considerando Decimoséptimo: “Las disposiciones de la presente Directiva se entienden sin perjuicio de la aplicación de las normas sobre la competencia en virtud de los artículos 81 y 82 del Tratado, si un proveedor en posición dominante se niega a hacer disponible una información necesaria para la interoperabilidad definida en la presente Directiva”.

<sup>516</sup> Vid. “Conclusiones decididas por la Comisión”, en “Propuesta de Directiva del Consejo sobre protección jurídica de programas informáticos”, DOCE Núm. C91/04 de 12 de abril de 1989, pág. 16.

<sup>517</sup> B., CZARNOTA, J., HART, J., *Legal Protection of...*, *op.cit.*, pág. 76; T., DREIER, O., MEYER-VAN RAAY, “Computerrecht”, en M., SCHULTE, R., SCHRÖDER (eds.), *Handbuch des Technikrechts*, Berlin, 2ª ed., Springer, 2011, pág. 825. Compartimos la crítica de la doctrina española a que en este artículo de la Directiva se utiliza una regulación que se pretende mantener dentro del “Derecho de Autor” al servicio de unas finalidades de estandarización en la tecnología informática y de las telecomunicaciones y de regulación de la competencia, como instrumento alternativo de otras medidas, en especial el derecho de la competencia. Vid. J.P., APARICIO-VAQUERO, J., DELGADO ECHEVERRÍA, “Comentarios al Libro Primero, Título VII” ..., *op.cit.*, pág. 1383.



considerablemente incrementado su número de opciones en el mercado<sup>518</sup>. Pero la excepción del artículo 6 ni autoriza la libre descompilación, ni configura un derecho a descompilar, ni cumple con el objetivo pretendido por la Comisión.

El siguiente análisis pretende demostrar que la excepción al derecho de autor que configura el artículo 6 no es la herramienta más apropiada para garantizar una materia de interés público como la interoperabilidad. Esta excepción protege el acceso a la información, legitimando un abuso de derecho del titular del programa y obligando indirectamente a recurrir al derecho de defensa de la competencia para garantizar el acceso a la información interoperable. Es más, esta excepción tal como finalmente fue configurada no tiene como objeto principal garantizar la interoperabilidad sino que su objeto es la autorización de un proceso de ingeniería inversa (la descompilación) que conlleva la realización de actos de reproducción, exclusivos del titular del derecho de autor.

#### 2.4. Análisis del artículo 6 de la Directiva.

El artículo 6 de la Directiva de Software además de ser la disposición más extensa que contiene esta norma comunitaria es una de las más controvertidas<sup>519</sup>. Formada por tres apartados, que a su vez contienen tres sub apartados en los cuales se establecen, en primer lugar, los actos para los que no es necesaria la autorización del titular, a continuación los requisitos que han de cumplirse, y en tercer lugar, el uso que pueda darse a la información obtenida. La relevancia práctica de este precepto ha sido además, bastante baja, posiblemente porque algunos fabricantes de software, como IBM, divulgan la información interoperable de sus programas de manera voluntaria y debido al continuo auge del movimiento del software libre, donde la información interoperable está a disposición del usuario del programa.

Para una correcta interpretación de este precepto, habrá que remitirse al principio establecido en el artículo 1.2 de la Directiva: “las ideas y los principios en los que se base cualquiera de los elementos de un programa de ordenador, *incluidos los que*

---

<sup>518</sup> C., MENINSKY, “Locked Out: New Hazards of Reverse Engineering”, J. Marshall J Computer & Info. L. 21, (2003), pág. 591 (592).

<sup>519</sup> T., VINJE, “El desarrollo de productos interoperables bajo la directiva sobre software de la CE”, RGD, año L, núm. 589-90, octubre-noviembre, 1993, pág. 10133 (10134); A., WIEBE, Comentario al art. 69e “Dekompilierung”, en G., SPINDLER, F., SCHUSTER, F., *Recht der elektronischen Medien*, München, 2ª ed., Beck, 2011, nota 1.

*sirven de fundamento a sus interfaces, no estarán protegidos mediante derechos de autor con arreglo a la presente Directiva”.*

#### *A) Principios que rigen la excepción de la descompilación.*

Antes de entrar a examinar e interpretar el contenido del artículo 6, es necesario establecer los principios que rigen la excepción.

##### *1. Principio del ius prohibendi.*

En primer lugar, el derecho de autor sobre un programa conlleva el derecho a prohibir (*ius prohibendi*) a terceros la copia y el uso del contenido del programa<sup>520</sup>. No es menester que el derecho de autor se ocupe de las garantías de acceso a la obra, aun cuando el titular no tenga derecho a impedir su uso, sino que más bien, son las excepciones o limitaciones las que restringen la capacidad del titular de la obra para prohibir determinados usos<sup>521</sup>.

La técnica de ingeniería inversa que constituye la descompilación conlleva la realización de actos tanto de reproducción como de transformación del programa. Por lo tanto, al tratarse de actos reservados al titular del derecho de autor, la descompilación de un programa protegido por este derecho de exclusiva, requiere autorización de su titular.

##### *2. Principio de restrictividad.*

Las excepciones, como mayor límite al derecho de exclusiva del autor<sup>522</sup>, son un elemento normativo que otorgan el derecho a usar la obra a un tercero sin tener que pedir autorización al titular y sin tener que remunerar tal uso<sup>523</sup>. Son herramientas del legislador claves para asegurar el equilibrio de intereses entre los titulares de los derechos de autor y el interés público<sup>524</sup>.

---

<sup>520</sup> J.H., SPOOR, et al., *Auteursrecht: auteursrecht, naburige rechten en databankenrecht*, Deventer, Kluwer, 2005, pág. 2; W., BLOCHER, “Art. 6 Decompilation”, en M., WALTER, S. VON, LEWINSKI (eds.), *European copyright law...*, *op.cit.*, pág. 168, nota 5.6.20.

<sup>521</sup> L., GUIBAULT, *Copyright limitations and...*, *op.cit.*, pág. 109

<sup>522</sup> C., GEIGER, “Fundamental Rights, a Safeguard for the Coherence of Intellectual Property Law?”, IIC 35(3), 2004, pág. 268 (270).

<sup>523</sup> L., GUIBAULT, *Copyright limitations and...*, *op.cit.*, pág. 21.

<sup>524</sup> A., KUR, “Of Oceans, Islands and Inland Water – How much Room for Exception and Limitations under the three step test”, MPI Res. Paper Series, nº 8 – 04, 2008, pág. 1; A., LUCAS, *Traité de la*

Por otra parte, la excepción que contiene el artículo 6 de la Directiva es clara muestra de un sistema cerrado de limitaciones como es el derecho de autor continental: es analítica y muy detallada<sup>525</sup>. Ello debiera traducirse en que el ámbito de las prerrogativas atribuidas al autor es abierto y es el quien juez debe interpretar la excepción de manera que no perjudique los intereses de los creadores, haciendo además una lectura restrictiva de la excepción que no da lugar jamás a que se origine un derecho por parte del usuario<sup>526</sup>.

Las condiciones que impone la excepción del artículo 6 solo permiten acudir a ella si la información interoperable no ha sido puesta previamente, y de manera fácil y rápida, a disposición del usuario legítimo del programa, y siempre y cuando tal información “sea indispensable para la consecución de la interoperabilidad de un programa creado de forma independiente”. Es decir, sólo ha de acudirse a esta opción como última ratio y en ningún otro caso que no sean los enunciados en la norma.

Este grado de restrictividad de la excepción llevado a sus últimas consecuencias supondría que el adquirente legítimo de un programa no pudiera usar, reproducir, ni adaptar el programa a sus necesidades, ni investigar sus principios operativos para la interoperabilidad o para versiones sucesivas, sin la expresa autorización del titular, y posiblemente sujeta a contraprestación. En este escenario el mercado del software y por extensión de la informática se hunde: nadie va a querer comprar o alquilar programas de ordenador si su usuario legítimo (sea por título legal o contractual) no puede realizar actos de estricta explotación utilitaria y técnica<sup>527</sup>.

El derecho de autor lleva en su seno sus propias limitaciones porque, en su calidad de derecho especial, deben inscribirse armoniosamente dentro del sistema jurídico, es decir, también debe plegarse a la lógica de los Principios Generales del

---

*propriété littéraire et artistique*, Paris, 3<sup>o</sup> ed., LexisNexis Litec, 2006, pág. 33; A., LEPAGE, “Panorama general de las excepciones y limitaciones al derecho de autor en el entorno digital”, E-Boletín de derecho de autor, Enero-Marzo, 2003, pág. 5, disponible en: <http://unesdoc.unesco.org/images/0013/001396/139696s.pdf>, (última consulta 1 de junio, 2013).

<sup>525</sup> Sobre los sistemas cerrados y sistemas abiertos de limitaciones y excepciones vid. P., SIRINELLI, “Excepciones y limitaciones al derecho de autor y los derechos conexos”, en Taller sobre cuestiones de aplicación del Tratado de la OMPI sobre derecho de autor y Tratado de la OMPI sobre interpretación o ejecución y fonogramas, Ginebra 6 y 7 de diciembre de 1999, Doc. WCT-WPPT/IMP/1, 1999, pág. 19 a 22; disponible en: [http://www.wipo.int/meetings/es/doc\\_details.jsp?doc\\_id=1266](http://www.wipo.int/meetings/es/doc_details.jsp?doc_id=1266) (última consulta 1 de junio, 2013)

<sup>526</sup> P., SIRINELLI, “Excepciones y limitaciones...”, *op.cit.*, pág. 22.

<sup>527</sup> J.M., RODRIGUEZ TAPIA, “Los derechos del usuario legítimo de bases de datos y de programas de ordenador”, en R., ANTEQUERA PARILLI, C., ROGEL VIDE (coord.), *Los límites del derecho de autor*, Reus, Madrid, 2006, pág. 213.

Derecho<sup>528</sup>, de manera que la excepción debiera garantizar el justo equilibrio entre protección y usos libres<sup>529</sup>.

### 3. Principio de imperatividad.

De acuerdo con el segundo párrafo del artículo 8 de la Directiva (antes art. 9.1), cualquier disposición contractual que sea contraria a lo dispuesto en el artículo 6, será nula y sin valor ni efecto alguno.

Este reconocimiento del carácter obligatorio de la excepción tiene su origen en el objetivo de la Comisión y del Consejo, de limitar la libertad contractual de las partes a fin de no vaciar de contenido el uso libre reconocido en las excepciones<sup>530</sup>.

Por lo tanto, si un usuario legítimo cumple con todas las condiciones y requisitos establecidas en el artículo 6, a priori, tal como indica la Directiva, el uso libre que permite la excepción no puede limitarse ni modificarse a través de disposiciones contractuales.

Sin embargo, al implementación de esta norma en los ordenamientos nacionales de los Estados Miembros, resultó en un abanico de distintos grados de “imperatividad”<sup>531</sup>.

#### B) Actos permitidos por la excepción.

El objetivo de la excepción del artículo 6 es evitar que el titular del programa, a través de las facultades exclusivas concedidas por el artículo 4 de la Directiva, impida la creación e innovación de programas que sean técnicamente compatibles con su programa o con otros<sup>532</sup>. Por esa razón, al inicio de la norma, quedan establecidos los actos que el titular del programa no puede impedir al usuario que cumpla con las condiciones de la excepción. Literalmente la norma habla de: “la reproducción del

---

<sup>528</sup> Ibidem, pág. 6.

<sup>529</sup> C., GEIGER, “Exclusivity in Copyright ...”, *op.cit.*, pág. 517.

<sup>530</sup> J.F., VERSTRYNGE, “Protecting Intellectual Property...”, *op.cit.*, pág. 6.

<sup>531</sup> L., GUIBAULT, *Copyright limitations and...*, *op.cit.*, pág. 219. Por ejemplo, la Ley de Derecho de Autor alemana en su artículo 69(g)(2), transcribe casi literalmente lo establecido en el artículo 8 de la Directiva y explícitamente prohíbe las cláusulas contractuales que no permitan la descompilación: El art. 69(g)(2) UhrG dispone: “*Vertragliche Bestimmungen, die in Widerspruch zu[r] ... [Ausnahmeregelung für Dekompilierung in §69(e)] stehen, sind nichtig*”. En igual sentido, el artículo L122-6-1 del Código de Propiedad Intelectual francés, en su apartado V, párrafo segundo: “*Toute stipulation contraire aux dispositions prévues aux II, III et IV du présent article est nulle et non avenue.*”

<sup>532</sup> L., BENTLY, L., “Computer Programs Directive” ..., *op.cit.*, art. 6, pág. 230.

código” y “la traducción de su forma”, “a efectos del artículo 4, apartado 1, letras a) y b)”.

La primera cuestión que llama la atención, como ha manifestado BLOCHER<sup>533</sup>, es que el legislador no utilice la expresión “programa de ordenador” sino que se refiera al “código” y su “forma”. Hemos de remitirnos al artículo 1.1 de la Directiva, donde se define el objeto de protección, para encontrar una explicación. Si bien en el ámbito científico, los términos programa de ordenador, software y código son intercambiables porque se refieren a la misma realidad, desde el punto de vista de la Directiva son realidades distintas. El programa de ordenador como objeto de protección, tal como declara el artículo 1.1 comprende, además del código, la documentación preparatoria.

Una primera conclusión es que el artículo 6 sólo permite actos de reproducción (y traducción) de un elemento del programa, el código.

Desde el punto de quien ejecuta la descompilación, que la norma se refiera al código en lugar de al programa no tiene mayor importancia, porque el proceso de descompilación siempre va a realizarse sobre un código binario para tratar de extraer del mismo algo similar al código fuente<sup>534</sup>, y precisamente porque no se dispone de la documentación preparatoria.

Sin embargo, desde el punto de vista jurídico, el hecho de que los actos autorizados se limiten solo a esta parte del programa conllevan la prohibición implícita de intentar reconstruir cualquier tipo de documentación preparatoria del programa, aunque la misma derive de las ideas y principios que haya podido deducir del código a través del proceso de descompilación o través del proceso de ingeniería inversa que autoriza la excepción del artículo 5.3 de la Directiva.

La segunda cuestión en relación a los actos permitidos radica en determinar si la excepción autoriza toda clase de reproducciones y transformaciones del código o no. Para analizar esta cuestión hay que remitirse al artículo 4.1, letras a) y b) de la Directiva.

---

<sup>533</sup> W., BLOCHER, “Art. 6 Decompilation”, ..., *op.cit.*, pág. 171, nota 5.6.27.

<sup>534</sup> No debe perderse de vista que en la creación del programa de ordenador, al compilar el código fuente, toda la información escrita en lenguaje de programación de alto nivel se pierde por completo, y el código objeto (binario) resultante no contiene la información necesaria para reconstruir algo similar a lo que podría ser el código fuente primario. Vid. E., EILAM, *Reversing...*, *op.cit.*, pág. 458.

La facultad exclusiva de reproducción que establece la Directiva en la letra a) del citado artículo, abarca toda reproducción, total o parcial, por cualquier medio y bajo cualquier forma, ya sea permanente o transitoria, incluidos los actos de carga, presentación, ejecución, transmisión o almacenamiento del programa.

Comparada con el Convenio de Berna, la definición de reproducción del artículo 4, letra a), es más amplia que contenida en su artículo 9.1, que define la reproducción de las obras literarias y artísticas como la realizada “por cualquier procedimiento y bajo cualquier forma”. Una reproducción constituye una copia de la obra, su multiplicación, sin tener en cuenta el método que se haya empleado para ello ni el medio en que se fije la copia<sup>535</sup>.

Comparada con la definición de reproducción contenida en el artículo 2 de la Directiva 2001/29, el concepto de reproducción de la letra a) del artículo 4 es menos amplio, porque no incluye la reproducción indirecta de la obra, que sí menciona expresamente el artículo 2 de la Directiva 2001/29<sup>536</sup>.

El “Memorandum Explicativo” de la Directiva dejaba claro que el legislador comunitario tenía en mente una concepción específica de reproducción cuando redactó el artículo 4, que se apartaba de la tradición europea continental, especialmente de la tradición alemana. La justificación para elegir una noción tan amplia del derecho de reproducción se encuentra en el citado documento, y consiste en la necesidad de incluir cualesquiera actos que potencialmente pudieran perjudicar los intereses del autor, por ejemplo, la carga, visualización, ejecución, transmisión, o almacenamiento del programa, los cuales solo tienen lugar a través de la reproducción del programa<sup>537</sup>. Así

---

<sup>535</sup> T., DREIER, Comentario a la Convención..., *op.cit.*, pág. 41.

<sup>536</sup> Artículo 2: “Los Estados miembros establecerán el derecho exclusivo a autorizar o prohibir la reproducción directa o indirecta, provisional o permanente, por cualquier medio y en cualquier forma, de la totalidad o parte (...)”. El concepto de reproducción indirecta fue introducido por el Tratado WPPT de 1996, en su artículo 7: “Los artistas intérpretes o ejecutantes gozarán del derecho exclusivo de autorizar la reproducción directa o indirecta de sus interpretaciones o ejecuciones fijadas en fonogramas, por cualquier procedimiento o bajo cualquier forma”. Un comentario puede verse en P. BRISON, “WPPT, Art. 7” en T., DREIER, P.B., HUGENHOLTZ (eds.), *Concise European Copyright...*, *op.cit.*, pág. 177.

<sup>537</sup> El texto original decía: “all acts which could prejudice the interests of the author, for example, the loading, viewing, running, transmission, or storage of the program, can be processed only by way of reproducing the program”. Vid. Explanatory Memorandum, Proposal for a Council Directive on the legal protection of computer programs, COM (88) 816 final-SYN 183, 89/C 91/5, Parte Segunda, nota en el Art 4(1)a.

pues, el propósito de la Directiva es que cualquier acto de mera utilización del programa quede sujeto a autorización del titular<sup>538</sup>.

Volviendo al artículo 6.1, éste hace referencia al derecho de reproducción del código sin mayores matizaciones, por tanto, si el usuario cumple con las condiciones establecidas por la excepción en los apartados siguientes, parece que podrá realizar cualquier acto de mera utilización del código en el sentido indicado por el artículo 4 la Directiva.

Al referirse al derecho de transformación, el texto del artículo 6.1 solo menciona la “traducción”. El artículo 4.1 letra b), donde se regula esta facultad patrimonial sobre el programa, habla de “traducción, adaptación, arreglo y cualquier otra transformación (...)”, así pues, siguiendo el principio de interpretación restrictiva de la excepción, el usuario que cumpla con las condiciones de la norma, sólo está autorizado a traducir la forma del código.

Comparada con cualquier otra obra literaria, esta restricción sobre el derecho de transformación de la obra puede parecer excesiva. La causa de esta restricción reside en la naturaleza funcional del programa. Dicho de otro modo, cualquier persona que adquiera legítimamente una obra literaria como un ensayo, una novela, etc., podrá, por ejemplo, traducirla a otro idioma, recortar sus páginas, o hacer un resumen y nada de eso infringirá los derechos de autor sobre la obra si no lo divulga. En cambio, a la vista del texto del artículo 6.1 de la Directiva, resulta imposible adaptar o transformar en sentido alguno el código de un programa, si no se dispone de permiso del titular.

Por otra parte, la excepción autoriza a realizar un proceso específico de ingeniería inversa como es la descompilación, a través del cual, el programador intentará generar una representación del programa en lenguaje de alto-nivel desde los códigos binarios más básicos. El resultado no va a ser por tanto una traducción de la obra tal como puede pensarse en una obra literaria tradicional, sino una versión funcionalmente idéntica (en el mejor de los casos) del programa original<sup>539</sup>. Como se adelantó en el apartado B.2 de este capítulo, en el entorno digital, la codificación o descodificación de una obra digital no altera su composición o su forma de expresión,

---

<sup>538</sup> W., BLOCHER, “Art. 4 Restricted Acts”, en M., WALTER, S. VON., LEWINSKI, (eds.) *European copyright law...*, *op.cit.*, nota 5.4.19.

<sup>539</sup> Vid. E., EILAM, *Reversing...*, *op.cit.*, pág. 458.

por lo tanto, el resultado de la conversión es una reproducción, no una traducción de la obra *stricto sensu*<sup>540</sup>.

Esto significa que en realidad, el artículo 6.1 solo autoriza actos de reproducción sobre el programa, y aunque a priori parece que cualesquiera clase de actos de reproducción, el segundo párrafo de la norma, en conjunción con las letras b) y c), van a restringir de una forma cuasi leonina las partes del código sobre los que la persona autorizada puede llevar a cabo cualquier acto de reproducción. De tal modo que las reproducciones que permite el artículo 6 se refieren a cualquier acto que tenga por finalidad la obtención de la información indispensable y necesaria para la interoperabilidad, de manera que los actos de reproducción se limitan a las partes del programa original que resulten necesarias para conseguir la interoperabilidad, es decir, las interfaces. Esta autorización queda sin efecto cuando la información interoperable haya sido puesta previamente, y de manera fácil y rápida, a disposición de la persona autorizada.

En un proceso de descompilación, el primer paso pasa por ejecutar el programa en un ordenador y cargarlo en el descompilador. Pero el software no puede ser ejecutado por la máquina sin realizar en primer lugar una copia temporal de todo el programa en su memoria RAM (random access memory o memoria de acceso aleatorio)<sup>541</sup>. Este acto de reproducción temporal, siguiendo lo prescrito por el artículo 4 de la Directiva, se encuentra dentro de las facultades exclusivas del titular, y va más allá de lo permitido por la excepción, ya que el programador necesita reproducir todo el código binario del programa, no solo las partes que contienen la información interoperable. Técnicamente es imposible reproducir solo las partes en que se encuentra la información necesaria, porque ello supondría además que el programador ya conoce cuáles son esas partes antes de descompilar el programa.

A través de este juego de condiciones, la excepción permite al titular del derecho de autor impedir al usuario legítimo la realización de copias temporales indispensables

---

<sup>540</sup> P.B., HUGENHOLTZ, “Adapting Copyright to ...”, *op.cit.*, sec. 2.1.

<sup>541</sup> L., GUIBAULT, O., VAN DAALEN, *Unravelling the Myth ...*, *op.cit.*, pág. 100; H., HABERSTUMPF “Der urheberrechtliche Schutz...”, *op.cit.* pág. 136; J.H., SPOOR et al., *Auteursrecht: auteursrecht, naburige...*, *op.cit.*, pág. 595



para descompilar el programa, y el acceso al código fuente del programa, donde se encuentra la información interoperable<sup>542</sup>.

Aquí reside la clave de la sobreprotección que la Directiva concede al titular del programa, y lo que es más llamativo, a través de una excepción legal: al permitir al titular controlar la reproducción temporal del programa necesaria para la descompilación, el derecho de autor limita el acceso del usuario legítimo a la parte más valiosa del programa, sus ideas y principios o “know-how”<sup>543</sup>. Con lo cual, a través de la configuración de una excepción legal, la Directiva establece una protección que si bien formalmente parece acorde con los principios de derecho de autor, en el fondo no lo es. En realidad, establece un régimen de protección *sui generis* que restringe el acceso a los principios e ideas del programa, al modo de la protección reforzada del secreto empresarial<sup>544</sup>.

*C) El objeto: La información interoperable de las interfaces.*

*C.1.) Las interfaces del programa ante la dicotomía idea-expresión.*

La información necesaria para alcanzar la interoperabilidad se encuentra en las interfaces del programa. Este elemento no aparece mencionado explícitamente en el artículo 6. Es necesario acudir al Considerando Décimo, donde la Directiva define a las interfaces como pieza que permite interoperar<sup>545</sup>. En el considerando siguiente, el

---

<sup>542</sup> Vid., entre otros, T., VINJE, “The Legislative History of the EC Software Directive”, en M., LEHMANN, C., TAPPER (eds.), *A Handbook of...*, *op.cit.*, p. 49; T., DREIER, “Rechtsschutz von Computerprogrammen: Die Richtlinie des Rates der EG von 14. Mai 1991”, CR 7, 1991, pág. 577 (581); D., SCHULTE “Der Referententwurf eines Zweiten Gesetzes zur Änderung des Urheberrechtsgesetzes”, CR 8, (1992), pág. 648 (653).

<sup>543</sup> A.VAN, ROOIJEN, *The software interface...*, *op.cit.*, pág. 69; P., SAMUELSON *et al*, “A Manifesto...”, *op.cit.*, pág. 2392; J., BARTMANN, *Grenzen der Monopolisierung...*, *op.cit.*, pág. 123; J.H., SPOOR, “Copyright Protection and Reverse Engineering of Software: Implementation and Effects of the EC Directive”, U. Dayton Law Review 19, 1994, pág. 1063 (1082).

<sup>544</sup> J., BARTMANN, *Grenzen der Monopolisierung...*, *op.cit.*, pág 123; P., SAMUELSON, “Comparing U.S. and E.C. Copyright Protection for Computer Programs: Are They More Different Than They Seem?”, *Journal of Law and Commerce* 13, 1994, pág. 279; P., SAMUELSON, “Are patents on ...”, *op.cit.*, pág. 1959; M., GÓMEZ PERALS, *La cesión de uso de los programas de ordenador*, Madrid, Colex, 1999, pág. 279.

<sup>545</sup> Considerando Décimo: “Las partes del programa que establecen (dicha) interconexión e interacción entre los elementos de software y hardware suelen denominarse interfaces”.

Undécimo<sup>546</sup>, declara que las interfaces están excluidas del régimen de protección diseñado por la Directiva. Sin embargo, esta afirmación no encuentra correspondencia explícita en ninguno de los preceptos sustantivos de la norma comunitaria<sup>547</sup>.

Para determinar el objeto de protección del programa es necesario acudir al artículo 1.2 de la Directiva. Este precepto, recogido en dos tratados internacionales como son el ADPIC y el WCT<sup>548</sup>, reitera el principio general del derecho de autor, según el cual se protege la expresión de la obra pero no las ideas en que dicha expresión se ha basado, porque estas últimas han de permanecer libres para que otros puedan usarlas y seguir innovando<sup>549</sup>.

El artículo dispone: “La protección prevista en la presente Directiva se aplicará a cualquier forma de expresión de un programa de ordenador. Las ideas y principios en los que se base cualquiera de los elementos de un programa de ordenador, incluidos los que sirven de fundamento a sus interfaces, no estarán protegidos mediante derechos de autor con arreglo a la presente Directiva.”

El legislador comunitario aplica este principio general por igual a los programas de ordenador como a sus interfaces, proporcionando muy poca ayuda a la hora de

---

<sup>546</sup> Considerando Undécimo: “Para evitar cualquier duda, debe establecerse claramente que solo se protege la expresión del programa de ordenador y que las ideas y principios implícitos en los elementos del programa, incluidas sus interfaces, no pueden acogerse a la protección de los derechos de autor (...)”.

<sup>547</sup> La Comisión originalmente consideró excluir expresamente de protección, total o parcial, las interfaces del programa. Vid. Libro Verde sobre los derechos de autor y el desafío tecnológico, problemas de derechos de autor que requieren una acción inmediata, Doc. COM(88) 172 final, de 23 de agosto de 1988, p. 170. Vid. también, M., LEHMANN, “Standardization and the EC Directive of 14 May 1991 on the Legal Protection of Computer Programs”, in W.F., KORTHALS, et al., *Information Law ...*, op.cit., pág. 363.

<sup>548</sup> Art. 9.2 Acuerdo sobre los ADPIC y Art. 2 Tratado WCT: “La protección del derecho de autor abarcará las expresiones pero no las ideas, procedimientos, métodos de operación o conceptos matemáticos en sí”. En opinión de algunos autores, en el ámbito de lo que se ha denominado como “derecho de autor tecnológico”, este principio podría ilustrar, la función del derecho de autor de impulsar el equilibrio económico, de manera que el incentivo a la creación de obras no restrinja excesivamente el dominio público. Vid. G., GHIDINI, *Intellectual Property and Competition Law. The Innovation Nexus*, Cheltenham, Edward Elgar, 2006, págs. 52-77, en especial págs. 63 a 65; L., KURTZ, “Speaking to the Ghost: Idea and Expression in Copyright”, U. MIAMI L.REV 47, 1993, pág. 1221 (1224); P., SAMUELSON, “A Square Peg in a Round Hole? Copyright Protection for Computer Programs”, en B., SHERMAN, L., WISEMAN, *Copyright and the Challenge of the New*, Alphen aan den Rijn, Kluwer Law International, 2012, pág. 265-270.

<sup>549</sup> Vid. Memoria explicativa Parte 2 Artículo 1 núm. 2 y 3, Propuesta de modificación Art. 1(2), y propuesta de modificación del Parlamento Europeo nº 3 sobre el Art. 1(3) de la Propuesta para una Directiva Comunitaria de protección de los programas de ordenador; H. HABERSTUMPF., “Der urheberrechtliche Schutz...”, op.cit., pág. 100 ff., nos. 54-62; W., BLOCHER, M., WALTER, “Article 1, Object of Protection”, en M., WALTER, S. VON, LEWINSKI (eds.), *European copyright law...*, op.cit., pág. 103.

determinar si las interfaces del programa forman parte del objeto de protección<sup>550</sup>. Este hecho, permite elaborar dos planteamientos de partida:

Por un lado, si los programas de ordenador, tal cual indica el artículo 1.1 de la Directiva, son objeto de protección del derecho autor como obras literarias, entonces, el ámbito primario de esta protección se circunscribe al código escrito del programa, más que a su funcionalidad o comportamiento<sup>551</sup>. Consecuentemente, al ser las interfaces del programa parte de este código escrito podrían constituir también parte del objeto de protección.

Ahora bien, las interfaces de un programa, al contrario del comportamiento de éste, son unas informaciones muy concretas y detalladas que pueden ser representadas por escrito. La esencia de la propia interfaz informática radica en su concreción, gracias a la cual la máquina (el ordenador) es capaz de ejecutar la interconexión y el intercambio de información entre distintos programas o entre un programa y un componente físico del mismo. Por esta razón, son expresiones concretas que forman parte del código del programa. Desde esta perspectiva, son por tanto, expresión del programa y no ideas.

Si no es posible descartar la protección de las interfaces a través de la interpretación literal de las normas de la Directiva, ¿existe alguna otra vía que permita ayudar al usuario de un programa acceder a la información interoperable contenida en este elemento del programa, haciendo uso de la excepción contenida en el artículo 6 de la Directiva, sin riesgo de estar cometiendo una infracción sobre el derecho del titular?

Desde la aprobación de la Directiva 91/250/CEE hemos esperado más de veinte años a que el Tribunal de Justicia de la Unión Europea se pronunciase respecto de si las interfaces son o no “*forma de expresión de un programa de ordenador*”.

Dos sentencias, la primera en 2010 y la segunda en 2012, empiezan a despejar importantes dudas a este respecto, aunque como veremos, abren una nueva serie de interrogantes que no dejan cerrada del todo la cuestión.

---

<sup>550</sup> En igual sentido se han pronunciado D., BAINBRIDGE, *Intellectual Property, ...*, *op.cit.*, pág. 245, para quien la Directiva falla al no establecer el objeto protegido de las interfaces de manera explícita. Vid también, A.VAN, ROOIJEN, *The software interface...*, *op.cit.*, pág. 74; L. BENTLY, “Computer Programs Directive”, en T., DREIER, P.B., HUGENHOLTZ (eds.), *Concise European Copyright...*, *op.cit.*, pág. 216.

<sup>551</sup> P., SAMUELSON *et al*, “A Manifesto...”, *op.cit.*, pág. 2360.

Antes de entrar a comentar estas dos importantes resoluciones, conviene hacer referencia a la jurisprudencia francesa y a la británica, que en alguna ocasión anterior ya tuvieron ocasión de pronunciarse sobre la cuestión de la protección de las interfaces y, como veremos, las líneas argumentales de sus tribunales nacionales están en consonancia con el criterio establecido por el Tribunal de Justicia de la Unión Europea.

Los tribunales franceses fueron quizás los primeros en analizar la concreta cuestión de si las interfaces de un programa formaban parte del objeto protegido por el derecho de autor. En la sentencia de la Corte de Apelación de París, en el caso *Nomai c. Iomega Corporation*<sup>552</sup>.

La compañía norteamericana IOMEGA Corporation había desarrollado y distribuido en el mercado un disco ZIP<sup>553</sup> junto con un programa de ordenador llamado "Oscar 100". La compañía Nomai, especializada en la producción de discos duros y discos de almacenamiento informático denominados "Super Floppy XHD", sin pedir permiso a IOMEGA, adaptó la secuencia de datos original (denominada "pista Z") y el programa "Oscar 100" del disco zip, creando un disco compatible con el programa de su competidor. IOMEGA demandó a Nomai alegando la infracción de sus derechos de autor sobre los datos de las secuencias originales de datos y la copia ilícita de su programa "Oscar". También alegó actos de competencia desleal por parte de Nomai por publicitar su nuevo disco "Super Floppy XHD" como "*100% compatible IOMEGA ZIP*".

El Tribunal sostuvo que, si bien el programa "Óscar 100" de IOMEGA podría estar protegido por derechos de autor, la "pista Z" del disco ZIP, al ser "una secuencia de datos que permite a la memoria comunicarse con un lector"<sup>554</sup> era una interfaz no protegida por derechos de autor. Aplicando el artículo 1.2 de la Directiva, el Tribunal concluyó que la Nomai tenía derecho a reproducir la secuencia de datos, porque era esencial para producir disquetes compatibles<sup>555</sup>.

---

<sup>552</sup> Asunto *Nomai v. IOMEGA*, sentencia de 12 de diciembre de 1997, Corte de Apelaciones de París (Cour d'Appel), section B, Chambre 14.

<sup>553</sup> ZIP es un concepto utilizado en software para referirse a formatos de almacenamiento de datos.

<sup>554</sup> Asunto *Nomai v. IOMEGA*, op.cit., pág. 2: "la séquence originale de données utilisées à des fins d'identification."

<sup>555</sup> Asunto *Nomai v. IOMEGA*, op.cit. pág. 3: "*les idées et les principes qui sont à la base des interfaces d'un programme d'ordinateur ne sont pas protégés par le droit d'auteur*". (Las ideas y los principios que son el fundamento de las interfaces de un programa de ordenador no están protegidas por el derecho de autor).

En relación a la jurisprudencia británica, se trata de la sentencia de 30 de julio de 2004 del Tribunal Supremo inglés<sup>556</sup>, un caso donde se decidía sobre la infracción de copyright sobre un programa de ordenador por imitar el demandado la funcionalidad y las operaciones del programa del demandante. Para crear el nuevo programa, el demandado había estudiado al detalle las utilidades del anterior, observando cómo se comportaba, qué funciones y operaciones podía ejecutar, cómo era tratada, almacenada y recuperada la información, sin acceder al código fuente del programa en ningún momento.

La cuestión que fue planteada el Tribunal Supremo fue si otras partes del programa, como la apariencia en pantalla (el denominado “look and feel”), que no constituían específicamente código, constituían elementos del programa protegidos por copyright. El Tribunal aplicó directamente el artículo 1.2 de la Directiva 91/250/CEE, apoyándose en los Considerandos Décimotercero y Décimocuarto – correspondientes a los actuales Décimo y Undécimo- para concluir que las interfaces del programa no forman parte del objeto de protección<sup>557</sup>.

Este criterio fue seguido y matizado poco después, en 2007, del siguiente modo<sup>558</sup>:

1º La elaboración de un programa de ordenador que imita el funcionamiento de otro anterior pero que no copia su código o su infografía, no constituye infracción.

2º Las ideas implícitas del programa no están protegidas por copyright.

Esta misma línea argumental se mantiene en la sentencia del TJUE de fecha 22 de diciembre de 2010 en el asunto C-393/09<sup>559</sup>, al interpretar el artículo 1.2 de la

---

<sup>556</sup> *Navitaire v. EasyJet Airline Co.*, sentencia del Tribunal Supremo (Chancery Division) de 30 de julio de 2004 [2004] EWHC 1725 (Ch.), vid. también en RPC 2006, pág. 111.

<sup>557</sup> Pár. 130: “If it is the policy of the Software Directive to exclude both computer languages and the underlying ideas of the interfaces from protection, then it should not be possible to circumvent these exclusions by seeking identify some overall function or functions that it is the sole purpose of the interface to invoke and relying on those instead. As a matter of policy also, it seems to me that to permit the “business logic” of a program to attract protection through the literary copyright afforded to the program itself is an unjustifiable extension of copyright protection into a field where I am far from satisfied that it is appropriate.”

<sup>558</sup> *Nova Productions Ltd. v. Mazooma Games Ltd and others*, sentencia del 14 de marzo de 2007, Tribunal de Apelaciones, [2007] EWCA Civ. 219; vid. también en: RPC 2007, pág. 25. De especial interés son los aps. 32, 35 y 45 de la sentencia.

Directiva. Según el TJUE, la expresión de un programa es cualquier forma del mismo que permita reproducirlo en diferentes lenguajes informáticos, tales como el código fuente y el código objeto<sup>560</sup>. El Tribunal comunitario se sitúa a la par que la jurisprudencia británica, pero añade algo más: el objeto de la protección de la Directiva engloba las formas de expresión de un programa de ordenador así como los trabajos preparatorios de concepción que pueden llevar respectivamente a la reproducción o a la creación ulterior de tal programa<sup>561</sup>.

El punto de partida para la exégesis y definición del enunciado “cualquier forma de expresión” es la doctrina Infopaq<sup>562</sup>, según la cual cuando un concepto no haya sido precisado por una Directiva, éste ha de ser definido en función del tenor literal y del contexto del artículo en que figura, y a la luz de los objetivos de dicha Directiva en su conjunto y del Derecho Internacional.

En primer lugar el TJUE repasa la definición de programa de ordenador. Recuerda que conforme al artículo 1 de la Directiva, los programas de ordenador se protegerán como obras literarias en los términos indicados en el Convenio de Berna. A continuación sitúa el contexto normativo, matizando que a la luz de la Directiva, tal como dispone su considerando Séptimo, el término “programa de ordenador” incluye los programas en cualquier forma, incluso los que están incorporados en el “hardware”. Pone un énfasis especial en la parte final de este considerando, donde el legislador comunitario establece que el término programa de ordenador incluye también el trabajo preparatorio de concepción que conduzca al desarrollo del programa. Completa el Tribunal su criterio acudiendo al Derecho Internacional, al citar el Acuerdo ADPIC, el cual declara que los programas de ordenador, sean programas fuente o programas objeto, serán protegidos como obras literarias en virtud del Convenio de Berna.

Establecido el tenor literal de la norma y su contexto, el Tribunal deduce que el objeto de protección de la Directiva, abarca el programa en todas las formas de expresión que conlleven su reproducción en diferentes lenguajes informáticos siempre que tal reproducción permita al programa cumplir su función, como sucede en el caso

---

<sup>559</sup> Sentencia del Tribunal de Justicia (Sala Tercera) de 22 de diciembre de 2010, asunto C-393/09, *Bezpečnostní softwarová asociace - Svaz softwarové ochrany contra Ministerstvo kultury*, DOUE C63 de 26.2.2011, pág. 8.

<sup>560</sup> *Ibidem*, ap. 35

<sup>561</sup> *Ibidem*, ap. 37.

<sup>562</sup> Sentencia del Tribunal de Justicia (Sala Cuarta) de 16 de julio de 2009, asunto C-5/08, *Infopaq International A/S contra Danske Dagblades Forening.*, DO C 220 de 12.09.2009 pág. 7, ap. 32

del código fuente y el código objeto<sup>563</sup>. Además, como obra literaria, el Tribunal establece que también son formas de expresión del programa los trabajos preparatorios de concepción del programa siempre que puedan llevar respectivamente a la reproducción o a la creación ulterior de tal programa.

Por lo tanto, el enunciado “cualquier forma de expresión de un programa de ordenador” ha de interpretarse como aquella expresión que suponga la reproducción del programa, teniendo en cuenta que en una obra utilitaria, para que exista reproducción, tal acto ha de conseguir que la obra funcione, o que permita la creación ulterior de una nueva. Los actos de reproducción de formas de expresión que no tengan como efecto directo hacer funcionar el programa o la creación de uno nuevo, no podrán ampararse bajo la protección específica otorgada por la Directiva.

Definido el enunciado, el Tribunal pasa a analizar si las interfaces gráficas de usuario pueden ser calificadas como forma de expresión del programa. La conclusión a la que llega el Tribunal es negativa.

Su argumentación se centra por un lado, en la especial naturaleza del programa de ordenador como bien funcional o utilitario, de donde se deriva la especificidad del derecho de autor aplicable al programa, y por otro en si las interfaces gráficas suponen una reproducción del programa o no<sup>564</sup>.

El TJUE se remite en varias ocasiones a las conclusiones del Abogado General Sr. Yves Bot<sup>565</sup>, y por lo mismo, algunas merecen ser destacadas. En primer lugar, el Abogado General realiza una clasificación técnica de las interfaces de un programa, aclarando que las interfaces gráficas de usuario pertenecen al grupo de interfaces de interacción, porque permiten una comunicación entre el programa y el usuario, de tal modo que pueden consistir en la mera difusión de información, o pueden permitir al usuario comunicar instrucciones al programa mediante la utilización de comandos<sup>566</sup>.

En segundo lugar, recuerda el objetivo de la Directiva, que no es otro que proteger los programas contra reproducciones no autorizadas por el titular del derecho, y vuelve a insistir en que para que exista una forma de expresión de un programa debe

---

<sup>563</sup> Ibidem, aps. 35 y 38.

<sup>564</sup> Ibidem, ap. 38

<sup>565</sup> Asunto C-393/09, Conclusiones del Abogado General Sr. Yves Bot, presentadas el 14 de Octubre de 2010. Disponible en: <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:62009C0393:ES:HTML> (última consulta 1 de junio, 2013).

<sup>566</sup> Ap. 56 Conclusiones Generales.

producirse la reproducción del mismo, precisamente porque se trata de una obra utilitaria<sup>567</sup>.

El Tribunal recoge ambas argumentaciones y añade lo dispuesto en los considerandos Décimo y Undécimo de la Directiva 91/250, conforme a los cuales las interfaces son las partes del programa de ordenador que establecen la interconexión y la interacción para permitir a los elementos de los soportes físicos y lógicos trabajar con otros soportes físicos y lógicos así como con usuarios en la forma prevista. En particular, una interfaz gráfica de usuario es una interfaz de interacción que permite una comunicación entre el programa de ordenador y el usuario. De esta forma, la interfaz no permite reproducir el programa de ordenador sino que sólo constituye un elemento de dicho programa por medio del cual los usuarios utilizan las funcionalidades de este<sup>568</sup>.

En base a lo anterior, dictamina que la interfaz gráfica de usuario no constituye una forma de expresión de un programa de ordenador en el sentido del artículo 1, apartado 2 de la Directiva 91/250, y por tanto no puede disfrutar de la protección específica por el derecho de autor sobre los programas de ordenador en virtud de la Directiva<sup>569</sup>.

No obstante, a ello añade la posibilidad de que la interfaz gráfica de usuario pueda protegerse a través de los derechos de autor como una obra autónoma, en el sentido establecido por la Directiva 2001/29 siempre que la obra sea original, entendido este requisito como una creación intelectual propia de su autor. Seguidamente, deriva a los tribunales nacionales la tarea de apreciar si concurren los anteriores requisitos en la interfaz, a la vez que proporciona unas directrices que debieran seguir en su examen los tribunales nacionales:

1º Han de tener en cuenta la disposición o la configuración específica de todos los componentes que forman parte de la interfaz gráfica de usuario.

2º Los componentes de una interfaz que sólo se caractericen por su función técnica nunca podrán considerarse originales, porque en tales casos, las diferentes maneras de poner en práctica una idea serían tan limitadas que la idea y la expresión se confundirían<sup>570</sup>. Es decir, en relación a esta clase de

---

<sup>567</sup> Aps. 58 y 61 Conclusiones Generales.

<sup>568</sup> Sentencia del Tribunal de Justicia (Sala Cuarta) de 16 de julio de 2009, asunto C-5/08, Infopaq, *op.cit.*, aps. 39 a 41

<sup>569</sup> *Ibidem*, ap. 42

<sup>570</sup> *Ibidem*, ap. 46-49.



interfaces los jueces nacionales han de aplicar la doctrina de la fusión o *merger doctrine*.

La sentencia de 2 de mayo de 2012, en el asunto C-406/10<sup>571</sup>, no sólo recoge la interpretación del artículo 1.2 del caso *Bezpečnostní*, sino que además matiza determinados elementos que no pueden constituir la forma de expresión de un programa porque los mismos no pueden llevar a su reproducción ni a su copia. Así declara que “ni la funcionalidad ni el lenguaje de programación o el formato de los archivos de datos utilizados en un programa de ordenador para explotar algunas de sus funciones constituyen una forma de expresión del mismo.”<sup>572</sup>

Sin embargo no llega a definir qué ha de entenderse por “funcionalidad” del programa, ni declara claramente que las interfaces sean “elementos funcionales”, como sí había hecho el Abogado General Bot en sus Conclusiones Generales<sup>573</sup>, quien había definido las interfaces como “el conjunto de posibilidades que ofrece un sistema informático, es decir, las acciones propias de dicho programa.”<sup>574</sup>

Es más, el Abogado General recurre al uso de un ejemplo que no deja duda sobre su postura: “Cuando un programador decide desarrollar un programa de ordenador de reserva de billetes de avión dicho software incluirá múltiples funcionalidades necesarias para realizar dicha reserva. En efecto, el programa de ordenador deberá, sucesivamente, encontrar el vuelo que busca el usuario, verificar las plazas disponibles, reservar el asiento, registrar los datos del usuario, tomar en consideración los datos de pago online y, por último, editar el billete electrónico de dicho usuario. Todas estas funcionalidades, estas acciones, vienen definidas por un objeto preciso y limitado. En este sentido son similares a las ideas. En consecuencia, pueden existir programas de ordenador que ofrezcan las mismas funcionalidades. Sin embargo, existe una multitud de medios para lograr concretar dichas funcionalidades y son estos medios los que podrán protegerse mediante el derecho de autor en virtud de la Directiva 91/250. En efecto, como se ha visto, la creatividad, destreza e inventiva se ponen de manifiesto en el propio método de creación del programa, en su escritura. El

---

<sup>571</sup> Sentencia del Tribunal de Justicia (Gran Sala) de 2 de mayo de 2012, asunto C-406/10, *SAS Institute Inc. v. World Programming Ltd.*, DO C 174 de 16.6.2012, pág. 5.

<sup>572</sup> *Ibidem*, ap. 39.

<sup>573</sup> Conclusiones Generales del Abogado General de 29 de Noviembre de 2011 en el asunto C-406/10, *SAS Institute, Inc. v. World Programming Ltd.*, *op.cit.*

<sup>574</sup> *Ibidem*, ap. 52.

programador utiliza *fórmulas, algoritmos que, como tales, están excluidos de la protección mediante el derecho de autor, puesto que se trata del equivalente de las palabras con las que el poeta o el novelista crea su obra literaria. Sin embargo, la manera en que se disponen dichos elementos, como el estilo de la escritura del programa de ordenador, puede reflejar una creación intelectual propia de su autor y, por lo tanto, podrá ser objeto de protección.*<sup>575</sup>

Al no acoger ni respaldar esta definición del Abogado General, podría parecer que el Tribunal se muestra más restrictivo en relación a qué ha de entenderse por funcionalidad. Pero lo cierto es que no es así, como se deduce del siguiente pasaje: “el derecho de autor sobre el programa (...) ofrece la flexibilidad suficiente para permitir a otros autores crear programas similares o incluso idénticos siempre que se abstengan de toda copia”<sup>576</sup>.

Dos importantes conclusiones pueden extraerse de las decisiones del TJUE, a efectos de considerar o no a las interfaces del programa como idea o expresión.

En primer lugar, a juicio del TJUE en bienes de naturaleza utilitaria como los programas de ordenador, para diferenciar entre idea y expresión del programa ha de recurrirse al test de la reproducción no meramente funcional. Es decir, en las obras utilitarias la reproducción es inherente al disfrute de la obra. No puede disfrutarse la obra si no se reproduce, pero cuando la reproducción de alguno o algunos de sus elementos no permiten acceder a los códigos del programa, sino que obedece a un objetivo meramente funcional en sentido técnico, que no sirve para explotar alguna de las funcionalidades concretas del programa, en ese caso, esa información no constituirá una forma de expresión del programa sino una idea, y por tanto, no protegida por el derecho de autor<sup>577</sup>.

Además, con ello parecería recibir respaldo el criterio ya señalado por algunos comentaristas principalmente alemanes, para quienes la interpretación del término “reproducción” debiera basarse en el criterio del interés para el titular de los derechos de

---

<sup>575</sup> Ibidem, aps. 54-55. (La cursiva es nuestra).

<sup>576</sup> Sentencia del Tribunal de Justicia (Gran Sala) de 2 de mayo de 2012, asunto C-406/10, SAS Institute Inc. v. World Programming Ltd., *op.cit.*, p. 41.

<sup>577</sup> Este test basado en la reproducción y aplicado a los programas de ordenador para discernir si una interfaz constituye idea o expresión, no debe confundirse con el test que se aplica para determinar si existe reproducción de una parte sustancial de una obra, establecido en la sentencia del caso Infopaq, que parte del examen de la originalidad y se aplica con carácter general a toda clase de obras. Vid. Sentencia del Tribunal de Justicia (Sala Cuarta) de 16 de julio de 2009, asunto C-5/08, Infopaq, *op.cit.*, aps. 31 a 39.

autor en participar en los beneficios económicos legítimos resultantes de su programa. De esta forma, un acto que técnicamente se podría considerar como una reproducción, pero que no condujese a un aumento en el uso del programa (en el sentido de poder copiar el programa), no debiera considerarse una reproducción a efectos de los derechos de autor<sup>578</sup>.

En segundo lugar, los trabajos preparatorios de creación de un programa de ordenador se considerarán una forma de expresión del mismo, y por tanto, objeto de la protección de la Directiva, siempre que los mismos puedan conducir al desarrollo ulterior de un programa o bien conlleven la reproducción del programa en términos que no sean técnicos ni sirvan para explotar alguna de las funciones del propio programa.

Por consiguiente, a la luz de las directrices establecidas por el TJUE en estas dos sentencias, parece posible afirmar que la Directiva 2009/24/CE no considera a las interfaces del programa parte su objeto, por lo que no gozan de la protección específica de derecho de autor para el software que la Directiva regula.

Este razonamiento ha sido confirmado por la sentencia del Tribunal Supremo británico en el caso SAS Institute Inc v World Programming Ltd, de 25 de Enero de 2013<sup>579</sup>. En el fallo, el juez Arnold, siguiendo los criterios interpretativos del TJUE, declara que el derecho de autor sobre un programa de ordenador no protege ni el lenguaje de programación en que está escrito ni sus interfaces (especialmente, los formatos de datos) o su funcionalidad ni impide su copia<sup>580</sup>.

### *C. 2) Las interfaces del programa ante el requisito de originalidad.*

Si de la aplicación del principio establecido en el artículo 1.2 de la Directiva se concluye que la interfaz es una forma de expresión del programa, habría que examinar si tal forma de expresión es original, y de responder afirmativamente, en principio, no podría impedirse su consideración como parte del objeto protegido por la norma comunitaria.

---

<sup>578</sup> T., DREIER, “The Council Directive of 14 May 1991 on The Legal Protection of Computer Programs”, EIPR 9, 1991, pág. 319 (321); M., LEHMANN, “Die Europäische Richtlinie über den Schutz von Computerprogrammen”, GRURInt. 5, 1991, pág. 327.

<sup>579</sup> SAS Institute Inc v World Programming Ltd, sentencia del Tribunal Supremo (Chancery Division) de 25 de Enero de 2013, [2013] EWHC 69 (Ch).

<sup>580</sup> *Ibidem*, par. 16: “(...) *copyright in a computer program does not protect either the programming language in which it is written or its interfaces (specifically, its data file formats) or its functionality from being copied.*”

El requisito de la originalidad se regula en el artículo 1.3 de la Directiva en los siguientes términos: “El programa de ordenador quedará protegido si fuere original en el sentido de que sea una creación intelectual propia de su autor. No se aplicará ningún otro criterio para conceder la protección”.

El concepto clave de este artículo es saber qué ha de entenderse por “una creación intelectual propia de su autor”. Para ello, empezaremos nuestra exposición recordando la opinión doctrinal mayoritaria sobre el texto del artículo 1.3 de la Directiva, para a continuación examinar la nueva y reciente jurisprudencia del Tribunal de Justicia de la Unión Europea<sup>581</sup>, que tras más de veinte años de silencio por fin se ha pronunciado sobre esta cuestión, despejando bastantes dudas al respecto de la originalidad de las interfaces de software.

Desde el punto de vista teórico, la originalidad de una obra en derecho de autor abarca dos aspectos: en primer lugar, la obra debe ser original en el sentido de que provenga de la actividad creativa de su autor y no de la copia de una obra anterior (originalidad objetiva); en segundo lugar, la originalidad de la obra ha de contener la marca, el sello, el reflejo de la personalidad de su autor (originalidad subjetiva). Por lo tanto, con carácter general, el nivel de originalidad exigible para la protección de una obra dependerá de su naturaleza de, por ende, de la mayor o menor libertad creativa y del grado de complejidad alcanzable por su autor<sup>582</sup>.

En relación a los programas, debido a su naturaleza funcional, su creación tiene lugar en el marco de una reglas de arte y de pasos uniformes de forma que las innovaciones ocurren mediante mejoras paulatinas y acumulativas, al modo de las obras derivadas que por definición suelen incorporar pocos elementos nuevos o se encuentran muy condicionadas por exigencias técnicas<sup>583</sup>. Esta especial característica del programa,

---

<sup>581</sup> El elevado número de ocasiones en las que el Tribunal de Justicia de la Unión Europea ha tenido que interpretar el criterio de originalidad de las directivas sobre derechos de autor, entre 2008 y 2012, ha llevado a algún autor a hablar de hasta tal punto que algún autor ha hablado ya de la armonización oculta del concepto de originalidad. Vid. L., BENTLY, “Harmonization by Stealth: Copyright and the ECJ”, ponencia presentada en la 18ª Fordham Intellectual Property Law and Policy Conference, celebrada el 8-9 de abril de 2010 en Fordham Law School, New York; disponible en: [http://fordhamipconference.com/wp-content/uploads/2010/08/Bently\\_Harmonization.pdf](http://fordhamipconference.com/wp-content/uploads/2010/08/Bently_Harmonization.pdf) (última consulta 1 de junio, 2013)

<sup>582</sup> Sobre la originalidad de la obra en general, vid. R. BERCOVITZ RODRÍGUEZ-CANO, “Comentario al Artículo 10”, en R., BERCOVITZ RODRÍGUEZ-CANO, R. (coord.), *Comentarios...*, *op.cit.*, pág. 154; P., GOLDSTEIN, P.B., HUGENHOLTZ, *International Copyright. Principles, Law, and Practice*, Oxford, Oxford Univ. Press, 2010, pág. 185.

<sup>583</sup> J., DELGADO ECHEVERRÍA, “Comentario al Libro Primero, Título VII” en R., BERCOVITZ RODRÍGUEZ-CANO, (coord.), *Comentarios...*, *op.cit.*, pág. 1406; J., FARRELL, “Standardization and Intellectual...”, *op.cit.*, pág. 41.

unida a la tradición jurídica de cada país, había llevado a que los jueces de distintos Estados miembro, como Alemania, Francia, o el Reino Unido, mantuviesen criterios de originalidad muy diferentes sobre el programa que afectaban negativamente al mercado interior<sup>584</sup>.

De la lectura del texto de la Directiva, en especial del siguiente considerando: “Entre los criterios que deben utilizarse para determinar si un programa de ordenador constituye o no una obra original, *no deben aplicarse los de carácter cualitativo o los relativos al valor estético del programa*”<sup>585</sup>, parece claro que la intención del legislador comunitario fue que la originalidad de los programas de ordenador fuese valorada únicamente bajo el criterio de originalidad objetiva (la doctrina *the skill and labour*), porque debido a la naturaleza funcional de esta clase de obra, la impronta del sello de su autor en el código de programa es difícilmente demostrable. Es decir, el único criterio para determinar la originalidad de un programa es que no sea la copia de otro<sup>586</sup>.

Dos razones justificaban la adopción de un nivel de originalidad tan minimalista:

1ºLa Directiva protege “cualquier forma de expresión” del programa (art. 1.1), por tanto la misma protección recibe el código fuente que el código objeto del programa. El código fuente es la única forma de expresión comprensible para un ser humano, y ésta es normalmente guardada como secreto empresarial, de manera que cuando un tribunal haya de decidir sobre la originalidad de un programa, solo tendrá acceso al código objeto, por lo que para examinar este requisito de la obra se verá obligado a acudir a factores indirectos como el tamaño del programa o la complejidad visual externa que presenta<sup>587</sup>. En este caso, salvando curiosas excepciones como la acontecida

---

<sup>584</sup> Considerando Cuarto de la Directiva 91/250/CEE de 14 de mayo de 1991: “Determinadas diferencias existentes en cuanto a la protección jurídica de los programas de ordenador que ofrecen las legislaciones de los Estados miembros producen efectos negativos y directos sobre el funcionamiento del mercado interior en lo relativo a los programas de ordenador.” Sobre los criterios de originalidad, vid. *supra*, capítulo 3.1.2.A).

<sup>585</sup> Considerando Octavo de la Directiva 91/250/CEE de 14 de mayo de 1991. (La cursiva es nuestra)

<sup>586</sup> Vid. Explanatory Memorandum, Proposal for a Council Directive on the legal protection of computer programs, COM (88) 816 final-SYN 183, 89/C 91/5, Parte Segunda, punto 3.

<sup>587</sup> Vid. Sentencia de 15 de junio de 2006 del Tribunal de Distrito de Haarlem (Holanda), caso Inter IT c. Webcash, en Stibbe ICTLaw newsletter, nº 25, september 2006, p. 10. Un comentario del caso en: A., VAN ROOIJEN, “Case Comment”, AMI 6, (2007), pág. 192. En la doctrina española también lo han reflejado autores como APARICIO VAQUERO y DELGADO ECHEVERRÍA, al afirma que los jueces y juristas en general se ven inclinado a apreciar la similitud sustancial entre programas juzgando por lo que ven en la pantalla del ordenador. En J.P. APARICIO VAQUERO, J. DELGADO ECHEVERRÍA, J., “Comentario al Libro Primero, Título VII”..., *op.cit.*, pág. 1292.

en el juicio que tuvo lugar en Norteamérica entre Oracle y Google<sup>588</sup> en 2012, donde el juez *Alsup* resultó tener conocimientos técnicos por haber estudiado la carrera de Matemáticas y ser programador aficionado<sup>589</sup>, la regla general es que los jueces no disponen de los conocimientos técnicos para valorar la originalidad atribuible al autor de un programa en base a los diferentes estilos de codificar<sup>590</sup>.

2º La mayor o menor originalidad de la expresión del código del programa no guarda relación directa con una funcionalidad superior de éste, de manera que la originalidad de un programa no es en sí misma una característica beneficiosa para el programa<sup>591</sup>.

Sin embargo, la reciente jurisprudencia del Tribunal de Justicia de la Unión Europea deja claro que el criterio de originalidad aplicado a los programas de ordenador no es que no sea la copia de otro.

En primer lugar, en el asunto C-393/09, el TJUE al examinar la originalidad de las interfaces gráficas de usuario, ha declarado que cuando la expresión de los componentes de la interfaz viene impuesta por su función técnica, el criterio de originalidad no se cumple, ya que las diferentes maneras de poner en práctica una idea son tan limitadas que la idea y la expresión se confunden<sup>592</sup>.

Ha sido en 2012 cuando, por fin, el Tribunal de Justicia de la Unión Europea, ha respaldado que esta concepción de “creación intelectual propia de su autor”, no ha de interpretarse como que el programa no sea la copia de otro.

En el caso conocido como *Football Dataco I*<sup>593</sup>, el Tribunal de Apelación de Reino Unido<sup>594</sup>, planteó, entre otras, una cuestión prejudicial relativa a la interpretación del concepto “creación intelectual propia de su autor” aplicada a las bases de datos<sup>595</sup>.

---

<sup>588</sup> Oracle America Inc. v. Google Inc., *op.cit.*, vid. supra. cap. 2.1.4.c).

<sup>589</sup> Vid. [http://en.wikipedia.org/wiki/William\\_Alsup](http://en.wikipedia.org/wiki/William_Alsup) (última visita 31 de enero 2013).

<sup>590</sup> Vid. S., WEBE, *The Success of Open Source*, Cambridge, Harvard University Press, 2005, pág. 136.

<sup>591</sup> P., GOLDSTEIN, “Infringement of Copyright in Computer Programs”, *University of Pittsburgh Law Review* 47, (1986), pág. 1119 (1122);

<sup>592</sup> Sentencia del Tribunal de Justicia (Sala Tercera) de 22 de diciembre de 2010, asunto C-393/09, *Bezpečnostní softwarová asociace ...*, *op.cit.*, ap. 49.

<sup>593</sup> Sentencia del Tribunal de Justicia (Sala Tercera) de 1 de marzo de 2012, asunto C-604/10, *Football Dataco Ltd y otros v. Yahoo UK Limited y otros*, DO C118 de 21.4.2012, pág. 5.

<sup>594</sup> Court of Appeal (England & Wales) (Civil Division) (Reino Unido).

El Tribunal de Justicia de la Unión define una base de datos original como aquella en la que su autor, mediante la selección o la disposición de los datos que contiene, expresa su capacidad creativa tomando elecciones libres y creativas, imprimiendo su “toque personal” a la obra<sup>596</sup>.

A continuación, declara que el criterio de originalidad no se cumple cuando las elecciones se deben a consideraciones técnicas, reglas o exigencias que dejan lugar a la libertad creativa.<sup>597</sup>

Además, el Tribunal fundamenta su definición en la aplicación analógica de lo dispuesto en casos anteriores sobre el criterio de originalidad para obras distintas a las bases de datos: obras literarias y artísticas (caso Infopaq y caso Football Association Premier League), fotografías (caso Painer) y programas de ordenador (caso Bezpečnostní). De tal modo que el TJUE, indirectamente, armoniza el estándar de originalidad a nivel comunitario.

Meses después, siguiendo esta línea interpretativa, y por si quedase alguna duda en relación a los programas de ordenador, la sentencia del asunto C-406/10 SAS Institute v. World Programming, falla: “El artículo 1, apartado 2, de la Directiva 91/250/CEE del Consejo, de 14 de mayo de 1991, sobre la protección jurídica de programas de ordenador, debe interpretarse en el sentido de que **ni la funcionalidad de un programa de ordenador ni el lenguaje de programación o el formato de los archivos de datos utilizados en un programa de ordenador para explotar algunas de sus funciones constituyen una forma de expresión de ese programa** y, por ello,

---

<sup>595</sup> En concreto la cuestión planteada era la siguiente: “¿Qué debe entenderse, a efectos del artículo 3, apartado 1, de la Directiva 96/9 [...], por “las bases de datos que por la selección o la disposición de su contenido constituyan una creación intelectual de su autor”? concretamente:

a) ¿Debe entenderse que quedan excluidos el esfuerzo intelectual y la pericia destinados a la creación de datos?

b) ¿Queda comprendida en “la selección o la disposición” la acción de otorgar una relevancia especial a un dato preexistente (como ocurre con la fijación de una fecha para un partido de fútbol)?

c) ¿Requiere la “creación intelectual de su autor” algo más que un considerable esfuerzo y pericia del autor, y en ese caso, qué requiere?” (Apartado 24 de la sentencia).

<sup>596</sup> Textualmente el apartado 38 de la sentencia declara: “Por lo que se refiere a la constitución de una base de datos, ese criterio de la originalidad se cumple cuando, mediante la selección o la disposición de los datos que contiene, su autor expresa su capacidad creativa de manera original tomando elecciones libres y creativas (véanse, por analogía, las sentencias antes citadas Infopaq International, apartado 45; Bezpečnostní softwarová asociace, apartado 50, y Painer, apartado 89) e imprime así su «toque personal» (sentencia Painer, antes citada, apartado 92).”

<sup>597</sup> *Ibidem*, ap. 39: “En cambio, ese criterio no se cumple cuando la constitución de la base de datos es dictada por consideraciones técnicas, reglas o exigencias que no dejan lugar a la libertad creativa (véanse, por analogía, las sentencias Bezpečnostní softwarová asociace, apartados 48 y 49, y Football Association Premier League y otros, apartado 98).”

carecen de la protección del derecho de autor sobre los programas de ordenador en el sentido de esa Directiva.”<sup>598</sup>.

La aplicación de este criterio de originalidad tiene importantes consecuencias para las interfaces de software:

En primer lugar hay que diferenciar entre los tipos de interfaces que podemos encontrar en un programa de ordenador<sup>599</sup>: por un lado, aquellas que obedecen a requisitos esenciales de funcionamiento, comportamiento y comunicación entre dos o más elementos del programa. Son las interfaces de programas de aplicación, las interfaces de datos y las interfaces de comunicación. Por otra parte, las interfaces que responden a razones de diseño, es decir, que no suponen una interacción de software a software o de software a dispositivo. Éstas son las interfaces de usuario, las conocidas como el “look and feel” del programa.

A la vista del criterio de originalidad establecido por el TJUE, las del primer grupo, que son las relevantes a efectos de alcanzar la interoperabilidad, no son obras originales y por lo tanto, no están protegidas por derecho de autor. Es decir, si bien esta clase de interfaces pueden concebirse en diferentes formatos, y la utilización de unos u otros dependerá de la elección personal del programador, sin embargo, por su propia naturaleza, responden a consideraciones técnicas, reglas y especificaciones unificadas y estándares de comunicación del programa, no a la libertad creativa del programador.

En cuanto a las interfaces de usuario, sin embargo, no puede concluirse lo mismo. Si bien la sentencia del TJUE en el asunto C-393/09 declaró que la interfaz de usuario no constituye una forma de expresión de un programa de ordenador en el sentido del artículo 1.2 de la Directiva 91/250/CEE del Consejo, de 14 de mayo de 1991, no obstante esa interfaz puede ampararse, como una obra, en la protección del derecho de autor como obra autónoma concedida por la Directiva 2001/29/CE del Parlamento Europeo y del Consejo, de 22 de mayo de 2001<sup>600</sup>, si dicha interfaz constituye “una creación intelectual propia de su autor” acorde al criterio establecido en el asunto C-604/10.

---

<sup>598</sup> Sentencia del Tribunal de Justicia (Gran Sala) de 2 de mayo de 2012, asunto C-406/10, SAS Institute Inc. v. World Programming..., op.cit., fallo de la sentencia. (la negrita es nuestra).

<sup>599</sup> Vid. supra, capítulo 1.4.

<sup>600</sup> Sentencia del Tribunal de Justicia (Sala Tercera) de 22 de diciembre de 2010, asunto C-393/09, Bezpečnostní softwarová asociace ..., op.cit., ap. 51.



Este nuevo criterio de originalidad, aplicado a las interfaces, es además acorde con el principio de proporcionalidad que debe regir la Directiva de Software, de manera que el desarrollo de programas parecidos o idénticos funcionalmente a otro debe ser expresamente posible para nuevos programadores, a fin de evitar la aparición de monopolios anticompetitivos<sup>601</sup>.

*D) Las condiciones de la descompilación: Artículo 6.1 de la Directiva.*

*D.1.) La interoperabilidad de un programa creado de forma independiente con otros programas.*

Las dos cuestiones que plantéo este requisito fueron las siguientes:

- 1). El significado de “programa creado de forma independiente”.
- 2). El significado de “interoperabilidad con otros programas”.

*D.1.1) El significado de “programa creado de forma independiente”.*

La excepción solo puede aplicarse cuando la descompilación tiene como finalidad alcanzar la interoperabilidad con un programa creado de forma independiente. Si bien la norma implícitamente sugiere que el programa, en favor del cual se descompila otro, ha de existir con anterioridad al momento de la descompilación, sin embargo la norma no aclara cuánto antes, ni si el programa anterior ha de estar completamente finalizado o no.

El principio básico del derecho de autor es que se protege la forma de expresión de las ideas desde su creación, pero, si para poder crear el nuevo programa necesitamos la información interoperable del previo, ¿cuándo se determina el momento de su creación? Que se haya creado con carácter previo a la descompilación o que su creación vaya a ser consecuencia de realizar la descompilación del programa con el que quiere alcanzar la interoperabilidad tiene consecuencias totalmente distintas.

Algún autor ha indicado que la Directiva indica implícitamente que el programa debe pre-existir en su totalidad o al menos la documentación preparatoria debe estar

---

<sup>601</sup> J., MARLY, “Der Schutzgegenstand des urheberrechtlichen Software-schutzes”, GRUR (2012), pág. 773 (780); S., WESTON, “Software Interfaces – Stuck..., *op.cit.*, pág. 450.

completa<sup>602</sup>. En igual sentido, en el Informe de la Comisión sobre la transposición de la Directiva del año 2000, se hace referencia a la pre-existencia de este segundo programa. La Comisión afirma que “este elemento – el programa creado de forma independiente - se introdujo en la Directiva para garantizar que la descompilación de un programa objeto no se produzca antes de que exista el program creado de forma independiente, aunque solo sea en forma de documentación preparatoria”<sup>603</sup>.

Si el programa no pre-existiera, y su creación se situase después de obtenida la información interoperable, nada impediría su calificación como obra derivada del programa descompilado. Por eso la norma habla de “creación de forma independiente”.

En el plano teórico esto parece razonable, pero si recordamos las fases de creación de un programa<sup>604</sup>, tanto la documentación preparatoria, como la forma en que el programador decide representar los datos del código pueden variar a lo largo de todo el proceso creativo. La creación de un programa comienza con el planteamiento de un problema que pretende ser resuelto con él y termina cuando el código fuente se ha compilado en código objeto.

Esto significa que la forma de expresión del programa protegida por el derecho autor (los códigos) que se haya creado con la información interoperable solo podrá tener lugar después de la descompilación. La idea de qué se quiere conseguir con el programa sí puede pre-existir a la descompilación, pero la forma, el cómo, no podrá expresarse hasta después de tener a disposición la información interoperable. De este modo, el artículo 6 podría estar ampliando el objeto de protección del propio artículo 1.2 de la Directiva, al proteger la expresión de información que es funcional en el programa previo (la información interoperable) sin la cual el nuevo programa no puede existir. Es decir, la norma estaría introduciendo una protección de tipo patente en el derecho de autor sobre el programa de ordenador<sup>605</sup>.

---

<sup>602</sup> L., BENTLY, “Computer Programs Directive”, T., DREIER, P.B., HUGENHOLTZ, (eds.), *Concise European Copyright Law*, Alphen aan den Rijn, The Netherlands, Kluwer Law International, 2006, art. 9, nota 1, pág. 230.

<sup>603</sup> Informe de la Comisión al Consejo, al Parlamento Europeo y al Comité Económico y Social sobre la transposición y los efectos de la Directiva 91/250/CEE sobre la protección jurídica de programas de ordenador, Bruselas, 10.04.200, COM (2000) 199 final, pág. 14.

<sup>604</sup> *Vid. supra*. Cap. 1.2.C.

<sup>605</sup> J.T., SOMA, G., WINFIELD, L., FRIESEN, “Software Interoperability and Reverse Engineering”, *Rutgers Computer & Tech. L.J.* 20, (1994), pág. 189 (239).

Es más, si por interoperabilidad entre los programas entendiéramos que exista una compatibilidad del 100% en el sentido de que el programa creado de forma independiente realice, exacta y en igual forma, todas las funciones del programa descompilado, el artículo 6 estaría legitimando la clonación de programas<sup>606</sup>.

#### *D.1.2) El significado de “interoperabilidad con otros programas”.*

La segunda cuestión que plantea este requisito ya fue objeto de debate durante la elaboración de la Directiva: Si se permite la descompilación de un programa para lograr la interoperabilidad con otro programa creado de forma independiente, ¿hasta qué extremo se está legitimando la creación de programas no sólo competitivos con el descompilado sino también sustituibles? Un programa creado a través de la descompilación de otro se va convertir probablemente en un producto que compita con el programa descompilado en el mercado.

La Comisión trató de aclarar esta cuestión al Parlamento en los siguientes términos: “ *(An independently created program) may connect to the program subject to decompilation. Alternatively it may compete with the decompiled program and in such cases will not normally connect to it*”<sup>607</sup>.

Pero el motivo que por el cual se permite la descompilación en la Directiva es la consecución de interoperabilidad entre programas, no el aprovechamiento del programa descompilado<sup>608</sup>. Sin embargo, la Comisión, al hacer referencia a la posible competencia en el mercado de los programas y a razones económicas, está yendo más allá del ámbito del derecho de autor. Según parte de la doctrina, los efectos derivados de la descompilación como la reducción de costes o la facilitación de desarrollo de programas a la competencia están fuera de lo que se entiende por interoperabilidad en la Directiva<sup>609</sup>.

---

<sup>606</sup> B., CZARNOTA, J., HART, J., *Legal Protection of...*, *op.cit.*, pág. 78.

<sup>607</sup> EUR. DOC. SEC (91) 87 FINAL-SYN 183, 1991, pág. 5. (Un programa creado de forma independiente, debe poder conectarse al programa objeto de descompilación. Otra posibilidad es que pueda competir con el programa descompilado y en tal caso, no se conectaría con él)

<sup>608</sup> Vid. por ejemplo, M. LEHMANN, “Die Europäische Richtlinie über den Schutz von Computerprogrammen”, en M., LEHMANN, *Rechtsschutz und Werwetung...*, *op.cit.*, pág. 22; J.F., VERSTRYNGE, “Protecting Intellectual Property...”, *op.cit.*, pág. 9.

<sup>609</sup> B., CZARNOTA, J., HART, *Legal Protection of...*, *op.cit.*, pág. 83.

## D.2) La persona autorizada.

La letra a) del artículo 6.1 indica que los actos de reproducción que permite la excepción, pueden ser ejecutados por “el licenciataria” o “por cualquier otra persona facultada para utilizar una copia del programa”, o “en su nombre por parte de una persona debidamente autorizada”.

El legislador podría haberse limitado a repetir la misma expresión que emplea en la excepción de ingeniería inversa del artículo 5.3 de la Directiva, “usuario legítimo”, ya que las tres personas que enumera, podría decirse que son usuarios legítimos del programa.

Además, la introducción de este concepto se produjo por primera vez en el Derecho de Autor continental a través de esta Directiva, lo cual supuso la introducción de una nueva percepción en la delimitación de este derecho de exclusiva<sup>610</sup>. Si bien esta percepción concuerda con la realidad contractual de las transacciones sobre bienes de información, también implica una especie de “despersonalización” del Derecho de Autor: la razón de protección del sistema deja de ser únicamente el autor de la obra para pasar a ser la inversión económica realizada por la industria cultural<sup>611</sup>, y además el usuario legítimo solo puede utilizar la obra en el contexto de los límites específicamente descritos por el autor<sup>612</sup>.

Ahora bien, después de la sentencia del caso C-406/10 de 2 de mayo de 2012 que el TJUE parece que la triple enumeración del artículo 6 no se debió al azar ni a descuido. En esta sentencia, el Tribunal se refiere al “usuario legítimo” del artículo 5.3 como la persona que haya obtenido una copia con licencia de un programa de

---

<sup>610</sup> T., SYNODINOU, “The Lawful User and a Balancing of Interests in European Copyright Law”, IIC 41(7), 2010, pág. 819 (823); P.B., HUGENHOLZT, “Copyright, Contract and Technology: What will remain in the public domain?”, en S., DUSOLLIER, J., GINSBURG, P.B., HUGENHOLZT *et al*, *Le droit d'auteur : un contrôle de l'accès aux oeuvres ? - Copyright : a right to control access to works ?*, Cahier du CRID (Centre de Recherches Informatique et Droit), 18, Bruylant, Brussels, 2000, pág. 84-85.

<sup>611</sup> T., SYNODINOU, “The Lawful User...”, *op.cit.*, pág. 820; VIVANT, M., “Deuxième exception légale: la décompilation”, *Lamy droit de l'informatique et des réseaux*, 199-202, 2005, pág. 122. Lo cual por otra parte confirma que el objetivo de protección del derecho de autor en el caso de los programas es la utilidad del bien y no la persona del autor. Vid. *supra*, capítulo 3.2.1.

<sup>612</sup> L., GUIBAULT, *Copyright limitations and contracts...*, *op.cit.* pág. 17.

ordenador<sup>613</sup>. Es decir, parece sugerir que el único “usuario legítimo” de un programa sería su licenciatarario.

Quizás lo más que se pueda reprochar entonces es que en lugar de emplear el término “licenciatarario”, podría haberlo sustituido por el de “usuario legítimo” para no dejar lugar a ninguna duda interpretativa.

Los otros dos sujetos que menciona el artículo (cualquier persona facultada para utilizar una copia, o un representante autorizado del licenciatarario) parecen referirse al programador o equipo de programación que tenga que llevar a cabo la descompilación. Quizás, el legislador comunitario tenía en mente la figura norteamericana de “*works made for hire*” al redactar el artículo 6<sup>614</sup>. De hecho, la documentación que recoge la historia legislativa de la Directiva revela que esta noción de la doctrina norteamericana fue inicialmente contemplada, pero finalmente desechada<sup>615</sup>, aunque en realidad no lo parezca.

El ámbito personal del artículo 6 es por tanto más amplio que el del artículo 5.3, que sólo autoriza a licenciatararios.

Por otra parte, no ha de perderse de vista que la práctica totalidad de los programas son distribuidos hoy por hoy a través de licencias de uso no personalizadas, lo cual significa que el término usuario hace referencia tan sólo al sujeto que utiliza el programa, no conlleva calificación jurídica alguna<sup>616</sup>, y por lo tanto, no ha de confundirse con la persona autorizada por la excepción del artículo 6.

---

<sup>613</sup> Sentencia del Tribunal de Justicia (Gran Sala) de 2 de mayo de 2012, asunto C-406/10, SAS Institute..., *op.cit.*, ap. 47.

<sup>614</sup> Vid. P., HEINDL, *A Status Report...*, *op.cit.*, pág. 62; D., KELLEHER, K., MURRAY, *IT Law in the European Union*, London, Sweet & Maxwell, 1999, pág. 11. Los autores hacen referencia al caso de Atari Games Corporation v. Nintendo of American Inc. donde el acceso a la copia del programa descompilado tuvo lugar de forma ilegítima.

<sup>615</sup> Amendment Proposal for a Council Directive on the Legal Protection of Computer Programs, 1991 OJ C 321, 81, art. 2(4), en nota 79, sugiriendo la inserción del siguiente texto: “Where a computer programa is created by an employee in the execution of his duties or following the instructions given by his employer, the employer shall be entitled to exercise all economic rights in the program so created, unless otherwise provided by contract”.

<sup>616</sup> J.P., APARICIO VAQUERO, *Licencias de uso no personalizadas de programas de ordenador*, Granada, Comares, 2004, pág. 81.

#### *D.3) La información necesaria para conseguir la interoperabilidad.*

La información a la que permite acceder la excepción configurada en el artículo 6 de la Directiva, es “la necesaria para la interoperabilidad de un programa de ordenador creado de forma independiente con otros programas”. Sin embargo, el texto de la norma no ofrece ningún criterio para determinar cuál es la información necesaria. Ciertamente es de la lectura del Considerando Décimo cabe deducir que la información necesaria es la contenida en las interfaces<sup>617</sup>, ya que gracias a ellas puede establecerse la interconexión e interacción funcional entre programas<sup>618</sup>, es decir, la interoperabilidad.

Este requisito se entrelaza con el de la indispensabilidad de la descompilación, ya que si al Considerando Décimo añadimos lo dispuesto por la letra c) del artículo 6.1, que dispone que los actos de reproducción autorizados por la excepción son únicamente los que se circunscriban a las partes del programa original que resulten necesarias para conseguir la interoperabilidad, el resultado final es que esas partes son las interfaces.

#### *D.4) La indispensabilidad de la descompilación.*

La condición de la indispensabilidad, como han indicado algunos autores<sup>619</sup>, ha de valorarse teniendo en cuenta lo dispuesto en el párrafo primero del artículo 6.1 junto con el contenido de la letra c) del mismo apartado<sup>620</sup>. De su lectura conjunta cabe concluir que el código a descompilar debe limitarse al mínimo imprescindible para la consecución de su objetivo. Por lo tanto, solo si la persona autorizada no dispone de información suficiente a través del titular del programa (artículo 6.1, letra b), o si dicha información no puede ser conocida a través de otros medios legítimos, como la excepción configurada en el artículo 5.3 de la Directiva<sup>621</sup>, entonces la reproducción de

---

<sup>617</sup> “(...) Las partes del programa que establecen dicha interconexión e interacción entre los elementos de software y hardware suelen denominarse interfaces. (...)”

<sup>618</sup> Vid. Considerando Décimo de la Directiva Software.

<sup>619</sup> W., BLOCHER, “Article 6 Decompilation” en M., WALTER, S. VON, LEWINSKI, (eds.) *European copyright law...*, *op.cit.*, pág. 175, nota 5.6.35; J.F., VERSTRYNGE, “Protecting Intellectual Property ...”, *op.cit.*, pág. 8; J., MASSAGUER FUENTES, “La adaptación de...”, *op.cit.*, pág. 66; M., GÓMEZ PERALS, *La cesión de...*, *op.cit.*, pág. 281.

<sup>620</sup> Artículo 6.1.c) de la Directiva: “que dichos actos se limiten estrictamente a aquellas partes del programa original que resulten necesarias para conseguir la interoperabilidad.”

<sup>621</sup> Artículo 5.3: “El usuario legítimo de la copia de un programa de ordenador estará facultado para observar, estudiar o verificar su funcionamiento, sin autorización previa del titular, con el fin de determinar las ideas y principios implícitos en cualquier elemento del programa, siempre que lo haga

esa parte del código será indispensable para lograr la interoperabilidad, cumpliendo con este requisito de la excepción.

En la práctica esta condición es incongruente porque presupone que la persona autorizada a descompilar el código binario de un programa tiene conocimiento previo sobre la ubicación de las interfaces. La norma solo permite usar la descompilación para reproducir partes del programa que tengan relación con la información interoperable, y esto es imposible porque el programador precisamente recurre a la descompilación por no haber tenido otro modo de acceder a la información interoperable.

Aunque si el usuario no conoce la ubicación de esta información, que suele ser lo habitual, también podría interpretarse el texto de la norma en el sentido de que permite descompilar el programa completo<sup>622</sup>.

En esta línea interpretativa ha planteado la doctrina alemana que si, además de la información de las interfaces definidas como tal por el fabricante del programa, podría incluirse entre la información necesaria otra clase de información técnica que el fabricante no defina como tal, como los datos referentes a potenciales puntos de conexión con el programa que tuvieran interés para el creador del nuevo programa al proceder con la descompilación.

El problema tanto de esta interpretación como de la que la norma permite la descompilación del programa completo, es que irían en contra del principio de interpretación restrictiva de las excepciones, y además, en relación a los intereses del titular, aceptar dichas interpretaciones supondría conceder acceso a información del programa que, a priori, no sea 100% técnica, con lo cual, la expresión del programa quedaría desprotegida<sup>623</sup>.

Otro problema de esta condición es la carga probatoria a la que tenga que enfrentarse la persona que descompila el programa, porque la prueba de que la

---

durante cualquiera de las operaciones de carga, visualización, ejecución, transmisión o almacenamiento del programa, que tiene derecho a hacer”.

<sup>622</sup> U., MYLLY, “An Evolutionary Economics ...”, *op.cit.*, pág. 317.

<sup>623</sup> T., DREIER, “Comentario §69e”, en T., DREIER, G., SCHULZE, *Urheberrechtsgesetz: Kommentar*, Munich, 4ª ed., Beck, 2012, ap. 16; M., GRÜTZMACHER, “Comentario §69e”, en A.A., WANDTKE, W., BULLINGER, *Praxiskommentar zum Urheberrecht*, 3ª ed., Munich, Beck, 2009, ap. 16; T., VINJE “The EC Directive on the protection of Computer Programs and the Question of Interoperability” (Die EG-Richtlinie zum Schutz von Computerprogrammen und die Frage der Interoperabilität), *GRUR Int.* 4, 1992, pág. 250 (257).

descompilación era técnicamente inevitable, que lo era sobre partes indispensables para lograr la interoperabilidad y no sobre otras no susceptibles de uso legítimo, y que no conocía la ubicación de la información sobre las interfaces, corresponde a quien realiza la descompilación.

Esta carga probatoria está implícita en el artículo 6.1 letra b), que indica que la descompilación no será indispensable cuando la información necesaria, es decir la de las interfaces, haya sido puesta previamente, y de manera fácil y rápida, a disposición de las personas autorizadas a descompilar el programa.

Por lo tanto, la descompilación no está amparada por la excepción si la información interoperable ha sido publicada. En la práctica esto se ha traducido en que muchas compañías de software publican información sobre sus interfaces. Ello obedece a razones económicas: el hecho de que un programa sea interoperable con otros lo hace más atractivo al consumidor. Como se había indicado en el capítulo primero, la capacidad de interoperar de un programa está relacionada con los efectos red (network effects), ya que cuantos más productos interoperables haya en el mercado, más valioso será ese producto para el consumidor. De nuevo nos lleva esta condición a la protección de la inversión económica realizada y no a la protección del autor.

Si lo comparamos con la doctrina del fair use norteamericano donde los competidores solo pueden acudir a la ingeniería inversa del programa cuando esta clase de procedimiento es el único medio que permite acceder a la información interoperable, es posible que con la inclusión de esta condición la Directiva tratase de incentivar la publicación de información interoperable por los fabricantes de software. Por otro lado, las situaciones en que puede realizarse la ingeniería inversa sobre un programa en Estados Unidos van más allá que para la interoperabilidad de otro programa creado de forma independiente.

Esta es la única justificación de la descompilación permitida por el artículo 6: la consecución de interoperabilidad. Así lo declara el párrafo primero del artículo y la letra a) del apartado segundo:

*Art. 6.1: “No se exigirá la autorización del titular del derecho cuando la reproducción del código y la traducción de su forma (...) se indispensable para obtener la información necesaria para la interoperabilidad (...)”.*



*Art. 6.2, letra a): “La aplicación de lo dispuesto en el apartado 1 no permitirá que la información así obtenida: se utilice para fines distintos de la consecución de la interoperabilidad del programa de ordenador creado de forma independiente”.*

La norma deja claro que la descompilación es la última opción, una vez agotados los demás medios para acceder a la información interoperable, sea porque no están a disposición del usuario o sea porque los medios no son apropiados. En este sentido se pronuncia Memoria Explicativa de la Comisión en su apartado 3.14, al indicar que aunque la descompilación es técnicamente posible, se trata de un proceso lento, costoso y poco eficiente. Suele ser más eficiente que las partes negocien los términos en los que se ponga a disposición la información interoperable<sup>624</sup>.

Por lo tanto, el requisito de la indispensabilidad, en contraste con el de la necesidad, sugiere un criterio totalmente objetivo<sup>625</sup>, que no se habrá satisfecho si el demandante afirma que existían otras vías alternativas a la descompilación para obtener la información interoperable, como por ejemplo, el análisis de caja negra descrito en el artículo 5.3 de la Directiva, o la negociación de dicha información, aunque nada se dice sobre las condiciones de tal negociación. Correspondería al demandado probar que tales vías no eran suficientes y que la descompilación era indispensable.

De este modo, la Directiva deja el problema de la indispensabilidad de la descompilación en manos de la aplicación que los tribunales nacionales de los Estados Miembros hagan al respecto<sup>626</sup>.

#### *E) El uso de la información obtenida con la descompilación.*

Una vez efectuada la descompilación en los términos indicados en el apartado primero del artículo 6, su apartado segundo regula qué puede hacerse con dicha información.

---

<sup>624</sup> Explanatory Memorandum, Proposal for a Council Directive on the legal protection of computer programs. COM (88) final-SYN 183, p. 96.

<sup>625</sup> B., CZARNOTA, J., HART, *Legal Protection of...*, *op.cit.*, pág. 77; W., BLOCHER, “Article 6 Decompilation” en M., WALTER, S.VON, LEWINSKI, (eds.) *European copyright law...*, *op.cit.*, pág. 172, nota 5.6.28.

<sup>626</sup> B., CZARNOTA, J., HART, *Legal Protection of...*, *op.cit.*, pág. 80; C., GUILLOU, “The Reverse Engineering of Computer Software in Europe and the United States”, *Colum.-VLA J.L. & Arts* 22, 1998, pág. 533 (543); K., GILBERT-MACMILLAN, “Intellectual Property Law for Reverse Engineering Computer Programs in the European Community”, *Santa Clara CHTLJ* 9, 1993, pág. 247 (259).

Para ello la norma diferencia tres supuestos distintos, que pueden resumirse en dos: Que se utilice la información para usos distintos a la consecución de la interoperabilidad, lo cual normalmente conllevará un acto que infrinja los derechos de autor sobre el programa, y que la información se comunique a terceros.

*E.1.) La excesiva restricción sobre los usos de la información descompilada.*

La condición de que la única finalidad de la información obtenida por descompilación sea la interoperabilidad, es la que conlleva las más importantes consecuencias sobre los usos de la información interoperable.

El segundo apartado del artículo 6 regula los usos de la información obtenida a través de la descompilación autorizada por el apartado primero. Las dos claves se encuentran en qué ha de entenderse por “información obtenida” y qué por “terceros”.

En relación a la “información obtenida”, para algunos, la interpretación literal de la norma impone una obligación legal de secreto: no cabe la reutilización de la información obtenida desde el código por nadie, aunque tal información consista en elementos no protegidos por el derecho de autor, para otra finalidad que no sea la de alcanzar la interoperabilidad con un programa creado de forma independiente<sup>627</sup>. De este modo, la excepción estaría impidiendo la libre utilización de las ideas y los principios que se hubiesen conocido mediante la descompilación para la creación de un nuevo programa distinto en su forma pero que se basase en las mismas ideas y principios<sup>628</sup>.

Sin embargo, otros autores ponen el acento en el significado del verbo permitir, de manera que la no permisibilidad no puede referirse a información que no está protegida por el Derecho de Autor, como son las interfaces, sino a cualquier otra información distinta que haya sido obtenida siguiendo los requisitos establecidos por el apartado primero de la norma<sup>629</sup>.

---

<sup>627</sup> B., CZARNOTA, J., HART, *Legal Protection of...*, *op.cit.*, pág. 77; MASSAGUER FUENTES, “La adaptación de...”, *op.cit.*, pág. 67.

<sup>628</sup> J.P., APARICIO VAQUERO, J., DELGADO ECHEVERRÍA, “Comentarios al Libro Primero, Título VII” ..., *op.cit.*, pág. 1388.

<sup>629</sup> W., BLOCHER, “Article 6 Decompilation” en M., WALTER, S.VON, LEWINSKI, (eds.) *European copyright law...*, *op.cit.*, pág. 176, nota 5.6.41.

Lo cierto es que si los redactores de la Directiva quisieron introducir una prohibición sobre el uso y la cesión a terceros de información no protegida por el Derecho de Autor, tendría que haber sido a través de una mención explícita<sup>630</sup>.

Además, la postura de que una interpretación literal de la norma no permite la reutilización de la información obtenida, merece dos críticas:

1ºLa técnica legislativa empleada no es la más adecuada porque, si la mención que hace el apartado 6.2.b) a la comunicación a terceros de información interoperable va más allá de la reproducción del código, en este apartado se estaría incluyendo una excepción dentro la propia excepción. En concreto sobre la facultad de distribución, que no está contemplada en el artículo 6<sup>631</sup>.

2ºNo permitir el uso de información procedente de la descompilación para actividades de investigación y desarrollo, podría vulnerar el artículo 179(1) del TFUE, que enumera la investigación y el desarrollo tecnológico entre los objetivos de la Unión Europea<sup>632</sup>.

En relación al significado de “terceros”, se siga una interpretación literal o se parta del verbo permitir, la persona o personas a quienes se puede comunicar la “información obtenida” podrían ser o bien otro competidor o creador que necesite lograr interoperabilidad con el programa creado por el descompilador. Aunque la norma tampoco deja claro si ese tercero ha de ser el titular de un programa anterior, el titular de un programa nuevo, o una persona autorizada a usar un programa anterior.

Esta complicación en la determinación de quiénes son los “terceros” deriva de la referencia a “salvo cuando sea necesario a efectos de interoperabilidad”. Si por necesario a efectos de interoperabilidad la norma remite al apartado primero, y nada parece indicar lo contrario, la Directiva no permitiría la descompilación que tenga por objeto un uso estrictamente privado, ni para investigación, labores de mantenimiento o

---

<sup>630</sup> Ibidem, pág. 176, nota 5.6.41.

<sup>631</sup> W., BLOCHER, “Article 6 Decompilation” en M.,WALTER, S.VON, LEWINSKI (eds.), *European copyright law...*, *op.cit.*, pág. 176, nota 5.6.42.

<sup>632</sup> Art. 179(1) TFUE: “La Unión tendrá por objetivo fortalecer sus bases científicas y tecnológicas, mediante la realización de un espacio europeo de investigación en el que los investigadores, los conocimientos científicos y las tecnologías circulen libremente, y favorecer el desarrollo de su competitividad, incluida la de su industria, así como fomentar las acciones de investigación que se consideren necesarias en virtud de los demás capítulos de los Tratados.”

de mejora del programa como corrección de errores, seguridad informática, o incluso la creación de programas optimizados<sup>633</sup>.

Sin embargo, a falta de criterios interpretativos claros por parte del TJUE, el uso que puede hacerse de la información interoperable obtenida de la descompilación, sigue siendo una cuestión no resuelta.

### *E.2.) El procedimiento de “sala blanca”: ¿Un salvoconducto?*

Una de las razones por las que cabe pensar que el artículo 6.2.b) prohíbe la comunicación de información interoperable a terceros, es para tratar de evitar los procedimientos de “*cleanroom*” o “sala blanca” (también traducido como “sala limpia”), que tienden a escribir un programa totalmente original en su expresión (un código totalmente nuevo) que realice las mismas funciones de otro, con el que competirá en el mercado. Para evitar la contaminación entre el código del programa cuyas ideas y funciones se toman y el que se va crear, el proceso se realiza con dos equipos. El primero, mediante todas las técnicas de “ingeniería inversa”, entre ellas la descompilación, analiza el programa para entender sus ideas y principios. Hecho esto, al segundo equipo le corresponde desarrollar un programa que cumpla con las ideas descritas por el primer equipo<sup>634</sup>. De este modo, es casi imposible que la escritura del código del segundo programa coincida en algo con el primero, mientras que podrá desarrollar casi exactamente las mismas tareas o funciones.

Esta similitud funcional entre el programa descompilado y el nuevo está permitida por el artículo 6.2.c), según el cual, la información obtenida a través de la descompilación “no puede utilizarse para el desarrollo, producción o comercialización de un programa de ordenador sustancialmente similar en su expresión, o para cualquier otro acto que infrinja los derechos de autor”.

En el Informe sobre la política de competencia de 1991, la Comisión afirmaba que un programa podrá competir con el programa que haya sido descompilado y en tal

---

<sup>633</sup> J.F., VERSTRYNGE, "Protecting Intellectual Property...", *op.cit.*, pág. 8; B., CZARNOTA, J., HART, *Legal Protection of...*, *op.cit.*, pág. 80; T., VINJE "The EC Directive on...", *op.cit.*, pág. 258; J., MARLY, *Urheberrechtsschutz für Computersoftware in der Europäischen Union*, München, Beck, 1995, pág. 329

<sup>634</sup> Vid. S.J., POWEL, C.J., TRAMMELL, R., LINGER et al. *Cleanroom Software Engineering. Technology and Process*, Addison-Wesley, kindle ed., Mass., 1999.

caso, por lo general, no se conectará con él<sup>635</sup>. La intención de la Comisión era que el usuario legítimo tuviera derecho a utilizar la información obtenida a través de la descompilación para la fabricación de un producto que compitiera con el programa original o que se uniera al él, siempre que no fuera sustancialmente similar en su expresión.

Como ha declarado expresamente la Corte de Casación francesa, “*le principe de libre concurrence ne saurait justifier l'utilisation d'un logiciel au mépris des droits d'auteur*”<sup>636</sup>. Ahora bien, no puede existir infracción del derecho de autor allá donde idea y expresión sean idénticas por ser ésta la única forma de exponer aquélla<sup>637</sup>.

Si a esto añadimos los criterios del TJUE en los asuntos C-393/09 y C-406/10, según los cuales, ni la funcionalidad ni el lenguaje de programación o el formato de los archivos de datos utilizados en un programa de ordenador para explotar algunas de sus funciones constituyen una forma de expresión del mismo, y el criterio interpretativo de “creación intelectual propia de su autor” elaborado por la jurisprudencia comunitaria<sup>638</sup>, resulta que un procedimiento de sala blanca llevado a cabo para crear un programa independiente, siempre que cumpla con las restrictivas condiciones del artículo 6.1, no infringiría el derecho de autor sobre el programa.

Así lo reconoce el TJUE en la sentencia del caso C-406/10 al declarar que la reproducción de la funcionalidad del programa sin acceder directamente al código fuente no infringe el derecho de autor sobre el programa. Sólo si la persona autorizada se sirviera del código fuente (que presumiblemente hubiera reconstruido tras un proceso de descompilación exitoso) para crear elementos similares en su propio programa, tal comportamiento podría constituir una reproducción parcial<sup>639</sup>.

---

<sup>635</sup> Comunidades Europeas, Comisión, XX Informe sobre la política de competencia, Luxemburgo, Oficina de Publicaciones Oficiales de las Comunidades Europeas, 1991, CM-69-91-410, pág. 77

<sup>636</sup> (El principio de la libre competencia no puede justificar el uso de cualquier software que viole el derecho de autor) En Cass. Civ. 1re, 20 octubre 2011, pourvoi n°10-14069, Bull. civ., ap. 4°.

<sup>637</sup> MASSAGUER FUENTES, “La adaptación de...”, *op.cit.*, pág. 67.

<sup>638</sup> Vid. *supra*, capítulo 3.2.3.C)

<sup>639</sup> Sentencia de 2 de mayo de 2012, asunto C-406/10, SAS Institute, *op.cit.*, ap. 41 - 43.

### *E.3.) La invitación implícita a negociar entre las partes.*

La complejidad técnica de realizar la descompilación de un programa con éxito, sumado a los costes económicos y temporales que conlleva, unidos con la dificultosa interpretación del texto de la excepción del artículo 6 de la Directiva, puede llevar a pensar que una finalidad implícita de esta norma era la estimulación de las negociaciones amistosas entre fabricantes de software para suministrarse información interoperable<sup>640</sup>.

Tanto el titular de los derechos de autor sobre el programa como el programador del nuevo programa que haya de interoperar con aquél están expuestos a elevados riesgos: Para el titular, la vulnerabilidad del código del programa, especialmente la información contenida en el código fuente. Para el programador independiente, los costes económicos y temporales que supone la descompilación. Además, ambos están expuestos a los costes legales derivados del incumplimiento de alguna de las condiciones de la norma, especialmente, la carga de la prueba de la indispensabilidad que recae sobre el programador del nuevo programa. Es más, una visión global de la situación demuestra que los resultados que puedan adquirirse a través de un proceso como la descompilación suponen la duplicación de recursos dirigidos a obtener una información que el programador original ya posee<sup>641</sup>.

La opción de acudir a un sistema de licencias voluntarias de la información interoperable podría evitar los riesgos expuestos y ser además una solución más eficiente. De tal modo que prescindir de la ingeniería inversa sería el estímulo para sentarse a negociar cuando un fabricante de software necesitase información

---

<sup>640</sup> Proposal for a Council Directive on the legal protection of computer programs, COM (88) 816 final-SYN 183, 89/C 91/5, Parte Segunda, 3.14: “It is usually more efficient for the parties concerned to agree on the terms under which the information will be made available.”

<sup>641</sup> Como indicaba la Comisión en Proposal for a Council Directive on the legal protection of computer programs, COM (88) 816 final-SYN 183, 89/C 91/5, Parte Segunda, 3.14, “aunque es técnicamente posible descompilar un programa (...) es un procedimiento largo, costoso e ineficiente”. J., GIBSON, “Once and Future...”, *op.cit.*, pág. 197; P., SAMUELSON, S., SCOTCHMER, “The Law and Economics...”, *op.cit.*, pág. 1588; D., SCHMIDTCHEN, C. KOBOLDT, “A Pacemaker That Stops Halfway: The Decompilation Rule in the EEC Directive on the Legal Protection of Computer Programs”, *International Review of Law and Economics* 13, 1993, pág. 413 (422); B., CZARNOTA, J., HART, *Legal Protection of...*, *op.cit.*, pág. 80. En contra de esta opinión vid: L., BENTLY, “Computer Programs Directive”, en T., DREIER, P.B., HUGENHOLTZ (eds.), *Concise European Copyright...*, *op.cit.*, pág. 231.

interoperable de un competidor para poder elaborar un programa interoperable, a la vez que posiblemente competitivo<sup>642</sup>.

Sin embargo, la propia negociación entraña costes para ambas partes que pueden superar la voluntad de las partes en invertir tiempo en negociar, en especial en un mercado atípico como el del software, sujeto a “network effects”<sup>643</sup>. Entre estos costes pueden enumerarse: el estado de la técnica de las tecnología de ingeniería inversa, las condiciones legales prescritas por la Directiva, los recursos con que cuente el competidor o la complejidad del programa a descompilar<sup>644</sup>.

El texto de donde cabe deducir la invitación implícita a negociar se encuentra en el artículo 6.1.b) de la Directiva, a través de la no autorización de la descompilación cuando “la información necesaria no haya sido puesta previamente, y de manera fácil y rápida,” a disposición de las personas autorizadas<sup>645</sup>. Es una invitación sucinta al titular del derecho de autor de poner a disposición de terceros la información interoperable de su programa, para evitar la descompilación.

Sin embargo, el artículo 6 no especifica en ningún momento cuándo ha de considerarse que la información se ha puesto a disposición ni qué cantidad de información debe facilitarse. Ni siquiera si ha de hacerlo de manera gratuita o a cambio de una contraprestación. En opinión de autores como DREIER o SCHRICKER, la exigencia de una contraprestación a cambio de la información interoperable no tiene cabida en la norma indicada<sup>646</sup>. Para otros, sin embargo, la exigencia de una contraprestación es aceptable, aunque sujeta a las normas de derecho de defensa de la competencia<sup>647</sup>.

---

<sup>642</sup> Propuesta realizada por A. VAN ROOIJEN, *The Software Interface...*, *op.cit.*, pág. 84.

<sup>643</sup> Vid. supra, capítulo 1.3.A) y B).

<sup>644</sup> D., SCHIDTCHEN, C., KOBOLDT, “A Pacemaker That ...”, *op.cit.*, pág. 422.

<sup>645</sup> B., CZARNOTA, J., HART, *Legal Protection of...*, *op.cit.*, pág. 80; O., STAFFELBACH, *Die Dekompilierung von Computerprogrammen gemäss Art. 21 URG*, Bern, Stämpfli, 2003, pág. 117.

<sup>646</sup> T., DREIER, “The Council Directive of 14 May 1991 on the Legal Protection of Computer Programs”, *EIPR* 13, 1991, pág. 319 (324); G., SCHRICKER, *Urheberrecht*, München, Beck, 1999, pág. 1122. En igual sentido, O., STAFFELBACH, *Die Dekompilierung von...*, *op.cit.*, pág. 119. “Readily available should be construed as meaning information which must be given upon request and without payment of a royalty” en M., LEHMANN, “Germany” en M. LEHMANN, C. TAPPER (eds.) *A Handbook of...*, *op.cit.*, pág. 31.

<sup>647</sup> B., CZARNOTA, J., HART, J., *Legal Protection of...*, *op.cit.*, pág. 80; H., PEARSON, C., MILLER, N., TURTLE, “Commercial Implications of the European Software Copyright Directive”, *International Quarterly* 5(1), 1993, pág.108 (110), J.A., GÓMEZ SEGADÉ, “Spain”, en LEHMANN, M., TAPPER, C., (eds.), *A Handbook of...*, *op.cit.*, pág. 39.

Esta opción de que si las partes negociadoras llegan a un acuerdo sobre el canon a pagar, tal canon sea revisado por el derecho de defensa de la competencia, no sería problemática si obviamos que las intenciones de Consejo y Comisión al redactar el artículo 6 de la Directiva, eran por un lado la promoción de la innovación y el desarrollo de estándares y sistemas abiertos, y por otro lado, evitar la intervención del derecho de defensa de la competencia<sup>648</sup>.

Por lo tanto la posición de DREIER, LEHMANN y SCHRICKER parece la más equilibrada.

Además, si el titular del derecho de autor sobre el programa es obligado a aceptar un determinado derecho de licencia antes de que se realice la descompilación, la razón de ser de la excepción de la descompilación desaparece.

En conclusión, el artículo 6 no contiene ninguna obligación implícita a negociar, o que permita forzar al titular del programa que no ponga a disposición de forma rápida y fácil la información interoperable a hacerlo, de manera que no puede afirmarse que exista una presunta invitación en el texto del apartado primero, letra b) de la excepción.

### **3. La interpretación del artículo 6 de la Directiva por los tribunales de los Estados miembro.**

#### *3.1. Alemania.*

La transposición de la Directiva tuvo lugar en Alemania a través de la modificación de la Ley de Derecho de Autor de 24 de Junio de 1993<sup>649</sup>. Con esta modificación se introdujo un nuevo Título VII (§ 69 a et seq. UrhG) bajo el epígrafe “Disposiciones específicas para programas de ordenador”.

---

<sup>648</sup> Annex 1, area 3, of Council Decision 91/394/EEC of 8 July 1991, adopting a specific research and technological development programme in the field of information technologies (1990 to 1994) (OJL 218, 22): Another strategically important research direction is open systems for the integration of heterogeneous software components supporting different applications”. Annex 1, point 3 of Council Decision 91/691/EEC of 12 December 1991 adopting a programme for the establishment of an internal information services market: As a complement to current efforts for Open Systems Interconnection (OSI), the Commission will promote development of open information interchange standards in cooperation with existing standardization structures such as EWOS, ETSI and CEN/Cenelec. The demonstration and efficient application of information standards or industry norms for the encoding and structuring of information will be supported. Incentives will be provided to the acting parties to complete and extend existing information standards”.

<sup>649</sup> *Zweites Gesetz zur Änderung des Urheberrechtsgesetzes vom 9. Juni 1993, BGBl I* pág. 910.



El artículo que corresponde a la excepción de la descompilación informática es el 69e. En conformidad con la ratio legis de la Comisión y el Parlamento comunitarios, la doctrina alemana mantiene que la finalidad de la descompilación autorizada por la norma es equilibrar el elevado estándar de protección diseñado por la norma comunitaria con el mantenimiento y la garantía de una competencia eficaz en el mercado<sup>650</sup>.

A pesar de la escasez de resoluciones judiciales sobre esta materia en Europa, la jurisprudencia alemana cuenta el mayor número de sentencias que han tratado expresamente la interpretación del artículo 69e y que han hecho referencia a esta excepción legal.

Dos aspectos que tienen en común los casos presentados ante los tribunales civiles, son, en primer lugar, que la excepción de la descompilación ha sido empleada por competidores del titular de los derechos. En segundo lugar, la excepción no se ha utilizado para copiar ni la forma de expresión del programa (el código) ni su estructura, sino para obtener la información necesaria a fin de elaborar un programa interoperable con el descompilado.

Varios conceptos han sido objeto de desarrollo por la jurisprudencia alemana:

En primer lugar, ha proporcionado una definición de descompilación a efectos del derecho de autor. Así, por descompilación ha de entenderse la acción de reconvertir, copiar o editar el código objeto, aunque el resultado final de dicha acción no consista en el código fuente completo del programa analizado<sup>651</sup>.

Respecto a la finalidad de la descompilación, la jurisprudencia alemana ha venido matizado que los requisitos del artículo 69e UrhG no se verán cumplidos cuando el programa sea descompilado y alterado con la finalidad de anular una medida de protección anticopia<sup>652</sup>. En todos los casos el demandado había alegado que la medida

---

<sup>650</sup> M., LEHMANN, “Germany”, en M. LEHMANN, C. TAPPER (eds.), *A Handbook of...*, *op.cit.*, pág. 29.

<sup>651</sup> OLG Düsseldorf, sentencia de 16 de enero de 2001, CR 2001, pág. 371 (372). “*Dekompilierung bezeichnet die Rückübersetzung des Objektformats als Vervielfältigung oder Bearbeitung, wobei allerdings im daraus entstehenden Quellcode nicht mehr vollständig die ursprünglichen Programminformationen enthalten sind*”.

<sup>652</sup> OLG Karlsruhe sentencia de 10 de enero 1996 (*caso Dongle-Umgehung*), ZUM-RD 1997, pág. 340 (341) y OLG Düsseldorf sentencia de 27 de marzo 1997 (*caso Dongle-Umgehung*), CR 1997, pág. 337: “*Die Entdonglierung eines Computerprogramms greift in das Vervielfältigungs- und Umgestaltungsrecht*”.

de protección anticopia ocasionaba problemas técnicos que impedían el normal uso del programa, y debido a ello la descompilación del programa que ejecutaba la medida tecnológica de protección (MTP) y su anulación debía calificarse como un acto de corrección de errores. Esta argumentación fue rechazada porque la alteración del programa, eliminando la MTP, no era una operación indispensable para conseguir la información interoperable.

En la sentencia más reciente, emitida en abril de 2012 por el OLG de Hamburgo, la cuestión de fondo era decidir si los cambios temporales realizados en un programa sin autorización del titular constituían infracción del derecho de autor o podían ampararse en alguna de las excepciones sobre ingeniería inversa que contiene el Título VII de la UrhG. El Tribunal reitera el criterio de los casos *Dongle-Umgehung*. Por tanto, la finalidad de la excepción de la descompilación no se ve satisfecha cuando la descompilación no se realice para alcanzar interoperabilidad con un programa creado de forma independiente, sino únicamente para lograr la interoperabilidad con un programa que no es funcionalmente independiente del programa descompilado<sup>653</sup>.

Acerca del destino de la información interoperable, representado por el requisito de que la consecución de interoperabilidad con un programa creado de forma independiente, la jurisprudencia alemana ha matizado que el objeto de la excepción ha de entenderse como la creación de un programa nuevo que sea competitivo con el programa descompilado pero interoperable con un tercer programa que no ha de ser necesariamente el descompilado, sino que puede ser un tercer programa<sup>654</sup>.

En relación al carácter imperativo de la excepción, establecida en el artículo 8, párrafo segundo de la Directiva y que se corresponde con el artículo 69g.2) UrhG, la jurisprudencia alemana ha indicado que al ser su contenido de carácter imperativo,

---

*des Urhebers ein und ist nicht von § 69e I und II UrhG gedeckt*"; OLG Hamburg sentencia de 23 de abril de 2012, BeckRS 2012, pág. 17890.

<sup>653</sup> OLG Hamburg sentencia de 23 de abril de 2012, BeckRS 2012, pág. 17890; *“Da die angegriffene Software keine eigenständige Funktion hat, geht es hier nicht lediglich um die Herstellung der Interoperabilität der „A. Replay PSP“-Software mit dem Betriebssystem der PSP, wie die Beklagte in erster Instanz geltend gemacht hat”*.

<sup>654</sup> OLG Düsseldorf, sentencia de 16 de enero de 2001, CR 2001, pág. 371 (371) *“Ziel einer zulässigen Dekompilierung kann angesichts der Gesetzesfassung auch die Herstellung eines Konkurrenzprogramms sein, das nicht mit dem dekompierten Programm interoperabel ist, sondern mit einem dritten Programm, mit dem auch das dekompierte Programm kompatibel ist”*.

podría tener cabida la aplicación de la normativa de competencia desleal<sup>655</sup>. Esta interpretación tiene especial relevancia en el caso de que la información de las interfaces contenida en código fuente se haya protegido como secreto empresarial, porque podría entrar en juego el artículo 17(2) UWG, que prohíbe la obtención de información en forma deshonesta por cualquier persona<sup>656</sup>. El Tribunal rechazó esta posibilidad por ser la sección 69e UrhG una norma de derecho especial, que tiene prioridad de aplicación sobre las normas sobre competencia desleal.

Por lo tanto el requisito de adquisición deshonesto del art. 17(2) UWG no se cumpliría si la descompilación puede ampararse en la excepción del artículo 69e. Aun cuando este último requisito no se cumpliera, estaríamos en todo caso ante una infracción de derecho de autor, no de competencia desleal.

### 3.2. Francia.

La transposición de la Directiva al derecho francés tuvo lugar a través de la Ley Nº 94-361 de 10 de mayo de 1994<sup>657</sup>. A diferencia tanto del derecho alemán como del derecho inglés y del derecho español, que introdujeron un Título específico para las disposiciones sobre programas de ordenador, el legislador francés no lo incluyó por la razón de que dicho proceder se consideraba conforme con el criterio jurisprudencial francés<sup>658</sup>.

El artículo 6 de la Directiva, se reflejó en el artículo L. 122-6-1 de manera textual, para que el derecho francés “reflejase de manera fidedigna la posición común

---

<sup>655</sup> LG Mannheim, sentencia de 20 de enero de 1995, NJW 1995, pág. 3322 (3323).

<sup>656</sup> § 17.2 UWG: *Ebenso wird bestraft, wer zu Zwecken des Wettbewerbs, aus Eigennutz, zugunsten eines Dritten oder in der Absicht, dem Inhaber des Unternehmens Schaden zuzufügen, 1) sich ein Geschäfts- oder Betriebsgeheimnis durch a) Anwendung technischer Mittel, b) Herstellung einer verkörperten Wiedergabe des Geheimnisses oder c) Wegnahme einer Sache, in der das Geheimnis verkörpert ist, unbefugt verschafft oder sichert oder 2) ein Geschäfts- oder Betriebsgeheimnis, das er durch eine der in Absatz 1 bezeichneten Mitteilungen oder durch eine eigene oder fremde Handlung nach Nummer 1 erlangt oder sich sonst unbefugt verschafft oder gesichert hat, unbefugt verwertet oder jemandem mitteilt.* Comentarios sobre el art. 17(2) UWG en H., KÖHLER, *Gesetz gegen den unlauteren Wettbewerb*, München, 29 ed., Beck, 2011, §17 pár. 37.

<sup>657</sup> Loi n° 94-361 du 10 mai 1994, portant mise en oeuvre de la directive (C.E.E.) n° 91-250 du Conseil des communautés européennes en date du 14 mai 1991 concernant la protection juridique des programmes d'ordinateur et modifiant le code de la propriété intellectuelle. JOF 11/05/1994.

<sup>658</sup> Vid. A, LUCAS, L., PANHALEUX, L., “France”, en M. LEHMANN, C. TAPPER (eds.) *A Handbook of...*, *op.cit.*, pág.11.

que refleja un compromiso adoptado por profesionales”, tal como indica el preámbulo de la Ley de 10 de mayo de 1994.

Respecto a la interpretación de los tribunales sobre este precepto, la jurisprudencia francesa cuenta con varios casos, algunos muy recientes que han examinado el contenido sustantivo de esta excepción. A diferencia de los tribunales alemanes, las situaciones conflictuales no versan únicamente sobre elusión de medidas tecnológicas de protección, sino que la excepción es utilizada por un usuario que es competidor del titular del derecho de autor sobre el programa descompilado, para obtener la información interoperable a fin de desarrollar un programa independiente.

El primer caso que trató la excepción de la descompilación por razones de interoperabilidad en Francia data de 1997, en la Corte de Apelaciones de París. En el caso de *Nomai v. IOMEGA*<sup>659</sup>, el demandante, IOMEGA, había acusado a Nomai de copiar su programa a través de un procedimiento de descompilación. El Tribunal de Apelación sostuvo que Nomai tenía derecho a invocar la excepción legal en materia de interoperabilidad, porque “la utilización de ingeniería inversa para lograr copiar el programa "Oscar 100" era una necesidad técnica para garantizar la interoperabilidad que no infringía ningún derecho protegido IOMEGA<sup>660</sup>”. Llegó incluso a afirmar que “la autorización del titular del programa, según el artículo L 122-6 1-IV de la Ley de Propiedad Intelectual no resulta necesaria cuando la reproducción del código del programa es indispensable para obtener la información necesaria para la interoperabilidad de un programa creado de forma independiente con otros programas<sup>661</sup>”.

Sobre la prohibición de la descompilación de programas con carácter general, el Tribunal señaló que esta prohibición se aplica al software como tal y no a partes que son accesorias de un elemento más complejo y que además constituyen un puerta para la comercialización, de un nuevo producto, como en este caso, una unidad de disco compatible con otra existente en el mercado.

---

<sup>659</sup> Asunto *Nomai v. IOMEGA*, sentencia de 12 de diciembre de 1997, Corte de Apelaciones de París (Cour d'Appel), op.cit. Vid. supra, capítulo 3.2.4.C.1.

<sup>660</sup> *Ibidem*, pág. 3: “*le recours par NOMAI à une « ingénierie à rebours » pour parvenir à copier le logiciel « Oscar 100 » était une nécessité technique pour assurer une interopérabilité et qu'elle ne portait pas atteinte à aucun droit protégé de IOMEGA.*”

<sup>661</sup> *Ibidem*, pág. 4: “*l'autorisation de l'auteur d'un logiciel n'a pas lieu d'être demandée lorsque la reproduction du code du logiciel est indispensable pour obtenir les informations nécessaires à l'interopérabilité d'un logiciel créé de façon indépendante avec d'autres logiciels.*”

Después de una sentencia que puede considerarse ejemplar, hasta el año 2011, no se dan más casos en el derecho francés. El primer caso procede nuevamente de la Corte de Apelación de París, de fecha 26 de septiembre<sup>662</sup>, y el segundo, de la Corte de Casación el 20 de octubre<sup>663</sup>.

La sentencia de 26 de septiembre de 2011 trata la descompilación de una medida tecnológica de protección de una consola de videojuegos por razones de interoperabilidad, y el caso de la Corte de Casación, sobre la elaboración de un nuevo programa de ordenador creado por ex-empleados de la empresa fabricante del software descompilado, con el cual es interoperable y además compite en el mercado.

En el primer caso, Nintendo, para evitar la piratería sobre sus juegos, había instalado en su consola una medida tecnológica de protección. Dicha MTP consiste en un software denominado que no permite ejecutar videojuegos pirateados en la máquina. Uno de los codemandados, distribuidor autorizado de videojuegos de Nintendo, vendía asimismo en su sitio web otro programa (*linkers*) que permitía eludir la MTP de la consola. El demandante argumentó que para eludir la MTP, el *linker* tenía que reproducir el código fuente de sus videojuegos, lo cual constituía una infracción de su derecho de autor sobre el programa.

La Corte de Apelación, condena a los creadores del programa antiMTP, por tratarse de un programa cuyo único fin era dañar las MTP en el sentido del artículo L.331-5 del CCI, permitiendo el funcionamiento de copias infractoras en los dispositivos del demandante. Sin embargo lo interesante del fallo es el análisis que realiza el tribunal de la excepción de la descompilación, cuando examina la efectiva reproducción del código fuente del programa protegido.

---

<sup>662</sup> Asunto Sarl AAkro Pure Tronic et a. v. Nintendo, sentencia de 26 de septiembre de 2011, Corte de Apelaciones de París (Cour d'appel), Pôle 5, Chambre 12. Para un comentario sobre el caso desde su inicio, en 2007 vid. A.S. LAMPE, S. LERICHE, "Contrefaçon de logiciel, exception de décompilation et contournement de Mesures Techniques de Protection (MTP): l'affaire Nintendo, ou la difficulté de la preuve au secours de distributeurs de *linkers*", Lamy droit de l'immatériel 57, 2010, pág. 22.

<sup>663</sup> Asunto La société Fiducial v. la société Développement professionnel spécialisé informatique (DPSI) ; et autres, sentencia de 20 de octubre de 2011 de la Corte de Casación (Cour de cassation - Première chambre civile), n° 975, (10-14.069).

El Tribunal asume que los creadores del programa antiMTP han tenido acceso al código fuente del programa a través de un proceso de descompilación, para poder anularlo.

Dicha descompilación, podría haberse amparado en la excepción configurada en el art. L 122-6-1 CPI, si su finalidad fuese alcanzar la interoperabilidad. El Tribunal examina si se cumplen los requisitos de la norma:

1º Los demandados eran usuarios legítimos del software porque, afirma la Corte, cualquier usuario legítimo de una consola lo es asimismo del software que contiene<sup>664</sup>.

2º La descompilación era indispensable porque el demandante ni publica ni divulga la información interoperable de sus consolas<sup>665</sup>.

Pero la finalidad del proceso de ingeniería inversa, no era la creación de un programa que además de interoperable con el descompilado, tuviese alguna función independiente. El objetivo real de la descompilación era permitir una infracción del derecho del autor, consistente en la elusión de la MTP que permitía el uso de copias falsificadas. Para tal fin, el creador del programa nuevo e independiente necesita acceder a la información interoperable, pero, tal como indica el art. L. 122-6-1 V, que se corresponde con la Regla de los Tres Pasos, la excepción no puede interpretarse de manera que permita que su aplicación perjudique de forma injustificada los intereses legítimos del titular.

La conclusión a que llega el tribunal francés, por un lado, sigue la línea mantenida por la jurisprudencia alemana en cuanto a que el nuevo programa ha de tener funciones independientes y distintas del programa descompilado.

Sentado esto, da un paso más en cuanto a la finalidad de la información interoperable: si el acceso a esta información tiene como objeto facilitar una infracción

---

<sup>664</sup> “que dans ce cadre il ne peut être opposé la condition “d'utilisateur légitime”, aux prévenus, ces derniers n'ayant pas eux-mêmes procédé aux actions de reproduction et de traduction du logiciel ; que tout utilisateur d'une console de jeux vidéo est un utilisateur légitime du logiciel embarqué dans ce matériel.”. Dicho de otro modo, quien puede lo más, puede lo menos.

<sup>665</sup> “que les dispositions du présent article n'impose en aucune façon aux utilisateurs d'un logiciel de demander à l'auteur l'accès aux informations préalablement à l'acte de décompilation ou reproduction, il interdit seulement de procéder à de tels actes si les informations sont déjà disponibles, ce qui est précisément l'inverse, puisque Nintendo ne publiera jamais ni ne communiquera aucune information permettant de rendre un linker interoperable avec l'une de ses consoles.”

del derecho de autor, aunque se cumplan todos los demás requisitos de la excepción, no será posible ampararse en ella por ir en contra de la naturaleza de la norma<sup>666</sup>.

La segunda sentencia, procedente de la Corte de Casación, plantea una situación de descompilación por un competidor. En este caso, el demandado, antiguo empleado de la empresa del programa del demandante (*Fiducial Informatique*), había creado un programa de ordenador independiente interoperable con el del demandante y que competía con él en el mercado. Un de los alegatos principales de los demandantes consistía en que las cláusulas de la licencia de uso de su software no autorizaban la realización de operaciones de migración de datos que permitían acceder al código fuente.

Aplicando la cláusula imperativa del párrafo final del art. L.122-6-1 CPI, que se corresponde con el art. 8, párrafo segundo de la Directiva, la Corte de Casación declara la nulidad de dichas estipulaciones y seguidamente afirma que las operaciones de migración de datos técnicos del código fuente llevada a cabo por el creador de un programa, autorizado a tal efecto por los titulares de la licencia de uso de otro para recuperar archivos, y destinados a la creación de un programa nuevo e interoperable con el primero que cumpla con los demás requisitos de la excepción, no constituye infracción del derecho de autor<sup>667</sup>.

Las clave de la declaración de la Corte no se encuentran en el término “datos técnicos” sino en la referencia a la licencia de uso. Los datos técnicos son los que contienen las interfaces del programa. En consonancia con el criterio mantenido por el

---

<sup>666</sup> “*que les cartes linkers, ont pour unique fonction de rendre des cartes mémoires au format micro-DS lisibles et compatibles avec la console Nintendo, en d’autres termes la seule fonction des linkers est d’être compatible avec la console Nintendo DS de façon à pouvoir fonctionner avec elle, lire des fichiers numériques et exécuter des programmes sur cette console, autres que les jeux d’origine Nintendo. que leur protection est assurée par l’article L 122-6 de ce même code qui interdit de procéder sans autorisation de son auteur à la reproduction permanente ou provisoire d’un logiciel en tout ou partie par tout moyen et sous toute forme, ainsi qu’à la “traduction, l’adaptation, l’arrangement ou toute autre modification d’un logiciel et la reproduction du logiciel en résultant ....” ; que la violation de ce texte est pénalement sanctionnée aux termes de l’article L 335-3 de ce code qui dispose : “est également un délit de contrefaçon toute reproduction, représentation ou diffusion par quelque moyen que ce soit, d’une œuvre de l’esprit réalisée en violation des droits des auteurs tels que définis et réglementés par la loi. Est également un délit de contrefaçon la violation de l’un des droits de l’auteur d’un logiciel, tels qu’ils sont définis à l’article L 122-6.”*

<sup>667</sup> “(...) les opérations de migrations de données, réalisées par M. X... et la société Alphapi, habilités à cette fin par les huissiers de justice titulaires de la licence d'utilisation du logiciel "H. Open", pour récupérer les fichiers de ce programme, s'inscrivaient dans les strictes nécessités de l'interopérabilité autorisée par l'article L. 122-6-1 IV du code de la propriété intellectuelle qui prévoit la nullité de toute stipulation contraire, et a, par ce seul motif, légalement justifié sa décision”

TJUE desde el asunto C-393/09, los elementos técnicos, utilitarios o funcionales de un programa no forman parte de la expresión protegida por el derecho de autor, por lo tanto su uso es libre. Sin embargo, el tribunal entiende que el acceso a esos datos técnicos no constituye infracción porque la licencia de uso autorizaba al demandado y descompilador del programa, a recuperar archivos. De no haber estado autorizado por el titular a través de la licencia, los actos de migración, a pesar de ser sobre datos técnicos, constituirían una infracción. Por lo tanto, esos datos técnicos sí estarían protegidos pero no por el derecho de autor sino en virtud del derecho contractual. Por tanto, aún habiendo concurrido los demás requisitos de la excepción de la descompilación no podría haber sido aplicada.

### 3.3. Reino Unido.

La transposición de la Directiva tuvo lugar en Reino Unido a través de la Copyright (Computer Programs) Regulations 1992, modificativa de la Parte I (copyright) y de la Parte VII (miscellaneous and general) de la Copyright, Designs and Patents Act 1988<sup>668</sup>.

En relación con la descompilación, la transposición del artículo 6 en el ordenamiento británico es la que más se diferencia de los otros ordenamientos objeto de este estudio.

La primera diferencia es que la excepción fue implementada por dos vías:

Por un lado, a través de la modificación del artículo 29, que regula la excepción general de *fair dealing*. Se excluyó expresamente de su ámbito la conversión de un programa de ordenador expresado en un lenguaje de bajo nivel en una versión expresada en un lenguaje de alto nivel<sup>669</sup> y la copia que casualmente pueda producirse en el transcurso de la conversión<sup>670</sup>. Por otro lado, se introdujo el artículo 50B, bajo el título “*decompilation*”.

---

<sup>668</sup> Copyright, Designs and Patents Act 1988, No. 3233.

<sup>669</sup> “*It is not fair dealing (a) to convert a computer program expressed in a low level language into a version expressed in a higher level language*”.

<sup>670</sup> “*(It is not fair dealing) (b) incidentally in the course of so converting a program to copy it*”.



La segunda diferencia reside precisamente en el texto del artículo 50B, que no transpone textualmente, ni casi textualmente el contenido del artículo 6 de la Directiva.

El Informe de la Comisión del año 2000 señalaba varias irregularidades en la transposición de términos concretos. A título ejemplificativo, cabe citar dos de ellas:

El legislador británico había empleado el término “usuario legal” para referirse a todas las personas autorizadas por la excepción, de modo que el artículo 50B no comprendería ni “cualquier persona autorizada en nombre del licenciataria”, ni “cualquier persona facultada para utilizar una copia del programa”.

Otra diferencia sorprendente es que la norma británica define la descompilación (*to convert a computer program expressed in a low level language into a version expressed in a higher level language*). Esta definición no encuentra correspondencia en la Directiva, y podría haber sido utilizada para limitar el ámbito de aplicación de la propia Directiva, a la que no interesa la jerarquía de lenguajes, sino los actos de reproducción del código y la traducción de su forma<sup>671</sup>.

Sin embargo, hasta el momento no ha habido noticia de ningún caso sobre descompilación de un programa. En cambio, la jurisprudencia inglesa sí ha tenido ocasión de pronunciarse sobre la otra excepción de ingeniería inversa que configura la Directiva en el artículo 5.3, a pesar de que transposición al ordenamiento británico no tuvo lugar hasta 2003<sup>672</sup>. Aunque el análisis de dicha jurisprudencia ha tenido lugar en apartados anteriores a los que remitimos<sup>673</sup>, baste aquí recordar las razones de su interés en relación a la interpretación de los requisitos del artículo 6:

1ºLa jurisprudencia sobre este artículo ha sido la primera en aportar claridad sobre el concepto de interfaz con carácter general, el cual constituye la piedra angular para determinar cuál es la información interoperable del programa, la única a la que la excepción autoriza acceder.

2ºComo se ha indicado al examinar el concepto de interfaces en la Directiva, las directrices sentadas por el Tribunal Supremo inglés parecen haber tenido una

---

<sup>671</sup> Informe de la Comisión Año 2000, p. 10; C., TAPPER, “United Kingdom”, en M. LEHMANN, C. TAPPER (eds.) *A Handbook of...*, *op.cit.*, pág. 8.

<sup>672</sup> Artículo 50BA añadido el 31.10.2003 a través de The Copyright and Related Rights Regulations 2003 (S.I. 2003/2498), reg. 15(1).

<sup>673</sup> Vid. *supra*, apartado. 2.3.2.C.2)

importante influencia en el reciente criterio establecido por el TJUE en las sentencias de 2009 y 2012 respectivamente, de que dichos elementos no forman parte del objeto de protección de la Directiva.

#### 3.4. España.

El derecho español implementó la Directiva en su ordenamiento a través de la Ley 16/1993 de 23 de diciembre, de incorporación al Derecho español de la Directiva 91/250/CEE, de 14 de mayo de 1991, sobre protección jurídica de los programas de ordenador (en adelante, LIPRO)<sup>674</sup>. Los artículos 95 a 104 TRLPI que constituyen el Título VII del Libro I, regulan el derecho de autor sobre los programas de ordenador.

Si bien el artículo 6 de la Directiva se transpuso casi textualmente a la Ley de Propiedad Intelectual, el legislador español optó por incluir todas las excepciones a los derechos de explotación que contiene la Directiva en un único artículo, el 100. Es su apartado 5 el que se corresponde con la excepción de la descompilación.

Para empezar, cabe hacer alguna consideración doctrinal acerca del modo en que el legislador español incorporó la excepción de la descompilación al texto del TRLPI.

En primer lugar, el hecho de haber cambiado el término “licenciatario” de la Directiva por el de “usuario legítimo” no es problemático porque, a diferencia del legislador inglés, el español mantuvo la otras dos enumeraciones de la Directiva (“cualquier otra persona facultada para utilizar una copia del programa” y “en su nombre (del usuario legítimo) por parte de una persona debidamente autorizada”)<sup>675</sup>.

Finalmente, respecto del inciso final del artículo 8 de la Directiva, no hay justificación alguna para que el legislador nacional no lo hubiese recogido ni en la Ley 16/1993 ni en el TRLPI. Si bien es cierto que la Directiva no deja lugar a duda sobre el carácter imperativo de la excepción dispuesta en el artículo 6, lo cierto es que la

---

<sup>674</sup> BOE núm. 307, de 24 de diciembre 1993, pág. 36816. Un comentario sobre la incorporación de la Directiva al derecho español en. J.A., GOMEZ SEGADE, *Tecnología y Derecho...*, *op.cit.*, pág. 848.

<sup>675</sup> Art. 100.1 LPI.

cláusula de cierre del Título VII de la TRLPI, no hace mención al derecho contractual<sup>676</sup>.

Por desgracia no existe jurisprudencia interpretativa sobre esta excepción que ayude a constatar el estado de la cuestión de la descompilación informática a efectos de interoperabilidad en España. Sin embargo, al igual que en el Reino Unido, los jueces españoles sí han tenido ocasión de pronunciarse acerca de algún aspecto estrechamente relacionado con la excepción, como es el concepto de interfaz.

En dichas ocasiones, las audiencias provinciales examinaron el objeto de protección del derecho de autor, llegando a resultados distintos al valorar si la interfaz gráfica de usuario del programa formaba parte de dicho objeto o no.

La resolución más antigua, de la Audiencia Provincial de Cádiz de 2007<sup>677</sup>, es un claro ejemplo de la tendencia extensiva de protección del programa más allá de sus elementos literales. El juez parte de que el art. 96 LPI ofrece un concepto de programa de ordenador en un doble sentido. Afirma que el legislador da una definición positiva en el apartado número uno, al establecer unos requisitos mínimos a la hora de considerar una creación intelectual como un programa de ordenador, y a continuación define el objeto de protección en sentido negativo, al fijar una serie de exclusiones. Dicho esto, sin embargo, considera que si el legislador hubiera querido negar la protección a las ideas y principios que sirven de fundamento a las interfaces así lo hubiera hecho de modo expreso y directo. Añade que, por otra parte, a través de una interpretación gramatical del precepto no se puede llegar a otra conclusión.

En cuanto a qué ha de entenderse por interfaces de usuario, las define como “un conjunto de objetos, herramientas y representaciones visuales que sirven para gestionar la comunicación entre usuario y aplicación informática, (que) vienen a constituir la parte externa o visible del programa”.

Por lo tanto, por ser la interfaz de usuario la parte visible del programa, constituye uno de los elementos de la aplicación informática, por lo que la protección de

---

<sup>676</sup> Art. 104 LPI: “Lo dispuesto en el presente Título se entenderá sin perjuicio de cualesquiera otras disposiciones legales tales como las relativas a los derechos de patente, marcas, competencia desleal, secretos comerciales, protección de productos semiconductores o derecho de obligaciones”.

<sup>677</sup> AP Cádiz, Sección 5ª, sentencia núm. 30/2007 de 19 de enero, AC 2007\1018.

la misma por la nominada Ley es innegable en cuanto que contiene elementos propios del derecho de autor<sup>678</sup>.

En sentido contrario, la Audiencia Provincial de Girona en sentencia de 2010<sup>679</sup>, define el objeto de protección que la Ley de Propiedad Intelectual describe en el art. 96 declarando que ésta protege no la idea sino la expresión de la misma.

El juez aplica el principio de la dicotomía entre expresión e idea, acercándose al reciente criterio establecido por el TJUE en la sentencia de mayo de 2012, y sitúa la frontera entre una y otra “en la finalidad última que la obra pretender realizar, (la cual) en el caso del programa, será la idea de la obra, mientras que la forma de conseguir la finalidad perseguida, el medio específico elegido en el caso concreto, habrá que considerarlo como expresión de una idea, y ese medio, de entre los varios posibles, será el objeto de la propiedad intelectual”. Tras este análisis, concluye el juez que “el objeto de protección no es lo que se pretende sino el cómo se consigue: la forma de plasmar la idea, su ordenación, configuración y estructura”.

Quizás en un futuro no muy lejano, el Tribunal Supremo tenga ocasión de pronunciarse sobre esta cuestión, pero por el momento, parece que el criterio de la Audiencia Provincial de Girona está en sintonía con el reciente criterio de originalidad establecido por el TJUE en los asuntos C-393/09, C-604/10, y C-406/10.

#### **4. Conclusión de capítulo.**

La aprobación de la Directiva 91/250/CEE de 14 de mayo sobre protección jurídica de los programas de ordenador marcó un hito histórico por dos razones: Fue el primer instrumento normativo que a nivel comunitario declaró la protección de los programas de ordenador como obras literarias en el marco del Convenio de Berna. Esta declaración fue seguida posteriormente en dos importantes tratados, ADPICS en 1994 y WCT en 1996, internacionalizando esta vía de protección. Además, por primera vez a nivel mundial, una norma jurídica establecía una excepción específica a los derechos del titular sobre el programa con el fin de facilitar la interoperabilidad informática y así garantizar la competitividad del mercado.

---

<sup>678</sup> Ibidem, FJ 4

<sup>679</sup> AP Girona (Sección 1ª), sentencia núm. 86/2010 de 3 marzo. AC 2010\940.

Sin embargo, si tanto en Estados Unidos como en Europa, la finalidad de proteger los programas de ordenador a través del derecho de autor consiste en equilibrar los intereses de protección del autor al tiempo que se facilita la libre competencia en el sector a través de la garantía de la interoperabilidad, el derecho de exclusiva del autor no debiera alcanzar las interfaces del programa, ni ampliar el objeto de protección del titular a través de una excepción legal, que legitima la aparición de situaciones de abuso del derecho de exclusiva.

Si la información interoperable juega un papel clave para la industria informática, las normas que regulen el acceso a ella y su protección a través de un derecho de propiedad especial son de suma importancia. Los derechos de propiedad intelectual no deben proporcionar protección a elementos como las interfaces, porque estos elementos son métodos de operación y están excluidos de la protección del derecho de autor a nivel internacional.

Si no forman parte del objeto de protección, el legislador no puede consentir que se protejan a través de una excepción legal al derecho de exclusiva, porque esto constituye una perversión del propio sistema de protección. Pero el análisis del artículo 6 de la Directiva, ha demostrado que las estrictas condiciones impuestas al acceso y uso de la información interoperable, otorgan una protección *de facto* sobre estos elementos del programa. No favorecen tampoco la interoperabilidad con carácter general, porque al sólo permitir la descompilación para alcanzar la interoperabilidad con un programa creado de forma independiente, la norma no deja lugar a la descompilación cuya finalidad sea detectar errores que no permiten la interoperabilidad, o la descompilación a efectos de investigación para mejorar las condiciones de interoperabilidad entre programas o programas y dispositivos.

Aunque la escasa jurisprudencia sobre esta norma deja claro que la descompilación no puede ser utilizada para crear un nuevo programa que tenga como fin facilitar la infracción del derecho de autor, pero sí para crear un nuevo programa que compita con el descompilado, funcionalmente similar pero diferente en su forma de expresión a éste, sigue sin dejar hueco para la investigación.

Si a esto unimos que la concepción de un derecho de reproducción de la Directiva alcanza la realización de copias temporales, aunque sean de carácter técnico,

el artículo 6 resulta ser más un obstáculo a efectos de desarrollos interoperables realizados mediante descompilación, que una solución.

Si el verdadero objetivo del Plan de Acción de la Comisión<sup>680</sup> es que pueda alcanzarse la interoperabilidad informática en general, no sólo entre programas sino entre toda clase de dispositivos electrónicos con el mayor número de programas, entonces entre las medidas de acción debe incluirse la revisión del artículo 6 que establece la excepción de descompilación a efectos de interoperabilidad en la Directiva de Software.

---

<sup>680</sup>«Comunicación de la Comisión al Parlamento Europeo, al Consejo, al Comité Económico y Social Europeo y al Comité de las Regiones — Internet de los objetos — Un plan de acción para Europa» [COM(2009) 278 final]

## **Capítulo 4**

**La relación de la Directiva sobre protección jurídica de los programas de ordenador con otras ramas del ordenamiento jurídico en materia de interoperabilidad: Más razones para la reforma del Artículo 6.**





## **1. La relación entre la Directiva y el Derecho de defensa de la competencia en materia de interoperabilidad.**

### *1.1. Políticas fundamentales de protección de los derechos de Propiedad Industrial e Intelectual.*

En los países de tradición jurídica continental, la clásica división entre derechos de Propiedad Industrial y derechos de Propiedad Intelectual se debe a que sus objetivos de protección no son exactamente idénticos. Por un lado la denominada “propiedad industrial”, (patentes, marcas, diseños, indicaciones geográficas y secretos comerciales) el fin es recompensar el esfuerzo y la inversión del titular. Por otro lado, la “propiedad intelectual” o los derechos de autor, tienen como fin la protección integral del autor.

El derecho de autor en materia de programas de ordenador, tanto en Europa como en Estados Unidos, no sólo tiene como interés principal garantizar la protección del autor de una obra original, para lo cual le otorga un conjunto de facultades exclusivas como recompensa a su labor y de este modo, sino también beneficiar al interés general al fomentar la creación de nuevos programas y la innovación de la industria informática<sup>681</sup>.

Dos son básicamente las razones para conceder derechos de propiedad exclusiva: El primer fundamento es de tipo moral y el segundo de tipo económico<sup>682</sup>.

Los conceptos claves son motivar y promover.

La idea implícita en la concesión de un derecho de propiedad exclusiva es motivar. En el caso de la patente, se busca incentivar al inventor, para que cuando su idea cobre vida, pueda obtener un beneficio por su esfuerzo. En el caso de las marcas, se incentiva al titular del signo distintivo a invertir en la reputación de su producto o servicio, para que los consumidores lo identifiquen por esa concreta marca.

---

<sup>681</sup> Vid. supra, capítulo 3.2.1 sobre la Directiva de protección de programas de ordenador, y capítulo 2.1 sobre Estados Unidos.

<sup>682</sup> Vid. entre otros: World Intellectual Property Organization, *Intellectual Property Handbook: Policy, Law and Use*, WIPO Pub. N° 489(E), 2ª ed., 2008, pág. 163; disponible en: <http://www.wipo.int/export/sites/www/about-ip/en/iprm/pdf/ch1.pdf> (última consulta 1 de junio, 2013); Comunicación de la Comisión al Parlamento Europeo, al Consejo y al Comité Económico y Social Europeo: Derechos de propiedad industrial: una estrategia para Europa COM(2008) 465 final (no publicada en DO), pág. 1.

La promoción de la creatividad, sirve para aumentar el acervo de sabiduría, y con ello se estimula el comercio justo y la contribución de todos al crecimiento social y económico<sup>683</sup>. En términos económicos, el objetivo de este derecho es la creación y explotación comercial de un monopolio concedido por el Estado<sup>684</sup>.

Estas dos razones permiten afirmar que la concesión de estos derechos está justificada por la función que cumplen<sup>685</sup>. Es decir, la razón de existir y su ámbito de protección deben ser analizados en relación a su función. Si el derecho de exclusiva no cumple con su propósito, habrá un elevado riesgo de sobreprotección o de infraprotección. En una situación como esta, el derecho de propiedad intelectual traerá consigo más desventajas que ventajas, junto con un elevado número de críticas que requerirán o bien su limitación o en última instancia, su supresión.

Pero simultáneamente, los derechos de Propiedad Industrial e Intelectual constituyen una excepción, quizás la principal, a la libertad de empresa y a la libre competencia. Esta limitación a la libertad de empresa viene determinada por la propia naturaleza de estos derechos, que otorgan a su titular un derecho exclusivo y excluyente sobre su objeto, de forma que a sus competidores no les estará permitido fabricar o vender y, en general, realizar actos de contenido económico sobre los objetos protegidos por estos derechos<sup>686</sup>.

Percibidos *ex post*, el incentivo a innovar de los derechos de Propiedad Industrial e Intelectual puede actuar como una barrera de entrada. En palabras de Joseph Schumpeter: “the competition which counts [is] competition from the new commodity, the new technology, the new source of supply, the new type of organization . (...) competition which commands a decisive cost or quality advantage and which strikes not at the margins of the profits and the outputs of the existing firms but at their foundations and their very lives.”<sup>687</sup> Es decir, si bien existe un equilibrio entre los incentivos a

---

<sup>683</sup> Sirva a título de ejemplo las declaraciones del Director General Adjunto Stoll, de la Dirección General del Mercado Interior (Comisión Europea) en The Pan-European IP Summit, celebrado en Brusela el 2 de diciembre de 2004: “*In the context of the Lisbon objectives, there is a need for strong IP protection to foster innovation in Europe.*”

<sup>684</sup> F., LÉVÉQUE, *The Economics of ...*, *op.cit.*, pág. 85. Vid.

<sup>685</sup> Vid. C., GEIGER, “Fundamental Rights, ...”, *op.cit.*, pág. 270.

<sup>686</sup> J.M., OTERO LASTRES en C., FERNÁNDEZ NÓVOA, J.M., OTERO LASTRES, M. BOTANA AGRA, *Manual de la Propiedad Industrial*, Madrid, 2ª ed., Marcial Pons, 2013, pág. 62 y ss.

<sup>687</sup> J.A. SCHUMPETER, *Capitalism, Socialism and Democracy*, Londres, George, Allen & Unwin Ltd 1943, pág. 84. (La competición que cuenta [es] la competencia de la nueva mercancía, la nueva tecnología, la nueva fuente de alimentación, el nuevo tipo de organización. (...) la competencia que

innovar y la concesión del derecho exclusivo, la competencia en la innovación suele ser más importante que la competencia con un rival proporciona el mismo producto de la misma forma al mercado.

Pero, a menos que creamos que los gobiernos pueden juzgar de antemano qué tipo de investigación y desarrollo tiene más probabilidades de fructificar, y otorgar becas de investigación de manera eficiente, hemos de dejar que el mercado fomente la inversión en innovación. La única manera eficaz hasta ahora conocida es la concesión de derechos de propiedad exclusivos por un tiempo limitado.

### *1.2. Políticas fundamentales de protección del Derecho de defensa de la competencia.*

La competencia ha sido definida como “el esfuerzo de las empresas por obtener una ventaja en el mercado mediante la mejora de la oferta dirigida a la clientela”<sup>688</sup>. El Derecho de defensa de la competencia constituye un sector normativo cuya finalidad de protección es garantizar el funcionamiento de los mercados basados en la libre competencia<sup>689</sup>. Por tanto, los Estados con sistemas económicos basados en los principios de libre competencia consideran la defensa de este principio como una pieza clave de sus sistemas jurídicos, ya que son muchos los beneficios que conlleva para el correcto funcionamiento de la economía, impulsando a su vez el desarrollo económico y la innovación.

En un mercado basado en la libre competencia, los consumidores buscan adquirir los productos que mejor se ajusten a sus necesidades, de la mejor calidad y al menor precio. Los proveedores, tratan de atender las demandas de los primeros,

---

domina un coste decisivo o ventaja de la calidad y que no incide en los márgenes de los beneficios y los resultados de las empresas existentes, sino a sus fundaciones y sus propias vidas).

<sup>688</sup> F., GARCÍA CACHAFEIRO, “Fundamentos del derecho de libre competencia” en P., BELLO MARTÍN-CRESPO, F., HERNÁNDEZ RODRÍGUEZ, *Derecho de la Libre Competencia Comunitario y Español*, Navarra, Thomson Aranzadi, 2009, pág. 37.

<sup>689</sup> Vid. artículo 119 del TFUE: “1. Para alcanzar los fines enunciados en el artículo 3 del Tratado de la Unión Europea, la acción de los Estados miembros y de la Unión incluirá, en las condiciones previstas en los Tratados, la adopción de una política económica que se basará en la estrecha coordinación de las políticas económicas de los Estados miembros, en el mercado interior y en la definición de objetivos comunes, y que se llevará a cabo de conformidad con el respeto al principio de una economía de mercado abierta y de libre competencia.” (El subrayado es nuestro).

ofreciendo los productos que los consumidores desean. Es decir, no es el estado sino el consumidor quien conduce u orienta la producción<sup>690</sup>.

Libre competencia también significa también el derecho a emprender cualquier iniciativa empresarial. Pero esta posibilidad puede verse coartada por la existencia de un derecho, que otorga un monopolio legal (que no económico). Es decir, desde el punto de vista del Derecho de defensa de la competencia, los derechos de propiedad exclusiva puede entenderse como una forma de reducir la competencia<sup>691</sup>.

Un ejemplo de esta clase puede observarse a través de la protección conferida a los programas de ordenador en relación con nuevos programas compatibles e interoperables. Si al titular del derecho de autor sobre el programa ordenador se le permite utilizar su *ius prohibendi* para evitar la ingeniería inversa sobre su obra, controlando cualquier tipo de reproducción de su programa, se le está legitimando para bloquear el acceso a nuevos competidores, porque puede evitar que otros diseñen programas y dispositivos que puedan funcionar con su software. Ciertamente es que para llegar a este extremo, el programa en cuestión habrá alcanzado popularidad considerable en el mercado. Nada hay que objetar cuando es el éxito de su producto el que permite al titular del derecho de exclusiva, colocarse en una situación de monopolio económico en el mercado. Distinto es que el titular del derecho se sirva de él para impedir, de manera consciente, la entrada de nuevos competidores en el mercado.

Situaciones de estas características se ha venido corrigiendo a través del derecho de la competencia y, en concreto, con la figura del abuso de posición dominante. Claro ejemplo de ello fue el caso IBM en EEUU a principios de los años ochenta<sup>692</sup> y el caso Microsoft, especialmente a este lado del Atlántico.

Quizás porque la generalización de las leyes defensoras de la libre competencia ha llevado a que los empresarios ya no tengan únicamente el derecho a competir que les otorga el principio de libre competencia, sino también la obligación de hacerlo<sup>693</sup>.

---

<sup>690</sup> G. MONTI, *EC Competition Law (Law in context)*, Cambridge, Cambridge University Press, 2007, pág. 2.

<sup>691</sup> O., KOLSTAD, "Competition Law and intellectual property rights – outline of an economics-based approach" en J., DREXL (ed.), *Research Handbook on Intellectual Property and Competition Law*, Cheltenham, Edward Elgar, 2008, pág. 3.

<sup>692</sup> Vid. XIV Report on Competition Policy, Comisión Europea, 1985, apartados 94 a 96; disponible en: [http://ec.europa.eu/competition/publications/annual\\_report/index.html](http://ec.europa.eu/competition/publications/annual_report/index.html) (última consulta 31 de enero 2013).

<sup>693</sup> J., SEGURA SÁNCHEZ, "La Política de Defensa de la Competencia: objetivos, fundamentos y marco constitucional", *Ekonomiaz*, núm. 61, 1º cuatrimestre, 2006, pág. 16 (18).

### *1.3. La relación entre el Derecho de defensa de la competencia y el Derecho de autor en la Directiva de software.*

En el derecho europeo, las autoridades de la Unión han abordado la relación entre derechos de propiedad industrial e intelectual en general y derecho de defensa de la competencia bajo la premisa de que los primeros no han de ser inmunes al segundo, y así lo recoge el Considerando Séptimo (7) de las Directrices a la aplicación del artículo 81 del Tratado de la CE (ahora artículo 101 del TFUE) a los acuerdos de transferencia de tecnología al declarar que: “el hecho de que la legislación sobre propiedad intelectual confiera derechos exclusivos de explotación no implica que los derechos de propiedad intelectual sean inmunes al Derecho de la competencia.”<sup>694</sup>

Esto se traduce en que si bien no existe conflicto entre la libre competencia y los derechos de propiedad exclusiva, la relación entre ambos sistemas no está carente de tensiones. Dichas tensiones obedecen a la diferente índole de los intereses que ambos derechos protegen, aunque en última instancia ambos sistemas de protección favorezcan la promoción de la innovación, el desarrollo económico y beneficien a la sociedad.

La Directiva de Software menciona en su Considerando Decimoséptimo el Derecho de defensa de la competencia, al indicar que además de la excepciones de los artículos 5 y 6, los artículos 101 y 102 TFUE podrán ser aplicados si el titular del programa en una posición dominante abusa de su derecho.

Argumentos a favor de que la intención de los redactores de la Directiva era mantener la aplicación del Derecho de defensa de la competencia como excepcional son:

- En primer lugar, el Considerando Décimo declara que las ideas y los principios implícitos en las interfaces del programa no están protegidos por el derecho de autor sobre el programa.

---

<sup>694</sup> Directrices a la aplicación del artículo 81 del Tratado CE a los acuerdos de transferencia de tecnología (2004/C 101/02) C 101/2, DO de 27.04.2004

- En segundo lugar, la propia existencia de la excepción de la descompilación regulada en el artículo 6.
- En tercer lugar, el uso de la expresión “sin perjuicio” en el Considerando Décimo parece reflejar el propósito del legislador de que el derecho de defensa de la competencia quede reservado para situaciones donde las excepciones de la Directiva no puedan dar solución al problema de la interoperabilidad<sup>695</sup>.

Dicho de otro modo, las conductas anticompetitivas que puedan surgir debido a la retención de la información interoperable por parte del titular del derecho de autor sobre el programa intentan ser incorporadas dentro de la Directiva<sup>696</sup>. Sin embargo, esta internalización de conductas y la relación de excepcionalidad que propone la Directiva es poco vinculante para el derecho de la competencia por razones de primacía normativa.

Atendiendo al principio de jerarquía normativa, las normas de defensa de la competencia del TFUE son de rango superior a la Directiva de Software, que es una norma de derecho secundario. Por lo tanto, la Directiva, a pesar de su transposición a los ordenamientos internos de los Estados miembros, no puede prevalecer sobre las normas de competencia. Esto significa que con independencia de la mención al Derecho de defensa de la competencia del Considerando Decimoséptimo de la Directiva, la aplicación del artículo 102 tendrá lugar a pesar de las normas previstas en la Directiva<sup>697</sup>.

En contra de este argumento, algunos autores han defendido la superioridad del artículo 345 TFUE (antiguo art. 295 TCE), que declara: “*los Tratados no prejuzgan en modo alguno el régimen de la propiedad en los Estados miembros*”.

Ahora bien, el TJUE ha mantenido que el art. 345 TFUE “no puede ser interpretado en el sentido de que reserva al legislador nacional, en materia de propiedad industrial y comercial, la facultad de adoptar medidas contrarias al principio de libre

---

<sup>695</sup> Vid. A. VAN, ROOIJEN, “Essential Facilities: Exploring the Software Directive’s equilibrium between intellectual property rights and competition law”, CRi, 2007, pág. 129 (131).

<sup>696</sup> J., SCHOVSBO, “As If Made for Each Other – Intellectual Property Rights and Protection of Compatible Products”, IIC 20(5), (1998), pág. 510 (524).

<sup>697</sup> A., HARATSCH, C., KOENIG, M., PECHSTEIN, *Europarecht*, Tübingen, 8ª ed., Mohr Siebeck, 2012, pág. 130. Sentencia del Tribunal de Primera Instancia (Gran Sala) de 17 de septiembre de 2007, Microsoft Corp. contra Comisión de las Comunidades Europeas, asunto T-201/04, Rec 2007 II-03601, ap. 227.

circulación de mercancías dentro del mercado común.”<sup>698</sup> Esta jurisprudencia del Tribunal de Justicia de la Unión Europea ha declarado que el Tratado no afecta a la mera existencia de derechos de propiedad intelectual configurados por las legislaciones nacionales, a la vez que ha entendido que el art. 345 TFUE (anterior 295 TCE) dispone que el Tratado no debe perjudicar las normas de los Estados miembros sobre derechos de propiedad. De tal modo que la relevancia del art. 345 TFUE para los derechos de propiedad intelectual sigue siendo incierta, sobre todo si se tiene en cuenta su origen legislativo, que no es otro que garantizar la libertad de los Estados miembros a optar entre la propiedad pública o privada de las empresas. No puede, por tanto, este precepto jugar un papel decisivo como defensa ante una violación de las normas comunitarias de defensa de la competencia<sup>699</sup>.

Otro argumento que podría añadirse al anterior es que el artículo 345 TFUE no debiera poder aplicarse a derechos de propiedad que han sido objeto de armonización vertical, porque la normativa reguladora sobre una materia armonizada no ha sido definida por el legislador nacional de cada Estado miembro, sino por el legislador comunitario<sup>700</sup>. Es esta la situación en que encuentra el derecho de autor sobre los programas de ordenador.

En resumen, aunque la intención del legislador comunitario al elaborar la Directiva en 1991 fuera que la doctrina de la dicotomía entre idea y expresión y la excepción legal del artículo 6 evitasen la entrada en juego de las normas de defensa de competencia, este hecho es más que una declaración de intenciones sin efecto material alguno.

Igual consideración merece la mención a los artículos 101 y 102 del Tratado (anteriores 81 y 82) en el Considerando Décimo Séptimo (17) de la Directiva por dos razones de técnica legislativa: Primero, los considerandos de una Directiva sirven para interpretar ésta, pero no tienen un valor normativo. En segundo lugar, como se ha comentado líneas más arriba, una directiva no puede sustituir ni modificar normas del

---

<sup>698</sup> Sentencia del Tribunal de Justicia de 18 de febrero de 1992, asunto C-30/90, Comisión de las Comunidades Europeas contra Reino Unido de Gran Bretaña y de Irlanda del Norte, Rec 1992 I-829, ap. 18; sentencia del Tribunal de Justicia de 13 de julio de 1995, asunto C-350/92, Reino de España contra Consejo de la Unión Europea, Rec 1995 I-1985 ap. 18.

<sup>699</sup> Vid. P.B., HUGENHOLTZ et al., “The recasting of copyright and related rights for the knowledge economy: final report”, European Commission DG Internal Market Study Contract No. ETD/2005/IM/D1/95, Institute for Information Law, Amsterdam, 2006, pág. 8.

<sup>700</sup> *Ibidem*, pág. 14

Tratado de aplicación directa y jerarquía superior. La Directiva no puede derivar responsabilidad respecto de una norma de rango jerárquico superior en una situación de la que no es responsable<sup>701</sup>.

A pesar de lo anterior, en la única situación en que el TJUE ha tenido que resolver un conflicto entre derecho de autor sobre un programa de ordenador y defensa de la competencia, la simple aplicación del principio de jerarquía normativa no ha tenido lugar, sino que el Tribunal de Justicia ha acudido a la aplicación de un test que parte de la existencia de circunstancias excepcionales. Este hecho es indicativo de que la relación entre ambos sistemas normativos no es tan simple como teóricamente parece<sup>702</sup>.

Para el Tribunal de Justicia, aunque el ejercicio del derecho de autor sobre un programa por parte de una empresa con posición dominante no es en sí mismo abusivo, su ejercicio inadecuado sí es susceptible de infringir el artículo 102<sup>703</sup>.

Partiendo de estos argumentos, en los siguientes apartados analizaremos el papel tutelar que el Derecho de defensa de la competencia viene jugando sobre los derechos de Propiedad Intelectual. En el concreto caso de los programas de ordenador, esta creciente tutela, ha situado al Derecho de defensa de la competencia como garante necesario de la interoperabilidad para no frenar la innovación y el libre mercado de programas. Para ello se analizará la repercusión que para el Derecho de autor ha tenido el único caso que hasta la fecha ha tratado la interoperabilidad informática cuando el titular del programa se niega a suministrar la información necesaria para alcanzarla a sus competidores: el caso Microsoft.

Para ello proponemos una exposición comparativa entre el concepto de la indispensabilidad de la descompilación para acceder a la información necesaria para alcanzar la interoperabilidad de la Directiva, y la indispensabilidad como requisito necesario para aplicar la doctrina de las “*essential facilities*” al evaluar un abuso de posición dominante derivado del ejercicio abusivo de un derecho de exclusiva.

La razón de esta comparación es que, en el fondo, ambas normas (Directiva de software y artículo 102 del TFUE) pretende el mismo objetivo: la garantía de la

---

<sup>701</sup> A. VAN, ROOIJEN, “Essential Facilities: Exploring ...”, *op.cit.*, pág. 131.

<sup>702</sup> F., HERNÁNDEZ RODRÍGUEZ, “Derecho de autor y abuso de posición dominante en la Unión Europea”, ADI 16, 1995, pág. 331 (340); J., TURNEY, J., “Defining the Limits of the EU Essential Facilities Doctrine on Intellectual Property Rights: The Primacy of Securing Optimal Innovation”, NW.J.TECH. & INTELL. PROP. 3, 2005, pág. 179 (194).

<sup>703</sup> Vid. C., BELLAMY, G., CHILD, *Derecho de la competencia en el Mercado Común*, ed. española a cargo de Enric Picañol, Madrid, 1992, pág. 449.



interoperabilidad entre sistemas informáticos para mantener la libre competencia en el sector. Este análisis pondrá de manifiesto no sólo las carencias del derecho de autor para garantizar la interoperabilidad en los términos que el artículo 6 está vigente, sino los retos que la garantía de la interoperabilidad informática plantea al Derecho de defensa de la competencia. No puede olvidarse que las autoridades de competencia no disponen de la pericia técnica ni científica necesaria para juzgar si un determinado derecho de propiedad intelectual o industrial es demasiado amplio o reducido. Esta tarea no es de su competencia. Por lo tanto, debiera preferirse la reforma de la normativa reguladora del derecho de propiedad intelectual concreto antes que la utilización del derecho de la competencia para reducir los defectos y los excesos de protección otorgados por los primeros<sup>704</sup>.

#### *1.4. El creciente papel tutelar del Derecho de defensa de la competencia.*

La libre competencia, ha de ser la “estrella polar” del sistema de propiedad intelectual, pero si el derecho de propiedad exclusiva que se concede no cumple con su finalidad, sino que utiliza como una herramienta de control del mercado, puede entrañar un alto coste social. Esta afirmación tiene dos consecuencias inmediatas: de una parte, la taxatividad que implica que no puede haber más monopolios que los establecidos en la leyes y, en segundo lugar, la lectura procompetitiva de los mismos, que significa que cuando existan diferentes interpretaciones, ha de preferirse aquella que resulte más favorable a la libre competencia<sup>705</sup>.

Con carácter general el papel tutelar del derecho de la competencia se manifiesta de dos maneras:

En primer término en la elaboración de normas jurídicas. El legislador ha de tener presente que los derechos de propiedad intelectual han de estar justificados por la necesidad de proteger intereses superiores a la libre competencia, ya que éste es un principio rector de la política económica de la Comunidad y de sus Estados miembros.

Ejemplo de buena práctica lo constituye la Directiva de 13 de octubre de 1998 sobre protección jurídica de diseño industrial y el Reglamento posterior<sup>706</sup>, que ha

---

<sup>704</sup> F., LÉVEQUE, *The Economics of ...*, *op.cit.*, pág. 87.

<sup>705</sup> G., GHIDINI, *Aspectos actuales del Derecho Industrial: Propiedad Intelectual y Competencia*, Granada, Comares, 2002, pág. 12.

<sup>706</sup> Reglamento (CE) nº 6/2002 del Consejo, de 12 de diciembre, de 2001 sobre los diseños o modelos comunitarios, DO L 3, 5.1.2002

tenido especial cuidado en garantizar la libre competencia a través de dos preceptos. La denominada cláusula *must-fit*, contenida en el artículo 8.2 del Reglamento, facilita la interoperabilidad de productos de diferentes fabricantes. Este precepto excluye del ámbito de protección las interconexiones (*must-fit*) a fin de que el derecho de diseño no se convierta en un obstáculo para que los terceros competidores ofrezcan productos que puedan interoperar con el producto protegido.

El segundo ejemplo se encuentra en la denominada “cláusula de reparación” por la que se persigue evitar que el derecho sobre un diseño pueda ser empleado para monopolizar el mercado secundario de intercambios. Este precepto permite a los terceros fabricar libremente repuestos protegidos por un derecho de diseño, previa remuneración al titular y bajo una serie de condiciones<sup>707</sup>.

El derecho de Defensa de la competencia también puede actuar a posteriori, es decir, una vez que la norma existe y la protección ya ha sido otorgada.

Por ejemplo, en los casos *Magill* e *IMS*, que analizaremos en las siguientes páginas, se había otorgado una protección basada en el derecho de autor para creaciones de efímero, por no decir nulo, valor creativo. El derecho de autor, que había nacido para proteger integralmente al autor, se había devaluado hasta el punto de dar cabida prácticamente a cualquier cosa: en el caso de *Magill*, a un listado de programación televisiva y en el caso *IMS* a la estructura de segmentos de una base de datos para realizar estadísticas de ventas de medicamentos.

Para corregir estos excesos hiperproteccionistas, el TJUE se vio obligado a recortar el alcance de estos derechos mediante la figura de la licencia obligatoria, propia del derecho de patentes, pero desconocida en el derecho de autor.

El derecho de autor concede a su titular una exclusividad total sobre su obra, entre lo que se incluye el derecho a rechazar licenciar su obra. Sin embargo, el derecho de propiedad exclusiva puede ser tutelado y corregido por el derecho de defensa de la competencia cuando el titular del derecho de autor tiene una posición dominante en el mercado y el rechazo a licenciar constituya un abuso de su derecho. Es decir, ante la

---

<sup>707</sup> Sobre las dos cláusulas vid. J. M., OTERO LASTRES, “Concepto de diseño y requisitos de protección en la nueva Ley 20/2003”, ADI 24, (2003) , pág. 53; C., LENCE REIJA, “El objeto protegido en la Directiva sobre diseño industrial”, ADI 19, (1998), pág. 273; A., GARCÍA VIDAL, (coord.), *El diseño comunitario. Estudios sobre el Reglamento (CE) núm. 6/2002*, Thomson-Reuters-Aranzadi, 2012, pág. 281.

tendencia expansionista del derecho de autor, las licencias obligatorias se conciben como correctivos excepcionales para paliar las consecuencia anticompetitivas de esos derechos.

Sin embargo no toda la doctrina comparte este planteamiento. Algunos autores consideran que las políticas de competencia que debilitan los derechos de propiedad intelectual afectan a los incentivos a la innovación para las empresas en prácticamente cualquier industria. El resultado es menos margen para la innovación y perjuicio al bienestar de los consumidores. Las políticas de competencia que debilitan las protecciones internacionales de propiedad intelectual reducen la difusión de la innovación a través de las fronteras internacionales y disminuye las ganancias potenciales del comercio asociado a los mercados internacionales de tecnología<sup>708</sup>.

El espíritu de la libre competencia debe presidir la regulación de los derechos de propiedad intelectual, de forma que la protección no se extienda más allá de lo necesario para alcanzar la finalidad perseguida por los mismos. Esta conciliación también es positiva para fomentar la igualdad entre competidores y, en última instancia, la cohesión económica, que es uno de los fines perseguidos por el Tratado.

La compatibilización entre ambos sistemas debiera hacer pensar a los poderes públicos y, en particular, a aquellos que ejercen algún tipo de responsabilidad en la elaboración de normas sobre propiedad intelectual, que al realizar esta tarea deben medirse cuidadosamente las consecuencias económicas y sociales que tiene una protección excesiva de las creaciones o de las invenciones, y que la corrección de estos derechos de propiedad a través del derecho de la competencia solo debiera hacerse en situaciones de excepcionalidad.

## ***2. El papel de la doctrina de las “essential facilities” en el debate sobre la interoperabilidad informática.***

### *2.1. La doctrina de las “essential facilities” en la jurisprudencia comunitaria como correctora excepcional del ejercicio abusivo de los derechos de Propiedad Industrial e Intelectual.*

---

<sup>708</sup> D.F., SPULBER, “Competition Policy and the Incentive to Innovate: The Dynamic Effects of Microsoft v. Commission”, Northwestern Law & Econ Research Paper No. 08-18, 2008, pág. 61, disponible en SSRN: <http://ssrn.com/abstract=1146451> or <http://dx.doi.org/10.2139/ssrn.1146451> (última consulta 1 de junio, 2013).

La doctrina de las “essential facilities” se incluye dentro de la figura de abuso de posición de dominio del Derecho de defensa de la competencia. El Tribunal de Justicia de la Unión ha definido el abuso de posición dominante como aquel comportamiento excluyente de una o varias empresas, que con poder de mercado, afecta al comercio del mercado interior<sup>709</sup>.

La doctrina de las “essential facilities” responde a este tipo de actuación abusiva, y consiste en la obligación de compartir activos con un competidor.

La jurisprudencia y doctrina norteamericanas crearon esta doctrina para determinar los casos en que una empresa puede verse obligada a compartir recursos con sus competidores. Un recurso esencial (*essential facility*) es aquello que debe estar a disposición de todos los competidores, pues esta disponibilidad es imprescindible para que exista competencia en el mercado en cuestión<sup>710</sup>.

Sobre esta base, el Tribunal de Justicia de la Unión Europea ha utilizado esta doctrina para obligar al titular de un derecho de propiedad industrial o intelectual a conceder licencias a sus competidores en los casos en que tal licencia sea indispensable para competir en el mercado en cuestión. Es decir, aunque la regla general es que la negativa a conceder licencias no constituye en sí misma un abuso de posición dominante, en circunstancias excepcionales, el artículo 102 TFUE se ha utilizado para obligar a una empresa dominante a concederlas a sus competidores. Ello se traduce en que si el derecho de propiedad industrial o intelectual es calificado como esencial para poder competir en el mercado, su titular no podrá negarse a conceder licencias a sus competidores y, si se niega, podrá ser sancionado por abuso de posición de dominio.

En el derecho comunitario, esta doctrina ha sido objeto de amplio desarrollo jurisprudencial, empleándose como instrumento corrector de situaciones de abuso de derechos de propiedad industrial e intelectual por sus titulares.

---

<sup>709</sup> Vid. entre otras, el apartado 30 de la sentencia del Tribunal de Justicia de 9 de noviembre de 1983, asunto 322/81, NV Nederlandsche Banden Industrie Michelin v. Comisión de las Comunidades Europeas, REC 1983-03461. En igual sentido, la Comunicación de la Comisión — Orientaciones sobre las prioridades de control de la Comisión en su aplicación del artículo 82 del Tratado CE a la conducta excluyente abusiva de las empresas dominantes (Texto pertinente a efectos del EEE), DO n° C 045 de 24/02/2009, pág. 7, ap. 1. La literatura jurídica es muy abundante y puede consultarse, entre otros: P., BELLO MARTÍN-CRESPO, P., HERNÁNDEZ RODRÍGUEZ, *Derecho de la Libre Competencia Comunitario y Español*, Navarra, Thomson Aranzadi, 2009, págs. 113-140; G., MONTI, *EC Competition Law (Law in context)*, Cambridge, Cambridge University Press, 2007, pág. 159-243.

<sup>710</sup> Aunque a ambos lados del Atlántico la doctrina de las “essential facilities” ha sido polémica desde su origen porque cercena la libertad a contratar. Vid. entre otros, P.E., AREEDA, “Essential facilities: an epithet in need of limiting principles”, *Antitrust L.J.* 58, 1990, pág. 841 (852).

#### A) Volvo.

Este es el primer asunto donde se plantea el conflicto entre los derechos de propiedad exclusiva y la libre competencia. En el caso Volvo<sup>711</sup> el Tribunal Supremo británico remitió una cuestión prejudicial en la que planteaba la si constituía una posición a efectos del artículo 86 del Tratado CEE la situación en que una empresa fabricante de automóviles (Volvo), y titular de diseños industriales registrados sobre partes de la carrocería que le otorgan el derecho exclusivo a fabricar e importa las piezas de recambio. De ser así, planteaba el tribunal británico si la negativa a conceder a terceros una licencia para suministrar piezas de recambio, a pesar de que éstos estuvieran dispuestos a pagar una contraprestación económica, constituía una explotación abusiva a efectos del mismo artículo del Tratado.

El relación a la primera cuestión, el Tribunal comunitario declaró que la facultad del titular de un derecho de propiedad industrial de impedir a terceros usos no autorizados constituye “el contenido mínimo de su derecho exclusivo<sup>712</sup>” y que, en consecuencia, una obligación impuesta al titular de conceder una licencia afectaría a la “esencia” del derecho en cuestión. Sin embargo a continuación, proporciona tres ejemplos de cuándo una negativa a conceder una licencia puede constituir abuso de posición dominante: “la negativa arbitraria a suministrar piezas de recambio a los talleres independiente, la fijación de los precios para las piezas de recambio a un nivel no equitativo o la decisión de no seguir produciendo piezas para un determinado modelo cuando todavía circulan muchos vehículos de ese tipo<sup>713</sup>”. Además, destaca el Tribunal que cualquiera de estas tres situaciones debe ser susceptible de afectar al comercio entre Estados miembros, es decir, afectar a la competencia en general, no a un competidor en particular.

---

<sup>711</sup> Sentencia del Tribunal de Justicia de 5 de octubre de 1988, asunto 238/87, AB Volvo v. Erik Veng (UK) Ltd., Rec 1988, 06211.

<sup>712</sup> Ibidem, ap. 8.

<sup>713</sup> Ibidem, ap. 9.

## B) Magill.

El caso Magill fue el primero donde el Tribunal de Justicia aplicó la doctrina de las “essential facilities” a una situación donde titulares de derechos de autor se negaron a conceder licencias a un competidor<sup>714</sup>. En este caso, una empresa de radiodifusión televisiva de Irlanda, *Magill TV Guide Ltd.*, había intentado publicar una guía semanal general de televisión, pero las empresas de teledifusión que operaban en Irlanda (RTE, ITP y BBC), se lo impidieron, no suministrándole información acerca de cuáles iban a ser los contenidos a emitir. A su vez, las tres empresas mencionadas, elaboraron sus propias guías de programación semanal.

Esas listas de programación televisivas estaban protegidas por derechos de autor por lo que, técnicamente, sus titulares estaban facultados para decidir si autorizaban o denegaban su publicación a terceros. A pesar de ello, la negativa a ceder la publicación de estos listados a terceros fue considerada por el TJUE un abuso de posición dominante, destacando como comportamiento abusivo, el hecho de que las empresas que le habían negado a Magill la autorización de publicación habían impedido la aparición de un producto nuevo en el mercado<sup>715</sup>.

Para valorar la existencia de abuso de posición dominante el Tribunal aplicó un test que consideraba las siguientes circunstancias:

- 1º si el acceso a esa información era indispensable para el ejercicio de una actividad en un mercado;
- 2º si la negativa del titular del derecho de autor podía impedir la aparición de un nuevo producto para el cual existiese potencial demanda de los consumidores, y
- 3º si su negativa podía excluir la competencia en un mercado conexo<sup>716</sup>.

La presencia de estas circunstancias era, en palabras del Tribunal, suficiente para activar la doctrina de las “essential facilities” ante el principio general de que una empresa puede libremente elegir con quien contrata. Si la empresa ostenta una posición

---

<sup>714</sup> Sentencia del Tribunal de Justicia de 6 de abril de 1995, asuntos acumulados C-241/91 P y C-242/91 P, *Radio Telefís Eireann (RTE) e Independent Television Publications Ltd (ITP) v. Comisión de las Comunidades Europeas*, Rec. 1995 I-743. (En adelante *Magill*).

<sup>715</sup> *Ibidem*, ap. 54. Un comentario detallado sobre el asunto vid. F., HERNÁNDEZ RODRÍGUEZ, “Derecho de autor y abuso ...”, *op.cit.*, pág. 331.

<sup>716</sup> *Ibidem*, aps. 54 a 56.

de dominio, y se producen las condiciones citadas, no habrá justificación objetiva para negarse a negociar, sino obligación de negociar. Ahora bien, el Tribunal no aclaró si estas circunstancias eran acumulativas o exhaustivas.

En relación al Derecho de autor, el Tribunal consideró que el ejercicio por un autor de su derecho a impedir la reproducción de su obra, a pesar de formar parte del objeto específico del derecho y de no extender sus efectos a ningún mercado secundario, cuando persiguen una finalidad manifiestamente contraria a los objetivos del artículo 86 del Tratado CEE, constituye un abuso de posición dominante. Además, en tales circunstancias, el ejercicio del derecho de autor deja de responder a su función esencial de garantizar la protección moral de la obra y la retribución del esfuerzo del creador<sup>717</sup>.

Hasta el caso Magill, el ejercicio de los derechos de propiedad intelectual (e industrial) por una empresa con posición dominante estaba sometido a dos límites por el derecho comunitario de la competencia: Su ejercicio tenía que formar parte del contenido esencial del derecho, y los efectos de dicho ejercicio habían de circunscribir sus efectos al mercado de los productos o servicios protegidos<sup>718</sup>. A partir de Magill, hay que añadir este test de circunstancias excepcionales a los límites de ejercicio del derecho de propiedad exclusiva.

### C) Oscar Bronner.

El siguiente caso del Tribunal comunitario que hemos escogido en nuestro análisis es el caso de Oscar Bronner<sup>719</sup>, que a pesar de no referirse directamente a los derechos de propiedad industrial e intelectual, hace una importante referencia a los casos Volvo y Magill, con la que el Tribunal da a entender que los criterios allí establecidos no tienen porqué aplicarse únicamente a los conflictos entre derechos de propiedad exclusiva y libre competencia.

El Sr. Bronner, propietario de una pequeña empresa de prensa diaria, invocando el artículo 82 del Tratado CE, había demandado a Mediaprint, empresa dominante del

---

<sup>717</sup> *Ibíden*, ap. 28.

<sup>718</sup> F., HERNÁNDEZ RODRÍGUEZ, “Derecho de autor y abuso ...”, *op.cit.*, pág. 335.

<sup>719</sup> Sentencia del Tribunal de Justicia (Sala Sexta) de 26 de noviembre de 1998, asunto C-7/97, Oscar Bronner GmbH & Co. KG contra Mediaprint Zeitungs- und Zeitschriftenverlag GmbH & Co. KG, Mediaprint Zeitungsvertriebsgesellschaft mbH & Co. KG y Mediaprint Anzeigengesellschaft mbH & Co. KG. Rec I-7817.

ramo en Austria, por haberle negado el acceso a su sistema nacional de reparto a domicilio.

Aunque la cuestión planteada se limita a la negativa a suministrar un servicio, el Tribunal abre la puerta a una visión de conjunto sobre el abuso de posición dominante en todos los casos de “negativa a negociar”, válida tanto para situaciones sobre el ejercicio abusivo de los derechos de propiedad industrial e intelectual, como para situaciones donde se trate la negativa a suministrar un bien o servicio.

Al analizar la posible aplicación al caso de la doctrina *Volvo* y *Magill*, el Tribunal afirma que la misma puede ser invocada en una situación como la objeto de examen, pero para ello sería preciso que no sólo la denegación del servicio pudiera eliminar toda competencia en el mercado examinado (los diarios) por parte de quien solicita el servicio y no pudiera justificarse objetivamente, sino que, además, el servicio en sí mismo, fuera indispensable para el ejercicio de la actividad de éste, en el sentido de que no hubiera ninguna alternativa real o potencial al citado sistema de reparto a domicilio.

Además, con esta declaración, el Tribunal parece admitir el criterio de que el comportamiento abusivo basado en la negativa a negociar, debe ser susceptible de eliminar a un concreto competidor en particular, en lugar de afectar a la competencia del mercado interior<sup>720</sup>, como había declarado en *Magill* y en *Volvo*.

A pesar de que el Tribunal declaró que no constituía abuso de posición dominante el hecho de que una empresa periodística, aún en situación de dominio, denegase el acceso a su sistema de distribución, a cambio de una contraprestación adecuada, a un competidor, que no está en condiciones de crear y de gestionar, en condiciones económicamente rentables, por sí solo o en colaboración con otros editores, su propio sistema de reparto a domicilio, el concepto de “essential facility” se vuelve bastante borroso tras este fallo. Después de haber insistido en *Magill* en que el acceso a la información debía ser “indispensable” y no únicamente conveniente para un nuevo

---

<sup>720</sup> Sentencia del Tribunal de Justicia (Sala Sexta) de 26 de noviembre de 1998, asunto C-7/97, Oscar Bronner..., *op.cit.*, ap. 41.



competidor, el Tribunal define indispensable en el caso *Oscar Bronner* como “económicamente no rentable”<sup>721</sup>.

#### D) IMS Health.

En el caso *IMS Health*<sup>722</sup>, el Tribunal comunitario responde a una cuestión prejudicial planteada por un órgano judicial alemán sobre la interpretación del artículo 82 del Tratado CE, en el caso de que una empresa con posición dominante en un mercado se niegue a celebrar un contrato de licencia para la utilización de una base de datos amparada por un derecho de propiedad intelectual, con una empresa que desea introducirse en el mismo mercado geográfico y material<sup>723</sup>.

La pregunta se planteó a consecuencia de que una empresa alemana, IMS, titular de un derecho de autor sobre una estructura de segmentos para realizar estudios sobre ventas regionales de medicamentos en Alemania, se negó a suministrar a sus competidores una licencia sobre dicha estructura de segmentos<sup>724</sup>.

El Tribunal no respondió si la empresa IMS abusaba de su posición de dominio al negarse a conceder una licencia sobre sus estructuras de segmentos, sino que partiendo de los criterios establecidos en el caso *Magill*, los desarrolla y va un paso más lejos, declarando que las tres circunstancias excepcionales deben cumplirse de forma acumulativa para que la negativa a conceder una licencia a un competidor sobre un derecho de propiedad intelectual por parte de una empresa con posición de dominio, constituya un abuso de posición dominante y pueda ser obligada a negociar.

En palabras del propio Tribunal:

*“1) Para el examen del carácter eventualmente abusivo de la negativa de una empresa que ocupa una posición dominante a otorgar una licencia de utilización sobre*

---

<sup>721</sup> *Ibidem*, ap. 46.

<sup>722</sup> Sentencia del Tribunal de Justicia de 29 de abril de 2004, asunto C-418/01, *IMS Health GmbH & Co. OHG v. NDC Health GmbH & Co. KG*, Rec. 2004 I-05039.

<sup>723</sup> *Ibidem*, ap. 17.

<sup>724</sup> Para un comentario detallado de la sentencia vid. B., CONDE GALLEGO, “La negativa unilateral de una empresa en posición de dominio a conceder una licencia para la utilización de un derecho de propiedad intelectual [Comentario a la Sentencia del TJCE de 29 de abril de 2004, asunto C-418/01, *IMS Health GmbH & Co. OHG v. NDC Health GmbH & Co. KG*], en ADI 24, 2003, pág. 465.

*una estructura de segmentos protegida por un derecho de propiedad intelectual del que es titular, el grado de participación de los usuarios en el desarrollo de esa estructura y el esfuerzo, en particular, en cuanto al coste, que los usuarios potenciales deberían realizar para poder comprar estudios sobre las ventas regionales de productos farmacéuticos presentados sobre la base de una estructura alternativa son elementos que deben ser tomados en consideración para determinar si la estructura protegida es indispensable para la comercialización de estudios de esta naturaleza.*

*2) La negativa de una empresa, que ocupa una posición de dominio y que es titular de un derecho de propiedad intelectual sobre una estructura de segmentos indispensable para la presentación de datos sobre las ventas regionales de productos farmacéuticos en un Estado miembro, a otorgar una licencia para la utilización de esta estructura a otra empresa, que desea asimismo suministrar tales datos en el mismo Estado miembro, constituye un abuso de posición en el sentido del art. 82 CE siempre que concurran los siguientes requisitos:*

- la empresa que ha solicitado la licencia pretenda ofrecer, en el mercado del suministro de datos de que se trata, productos o servicios nuevos que el titular del derecho de propiedad intelectual no ofrece y para los cuales existe una demanda potencial por parte de los consumidores;*
- la negativa no esté justificada por consideraciones objetivas;*
- la negativa pueda reservar a la empresa titular del derecho de propiedad intelectual el mercado del suministro de datos sobre ventas de productos farmacéuticos en el Estado miembro de que se trate, excluyendo toda competencia sobre este<sup>725</sup>”.*

En relación al ejercicio abusivo del derecho de autor, el Tribunal, reitera la jurisprudencia de Volvo y Magill, insistiendo en que la simple negativa a conceder una licencia por parte del titular de un derecho de propiedad intelectual, aún en el caso de ostentar una posición de dominio, no constituye abuso<sup>726</sup>.

---

<sup>725</sup> Sentencia del Tribunal de Justicia de 29 de abril de 2004, asunto C-418/01, IMS Health..., op.cit., ap. 28 a 30.

<sup>726</sup> *Ibíd*em, ap. 34: “El derecho exclusivo de reproducción forma parte de las prerrogativas del titular de un derecho de propiedad intelectual, por lo que la negativa a conceder una licencia, aunque sea por parte de una empresa en posición dominante, no puede constituir en sí misma un abuso de ésta (sentencias de 5 de octubre de 1988, Volvo, 238/87, Rec. p. 6211, apartado 8, y Magill, antes citada, apartado 49).”

El concepto de “indispensable” que se plantea en IMS Health se ha calificado como excepcional en relación con el derecho de autor<sup>727</sup>. El elemento fundamental para calificar a la estructura de segmentos protegida por derechos de autor como indispensable se encuentra en que dicha estructura se había podido convertir en un estándar del sector<sup>728</sup>. Esta definición excepcional en el contexto de los derechos de propiedad intelectual, veremos en las siguientes páginas, deja de serlo al tratar la información interoperable que contienen los códigos de los programas de ordenador.

## *2.2. La aplicación de la doctrina de las “essential facilities” en situaciones donde no se proporciona la información interoperable a los competidores.*

### A) Microsoft.

El caso Microsoft trata del acceso de las personas al mejor sistema operativo posible. El problema se presenta en Europa después de haber tenido lugar un proceso homónimo en Estados Unidos. En noviembre de 1998 una empresa competidora (Sun Microsystems) denuncia a Microsoft ante la Comisión por su negativa a proporcionarle la información y la tecnología necesarias que permitiesen la interoperabilidad de sus sistemas operativos Windows para servidores de grupos de trabajo con el sistema operativo Windows para ordenadores personales clientes<sup>729</sup>.

En la Decisión de la Comisión de 24 de marzo de 2004, confirmada por el Tribunal de Justicia en 2007, la Comisión Europea impuso a Microsoft una multa de 497,2 millones de euros por negarse a concederle a Sun Microsystems una licencia sobre sus derechos de propiedad intelectual para que ésta desarrollase sus propios programas para servidores y, en definitiva, para que pudiera diseñar un sistema

---

<sup>727</sup> B., CONDE GALLEGO, “Unilateral refusal to license indispensable IP rights” en J., DREXL, (ed.), *Research handbook on intellectual ...*, *op.cit.*, pág. 229.

<sup>728</sup> *Ibidem*, ap. 28 a 30.

<sup>729</sup> Los plazos en esta primera fase son muy largos. La solicitud de información a Microsoft por parte de Sun se hace el 15 de septiembre de 1998. La denuncia es presentada ante la Comisión el 10 de diciembre de 1998. Ésta remite un primer pliego de cargos a Microsoft el 2 de agosto de 2000, que responde el 17 de noviembre de ese año. Vid. Sentencia del Tribunal de Primera Instancia (Gran Sala) de 17 de septiembre de 2007, Microsoft Corp., *op.cit.*, aps. 2 a 9.

operativo para servidores de grupos de trabajo capaz de intercambiar información y comunicarse con los servidores de iguales características de Microsoft<sup>730</sup>.

A Microsoft se le denunció por llevar a cabo varias prácticas anticompetitivas. Por un lado, se le acusa de no permitir la interoperabilidad al no facilitar información para la interconexión. Por otro lado, también se denunció la realización de contratos vinculados al exigir al comprador de Windows que adquiriera también el Windows Media Player. En este estudio el comportamiento que interesa es el primero. En palabras del TJUE, “el primer comportamiento abusivo reprochado a Microsoft consiste en la negativa de esta a facilitar a sus competidores la información relativa a la interoperabilidad y a autorizar su uso para el desarrollo y la distribución de productos que compiten con los suyos en el mercado de los sistemas operativos para servidores de grupos de trabajo durante el período comprendido entre el mes de octubre de 1998 y la Decisión impugnada<sup>731</sup>”.

El papel tutelar del Derecho comunitario de la competencia sobre el Derecho de autor queda confirmado en la sentencia del caso Microsoft cuando el Tribunal declara que, en última instancia, tanto el software como las especificaciones pueden estar protegidas por derechos de propiedad intelectual e industrial, y estos derechos delimitan los límites donde la competencia tiene lugar<sup>732</sup>.

Se trata además del primer caso en que el Tribunal de Justicia comunitario valora la importancia de la interoperabilidad para el sector de las TICs. En el caso Microsoft la información interoperable se presenta como indispensable en los mismo términos que en el caso IMS, es decir, el propietario de una información protegida por derechos de propiedad intelectual que se ha podido convertir en una norma del sector (estándar *de facto*). Pero además, en este caso, la negativa del titular a negociar puede tener como consecuencia la obstrucción a la innovación en el sector del software.

---

<sup>730</sup> Es interesante destacar que para algún autor, el caso Microsoft es ejemplo de cómo la doctrina de las “essential facilities” se ha empleado no para obligar a licenciar sino para obligar a divulgar un secreto empresarial al competidor que abusa de su posición de dominio (G., SURBLYTÉ, *The Refusal to Disclose Trade Secrets as an Abuse of Market Dominance – Microsoft and Beyond*, Berne, Stämpfli 2011, pág. 142). En nuestro trabajo, el análisis de la interoperabilidad informática parte de la normativa de propiedad intelectual sobre el programa de ordenador, y como se ha demostrado en los capítulos anteriores, a la luz de la normativa vigente, es posible afirmar que las interfaces informáticas (al menos las especificaciones) estén fuera del ámbito de protección del derecho de autor sobre el programa de ordenador, con lo cual esta postura parece razonable.

<sup>731</sup> Sentencia del Tribunal de Primera Instancia (Gran Sala) de 17 de septiembre de 2007, Microsoft Corp, *op.cit.*, ap. 36.

<sup>732</sup> J., FARRELL, “Standardization and Intellectual...”, *op.cit.*, pág. 48-49.

Respecto a la concreta cuestión del derecho de autor sobre el programa de ordenador del que Microsoft es titular, ni la Comisión en su Decisión, ni el Tribunal en la sentencia de apelación, reconocen expresamente que la información necesaria para la interoperabilidad esté protegida por el derecho de autor. Ambos órganos basan su decisión de no pronunciarse a este respecto en que no era necesario, porque Microsoft abusa de su posición de dominio y cumple las condiciones de la doctrina de las circunstancias excepcionales o “essential facilities”<sup>733</sup>. Sin embargo sí reconocen que la divulgación de esa información a los competidores por parte de Microsoft requiere una licencia obligatoria<sup>734</sup>.

De esta manera, a pesar de reconocer indirectamente la existencia de derechos de propiedad intelectual sobre la información que Microsoft se negó a licenciar a sus competidores, la Comisión y el TJUE dejan la cuestión del ámbito de esa protección sobre estándares de programas de ordenador e información interoperable sin respuesta.

El Tribunal reitera la jurisprudencia anterior respecto de qué ha de entenderse por ejercicio abusivo de un derecho de propiedad intelectual<sup>735</sup>.

Queda claro que para el TJUE la regla general es la prevalencia del derecho de propiedad intelectual o industrial, y el abuso es la excepción<sup>736</sup>. En circunstancias normales el titular de un derecho de propiedad intelectual o industrial está legitimado

---

<sup>733</sup> Sentencia del Tribunal de Primera Instancia (Gran Sala) de 17 de septiembre de 2007, Microsoft Corp, *op.cit.*, ap. 287

<sup>734</sup> Decisión 2007/53/CE de la Comisión de 24 de marzo de 2004, relativa a un procedimiento con arreglo al artículo 82 CE y al artículo 54 del Acuerdo EEE contra Microsoft Corporation (Asunto COMP/C-3/37.792 – Microsoft) DO 2007, L32, pág. 23, ap. 30 y 31: “(...) En la medida en que la Decisión pudiera hacer necesario que Microsoft se abstuviera de ejercer plenamente alguno de sus derechos de propiedad intelectual, ello estaría justificado por la necesidad de poner término al abuso” y “las condiciones bajo las cuales Microsoft debe revelar la información y permitir su uso deben ser razonables y no discriminatorias. Este requisito (...) [se refiere] a cualquier retribución que Microsoft pueda exigir por la información facilitada”.

Sentencia del Tribunal de Primera Instancia (Gran Sala) de 17 de septiembre de 2007, Microsoft Corp, *op.cit.* ap. 288: “(...) en la Decisión impugnada no se había determinado en absoluto que la información relativa a la interoperabilidad no estaba protegida por una patente o por un derecho de autor, ni tampoco, por el contrario, que sí lo estaba. La Comisión consideró que no era necesario pronunciarse al respecto, dado que, en cualquier caso, “concurrían los requisitos para apreciar la existencia de un abuso y para imponer la medida correctiva, estuviera protegida o no la información por cualquier patente o derecho de autor”.

<sup>735</sup> Sentencia del Tribunal de Primera Instancia (Gran Sala) de 17 de septiembre de 2007, Microsoft Corp, *op.cit.*, ap. 331.

<sup>736</sup> A., CALVO CARAVACA, J., RODRÍGUEZ RODRIGO, *La Doctrina de las Infraestructuras Esenciales en Derecho Antitrust Europeo*, Revista de Derecho de la Competencia y la Distribución, La Ley, España, 2012, pág. 163. En igual sentido, F., HERNÁNDEZ RODRÍGUEZ, “Derecho de autor y abuso ...”, *op.cit.*, pág. 331; S., ANDERMAN, “Microsoft v. Commission and the interoperability issue”, en L., RUBINI (ed.), *Microsoft on Trial*, Cheltenham, Edward Elgar, 2010, pág. 258.

para negarse a conceder una licencia, precisamente porque la exclusividad es la esencia de este clase de derechos de propiedad especial.

Pero en Microsoft las circunstancias no eran normales. El reproche a la conducta de Microsoft reside en que de manera deliberada estaba restringiendo la interoperabilidad entre sus sistemas operativos (Windows) y otra clases de sistemas operativos como los desarrollados por Sun Microsystems (en concreto sistemas operativos para servidores de grupos de trabajo). Teniendo en cuenta que el sistema operativo Windows era el más vendido en el mundo, y que, por ello, Microsoft tenía posición de dominio en ese mercado, el que esta compañía se negase a facilitar la información necesaria para la interoperabilidad (los protocolos de las interfaces), lo convertía en la única empresa que abastecía ese mercado: el comportamiento de la compañía impide la competencia.

La ironía del caso es que una aplicación literal de las reglas establecidas en los casos precedentes Magill e IMS Health habrían llevado a la conclusión de que *Microsoft* no había abusado de su posición de dominio, incluso cuando lo que estaba en juego en este caso era de mucha más importancia<sup>737</sup>.

Si bien el Tribunal utiliza los requisitos recogidos en los casos anteriores Magill e IMS Health, hay dos factores que el Tribunal parece tener muy en cuenta para valorar si la negativa del titular de los derechos de propiedad intelectual puede impedir la aparición de un nuevo producto con potencial demanda de los consumidores:

- en primer lugar, la necesidad de mantener la interoperabilidad en el sector de las tecnologías de la información y,
- en segundo lugar, la salvaguarda de la innovación.

Sun había demostrado que era una fuente real de innovación y prometía continuar siéndolo. Microsoft no impedía la entrada de nuevos competidores en el mercado. Sun y otras empresas desarrolladoras de servidores de trabajo en grupo habían disfrutado de un período de interoperabilidad con Windows, basado en el hecho de Microsoft sí divulgaba la información de las interfaces, aunque ese intercambio de información se producía con un competidor distinto. La negativa de Microsoft a

---

<sup>737</sup> E., FOX, “The EC Microsoft case and duty to deal: the transatlantic divide”, en L., RUBINI (ed.), *Microsoft on Trial. Legal and Economic Analysis of a Transatlantic Antitrust Case*, Cheltenham, Edward Elgar, 2010, pág. 278.

continuar con esa práctica, tras recibir la carta de Sun, podría objetivamente haberse catalogado como un abuso en el ejercicio de sus derechos de exclusiva desde el caso *Commercial Solvents*<sup>738</sup>. Pero la negativa de *Microsoft* fue interpretada en sentido amplio, como la obstaculización a la aparición de un nuevo producto, porque limitaba el desarrollo técnico en perjuicio de los consumidores<sup>739</sup>.

Siguiendo la doctrina del caso *Microsoft*, podríamos asumir que donde un operador tenga una posición de dominio, protegida además con derechos de propiedad intelectual o industrial, éste haya estado tratando con empresas que operan en un mercado secundario dependiente y la competencia en ese mercado secundario (i) cuenta con características avanzadas en sus propios productos, es decir, dichos competidores invierten en innovación, (ii) pero para garantizar la viabilidad de sus innovaciones requieren acceso a la información interoperable de la empresa dominante, el artículo 102 TFUE podría ser aplicado.

El riesgo de esta adaptación, que cambia las reglas del test, puede conllevar que allí donde se produzca la explotación de un derecho de propiedad intelectual o industrial de una manera que sea acorde con la normativa reguladora, la práctica adoptada para explotar el derecho de propiedad intelectual o industrial pueda, en determinadas circunstancias, calificarse como ilícita con arreglo a la normativa de defensa de la competencia.

Por eso el caso *Microsoft* ha sido objeto de duras críticas, al considerarlo un falso positivo de la doctrina de las circunstancias excepcionales, penalizando el éxito en ausencia de abuso, lo que podría minorar los incentivos para innovar y una competencia dinámica en el sector informático<sup>740</sup>. Es decir, los innovadores podrían ampararse en la protección del abuso de posición de dominio porque la negativa del titular en estas circunstancias supondría la limitación del desarrollo técnico del mercado.

---

<sup>738</sup> *Commercial Solvents* fue el primer asunto en el que el TJUE se pronunció sobre un problema de infraestructuras esenciales. Sentencia del Tribunal de Justicia de 6 de marzo de 1974, *Istituto Chemioterapico Italiano S.p.A. y Commercial Solvents Corporation v. Comisión*, asuntos acumulados 6/73 y 7/73, Rec no disponible. Comentario sobre la misma en: A., CALVO CARAVACA, J., RODRÍGUEZ RODRIGO, *La Doctrina de...*, *op.cit.*, pág. 95.

<sup>739</sup> Sentencia del Tribunal de Justicia (Gran Sala) de 17 de septiembre 2007, *Microsoft/Comisión*, T-201/04, *op.cit.*, ap. 665.

<sup>740</sup> Vid. entre otros, A., DEVLIN *et.al.*, “Success, Dominance and Interoperability”, *Ind. L.J.* 84, (2009), pág. 1156 (1168); D., SPULBER, “Competition Policy and the Incentive to Innovate: The Dynamic Effect of *Microsoft v. Commission*”, *Yale J. on Reg.* 25(2), (2008), pág. 247; G., SURBLYTÉ, *The refusal to disclose trade secrets as an abuse of market dominance – Microsoft and beyond*, *Munich Series on European and International Competition Law*, vol. 28, Berne, Stämpfli Publishers, 2011, pág. 138-139

En consecuencia, las condiciones excepcionales con arreglo a las cuales el derecho de la competencia puede calificar una negativa a compartir un derecho de propiedad intelectual o industrial como abusiva, son desde *Microsoft* son menos claras<sup>741</sup>.

## B) Apple.

Poco después de que la Comisión hubiera emitido su Decisión sobre el caso *Microsoft* en 2004, tuvo lugar en Francia el caso *Apple v. VirginMega*<sup>742</sup>, donde la segunda denunció a la primera por abuso de posición dominante al negarse a conceder una licencia y proporcionar la información interoperable de su programa “Fair Play”.

En esta ocasión, el Consejo de Competencia francesa, aplicando la doctrina de las “essential facilities” comunitaria, llegó a la conclusión de que no existía abuso de posición dominante ni por tanto, obligación a conceder la licencia.

Ambas empresas se dedicaban al negocio de proveer música a través de Internet. Los archivos musicales adquiridos en la plataforma de Apple (iTunes Music Store) estaban protegidos por un programa denominado “Fair Play”, consistente en una MTP. De tal modo, que los consumidores que adquirirían música procedente de Apple sólo podían reproducirla en dispositivos que fueran compatibles con el programa “Fair Play”, producidos por Apple (iTunes Player y iPod). De tal modo, que los consumidores que utilizaban la plataforma de música online de VirginMega no podía reproducir sus archivos musicales en dispositivos de Apple. En este escenario, VirginMega solicita a Apple que le conceda una licencia sobre su programa MTP “Fair Play”, a lo que ésta se niega porque no quiere que su competidor pueda acceder a la información interoperable del programa.

Aunque el Consejo de Competencia francés concluyó que no había abuso de posición dominante de Apple, lo interesante de la resolución se encuentra en la

---

<sup>741</sup> Vid. en este sentido, E., FOX, “The EC Microsoft case ...”, *op.cit.*, pág. 275.

<sup>742</sup> Decisión del Consejo de la Competencia de la República francesa nº 04-D-54 de 9 de noviembre de 2004, (*Décision n° 04-D-54 du 9 novembre 2004, relative à des pratiques mises en œuvre par la société Apple Computer, Inc. dans les secteurs du téléchargement de musique sur Internet et des baladeurs numériques*), disponible en: <http://www.autoritedelaconcurrence.fr/pdf/avis/04d54.pdf> (última consulta 31 de enero 2013)



interpretación que hace la Autoridad de Competencia francesa de la jurisprudencia comunitaria sobre la doctrina de las “essential facilities”.

Para determinar si hay abuso de dominio en la negativa a conceder la licencia, el órgano francés se remite a la doctrina de los casos Magill, Bronner, IMS Health y Microsoft y examina las siguientes cuatro condiciones:

1º La negativa debe referirse a un producto o servicio indispensable para acceder al mercado.

2º La negativa debe conllevar la eliminación de la competencia.

3º La negativa debe impedir la aparición de nuevos productos.

4º La negativa debe carecer de una justificación objetiva.

La indispensabilidad del producto es interpretada como “la ausencia de actuales o potenciales sustitutos<sup>743</sup>”. A juicio del *Conseil*, el programa “Fair Play” no es indispensable en el mercado de música online porque:

a) La música online no se disfruta únicamente en reproductores musicales digitales. La mayoría de los usuarios escuchan la música en sus Pcs. VirginMega puede vender a otros consumidores distintos de los que usan iPods;

b) Los usuarios de iPod pueden descargar música de la plataforma de VirginMega y luego transferirla a sus iPods copiándola previamente en un CD;

c) Muchos reproductores de música digital están apareciendo en Francia. El tamaño del mercado potencial de Virgin está en expansión<sup>744</sup>.

La eliminación de la competencia, es interpretada como la eliminación total de la misma<sup>745</sup>, y esta circunstancia no se produce porque el mercado está en plena

---

<sup>743</sup> *Ibidem*, ap. 96: “(...) le caractère indispensable de l'accès a une facilité essentielle, notamment le fait qu'il ne doit pas exister de substitut réel ou potentiel réaliste.”

<sup>744</sup> *Ibidem*, ap. 97.

<sup>745</sup> *Ibidem*, vid. ap. 68, 70 y 96. En especial, la referencia al caso Microsoft del apartado 70 in fine: “ (...) le caractère avéré du risque d'élimination de la concurrence et sur la démonstration du lien causal entre l'évolution du marché et l'avantage de Microsoft en terme d'interopérabilité avec Windows”.

ebullición<sup>746</sup>. De lo cual también deduce el Consejo que no hay perjuicio para el consumidor.

Tampoco la aparición de nuevos productos se ve comprometida por el comportamiento de Apple, porque a juicio del Consejo, este requisito no se cumple en realidad, ya que VirginMega pretende sumarse a la oferta comercial de Apple, que está condicionada por el acceso a su programa MTP<sup>747</sup>. La negativa de Apple no es tampoco injustificada, ya que, aplicando un enfoque dinámico para evaluar esta circunstancia a través del examen del vínculo causal entre la conducta abusiva y las condiciones de competencia del mercado, la conducta de Apple puede explicarse por la superioridad de su oferta comercial<sup>748</sup>.

Este caso ejemplifica las carencias que presenta el Derecho de defensa de la competencia en una situación donde las condiciones del mercado, en el momento que la negativa a proporcionar la información interoperable tiene lugar, pueden no cumplir con las circunstancias excepcionales de la doctrina de las “essential facilities”.

Si bien Apple ostentaba una posición de dominio en el mercado de la música online, ni el acceso ni el uso de las interfaces interoperables del programa Fair Play de Apple se podría haber garantizado a través de la excepción del artículo 6 de la Directiva de Software ni a través del artículo 102 TFUE.

Desde un punto de vista político, la capacidad de Apple para limitar la interoperabilidad y así aumentar la venta de sus dispositivos iPod, no redundaría en la consecución de un nivel de bienestar óptimo para el consumidor y es susceptible de dañarlo más que de beneficiarlo. La mera existencia de diferentes MTP y diferentes códigos conlleva costes adicionales tanto para los consumidores como para los demás

---

<sup>746</sup> *Ibidem*, ap. 100. El Consejo menciona la entrada de dos nuevos competidores en el mercado, la existencia de nuevos potenciales competidores y la gran presión existente sobre los márgenes de beneficio.

<sup>747</sup> *Ibidem*, ap. 101: “A titre subsidiaire, ou peut aussi relever que la condition posée par la Cour de justice dans l'arrêt IMS Health n'est pas non plus remplie, VirginMega n'ayant pas déclaré vouloir proposer un produit ou un service nouveau qu'Apple ne souhaiterait pas offrir et dont la commercialisation serait conditionnée à un accès au DRM d'Apple.”

<sup>748</sup> *Ibidem*, ap. 100.

productores de dispositivos<sup>749</sup>. Si el programa Fair Play de Apple pudiera haber sido descompilado en condiciones menos estrictas que las propuestas por el artículo 6 de la Directiva, y los demás reproductores de música digital pudieran reproducir todos los formatos existentes en el mercado, la competencia entre Apple con su dispositivo iPod y los demás fabricantes, sería más agresiva, y posiblemente los consumidores podrían descargar legalmente música a un coste menor, beneficiando también al sector de producción musical<sup>750</sup>.

### C) The Mathworks.

El pasado 1 de marzo de 2012, la Comisión Europea abrió una investigación contra la empresa productora de software MathWorks<sup>751</sup>. La investigación se inició debido a la denuncia de otra empresa del sector, National Instruments, que alega que MathWorks se ha negado a concederle una licencia de usuario final y la información interoperable de MATLAB y Simulink, dos programas de ordenador que utiliza el denunciante para el diseño y simulación de sistemas de control comercial. National Instruments alega que, al negarse a conceder las licencias, MathWorks abusa de su posición de dominio, evitando que sus competidores descompilen o utilicen otras técnicas de ingeniería inversa de forma legítima para lograr interoperabilidad con esos dos programas.

Los dos programas de MathWorks', MATLAB y Simulink, se utilizan en los departamentos de investigación, desarrollo e innovación de un amplio rango de organizaciones y sectores como el aeroespacial y defensa, automatización y maquinaria, generación de energía, electrónica y comunicaciones, automoción y finanzas. La razón es que el software de MathWorks permite diseñar y desarrollar una gran variedad de productos avanzados en electrónica del automóvil, de aviónica, control de vuelo y propulsión en aeronáutica, sistemas de posicionamiento y guiado de satélites,

---

<sup>749</sup> U., GASSER, J., PALFREY et al., “iTunes. How Copyright, Contract and Technology Shape the Business of Digital Media – A Case Study”, Berkman Center for Internet & Society at Harvard Law School, Junio, 2004, pág. 47.

<sup>750</sup> Curiosamente, en octubre del 2006, un hacker noruego, Jon Lech Johansen anunció que además de deshabilitar el programa FairPlay, lo había descifrado mediante ingeniería inversa. Su empresa “Double Twist Ventures” se ofrecía a licenciar el uso del sistema FairPlay a las compañías que quisieran ofrecer sus productos para ser reproducidos en el iPod sin tener que firmar un acuerdo de distribución con Apple.

<sup>751</sup> Caso MathWorks, número 39840.

telecomunicaciones y otros equipos electrónicos, maquinaria industrial y dispositivos médicos, etc. en distintas plataformas hardware, microcontroladores, DSPs o FPGAs<sup>752</sup>.

La apertura de esta investigación, de la que a fecha de cierre de este trabajo no hay más noticias, pone de nuevo la cuestión de la interoperabilidad y sus efectos para la innovación en manos de las Autoridades de Competencia.

### *2.2.1. La indispensabilidad de la información interoperable en Microsoft y su relación con la descompilación.*

Con carácter general, la indispensabilidad de una “essential facility” se ha definido como la que no presenta alternativas posibles y por tanto, resulta imposible crear una segunda infraestructura<sup>753</sup>.

El epicentro del análisis sobre la indispensabilidad en el caso Microsoft se basa en que la información interoperable exige que el comportamiento de su propietario no tenga como consecuencia impedir “la viabilidad de la competencia en el mercado<sup>754</sup>”.

Para analizar esta cuestión, la Comisión hace referencia al artículo 6 de la Directiva<sup>755</sup>, destacando la indispensabilidad de la información necesaria para alcanzar la interoperabilidad. En ningún momento la Comisión valora la efectividad de dicha norma como alternativa de un competidor para acceder a la información interoperable, (lo cual por otra parte constituiría una extralimitación de sus funciones), y la Decisión de obligar a divulgar la información interoperable a Microsoft se fundamenta en el artículo 102 TFUE, cuyo ámbito va más allá de la actuación pasiva del propietario de un programa de ordenador ante un tercero que descompile el código objeto de su programa

---

<sup>752</sup> Vid. [www.mathworks.es](http://www.mathworks.es) (última consulta 31 de enero 2013).

<sup>753</sup> R., WHISH, D., BAILEY, *Competition Law*, Oxford, 7ª ed., Oxford University Press, 2011, pág. 694. Aunque el apartado 80 de la Comunicación de la Comisión — Orientaciones sobre las prioridades de control de la Comisión en su aplicación del artículo 82 del Tratado CE a la conducta excluyente abusiva de las empresas dominantes (Texto pertinente a efectos del EEE) DO n° C 045 de 24/02/2009 pág. 7, habla de un “producto o servicio objetivamente necesario para poder competir con eficacia en un mercado descendente”.

<sup>754</sup> Sentencia del Tribunal de Justicia (Gran Sala) de 17 de septiembre 2007, Microsoft v. Comisión, asunto T-201/04, *op.cit.*, ap. 229

<sup>755</sup> Decisión 2007/53/CE...*op.cit.*, ap. 747.

por razones de interoperabilidad<sup>756</sup>. Sin embargo, el concepto de indispensabilidad empleado por la doctrina de las “essential facilities” en el caso Microsoft, respalda no sólo la vital importancia de esta característica para la industria informática, sino que sirve de argumento crítico al artículo 6 de la mencionada Directiva.

Si bien algún autor (y Microsoft)<sup>757</sup> han afirmado que la Comisión utilizó un concepto de interoperabilidad más amplio que el definido por la Directiva y en concreto por el artículo 6 de la misma, esta afirmación, es zanjada a través del recurso al principio de jerarquía normativa de las normas de derecho de defensa de la competencia, tampoco puede sostenerse si se analiza desde la perspectiva del derecho de autor.

En cualquier caso, lo que haya de entenderse por interoperabilidad informática a efectos de la Directiva, lo indica la norma en el Considerando Décimo, no la excepción que limita las facultades exclusivas del titular del programa, y no existen distintos tipos de interoperabilidad<sup>758</sup>. Pero como se ha indicado en el capítulo tercero, las estrictas condiciones que el artículo 6 de la Directiva impone sobre la forma de acceso a la información interoperable y su uso, constituyen más un obstáculo que una salvaguarda para los desarrollos interoperables realizados mediante descompilación, impidiendo la viabilidad de la competencia

Los argumentos para defender la indispensabilidad de la interoperabilidad fueron por un lado, los estudios de mercado que mostraban claramente que la interoperabilidad era el factor decisivo que los consumidores tenían en cuenta a la hora de adquirir su material informático, y por otro, la Directiva de Software<sup>759</sup>. En esta Directiva, el legislador comunitario destaca la importancia de la interoperabilidad informática. En opinión de la Comisión, la Directiva trata la interoperabilidad como la

---

<sup>756</sup> Ibidem, ap. 747.

<sup>757</sup> La plena interoperabilidad, según Microsoft, puede realizarse gracias a la divulgación de información sobre las interfaces, a la que ya procede dicha empresa, en especial a través de su producto denominado “MSDN” o de las conferencias que organiza para los “Professional Developers”, o bien mediante varios otros métodos disponibles en el mercado. Microsoft afirma que el concepto de interoperabilidad entendido por la Comisión supone permitir la clonación de sus productos a los competidores, o de algunas de sus características, y que si tuviera que divulgar tal información, se lesionaría el libre ejercicio de sus derechos de propiedad intelectual o industrial, y se mermarían sus incentivos para la innovación (aps. 110 y 111 Sentencia). Vid. R.J., HART, “Interoperability Information and the Microsoft Decision”, EIPR, 28, 7, (2006) pág. 361.

<sup>758</sup> Vid. supra capítulo 1.1

<sup>759</sup> Sentencia del Tribunal de Primera Instancia (Gran Sala) de 17 de septiembre de 2007, Microsoft Corp., *op.cit.*, ap. 226.

esencia del uso efectivo de una computadora. Esto sugiere que, en el contexto de las políticas de la Unión, la interoperabilidad informática debiera ser valorada por encima de los derechos de propiedad industrial o intelectual sobre los protocolos de las interfaces<sup>760</sup>.

De modo similar al caso IMS Health, la indispensabilidad de la interoperabilidad no se fundamenta tanto en los derechos de propiedad intelectual sobre la información que permite alcanzarla, sino en que esa información a efectos de interoperabilidad, permite que los diseñadores de sistemas operativos para servidores competidores de Microsoft permanezcan presentes de forma viable en el mercado de esos sistemas operativos, comprometiendo el desarrollo tecnológico<sup>761</sup>.

Esta definición de la indispensabilidad respalda nuestra postura sobre que la descompilación es un proceso lento y muy costoso, que no resulta viable y por tanto al que los competidores no van a recurrir para copiar deliberadamente el programa, pero sí podría resultar una opción viable para acceder a la información interoperable o para labores de investigación y desarrollo, de no estar sometida a unas condiciones cuasi-imposibles<sup>762</sup>.

### *2.2.2. Abuso del derecho de autor y abuso de posición dominante en relación a la interoperabilidad.*

El abuso del derecho de autor y el abuso de posición dominante mantienen una relación que en algunas situaciones se superpondrá; cuando se produzca una situación de abuso de posición de dominio sobre un bien protegido por derechos de propiedad intelectual, esa conducta puede a la vez suponer un abuso del propio derecho de exclusiva. Sin embargo, la situación inversa no se produce: incluso cuando la situación de abuso de posición de dominio no sucede, el abuso del derecho de propiedad exclusiva puede tener lugar.

El ordenamiento comunitario proporciona dos mecanismos para garantizar el equilibrio de esta relación: un equilibrio interno entre la exclusividad y el acceso dentro

---

<sup>760</sup> E., FOX, “The EC Microsoft case ...”, *op.cit.*, pág. 277.

<sup>761</sup> Sentencia del Tribunal de Primera Instancia (Gran Sala) de 17 de septiembre de 2007, Microsoft Corp., *op.cit.*, ap. 229: “(...) si (...) el grado de interoperabilidad existente no permite que los diseñadores de sistemas operativos para servidores competidores de Microsoft permanezcan presentes de forma viable en el mercado de esos sistemas operativos, de ello resultaría que se perjudica el mantenimiento de una competencia efectiva en ese mercado.”

<sup>762</sup> Vid. *supra*, capítulo 1.3 y capítulo 3.2.3.

de la normativa de los derechos de propiedad intelectual junto con la posibilidad de contar con el límite externo del derecho de defensa de la competencia. Una muestra de este doble sistema de control es el tratamiento de la cuestión descompilación en virtud del artículo 6 de la Directiva Software. Incluso si las normas del derecho de propiedad intelectual pudieran proporcionar un método alternativo de resolución de la disputa, si la conducta del titular de los derechos transgrede la prohibición del artículo 102 TFUE, el derecho de defensa de la competencia puede proporcionar un remedio para corregirlo.

Esto plantea la interesante cuestión de si, en especial respecto de los programas de ordenador, el derecho de autor debe reformarse para reducir al mínimo la intervención de la normativa de defensa de la competencia o debería dejarse en manos del Derecho de defensa de la competencia para que pueda intervenir rápidamente cuando se infrinjan sus normas a través del ejercicio anticompetitivo de un derecho de propiedad intelectual<sup>763</sup>. Porque aunque ambos regímenes buscan mantener el bienestar social, las estrategias empleadas difieren. El derecho de defensa de la competencia corrige conductas mientras los derechos de propiedad intelectual e industrial conceden derechos.

En situaciones de productos vinculados, como sucede con la interoperabilidad informática, la defensa de la competencia, a través del abuso de posición dominante y el abuso del derecho de exclusiva se centran en aspectos distintos.

Donde el abuso de derecho identifica un producto protegido por derecho de autor y determina cómo su titular utiliza sus facultades para vincular su producto con otros, el derecho de la competencia evalúa la situación del mercado de los productos implicados y el efecto de su combinación. En un entorno óptimo desde el punto de vista económico, el derecho de defensa de la competencia es más apropiado a fin de determinar la equidad para comercializar productos protegidos por derecho de autor y evaluar las externalidades del mercado en juego<sup>764</sup>.

El papel tutelar del Derecho de defensa de la competencia sobre el Derecho de autor a través de la doctrina de las “essential facilities” puede resultar útil para garantizar la distinción entre la existencia y el ejercicio de los derechos de propiedad intelectual y corregir situaciones en las que el titular de un derecho de autor con una posición dominante en el mercado ejercita su *ius prohibendi* de manera anticompetitiva.

---

<sup>763</sup> S., ANDERMAN, S., “Microsoft v. Commission and ...”, *op.cit.*, pág. 272.

<sup>764</sup> L., GUIBAULT, *Copyright limitations and contracts ...*, *op.cit.*, pág. 190

Sin embargo una utilización excesivamente amplia de la mencionada doctrina puede llevar a resultados no deseados, perjudiciales para la competencia y, en última instancia, para el consumidor. Una interpretación flexible de esta doctrina puede dar lugar a una importante reducción del contenido de los derechos de propiedad intelectual (e industrial), porque cuando las circunstancias lo requieran, una empresa dominante puede ser obligada a licenciar tecnología esencial o a compartir activos con otro competidor, siempre que los productos que éste quiera introducir en el mercado sean distintos a los de la empresa dominante y exista para ellos una demanda potencial<sup>765</sup>.

### 2.2.3. *¿Son las licencias obligatorias la solución más equilibrada para el debate de la interoperabilidad?*

Como hemos visto a lo largo del trabajo, algunas plataformas informáticas publican sus interfaces, mientras que otras las mantienen en el más estricto de los secretos, y otras optan por licenciar esta información, sea en condiciones más o menos abiertas<sup>766</sup>. Que el software sea “privativo” lleva implícita la negativa a compartir con otros la información interoperable. Descompilar un programa, como alternativa de acceso a la información interoperable puede, aunque remotamente, conllevar también el acceso a la “caja fuerte” del programa, es decir, a más información que la necesaria a efectos de interoperabilidad. Como efecto colateral de este proceso, la descompilación puede poner a disposición de terceros información que sí está protegida por el derecho de autor sobre el programa y que puede ser utilizada para la creación de productos que compitan directamente con el programa descompilado. Estas consecuencias tan desagradables podrían evitarse si el titular del programa decidiese licenciar solo la información interoperable contenida en el código fuente que el potencial descompilador vaya a necesitar.

Partimos de la premisa de que el Derecho de defensa de la competencia tiene entre sus objetivos la creación de un mercado eficiente, en tanto que la propiedad intelectual (e industrial) concede una propiedad exclusiva por un período de tiempo

---

<sup>765</sup> F., HERNÁNDEZ RODRÍGUEZ, “Derecho de autor y abuso...”, *op.cit.*, pág 334; K., MCMAHON, “Interoperability: indispensability and special responsibility in High Technology Markets”, *Tul. J. Tech. & Intell. Property*, vol. 9, 2007, págs. 123 (171); M.A., DIZON, “Decompiling the Software Directive, the Microsoft CFI case and the i2010 strategy: How to reverse engineer an international interoperability regime”, *CTLR* 14, 2008, pág. 213 (225).

<sup>766</sup> P. SAMUELSON, S. SCOTCHMER, “The Law and Economics...”, *op.cit.*, pág. 1616.



limitado, pudiendo constituir una barrera de entrada a un mercado competitivo. En esta situación de conflicto, las licencias obligatorias pueden servir como una herramienta apropiada para lograr el equilibrio entre estas dos ramas del ordenamiento jurídico.

Aplicando las anteriores afirmaciones al debate de la interoperabilidad, el titular del derecho de autor sobre el programa podría asegurar la exclusividad de su producto si opta por licenciar bajo la condición de exclusividad<sup>767</sup>. Sin perjuicio de que el mero hecho de negarse a licenciar no constituye una conducta anticompetitiva *per se*, sin embargo, licenciar el código del programa a ingenieros informáticos para que escriban programas para un sistema operativo o una plataforma determinada, sí podría constituir una conducta anticompetitiva en el caso de que, por ejemplo, el licenciante prohíba a esos ingenieros escribir programas para otros sistemas operativos o plataformas, competencia de la licenciante<sup>768</sup>. Además de estar dificultando la entrada de nuevos competidores, el licenciante estaría recuperando las inversiones realizadas en su software a través de las regalías de la licencia<sup>769</sup>.

Si la preocupación de los titulares de programas que no publican la información de las interfaces, que se niegan a licenciar esta información y que defienden la prohibición de ingeniería inversa es no recuperar la inversión realizada en su programa a fin de seguir innovando<sup>770</sup>, ante una situación como la descrita: ¿cuándo son suficientemente elevados los costes de ingeniería inversa para los competidores?

En el caso Microsoft, esta compañía logró que su sistema operativo Windows se convirtiese en un estándar de facto para cualquiera que quisiese crear un programa de

---

<sup>767</sup> En los casos Sega y Nintendo, ambas compañías trataron de mantener la privacidad de sus interfaces prohibiendo a sus licenciatarios crear videojuegos para otras plataformas que no fueran las suyas. Sega interpuso la demanda contra Accolade porque desarrolló videojuegos para la consola de Sega sin haber negociado una licencia y además, adaptó los videojuegos de Sega para que pudieran ejecutarse en consolas que fueran de Sega. Vid. supra. capítulo 2.2.2. y casos Sega v. Accolade, *op.cit.*, pág. 1510 y Atari v. Nintendo, *op.cit.*, pág. 832.

<sup>768</sup> S., BISHOP, M., WALKER, *The Economics of EC Competition Law: Concepts, Application and Measurement*, London, 2ª ed., Sweet & Maxwell, 2002, pág. 173-174.

<sup>769</sup> Como indican MORITZ y TYBUSSECK, en el caso IBM v. Fujitsu, (International Business Machines Corp. v. Fujitsu Limited, Opinion of 15 September 1987, case no. 13T-117-063-85) los árbitros reconocieron el interés legítimo del creador del programa original a participar en la explotación que de él pudiera hacer un nuevo competidor interoperable con el original, cuando el segundo había logrado la interoperabilidad gracias a la descompilación. H., MORITZ, B., TYBUSSECK, B., *Computersoftware: Rechtsschutz und Vertragsgestaltung – Eine fächerübergreifende Darstellung nach deutschen und EG-Recht*, München, 2ª ed., Beck, 1992, pág. 81. Comentarios sobre el caso IBM v. Fujitsu, entre otros, vid.: A., SHELLEY, “ An Alternative Approach to Computer Pirating Disputes, the Mnookin-Jones Settlement: IBM v. Fujitsu”, *Temp. Int'l & Comp. L.J.* 3,(1989), pág. 113; A., STORK, “The use of arbitration in copyright disputes: IBM v. Fujitsu”, *High Technology Law Journal*, (1987), pág. 1.

<sup>770</sup> Vid. supra. Cap. 3.2.2 y Cap. 2.3.1.

aplicación que tuviese alguna posibilidad de triunfar en el mercado de los ordenadores personales. Cuando Windows fue lanzado al mercado y durante los años iniciales, todos los competidores de Microsoft disfrutaron de un período de interoperabilidad con Windows, porque divulgaba la información de las interfaces. Con el tiempo, la información interoperable se volvió más compleja y sofisticada y Microsoft empezó a modificarla con relativa frecuencia. Para un nuevo competidor que no quisiera optar por la vía de negociar un contrato de licencia con Microsoft, la opción de la descompilación del código de Windows sería una alternativa costosa en términos económicos y temporales. Sería una decisión de poco sentido común. La única razón que un nuevo competidor podría tener para recurrir a la descompilación de Windows sería evitar los términos que Microsoft impusiese en la licencia, como sucedió en el caso de Sega con Accolade.

Esto lleva a la siguiente pregunta: ¿podría el pago de los *royalties*, en los casos de concesión de licencias obligatorias, llegar a conciliar las consecuencias de la cuestionada excepción de descompilación con la garantía de una competencia justa y sin distorsiones en el mercado interno que exige el Tratado de la Unión Europea?

Aunque algún autor defiende esta postura<sup>771</sup>, y parece que la Agenda Digital para Europa 2010-2020 está considerando esta opción<sup>772</sup>, en nuestra opinión la solución de obligar a licenciar la información interoperable no debiera ser la regla general, sino sólo la excepción en determinadas circunstancias.

Desde el punto de vista del derecho de autor, la protección de la información contenida en las interfaces no forma parte del objeto de protección de la Directiva. La reciente jurisprudencia comunitaria de los asuntos C-393/09 y C-406/10 sobre interfaces, ha dejado claro que, por un lado, que las interfaces gráficas de usuario no están protegidas por el derecho de autor como parte del programa<sup>773</sup>, y que los elementos funcionales del programa no forman parte del objeto de protección de la

---

<sup>771</sup> P., HEINDL, *A Status Report...*, *op.cit.*, pág. 199

<sup>772</sup> La Acción 25 de la Agenda Digital consiste en “identify and assess means of requesting significant market players to licence information about their products or services”. Con esta acción, la Comisión pretende: “Examine the feasibility of measures that could lead significant market players to license interoperability information.” Vid.: <http://ec.europa.eu/digital-agenda/en/pillar-ii-interoperability-standards/action-25-identify-and-assess-means-requesting-significant> (última consulta 1 de junio, 2013)

<sup>773</sup> Vid. supra. capítulo 3.C.1. Un comentario sobre la protección de las interfaces gráficas de usuario y la sentencia del Tribunal de Justicia (Sala Tercera) de 22 de diciembre de 2010, asunto C-393/09, *op.cit.*, en B., GONZALEZ OTERO, “Protección de las interfaces gráficas de usuario de programas de ordenador en el ámbito de los derechos de autor: evolución y perspectivas”, ADI 31 (2011), pág. 205.

Directiva Software<sup>774</sup>. Si a esto añadimos la sentencia del caso C-604/10 sobre originalidad, que puede aplicarse a los programas de ordenador y según la cual “ese criterio (el de originalidad) no se cumple cuando la constitución de la obra (base de datos) es dictada por consideraciones técnicas, reglas o exigencias que no dejan lugar a la libertad creativa<sup>775</sup>”, la conclusión es que las interfaces que continene la información interoperable, no están protegidas por derecho de autor.

Por lo tanto, el debate de la interoperabilidad y las condiciones de las licencias a las que los nuevos competidores buscan escapar, tendría una solución más proporcionada dentro del derecho de autor si las condiciones para realizar la descompilación del programa fuesen más laxas, que en sede de derecho de defensa de la competencia, a través del recurso a las licencias obligatorias.

Es innegable que el sistema de licencias obligatorias que deriva de la aplicación de la doctrina de las circunstancias excepcionales (doctrina de las “essential facilities”) va más allá de la mera interoperabilidad entre programas y es una herramienta útil cuando se trata de proteger la innovación. También que los principios rectores del Derecho de defensa de la competencia y del Derecho de autor, o de los derechos de Propiedad Intelectual e Industrial, parten de fundamentos distintos. Sin embargo, convertir esta herramienta en la regla general podría agrandar el escenario de inseguridad jurídica que la Directiva Software ha establecido sobre la interoperabilidad. Si las partes implicadas, el titular del programa y el nuevo competidor, no están dispuestos a negociar los términos de acceso a la información interoperable, ¿cómo podrán la Comisión o los tribunales proceder a determinar y supervisar las condiciones de acceso<sup>776</sup>? Establecer condiciones justas y razonables en el sector de las TICs, donde no existen precedentes del mercado<sup>777</sup>, puede ser una tarea muy compleja; incluso más cuando la información interoperable de un nuevo programa no ha sido proporcionada con anterioridad.

---

<sup>774</sup> Sentencia del Tribunal de Justicia (Gran Sala) de 2 de mayo de 2012, asunto C-406/10, SAS Institute Inc..., *op.cit.*, vid. supra. Cap. 3.C.2.

<sup>775</sup> Sentencia del Tribunal de Justicia (Sala Tercera) de 1 de marzo de 2012, asunto C-604/10, Football Dataco Ltd., *op.cit.*, ap. 39.

<sup>776</sup> F., LÉVÊQUE, “Innovation, Leveraging and Essential Facilities: Interoperability Licensing in the EU Microsoft Case” *World Competition* 28, (2005) pág. 71 (87).

<sup>777</sup> Vid. en general, D.G., Swanson, W.J., Bauriol, “Reasonable and Nondiscriminatory (RAND) Royalties, Standards Selection, and Control of Market Power”, *Antitrust Law Journal* 73, (2005), pág. 1.

Es más, aplicar la opción de las licencias obligatorias a los casos de descompilación como regla general, podrían poner en riesgo el objetivo de la Unión de reforzar la competitividad de la industria comunitaria y garantizar un régimen de competencia no distorsionada en el mercado común. Podría incluso vulnerar el principio de proporcionalidad contenido en el artículo 5.4 del Tratado de la Unión, según el cual “el contenido y la forma de la acción de la Unión no excederán de lo necesario para alcanzar los objetivos de los Tratados”<sup>778</sup>.

Si una compañía dominante no proporciona la información necesaria para alcanzar interoperabilidad, es importante que las otras compañías tengan la libertad para usar herramientas de ingeniería inversa, como la descompilación. Ahora bien el enfoque de este problema basado en una solución unilateral de licencias obligatorias podría abrir la puerta a conseguir la interoperabilidad principalmente cuando el programa de destino sea de alguna de las grandes compañías de software<sup>779</sup>, lo cual podría llegar a poner en riesgo la innovación del sector.

La solución a este debate debería articularse a través de una combinación de políticas entre el derecho de defensa de la competencia y el derecho de autor, y que, en relación al derecho de autor deber comenzar por una revisión del artículo 6 de la Directiva.

En situaciones donde sólo el acceso a las interfaces del programa se ve comprometido, y no su uso, como cuando estén protegidas por otro derecho de exclusiva como una patente, o cuando el titular del programa se niega a licenciar pero no ostenta una posición dominante en el mercado, permitir la descompilación en condiciones menos restrictivas que las actuales para acceder a la información interoperable, e incluso para fines de investigación, podría no sólo animar a la empresa que impide el acceso a negociar para evitar que los competidores tenga la posibilidad de acceder al “corazón” del programa, sino que también podría servir de estímulo a la divulgación de nuevos programas, cumpliendo con la finalidad de la Directiva.

La doctrina de las “essential facilities” sí podría ser una solución en situaciones donde no el acceso, sino el uso de la información interoperable se viera comprometido por parte de una empresa que ostenta una posición de dominio. La excepción del

---

<sup>778</sup> Tratado de la Unión Europea, OJ C 83, 30.3.2010

<sup>779</sup> J. MARLY, *Urheberrechtsschutz für Computersoftware...*, *op.cit.*, pág. 328.

artículo 6<sup>780</sup>, no proporciona un derecho a usar fuera de los supuestos que establece. Por tanto, si la información obtenida con la descompilación está protegida por otro derecho de exclusiva, como una patente, la autorización del titular (la licencia) es necesaria para usar dicha información en nuevos desarrollos comerciales. De manera que incluso en situaciones donde se haga uso lícito de la excepción de la descompilación y se tenga éxito, el competidor *puede no poder* crear un programa interoperable si el titular se niega a concederle una licencia. En una hipotética situación como esta, si quien se niega a conceder la licencia ostenta una posición de dominio, la doctrina de las “essential facilities” podría solventar el problema.

No debemos olvidar que en situaciones donde existan alternativas al uso de la doctrina de las “essential facilities”, aplicarla puede incentivar el parasitismo económico y castigar la innovación y la inversión y, al final, quien se verá perjudicado es el consumidor que dejará de tener dos productos competitivos a su alcance<sup>781</sup>.

### **3. La relación entre la Directiva y el derecho contractual en materia de interoperabilidad.**

#### *3.1. El principio de la libertad contractual ante el Derecho de autor en el caso del software.*

La protección del programa de ordenador a través del derecho de autor conlleva la atribución a su titular de un derecho de exclusividad sobre el mismo. Esto implica que sólo él podrá realizar o autorizar la realización de actos de explotación sobre el programa, salvo en aquellos casos en los que el ordenamiento establezca una excepción al derecho de exclusividad.

Esta autorización se materializa en un contrato de licencia de derechos, en el caso de los programas de ordenador, denominadas “licencias de uso”. Su tipología es muy amplia, desde las que son negociadas personalmente en materias de obligaciones de uso y mantenimiento, pasando por las licencias de distribución y terminando con las licencias en masa o despersonalizadas<sup>782</sup>. Esta diferente tipología responde a las

---

<sup>780</sup> Tal como está configurada ni en las propuestas de reforma que se analizarán en el capítulo 5.

<sup>781</sup> A., CALVO CARAVACA, J., RODRÍGUEZ RODRIGO, *La Doctrina de...*, *op.cit.*, pág. 190.

<sup>782</sup> También conocidas como licencias *click-wrap*, *shrinwrap* y *broweswrap*.

características específicas del programa objeto del contrato, o a la demanda del consumidor, o normalmente, a una combinación de las anteriores circunstancias.

Sin embargo, todas estas licencias cuentan con el mismo problema en relación al derecho de autor sobre el programa: la utilización de licencias en lugar de compraventas<sup>783</sup>. Ello significa que el titular del derecho de autor autoriza a un tercero a llevar a cabo determinados actos de explotación, descritos en el contrato, necesarios para el uso o acceso a la obra, y la autorización viene normalmente referida al derecho de reproducción<sup>784</sup>. Si bien se ha argumentado a favor de esta postura que fomenta la flexibilización de los modelos de distribución del software y la discriminación de precios<sup>785</sup>, también puede chocar con consideraciones de tipo político implícitas en los regímenes de propiedad intelectual. Por ejemplo, no existe regulación internacional para este tipo de contratos, ni tampoco resultan aplicables las normas especiales establecidas por los ordenamientos nacionales en sus leyes de propiedad intelectual para las licencias de derechos de explotación, ya que la finalidad de estas licencias de uso es diferente<sup>786</sup>. Es más, debido a la proliferación de la computación en la nube, algunas empresas fabricantes de software ya se han lanzado a otro modelo de distribución que consisten en el alquiler del programa, en lugar de las licencias de uso, lo cual planteará aún más preguntas<sup>787</sup>.

Todo eso plantea preguntas como si esta clase de contratos pueden hacerse valer igual que cualquier otra clase de contrato, y sobre todo, si esta clase de contratos, regidos a priori por el principio de libertad contractual y la voluntad de las partes, pueden anular y redefinir el marco establecido por el derecho de autor sobre el

---

<sup>783</sup> Sin perjuicio de lo dispuesto sobre el agotamiento del derecho de distribución y la consideración de reventa de segunda mano en el caso *Usedsoft* por el Tribunal de Justicia de la UE. Vid. Sentencia del Tribunal de Justicia (Gran Sala) de 3 de julio de 2012, asunto C-128/11, *UsedSoft GmbH contra Oracle International Corp.*, ap. 83-85.

<sup>784</sup> A., LÓPEZ-TARRUELLA MARTÍNEZ, “Las licencias de uso de bienes digitales: El difícil equilibrio entre los titulares de los derechos y los usuarios”, *SCRIPT-ed* (3)4 June (2006), pág. 437 (443).

<sup>785</sup> Sobre las razones por las que los fabricantes de software prefieren la licencia de derechos a la venta del programa vid. M.A., LEMLEY, P.S., MENELL, R.P., MERGES, et al., *Software and Internet Law*, New York, 5ª ed., Wolters Kluwer Law and Business, capítulo 5, pág. 227.

<sup>786</sup> Sólo el Derecho español cuenta con una norma especial en materia de licencias para programas de ordenador, contenida en el artículo 99, párrafo segundo de la LPI: “(...) cuando se produzca cesión del derecho de uso de un programa de ordenador se entenderá salvo prueba en contrario, que dicha cesión tiene carácter no exclusivo e intransferible, presumiéndose, asimismo, que lo es para satisfacer únicamente las necesidades del usuario”.

<sup>787</sup> Adobe elimina el pago de una licencia única para poner todo su software en alquiler. Disponible en: <http://www.genbeta.com/herramientas/adobe-creative-cloud-sus-alquileres-el-fin-de-la-pirateria-y-el-riesgo-de-este-nuevo-modelo>; <http://www.elmundo.es/blogs/elmundo/el-gadgetoblog/2013/05/07/photoshop-si-pero-a-plazos.html> (última consulta 1 de junio 2013).

programa, es decir, ¿deberíamos permitir que el “derecho privado” resultado de esas licencias sustituya consideraciones de política públicas, en particular las que afectan a la descompilación y la interoperabilidad?<sup>788</sup>

El usuario debe aceptar una licencia de uso con las condiciones establecidas por el titular del derecho de autor antes de poder acceder al propio programa, y por ende, antes de tener derecho a analizar, investigar o descompilar el programa<sup>789</sup>. Por tanto, si no hay acuerdo de licencia aceptado, tampoco habrá garantía mínima de los actos permitidos a efectos de interoperabilidad.

Las excepciones legales que garantizan la interoperabilidad tienen como fin equilibrar los intereses de los titulares de derechos y los de la sociedad en general<sup>790</sup>. Si pueden ser derogadas por contrato, ese equilibrio se rompe. Se sobreprotegen los intereses de los titulares de derechos en detrimento de los de la comunidad en general y de los usuarios.

### 3.2. *La imperatividad del artículo 8 de la Directiva.*

El art. 8 de la Directiva sobre protección jurídica de los programas de ordenador establece que cualquier disposición contractual que sea contraria a los apartados 2 y 3 del art. 5 y al art. 6 se considerará nula y sin valor ni efecto alguno. A estas excepciones se las ha denominado el “núcleo imperativo” de la Directiva.

Lo primero que hay destacar es que tanto la doctrina como la jurisprudencia sobre la cuestión de la relación entre excepciones legales imperativas y el principio de libertad contractual es muy escasa<sup>791</sup>. Únicamente una sentencia alemana ha reconocido expresamente el carácter imperativo de la norma<sup>792</sup>.

Atendiendo a la norma comunitaria, la regla general que puede extraerse de la Directiva es que deja la cuestión de permitir o prohibir actos de ingeniería inversa al derecho contractual de cada Estado miembro. Sólo en relación a los actos de

---

<sup>788</sup> Este trabajo no tiene en cuenta la posible aplicación de las normas sobre protección de los consumidores, porque el ámbito del trabajo se circunscribe a un escenario empresarial, entre potenciales competidores, es decir business-to-business.

<sup>789</sup> H., ULLRICH, E., KÖRNER, *Der International ...*, op.cit., pág. 76,

<sup>790</sup> T., VINJE, “Copyright Imperilled?”, *EIPR* 4, 1999, pág. 192 (195); C., CORREA, “Fair use in the Digital Era”, *IIC* 33(5), 2002, pág. 570 (582).

<sup>791</sup> Así lo reconoce en el Informe de la Oficina de Propiedad Intelectual del Reino Unido (*Intellectual Property Office*) *The Relationship Between Copyright and Contract Law* 2010 (04), pág. 7; disponible en: [www.ipo.gov.uk/ipresearch-relation-201007.pdf](http://www.ipo.gov.uk/ipresearch-relation-201007.pdf) (Última consulta 1 de junio, 2013).

<sup>792</sup> LG Mannheim, sentencia de 20 de enero de 1995, NJW 1995, pág. 3322 (3323) en Alemania.

descompilación del artículo 6, la realización de copia de seguridad del artículo 5.2 y el análisis de caja negra del artículo 5.3 la Directiva regula los posible conflictos entre la ella y el derecho contractual de los estados miembros<sup>793</sup>.

De acuerdo con el segundo párrafo del artículo 8, la excepción de la descompilación constituye *ius cogens*; en consecuencia no debiera otorgarse ningún efecto a las cláusulas que recogen las licencias de software en las que se establece la prohibición total de descompilar, realizar ingeniería inversa, desensamblar o reducir el programa a su forma comprensible por el ser humano. Salvo se acuerde libremente de otro modo, el licenciataria del programa podrá usarlo de acuerdo con los términos del contrato, entre los cuales lo dispuesto por el artículo 5(3) y el artículo 6 de la Directiva Software no pueden restringirse porque constituyen el umbral mínimo que el Derecho garantiza para analizar el programa<sup>794</sup>.

Aún así, la imperatividad de esta provisión es un tema controvertido para la doctrina por tres clases de argumentos:

En primer lugar, no hay apenas jurisprudencia ni de origen comunitario ni de los tribunales de los Estados miembros sobre esta cuestión<sup>795</sup>. Esto no ayuda a establecer los grados de prevalencia entre las excepciones de derecho de autor y la libertad contractual.

Alguna decisión de alguno de los Estados miembro objeto de estudio en este trabajo sí ha examinado la cuestión de la prevalencia en relación a la copia privada de la obra, y han declarado que no es un derecho<sup>796</sup>, lo cual no sirve aquí de mucha ayuda. Además, los asuntos que han tratado el tema de la copia privada se centraron en la obligación del titular de hacer pública la existencia y el funcionamiento de las medidas

---

<sup>793</sup> B. CZARNOTA, J. HART, *Legal Protection of...*, op.cit., pág. 96.

<sup>794</sup> H., ULLRICH, E., KÖRNER, *Der internationale ...*, op.cit., pág. 75.

<sup>795</sup> Si bien la sentencia alemana de 1995, (LG Mannheim, sentencia de 20 de enero de 1995, NJW 1995, pág. 3322 ) reconoce expresamente la imperatividad de la excepción de la descompilación, sin embargo, en Francia, en la sentencia del caso Fiducial, (Asunto La société Fiducial v. la société Développement professionnel spécialisé informatique (DPSI) ; et autres, sentencia de 20 de octubre de 2011 de la Corte de Casación (Cour de cassation - Première chambre civile), nº 975, (10-14.069)), el tribunal declaró que la descompilación no constituía una infracción porque, aunque se daban los requisitos para realizarla, la licencia de uso permitía la migración de datos (vid. supra, capítulo 3.3.2).

<sup>796</sup> Asunto François M., v. EMI France, sentencia de la Corte de Apelación de Versalles, Sala Primera, Sección Primera (Cour d'Appel, 1re Chambre, 1re Section), de 30 de septiembre 2004; asunto Stéphane P, UFC Que Choisir v. Universal Pictures Video France et Autres, sentencia del Tribunal Supremo, Sala de lo Civil, (Cour de Cassation, Première chambre civile) de 28 de febrero de 2006; -BverfG, 3. Kammer des Ersten Senats, sentencia de 25 de julio de 2005, asunto 1 BvR 2182/04 Eigentum und digitale Privatkopie, GRUR 2005, pág. 1032.



tecnológicas de protección sobre la obra, por ser esto considerado como una característica esencial del producto. La perspectiva desde la que los tribunales analizaron la cuestión fue más la del consumidor que la de la relación contractual establecida a través de un derecho de exclusiva. Las cláusulas de la licencia en particular no fueron examinadas por los tribunales. Por tanto, lo que esta jurisprudencia deja claro es que la incógnita acerca de la prevalencia del derecho contractual sobre las excepciones al derecho de autor en general sigue sin estar despejada.

En segundo lugar, la doctrina se ha cuestionado la capacidad de la Directiva Software para establecer excepciones imperativas sobre las licencias de programas de ordenador, cuando en el sector de las TICs las condiciones de estas licencias pueden ser únicamente definidas por contrato<sup>797</sup>. Tampoco queda nada claro hasta qué punto la naturaleza obligatoria de la excepción contenida en el artículo 6 puede eludirse por vía contractual entre partes que decidan someter el acuerdo a la ley de un Estado no miembro de la UE<sup>798</sup>.

En tercer lugar, los Estados miembro no han implementado este precepto de manera uniforme, con lo cual han creado diferentes niveles de protección dentro de la Unión Europea<sup>799</sup>.

Aún así, Considerando las diferentes maneras en que el segundo párrafo del artículo 8 de la Directiva Software ha sido implementado en los ordenamientos nacionales<sup>800</sup>, la postura de los distintos legisladores nacionales puede interpretarse tanto a favor como en contra del argumento de la imperatividad de la excepción de la descompilación.

Si bien en la práctica contractual de los últimos años, parece que la imperatividad de las excepciones sobre ingeniería inversa que garantizan la

---

<sup>797</sup> S., DUSOLLIER, *Droit d'auteur et protection...*, *op.cit.*, pág. 498.

<sup>798</sup> Como indica OHLY, "while the EC Software Directive declares contract clauses void which circumvent the provision on interoperability, shrink wrap licenses which prohibit reverse engineering have been held to be valid in the US (Bowers v. Baystate Technologies, Inc., 320 F.3d, 1317, 1323 et seq. (Fed. Cir. 2003)). Vid. A., OHLY, "Reverse Engineering: ...", *op.cit.*, pág. 545.

<sup>799</sup> L., GUIBAULT, *Copyright limitations and contracts ...*, *op.cit.*, pág. 219; L. GUIBAULT, "Wrapping Information in Contract: How Does it Affect the Public Domain?" en L. GUIBAULT, P.B., HUGENHOLTZ (eds.), *The Future of the Public Domain (Identifying the Commons in Information Law*, Kluwer Law International, Netherlands, 2006, pág. 97.

<sup>800</sup> L., GUIBAULT, *Copyright limitations and contracts ...*, *op.cit.*, pág. 220. El Informe Directiva 2000, *op.cit.*, en su pág. 15 declaraba al respecto que todos los Estados miembros habían implementado correctamente este precepto.

interoperabilidad sí es tomada en cuenta por los titulares de los programas, en el sentido de que los titulares incluyen cláusulas donde tras declarar prohibido cualquier acto de ingeniería inversa sobre el programa, exceptúan los actos que la ley aplicable al contrato permita por razones de interoperabilidad.

Sirvan como ejemplo:

- Licencia Photoshop CS3 End User License Agreement, cláusula 4.2:

“Except as permitted in Section 14, you may not modify, adapt or translate the Software. You may not reverse engineer, decompile, disassemble or otherwise attempt to discover the source code of the Software except to the extent you may be expressly permitted under applicable law to decompile only in order to achieve interoperability with the Software.”

- Licencia de uso de Android Software Development Kit License Agreement, cláusula 3.3:

“You may not use the SDK for any purpose not expressly permitted by this License Agreement. Except to the extent required by applicable third party licenses, you may not: (a) copy (except for backup purposes), modify, adapt, redistribute, decompile, reverse engineer, disassemble, or create derivative works of the SDK or any part of the SDK”.

Sin embargo, si volvemos a los Considerandos de la Directiva, en concreto al Dieciséis, donde se declara que “la protección de programas de ordenador al amparo de la legislación sobre derechos de autor debe entenderse sin perjuicio de la aplicación de otros tipos de protección”, ¿qué sucede si en la licencia el titular del programa define que la información interoperable tendrá la consideración de un secreto empresarial, prohibiendo su uso para cualquier propósito que no sean los definidos por la licencia?

En este caso, el artículo 8 de la Directiva no tendría ninguna relevancia, añadiendo al abanico de dificultades para el licenciataria que la normativa sobre secretos empresariales no ha sido objeto de armonización comunitaria y su régimen depende por tanto de lo establecido en los ordenamientos nacionales de cada Estado miembro.

Sirvan también como muestra:

- Licencia Photoshop CS3 End User License Agreement, cláusula 3:

“The Software and any authorized copies that you make are the intellectual property of and are owned by Adobe Systems Incorporated and its suppliers. The structure, organization and code of the Software are the valuable trade secrets and confidential information of Adobe Systems Incorporated and its suppliers. The Software is protected by law, including but not limited to the copyright laws of the United States and other countries, and by international treaty provisions. Except as expressly stated herein, this agreement does not grant you any intellectual property rights in the Software and all rights not expressly granted are reserved by Adobe and its suppliers.”

- Licencia de uso de Android Software Development Kit License Agreement, cláusula 3.2:

“You agree that Google or third parties own all legal right, title and interest in and to the SDK, including any Intellectual Property Rights that subsist in the SDK. "Intellectual Property Rights" means any and all rights under patent law, copyright law, trade secret law, trademark law, and any and all other proprietary rights. Google reserves all rights not expressly granted to you.”

#### **4. Propuestas de *lege ferenda* en relación con el artículo 6 y sus consecuencias.**

##### *4.1. Propuestas de lege ferenda.*

##### *4.1.1. La propuesta de una excepción general para la ingeniería inversa.*

La excepción de la descompilación de la Directiva Software ha sido objeto de una propuesta de modificación hace apenas dos años, en el “European Copyright Code” del *Wittem Project*<sup>801</sup>.

---

<sup>801</sup> *European Copyright Code*, elaborado por *The Wittem Group*, publicado en abril, 2010; disponible en: [www.copyrightcode.eu](http://www.copyrightcode.eu) (última consulta 1 de junio, 2013).

El *Wittem Project* es un proyecto de colaboración académica fundado en 2002. Movidos por la preocupación de la dirección que el derecho de autor ha tomado en los últimos veinte años, académicos de diferentes países de la Unión Europea, iniciaron este proyecto doctrinal para ayudar a promover la transparencia y la consistencia del derecho de autor comunitario. Como declaran en la Introducción del “European Copyright Code”, la voz de la doctrina no ha sido apenas escuchada por el legislador comunitario en relación al derecho de autor<sup>802</sup>.

Si bien desde 1991, el legislador comunitario ha aprobado siete Directivas sobre derechos de autor, el código que proponen no constituye una recodificación del derecho comunitario existente, también los Tratados Internacionales que afectan sustantivamente al derecho de autor, como el Convenio de Berna y el Acuerdo sobre los ADPIC<sup>803</sup>, son tenidos en cuenta ya todos los Estados de la Unión son partes contratantes de ambos.

La intención del *Wittem Group* es contribuir a establecer un cuerpo normativo transparente y consistente en el derecho de autor, que no sólo proteja los derechos morales y económicos de los autores sino que también sirva al interés público a través de la promoción de la producción y diseminación de obras literarias, artísticas y científicas.

Por ello, la solución que proponen a cuestiones escogidas, como la excepción de la descompilación o ingeniería inversa, se aparta considerablemente de la regulación establecida por la Directiva Software.

El Código reconoce que la rápida evolución de la tecnología ha originado la aparición de nuevas formas de explotación y usos de obras protegidas por derecho de autor antes impredecibles, y que por lo mismo, requieren un sistema de derechos y limitaciones más flexible.

La propuesta concreta para la descompilación se encuentra en el artículo 5.4.1.b. Su lectura ha de hacerse junto con el artículo 5.4.2, ambos bajo la rúbrica “Uses for the purpose of enhancing competition”:

- (1) The following uses for the purpose of enhancing competition are permitted without authorisation and without remuneration, to the extent justified by the purpose of the use:*
- b) use for the purpose of reverse engineering in order to obtain access to information, by a person entitled to use the work.*

---

<sup>802</sup> *European Copyright Code*, op.cit., pág. 5.

<sup>803</sup> *Ibidem*, pág. 6.

*(2) Uses of news articles, scientific works, industrial designs, **computer programs** and databases are permitted without authorisation, but only against payment of a negotiated remuneration, and to the extent justified by the purpose of the use, provided that:*

*(i) the use is indispensable to compete on a derivative market,*

*(ii) the owner of the copyright in the work has refused to license the use on reasonable terms, leading to the elimination of competition in the relevant market and*

*(iii) the use does not unreasonably prejudice the legitimate interests of the owner of the copyright in the work.*

Sin entrar en consideraciones de fondo como la clasificación de limitaciones y excepciones que propone el Wittem Group, quizás la propuesta presente demasiada indeterminación por varias razones:

–El artículo parece identificar “descompilación” con “ingeniería inversa”.

–No hay una sola mención al término “interoperabilidad”.

–Se habla de “acceso a información”, sin especificar a qué tipo de información.

–Es cierto que el artículo 6 de la Directiva Software, con sus 35 líneas y las condiciones tan restrictivas que impone al usuario legítimo dificultan mucho su interpretación. Sin embargo, solucionar este problema de complejidad del artículo 6 y del artículo 5(3) de la Directiva en dos líneas escasas, pueden situar al intérprete en el mismo lugar. Regular por exceso es tan perjudicial como quedarse cortos.

–Las condiciones para el uso no se refieren a la información del artículo anterior, sino a la obra en general, al programa.

–Estas condiciones del apartado 2, además, son una especie de mezcla entre la doctrina de las circunstancias excepcionales, del derecho de defensa de la competencia, y la doctrina de los tres pasos del derecho de autor.

Si bien el artículo 6 de la Directiva Software debiera ser revisado por el legislador, esta propuesta del “European Copyright Code”, quizás sea demasiado flexible. Aunque pone la atención en el punto neurálgico, la competitividad, garantizar la interoperabilidad informática permitiendo la ingeniería inversa sobre el programa con carácter general y sin más especificaciones puede llevar justo a lo contrario, al parasitismo y la piratería.

#### 4.1.2. La propuesta de un nuevo artículo 6 en dos niveles.

La propuesta más completa de modificar el artículo 6 en una excepción de dos niveles procede del Dr. SHEMTOV<sup>804</sup>. En la misma se recogen dos supuestos claramente diferenciados en los que se permite descompilar el programa: por un lado, lograr la interoperabilidad, y por otro, y aquí reside la principal novedad, cuando se quiera acceder a los elementos no protegidos del programa, con arreglo a lo dispuesto en el artículo 1.2 de la Directiva, por razones de interés público<sup>805</sup>.

*1. The authorization of the rightholder shall not be required where reproduction of the code and translation of its form within the meaning of Article 4 (a) and (b) are indispensable to obtain the information necessary to achieve the interoperability of an independently created computer program, **or an article, at present or in the future, with other programs**, provided that the following conditions are met:*

*(a) these acts are performed by the licensee or by another person having a right to use a copy of a program, or on their behalf by a person authorized to do so;*

*(b) the information necessary to achieve interoperability has not previously been readily available to the persons referred to in subparagraph (a); and*

*(c) these acts are **performed to the extent reasonably necessary to achieve interoperability of the independently created computer program, or article.***

*(d). The provisions of paragraph 1(a) – (c) shall not permit the information obtained through its application:*

*1. to be used for goals other than to achieve the interoperability of the independently created computer program **or article, subject to paragraph 2;***

---

<sup>804</sup> N., SHEMTOV, *The Legal Regulation of Decompilation of Computer Program...*, *op.cit.*, pág. 151.

<sup>805</sup> Sobre el texto vigente de la Directiva, hemos destacado en negrita, las modificaciones que propone SHEMTOV.

2. to be given to others, except when necessary for the interoperability of the independently created computer program, or article; for the avoidance of doubt, the terms “others” in this provision shall include others persons seeking to achieve interoperability of an independently created computer program or article;

3. to be used for the development, production or marketing of a computer program substantially similar in its expression, or for any other act which infringes copyright.

*(e) to the extent that the new program reproduces or translates the original program, it shall do so only to the extent necessary to enable the new program to interoperate with the original or another program.*

*2. The authorization of the rightholder shall not be required where reproduction of the code and translation of its form within the meaning of Article 4 (a) and (b) are indispensable to obtain access to unprotectable elements of a computer program within the meaning of Article 1(2), provided that the following conditions are met:*

*(a) these acts are performed in a fair and reasonable manner. In determining whether in any particular case these acts are performed in a fair and reasonable manner due regard shall be taken for, inter alia, the public interest in:*

- (1) education,*
- (2) science,*
- (3) research,*
- (4) innovation,*
- (5) preservation of competition.*

*(b) The acts specified in Article 4(a) and (b) shall be regarded as fair and reasonable provided that the following conditions are met:*

- (1) these acts are performed by the licensee or by another person having a right to use a copy of a program, or on their behalf by a person authorized to do so;*
- (2) the information defined in paragraph (2) has not previously been readily available to the persons referred to in subparagraph (1); and*
- (3) these acts are performed to the extent reasonably necessary to the information defined in paragraph (2).*

***(c) The acts specified in Article 4(a) and (b) shall not be regarded as fair and reasonable where they are carried out for goals other than to obtain access to the ideas and functional concepts contained in a computer program.***

*3. In accordance with the provisions of the Berne Convention for the protection of Literary and Artistic Works, the provisions of this Article may not be interpreted in such a way as to allow its application to be used in a manner which unreasonably prejudices the right holder's legitimate interests or conflicts with a normal exploitation of the computer program.*

De esta propuesta destacamos los siguientes aspectos:

En primer lugar, en relación a la consecución de la interoperabilidad, al incluir la mención “or an article” en el apartado 1, se despejan las dudas de si la descompilación permitida alcanza solo a la interoperabilidad entre programas, sino que también se permite para alcanzar la interoperabilidad entre programas y dispositivos.

Las referencias a “at present or in the future”, para aclarar el problema de interpretación del texto vigente cuando se refiere a “la interoperabilidad de un programa de ordenador creado de forma independiente con otros programas”, podrían interpretarse como que la excepción permite alcanzar la interoperabilidad a programas no sólo competitivos sino también sustituibles con el descompilado<sup>806</sup>, siempre que no se trate de una obra sustancialmente similar en su expresión o infrinja los derechos de autor, según dispone el apartado 1, letra (d)3 de la propuesta.

Con la introducción de la expresión en el apartado 1, letra (c), “performed to the extent reasonably necessary to achieve interoperability”, se pondría fin al complejo requisito vigente de que indispensabilidad de la descompilación esté limitada estrictamente a las partes necesarias para conseguir interoperabilidad. Como indicamos en el capítulo tercero, este requisito resulta incongruente porque presupone que la persona autorizada para descompilar tiene conocimiento previo sobre la ubicación de las interfaces. Con la modificación propuesta, la norma no recupera coherencia, sino que evitaría tener que traicionar el principio de interpretación restrictiva de las excepciones de derecho de autor al determinar cuáles son las partes necesarias, y sobretodo, la carga probatoria para la persona autorizada deja de ser cuasi imposible<sup>807</sup>.

---

<sup>806</sup> Vid. supra, capítulo 3.2.4.D.1.2)

<sup>807</sup> Vid. supra capítulo 3.2.4.D.4)



En relación a los usos de la información obtenida, el apartado propuesto aclara que además de lograr la interoperabilidad con un programa o dispositivo creado de forma independiente, los elementos no protegidos del programa no están sujetos a restricción, y además aclara qué ha de entenderse por “terceros” a los que se puede comunicar esa información.

La segunda parte del artículo propuesto, es nueva por completo, y con ella se soluciona la otra cuestión problemática de la descompilación: el acceso a las ideas y principios del programa.

Compartimos el objetivo pretendido, ya que con esta modificación, el uso libre de partes estandarizadas de una obra, como son las interfaces donde se contienen la información interoperable, no se vería obstaculizada por la incapacidad del sistema de derecho de autor para equilibrar los beneficios derivados de la promoción de la originalidad con los beneficios derivados de la utilización de expresiones idénticas, que pueden ser imprescindibles para la consecución de la interoperabilidad<sup>808</sup>, pero la forma no nos parece la más adecuada.

Aunque la descompilación afecta a las facultades exclusivas de reproducción y transformación de la obra, no hemos de olvidar, como se indicó repetidas veces a lo largo del trabajo, que es un proceso complejo, largo y costoso, al que los potenciales competidores sólo recurrirán como última opción, si no pueden acceder a las ideas y principios del programa de otro modo. Pero al incluir la referencia al carácter indispensable, la persona que descompile por alguna razón de interés público, ante un posible demanda por infracción del derecho de autor, se verá en la tesitura de tener que demostrar su carácter indispensable, que según el apartado (b) del artículo, números (2) y (3), se cumplirá cuando:

- la información de los elementos no protegidos no haya sido puesta a su disposición, y
- la descompilación tenga por objeto el acceso razonablemente necesario a esos elementos.

---

<sup>808</sup> M., LEHMANN, “Standardization and the EC ...”, *op.cit.*, pág. 367.

En nuestra opinión, la referencia al carácter indispensable de la descompilación, cuando se trata de información que el derecho de autor sobre el programa no protege, sería innecesaria.

En segundo lugar, nos parece muy arriesgado incluir entre las razones de interés público que justifican la excepción de la descompilación “el mantenimiento de la competencia”, por un claro motivo. Aunque la garantía de la interoperabilidad en la Directiva responde a un objetivo de competitividad en el mercado, ¿cómo se puede justificar en una situación concreta que la descompilación de un programa mantiene la competencia? ¿Tendría que acreditar la persona autorizada a descompilar, el mantenimiento de la competencia en el sector del mercado concreto en que opera el descompilador, o en el sector del mercado del programa que se descompila, si son distintos, o en un mercado secundario, o incluso en un potencial nuevo mercado? ¿Cabría prueba en contrario del titular del programa? La relación entre ambas ramas del ordenamiento jurídico es lo suficientemente delicada como limitarla a circunstancias excepcionales, como indica el Considerando Décimoséptimo de la Directiva.

En nuestra opinión, a fin de garantizar el acceso a los elementos no protegidos del programa a través del recurso a la descompilación, la propuesta que había realizado la ECIS en 1990, sigue siendo la mejor opción:

*“Reproduction and translation of a computer program, to the extent necessary to research, study or extract the unprotectible elements underlying the program shall not be restricted acts”<sup>809</sup>*

No debe perderse de vista que de acuerdo con la cláusula de cierre del artículo 6 en vigor, lo dispuesto en el mismo está sujeto a la aplicación de la Regla de los Tres Pasos<sup>810</sup>, con lo cual, cualquier uso de información adquirida a través de la descompilación para objetivos que no sean alcanzar interoperabilidad con un programa

---

<sup>809</sup> ECIS Report on Interoperability, “Protecting Computer Software while encouraging interoperability and innovation”, 1990, pág. 38, disponible en: <http://www.ccianet.org/CCIA/files/ccLibraryFiles/Filename/000000000667/ECIS%20Report%20on%20Interoperability%201990.pdf> (última consulta 1 de junio, 2013).

<sup>810</sup> Sobre la aplicación de la Regla de los Tres Pasos, vid., entre otros: A., KUR (ed.), *Intellectual Property Rights in a Fair World Trade System*, Cheltenham, Edward Elgar, 2011, pág. 208; Declaración por una interpretación equilibrada de la «Regla de los Tres Pasos» en el derecho de autor, ADI 28, 2007-2008, págs. 1509.

creado de forma independiente en las condiciones estipuladas, o la investigación, estudio y extracción de las ideas y principios del programa que perjudique los intereses legítimos del titular o sean contrarias a una explotación normal de la obra, estarían fuera del ámbito de lo permitido por la excepción.

Con esto se dejaría a los tribunales que, teniendo presente el objetivo de protección de la Directiva sobre protección jurídica de programas de ordenador, puedan valorar los intereses de las partes en juego en el caso concreto.

#### *4.2. Una enseñanza de derecho comparado: La reforma de la excepción sobre ingeniería inversa en la Ley sobre Propiedad Intelectual de Chile.*

Otra propuesta que podría tenerse en cuenta a fin de reformar el artículo 6 de la Directiva Software es la excepción sobre ingeniería inversa introducida en la Ley sobre Propiedad Intelectual de Chile en 2010.

Chile es parte de los principales Tratados Internacionales en materia de derechos de autor, como el Convenio de Berna, el Acuerdo sobre los ADPIC y el Tratado WCT. Precisamente, obedeciendo a los compromisos internacionales se llevó a cabo la reforma de la Ley sobre Propiedad Intelectual de 1970 en 2010<sup>811</sup>.

La Ley N° 20.435 de 4 de mayo de 2010, que modifica la Ley N° 17.336 sobre Propiedad Intelectual, incluye un catálogo de excepciones que fue elaborado teniendo especialmente presente la abundante experiencia internacional y los principios contenidos en diversos tratados internacionales multilaterales de la Organización Mundial de la Propiedad Intelectual (OMPI) y la Organización Mundial del Comercio (OMC)<sup>812</sup>.

Es su artículo 71 Ñ el que establece la excepción de la ingeniería inversa sobre programas de ordenador:

*Las siguientes actividades relativas a programas computacionales están permitidas, sin que se requiera autorización del autor o titular ni pago de remuneración alguna:*

*b) Las actividades de ingeniería inversa sobre una copia obtenida legalmente de un programa computacional que se realicen con el único propósito de lograr la*

---

<sup>811</sup> Sobre las razones de la reforma vid: D. ÁLVAREZ VALENZUELA, M. PAIVA VÉLIZ, “Reformas a la ley chilena de propiedad intelectual: el desafío de una regulación equilibrada”, ICTSD 11, n° 2, julio (2010), artículo en línea; disponible en: <http://ictsd.org/i/news/puentes/80829/> (última consulta 1 de junio, 2013)

<sup>812</sup> Ibidem.

*compatibilidad operativa entre programas computacionales o para fines de investigación y desarrollo. La información así obtenida no podrá utilizarse para producir o comercializar un programa computacional similar que atente contra la presente ley o para cualquier otro acto que infrinja los derechos de autor.*

Esta norma limita la aplicación de la ingeniería inversa en general, incluyendo por tanto la descompilación a:

- Las copias que se obtengan de forma legítima. Por lo tanto, habrá que ser usuario legítimo de la obra, quedando las copias ilícitas fuera del ámbito de aplicación de la excepción.
- Lograr la interoperabilidad entre programas (“compatibilidad operativa”). La información interoperable no podrá utilizarse para clonar o copiar un programa (producir o comercializar un programa computacional similar) pero sí para crear un programa nuevo que compita con el anterior.
- Fines de investigación y desarrollo. Como se indicó en el capítulo primero, la ingeniería inversa no tiene como fin inmediato copiar un programa. Sus aplicaciones son de mucha más ayuda en el campo de la investigación, de la seguridad del software y de la educación. De esta forma, también se fomenta no sólo la competitividad sino también la innovación en el sector.
- Con la referencia, a que la información interoperable no puede utilizarse para crear “un programa similar que atente contra la presente ley o para cualquier otro acto que infrinja los derechos de autor”, recuerda a la norma contenida en el artículo 6 de la Directiva, aunque sin la precisión de que el programa sea “sustancialmente similar en su expresión”.

Aunque por el momento no ha habido pronunciamientos judiciales sobre esta excepción, a priori, parece lo suficiente flexible como para no otorgar una sobreprotección al titular del programa y proporcionar seguridad jurídica suficiente al titular del programa y los demás competidores.

#### 4.3. El fomento de los estándares abiertos.

Si Internet, como se ha indicado a lo largo de este trabajo y como ha declarado la Comisión Europea<sup>813</sup>, es el mayor y mejor ejemplo de interoperabilidad, el camino que debe seguir las políticas legislativas sobre el derecho de autor sobre el programa de ordenador deberían tomar más ejemplo de cómo se ha conseguido este objetivo: los estándares abiertos.

Es cierto que el ejemplo de Internet ha tenido considerable influencia también en la aparición del software libre, aunque no se trata aquí de decidir si el software libre es más beneficioso que el software privativo. Ambos sistemas tienen espacio para la convivencia y obedecen a modelos de negocio diferentes, y el legislador debe dejar espacio a la pluralidad porque con ello se favorece la libre competencia. De lo que se trata es de ofrecer alternativas de adecuación a un sistema de protección que ha demostrado estar en desequilibrio y que no sólo no favorece la interoperabilidad sino que favorece la aparición de situaciones anticompetitivas.

Políticas de estandarización ya se han llevado a cabo en otros sectores como el de Telecomunicaciones, y permiten refutar el argumento de que la estandarización de formatos fomenta el parasitismo competitivo y perjudica la innovación<sup>814</sup>.

Si el derecho de autor sobre el programa de ordenador flexibiliza la excepción de la descompilación en la línea sugerida, el acceso a la información interoperable contenida en las interfaces, a las ideas y principios en que se basa un programa se situaría en un escenario de mayor seguridad jurídica para los operadores del mercado.

El temor de los titulares de permitir el acceso a la “fórmula” del programa si se flexibilizase la excepción de la descompilación podría mitigarse con la implementación de las políticas de estándares abiertos en los que ya se está trabajando la Unión Europea.

La aprobación Reglamento (UE) nº 1025/2012 del Parlamento Europeo y del Consejo, de 25 de octubre de 2012<sup>815</sup>, en el cual por primera vez se incluye una

---

<sup>813</sup> “The internet is a great example of interoperability – numerous devices and applications working together anywhere in the world. Europe must ensure that new IT devices, applications, data repositories and services interact seamlessly anywhere – just like the internet. The Digital Agenda identifies improved standard-setting procedures and increased interoperability as the keys to success.” Disponible en: <https://ec.europa.eu/digital-agenda/en/our-goals/pillar-ii-interoperability-standards> (última consulta 1 de junio, 2013)

<sup>814</sup> Sobre las políticas de estandarización en el sector de las Telecomunicaciones, vid. A. ROOIJEN, *The software interface...*, *op.cit.*, pág. 170.

<sup>815</sup> Reglamento (UE) nº 1025/2012 del Parlamento Europeo y del Consejo, de 25 de octubre de 2012, sobre la normalización europea, por el que se modifican las Directivas 89/686/CEE y 93/15/CEE del

definición de “estándares” en el área de las TICs demuestra que el legislador comunitario va por este camino. Como ha declarado la Vicepresidenta de la Comisión Europea, responsable de la Agenda Digital, “*in all sectors, standards and standardisation drive competitiveness, promote innovation, and benefit consumers through competition. Standards are indispensable for openness, freedom and choice. Likewise in the ICT sector, having the right standard-setting procedures and interoperability rules creates the level playing field needed for all parts of the machine to fit together: devices, applications, data repositories, services and networks.*”<sup>816</sup>”

Con interfaces estandarizadas y abiertas, los precios de los distintos elementos de un sistema informático no estarían tan condicionados por el poder de mercado derivado de los *network effects* sobre una plataforma informática determinada<sup>817</sup>. Favorecerían también la especialización en la creación de un componente particular del sistema, diferenciando su producto, añadiendo valor al mismo, permitiéndole a su autor ofrecerlo como una parte intercambiable para múltiples sistemas<sup>818</sup>. Se reduciría el tiempo, tamaño y coste de los proyectos de desarrollo de software, con lo cual se abriría el mercado a competidores más pequeños y se favorecería la colaboración<sup>819</sup>.

Para los consumidores, la estandarización de las interfaces y el uso de estándares abiertos conllevaría una mayor capacidad de opción en el mercado, evitando las situaciones de *lock-in* de un único proveedor, y la confianza en que podrá seguir usando

---

Consejo y las Directivas 94/9/CE, 94/25/CE, 95/16/CE, 97/23/CE, 98/34/CE, 2004/22/CE, 2007/23/CE, 2009/23/CE y 2009/105/CE del Parlamento Europeo y del Consejo y por el que se deroga la Decisión 87/95/CEE del Consejo y la Decisión nº 1673/2006/CE del Parlamento Europeo y del Consejo, DO nº 316 de 14 de noviembre, 2012, L-2012-82195, Anexo II, pág. 89 y ss.

<sup>816</sup> N., KROES, “Opening up Europe: from Common Standards to Open Data”, speech/11/596 en Open Forum Summit 2011, Brussels, 22 de septiembre, 2011; disponible en: [http://europa.eu/rapid/press-release\\_SPEECH-11-596\\_en.htm?locale=en](http://europa.eu/rapid/press-release_SPEECH-11-596_en.htm?locale=en) (última consulta 1 de junio, 2013)

<sup>817</sup> S., SCOTCHMER, *Innovation and Incentives*, Cambridge, MA, MIT Press, 2004, pág. 303; C. MAIR, “Openness, Intellectual Property and Standardization in the European Sector”, *IP Theory*, Vol. 2(2), 3, pág. 1 (6); disponible en: <http://www.repository.law.indiana.edu/ipt/vol2/iss2/3> (última consulta 1 de junio, 2013); ECIS, “Interoperability, Competition and open standards: The Keys to a Software Strategy for European citizens and Consumers”, Comunicado de 8 de junio de 2011, disponible en: [www.ecis.eu](http://www.ecis.eu) (última consulta 1 de junio, 2013)

<sup>818</sup> J., FARRELL, “Standardization and Intellectual ...”, *op.cit.*, pág. 36.

<sup>819</sup> S., WESTON, M., KRETSCHMER, “Opens Standards in Government IT: A Review of the Evidence”, Centre for Intellectual Property & Management (CIPPM), Bournemouth University, (final draft 10 September 2012), disponible en: [www.cippm.org.uk/publications.html](http://www.cippm.org.uk/publications.html) (última consulta 1 de junio, 2013), pág. 47.

su producto en el futuro con independencia de que algunos de sus proveedores actuales desaparezcan<sup>820</sup>.

Pero para que este camino sea eficaz, las acciones deben dirigirse también a la adecuación del actual sistema de derecho de autor sobre los programas de ordenador que establece la Directiva sobre la protección jurídica de los programas de ordenador.

La estandarización de formatos y la ingeniería inversa limitada a lograr la interoperabilidad con un programa de creación independiente en circunstancias más flexibles, como las propuestas, no estandariza el software en sí mismo, porque cada programador mantendrá la posibilidad de mejorar su programa en todos los otros aspectos comerciales que no tienen que ver con los formatos.

## **5. Conclusión de capítulo.**

La relación entre la Directiva de protección jurídica de los programas de ordenador con las dos ramas del ordenamiento jurídico a las que la Directiva hace mención, el Derecho de defensa de la competencia y el Derecho contractual, existe, tiene repercusiones importantes para la garantía de la interoperabilidad y refuerza la necesidad de revisión del artículo 6 sobre descompilación.

En relación al Derecho de defensa de la competencia, la mención del Considerando Décimo Séptimo no pasa de ser una mera declaración de intenciones; a la luz del ordenamiento comunitario, no puede decirse que exista una relación de coordinación entre las normas contenidas en la Directiva para garantizar la interoperabilidad con las normas de Derecho de defensa de la competencia que se ha empleado con esa finalidad, el artículo 102 TFUE a través de la doctrina de las “essential facilities”

Lo que pone claramente evidencia la relación entre la Directiva y el Derecho de defensa de la competencia es el cada vez mayor papel tutelar que la segunda rama del ordenamiento se ve obligada a prestar ante situaciones de abuso del derecho de autor sobre el programa, lo cual también releva la existencia de un desequilibrio interno en el régimen del derecho de autor configurado por la Directiva.

---

<sup>820</sup> *Ibidem*, pág. 48; J., PALFREY, U., GASSER, *Interop: The Promise...*, *op.cit.*, capítulo 9, pág. 157 y ss.

Si bien, no debe olvidarse que en las *Orientaciones sobre las prioridades de control de la Comisión en su aplicación del artículo 82 del Tratado CE a la conducta excluyente abusiva de las empresas dominantes*<sup>821</sup>, la Comisión recoge expresamente entre las prácticas abusivas la denegación de suministrar, la información de las interfaces, la solución al debate de la interoperabilidad no debiera buscarse únicamente a través de una u otra rama del ordenamiento jurídico por separado.

Si garantizar la interoperabilidad responde a un objetivo de competitividad en el mercado, las herramientas para una solución de equilibrio deben buscarse primero en la Directiva sobre protección jurídica de programas de ordenador, a fin de no poner en riesgo la sensible interrelación entre Derecho de autor y Derecho de defensa de la competencia que la norma comunitaria establece. Una forma de mantenerlo sería mantener el recurso a la licencias obligatorias consecuencia de la aplicación de un test de circunstancias excepcionales, como una herramienta excepcional, que podría actuar en situaciones de abuso de dominio donde no sólo el acceso sino sobretodo el uso de la información interoperable se viera comprometido y siempre que la situación cumpla las tres circunstancias cumulativas que han de valorarse: que la denegación se refiera a un producto o servicio objetivamente necesario para poder competir con eficacia en un mercado descendente; que sea probable que la denegación dé lugar a la eliminación de la competencia efectiva en el mercado descendente, y que sea probable que la denegación redunde en perjuicio de los consumidores<sup>822</sup>.

En los casos donde solo el acceso se viera comprometido y no exista posición de dominio, la garantía de la interoperabilidad debiera poder gestionarse de forma efectiva a través de la excepción, revisada, del artículo 6 de la Directiva.

En relación al derecho contractual, la presunta imperatividad del artículo 8, de la Directiva no puede afirmarse con rotundidad que sea más que una declaración de intenciones. El segundo párrafo del artículo 8 no garantiza ni el flujo de comunicación sobre la información interoperable per se, ni la divulgación parcial o total de la cantidad de información que quiera divulgar el autor del programa, la cual queda totalmente a su

---

<sup>821</sup> Comunicación de la Comisión — Orientaciones sobre las prioridades de control de la Comisión en su aplicación del artículo 82 del Tratado CE a la conducta excluyente abusiva de las empresas dominantes (Texto pertinente a efectos del EEE) DO n° C 045 de 24/02/2009 pág. 7.

<sup>822</sup> *Ibidem*, ap. 81.



discreción en la licencia contractual. El problema se sitúa una vez más en el régimen de protección del derecho de autor sobre el programa de ordenador.

En lo que respecta a la revisión del artículo 6, esta debiera no sólo aclarar los requisitos de la descompilación para lograr la interoperabilidad, como hace la segunda propuesta examinada, sino que también debiera incluir la permisibilidad de la descompilación para fines de investigación y desarrollo.

La idea de que la descompilación fomenta la copia ilícita y el parasitismo de la competencia debiera ser desterrada. No olvidemos que la ingeniería inversa se ha venido utilizando como una herramienta normal para fomentar la innovación en la práctica totalidad de los sectores industriales, salvo en la informática.

La propuesta de un nuevo artículo 6 debe también otorgar suficiente margen a los tribunales que hayan de juzgar la utilización de esta excepción para valorar las circunstancias concretas del caso, sobre todo en una industria como la del software, que se caracteriza por la presencia de “network effects”. Por ello, consideramos que la modificación propuesta a efectos de autorizar la descompilación para acceder a los elementos no protegidos no debiera ser tan precisa como la que autorice la interoperabilidad con un programa creado de forma independiente, fomentando de este modo la aplicación de la Regla de los Tres Pasos en casos de descompilación.

En último lugar, esta flexibilización del artículo 6 también ayudaría a la promoción de la creación de estándares abiertos en el sector del software, facilitando la interoperabilidad. De este modo, se favorecería también la competitividad en el sector, fomentando la innovación, ampliando la oferta de productos a los consumidores y disminuyendo los precios. Los consumidores podrían intercambiar o sustituir una parte de un programa de un proveedor con otra de otro proveedor, rebajando las situaciones de “lock-in”, tan habituales en el sector informático.



## **Conclusiones finales.**



## ***CONCLUSIONES.***

---

A la luz de lo que queda expuesto en el presente trabajo, se pueden formular, como más relevantes, las siguientes conclusiones:

- I.**La interoperabilidad como concepto multidimensional, juega un papel fundamental en el sector industrial del software. Por esta razón ha sido declarada materia de “interés público” a nivel comunitario en la Agenda Digital de la Comisión Europea. Esto supone que la interoperabilidad constituye un estándar prioritario y trascendental a efectos de garantizar la competencia y la innovación en el sector de las tecnologías de la información.
- II.**La relación entre ley e interoperabilidad es mucho más compleja de lo que puede parecer, porque es bidireccional y frágil, de manera que la forma en que se coordinan las distintas ramas del ordenamiento afectadas es la clave. En este trabajo se ha demostrado que el papel del Derecho de autor en relación a la promoción de la interoperabilidad es mayor del que a simple vista pueda parecer. Si bien es cierto que su rol es más limitado que otros derechos de exclusiva para las innovaciones en el sector de las tecnologías de la información, en lo que respecta a la interoperabilidad, es bastante significativo.
- III.**La adopción internacional del Derecho de Autor como sistema de protección de los programas de ordenador, asimilándolas a obras literarias, es una opción que sigue suscitando enormes dificultades tanto a los tribunales como a la doctrina. Por ello, es importante que al analizar el programa de ordenador como obra protegida por derecho de autor no se pierda de vista que los sistemas de protección son normalmente la expresión de las industrias y las culturas para las que son articulados más que una herramienta neutral que regula una situación jurídica.
- IV.**Una consecuencia de proteger los programas de ordenador como obras literarias es que, con carácter general, la información interoperable, situada en las especificaciones de sus interfaces, no es divulgada ni publicada. En esta situación, la

protección de derecho de autor sobre el programa choca con el interés general de garantizar la interoperabilidad. Esta fricción se ha materializado en dos elementos del derecho de autor: el objeto de protección y la facultad exclusiva de reproducción.

V. La cuestión que se plantea respecto del objeto de protección, es si las interfaces forman parte de este objeto o no. Las respuestas dadas por jurisprudencia y doctrina tanto en Estados Unidos como en la Unión Europea, aunque no son unánimes, comparten la utilización de dos criterios tradicionales del derecho de autor: el principio de la dicotomía idea-expresión y el criterio de la originalidad.

El examen de la normativa y jurisprudencia de los Estados Unidos pone de manifiesto que todavía no hay una solución para este debate que satisfaga a todas las partes implicadas.

El caso norteamericano *Google v. Oracle*, ha declarado por fin que las interfaces de aplicación (APIs) son métodos de operación, y por tanto, no pueden ser protegidas por el derecho de autor porque ideas y funciones no pueden ser monopolizadas a través del derecho de autor.

La importancia de esta resolución para el sector de las tecnologías de la información y de la comunicación, no debiera pasar desapercibida para el legislador, ni para el Tribunal Supremo, si en un futuro cercano se le presenta la oportunidad de pronunciarse sobre la cuestión. Conviene no olvidar lo dispuesto en el preámbulo del Tratado WCT sobre la obligación que las Partes Contratantes tienen de “mantener un equilibrio entre los derechos de los autores y los intereses del público en general, en particular en la educación, la investigación y el acceso a la información”, y que también refleja el Convenio de Berna.

En la Unión Europea, el Tribunal de Justicia, después de veinte años de silencio ha tenido ocasión de pronunciarse recientemente tanto sobre la consideración de idea o expresión de las interfaces, como de la interpretación del criterio de originalidad que ha de aplicarse a un programa de ordenador.

Sobre la aplicación del principio idea-expresión, cabe concluir que:

–A juicio del TJUE en bienes de naturaleza utilitaria como los programas de ordenador, para diferenciar entre idea y expresión del programa ha de recurrirse al test de la reproducción no meramente funcional. Es decir, cuando la reproducción de elementos del programa no permiten acceder a los códigos, sino que la reproducción obedece a un objetivo meramente funcional en sentido técnico, y que no sirve para explotar alguna de las funcionalidades concretas del programa, la información que se reproduce no constituirá una forma de expresión del programa sino una idea, y por tanto, no protegida por el derecho de autor.

–En segundo lugar, los trabajos preparatorios de creación de un programa de ordenador se considerarán una forma de expresión del mismo, y por tanto, objeto de la protección de la Directiva, siempre que los mismos puedan conducir al desarrollo ulterior de un programa o bien conlleven la reproducción del programa en términos que no sean técnicos ni sirvan para explotar alguna de las funciones del propio programa.

Por consiguiente, parece posible afirmar que la Directiva 2009/24/CE sobre protección jurídica de programas de ordenador, no considera a las interfaces como una forma de expresión.

En relación al criterio de originalidad, la reciente jurisprudencia del Tribunal de Justicia obliga a discernir entre el tipo de interfaz ante el que nos encontremos:

–Las interfaces que obedecen a requisitos esenciales de funcionamiento, comportamiento y comunicación entre dos o más elementos del programa, que son las relevantes a efectos de alcanzar la interoperabilidad, no son obras originales y por lo tanto, no están protegidas por derecho de autor. Hay que matizar que si bien esta clase de interfaces pueden concebirse en diferentes formatos, y la utilización de unos u otros dependerá de la elección personal del programador, sin embargo, por su propia naturaleza responden a

consideraciones técnicas, reglas y especificaciones unificadas y estándares de comunicación del programa, no a la libertad creativa del programador.

–Las interfaces de usuario, sin embargo, responden a razones de diseño, y aunque el asunto C-393/09 declara que la interfaz de usuario no constituye una forma de expresión de un programa de ordenador en el sentido del artículo 1.2 de la Directiva, esa interfaz puede protegerse, como obra independiente si constituye “una creación intelectual propia de su autor” según el criterio establecido en el asunto C-604/10.

Este nuevo criterio de originalidad, aplicado a las interfaces, es además acorde con el principio de proporcionalidad que debe regir la Directiva de Software, de manera que el desarrollo de programas funcionalmente parecidos o idénticos a otros debe ser expresamente posible para nuevos competidores, evitando la aparición de monopolios anticompetitivos.

**VI.** El otro punto de fricción entre derecho de autor y garantía de la interoperabilidad, se ha materializado en una excesiva e innecesaria restricción a la hora de permitir el uso de técnicas de ingeniería inversa como la descompilación, que en situaciones donde la información interoperable no es accesible, se convierten en una herramienta fundamental para lograr la interoperabilidad.

Las razones de esta resistencia a la descompilación han sido dos: Una de tipo jurídico, porque estas técnicas afectan a las facultades exclusivas de reproducción y transformación. Otra de tipo político, basada en la creencia de que la descompilación fomenta la copia y el parasitismo de la competencia.

El examen de la normativa y jurisprudencia de los Estados Unidos, como primer país que tuvo que tratar con casos sobre descompilación por razones de interoperabilidad, pone de manifiesto que el debate sigue abierto.

La aplicación de la doctrina del fair use desde el caso Segal, llevó a la práctica totalidad de los tribunales a considerar que, la copia del código con un



proceso de ingeniería inversa destinado a extraer la información de las interfaces para lograr la interoperabilidad es un acto que no infringe el copyright.

Sin embargo, desde la entrada en vigor de la DMCA la dificultad de gestionar políticas equilibradas sobre la interoperabilidad se ha vuelto a poner de manifiesto. Este hecho, unido a que desde el caso Segra el incremento de solicitudes de patentes para las interfaces de software ha sido llamativo, reflejan que la cuestión de la interoperabilidad solo encontrará una solución de equidad cuando los legisladores rechacen las políticas legislativas que permiten la concesión de derechos de propiedad intelectual generadores de grandes desequilibrios y se comprometan con una política de innovación abierta y multifacética.

En el Derecho comunitario, la descompilación se reguló a través de una excepción legal contenida en el artículo 6 de la Directiva 91/250/CEE de 14 de mayo. Si bien finalidad de esta excepción era garantizar la consecución de la interoperabilidad, y de esa manera mejorar la competitividad del sector de las tecnologías de la información, la primera conclusión que deriva de su análisis es que no cumple con ninguno de tales objetivos, sino que los obstaculiza.

En primer lugar, las condiciones que se imponen a la persona autorizada a descompilar para acceder y hacer uso de la información interoperable son tan estrictas que constituyen más un obstáculo a efectos de la elaboración de programas independientes interoperables con apoyo en la descompilación que un herramienta útil.

En segundo lugar, no existen criterios interpretativos del artículo proporcionados por el Tribunal de Justicia, posiblemente porque el recurso a la norma, aunque sea como última opción, es tan complicado para el creador del programa independiente, que se prefiere el recurso a otras opciones más “seguras”.

En tercer lugar, la escasa jurisprudencia de los tribunales de los Estados miembros objeto de estudio sobre esta norma, sólo permite establecer la siguiente conclusión con carácter general: La descompilación sobrepasa lo autorizado por la excepción cuando se realice para invalidar una medida tecnológica de protección

(MTP) con el único fin de alcanzar interoperabilidad. Por lo tanto, el requisito de la creación de un programa independiente es imprescindible, y el programa creado a través del recurso de la descompilación debe ser funcionalmente independiente.

Los datos anteriores sugieren que la excepción necesita ser revisada y modificada, de tal modo que el resultado proporcione crear un escenario jurídico que anime a los competidores a divulgar la información interoperable, de un modo similar al propuesto: la descompilación debiera permitirse no sólo para lograr la interoperabilidad con un programa creado de forma independiente, sino para acceder a los elementos no protegidos del programa para fines de interés público, siempre que no se atente de forma injustificada contra los intereses del autor.

**VII.**Respecto de la relación de la Directiva sobre protección jurídica de programas de ordenador con el Derecho de defensa de la competencia, el trabajo ha demostrado que en materia de interoperabilidad, la ineficacia de la excepción contenida en el artículo 6 de la Directiva, ha provocado la necesaria intervención del Derecho de defensa de la Competencia en aquellas situaciones donde un competidor que ostenta una posición de dominio se niega a proporcionar la información interoperable de su programa. Casos como Microsoft, Apple o The Mathworks son ejemplo de ello, y refuerzan la idea de que la interoperabilidad es una cuestión de primer orden en el mercado comunitario.

Sin embargo, los procedimientos sancionadores en derecho de la competencia suelen ser largos en el tiempo. En mercados muy dinámicos como el informático, 2 o 3 años pueden constituir una eternidad. ¿Cuántas nuevas tecnologías informáticas habrán muerto o ni siquiera habrán tenido una oportunidad en ese intervalo de tiempo por no disponer de unas condiciones de entrada justas al mercado?

Por lo tanto, el recurso a la licencias obligatorias, derivadas de la aplicación de la doctrina de las “essential facilities”, debiera ser una herramienta reservada para corregir situaciones excepcionales donde, ante comportamientos abusivos por negarse a negociar, se corrijan los desequilibrios sobre el acceso y uso de la información interoperable que no puedan ser anticipados por el derecho de autor.

Dar mayor protagonismo al Derecho de defensa de la competencia podría poner en riesgo el objetivo de protección de la Directiva sobre protección jurídica de programas de ordenador, perjudicando la innovación. Sin embargo, mientras el sistema articulado por la Directiva para acceder y utilizar la información interoperable no sea revisado, este argumento seguirá siendo débil porque podría ocasionar que las deficiencias estructurales del derecho de autor sobre el programa fuesen trasladadas al derecho de defensa de la competencia.

**VIII.**Respecto de la relación entre la Directiva sobre protección jurídica de programas de ordenador y el derecho contractual, de la presunta imperatividad del artículo 8 puede afirmarse que no pasa de ser una declaración de intenciones. El segundo párrafo del artículo 8 no garantiza ni el flujo de comunicación sobre la información interoperable per se, ni la divulgación parcial o total de la cantidad de información que quiera divulgar el autor del programa, la cual queda totalmente a su discreción en la licencia contractual. El problema se sitúa una vez más en el régimen de protección del derecho de autor sobre el programa de ordenador.

**IX.**La revisión del artículo 6 debiera no sólo centrarse en aclarar los requisitos de la descompilación para lograr la interoperabilidad con un programa creado de forma independiente, sino que también debe incluir la permisibilidad de la descompilación para fines de investigación y desarrollo.

La idea de que la descompilación fomenta la copia ilícita y el parasitismo de la competencia debiera ser desterrada. No olvidemos que la ingeniería inversa se ha venido utilizando como una herramienta corriente para fines de investigación y desarrollo en la práctica totalidad de los sectores industriales, salvo en la informática.

La propuesta de un nuevo artículo 6 debe también otorgar suficiente margen a los tribunales que hayan de juzgar la utilización de esta excepción para valorar las circunstancias concretas del caso, sobre todo en una industria como la del software, que se caracteriza por la presencia de “network effects”. Por ello, consideramos que

la modificación propuesta a efectos de autorizar la descompilación para acceder a los elementos no protegidos debe ser menos detallada que la que autorice la interoperabilidad con un programa creado de forma independiente, fomentando de este modo la aplicación de la Regla de los Tres Pasos en casos de descompilación.

En último lugar, esta flexibilización propuesta del artículo 6 también ayudaría a la promoción de la creación de estándares abiertos en el sector del software, facilitando la interoperabilidad. De este modo, se favorecería la competitividad y el dinamismo del sector, ampliando la oferta de productos a los consumidores y disminuyendo los precios. Los consumidores podrían intercambiar o sustituir una parte de un programa de un proveedor con otra de otro proveedor, rebajando las situaciones de “lock-in”, tan habituales en el sector informático.

**Anexo Mención Europea: Introduction and Conclusions.**



## ***INTRODUCTION***

---

### **1. Computer programs as a key factor of the information society**

Over the last thirty years, industrialized countries have become information societies. Technical progress, discovery and development of new media and communication technologies have demanded effort and fast adaptation to the new environment in Law.

The evolution of traditional media (telegraph, telephone, phonograph, radio and television) has lead us to a society where it is said "software is everywhere"<sup>823</sup>, due to improvements in microelectronics, the increasingly sophisticated software development and the proliferation of the Internet

This may sound cliché, but the truth is that software is one of the engines of our knowledge society and goes far beyond our personal computer. Like, for instance, domotic applications, which are used to improve the comfort of homes or to improve safety aspects; the so-called smartphones, laptops and tablets, in-car computer equipment like parking sensors, braking systems, engine control, etc. The list is endless. The truth is that, as consumers, we rely on this software without understanding or knowing it in any great depth.

It could be said that nowadays software is the alma mater of computers. It is the key factor that differentiates modern products and services. It can be found in all sorts of systems: transport, medical, telecommunications, military, industrial processes, entertainment, office, etc. In fact, software is almost unavoidable in a modern world.

Copyright Industries have become a significant self-production sector in which software represents its epicenter. Legal protection of this good still creates significant challenges for copyright, which put its ability to change and evolve to the test.

The emergence of computers and, with them, computer programs constituted what some authors have called the second industrial revolution.<sup>824</sup> Indeed, it has not yet

---

<sup>823</sup> More and more authors use this expression, among them: PRESSMAN, R. (2005), *Ingeniería del Software. Un enfoque práctico*, adapted by D., INCE, 5th ed., Mc Graw Hill, España, p. 4; NORTON, P. (2006) *Introducción a la computación*, 6th ed. Mc Graw Hill, México, p. 3.

<sup>824</sup> Expression used by KÖHLER, R. (1968), *Der urheberrechtliche Schutz der Rechenprogramme*, München, Beck, p. 1.

been sixty years since the idea of selling a computer program separately from the computer arose, a pioneering initiative from IBM<sup>825</sup>. Ever since then the software industry has steadily grown to become one of the prominent engines of the world economy<sup>826</sup>.

Two reasons justify the above statement:

First of all, the information technology and communications sector (ICTs), in which software is a part, is a constantly shifting sector. A computer program that was the latest hit yesterday may be obsolete today by the appearance of a new one. Updating is the keyword, and it is happening at such a dizzying pace that the supply of computer programs onto the market is constantly being renewed. Companies follow the time-to-market maximum<sup>827</sup>, that is to say, to be the first to be on the store shelf. For this reason, time factor has become a key element and companies run the risk of the product becoming obsolete even before recovering the initial investment.

Secondly, we live in an increasingly interconnected society, in which the way to interact between people, companies, and governments has changed significantly: companies use the Internet to interact with customers and suppliers, they use knowledge management tools to be more efficient, governments improve their presence on the Internet and provide all sort of online services for citizens and consumers are

---

<sup>825</sup> For a full historical view of the birth of the software industry Vid. BAND, J., KATOH, M. (1995) *Interfaces on trial: Intellectual Property and Interoperability in the Global Software Industry*, Westview Press, Boulder, p. 2; STOBBS, G.A. (2000) *Software patents*, Aspen Law & Business, Gaithersburg, p. 38; MERGES, R. (2009) *Intellectual Property in the new technology age*, Wolters Kluwer Law & Business, New York, Aspen, p. 603.

<sup>826</sup> From a strictly economic perspective, Gartner Inc, a leading company in consulting and research on information technology, said in the data presented in early 2010, that despite the difficult global financial situation, total estimated expenditure for technological innovation this year would amount to 3.4 trillion, 4.6 percent more than 2009. (For more information see [www.gartner.com](http://www.gartner.com)).

At institutional level, the European Commission said: "software development has shown steady growth in recent years. It has significantly influenced the entire European industry and contributes substantially to GDP and employment. In 1998, the market value of computer programs in Europe amounted to 39,000 million euros". (Proposal for a European Parliament Directive and the Council on the patentability of computer-implemented inventions, COM (2002) 92 final, p. 2). More recently: "Computer programs are playing an increasingly critical role in a wide range of sectors and, consequently, computer technology can be considered to be of paramount importance for the Community's industrial development". (Recital 2, Directive 2009/24/EC of the European Parliament and of the Council of 23 April 2009 on the legal protection of computer programs (Codified version), DO 2009/05/05, L111, p.16).

<sup>827</sup> KENNETH B., K. (ed.) (2005), *PDMA Handbook of New Product Development*, 2nd ed., John Wiley & Sons, p. 173-187; available online: [http://www.knovel.com/web/portal/browse/display? EXT\\_KNOVEL\\_DISPLAY\\_bookid=1416&Vertical\\_ID=0](http://www.knovel.com/web/portal/browse/display? EXT_KNOVEL_DISPLAY_bookid=1416&Vertical_ID=0) (last seen on 01, June, 2013).



incessantly increasing the use of computer tools in their professional and personal relationships.

In a society with such characteristics, whose fundamental axis is the access to information, the computer program is the main intermediary between information and humans. However, the goal of such a society should be for us all to be able to communicate with everyone without interference.

The Internet is the best example of this: no matter if we are users of a Windows, Linux or Mac IOS operating system, we all can access the Internet without barriers, provided that such communications are handled through the use of open standards such as TCP/IP, HTML, POP or SMTP.

Nonetheless, something we have crystal clear in communications over the Internet seems to be less clear when it comes to the way we store and distribute data, or even in the way we establish services on the Internet. Most of the users have been guided by the hand of technology creators who have succeeded in the market, all of them based on proprietary standards, which also means proprietary technologies.

Still, notwithstanding the technology we have decided to use, highly respectable though it may be, consumers should be aware that their technological choice could play tricks on them if certain precautions are not taken.

## **2. The legal debate on interoperability in the frame of European Copyright Law.**

The European institutions began to realize the importance of interoperability at the turn of this century and two major initiatives have been launched since then.

The first one started in June 2002, where the European Heads of State endorsed the eEurope 2005 Action Plan<sup>828</sup>. This document was a call made to the European Commission for it to publish an agreed interoperability framework to support the delivery of pan-European e-government services to citizens and enterprises.

The most recent, the Digital Agenda for Europe 2010-2020, launched by the European Commission, includes among its pillars, interoperability and standards. The

---

<sup>828</sup> The eEurope 2005 action plan, COM(2002)263.

Agenda itself recognizes interoperability as a priority issue that crosscuts a wide spectrum of IT laws and policies<sup>829</sup>.

Strikingly none of these initiatives address any measure on computer program copyright.

When the international community decided to protect computer programs with this exclusive property right, the guarantee of interoperability was also affected by it.

This study has not overlooked the fact that there are other instruments of protection for computer programs as well as for software interoperable information: patent law, trade secret, contract law or even criminal law. Nor can it be said that copyright plays an essential role, albeit limited, in the protection of computer innovations. However, following the statement in the preamble of the WCT Treaty<sup>830</sup>, this work argues that rules which break the balance between creators and new competitors or that jeopardize public interest should be modified or withdrawn, not only to respect the proportionality principle but also to prevent other areas of the legal system from being modified, or even threatening their own fundamental principles and thus putting the legal certainty of the whole system at risk.

Interoperability in EC Law is guaranteed by means of a legal exception to copyright. This exception is contained in Article 6 of the Software Directive, under the title “decompilation”. We consider this provision breaks the balance between copyright and public interest and it does not assure interoperability, so this has significant implications for innovation in the computer industry. Therein lays, according to us, the epicenter of the interoperability debate on copyright.

A very surprising fact about this provision is that, since its adoption over twenty years ago, questions have neither been raised about its interpretation before the European Court of Justice, nor has it been the subject of more than a few court decisions. These facts have led us to consider the following questions:

---

<sup>829</sup> <http://ec.europa.eu/digital-agenda/en/our-targets/pillar-ii-interoperability-standards>

<sup>830</sup> Preamble to the WIPO Copyright Treaty (from now on WCT) adopted in Geneva on December 20, 1996, BOE Num. 148, of 18 June 2010, p. 9638: “The Contracting Parties, (...) recognizing the need to maintain a balance between the rights of authors and general public interest, particularly in education, research and access to information, as reflected in the Berne Convention”.

If the text of the rule is so clear that it provides enough legal certainty to market operators as to avoid the use of this exception, why do we keep talking about interoperability issues affected by copyright?

If, on the contrary, because of the complexity and excessive restriction of this provision, an imbalance between copyright and a public interest assurance such as interoperability has been created, its usefulness will require further discussion.

This research deals with the above-mentioned issues and proposes a solution to the interoperability debate from the point of view of copyright.

The paper is structured as follows: the first chapter gives a broad overview of theoretical concepts of the debate: interoperability, computer programs, decompilation and interfaces.

The first part of this chapter deals with matters such as: What is the interoperable information of a program? Where is this information located? How does copyright affect access and the use of this information? Furthermore, we will also deal with the principal economic effects of interoperability for ICT markets.

The second part of the chapter discusses the concepts of computer programs and decompilation, paying particular attention to how this reverse engineering process interferes with the copyright on the program. The last section of this chapter examines the interfaces, a specific part of the software, which are the key to achieve interoperability.

From here on, the research exclusively focuses on the analysis of the debate from a legal perspective. Firstly, we present the current state of the debate on interoperability in U.S. copyright. The reasons for the choice of the United States are that it was the first country to protect computer programs by copyright and their courts have dealt with more issues related to reverse engineering for interoperability reasons.

Moreover, the United States, unlike civil law tradition countries, uses an open exception system, which allowed us to see which of the two regimes is more efficient when dealing with the problems presented by reverse engineering to achieve interoperability.

The third chapter examines European law, whose core is article 6 of the Directive on the legal protection of computer programs. Three different tests have been carried out in order to answer the questions raised by this work. Firstly, we analyze the European Directive from a historical perspective in order to understand how and why European lawmakers came to the text of article 6. Secondly, the provision has been assessed, highlighting the interpretational doubts that rise from the text, in order to consider the decisions issued by the European Court of Justice on other articles of the Software Directive that affect article 6. Thirdly, four Member States have been chosen: Germany, France, the UK and Spain in order to analyze how they have implemented the exception into their national law and how their courts have applied it.

The fourth chapter deals with the relationship between the Directive on the legal protection of computer programs and the two other areas of law, which are explicitly mentioned in the Directive to ensure interoperability. Firstly, Competition Law, mentioned in the Seventeenth Recital of the rule. Secondly, Contract Law, referred to the second paragraph of article 8 of the Community directive. This analysis seeks to answer the question of whether compulsory licenses for interoperable information are the most efficient solution to this debate and to check whether the decompilation exception of imperativeness has any practical relevance.

The chapter closes with analysis of the most recent proposals *de lege ferenda* on decompilation. A concrete proposal for revision is given and it also anticipates some of the consequences that could arise from interoperability if regulation on decompilation were more flexible.

Finally, a set of conclusions summarizes the most relevant results of the whole research.

## ***CONCLUSIONS***

---

As a result of the above discussion, the following conclusions can be drawn:

- I.** Interoperability, as a multidimensional concept, plays a key role in the software industry. Indeed, for this reason it has been declared a matter of "public interest" by the Digital Agenda of the European Commission. This means that interoperability is a priority and transcendental standard that ensures competition and innovation in the field of information technologies.
- II.** The relationship between law and interoperability is much more complex than it may seem because it is bidirectional and fragile; therefore the key lies in how the different affected law systems interface with each other. In this dissertation, it has been shown that the role of copyright in relation to the advancement of interoperability is greater than it may seem at first glance. Although copyright's role is more limited than other exclusive rights for innovations in the field of information technology, such as for instance patents, in terms of interoperability, it is very significant.
- III.** International adoption of Copyright as the legal protection system for computer programs, on the same basis as literary works, is an option which still raises paramount complications for both courts and doctrine. Therefore, when analyzing a computer program as work protected by copyright, it must be borne in mind that legal protection systems are usually the expression of industries and cultures for which they have been developed more than a neutral tool that regulates legal status.
- IV.** A consequence of protecting computer programs as literary work is that interoperable information located in the interface specifications is not normally disclosed or published. In this case, copyright protection on the program comes into conflict with the public interest of fostering interoperability. This conflict has been translated into two copyright elements: the object of protection and the exclusive right of reproduction.
- V.** The question that arises in regard to the object of protection is whether the interfaces are part of this object. Answers provided by case-law and doctrine in the

United States and the European Union, although not unanimous, share the use of two traditional criteria of copyright: the principle of the idea-expression dichotomy and the originality criterion.

The review of law and case-law in the United States shows that there is still no solution to this debate that satisfies all the parties involved.

In the US case *Google vs. Oracle*, it has been finally declared that application interfaces (APIs) are methods of operation; therefore, they cannot be protected by copyright because ideas and functions cannot be monopolized by copyright.

The importance of this judgment for the information technology and communication sector should not go unnoticed by the legislator, nor by the Supreme Court, if an opportunity to issue on this matter appears again in the near future. It should not be forgotten what it is declared in the preamble of the WCT Treaty in regard to the obligation of the Contracting Parties to "keep a balance between copyrights and public interests in general", which is also reflected in the Berne Convention.

Within the scope of the European Union, after twenty years of silence, the European Court of Justice has recently had the opportunity to decide on both the consideration of interfaces as an idea or expression, and the interpretation of the criterion of originality to be applied to a computer program.

On the application of the idea/expression principle, it can be concluded:

–According to the ECJ, in goods of a utilitarian nature such as computer programs in order to discern between idea and expression of the program the “not merely functional reproduction test” must be used. This means, when the reproduction of a program does not allow access to its codes, but follows a purely functional purpose in the technical sense, and that reproduction does not serve to exploit any of the specific features of the program, the information resulting from that reproduction will not constitute a form of expression of the program but an idea and, therefore, it will not be protected by copyright.

–Secondly, the preparatory design material of a computer program is considered to be a form of expression of the work and, therefore, subject to the protection conferred by the Software Directive, provided that it can lead to further development of a new program, or that the preparatory design material can involve a non-technical reproduction and that neither serve to exploit a functional feature of the program.

Therefore, it seems possible to say that, according to the Directive 2009/24/EC on the legal protection of computer programs, interfaces should not be considered as a form of expression.

With regard to the criterion of originality, the recent case law of the European Court of Justice requires the distinction between the interfaces we might be looking at:

–Interfaces resulting from essential operational requirements, performance and communication, between two or more elements of the program, which are also pertinent for interoperability purposes, are not original work and, therefore, are not protected by copyright. It needs to be clarified that, even though these types of interfaces can be designed in different formats, the use of one or another may depend on the personal choice of the programmer. However, due to their nature, they really depend on technical considerations, unified rules and specifications or software communication standards and not on the programmer's free creative will.

–On the other hand, user interfaces respond to design reasons, and although case C-393/09 stated that user interfaces are not a form of expression of a computer program within the meaning of Article 1.2 of the European Directive, this interface can be protected as independent work if it comes from "the author's own intellectual creation " according to the criterion established by Case C-406/10.

This new criterion of originality, when applied to interfaces, is also consistent with the principle of proportionality that must guide the Software Directive, so that the development of new programs with similar functions or those

functionally identical to former programs must be expressly permitted for new competitors, thus avoiding the emergence of anti-competitive monopolies.

**VI.** Another conflicting point between copyright and the guarantee of interoperability materializes in an excessive and unnecessary restriction on the use of reverse engineering techniques like decompilation, which, in cases where interoperable information is not available or accessible, becomes an essential tool for its achievement.

There have been two reasons for this refusal: one legal, because these techniques may affect the exclusive rights of reproduction and transformation. Another policy is based on the belief that decompilation encourages copy and competition parasitism.

The study of law and case-law in the United States, as the first country that had to deal with cases on decompilation for interoperability reasons, shows that the debate remains open.

Since the Sega case, the application of the fair-use doctrine has led almost all courts to consider that copying code through a reverse engineering procedure with the aim of extracting interface information in order to achieve interoperability does not constitute copyright infringement.

However, since the DMCA came into force the difficulty of managing balanced policies on interoperability has appeared again. This, together with the fact that, since the Sega case, the number of patent applications for software interfaces has increased significantly, reveals that the issue of interoperability will only find an equitable solution when lawmakers commit themselves to an open and multifaceted innovation policy and reject traditional policies that grant intellectual property rights and generate large imbalances.

In European Community law, decompilation was articulated through an exception contained in Article 6 of Directive 91/250/EEC of 14 May 1991. Even though this provision was established with the main objective of promoting and ensuring *ex-ante* computer interoperability and thereby improving competitiveness



in information technologies, the first conclusion after its analysis is that the exception does not just meet any of these objectives, but actually hinders them.

In the first place, the conditions imposed to the person authorized to decompile and access and use the interoperable information are so strict that they have become a hindrance rather than a useful tool for the development of interoperable independent programs assisted by decompilation.

Secondly, there are not any interpretative criteria provided by the ECJ, probably because using this legal exception, even as a last resort, is too complicated for the creator of an independent program, so he or she may prefer to use “safer” alternatives.

Thirdly, the scarce jurisprudence over this provision in the Member States covered by this research only allows the following general conclusion to be drawn: decompilation goes beyond what is permitted by the legal exception when it is performed to invalidate a technological protection measure (TPM) for the only purpose of achieving interoperability. Therefore, the requirement of creating an independent program becomes indispensable, and the program created by means of decompilation must be functionally independent.

The above data suggest that the exception needs to be revisited and reformed, so that, as proposed in this study, the result provides the creation of a legal framework that encourages competitors to disseminate interoperable information: decompilation should not only be allowed to achieve interoperability with an independently created program, but to access the unprotected elements of the program for public purposes, provided it does not unreasonably affect the author's interests.

**VII.** With regard to the relationship between the Directive on the legal protection of computer programs and Competition law, the study has demonstrated that, in terms of interoperability, the exception contained in Article 6 of the European Directive is ineffective and leads to the necessary intervention of Competition law in those situations where a competitor who holds a dominant market position refuses to provide interoperable information of his program. Cases like Microsoft, Apple and

The Mathworks illustrate this and add weight to the argument that interoperability is a major issue in the Community market.

However, disciplinary proceedings in competition law usually take a long time. In very dynamic markets such as software, 2 or 3 years can be an eternity. How many new computer technologies will die or not even have had an opportunity for the lack of fair conditions for entry into the market in that time slot?

Therefore, the use of compulsory licenses, resulting from the application of the essential facilities doctrine should be kept as an exceptional tool for situations where, and given abusive behaviors consist of the refusal to negotiate, the imbalance on the access and use of interoperable information cannot be anticipated by copyright and, therefore, it cannot be differently corrected.

Giving a greater role to Competition Law could jeopardize the protection purpose of the Directive on computer program legal protection, thus undermining innovation. However, without a revision of the system articulated by the Directive in order to access and use interoperable information, this argument will remain highly weak as it may be capable of causing the structural inefficiencies of copyright on software to be transferred to competition law.

**VIII.** On the relationship between the Software Directive and Contract Law, on the matter of the alleged imperativeness of article 8, it can be affirmed that it is no more than a statement of intentions. The article's second paragraph does not guarantee communication flow on interoperable information per se, nor a partial or total disclosure of information that the program's author may want to disclose. This depends on the legal terms of the contractual license. The problem moves, once again, to the copyright regime for computer programs.

**IX.** Revision of article 6 should not only focus on clarifying the conditions of decompilation in order to achieve interoperability with an independent created program, but should include permissibility of decompilation for research and development purposes.

The idea that decompilation fosters illegal copy (piracy) and competition parasitism should also be completely banished. It should not be forgotten that

reverse engineering has been used as an ordinary tool for research and development purposes in all industrial fields but software.

The proposal for a new article 6 should also give enough maneuver room to courts that might decide on the use made of the legal exception in order to take into account the particular circumstances of the case, especially in such a sector as software, which is characterized by the presence of network effects. Because of that, this study considers the modification proposed in order to permit decompilation to access non-protected elements should be less detailed than that allowing decompilation for interoperability reasons, in this way fostering the application of the Three-Step-Test in decompilation cases.

Lastly, this proposal for the softening of article 6 will also help to encourage the creation of open standards in the software sector, thus enabling interoperability. In this way, competitiveness and dynamism in the ICT sector will be favored, and accordingly will broaden the product offered to the consumer and see the lowering of prices. Consumers could exchange or replace a part of a program from a provider with another from a different provider, reducing the lock-in situations, which are so common within the software sector.



## Índice de jurisprudencia.

### 1. Jurisprudencia Comunitaria.

#### 1.1. Decisiones de la Comisión:

- Decisión 2007/53/CE de la Comisión de 24 de marzo de 2004, relativa a un procedimiento con arreglo al artículo 82 CE y al artículo 54 del Acuerdo EEE contra Microsoft Corporation (Asunto COMP/C-3/37.792 – Microsoft) DO 2007, L32, pág. 23.

#### 1.2. Tribunal de Justicia de la Unión Europea.

- Sentencia del Tribunal de Justicia de 6 de marzo de 1974, Istituto Chemioterapico Italiano S.p.A. y Commercial Solvents Corporation v. Comisión, asuntos acumulados 6/73 y 7/73, Rec no disponible.
- Sentencia del Tribunal de Justicia de 9 de noviembre de 1983, asunto 322/81, NV Nederlandsche Banden Industrie Michelin v. Comisión de las Comunidades Europeas, Rec 1983, 03461.
- Sentencia del Tribunal de Justicia de 5 de octubre de 1988, asunto 238/87, AB Volvo v. Erik Veng (UK) Ltd., Rec 1988, 06211
- Sentencia del Tribunal de Justicia de 18 de febrero de 1992, asunto C-30/90, Comisión de las Comunidades Europeas contra Reino Unido de Gran Bretaña y de Irlanda del Norte, Rec 1992 I-829.
- Sentencia del Tribunal de Justicia de 13 de julio de 1995, asunto C-350/92, Reino de España contra Consejo de la Unión Europea, Rec 1995 I-1985
- Sentencia del Tribunal de Justicia de 6 de abril de 1995, asuntos acumulados C-241/91 P y C-242/91 P, Radio Telefís Eireann (RTE) e Independent Television Publications Ltd (ITP) v. Comisión. Rec 1995 I-00743.
- Sentencia del Tribunal de Justicia (Sala Sexta) de 26 de noviembre de 1998, asunto C-7/97, Oscar Bronner GmbH & Co. KG contra Mediaprint Zeitungs- und Zeitschriftenverlag GmbH & Co. KG, Mediaprint Zeitungsvertriebsgesellschaft mbH & Co. KG y Mediaprint Anzeigengesellschaft mbH & Co. KG. Rec I-7817.

- Sentencia del Tribunal de Justicia de 29 de abril de 2004, asunto C-418/01, IMS Health GmbH & Co. OHG v. NDC Health GmbH & Co. KG, Rec. 2004 I-05039.
- Sentencia del Tribunal de Primera Instancia (Gran Sala) de 17 de septiembre de 2007, Microsoft Corp. v. Comisión de las Comunidades Europeas, asunto T-201/04, *Rec 2007 II-03601*.
- Sentencia del Tribunal de Justicia (Sala Cuarta) de 16 de julio de 2009, asunto C-5/08, Infopaq International A/S contra Danske Dagblades Forening., Rec. p. I-6569, *DO C 220* de 12.09.2009 pág. 7
- Sentencia del Tribunal de Justicia (Sala Tercera) de 22 de diciembre de 2010, asunto C-393/09, Bezpečnostní softwarová asociace - Svaz softwarové ochrany contra Ministerstvo kultury, *DOUE C63* de 26.2.2011, pág. 8.
- Sentencia del Tribunal de Justicia (Sala Tercera) de 1 de marzo de 2012, asunto C-604/10, Football Dataco Ltd y otros v. Yahoo UK Limited y otros, *DO C118* de 21.4.2012, pág. 5.
- Sentencia del Tribunal de Justicia (Gran Sala) de 2 de mayo de 2012, asunto C-406/10, SAS Institute Inc. v. World Programming Ltd., *DO C 174* de 16.6.2012, pág. 5.
- Sentencia del Tribunal de Justicia (Gran Sala) de 3 de julio de 2012, asunto C-128/11, UsedSoft GmbH v. Oracle International Corp., *DO C 287* de 22.9.2012, pág. 10.

### **1.3.Jurisprudencia alemana.**

#### **1.3.1.Juzgados de instancia:**

- LG Kassel sentencia de 12 de mayo de 1981 (BB 1983, pág. 992), confirmada por OLG Frankfurt sentencia de 6 de noviembre de 1984 (BB 1985, pág. 139 y CR 1986, pág. 13).
- LG Mannheim sentencia de 12 de junio de 1981 (BB 1981, pág. 1543) rechazada en apelación (OLG Frankfurt sentencia de 9 de febrero de 1983 – GRUR 1983, pág. 300) y confirmada por el BGH sentencia del 9 de mayo de 1985, BGHZ 94, pág. 276 y GRUR 1985, pág. 1041.

-LG Hannover, sentencia de 28 de octubre de 1987 (18 O 58/87), CR 1988, pág. 826

-LG Mannheim, sentencia de 20 de enero de 1995, NJW 1995, pág. 3322.

### **1.3.2. Tribunales de apelación:**

-OLG Karlsruhe, sentencia de 9 de septiembre de 1986, CR 1986, pág. 723.

-OLG Karlsruhe sentencia de 10 de enero 1996 (caso Dongle-Umgehung), ZUM-RD 1997, pág. 340 (341)

-OLG Düsseldorf sentencia de 27 de marzo 1997 (caso Dongle-Umgehung), CR 1997, pág. 337.

-OLG Düsseldorf, sentencia de 16 de enero de 2001, CR 2001, pág. 371.

-OLG Hamburg sentencia de 23 de abril de 2012, BeckRS 2012, pág. 17890.

### **1.3.3. Tribunal Supremo:**

-BGH sentencia del 9 de mayo de 1985, BGHZ 94, pág. 276 y GRUR 1985, pág. 1041.

-BGH sentencia de 4 de octubre de 1990, I ZR 139/89 "Betriebssystem", GRUR 1991, pág. 449. (versión en inglés en IIC 22, (1991), pág. 723.

-BGH sentencia de 20 de enero de 1994, Caso No. I ZR 267/91 Holzhandesprogramm, GRUR 1994, pág. 363.

### **1.3.4. Tribunal Constitucional:**

-*BverfG*, 3. Kammer des Ersten Senats, *sentencia de* 25 de julio de 2005, asunto 1 BvR 2182/04 Eigentum und digitale Privatkopie, GRUR 2005, pág. 1032

## **1.4. Jurisprudencia británica.**

### **1.4.1. Tribunales de Apelación:**

- Interlego AG v Tyco Industries Inc and others [1988] RPC 343, [1989] AC217.
- Nova Productions Ltd. v. Mazooma Games Ltd and others, RPC 25, [2007].

### **1.4.2. Tribunal Supremo:**

- University of London Press Ltd v. The University Tutorial Press Ltd, [1916] 2 Ch 601.
- Gates v. Swift, RPC 13, 339, 1982, (Ch.)
- Sega Enterprises Ltd. v. Richards, FSR 73, 1983, (Ch. D.)
- Thrustcode Ltd v. W.W. Computing Ltd, FSR 1983, 502 (Ch.D.)
- Navitaire v. EasyJet Airline Co., [2004] EWHC 1725, (Ch. D.)
- SAS Institute Inc v World Programming Ltd, [2013] EWHC 69 (Ch.)

## **1.5. Jurisprudencia española.**

- AP Cádiz, Sección 5ª, sentencia núm. 30/2007 de 19 de enero, AC 2007\1018.
- AP Girona (Sección 1ª), sentencia núm. 86/2010 de 3 marzo. AC 2010\940.

## **1.6. Jurisprudencia francesa.**

### **1.6.1. Tribunales de Apelación:**

- Asunto La Commande Electronique, sentencia de 20 de octubre de 1988, JCP ed. G. 1989, II, pág. 2188.
- Asunto Nomai v. IOMEGA, sentencia de 12 de diciembre de 1997, Corte de Apelaciones de Paris (Cour d'Appel), section B, Chambre 14.
- Asunto François M., v. EMI France, sentencia de la Corte de Apelación de Versailles, Sala Primera, Sección Primera (Cour d'Appel, 1re Chambre, 1re Section), de 30 de septiembre 2004.



–Asunto Sarl AAKro Pure Tronic et a. v. Nintendo, sentencia de 26 de septiembre de 2011, Corte de Apelaciones de París (Cour d’appel), Pôle 5, Chambre 12.

### **1.6.2.Tribunal Supremo:**

–Asunto Babolat v. Pachot, sentencia de 7 de marzo de 1986 de la Corte de Casación, Asamblea Plenaria (*Cour de cassation*, AP), RIDA 129, 1986, pág. 130.

–Asunto *Atari v. Williams Electronics*, sentencia de 7 de marzo de 1986 del Tribunal Supremo, Asamblea Plenaria (*Cour de Cassation*, AP), RIDA 1986, pág. 126.

–Asunto La société Fiducial v. la société Développement professionnel spécialisé informatique (DPSI) ; et autres, sentencia de 20 de octubre de 2011 de la Corte de Casación (Cour de cassation - Première chambre civile), n° 975, (10-14.069).

–Asunto Stéphane P, UFC Que Choisir v. Universal Pictures Video France et Autres, sentencia del Tribunal Supremo, Sala de lo Civil, (Cour de Cassation, Première chambre civil) de 28 de febrero de 2006.

### **1.6.3.Consejo de la Competencia de la República francesa:**

–Décision n° 04-D-54 du 9 novembre 2004, relative à des pratiques mises en œuvre par la société Apple Computer, Inc. dans les secteurs du téléchargement de musique sur Internet et des baladeurs numériques.

## **2.Jurisprudencia norteamericana.**

### **2.1.Juzgados de instancia:**

–Synercom Technology, Inc. v. University Computing Co., 42 F. Supp. 1003 (N.D. Texas 1978).

–Hubco Data Products Corp. v. Management Assistance, Inc. (219 U.S.P.Q. (BNA) 450 (D. Idaho, 1983).

- Midway Mfg. Co. v. Strohon 564 F.Supp. 741 (D.C.Ill.,1983).
- Broderbund v. Unison World, 648 F. Supp. 1127 (N.D. Cal, 1986).
- Digital Communications v. Softklone, 659 F. Supp. 449 (N.D. Ga. 1987).
- Pearl Systems, Inc. v. Competition Electronics, Inc. 8 USPQ2d (BNA), 1988, 1520 (S.D. Fla. 1988), (WL 146047).
- Secure Services Technology, Inc. v. Time & Space Processing, 722 F. Supp. 1354 (E.D. Va. 1989)
- Lotus Dev. Corp. v. Paperback Software Intern., 740 F. Supp. 37 (D.C. D. Massachusetts 1990).
- RealNetworks, Inc. v. Streambox, Inc., Case C99-2070Z (W.D. Wa. Dec. 20, 1999).
- Tradescape.com v. Shivaram, 77 F.Supp. 2d 408 (S.D. N.Y. 1999).
- SecureInfo Corp. v. Telos Corp., 387 F. Supp. 2d 593 (E.D. Va. 2005).
- Auto Inspection Services v Flint Auto Auction, 2006 U.S. Dist. (E.D. Mich. Dec. 4, 2006) Lexis 87366.
- Healthcare Advocates, Inc. v. Harding, Earley, Follmer & Frailey, 2007 U.S. Dist. (E.D. Pa. July 20, 2007) Lexis 52544.
- Goldman v. Healthcare Management Systems, Inc., 628 F.Supp. 2d 748 (W.D. Mich. 2008).
- Apple, Inc. v. Psystar Corp., 673 F.Supp.2d 931 (N.D. Cal. 2009)
- Realnetworks, Inc. v. DVD Copy Control Ass'n, No. C 08-04548, 641 F. Supp. 2d 913; 2009 U.S. Dist. Motion granted by, Claim dismissed by Realnetworks, Inc. v. DVD Copy Control Ass'n, 2010 U.S. Dist. (N.D. Cal., Jan. 6, 2010), (LEXIS 1433).
- DocMagic, Inc. v. Ellie Mae. Inc., 745 F. Supp. 2d 1119 (N.D. Cal. 2010).
- Burroughs Payment Systems, Inc. v Symco Group, Inc. Case C-11-06268 JCS (2012 U.S. Dist.) Lexis 67198.
- Oracle America, Inc. v. Google Inc., US District Court (31 May 2012, C10-03561 WHA)

## **2.2. Tribunales de Apelación:**

- Nichols v. Universal Pictures Corp., 45 F.2d 119 (2d Cir. 1930).
- Apple Computer Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3rd Cir. 1983); cert. dismissed, 464 US, 1033 (1984).
- Whelan Associates, Inc., v. Jaslow Dental Laboratory, Inc., 797 F. 2d 1222 (3d Cir., 1986).
- Plains Cotton Cooperative Association of Lubbock, Texas v. Goodpasture Computer Service, Inc., 1 USPQ 2d. 1635 (5<sup>th</sup> Cir. 1987).
- Vault Corp. v. Quaid Software Ltd., 655 F. Supp. 750 (E.D. La. 1987), aff'd, 847 F.2d 255, 7 U.S.P.Q.2d. (BNA) 1281 (5th Cir. 1988).
- Atari Games Corp. v. Nintendo of America Inc., 975 F.2d 832 (Federal Cir., 1992).
- Aymes v. Bonelli, 47 F.3d 23 (2d. Cir. 1995), 980 F.2d 857 (2d Cir. 1992).
- Brown Bag Software v. Symantec Corp., 960 F.2d 1465 (9<sup>th</sup> Cir. 1992).
- Computer Assoc. Int'l, Inc. v. Altai, Inc., 982 F.2d 693 (2d Cir. 1992).
- Sega Enterprises Ltd. v. Accolade, Inc., 977 F.2d 1510 (9th Cir. 1992).
- Apple Computer, Inc. v. Microsoft Corp. and Hewlett-Packard Co., 35 F. 3d 1435 (9<sup>th</sup> Cir. 1994).
- MAI Sys. Corp. v. Peak Computer, Inc., 991 F.2d 511, 26 U.S.P.Q.2d (BNA) 1458 (9th Cir. 1993), cert. disp., 510 U.S. 1033 (1994).
- Bateman v. Mnemonics, 79 F.3d 1523 (11th Circuit 1996).
- Lotus Development Corp. v. Borland Intern., Inc., 49 F. 3d 807 (1st Circuit 1995), aff'd by an equally divided Court, 516 U.S. 233 (1996).
- Stenograph L.L.C. v Bossard Associates, Inc, 144 F.3d 96, 46 U.S.P.Q.2d (BNA) 1936 (D.C. Cir. 1998).
- Sony Computer Entertainment v. Connectix Corp., 203 F.3d 596 (9th Cir. 2000).
- Universal City Studios, Inc. v. Reimerdes, 111 F. Supp. 2d 294, 55 U.S.P.Q.2d (BNA) 1873 (S.D.N.Y. 2000) judgment entered, 111 F.Supp. 2d 346 (S.D.N.Y. 2000), aff'd on other grounds, 273 F.3d 429, 60 U.S.P.Q.2d (BNA) 1953 (2d Cir. 2001).
- Dun & Bradstreet Software v. Grace Consulting, 307 F.3d 197 (3rd Cir. 2002).
- Chamberlain Group, Inc. v. Skyling Technologies, 381 F.3d 1178 (Fed. Cir. 2004).

- Davidson & Associates DBA Blizzard Entertainment, Inc.; Vivendi Universal Inc. v. Jung et al., 422 F.3d 630 (8th Cir. 2005).
- Lexmark International, Inc. v. Static Control Components, Inc, 253 F. Supp. 2d 943, 948-949 (E.D. Ky. 2003), rev'd, 387 F.3d (6th Cir. 2004), reh'g denied, 2004 U.S. App. (Dec. 29, 2004), reh'g denied, 2005 U.S. App. 522 (6th Cir. Feb. 15, 2005).
- William KRAUSE dba Special-T software v. TITLESERV, Inc. et al., 402 F.3d 119 (2d Cir. 2005); (*cert. denied*), S.Ct. 622 (2005).
- Darden v. Peters*, 488 F.3d 277 (4th Cir. 2007).
- MGE UPS Systems, Inc. v GE Consumer & Industrial, Inc., 612 F.3d 760 (5<sup>th</sup> Cir. 2010), withdrawn and superseded on reh'g, 622 F. 3d 361 (5<sup>th</sup> Cir. 2010).
- Verno v. Autodesk, Inc. 621 F.3d 1102 (9th Cir. 2010).
- MDY Indus., LLC v. Blizzard Entm't, Inc., 629 F.3d 928 (9th Cir. 2011).

### **2.3.Tribunal Supremo:**

- Sony Corp. v. Universal City Studios, Inc., 464 U.S. 417 (1984).
- Harper & Row Publishers, Inc. v. Nation Enterprises, 471 U.S. 539 (1985).
- Community for Creative Non-Violence v. Reid, 490 U.S. 730 (1989).
- Feist Publications, Inc. v. Rural Telephone Service Co., 499 U.S. 340 (1991).

## **Bibliografía.**

1. ABADÍA, M., CARDENAL, J. *et al.*, *Patentes, marcas, software*, Murcia, Laborum, 2001.
2. ABBOT, J., “Reverse Engineering of Software: Copyright and Interoperability”, *J.L. & Inf. Sci.*, nº 14, 2003, pág. 7-50.
3. ADKINS, J., “Fair Use and Unpublished Works: Public Law 102-492-An Amendment to the Fair Use Statute”, *St. Louis U.L.J.*, vol. 38, 1993, pág. 231-256.
4. AIPPI, *Annuaire 1988/N, Congrès d'Amsterdam 1989*, 4-10 juin 1988, Rapports des Groupes Q57, Protection du logiciel, AIPPI Zurich, 1988.
5. ÁLVAREZ VALENZUELA, D., PAIVA VÉLIZ, M., “Reformas a la ley chilena de propiedad intelectual: el desafío de una regulación equilibrada”, *ICTSD 11*, nº 2 julio, 2010, artículo en línea; disponible en: <http://ictsd.org/i/news/puentes/80829/> (última consulta 1 de junio, 2013).
6. ANTEQUERA PARILLI, R., ROGEL VIDE, C. (coords.), *Los límites del derecho de autor*, Reus, Madrid, 2006.
7. APRILL, E.P., *The Supreme Court's Opinions in Bilski and the Future of Tax Strategy Patents*, Legal Studies Paper No. 2010-35, *Journal of Taxation* 81, Loyola Law School, Los Angeles, 2010.
8. AREEDA, P.E., “Essential facilities: an epithet in need of limiting principles”, *Antitrust L.J.* 58, 1990, pág. 841-853.
9. AREZZO, E. & GHIDINI, G. (eds.), *Biotechnology and Software Patent Law. A comparative review of new developments*, Cheltenham, Edward Elgar, 2011.
10. BAINBRIDGE, D., *Intellectual Property*, Harlow, Munich, Pearson, 2012.
11. BAINBRIDGE, D., *Legal Protection of Computer Software*, Sussex, 5ª ed., Tottel Publishing, 2008.

12. BAINBRIDGE, D., *Software Copyright Law*, London, 4ª ed., Butterworlds, 1999.
13. BAKELS, R., HUGENHOLTZ, B., “The Patentability of Computer Programs: Discussion of European-Level Legislation in the Field of Patents for Software”, European Parliament, 2002.
14. BAND J., KATOH, M., *Interfaces on Trial: Intellectual Property and Interoperability in the Global Software Industry*, Westview Press, Boulder, 1995.
15. BAND, J., “The Digital Millenium Copyright Act”, ALA Washington Office, 1998, pág. 11, disponible en: <http://www.ala.org/advocacy/copyright/dmca> (última consulta 1 de junio, 2013).
16. BAND, J., KATOH, M., *Interfaces on Trial 2.0*, Cambridge, London, MIT Press, 2010; disponible en: <http://mitpress.mit.edu/books/interfaces-trial-20> (última consulta 1 de junio, 2013)
17. BARTMANN, J, *Grenzen der Monopolisierung durch Urheberrecht am Beispiel von Datenbanken und Computerprogrammen*, Köln, Berlin, München, Carl Heymanns, 2005.
18. BATH, U., “Access to Information v. IPR”, EIPR 14(3), 2002, pág. 138-143.
19. BAUER, K.A., “Rechtsschutz von Computerprogrammen in der Bundesrepublik Deutschland - eine Bestandsaufnahme nach dem Urteil des Bundesgerichtshof von 9. Mai 1985”, CR 1, 1985, pág. 5-12
20. BEIER, F. (ed.) asistido por LLEWELYN, D., STAUDER, D., *German industrial property, copyright and antitrust laws: legal texts with introd.*, IIC-studies vol. 6, Chemie, Weinheim, 1983.
21. BEIER, F.K., GÖTTING, H.P., LEHMANN, M., MOUFANG, R. (eds.), *Urhebervertragsrecht*, C.H. Bech'sche Verlagsbuchhandlung, München, 1995.
22. BELLAMY, C., CHILD, G., *Derecho de la competencia en el Mercado Común*, ed. española a cargo de Enric Picañol, Madrid, 1992.

23. BELLO MARTÍN-CRESPO, P., HERNÁNDEZ RODRÍGUEZ, F., *Derecho de la Libre Competencia Comunitario y Español*, Navarra, Thomson Aranzadi, 2009.
24. BENKLER, Y., “Free as the Air to Common Use: First Amendment Constraints on Enclosure of the Public Domain” *New York University Law Review* 74, mayo, (1999), pág. 354-446.
25. BENTLY, L., “Harmonization by Stealth: Copyright and the ECJ”, ponencia presentada en la 18º Fordham Intellectual Property Law and Policy Conference, celebrada el 8-9 de abril de 2010 en Fordham Law School, New York; disponible en: [http://fordhamipconference.com/wp-content/uploads/2010/08/Bently\\_Harmonization.pdf](http://fordhamipconference.com/wp-content/uploads/2010/08/Bently_Harmonization.pdf) (última consulta 1 de junio, 2013).
26. BERCOVITZ RODRIGUEZ-CANO, R. (coord.), *Comentarios a la Ley de Propiedad Intelectual*, Madrid, 3º ed., Tecnos, 2007.
27. BERCOVITZ RODRÍGUEZ-CANO, R. (ed.) *Comentarios a la Ley de Propiedad Intelectual*, Madrid, 2ª ed., Tecnos, 1997.
28. BERCOVITZ RODRÍGUEZ-CANO, R. (ed.), *Comentarios a la Ley de Propiedad Intelectual*, Madrid, Tecnos, 1989.
29. BIJLSMA, M., DE BIJL P., KOCSIS, V., *Competition, innovation and intellectual property rights in software markets*, CPB document, nº 181, CPB Netherlands Bureau for Economic Policy Analysis, The Hague, 2009.
30. BISHOP, S., WALKER, M. *The Economics of EC Competition Law: Concepts, Application and Measurement*, London, 2ª ed., Sweet & Maxwell, 2002.
31. BLOCHER, W., WALTER, M., “Computer Program Directive” en WALTER, M., LEWINSKI, S. VON (eds.), *European copyright law: A Commentary* (rev. ed. and translation of the German ed. 2001), Oxford, Oxford Uni. Press, 2010, pág. 81-248.
32. BOEHM, B.W., «Software engineering», *IEEE Transaction on Computers*, C-25, n. 12, Diciembre, 1976, pág. 1226.

33. BOTANA AGRA, M., “Los derechos de explotación de la obra de autor en la Ley española de propiedad intelectual”, ADI, 1987, pág 37-86.
34. BRESNAHAN, T., “New Modes of Competition: Implication for the Future Structure of the Computer Industry”, SIEPR Discussion Paper 500, (1998), disponible en: <http://siepr.stanford.edu/publicationsprofile/1885> (Última consulta 1 de junio, 2013).
35. BREYER, S., “The Uneasy Case for Copyright: A Study of Copyright in Books, Photocopies, and Computer Programs”, Harv. L. Rev. 84, 1970, pág. 281-351.
36. BRISON, P., “WPPT, Art. 7”, en DREIER, T., HUGENGOLTZ, P.B. (eds.), *Concise European Copyright Law*, Alphen aan den Rijn, The Netherlands, Kluwer Law International, 2006, pág. 165-194.
37. BRUUN, N. (ed.), *Intellectual Property Beyond Rights*, Helsinki, WSOY, 2005.
38. BUYDENS, M., DUSOLLIER, S., “Les exceptions au droit d’auteur: Evolutions dangereuses”, CCE 9, septiembre, 2001, pág. 10-15.
39. CALVO CARAVACA, A., RODRÍGUEZ RODRIGO, J., *La Doctrina de las Infraestructuras Esenciales en Derecho Antitrust Europeo*, Revista de Derecho de la Competencia y la Distribución, La Ley, España, 2012.
40. CAPOBIANCO, A., “The essential facility doctrine: similarities and differences between the American and the European approach” Eur.L.Rev. 26, 6, 2001, pág. 548-564.
41. CARLETON, D., “A Behaviour-Based Model for Determining Software Copyright Infringement”, BTLJ 10, nº 2, 1995, pág 405-432.
42. CARR, H., ARNOLD, R., *Computer Software: legal protection in the United Kingdom*, London, 2ª ed., Sweet & Maxwell, 1992.
43. CERDA SILVA, A., (ed.), *Acceso a la cultura y Derechos de Autor. Excepciones y Limitaciones al Derecho de Autor*, Santiago de Chile, Ford Foundation, 2008.
44. CHIKOFFSKY, ELLIOT J., “Reverse Engineering and Design Recovery: A Taxonomy”, IEEE Magazine, January 1990, pág. 13-17.



45. CIFUENTES, C., *Reverse Compilation Techniques*. [tesis doctoral] Queensland University of Technology (US), Julio 1994, disponible en: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.105.6048> (última consulta 1 de junio, 2013).
46. CLAPES, A. L. et al, “Silicon Epics and Binary Bards: Determining the Proper Scope of Copyright Protection for Computer Programs”, *UCLA Law Review*, 34, 1987, pág 1493-1594.
47. COHEN, J.E., “Copyright and the Jurisprudence of Self-Help”, *BTLJ* 13, 1998, pág. 1089-1138.
48. COHEN, J.E., LEMLEY, M.A., “Patent Scope and Innovation in the Software Industry”, *Calif. L.Rev.*, n° 89, 2000, pág. 1-58; disponible en: <http://ssrn.com/abstract=282790> or <http://dx.doi.org/10.2139/ssrn.282790> (última consulta 1 de junio, 2013)
49. COLE, R., “Substantial similarity in the Ninth Circuit: a “virtually identical” “look and feel”? *Apple Computer, Inc. v Microsoft Corp.*, 35F 3.d 1435 (9<sup>th</sup> Cir. 1994)”, en *Santa Clara J.C.H.T.*, 11, Julio, 1995, pág. 417-427.
50. COLEMAN, G., “Code is Speech: Legal Tinkering, Expertise, and Protest among Free and Open Source Software Developers”, *Cultural Anthropology*, vol. 24, núm. 3, pág. 420-454.
51. COLOMBET, C., *Propriété littéraire et artistique et droits voisins*, Paris, 9<sup>a</sup> ed., Dalloz, 1999.
52. COLOMBET, C., traducido del francés por ALMEIDA, P., *Grandes principios del Derecho de Autor y los Derechos Conexos en el Mundo. Estudio de Derecho Comparado*, UNESCO/CINDOC, 1997.
53. CONDE GALLEGO, B., “La negativa unilateral de una empresa en posición de dominio a conceder una licencia para la utilización de un derecho de propiedad intelectual [Comentario a la Sentencia del TJCE de 29 de abril de 2004, asunto C-418, *IMS Health GmbH & Co. OHG v. NDC Health GmbH & Co. KG*”, en *ADI*, vol. 24, 2003, pág. 465-484.

- 54.COOPER DREYFUSS, R., FIRST, H., LEENHEER ZIMMERMAN, D. (eds.), *Expanding the Boundaries of Intellectual Property*, Oxford University Press, Oxford, 2001.
- 55.CORREA, C., “Fair use in the Digital Era”, IIC 33(5), 2002, pág. 570-585.
- 56.COTTIER, T., *Concise international and European IP law: TRIPS, Paris Convention, European enforcement and transfer of technology*, Alphen aan den Rijn, 2º ed., Kluwer Law International 2011.
- 57.CREWS, K.D., “Fair Use of Unpublished Works: Burdens of Proof and the Integrity of Copyright”, *Arizona State Law Journal*, vol. 31, nº.1, 1999, pág. 1-94.
- 58.CZARNOTA, B., HART, J., *Legal Protection of Computer Programs in Europe. A Guide to the EC Directive*, London, Butterworths, 1991.
- 59.DAVID, P.A., “Clio and the Economics of QWERTY”, *The American Economic Review*, nº 75, 1985, pág. 332-337.
- 60.DAVIDSON, D.M, “Protecting Computer Software: A Comprehensive Analysis”, *Arizona St. L.J.* 72, 1983, pág. 611-784.
- 61.DAVIES, G., *Copyright and the Public Interest*, IIC Studies, vol. 14, Weinheim, New York, VCH, 1994.
- 62.DAVIS, M., “Process and Product: Dichotomy or Duality”, *Software Engineering Notes*, ACM Press, vol. 20, n. 2, Abril 1995, págs. 17-18.
- 63.DENARDIS, L. (ed), *Opening Standards: The Global Politics of Interoperability*, Cambridge, The Mitt Press, 2011.
- 64.DEIRDRE, MULLIGAN, URBAN, JENNIFER M., et al., “Letter to CEN/ISSS (February 2003)”; disponible en:  
[http://www.law.berkeley.edu/clinics/samuelson/projects\\_papers/CENISSS\\_Letter\\_022103.pdf](http://www.law.berkeley.edu/clinics/samuelson/projects_papers/CENISSS_Letter_022103.pdf) (Última consulta 1 de junio, 2013).
- 65.DELGADO PORRAS, A. (ed.), *Derecho de autor y derechos afines al de autor: Recopilación de artículos*, Madrid, Instituto de Autor, 2007.

- 66.DENNING, P.J., "Is Computer Science Science?", *Communications of the ACM*, vol. 48, n.4, Abril, 2005, pág. 27-31.
- 67.DESJEUX, X., "Le droit d'auteur dans la vie industrielle", *RIDA*, julio 1975, pág. 125-137.
- 68.DEVLIN, A., *et.al.*, "Success, Dominance and Interoperability", *Ind. L.J.* 84, 2009, pág. 1156-1201.
- 69.DIZON, M.A., "Decompiling the Software Directive, the Microsoft CFI case and the i2010 strategy: How to reverse engineer an international interoperability regime", *CTLR* 14, 2008, pág. 213 - 228.
- 70.DONAHOE, R.V., "Does Intermediate Copying of Computer Software for the purpose of Reverse Engineering a Non-Infringing Product infringe the Copyright in the Software?", *Boston College Intellec. Prop. & Tech. F.*, nº 16, 2001, pág. 111-122.
- 71.DREIER, T., "The Council Directive of 14 May 1991 on Legal Protection of Computer Programs", *EIPR* 13, 1991, pág. 319-330.
- 72.DREIER, T., "Rechtsschutz von Computerprogrammen: Die Richtlinie des Rates der EG vom 14. Mai 1991" *CR* 7, 1991, pág. 577-584.
- 73.DREIER, T., HUGENHOLTZ, B. (eds.), *Concise Euroean Copyright Law*, Kluwer Law International, Alphen aan den Rijn, 2006.
- 74.DREIER, T., MEYER-VAN RAAJ, O., "Computerrecht", en SCHULTE, M., SCHRÖDER, R. (eds.), *Handbuch des Technikrechts*, Berlin, 2<sup>a</sup> ed., Springer, 2011, pág. 813-855.
- 75.DREIER, T., SCHULZE, G., *Urheberrechtsgesetz: Kommentar*, Munich, 4<sup>a</sup> ed., Beck, 2012.
- 76.DREXL, J. (ed.), *Research Handbook on Intellectual Property and Competition Law*, Cheltenham, Edward Elgar, 2008.
- 77.DREXL, J., *What is protected in a Computer Program. Copyright protection in the United States and Europe*, IIC Studies, vol. 15, Weinheim, VCH, 1994.

- 78.DUPONT, S. N., “The Copyright and Trade Secret Protection of Communication Software: Placing a Lock on Interoperability”, *J. Marshall J. Computer & Info. L.*, vol. 13 (1), 1994, pág. 17-42.
- 79.DURIE, R., “UK Copyright, Designs and Patents Act 1988” *IIC* 20(5), 1989, pág. 637-662.
- 80.DUSOLLIER, S., "La contractualisation de l'utilisation des oeuvres et l'expérience belge des exceptions impératives" *Propriétés Intellectuelles* , 2007, pág. 443-452; disponible en: [http://works.bepress.com/severine\\_dusollier/13](http://works.bepress.com/severine_dusollier/13) (última consulta 1 de junio, 2013).
- 81.DUSOLLIER, S., “Technology as an Imperative for Regulating Copyright: From the Public Exploitation to the Private Use of the Work”, *EIPR* 27(6), 2005, pág. 201-204.
- 82.DUSOLLIER, S., *Droit d'auteur et protection des œuvres dans l'univers numérique : droits et exceptions à la lumière des dispositifs de verrouillage des œuvres*, Bruxelles, 2<sup>a</sup> ed., Larcier, 2007.
- 83.DUSOLLIER, S., GINSBURG, J., HUGENHOLZT, P.B., et.al., *Le droit d'auteur : un contrôle de l'accès aux oeuvres ? - Copyright : a right to control access to works ?*, Cahier du CRID (Centre de Recherches Informatique et Droit), 18, Bruylant, Brussels, 2000.
- 84.DUSOLLIER, S., GINSBURG, J., HUGENHOLZT, P.B., et.al., *Le droit d'auteur : un contrôle de l'accès aux oeuvres ? - Copyright : a right to control access to works ?*, Cahier du CRID (Centre de Recherches Informatique et Droit), 18, Bruylant, Brussels, 2000.
85. ECIS, “Interoperability, Competition and open standards: The Keys to a Software Strategy for European citizens and Consumers”, Comunicado de 8 de junio de 2011, disponible en: [www.ecis.eu](http://www.ecis.eu) (última consulta 1 de junio, 2013).
- 86.ECONOMIDES, N., “Public Policy in Network Industries”, *Law and Economics Research Paper No. 06-49*, Septiembre, NYU, (2006); disponible en SSRN: <http://ssrn.com/abstract=936469> o <http://dx.doi.org/10.2139/ssrn.936469> (última consulta 1 de junio, 2013).

- 87.ECONOMIDES, N., “The economics of Networks”, *International Journal of Industrial Organization*, vol. 14, n. 2, 1996 , pág. 1-37.
- 88.ECONOMIDES, N., WHITE, L., “One-way networks, two-way networks, compatibility, and public policy”, *Department of Economics Working Paper ED-93-14*, NYU, 1993, pág. 1331-1358.
- 89.EECHOUD, M. VAN, HUGENHOLT, P.B., GOMPEL, S. VAN *et al*, *Harmonizing European Copyright Law. The Challenges of Better Lawmaking*, Alphen aan den Rijn, Kluwer Law International, 2009.
- 90.EILAM, E., *Reverse, Secrets of Reverse Engineering*, Wiley Publishing, Inc., Indianapolis, Indiana, 2005.
- 91.ENGELHARDT, S. VON, “The Economic Properties of Software”, *Jena Economic Research Papers*, Max Plank Institute of Economics, nº 45, 2008.
92. *European Copyright Code*, elaborado por *The Wittem Group*, publicado en abril, 2010; disponible en: [www.copyrightcode.eu](http://www.copyrightcode.eu) (última consulta 1 de junio, 2013).
- 93.FABIANI, M., “A Profile of Copyright in Today’s Society”, *Copyright* 18, 5, 1982, pág. 152-156.
- 94.FARRELL, J., “Arguments for Weaker IP protection in Network Industries, in Standards Policy for Information Infrastructure”, en KAHIN, B., ABBATE, J., (eds.), *Standards Policy for Information Infrastructure*, Cambridge, MA and London, The MIT Press, 1995, pág. 368-377.
- 95.FARRELL, J., “Standardization and Intellectual Property”, *Jurimetrics J.*, 30, 1989, pág. 35-50.
- 96.FARRELL, J., KATZ, M.L., “The Effects of Antitrust and Intellectual Property Law on Compatibility and Innovation”, *Antitrust Bulletin* 43, 1998, pág. 609-650.
- 97.FECHNER, F., *Geistiges Eigentum und Verfassung*, Tübingen, Mohr Siebeck, 1999.
- 98.FERNANDEZ MASIÁ, E., *La protección de los programas de ordenador en España*, Tirant Monografías nº 52, Valencia, Tirant lo Blanch, 1996.

- 99.FERNÁNDEZ NÓVOA, C., OTERO LASTRES, J.M., BOTANA AGRA, M., *Manual de la Propiedad Industrial*, Madrid, 2ª ed., Marcial Pons, 2013.
- 100.*Final Report of the National Commission on New Technological Uses of Copyrighted Works, July 31, 1978*, Washington: Library of Congress, 1979; disponible online: <http://www.digital-law-online.info/CONTU> (Última consulta 1 de junio, 2013).
- 101.FITZGERALD, B., CIFUENTES, C., LEHMANN, M., “Innovation, Software and Reverse Engineering”, *Santa Clara Computer & High. Tech. L.J.*, vol. 18, 2001-2002, pág. 121-159.
- 102.FRY, D., “Circumventing access controls under the Digital Millennium Copyright Act: Analyzing the securom debate”, *Duke Law & Technology Review* 5, 2009, pág. 1-24.
- 103.GALÁN CORONA, E., “En torno a la protección jurídica del soporte lógico”, en *Revista de Derecho Comercial y Obligaciones*, n. 93, 1983, pág. 329-359.
- 104.GALÁN CORONA, E., *Comentarios EDERSA*, T.V., vol 4ºB, Madrid, 1995.
- 105.GARCÍA VIDAL, A. (coord.), *El diseño comunitario. Estudios sobre el Reglamento (CE) núm. 6/2002*, Thomson-Reuters-Aranzadi, 2012.
- 106.GARZA BARBOSA, R., *Derechos de autor y derechos conexos : marco jurídico internacional : aspectos filosóficos, sustantivos y de litigio internacional*, México, Editorial Porrúa, 2009.
- 107.GASSER, U., PALFREY, J. “Breaking down digital barriers: when and how ICT interoperability drives innovation”, *Berkman Publication Series*, November (2007), disponible en <http://ssrn.com/abstract=1033226> (Última consulta 1 de junio, 2013).
- 108.GASSER, U., PALFREY, J., et al., “iTunes. How Copyright, Contract and Technology Shape the Business of Digital Media – A Case Study”, *Berkman Center for Internet & Society at Harvard Law School*, Junio, 2004.

- 109.GAUBIAC, F., “Medidas Tecnológicas e Interoperabilidad en el derecho de autor y los derechos vecinos”, e-Boletín de Derecho de Autor, Abril-Junio, 2007.
- 110.GEIGER, C., “Exclusivity in Copyright Law”, JET Law, nº 12, 2010, pág. 515-548.
- 111.GEIGER, C., “Fundamental Rights, a Safeguard for the Coherence of Intellectual Property Rights?” IIC 35(3), 2004, pág. 268-280.
- 112.GELLER, P.E. (ed.), *International Copyright Law and Practice*, New York, Matthew Bender, 2006.
- 113.GERVAIS, D.J., “Making Copyright Whole: A Principled Approach to Copyright Exception and Limitations”, University of Ottawa Law & Technology Journal, 2008, pág. 1-41.
- 114.GHIDINI, G., AREZZO, E., DE RAIS, C., ERRICO, P., “Il software fra brevetto e diritto d'autore: Primi appunti sulla Proposta di Direttiva comunitaria sulle “invenzioni attuate per mezzo di elaboratori elettronici”, Riv. dir. ind. I, 2005, pág. 46 - 81.
- 115.GHIDINI, G., *Aspectos actuales del Derecho Industrial: Propiedad Intelectual y Competencia*, Granada, Comares, 2002.
- 116.GHIDINI, G., *Innovation, competition and consumer welfare in intellectual property law*, Cheltenham, Edward Elgar Publishing, 2010.
- 117.GHIDINI, G., *Intellectual Property and Competition Law. The Innovation Nexus*, Cheltenham, Edward Elgar, 2006.
- 118.GIBBONS, L. J., “Boldly (not going) where we have gone before: The principles of the Law of Software Contracts”, Working Paper Series, October (2010); disponible en SSRN: <http://ssrn.com/abstract=1686383> (Última consulta 1 de junio, 2013).
- 119.GIBSON, J., “Once and Future Copyright”, Notre Dame L. Rev., nº 81, 2005-2006, pág. 167-242.

120. GILBERT-MACMILLAN, K., “Intellectual Property Law for Reverse Engineering Computer Programs in the European Community”, Santa Clara CHTLJ 9, 1993, pág. 247 - 264.
121. GINSBURG, J. “Four Reasons and a Paradox: the Manifest Superiority of Copyright Over Sui Generis Protection of Computer Software”, Columbia Law Review, vol. 8, nº 94, 1995, pág. 2559-2572.
122. GINSBURG, J., “Copyright Use and Excuse on the Internet”, Columbia VLA Journal of Law & the Arts, Vol. 24, 2000; disponible en SSRN: <http://ssrn.com/abstract=239747> (Última consulta 1 de junio, 2013)
123. GINSBURG, J., “Creation and Commercial Value: Copyright Protection of Works of Information”, Colum. L.R. 90, 7, nov. 1990, pág. 1865-1938.
124. GOLDSTEIN, P., “Infringement of Copyright in Computer Programs”, University of Pittsburgh Law Review, 47, 1986, pág. 1119-1130.
125. GOLDSTEIN, P., *Goldstein on Copyright*, Boston, Little, Brown & Co., 1989.
126. GOLDSTEIN, P., HUGENHOLTZ, B., *International Copyright. Principles, Law, and Practice*, Oxford, Oxford Univ. Press, 2010.
127. GÓMEZ PERALS, M., *La cesión de uso de los programas de ordenador*, Madrid, Colex, 1999.
128. GÓMEZ SEGADE, J.A., “La protección jurídica de los programas de los ordenadores electrónicos”, RDM, nº 113, Julio-Septiembre, 1969, pág. 413-436.
129. GÓMEZ SEGADE, J.A., *Tecnología y Derecho: Estudios jurídicos del Prof. Dr. h.c. José Antonio Gómez Segade recopilados con ocasión de la conmemoración de los XXV años de cátedra*, Madrid, Pons, 2001.
130. GOUTAL, J.L., “La protection juridique du logiciel”, D.Chron.197, núm. 9, 1984, pág. 197-203.
131. GRAVENREUTH, G., “Computerspiele und Urheberrecht”, CR 1987, pág. 161-166.
132. GRAVENREUTH, G., “Juristische relevante technische Fragen zur Beurteilung von Computer-Programmen”, GRURInt. 1986, pág. 720-727.



- 133.GRÜTZMACHER, M., “Abschnitt §69e. Dekompilierung”, en WANDTKE, A.-A., BULLINGER, W., *Praxiskommentar zum Urheberrecht*, München, 3<sup>a</sup> ed., Beck, 2009.
- 134.GUGLIELMETTI, G. “La Proposta di direttiva sulla brevettazione delle invenzioni in materia di software”, *Riv. dir. ind.* I, 2002, pág. 438-460.
- 135.GUIBAULT, L., “The nature and scope of limitations and exceptions to copyright and neighbouring rights with regard to general interest missions for the transmission of knowledge: prospects for their adaptation to the digital environment”, *e-Copyright Bulletin*, October-December, 2003.
- 136.GUIBAULT, L., *Copyright limitations and contracts: an analysis of the contractual overridability of limitations on copyright*, Information Law Series, vol, 9, The Hague, Kluwer Law International, 2002.
- 137.GUIBAULT, L., HUGENHOLTZ, P.B. (eds.), *The future of the public domain : identifying the commons in information law*, Alphen aan den Rijn, Kluwer Law Internat., 2006.
- 138.GUIBAULT, L., VAN DAALEN, O., *Unravelling the Myth around Open Source Licenses*, The Hague, TMC Asser Press, 2006.
- 139.GUILLOU, C.M., “The reverse engineering of Computer Software in Europe and in the United States: A Comparative Approach”, *Colum.-VLA J.L. & Arts* 22, 1998, pág. 533-556.
- 140.HABERSTUMPF, H., “Grundsätzliches zum Urheberrechtsschutz von Computerprogrammen nach dem Urteil des Bundesgerichtshofs vom 9. Mai 1985”, *GRUR* 3, 1986, pág. 222-235
- 141.HALVEY, J.K., “Rose by Any Other Name: Computer Programs and the Idea-Expression Distinction “, *Emory L. J.*, 34, 1985, pág. 741-776.
- 142.HARATSCH, A., KOENIG, C., PECHSTEIN, M., *Europarecht*, 8<sup>a</sup> ed., Tübingen, Mohr Siebeck, 2012.
- 143.HART, R. J., “Interoperability Information and the Microsoft Decision”, *EIPR*, 28 (7), 2006, pág. 361-365.

- 144.HARTE-BAVENDAMM, H., “Wettberwersrechtliche Aspekte des Reverse Engineering von Computerprogrammen”, GRUR 1990, pág. 657-664.
- 145.HEINDL, P., *A Status Report from the Software Decompilation Battle: A Source of Sores for Software Copyright Owners in the United States and the European Union?*, TTLF Working Paper No 1, 2008, disponible en: [http://www.law.stanford.edu/program/centers/ttlf/papers/heindl\\_wp1.pdf](http://www.law.stanford.edu/program/centers/ttlf/papers/heindl_wp1.pdf). (Última consulta 1 de junio, 2013).
- 146.HEREDERO HIGUERAS, M. “La protección de los programas de ordenador en el proyecto de Ley de Propiedad Intelectual”, RCDI 63, núm., 581, 1987, pág.1021-1044.
- 147.HEREDERO HIGUERAS, M., “Observaciones sobre el borrador de directiva comunitaria relativa a la protección jurídica de los programas de ordenador. Documento COM (88) 816 de 21 de diciembre de 1988”, *Tecnolegis*, nº. 1, octubre, 1989, pág. 14-16.
- 148.HERNÁNDEZ RODRÍGUEZ, F., “Derecho de autor y abuso de posición dominante en la Unión Europea”, ADI 16, 1995, pág. 331-344.
- 149.HILTY, R., “Intellectual Property and the European Community’s Internal Market Legislation Copyright in the Internal Market”, IIC 35(7), 2004, pág. 760-775.
- 150.HILTY, R., GEIGER, C., “Towards a new instrument of protection for software in the EU? Learning the lessons from the harmonization failure of software patentability”, MPI for Intellectual Property and Competition Law Research Paper No. 11-01, 2011; disponible en: <http://ssrn.com/abstract=1746846> (Última consulta 1 de junio, 2013).
- 151.HIMMA, K.E., “The Justification of Intellectual Property: Contemporary Philosophical Disputes”, *Journal of the American Society for Information Science and Technology*, vol. 59, issue 7, mayo 2008, pág. 1143 – 1161.
- 152.HOEREN, T., “The protection of software in the Federal Republic of Germany – recent developments”, CLP, March/April, 1990, pág. 134-137.

- 153.HOVENKAMP, H., "Standards Ownership and Competition Policy", Boston College Law Review 48, 2007, pág. 87-109.
- 154.HUGENHOLTZ, B. "Adapting Copyright to the Information Superhighway", paper based on studies prepared for the European Commission (DG XIII and DG XV) in 1994 and 1995.
- 155.HUGENHOLTZ, P.B. *et al.*, "The recasting of copyright and related rights for the knowledge economy: final report", European Commission DG Internal Market Study Contract No. ETD/2005/IM/D1/95, Institute for Information Law, Amsterdam, 2006.
- 156.*IEEE Standard Glossary of Software Engineering Terminology*, The Institute of Electrical and Electronics Engineers, IEEE Computer Society, New York, 1990.
- 157.IGNATIN, G.R., "Let the hackers hack: Allowing the Reverse Engineering of Copyrighted Computer Programs to Achieve Interoperability", UPLR, 140(5), 1992, pág. 1999-2050.
- 158.J., LEMAN, "The Future of Unpublished Works in Copyright Law After the Fair Use Amendment", Iowa J.Corp.L., 18, 1993, pág. 619-652.
- 159.JACOBS, M.A., "Copyright and Compatibility, LaST Frontier Conference on Copyright Protection of Computer Software", Jurimetrics J., nº 30, 1989, pág. 91-106.
- 160.JEONG, J.K., "Comparative Study on Legal Protection in the USA, EU, Japan and Korea for Computer Programs: Focus on Program Reverse Engineering", Institute of Intellectual Property Bulletin, 2006, pág. 140-149.
- 161.JOHNSON-LAIRD, A. "Software Reverse Engineering in the Real World", U.Dayton L.Rev. 19, 1994, pág. 843-902.
- 162.JONAS, H-H., "Obfuscating and Software Reverse Engineering in Practice", MAS-IP Diploma Papers & Research Reports, nº 13, 2008; disponible en: [www.bepress.com/ndsip/reports/art13](http://www.bepress.com/ndsip/reports/art13) (Última consulta 1 de junio, 2013).
- 163.JONGEN, H. (coord.), *Copyright software protection in the EC*, Computer Law Series, Deventer, Kluwer Law and Taxation Publ., 1994.

164. JONGEN, H. (coord.), *Copyright Software Protection in the EC*, Computer Law Series, Deventer, Kluwer Law and Taxation Publ., 1994.
165. *Jornadas de Estudios ALAI*, 2006, Barcelona: Derecho de autor y libertad de expresión, Barcelona, Huygens, 2008.
166. KABEL, J.J.C., MOM, G.J.H.M (eds), *Intellectual Property and Information Law, Essays in Honour of H. Cohen Jehoram*, Information Law Series 6, The Hague, London, Kluwer Law International, 1998.
167. KAHIN, B., ABBATE, J., (eds.), *Standards Policy for Information Infrastructure*, Cambridge, MA and London, The MIT Press, 1995.
168. KARJALA, D., “Copyright protection of operating systems, copyright misuse and antitrust”, *Cornell J.L. & Pub. Pol.* 9, 1999, pág. 161; disponible en: <http://homepages.law.asu.edu/~dkarjala/Articles/CornellJLPP1999.html#FN;Fd1> (Última consulta 1 de junio, 2013).
169. KARJALA, D., “Copyright, Computer Software, and the New Protectionism”, *Jurimetrics*, J., núm. 28, 1987, pág. 33-96.
170. KARL, C., *Der urheberrechtliche Schutzbereich von Computerprogrammen*, München, Beck, 2009.
171. KELLEHER, D., MURRAY, K., *IT Law in the European Union*, London, Sweet & Maxwell, 1999.
172. KEMPER, A., *Valuation of Network Effects in Software Markets. A Complex Networks Approach*, Springer, Berlin, Heidelberg, 2010
173. KENNETH B. KAHN (ed.), *PDMA Handbook of New Product Development*, 2<sup>a</sup> ed., John Wiley & Sons, 2005.
174. KINDERMANN, M., “Special Protection Systems for Computer Programs – A Comparative Study”, *IIC* 7(3), 1976, pág. 301-331.
175. KINDERMANN, M., “The International Copyright of Computer Software: History, Status and Developments”, *Copyright*, Vol 24, n.º. 4, 1988, pág. 201-226.
176. KLETT, A., *Intellectual Property Law in Germany*, München, Beck, 2008.

- 177.KNUTH, D.E., *The Art of Computing Programming*, 1ª ed., Barcelona, Reverté, 1980.
- 178.KOCHMANN, K., *Schutz des Know-how gegen ausspähende Produktanalysen (Reverse Engineering)*, Schriften zum Europäischen Urheberrecht vol. 8, Berlin, De Gruyter Recht, 2009.
- 179.KÖHLER, H., *Gesetz gegen den unlauteren Wettbewerb*, München, 29 ed., Beck, 2011.
- 180.KÖHLER, R., *Der urheberrechtliche Schutz der Rechenprogramme*, München, Beck, 1968.
- 181.KORTHALS, W.F. (ed.), *Information Law Towards the 21st Century*, Deventer, Boston, Kluwer Law and Taxation, 1992.
- 182.KREISS, R.A., “Section 117 of the Copyright Act”, *BYU L. Rev.* 4, 1991, pág. 1497-1546.
- 183.KRETSCHMER, M., DERCLAYE, E., FAVALE, M., WATT, R., *The Relationship Between Copyright and Contract Law 2010 (04)*, Intellectual Property Office of the United Kingdom, disponible en: [www.ipo.gov.uk/ipresearch-relation-201007.pdf](http://www.ipo.gov.uk/ipresearch-relation-201007.pdf) (Última consulta 1 de junio, 2013).
- 184.KROES, N., “Opening up Europe: from Common Standards to Open Data”, speech/11/596 en Open Forum Summit 2011, Brussels, 22 de septiembre, 2011; disponible en: [http://europa.eu/rapid/press-release\\_SPEECH-11-596\\_en.htm?locale=en](http://europa.eu/rapid/press-release_SPEECH-11-596_en.htm?locale=en) (Última consulta 1 de junio, 2013).
- 185.KUR, A., “Of Oceans, Islands and Inland Water – How much Room for Exception and Limitations under the three step test”, Max Planck Institute for Intellectual Property, Competition & Tax Law Research Paper Series No. 08-04, (2008); disponible en: <http://ssrn.com/abstract=1317707> ó <http://dx.doi.org/10.2139/ssrn.1317707> (Última consulta 1 de junio, 2013).
- 186.KUR, A., (ed.), *Intellectual Property Rights in a Fair World Trade System*, Cheltenham, Edward Elgar, 2011.

- 187.KURTZ, L., “Speaking to the Ghost: Idea and Expression in Copyright”, U. MIAMI L.REV 47, 1993, pág. 1221-1262.
- 188.LADDIE, H., PRESCOTT, P., VITORIA, M., *The Modern Law of Copyright and Designs*, Vol. II, Londres, 4ªed., Butterworths, 2011.
- 189.LALLANA, E. C., following inputs from the UNDP GIF Study Group, *e-Government Interoperability: Guide*, UNDP, 2007.
- 190.LAMPE, A.S., LERICHE, S., “Contrefaçon de logiciel, exception de décompilation et contournement de Mesures Techniques de Protection (MTP): l’affaire Nintendo, ou la difficulté de la preuve au secours de distributeurs de *linkers*”, Lamy droit de l’immatériel 57, 2010, pág. 22-27.
- 191.LEE, D.L., “Reverse Engineering of Computer Programs under the DMCA: Recognizing a “Fair Access” Defense”, Marq. Intell. Prop. L. Rev., vol. 10, 2006, pág. 537-573.
- 192.LEHMANN, M. “Die Europäische Richtlinie über den Schutz von Computerprogrammen”, GRUR Int. 1991, pág. 327-336.
- 193.LEHMANN, M. (ed.), *Rechtsschutz und Verwertung von Computerprogrammen*, München, 2ª ed., Beck, 1993.
- 194.LEHMANN, M., "Erwiderung. Reverse Engineering ist kein Vervielfältigung i.S.d. §§ 16, UrhG 53", CR 6, 1990, pág. 94.
- 195.LEHMANN, M., “Freie Schnittstellen (“Interfaces”) und freier Zugang zu den Ideen (“Reverse Engineering”): Schranken des Urheberrechtsschutzes von Software”, CR 5, 1989, pág. 1057-1063.
- 196.LEHMANN, M., TAPPER, C (eds.), *A Handbook of Software European Law*, Oxford, Clarendon Press, (vol. 1) 1993, (vol.2) 1995.
- 197.LEINEMANN, F., Die Sozialbindung des “Geistigen Eigentums”, Nomos, Baden, 1998.
- 198.LEMLEY, M. A., “Intellectual Property Rights and Standard-Setting Organizations” , California Law Review, Vol. 90, (2002), pág. 1889-1973;

- disponible en: <http://ssrn.com/abstract=310122> y <http://dx.doi.org/10.2139/ssrn.310122> (Última consulta 1 de junio, 2013).
- 199.LEMLEY, M.A., MENELL, P.S., MERGES, R.P., et al., *Software and Internet Law*, New York, 5ª ed., Wolters Kluwer Law and Business.
- 200.LENCE REIJA, C., “El objeto protegido en la Directiva sobre diseño industrial”, ADI 19, 1998, pág. 273-306.
- 201.LEPAGE, A., “Panorama general de las excepciones y limitaciones al derecho de autor en el entorno digital”, E-Boletín de derecho de autor, Enero-Marzo, 2003.
- 202.LESSIG, L., *The future of Ideas: The Fate of the Commons in a connected world*, New York, Random House, 2001.
- 203.LÉVÊQUE, F., “Innovation, Leveraging and Essential Facilities: Interoperability Licensing in the EU Microsoft Case” *World Competition* 28, 2005, pág. 71 - 91.
- 204.LÉVÊQUE, F., *The Economics of Patents and Copyright*, Paris, La Découverte, 2003.
- 205.LEWINSKI, S. VON, *International Copyright Law and Policy*, Oxford, New York, Oxford University Press, 2008.
- 206.LIM, D., “Copyright under siege: An economic analysis of the essential facilities doctrine and the compulsory licensing of copyright works”, *Alb. L.J.Sci. & Tech*, 17, 2007, pág. 481-559.
- 207.LITMAN, J., “The Exclusive Right to Read”, *Cardozo Arts&Ent. L.J.* 13, 1994 pág. 29; disponible en: <http://www-personal.umich.edu/~jdlitman/papers/read.htm> (Última consulta 1 de junio, 2013).
- 208.LODIGKEIT, K., *Intellectual Property Rights in Computer Programs in the USA and Germany*, Frankfurt am Main, New York, Peter Lang, 2006.
- 209.LOEWENHEIM, U., DIETZ, A., SCHRICKER, G., *Urheberrecht: Kommentar*, München, 6ª ed., Beck, 2010.

- 210.LOGAN, M., “The Section 117 adaptation defense and Krause v. Titleserv. Inc.: An analysis of the various circuit’s approaches”, *Cardozo Arts & Ent. L.J.* 26, 2008, pág. 269-300.
- 211.LÓPEZ MAZA, S., *Límites del derecho de reproducción en el entorno digital*, Granada, Comares, 2009.
- 212.LÓPEZ-TARRUELLA MARTÍNEZ, A., “Las licencias de uso de bienes digitales: El difícil equilibrio entre los titulares de los derechos y los usuarios”, *SCRIPT-ed* (3)4 June, 2006, pág. 437- 462.
- 213.LUCAS, A, *Traité de la propriété littéraire et artistique*, 3ª ed., LexisNexis Litec, Paris, 2006.
- 214.LUCAS, A., *La Protection des Créations Industrielles Abstraites*, Centre D’Études Internationales de la Propriété Industrielle, Paris, Libraries Techniques (LITEC), 1975.
- 215.LUNNEY, G., “Lotus v. Borland: Copyright and Computer Problem”, *Tul. L. Rev.* N° 70, 6 Part B, 1996, págs. 2397-2436.
- 216.MAIR, C. “Openness, Intellectual Property and Standardization in the European Sector”, *IP Theory*, Vol. 2(2), 3, pág. 1-16; disponible en: <http://www.repository.law.indiana.edu/ipt/vol2/iss2/3> (última consulta 1 de junio, 2013).
- 217.MÄKELÄ, M.M., “Software Business: Position As a Field of Research and Avenues for Scholarly Contributions”, 14th International Conference of the International Association for Management of Technology (IAMO), Viena, Austria, Mayo 22-26, 2005, pág. 1-12.
- 218.MARLY, J., “Der Schutzgegenstand des urheberrechtlichen Software-schutzes”, *GRUR* 2012, pág. 773-780.
- 219.MARLY, J., *Urheberrechtsschutz für Computersoftware in der Europäischen Union*, München, C.H. Beck, 1995.
- 220.MASKUS, K.E. (ed.), *Intellectual Property, Growth and Trade*, Amsterdam, Elsevier, 2008.



- 221.MASSAGUER FUENTES, J., “La adaptación de Ley de Propiedad Intelectual a la Directiva CEE relativa a la protección jurídica de los programas de ordenador”, RDM, núm. 199-200, 1991, pág. 39-74.
- 222.MCMAHON, K., “Interoperability: indispensability and special responsibility in High Technology Markets”, Tul. J. Tech. & Intell. Property, vol. 9, 2007, págs. 123 a 172.
- 223.MENELL, P., “An analysis of the Scope of Copyright Protection for Application Programs”, Stand. L. Rev. 41, 1989, pág. 1045-1104.
- 224.MENELL, P., “The Challenges of Reforming Intellectual Property Protection for Computer Software”, Colum. L. Rev. 94, 1994, pág. 2644-2654.
- 225.MENINSKY, C., “Locked Out: New Hazards of Reverse Engineering”, J. Marshall J Computer & Info. L. 21, 2003, pág. 591-625.
- 226.MERGES, R., *Intellectual Property in the new technology age*, New York, Aspen, Wolters Kluwer Law & Business, 2009.
- 227.MESSERSCHMITT, D.G., SZYPERSKI, C., “Industrial and Economic Properties of Software: Technology, Processes, and Value”, Report UCB//CSD-00-1130, Berkeley University, California, 2000, disponible en: <http://www.eecs.berkeley.edu/~messer/PAPERS/01/Software-econ/Software-econ-TR-footnotes.PDF>, (Última consulta 1 de junio, 2013).
- 228.MINASI, M., *The Software Conspiracy: Why Companies Put Out Faulty Software, How They Can Hurt You And What You Can Do About It*, McGraw-Hill, 1999.
- 229.MÖHRING, P., “Die Schutzfähigkeit von Programmen für Datenverarbeitungsmaschinen”, GRUR 6, 1967, pág. 269-278.
- 230.MONTI, G., *EC Competition Law (Law in context)*, Cambridge, Cambridge University Press, 2007.
- 231.MORITZ, H.W., TYBUSSECK, B., *Computersoftware: Rechtsschutz und Vertragsgestaltung – Eine fächerübergreifende Darstellung nach deutschen und EG-Recht*, München, 2ª ed., Beck, 1992.

- 232.MYLLY, U., “An Evolutionary Economics Perspective on Computer Program Interoperability and Copyright”, IIC 41(3), 2010, pág. 284-315.
- 233.NACK, R., *Die patentierbare Erfindung unter den sich wandelnden Bedingungen von Wissenschaft und Technologie*, Max-Planck-Institut für Geistiges Eigentum, Wettbewerbs- und Steuerrecht, Band 121, Köln, Berlin, Bonn, München, Carl Heymanns, 2002.
- 234.NIMMER, M. *et al*, *Cases and materials on Copyrigh and other aspects of Entertainment litigation including Unfair Competition, Defamation, Privacy*, New York, 6ª ed., Lexis, cop., 2000.
- 235.NIMMER, M., NIMMER D., *Nimmer on Copyright*, vol. 4, New York, Bender, 2009.
- 236.NORTON, P., *Introducción a la computación*, México, 6ª ed., Mc Graw Hill, 2006.
- 237.OKEDIJI, R., “Toward an International Fair Use Doctrine”, *Columbia Journal of Transnational Law*, vol. 39, 2000, pág. 75-176.
- 238.OTERO LASTRES, J.M., “Concepto de diseño y requisitos de protección en la nueva Ley 20/2003”, *ADI* 24, 2003, pág. 53-80.
- 239.PALFREY, J., GASSER, U., *Interop: The Promise and Perils of Highly Interconnected Systems*, New York, Basic Books, 2012.
- 240.PASCUAL y VICENTE, J., *Diccionario de derecho y economía de la competencia en España y Europa*, Madrid, Civitas, 2002.
- 241.PEARSON, H., MILLER, C., TURTLE, N., “Commercial Implications of the European Software Copyright Directive”, *International Quarterly* 5(1), 1993, pág.108-126.
- 242.PERALS, M., *La cesión del uso de los programas de ordenador*, Colex, Madrid, 1999.
- 243.PERZANOWSKI, A.K., “Rethinking Anticircumvention’s Interoperable Policy”, *U.C. Davis L. Rev.* 42, 2008-2009, pág. 1549-1620.

- 244.PESSACH, G., “Copyright Law as a silencing restriction on non infringing materials: unveiling the scope of copyright’s diversity externalities”, Southern California L. Rev., vol. 76, 2003, pág. 1067-1104.
- 245.PESSACH, G., “Reverse Exclusion in Copyright Law - Reconfiguring Users' Rights”, 2011; disponible en SSRN: <http://ssrn.com/abstract=1813082> or <http://dx.doi.org/10.2139/ssrn.1813082> (Última consulta 1 de junio, 2013).
- 246.PETTEYS, D.A., “The Freedom to Link: The Digital Millennium Copyright Act Implicates the First Amendment in Universal City Studios, Inc. v. Reimerdes”, Seattle University Law Review, Vol. 25, nº 1, 2001, pág. 287-341.
- 247.PINO ABAD, M. “La protección de los programas de ordenador en la Ley de Propiedad Intelectual de 11 de noviembre de 1987”. Derecho de los Negocios, vol. 8, 1988, pág. 7 -12.
- 248.PLAISANT, R., “La protection du logiciel par le droit d’auteur”, Gaz. Pal., núm. 7, 1983, pág. 348.
- 249.PLOTKIN, R., “Fighting Keywords: Translating the First Amendment to Protect Software Speech”, Journal of Law, Technology and Policy, (2003), pág. 329-404; disponible en: <http://ssrn.com/abstract=631861> (Última consulta 1 de junio, 2013).
- 250.POIKELA, P., *IPR Protection of application programming interfaces*, Saarbrücken, LAP, Lambert Acad. Publ., 2010.
- 251.POWEL, S.J., TRAMMELL, C.J., LINGER, R., et al., *Cleanroom Software Engineering, Tecnology and Process*, Addison-Wesley, kindle ed., Mass., 1999.
- 252.PRESSMAN, R., HERRON, R, *Software Shock: The Danger and the Opportunity*, New York, Dorsett House, 1991.
- 253.PRESSMAN, R., *Ingeniería del Software. Un enfoque práctico*, adaptado por INCE, D., España, 5ª ed., Mc Graw Hill, 2005.
- 254.RAJANI, A.H., “Davidson & Associates v. Jung: (Re)interpreting Access Controls”, BTLJ 21, 2006, pág. 365-396.

- 255.RAMSEY, H., “Misusing Antitrust: The Search for Functional Copyright Misuse Standards”, *Stan. L. Rev.* 46, 1994, pág. 401-448.
- 256.REDING, V., “Filling the Missing Link in Fundamental Rights Protection: The European Union’s Accession to the European Convention on Human Rights”, *Comitte of Ministers of the Council of Europe, Strasbourg*, 7 de julio, 2010; disponible en <http://www.coe.int> (Última consulta 1 de junio, 2013).
- 257.REICHMAN, J.H., “Of Green Tulips and Legal Kudzu: Repacking Rights in Subpatentable Innovation”, *Vand. L. Rev.* 53, 2000, pág. 1744-1798.
- 258.REICHMAN, J.H., “Legal Hybrids between the Patent and Copyright Paradigms”, *Columbia Law Review* 94(8), 1994, pág. 2432-2558.
- 259.RILEY, M. C., “The Need for Software Innovation Policy”, *Journal of Telecommunications and High Technology Law* 5(589), (2007), pág. 529-628; disponible en <http://ssrn.com/abstract=910045> (Última consulta 1 de junio, 2013).
- 260.ROOIJEN, A. VAN, “Case Comment”, *AMI* 6, 2007, pág. 192-194.
- 261.ROOIJEN, A. VAN, “Essential Facilities: Exploring the Software Directive’s equilibrium between intellectual property rights and competition law”, *CRI* 2007, pág. 129 -137.
- 262.ROOIJEN, A. VAN, *The software interface between Copyright and Competition Law, A legal analysis of Interoperability in Computer Programs*, Great Britain, Kluwer Law International, 2010.
- 263.RÖTTINGER, M., “Abkehr vom Urheberrechtsschutz für Computerprogramme?”, *IuR* 1, 1986, pág. 12-18.
- 264.RUBINI, L. (ed.), *Microsoft on Trial. Legal and Economic Analysis of a Transatlantic Antitrust Case*, Cheltenham, Edward Elgar, 2010.
- 265.SADD, T.J., “Fair Use as a Defense under the Digital Millennium Copyright Act’s Anti-Circumvention Provisions”, *George Mason Law Review*, Vol. 10, núm. 2, 2002, pág. 321- 344.

- 266.SAMUELSON, P., “Computer Programs, user interfaces and Section 102(b) of the Copyright Act of 1976: A critique of Lotus v. Paperback”, High Tech. L. J. 6, 1991, pág. 209-269.
- 267.SAMUELSON, P., “Are patents on interfaces impeding interoperability?”, Minn. L. Rev. 93, (2008), pág. 1943-2019; disponible en: <http://scholarship.law.berkeley.edu/facpubs/1712>, (Última consulta 1 de junio, 2013).
- 268.SAMUELSON, P., “Comparing U.S. and E.C. Copyright Protection for Computer Programs: Are They More Different Than They Seem?”, Journal of Law and Commerce 13, 1994, pág. 279 - 300.
- 269.SAMUELSON, P., “CONTU Revisited: The Case Against Copyright Protection for Computer Programs in Machine Readable Form”, Duke Law Journal, 4, Septiembre, 1984, pág. 663 - 769.
- 270.SAMUELSON, P., “Copyright and Freedom of Expression in Historical Perspective”, J. Intell. Prop. L. 10, 2002-2003, pág. 319-344.
- 271.SAMUELSON, P., “Generativity of Sony v. Universal: The Intellectual Property Legacy of Justice Stevens”, Fordham L. Rev., vol. 74, 2005, pág. 1831-1876.
- 272.SAMUELSON, P., “Intellectual Property and the Digital Economy: Why the Anticircumvention Regulations Need to be Revised”, BTLJ 14, 1999, pág. 519 – 566.
- 273.SAMUELSON, P., “Preliminary Thoughts on Copyright Reform”, Utah Law Review 3, 2007, pág. 551-571.
- 274.SAMUELSON, P., “The Strange Odyssey of Software Interfaces and Intellectual Property Law”, UC Berkeley Public Law Research Paper No. 1323818, 2008, disponible en SSRN: <http://ssrn.com/abstract=1323818> ó <http://dx.doi.org/10.2139/ssrn.1323818> (Última consulta 1 de junio, 2013)
- 275.SAMUELSON, P., RANDALL, D., MITCHELL, D. KAPOR, REICHMAN, J.H., “ A Manifesto concerning the Legal Protection of Computer Programs”, Columbia Law Review 94(8), 1994, pág. 2308-2431.

- 276.SAMUELSON, P., SCOTCHMER, S., “The Law and Economics of Reverse Engineering”, Yale Law Journal, 111(7), 2002, pág. 1575-1664.
- 277.SAMUELSON, P., VINJE, T., CORNISH, W., “Does Copyright protection under the EU Software Directive extend to computer program behaviour, languages and interfaces?”, EIPR 34(3), 2012, pág. 158-166.
- 278.SANTORO, E., “Cenni sul diritto di riproduzione delle dell’ingegno”, Riv. Dir. Comm. I, 1966, pág. 65.
- 279.SAPHIRO, C., VARIAN, H.R., *Information Rules: A strategic guide to Network Economy*, Boston, Massachusetts, Harvard Business School Press, 1999.
- 280.SCHMIDTCHEN, D., KOBOLDT, C., “A Pacemaker That Stops Halfway: The Decompilation Rule in the EEC Directive on the Legal Protection of Computer Programs”, International Review of Law and Economics, 13, 1993, pág. 413-429.
- 281.SCHNEIDER, J. “Vervielfältigungsvorgänge beim Einsatz von Computerprogrammen: Vorrang urheberrechtlicher Grundwertungen gegenüber technischen Zufälligkeiten”, CR 6, 1990, pág. 503-508.
- 282.SCHOVSBO, J., “As If Made for Each Other – Intellectual Property Rights and Protection of Compatible Products”, IIC 20(5), 1998, pág. 510-534.
- 283.SCHRICKER, G. (ed.), *Urheberrecht, Kommentar*, München, Beck, 1987.
- 284.SCHULTE, D., “Der Referententwurf eines Zweiten Gesetzes zur Änderung des Urheberrechtsgesetzes”, CR 8, 1992, pág. 648.
- 285.SCHULTE, M., SCHRÖDER, R. (eds.), *Handbuch des Technikrechts*, Berlin, 2<sup>a</sup> ed., Springer, 2011.
- 286.SCHULZE, G. “Urheberrechtsschutz von Computerprogrammen -geklärte Rechtsfragen oder blosse Illusion”, GRUR 1985, pág. 997-1008.
- 287.SCHUMPETER, J.A., *Capitalism, Socialism and Democracy*, Londres, George, Allen & Unwin Ltd, 1943.

288. SCOTCHMER, S., *Innovation and Incentives*, Cambridge, MA, MIT Press, 2004.
289. SEGURA SÁNCHEZ, J., “La Política de Defensa de la Competencia: objetivos, fundamentos y marcos constitucionales”, *Ekonomiaz* núm. 61, 1º cuatrimestre, 2006, pág. 16-39.
290. SEVERANCE, C., “Posix: A model for future computing”, *Standard for Information*, January 1999, pág. 131-132.
291. SHEFF, D., *Game Over: how Nintendo zapped an american industry, captured your dollars, and enslaved your children*, New York, Random House, 1993.
292. SHELLEY, A., “An Alternative Approach to Computer Pirating Disputes, the Mookin-Jones Settlement: IBM v. Fujitsu”, *Temp. Int'l & Comp. L.J.* 3, 1989, pág. 113-128.
293. SHEMTOV, N., *The Legal Regulation of Decompilation of Computer Programs: Excessive, Unjustified and in Need of Reform*, 2012 (no publicada), disponible en: <https://qmro.qmul.ac.uk/xmlui/handle/123456789/3132> (Última consulta 1 de junio, 2013).
294. SHERMAN, B., WISEMAN, L., *Copyright and the Challenge of the New*, Alphen aan den Rijn, Kluwer Law International, 2012.
295. SHUSTER, T., “Originality in Computer Programs and Expert Systems”, *Transnat'l Law* 5, 1992, pág. 13-17.
296. SIDLER, P., *Der Schutz von Computerprogrammen im Urheber- und Wettbewerbsrecht*, Basel, Suiza, Verlag für Recht und Gesellschaft AG, 1969.
297. SILICIA, M. A., DE LA MORENA, V., “Qué es reingeniería de software”, artículo en línea 2009; disponible en: <http://cnx.org/content/m17438/latest>, (última consulta 1 de junio, 2013).
298. SIPONEN, M., “A Justification for Software Rights”, *ACM SIGCAS Computers and Society: selection of best papers 2004-2006*, vol. 36, issue 3, septiembre 2006, pág. 11-20.

- 299.SIRINELLI, P., “Excepciones y limitaciones al derecho de autor y los derechos conexos”, en Taller sobre cuestiones de aplicación del Tratado de la OMPI sobre derecho de autor y Tratado de la OMPI sobre interpretación o ejecución y fonogramas, Ginebra 6 y 7 de diciembre de 1999, Doc. WCT-WPPT/IMP/1, 1999; disponible en: [http://www.wipo.int/meetings/es/doc\\_details.jsp?doc\\_id=1266](http://www.wipo.int/meetings/es/doc_details.jsp?doc_id=1266) (Última consulta 1 de junio, 2013).
- 300.SOMA, J.T., WINFIELD, G., FRIESEN, L., “Software Interoperability and Reverse Engineering”, Rutgers Computer & Tech. L.J. 20, 1994, pág. 189-257.
- 301.SOUZA, C., REDMILES, D., CHENG, L., MILLEN, D., PATTERSON, J., “Sometimes you need to see through walls – a field study of application programming interfaces”, Computer Supported Cooperative Work (CSCW), ACM Press 10, 2004.
- 302.SPINDLER, G., SCHUSTER, F., *Recht der elektronischen Medien*, München, 2ª ed., Beck, 2011.
- 303.SPOOR, J.H., “Copyright Protection and Reverse Engineering of Software: Implementation and Effects of the EC Directive”, U. Dayton Law Review 19, 1994, pág. 1063-1066.
- 304.SPOOR, J.H., et al., *Auteursrecht: auteursrecht, naburige rechten en databankenrecht*, Deventer, Kluwer, 2005.
- 305.SPULBER, D. “Competition Policy and the Incentive to Innovate: The Dynamic Effects of Microsoft v. Commission”, Yale J. on Reg. 25(2), 2008, pág. 247-302.
- 306.STAFFELBACH, O., *Die Dekompilierung von Computerprogrammen gemäss Art. 21 URG*, Bern, Stämpfli, 2003.
- 307.STERN, R.H., “Section 117 of the Copyright Act: Charter of the Software Users' Rights or an Illusory Promise”, W. New Eng. L. Rev., 7, 1984-1985, pág. 459-486.
- 308.STOBBS, G.A., *Software patents*, Gaithersburg, Aspen Law & Business, 2000.



- 309.STORK, A., “The use of arbitration in copyright disputes: IBM v. Fujitsu”, High Technology Law Journal, 1987, pág. 1-25.
- 310.STRAUS, J. (ed.), *Patents and Technological Progress in a Globalized World: Liber Amicorum*, Berlin, Springer, 2009.
- 311.STROWEL, A., *Droit d’Auteur et Copyright. Divergences et convergences. Etude de droit comparé*, E.E. Bruylant, Bruselas, 1993.
- 312.SURBLYTÉ, G., *The Refusal to Disclose Trade Secrets as an Abuse of Market Dominance – Microsoft and Beyond*, Berne, Stämpfli 2011.
- 313.SWANSON, D.G., BAURNOL, W.J., “Reasonable and Nondiscriminatory (RAND) Royalties, Standards Selection, and Control of Market Power”, *Antitrust Law Journal* 73, 2005, pág. 1-58.
- 314.SYNDIKUS, B., “Computerspiele und Urheberrecht”, CR 1988, pág. 819-823.
- 315.SYNODINOU, T., “The Lawful User and a Balancing of Interests in European Copyright Law”, IIC 41(7), 2010, pág. 819-844.
- 316.TIEN, L., “Publishing Software as a Speech Act”, *Berkely Technology Law Journal*, nº 15, 2000, pág. 629-711; disponible en: <http://www.law.berkeley.edu/journals/btlj/articles/vol15/tien/tien.html>, (última consulta 1 de junio, 2013);
- 317.TOUBOL, F., (Traducción de ESPANÉS, M.), *El software : análisis jurídico*, Zavalía, Buenos Aires, 1990.
- 318.TURNEY, J., “Defining the Limits of the EU Essential Facilities Doctrine on Intellectual Property Rights: The Primacy of Securing Optimal Innovation”, *NW.J.TECH. & INTELL. PROP.* 3, 2005, pág. 179-202.
- 319.ULLRICH, H., KÖRNER, E., *Der International Softwarevertrag*, Frankfurt am Main, 2ª ed., Verl. Recht und Wirtschaft, 2006.
- 320.ULMER, E., *Der Urheberschutz wissenschaftlicher Werke unter besonderer Berücksichtigung der Programme elektronischer Rechenanlagen*, München, Bayerische Akademie der Wissenschaften, Philosophisch-Historische Klasse, Sitzungsberichte, 1967.

321. VERSTRYNGE, J.F., "Protecting Intellectual Property Rights within the New Pan-European Framework - Computer Software", en LEHMANN, M. & TAPPER, C., *A Handbook of Software European Law*, Oxford, Clarendon Press, (vol. 1) 1993, (vol.2) 1995, pág. 1-10.
322. VEZZOSO, S., "Copyright, Interfaces and a Possible Atlantic Divide", JIPITEC 2, 2012, pág. 153-161.
323. VINJE, T., "Compliance with art. 85 in software licensing", ECLR 13, 4, 1992, págs. 165-173.
324. VINJE, T., "Copyright Imperilled?", EIPR 4, 1999, pág. 192-297.
325. VINJE, T., "El desarrollo de productos interoperables bajo la directiva sobre software de la CE", RGD, año L, núm. 589-90, octubre-noviembre, 1993, págs. 10133-10153.
326. VINJE, T., "Magill: its impact on the information technology industry", EIPR 14, 1992, pág. 397-402.
327. VINJE, T., "The EC Directive on the protection of Computer Programs and the Question of Interoperability" (Die EG\_Richtlinie zum Schutz von Computerprogrammen und die Frage der Interoperabilität), GRURInt. 4, 1992, pág. 250 - 259.
328. VIVANT, M., "Deuxième exception légale: la décompilation", Lamy droit de l'informatique et des réseaux, 199-202, 2005, pág. 122-126.
329. VIVANT, M., "Table ronde sur la protection juridique des logiciels", JCP Les cahiers de droit de l'entreprise 3, 1983, pág. 2.
330. WALLAS, G., *The Art of Thought*, New York, Harcourt, Brace and Company, 1926.
331. WALTER, M., LEWINSKI, S. VON (eds.), *European copyright law: a commentary* (rev. ed. and translation of the German ed. 2001), Oxford, Oxford Uni. Press, 2010.
332. WANDTKE, A.A., BULLINGER, W., *Praxiskommentar zum Urheberrecht*, Munich, 3ª ed., Beck, 2009.

333. WANNER, E., “Navigating the Nexus: DMCA Anti-Circumvention Protection of Computer Software”, *Ariz. St. L.J.* 43, 2011, pág. 1081-1106.
334. WATERS, P., LEONARD, P.G., “The lessons of recent EC and US Developments for Protection of Computer Software under Australian Law”, *EIPR* 13, 1991, pág. 124-131.
335. WEBE, S., *The Success of Open Source*, Cambridge, Harvard University Press, 2005.
336. WESTON, S., “Software Interfaces – Stuck in the Middle: The Relationship Between the Law and Software Interfaces in Regulating and Encouraging Interoperability”, *IIC* 43(4), 2012, pág. 427-450.
337. WESTON, S., KRETSCHMER, M., “Opens Standards in Government IT: A Review of the Evidence”, Centre for Intellectual Property & Management (CIPPM), Bournemouth University, (final draft 10 September 2012), disponible en: [www.cippm.org.uk/publications.html](http://www.cippm.org.uk/publications.html) (última consulta 1 de junio, 2013)
338. WIEBE, A., “Interoperabilität von Software. Art. 6 der Computerprogramm-Richtlinie aus heutiger Sicht”, *JIPITEC* 2, 2011, pág. 89-96.
339. XALABARDER, R., “Declaración por una interpretación equilibrada de la «Regla de los Tres Pasos» en el derecho de autor”, *ADI* 28, 2007-2008, págs. 1509-1516.
340. YOUNG, E.W., “Universal City Studios, Inc. v. Reimerdes: Promoting the Progress of Science and the Useful Arts by Demoting the Progress of Science and the Useful Arts [notes]”, *Northern Kentucky Law Review*, Vol. 28, nº 4, 2001, pág. 847-876.