

**Christian Hess Araya**

# **La dimensión jurídica del software**

***Naturaleza, tutela jurídica, contratos y responsabilidad***

*1ª edición electrónica  
(corresponde a la 2ª edición impresa de 2004)*

*San José, Costa Rica*

2007

ISBN 9977-13-278-X

© Esta obra está protegida por la legislación internacional de derechos de autor.  
Prohibida su reproducción total o parcial.

## Índice de contenido

|   |             |
|---|-------------|
| <b>DEDICATORIA</b>  | <b>III</b>  |
| <b>PRESENTACIÓN</b>   | <b>IV</b>   |
| <b>NOTAS PRELIMINARES A ESTA EDICIÓN ELECTRÓNICA</b>                                | <b>VIII</b> |
| <b>CAPÍTULO I - NATURALEZA JURÍDICA DEL SOFTWARE</b>                                | <b>1</b>    |
| ASPECTOS TERMINOLÓGICOS   | 1           |
| CARACTERIZACIÓN JURÍDICA DEL SOFTWARE   | 4           |
| CLASIFICACIONES DEL SOFTWARE  | 6           |
| <i>Desde la perspectiva técnica</i>   | 6           |
| <i>Desde la perspectiva jurídica</i>  | 7           |
| <b>CAPÍTULO II - LA TUTELA JURÍDICA DEL SOFTWARE</b>                                | <b>16</b>   |
| PLANTEAMIENTO DEL PROBLEMA  | 16          |
| CONCEPTOS BÁSICOS DE PROPIEDAD INTELECTUAL  | 19          |
| PROTECCIÓN DEL SOFTWARE   | 21          |
| PROTECCIÓN POR MEDIO DEL DERECHO DE AUTOR   | 22          |
| <i>Panorámica del derecho autoral</i>   | 22          |
| <i>Aplicación al software</i>   | 26          |
| <i>El cuadro normativo</i>  | 36          |
| PROTECCIÓN POR MEDIO DE PATENTES DE INVENCIÓN                                       | 41          |
| <i>Nociones generales</i>   | 41          |
| <i>Aplicación al software</i>   | 44          |
| REGÍMENES DE PROTECCIÓN <i>SUI GENERIS</i>  | 50          |
| OTROS MEDIOS DE TUTELA  | 53          |
| <i>El software como expresión constitucionalmente protegida</i>                     | 53          |
| <i>Secretos comerciales</i>   | 55          |
| <i>Dibujos y modelos industriales</i>   | 59          |
| OTROS TEMAS PUNTUALES RELACIONADOS  | 60          |
| <i>Tutela de la propiedad intelectual de las bases de datos</i>                     | 60          |
| <i>Medidas tecnológicas de defensa de las obras digitales</i>                       | 67          |
| <i>Notas críticas sobre el problema de la piratería de software</i>                 | 75          |
| <b>CAPÍTULO III - CONTRATOS RELATIVOS AL SOFTWARE</b>                               | <b>78</b>   |
| CONTRATO DE LICENCIA DE USO DE SOFTWARE   | 78          |
| CESIÓN DE DERECHOS PATRIMONIALES DEL SOFTWARE                                       | 84          |
| DESARROLLO DE SOFTWARE A LA MEDIDA  | 87          |
| ASP: ARRENDAMIENTO EN LÍNEA DE APLICACIONES   | 91          |
| <b>CAPÍTULO IV - RESPONSABILIDAD CIVIL ORIGINADA EN EL DISEÑO Y USO DE SOFTWARE</b> | <b>96</b>   |

|   |            |
|---|------------|
| INTRODUCCIÓN  | 96         |
| ¿POR QUÉ FALLA EL SOFTWARE?   | 98         |
| CONDICIONES PARA LA EXISTENCIA DE UNA RESPONSABILIDAD Y SUS EVENTUALES EXIMENTES                      | 101        |
| <i>El software malicioso</i>  | 101        |
| <i>Diseño imprudente o negligente de aplicaciones</i>   | 104        |
| <i>La (tristemente célebre) cláusula “as is” y otros mecanismos de exoneración de responsabilidad</i> | 106        |
| SOLUCIONES PROPUESTAS   | 109        |
| SITUACIÓN NACIONAL  | 111        |
| <b>BIBLIOGRAFÍA</b>   | <b>114</b> |
| <b>ANEXO I: SITIOS RELEVANTES EN INTERNET</b>   | <b>122</b> |
| EN CASTELLANO   | 122        |
| EN INGLÉS   | 122        |
| <b>ANEXO II: DIRECTRICES RELATIVAS AL EMPLEO ILEGAL DE SOFTWARE EN LAS OFICINAS GUBERNAMENTALES</b>   | <b>124</b> |
| <b>SOBRE EL AUTOR</b>   | <b>128</b> |

# Dedicatoria

*A mi esposa e hijos, razón de ser de todos mis esfuerzos.*

*A las y los colegas estudiosos(as) del derecho informático iberoamericano,  
por su amistad y por el aprendizaje que he recibido de ellos(as).*

# Presentación

*Discurso pronunciado por el Dr. Alfredo Chirino Sánchez<sup>1</sup> durante el acto de presentación de la primera edición impresa de esta obra*

Muy buenas tardes, Dr. Román Solís, Magistrado de la Corte Suprema de Justicia, Lic. Christian Hess autor de la obra, familiares y amigos del autor, funcionarios judiciales, amigos y amigas todos.

Siempre he expresado mi regocijo por la presentación de una obra jurídica, no sólo porque cada una de ellas engalana el horizonte de proyección de la ciencia nacional, sino porque constituye un testimonio material del esfuerzo y el sacrificio de los autores nacionales por trascender.

Mi entusiasmo el día de hoy se ve especialmente recompensado no sólo porque presentamos un trabajo científico exhaustivo y valioso, sino porque es un aporte a una rama jurídica que, a pesar de su importancia en el tráfico de las relaciones comerciales del mundo globalizado, ha tenido escaso desarrollo y discusión en nuestro país.

Muchas son las razones por las cuales el desarrollo del derecho informático ha estado afectado de estos males, pero sin duda la más importante de ellas era la necesidad de que quien se dedique a ellas domine también técnicamente los complicados procesos que caracterizan a la informática. Esta combinación no es fácil de encontrar, lo que provoca pues un respetuoso recelo a analizar y discutir los problemas, generalmente muy complejos, que están implícitos en las relaciones jurídicas que tienen por objeto el estudio de las tecnologías de la información y de la comunicación.

Don Christian, querido amigo y brillante colega, tiene la doble condición de jurista y de Master en Ciencias de la Computación y de la Informática en la Universidad de Costa Rica, lo que lo convierte en una autorizada voz en el campo del derecho informático, lo que ha demostrado en este enjundioso estudio sobre el software y sus consecuencias jurídicas. No obstante, sus aportaciones no se han

---

<sup>1</sup> Director de la Escuela Judicial y Magistrado suplente de la Sala Tercera de la Corte Suprema de Justicia.

limitado al campo de este tipo de contratos, sino también en temas de tanta trascendencia para el país como lo son el de la protección de datos, el de la propiedad intelectual, y la materia del derecho electoral, donde también ha hecho valiosas aportaciones.

Sinceramente creo que varios hechos probablemente transformen la triste situación de esta apasionante rama del derecho. Ya desde hace varios años, Costa Rica viene construyendo en este campo, primero formalizando cátedras de derecho informático y de informática jurídica en diversas universidades del país, y, también, creando maestrías especializadas, las cuales muy pronto verán la luz pública, creando espacio para que los juristas expertos en estos campos puedan desarrollar una labor docente e investigativa más focalizada. Por otra parte, la propia dinámica social, ha venido impulsando la preparación de los juristas en los campos tecnológicos, al punto que ya no son más objeto de sorpresa mencionar como necesidades contractuales, la formalización de relaciones jurídicas de *outsourcing* o de tele proceso o de teletrabajo. Cada vez más, y de una manera más intensa, estaremos presenciando el advenimiento de nuevos y más complejos usos de las tecnologías, lo que sin duda nos motivará, como a don Christian, a reflexionar sobre los chances presentes y futuros para nuestro ordenamiento jurídico de llevar el pulso de estos cambios incesantes.

Solo una pequeña muestra de este creciente interés en temas generales del derecho informático lo revelan, por cierto, las recientes investigaciones periodísticas sobre el uso indiscriminado de datos personales por parte de empresas que se dedican a lucrar con el tratamiento electrónico de estos datos, pero también las posibilidades que se recortan en el horizonte de una posible autopista de información, hecha a la medida de las posibilidades de progreso de nuestro país. Esta autopista, imaginada por el señor ex Ministro de Ciencia y Tecnología, el Dr. Guy de Teramond, probablemente nos ubique en el concierto de aquellas naciones con capacidad e infraestructura informática suficiente para garantizarle a sus ciudadanos progreso y desarrollo en una época marcada profundamente por el signo tecnológico. Otros temas ocuparán a la discusión nacional también, como lo serán los contratos para defender software de Internet diseñado e implementado en Costa Rica, firmas de *e-commerce* con sede nacional, y, por supuesto, para potenciar las posibilidades de aumentar el mercado y el perfil de las jóvenes industrias de software de alta tecnología que ya se produce en el mercado nacional.

Un observador desapercibido de todos estos fenómenos probablemente reconozca en todo esto los signos de una época inevitable, donde ya se otea no solo la necesidad de una actividad urgente del legislador, sino también un cambio de muchas de nuestras concepciones civilistas decimonónicas, que pueden

retrasar, gravemente, el avance de esta joven ciencia jurídica, pero también el progreso del país.

Es por todo esto que la obra de don Christian llega en buen momento, no sólo para ponernos a pensar en los procesos que se están gestando, sino también para invitarnos a pensar sobre lo que es dable hacer para que todos los procesos tecnológicos se detengan por un estado de la cuestión insalvable.

Los recientes escándalos por la aplicación de la legislación antimonopolio en los Estados Unidos a la poderosa firma Microsoft, han vuelto a poner en la mesa de debate los problemas involucrados en un negocio de incalculables beneficios económicos. El dominio del mercado ya no cuenta en términos de capital invertido sino en influencia y capacidad para ubicar nuevos productos informáticos y aplicaciones al alcance del gran público, y este secreto puede generar dividendos que antes eran inimaginables. Al respecto recordamos con nostalgia, por ejemplo, las palabras de un ex Presidente de la Firma IBM que decía que el mercado para computadoras del hogar se reducía a unas dos o tres a los sumo. Hoy día el parque informático instalado en el mundo ostenta varios miles de millones de computadoras personales, tanto tipo laptop, como computadoras de escritorio. En todas ellas corre software de todo tipo, original, copiado, de shareware y de otros tipos, y ya ello nos permite imaginarnos la enorme complejidad de los problemas jurídicos que enfrentamos.

El Master Hess analiza, con justa razón, el estado actual del debate sobre la protección del software, una discusión que lejos de amainar se ha vuelto a recrudecer frente a las necesidades internacionales por lograr una efectiva protección de la obra intelectual puesta en la concepción, diseño y producción de un determinado programa de cómputo. El obstáculo para alcanzar la tutela jurídica, como siempre, consiste en la idea de concebir al software como un mero conjunto de instrucciones, a lo sumo como un esquema o método para calcular un grupo de operaciones traducibles al lenguaje en que opera la máquina. Con todo y todo, los esfuerzos estadounidenses y europeos insisten en conducirse a la consideración de los programas de cómputo como patentables y concederles dicho carácter, por lo menos así lo deja entrever el acuerdo del Consejo de Europa de 20 de febrero de 2002, que augura candentes discusiones no sólo en dicho entorno sino también en el mundo entero.

Pasa revista el autor a todos los esfuerzos que se han hecho por encontrar un modo idóneo de tutela, pero todos parecen chocar ante la naturaleza *sui generis* y si se quiere resbaladiza de nuestro objeto de estudio, que cada vez que creemos haberla asido con fuerza se nos escapa en la dirección menos pensada. Para muestra un botón, la discusión que se ha planteado acerca de la protección a partir de la apariencia y “sensación” que deja el software en el usuario para



fundamentar de allí una tutela jurídica. El camino sigue abierto y las observaciones del autor nos dan el material necesario para formalizar nuestro propio criterio y hacer los aportes indispensables.

El otro tema que particularmente me emocionó del texto es el que se hace sobre la protección de las bases de datos, las cuales tienen hoy una importancia capital en la sociedad de la información. Hoy los procesos propios de recopilación, organización, almacenamiento y tratamiento electrónico de datos se diligencian por medios informáticos y pasarán a formar parte de bancos de datos cada vez más grandes, sofisticados, pero al mismo tiempo cada vez más accesibles al bolsillo del gran público y de una velocidad impresionante para comparar los datos que en ellos se conservan. Casi todos los temas imaginables pueden ser reducidos a una aplicación o desarrollo de bancos de datos, y de hecho nuestra propia experiencia jurídica ya depende de diversas bases de datos como la de SINALEVI y el programa del Sistema de Información Jurídica de Costa Rica.

Con todos estos ejemplos podemos imaginar cómo va a surgir la pregunta inevitable de cómo proteger estos inmensos acervos de datos, si como sistemas en sí, como aplicaciones de software o como meras compilaciones organizadas y trabajadas con un objetivo o fin predeterminado. El sentido de colecciones no se menciona claramente pero sigue teniendo presencia en documentos internacionales sobre la materia como el Convenio de Berna para la protección de obras literarias y artísticas. Incluso en el marco regional, diversas disposiciones prevén para los autores de bases de datos tutelas especiales contra la extracción o la reutilización de información. A nivel nacional se ha preferido protegerlas como compilaciones, las cuales son susceptibles de registro.

Hoy en día no habría discusión de derecho informático si no pudiéramos plantearnos la discusión legítima sobre el derecho a conservar la información o datos de determinados dispositivos de grabación mediante herramientas tecnológicas, tema especialmente candente en el campo de la música digital, pero también de otros productos informativos.

## Notas preliminares a esta edición electrónica

La edición electrónica de esta obra, al igual que las dos ediciones impresas que le antecedieron, está dirigida fundamentalmente a profesionales en derecho y estudiantes que desean complementar su formación en el campo específico del derecho informático. Sin demasiadas pretensiones, se procura ilustrar la problemática actual del tratamiento jurídico del software en cuatro campos específicos:

- **Naturaleza.** Luego de definir algunos conceptos terminológicos de importancia, en este primer capítulo se intenta caracterizar jurídicamente al software, en tanto bien autónomo y susceptible de figurar como objeto de transacciones civiles y comerciales, proponiendo algunos esquemas de clasificación útiles.
- **Tutela.** En este aparte repasamos los principales mecanismos legales propuestos para brindar protección jurídica al software en tanto creación intelectual. A pesar de que es indudable que tal amparo resulta necesario (de lo contrario, poco o ningún incentivo habría para producir software), veremos que la doctrina y los ordenamientos están lejos de coincidir en cómo y en qué medida ofrecer esta seguridad.
- **Contratos.** El software, como ya se dijo, es un bien jurídico idóneo para la celebración de diferentes convenios jurídicos en torno a él. Sin embargo, por razones de extensión, deliberadamente limitamos nuestro examen a las principales figuras observadas en la práctica actual: el licenciamiento, la cesión en el marco del desarrollo a la medida y el arrendamiento en línea de software (ASP).
- **Responsabilidad.** El uso (y abuso) del software tiene consecuencias sobre personas y entidades reales. No opera en el vacío. Esos efectos pueden ser positivos o negativos. En este último caso, las circunstancias concretas podrían hacer nacer una responsabilidad y la consiguiente obligación de indemnizar los potenciales daños y perjuicios irrogados.

En todos los casos, se procura ir más allá de la discusión puramente doctrinaria para ilustrar situaciones del mundo real relacionadas con cada tema.

Es importante advertir además al (la) lector(a) que, inevitablemente, encontrará a través de la obra frecuentes referencias a la doctrina y casos judiciales estadounidenses. Digo que esto es inevitable porque, quiérase o no, la industria del software –y su consecuente problemática jurídica– proviene mayoritariamente de ese país.

Tanto en el texto de los diversos capítulos como en el anexo que se incluye al final, aparecen diversas referencias a sitios y páginas *web* en Internet, citados no solo como fuente bibliográfica sino también para efectos de obtener mayor información sobre los temas tratados. En este sentido, es menester hacer la advertencia adicional de que –debido al carácter dinámico de la red– es muy posible que los diversos contenidos y direcciones cambien con el tiempo. Si algún(a) lector(a) desea reportar un cambio o sugerir sitios o páginas de interés sobre la materia, agradeceré mucho hacérmelo saber por medio de mi sitio *web* en [www.hess-cr.com](http://www.hess-cr.com), desde el cual también ofrezco algunos materiales adicionales y referencias de interés.

Finalmente, es de rigor destacar que las diversas marcas comerciales y razones sociales mencionadas a través de esta obra, son propiedad de sus respectivos titulares.

*El autor*  
*San José, 4 de mayo del 2007.*

# Capítulo I - Naturaleza jurídica del software

## Aspectos terminológicos

Por oposición al término **hardware**, que hace alusión a los componentes *físicos* (corpóreos, tangibles) de un sistema computacional, la palabra **software** designa a todos sus componentes *lógicos*, que incluyen tanto a las aplicaciones como a los datos sobre los cuales ellas operan.<sup>1</sup>

Por “aplicación” (o “programa”), entendemos

*“[Toda] secuencia de instrucciones o indicaciones destinadas a ser utilizadas directa o indirectamente en un sistema informático, para realizar una función o tarea o para obtener un resultado determinado, cualquiera que fuere su forma de expresión o fijación.”<sup>2</sup>*

En un sistema informático, el software regula y gobierna todas las funciones de **procesamiento de datos** de las que deriva, en esencia, el provecho que podemos obtener de cualquier computadora.

En líneas generales, esta tarea típicamente involucra, en primer lugar, el ingreso al sistema de una serie de datos que funcionan como la materia prima del proceso (*entrada*). Ese insumo es entonces sometido a un tratamiento determinado por las instrucciones de la aplicación respectiva (*procesamiento*), que –finalmente– genera la *salida* de los resultados obtenidos.

Piénsese, por ejemplo, en la preparación de una planilla de pagos salariales a los empleados de una empresa. Sobre la base del “insumo” que representan los datos correspondientes a cada trabajador, el número de horas laboradas, el

---

<sup>1</sup> Ni “hardware” ni “software” son, desde luego, palabras castizas del idioma español. De hecho, en la jerga informática, “hardware” ni siquiera se emplea en su correcta acepción inglesa (que alude a los artículos de ferretería). Si los recogemos y empleamos a través de toda esta obra, es simplemente reconociendo el hecho de que ambos términos han llegado a penetrar nuestro lenguaje coloquial, al punto de que resulta más sencillo hablar de “hardware” y “software” que de “equipamiento” y “soporte lógico”, por ejemplo.

<sup>2</sup> CARRASCOSA LÓPEZ, Valentín y otros. *El comercio informático*. En “Ponencias del VI Congreso Iberoamericano de Derecho e Informática de la FIADI”, publicación del comité organizador. Montevideo, 1998.

estipendio por unidad de tiempo, las deducciones e impuestos aplicables, etcétera (entrada); el sistema efectuará un cálculo del monto pagadero a cada uno (procesamiento) y producirá un reporte, o quizás incluso imprimirá los giros correspondientes (salida).

Decíamos que el software regula y gobierna las funciones de procesamiento de datos, precisamente porque dependerá de la aplicación concreta con la que estemos trabajando, la manera en que los datos de entrada ingresen al sistema, la forma en que sean manipulados y el modo en que los resultados sean presentados al usuario.

Ninguna computadora, por grande y poderosa que sea, escapa al hecho de que –en última instancia– ella es tan sólo un conjunto de dispositivos carente de toda capacidad de análisis o raciocinio propio. De un modo u otro, todas las tareas que la máquina sea capaz de realizar, desde las más sencillas hasta las más complejas, dependen de que se les haya suministrado previamente las instrucciones precisas para efectuarlas. A esto se llama **programar** la computadora. Por tanto, qué tan bien (o mal) ejecute la computadora la tarea encomendada dependerá de lo bien (o mal) que haya sido programada.<sup>3</sup>

Esta realidad –la de que toda computadora está desprovista de razonamiento propio y que precisa ser programada antes de que pueda realizar cualquier tarea útil– es, desde luego, una contrariedad, por lo que probablemente querríamos que, en desagravio, la labor de programación fuese lo más sencilla posible.<sup>4</sup>

Desgraciadamente, nada más lejos de la realidad. Para comenzar, las computadoras no comprenden nuestro idioma. El lenguaje humano es infinitamente complejo, pletórico de abstracciones, sobreentendidos y factores culturales, todos los cuales están más allá de las limitadas posibilidades de un equipo informático. El **lenguaje de máquina** –que es el único que la computadora es capaz de manejar– es tan limitado que, de hecho, están compuesto únicamente por dos elementos: unos y ceros.<sup>5</sup> En consecuencia, para que un computador sea

<sup>3</sup> Ninguna computadora se equivoca. Equivocarse es algo que sólo puede hacerlo quien tiene la posibilidad de elegir entre distintas opciones y, por las razones que fuere, escoge erróneamente. Una máquina carece de tal albedrío. Sólo ejecuta mecánicamente lo que se le pide ejecutar. Excluyendo obviamente los supuestos de fallo mecánico, si el computador se equivoca es porque su programación era equivocada y esto, en definitiva, será imputable a alguna persona o personas. En el capítulo concerniente a la responsabilidad derivada del software ahondaremos en esta temática.

<sup>4</sup> Tal como se ve en algunas cintas de ciencia ficción, lo ideal sería simplemente dirigirnos de viva voz a la máquina para indicarle lo que deseamos que haga.

<sup>5</sup> Que, a su vez, meramente representan los estados “encendido” y “apagado” que son los únicos que pueden asumir los componentes electrónicos internos del hardware.

capaz de interpretar y ejecutar una instrucción cualquiera, ésta deberá expresarse primero en términos de unos y ceros (algo así como “10110011”). Y puesto que un programa informático no es más que una secuencia discreta de instrucciones, todo él deberá presentarse a la máquina como unos y ceros antes de que ésta sea capaz de ejecutarlo. A esto último llamamos el **código objeto**, o “código binario”.

Se comprende fácilmente que esta exigencia es harto problemática para nosotros, los seres humanos, cuyo cerebro –resultado de millones de años de evolución– es infinitamente más sofisticado; perfectamente capaz de manejar con toda soltura un vocabulario muy complejo, incluso desde la infancia. Si, para programar una computadora, tuviésemos siempre que pensar y expresarnos en términos de unos y ceros, no cabe duda de que la informática difícilmente habría podido surgir como disciplina provechosa, hasta sus actuales estadios de desarrollo.

Para resolver el problema, se ha creado diversos **lenguajes de programación**. Algunos de ellos son “de bajo nivel” (lo cual significa que se aproximan más al lenguaje de máquina que al natural), como el llamado lenguaje ensamblador (*assembly language*). Otros son “de alto nivel”, muy similares al idioma humano, como el BASIC.<sup>6</sup> Y otros son “de nivel intermedio”, como los lenguajes C, C++ o Java. Pero sea como fuere, el propósito de todos estos lenguajes es permitir a los programadores escribir sus aplicaciones de una manera más sencilla e inteligible. Cuando el software se expresa de esta manera, se le denomina **código fuente**.

Para efectos ilustrativos, considérese este segmento de un programa escrito en el lenguaje Visual BASIC de la empresa Microsoft:

```
HoraActual = Now
If HoraActual < 12 Then
Saludo = "Buenos días"
ElseIf HoraActual < 18 Then
Saludo = "Buenas tardes"
Else
Saludo = "Buenas noches"
End If
```

Incluso sin conocimientos de programación, una persona que domine el inglés probablemente será capaz de seguir con facilidad la lógica de estas pocas líneas (primero se asigna la hora actual a una variable llamada *HoraActual* y, dependiendo del resultado, se asigna a otra llamada *Saludo* un mensaje apropiado). Pero incluso ese breve y sencillo segmento sería totalmente

<sup>6</sup> Acrónimo de *Beginners All-purpose Symbolic Instruction Code*.

incomprensible para una computadora a menos que sea traducido al único lenguaje que ella “entiende”: el lenguaje de máquina. Ese proceso se denomina **compilación** y es efectuado por aplicaciones especiales, apropiadamente conocidas como **compiladores**.<sup>7</sup>

## Caracterización jurídica del software

Hasta la década de 1960 y aun durante buena parte de la de 1970, la mayoría del software era producido o distribuido por los propios fabricantes del hardware y era considerado como un accesorio de aquél. Esto era lógico, ya que la ausencia de estandarización en las plataformas de hardware determinaba que la producción de aplicaciones tuviera que quedar enteramente en las manos de quienes fabricaban los equipos, o sus licenciarios directos. De este modo, por ejemplo, la empresa IBM construía sus computadoras y además programaba la mayoría de las aplicaciones necesarias.

Sin embargo, comenzando a mediados de la década de 1960 (cuando a raíz de acciones tomadas por el Departamento de Justicia de los EE.UU. se forzó a la IBM a separar la distribución de hardware y software, por considerar lo contrario como anticompetitivo<sup>8</sup>) y especialmente a partir de la revolución de las computadoras personales (PC) de los años ochenta, podemos postular que el software ha adquirido una entidad y autonomía propias. Innumerables fabricantes de software actualmente producen aplicaciones orientadas a las diversas plataformas informáticas, con gran éxito por cierto.<sup>9</sup> En efecto, parte del secreto del éxito de la PC original de IBM fue el hecho de que esa empresa decidiera adoptar una arquitectura abierta, permitiendo que terceros fabricantes de hardware y productores de software pudiesen satisfacer la creciente demanda de componentes de expansión y aplicaciones. Esto eventualmente impuso a la PC de IBM como modelo dominante en el mercado, a pesar de que el concepto mismo de una computadora personal no es original de ese fabricante.<sup>10</sup>

Decimos, entonces, que *el software actualmente constituye un bien jurídico en sí mismo* y, por ende, es objeto de consumo y de comercio.<sup>11</sup>

<sup>7</sup> El hecho de que, técnicamente, no todos los lenguajes de programación sean compilados sino que algunos sean más bien “interpretados” (en los que la traducción a lenguaje de máquina ocurre durante la ejecución y no antes), es irrelevante para nuestros efectos.

<sup>8</sup> GLASS, Robert L., “A Look at the Economics of Open Source”. En *Communications of the ACM*, volumen 47, número 2, febrero del 2004, página 26.

<sup>9</sup> Hoy por hoy, algunas de las compañías más grandes y económicamente poderosas del mundo son, justamente, fabricantes de software, como Microsoft y Oracle.

<sup>10</sup> Podríamos decir que ese crédito es compartido por otras empresas como Texas Instruments (creadora del microprocesador), Altair y Apple.

En el plano jurídico y en lo que a su naturaleza concierne, podemos caracterizar entonces al software así:

- Es un bien *mueble*, sencillamente porque no califica como inmueble, usando el método de clasificación por exclusión del artículo 256 del Código Civil de Costa Rica.
- Es *inconsumible*, ya que no se destruye con el uso.
- Es *fungible*, porque cada copia es intercambiable y exactamente equivalente a las demás.
- Es un bien *indivisible*, puesto que no se puede ejecutar un programa parcialmente, ni distribuirlo por partes ejecutables separadamente.<sup>12</sup>
- Posee carácter *patrimonial*, porque es pecuniariamente estimable y susceptible de figurar como objeto de transacciones civiles y comerciales.

Alguien podría considerarlo como un bien corporal (artículo 258 de nuestro Código Civil), porque –en última instancia– todo software se fija y representa físicamente.<sup>13</sup> Sin embargo, esta perspectiva estática no traduciría correctamente lo que es el software, porque en realidad éste no adquiere una entidad real, tornándose aprovechable, sino hasta que es ejecutado y da paso a una interacción determinada con el usuario. Dicha perspectiva dinámica nos revela al software más bien como un bien *inmaterial*, intangible y volátil, que es el resultado de la ejecución por parte del computador de la secuencia de instrucciones que lo componen. Esta disyuntiva en realidad no es nueva, sino que es una característica de las obras intelectuales en general, en las que siempre cabe distinguir al bien incorpóreo o ideal (*corpus mysticum*), del soporte físico sobre el que esté fijado (*corpus mechanicum*).<sup>14</sup> Así pues, el derecho de propiedad que se ejerza sobre el soporte carece de nexos jurídico inmediato con la cuestión de la titularidad del

<sup>11</sup> Aunque algunas clases de software, especialmente el de sistema (sistema operativo de *mainframes*, controladores o *device drivers*, etcétera), continúa siendo producido en buena parte por los propios fabricantes de hardware.

<sup>12</sup> El hecho de que sea posible actualizar o modificar algunos elementos separadamente (como las librerías DLL de las aplicaciones Windows) no contradice el hecho de que todos ellos deban ser cargados y ejecutados unitariamente por el computador.

<sup>13</sup> Ya sea como impulsos eléctricos en la memoria de un computador, partículas magnetizadas de un disco duro, segmentos quemados por rayo láser en un disco compacto (CD), líneas de código fuente en el papel, etcétera.

<sup>14</sup> Una importante excepción, en el caso del software, la constituye el llamado *firmware*, al que nos referiremos un poco más abajo.



software en sí. Si adquiero e instalo en mi computadora una aplicación comercial que venga distribuida en uno o más disquetes, nada me impide borrar éstos posteriormente y reutilizarlos para almacenar archivos personales, o incluso obsequiar los disquetes en blanco a otra persona. El derecho de autor del titular del software no se verá afectado en modo alguno.<sup>15</sup>

## Clasificaciones del software

En cualquier disciplina científica, la clasificación tiene siempre un propósito eminentemente didáctico. Al subdividir un todo complejo en compartimentos menores, empleando un bien escogido parámetro de clasificación, la asimilación y aprendizaje del tema se facilita y las relaciones entre las partes y el todo se tornan aparentes. En lo que al software se refiere, proponemos seguidamente dos posibles mecanismos de clasificación, el primero de índole técnico y el segundo jurídico. Ambos puntos de vista –que no se excluyen entre sí– son provechosos para nuestros efectos.

### *Desde la perspectiva técnica*

A grandes rasgos, el software puede ser técnicamente agrupado en dos categorías generales: el *software de sistema* y el *software de aplicación*.

- **Software de sistema** alude a aquellos programas y datos requeridos para el funcionamiento básico del equipo informático. En este sentido, proveen una base para la posterior ejecución de las aplicaciones, pero sin formar parte de ninguna de ellas. En esencia, posibilitan la correcta operación del hardware, al establecer y regular la comunicación entre la unidad central de proceso (CPU) y todos los demás componentes físicos. El más importante de estos programas es el llamado *sistema operativo*, que la computadora debe cargar e iniciar desde el momento del encendido. Existen diversos sistemas operativos, cada uno diseñado específicamente para la clase de hardware sobre la que deban operar. Esto incluye tanto los productos comerciales (como el Windows de la

---

<sup>15</sup> Ahora bien, JIJENA LEIVA estima que caracterizaciones jurídicas como la que hemos intentado aquí son enteramente inapropiadas en tratándose del software: "... los programas computacionales no son una 'cosa corporal mueble' o un 'documento' de aquellos contra los cuales, cuando son copiados ilegalmente, pueden cometerse los delitos patrimoniales de hurto, robo o apropiación indebida. Ocurre que se trata de cosas intangibles e inmateriales no apropiables o no aprehensibles físicamente... en términos legales, las que además, cuando son copiadas o 'pirateadas' no le son privadas en forma permanente a la víctima del delito." JIJENA LEIVA, Renato Javier. "La protección legal del software en Chile. Análisis de la ley 17336 y sus modificaciones.", en *Ponencias del VI Congreso Iberoamericano de Derecho e Informática de la FIADI*. Publicación del Comité Organizador. Montevideo, 1998.

empresa Microsoft o el MacOS de Apple) como los no comerciales (como Linux).

- El **software de aplicación**, por su parte, reúne a todos aquellos programas y datos con los que propiamente trabajamos para obtener determinados resultados o desempeño: procesadores de texto, paquetes de contabilidad, administradores de bases de datos, juegos, navegadores de Internet, enciclopedias electrónicas, etcétera. Como se explicó, estas aplicaciones corren encima del sistema operativo, solicitando y obteniendo de él diversos servicios relativos al hardware, por ejemplo, para imprimir documentos o guardar archivos en disco.

### ***Desde la perspectiva jurídica***

En el plano jurídico, proponemos cuatro esquemas de clasificación complementarios del software, de los que podemos extraer diversas consecuencias legales relevantes: a) según el *grado de estandarización del software*; b) según el *grado de vinculación del software al hardware*; c) según la *presencia o ausencia de contraprestación y sus modalidades*; y, d) según la *modalidad de licenciamiento*.<sup>16</sup>

#### *i.- Según el grado de estandarización: software estándar o hecho a la medida*

En atención al grado de estandarización (uniformidad) del software, lo dividimos en *software estándar* o bien *hecho a la medida*.<sup>17</sup>

El **software estándar** incluye a todos los innumerables productos fabricados para distribución masiva. Es “estándar” precisamente porque es igual para todos, sin perjuicio de que –una vez instalado– el usuario pueda personalizar algunas de sus opciones.

A esta categoría se le suele llamar también “software de paquete”,<sup>18</sup> pero esa denominación es actualmente omisa en la medida en que, crecientemente, se aprovecha a la Internet como canal para la distribución de aplicaciones de toda clase. También es erróneo confundir esta categoría con la del “software

<sup>16</sup> En el capítulo relativo a contratos se explicará en qué consiste el licenciamiento de software.

<sup>17</sup> Algunos incluyen aquí la categoría del llamado “software comercial adaptable” o “personalizable”, pero no estimamos que constituya realmente una división propia sino tan sólo una subdivisión del grupo estándar o comercial.

<sup>18</sup> En la jerga angloparlante, *shrink-wrapped* software, haciendo alusión al plástico en que suelen venir envueltas las cajas.

comercial”, ya que existe abundante software estándar de distribución gratuita. Más abajo volveremos sobre esta distinción.

Por su parte, el **software hecho a la medida** es aquel que es producido específicamente contra pedido y para las necesidades concretas de un usuario o usuarios. Se hace aparente, entonces, que la diferencia entre este grupo y el anterior es similar a la que existe entre la ropa que podríamos adquirir en una tienda de departamentos y aquella que se encomienda a un sastre o modista. En el capítulo relativo a los contratos de software examinaremos más de cerca las connotaciones jurídicas de esta figura.

*ii.- Según el grado de vinculación del software al hardware: firmware o software autónomo*

De acuerdo con el grado de vinculación del software al equipo, hablamos de “*firmware*” o bien de *software autónomo*.

El (difícilmente traducible) término ***firmware***<sup>19</sup> esencialmente hace alusión a la clase de software que se almacena en la memoria ROM<sup>20</sup> de un computador. Su grado de fijación al equipo puede ser absoluto, como en el caso de los programas implantados vía circuitos trazados, o relativo, como en el caso de los ROM reprogramables (EPROM). El *firmware*, tal como el llamado “programa monitor” de las microcomputadoras, es responsable de las acciones iniciales que toma un sistema informático desde el instante del encendido, que esencialmente consisten en realizar un auto diagnóstico y luego buscar, cargar e iniciar la ejecución del sistema operativo. El *firmware* viene siempre pre-instalado en el equipo informático y es intrínsecamente software estándar.

El **software autónomo**, por exclusión, es todo aquél que, por no venir integrado en el hardware, es necesario obtener separadamente e instalar de previo a su empleo.

Desde la óptica jurídica, la implicación primaria que tiene esta subdivisión es la de reconocer que el *firmware* es un accesorio del hardware y, por tanto, debe ser tratado como tal.<sup>21</sup> El software autónomo, por el contrario, es un bien jurídico (mueble e ideal) individualizado, cuya suerte entonces no sigue a la del equipo.<sup>22</sup>

<sup>19</sup> Conocido también como microcódigo.

<sup>20</sup> ROM significa *read-only memory* (memoria de lectura exclusiva o no regrabable). Se diferencia de RAM o *random access memory* (memoria de acceso aleatorio), que es lo que normalmente conocemos como la memoria principal de un computador.

<sup>21</sup> Sin embargo, hay autores que sostienen un punto de vista divergente. Por ejemplo, CORREA.

<sup>22</sup> Rigurosamente, ningún software es ciento por ciento independiente del hardware, ya que, de un modo u otro, debe ser creado teniendo en cuenta las prestaciones y limitaciones que

*iii.- Según la presencia o ausencia de contraprestación y sus modalidades: software comercial, freeware y shareware*

Si atendemos a la existencia o inexistencia de una contraprestación debida por el usuario al autor o titular de los respectivos derechos sobre el software, lo dividiremos en comercial, *freeware* y *shareware*.

El **software comercial**, como su nombre lo implica, llega hasta los usuarios por canales establecidos de mercadeo (tiendas, catálogos, descargas en línea, etcétera) y es distribuido normalmente con ánimo de lucro. En otras palabras, es objeto de una transacción mercantil determinada, en virtud de la cual el usuario lo adquiere<sup>23</sup> previo pago del precio establecido.

El término **freeware** también se refiere a un modo de adquisición de software, caracterizado por la ausencia de contraprestación; es decir, por su gratuidad.<sup>24</sup> El usuario es normalmente libre de instalarlo y usarlo en una o más computadoras, así como de distribuir copias a otras personas. Es decir, el titular renuncia a sus derechos de remuneración y reproducción, pero las restantes atribuciones patrimoniales y morales permanecen intactas. La persistencia de éstos derechos implica que el titular del derecho de autor estará en plena capacidad de ejercitar sus facultades legales en el momento en que lo requiera o desee, por ejemplo, para establecer acciones legales que impidan la distribución de versiones modificadas del programa, o para reclamar contra el cobro indebido de sumas por su distribución a terceros.

Es posible también que el empleo gratuito del programa esté condicionado a que el usuario respete ciertas limitaciones o que reúna ciertas características, por ejemplo: que se trate de una persona física que pretenda usarlo para fines personales y no lucrativos; que sea un docente o estudiante; etcétera. En este caso, sus posibilidades de redistribución de copias del software a terceros usualmente estarán a su vez sujetas a que las brinde sólo a otras personas que reúnan el mismo perfil.

¿Por qué querría alguien hacer software gratuito? En realidad, hay varias y muy buenas razones:

---

presente el equipo en el que será ejecutado. En particular, una aplicación solo correrá en la medida en que sea traducible al juego de instrucciones básicas que soporte el procesador central (*instruction set*).

<sup>23</sup> Vía licencia de uso o cesión de derechos patrimoniales, como veremos más abajo.

<sup>24</sup> La gratuidad del *freeware* no debe llevarnos al error de pensar que se trata necesariamente de software de mala calidad. Algunos paquetes de esta categoría muestran un rendimiento igual o mejor que el de sus contrapartes comerciales.

- Algunas empresas fabricantes de software distribuyen gratuitamente versiones limitadas de sus productos comerciales, con el fin de atraer posibles interesados hacia estos últimos.
- Otras lo hacen con el fin de abrir un nicho de mercado que les permita crearse una reputación tecnológica y comercial, y así colocar sus otros productos y servicios.
- Otros prefieren regalar un producto, en la esperanza de obtener dinero mediante la prestación de servicios complementarios (soporte técnico, consultorías, etcétera) o a través de la venta de futuras actualizaciones (por ejemplo, en el caso del software antivirus).

El **shareware** constituye una modalidad especial relativa a la contraprestación debida al autor o distribuidor del software, que se caracteriza por estar sujeta al examen temporal del producto hasta el acaecimiento de una condición preestablecida, en cuyo momento el usuario debe decidir si desea conservarlo o no. Caso afirmativo, deberá satisfacer la contraprestación estipulada en el contrato por el licenciente.<sup>25</sup>

De nuevo, cabe preguntar, ¿por qué querría alguien acudir a esta variante? Como canal para la distribución de software, el **shareware** es particularmente apto para productos que –por su novedad, escasa publicidad, etcétera– el fabricante estima que los usuarios posiblemente no adquirirían sin antes convencerse de sus bondades. En efecto, numerosos productos han llegado a popularizarse por esta vía, desde juegos hasta software de productividad. Un excelente ejemplo lo constituye el muy popular programa de compresión y descompresión de archivos, WinZip.<sup>26</sup>

De los conceptos enunciados arriba, podemos extraer y ahondar en los siguientes elementos de interés con relación al **shareware**:

1. *Examen o prueba del software.* Hasta adónde se permita al usuario probar el producto durante el período de gracia, depende del fabricante. Algunos software tendrán funcionalidades limitadas total o parcialmente; otros no presentarán restricción de uso alguna. En ciertos casos, el software marcará de alguna manera los archivos de datos, imágenes o

<sup>25</sup> HESS ARAYA, Christian. “Apuntes en torno al software *shareware* o *trialware*”, en *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 37, agosto del 2001. Nótese, sin embargo, que al momento de escribir este artículo, se analizaba al **shareware** como una modalidad de licenciamiento cuando, a la fecha, estimo más precisa la clasificación presentada aquí.

<sup>26</sup> Véase <http://www.winzip.com>

impresiones generadas<sup>27</sup> o desplegará alguna otra clase de indicación al respecto.

2. *Condición preestablecida.* Lo más usual es que la condición consista en el vencimiento de un determinado plazo, que la costumbre suele fijar en treinta días. Sin embargo, en la práctica se observan también otras modalidades. Por ejemplo, se puede señalar un número máximo de utilidades del software. Exactamente qué ocurra entonces, en lo que toca al funcionamiento de la aplicación, dependerá también de cada proveedor. Por ejemplo, podría dejar de operar completamente o mostrar mensajes periódicos de recordatorio o advertencia al usuario.<sup>28</sup>
3. *Contraprestación estipulada.* Una vez acaecida la condición, el usuario estará obligado a desistir del empleo del producto y a desinstalarlo de su equipo, a menos que satisfaga una determinada contraprestación, que puede ser de dar o hacer. Lo más usual es que consista en el pago de un canon al fabricante. De nuevo, no obstante, en la práctica se encuentran con frecuencia prestaciones distintas, tales como efectuar una donación a una determinada institución de caridad (en cuyo caso se suele identificar al software bajo el denominador *charityware*), enviar una tarjeta postal al creador (*postcardware*), etcétera. En estos casos, el registro suele ser correspondido con la asignación al usuario de alguna contraseña o código que le permitirá acreditar que es titular legítimo de la aplicación, así como activar las funcionalidades previamente limitadas del paquete, si las hubiere.

Sean cuáles fueren las innumerables combinaciones que pueden darse de los tres elementos anteriores, es importante recalcar que –como ocurre siempre en el licenciamiento de software– en el *shareware* no se da, en ningún momento, transferencia alguna de los derechos patrimoniales de la obra al licenciatario. Esto específicamente implica que, una vez acaecida la condición, el uso continuado del programa sin satisfacer los términos de la contraprestación exigida hará incurrir al infractor en las responsabilidades civiles y penales propias de la violación de derechos de autor (esto es, la “piratería de software”).

La redistribución a terceros del software en cuestión está sujeta a que se haga de una copia no registrada del paquete (es decir, que el paquete se distribuya en la forma originalmente preparada por el autor) y que no se facilite a dichos

---

<sup>27</sup> Por ejemplo, mediante una leyenda de tipo “Esta imagen ha sido generada mediante una copia no registrada de ...” o similar.

<sup>28</sup> Opción que en el argot anglosajón se conoce, peyorativamente, como *nagware*.

terceros la contraseña o clave de registro del software. Lo contrario equivaldría también a una redistribución ilícita.

*iv.- Según la modalidad de licenciamiento: software propietario, libre y de dominio público*

La licencia de uso de una aplicación de software puede otorgar al usuario derechos más o menos limitados. Desde este punto de vista, podemos caracterizarlo como software propietario, software libre o software de dominio público.

En el **software propietario**, el autor o el titular de los derechos respectivos otorga una gama limitada de derechos a los usuarios del producto, los cuales aparecerán estipulados en el contrato de licencia.<sup>29</sup> La mayoría del software comercial se distribuye bajo esta modalidad, lo cual es fácilmente comprensible ya que por medio de la restricción de derechos se pretende asegurar la máxima rentabilidad en la explotación lucrativa del software en cuestión.

El concepto de **software libre** nace en el contexto de todo un movimiento filosófico –si se quiere– que surgió en el mundo de la informática en la década de 1970, aunque el término en sí no se popularizó sino hasta que Richard Stallman, investigador del MIT, fundó su Free Software Foundation en 1984.<sup>30</sup> Se le conoce también como *open source* (código abierto) y parte de la premisa de principio de que el software –o, más exactamente, la base de conocimiento que está contenido en él– debería ser libre y accesible para todos. Las aplicaciones *open source* se caracterizan porque en adición a poner los programas a disposición de todos, también se distribuye libremente el código fuente, con la idea de que otros programadores puedan estudiarlo e incluso modificarlo para fines de personalización o bien para introducirle mejoras, beneficiando de vuelta a los demás usuarios. Los más conocidos ejemplos de esta categoría son el sistema operativo Linux y el paquete de software de oficina OpenOffice.org<sup>31</sup>.

“Software libre”, contrario a lo que comúnmente se cree, no equivale a “software gratuito”. Aun cuando es frecuente encontrar ambas características en un mismo producto, en realidad nada impide que un paquete de software libre sea distribuido lucrativamente.<sup>32</sup> Lo “libre” del software reside en los derechos que

<sup>29</sup> Véase el capítulo de esta obra dedicado al contrato de licenciamiento de software.

<sup>30</sup> DEMPSEY, Bert J. y otros. “Who is an open source software developer?”, en *Communications of the ACM*, volumen 45, número 2, febrero del 2002, página 67.

<sup>31</sup> Véase <http://www.openoffice.org>

<sup>32</sup> En este sentido coincide RÍOS, Wilson Rafael. “Aspectos Legales del Software Libre (Open Source)”, en *Revista Electrónica de Derecho Informático*, N° 117 - Junio 2003, disponible en <http://www.alfa-redi.org/revista/data/61-1.asp>

respecto de él disfrutan sus usuarios, no en el precio que tenga.<sup>33</sup> Dichos derechos incluyen:

- La libertad de ejecutar la aplicación, para cualquier propósito.
- La libertad de estudiar su funcionamiento y adaptarlo a las necesidades particulares.
- La libertad de redistribuir copias a otras personas. Y,
- La libertad de mejorar el programa y ofrecer las mejoras al público, para beneficio de la comunidad.<sup>34</sup>

Es importante recordar además que los respectivos derechos de propiedad intelectual –particularmente los morales, que de toda suerte son, por definición, irrenunciables– subsisten siempre.

La gratuidad que con frecuencia tienen estos productos puede conducir también al error de creer que el software libre carece de licenciamiento. Lo cierto es que éste siempre existe, aunque adopte modalidades tan flexibles como la GNU/GPL (que es la más conocida, aunque no la única), que otorga a los usuarios el amplio conjunto de derechos y facilidades que precisamente le confieren a esta categoría de software el apelativo de “libre”. De hecho, del concepto de software libre nace el de “copyleft”,<sup>35</sup> que –por así decirlo– “restringe la posibilidad de restringir” los alcances de la licencia de software libre. Es decir, como lo explica RÍOS,

*“Así como el copyright le permite a su titular imponer condiciones, restricciones y limitantes; el copyleft, le impide a quien realice mejoras, actualizaciones, estudios o cualquier uso sobre un software libre, que al distribuirlas o redistribuirlas, imponga restricción alguna que afecte la naturaleza de libre que tenía el software al inicio el proceso y por lo tanto se conservarán y respetarán las mismas libertades que le fueron otorgadas.*

<sup>33</sup> La confusión probablemente provenga de la expresión “free software”, ya que, en inglés, la palabra “free” puede significar tanto “libre” como “gratuito”, dependiendo del contexto. Precisamente por eso es que en ese idioma se prefiere hoy la denominación “open source”.

<sup>34</sup> Para mayor detalle, puede consultarse en Internet el documento “La definición de software libre”, disponible en la dirección <http://www.gnu.org/philosophy/free-sw.es.html>. También RÍOS, *op. cit.*

<sup>35</sup> El término es intraducible al castellano, ya que consiste en un juego de palabras en inglés, a partir de la expresión “copyright”.



*El efecto práctico e inmediato que surte el efecto viral del copyleft, es que este se extiende y ‘contagia o contamina’ a todo lo que tenga contacto con él. Es decir que cuando tomamos y combinamos segmentos de herramientas de software libre con software privado o propietario, el resultado así logrado, debe ser software libre e indefectiblemente estará cobijada por el copyleft.”<sup>36</sup>*

Tampoco es lo mismo hablar de “software libre” que de “estándares abiertos”. Lo primero no necesariamente conlleva a lo segundo. En efecto, baste un solo ejemplo para ilustrar la diferencia: el procesador de texto Microsoft Word, en su versión 2000 y siguientes, emplea el estándar XML para crear documentos que permiten un intercambio de información ágil por medio de la Internet. Pero mientras que XML es un formato estándar y abierto, oficialmente sancionado por la W3C,<sup>37</sup> el paquete Microsoft Word no es un software libre sino propietario. Similarmente, en lo que respecta al código fuente, se suele tener la noción de que para poder garantizar las libertades en el uso del software libre, es necesario que éste se encuentre en un formato abierto. En realidad, el código fuente de un software libre perfectamente puede haber sido generado mediante herramientas propietarias, como, por ejemplo, el Visual C++ o el Visual Basic de la citada empresa Microsoft. La definición de software libre –al menos en el marco de las licencias GPL– no contiene ninguna restricción con respecto al formato en sí del código fuente. Lo importante es que, sea cual fuere ese formato, el código esté disponible para quienes tengan interés en acceder a él.

A propósito del software libre, un tema de gran actualidad es el de la conveniencia o inconveniencia de adoptar su uso obligatoriamente en el Sector Público, como alternativa para enfrentar el alto costo del licenciamiento de sistemas y aplicaciones informáticas. Esta propuesta forma parte de una tendencia mundial que surge en los últimos años como reacción a los altos precios y a la mediocre calidad de gran parte del software propietario disponible. Diversos gobiernos nacionales y locales ya han promovido iniciativas para la adopción del software libre en las dependencias estatales.

En mi opinión, es importante tener claro que la opción del software libre, igual que la del software comercial, posee tanto ventajas como desventajas que toda organización debe valorar antes de actuar. En cuanto a lo positivo, su virtud más obvia es la del costo, ya que –como se explicó– las licencias de software libre frecuentemente son gratuitas, permitiendo un importante ahorro con relación al software comercial. En adición, usualmente es el fruto del trabajo colaborativo,

---

<sup>36</sup> RÍOS, *op. cit.*

<sup>37</sup> “World Wide Web Consortium” ([www.w3c.org](http://www.w3c.org)), organización de estándares en materias relativas al acceso y difusión de información en la red.

desinteresado y transparente de grandes cantidades de programadores, lo cual tiende a dotarlo de una gran calidad y estabilidad. La accesibilidad del código fuente permite desarrollar soluciones altamente personalizadas y adecuadas a las necesidades de cada persona u organización. En lo que a desventajas se refiere, se debe considerar que la economía que se obtiene por la gratuidad de las licencias puede verse contrarrestada por los costos de capacitación y migración. Además, las aplicaciones de software libre no siempre cuentan con el respaldo de personas o entidades que puedan hacerse responsables de la corrección de defectos, la distribución de actualizaciones o de las consecuencias de los fallos que puedan surgir. Finalmente, pueden aparecer problemas a la hora de compartir información con otras personas o entidades, en caso de que la aplicación utilice un formato propietario para el almacenamiento de los datos o documentos.

En definitiva, creo que es erróneo poner la discusión entre software comercial y libre como una cuestión de todo o nada. Inclinarlos por uno, por otro o por una mezcla de ambos debe ser algo que se examine con cuidado y a la luz de las consideraciones relevantes en cada caso. De toda suerte, un problema más apremiante –al menos en mi criterio– es el de asegurar que la información pública que generan las dependencias estatales sea universalmente accesible (es decir, que se almacene y distribuya en un formato abierto), con independencia de cuál haya sido el software utilizado en su producción.

– 0 –

En el **software de dominio público**, el titular del derecho de autor voluntariamente renuncia a cualquier clase de derecho patrimonial sobre la aplicación y, por ende, el licenciamiento conferido al usuario es irrestricto. Desde luego, los derechos morales, a que nos referiremos en el capítulo siguiente, permanecerán necesariamente intactos, en virtud de su carácter irrenunciable.

A cambio, el usuario probablemente carecerá de toda opción de soporte técnico y estará vedado de exigir responsabilidad de ninguna clase al fabricante ante la eventualidad de que la aplicación no funcione como se esperaba. Todo esto se suele hacer ver en la licencia o documentación anexa del producto.

## Capítulo II - La tutela jurídica del software

### Planteamiento del problema

La cuestión de cómo proteger jurídicamente al software, en tanto creación intelectual que indudablemente es, se ha ido convirtiendo en uno de los temas más candentes y de mayor desarrollo del derecho informático actual.<sup>1</sup>

En efecto,

*“La transformación del software en un objeto separado de comercialización, y su vulnerabilidad –dada la facilidad de copiarlo a bajo costo– son las dos causas más importantes que han estimulado el debate sobre la protección jurídica.”<sup>2</sup>*

Sin duda, esto se debe en gran parte a la relevancia de los intereses económicos en juego. Ha llegado incluso a convertirse en tema de diferendos internacionales, al punto –por ejemplo– de que países como los Estados Unidos han dado a condicionar algunos de sus programas comerciales y de asistencia financiera a la promulgación y aplicación de legislaciones nacionales de tutela del software (entre otras producciones intelectuales) y sanción de la “piratería”.<sup>3</sup>

Sin embargo, como veremos, la concepción que llamaríamos “clásica” –en el sentido de que el régimen legal aplicable a estos efectos es el del derecho

---

<sup>1</sup> TÉLLEZ sitúa el inicio de la discusión jurídica en torno al problema en una reunión celebrada por la Asociación Internacional para la Protección de la Propiedad Intelectual, en la ciudad de San Francisco, en 1975. Ver TÉLLEZ VALDÉS, Julio. “La patentabilidad del software. El debate que continúa.”, en *Ponencias del IX Congreso Iberoamericano de Derecho e Informática de la FIADI*. Publicación en CD-ROM del Comité Organizador. San José, abril del 2002.

<sup>2</sup> CORREA, Carlos M. y otros. *Derecho informático*. Ediciones Depalma, Buenos Aires, primera reimpresión, 1994.

<sup>3</sup> Por “piratería” se conoce el “acto de reproducción con fines de lucro, ya sea venta, alquiler, importación, traspaso o cualquier otro modo de comercialización, de uno o varios ejemplares de fonogramas, videogramas o programas de cómputo, o de otro soporte material, reproducido por cualquier medio conocido o por conocerse, sin autorización expresa del titular del derecho de autor.” ZÚÑIGA CHAVES, Nuria. *Manual Práctico sobre Derechos de Autor*. Editado por la Asociación de Compositores y Autores Musicales de Costa Rica (ACAM), San José, 1998.

autoral– ha ido enfrentando crecientes e insoslayables críticas doctrinarias en cuanto a su justeza.

Pero, en primer lugar, ¿por qué se debe proteger al software? Las razones son varias y todas muy buenas:

- Porque su creación representa un gran esfuerzo creativo e inversión de tiempo y dinero. Es justo que los creadores de obras de este tipo reciban una retribución por ese costo y por los beneficios que los usuarios derivamos de sus creaciones.
- Porque la existencia de un régimen que garantice a los creadores de software que sus derechos serán respetados es evidentemente un *sine qua non* para que estén dispuestos a producirlo masivamente y a precios accesibles. Si no existiese esa seguridad, difícilmente habría un incentivo para distribuir las aplicaciones al público en general.
- Porque el hecho de que un país mantenga un sistema adecuado de protección de los derechos de propiedad intelectual sirve para atraer la inversión extranjera y facilitar la transferencia tecnológica. Esto, a su vez, genera empleos y estimula el desarrollo socioeconómico.

Ahora bien, el perenne desafío en materia de propiedad intelectual en general ha sido determinar cómo obtener el justo equilibrio entre la necesidad de proteger y estimular la creatividad y la innovación, de una parte, frente a la necesidad –igualmente perentoria– de garantizar que los avances técnicos, científicos, artísticos y culturales se difundan ampliamente a la comunidad en general y mejoren el bienestar de la mayoría.

*“El bien público es el mejoramiento de la sociedad que resulta de [promover] el progreso de las ciencias y de las artes útiles; el interés privado se encarna en el monopolio temporal (derecho de autor o patente) conferido a alguien que haya hecho una contribución a ese progreso. El reto es obtener y mantener un balance, posibilitando un control que incentive a los autores, inventores y editores, pero no tanto control que ponga en peligro importantes objetivos de política pública, tales como preservar la herencia cultural nacional, proveer amplio acceso a la información y promover la educación, la ciencia y la erudición.”<sup>4</sup>*

<sup>4</sup> DAVIS, Randall. “The digital dilemma”, en *Communications of the ACM*, volumen 44, número 2, febrero del 2001, página 77. Traducción libre del autor; los paréntesis no son del original.

Es decir, en un sistema jurídico adecuado de protección de la propiedad intelectual, podemos visualizar la existencia de una balanza imaginaria, por medio de la cual se equilibran los intereses de los creadores e inventores, por un lado, así como de los consumidores y de la población en general, por otro.<sup>5</sup>

En la actualidad, sin embargo, la tecnología informática ha venido a ofrecer un poderoso instrumento para inclinar la balanza en contra de los intereses de los creadores y de toda la enorme industria que opera a partir de sus creaciones. Además, ha venido a plantear un profundo cuestionamiento de las instituciones tradicionales de tutela jurídica de las producciones del intelecto. En efecto, las facetas fundamentales del proceso por el cual se disemina la información (publicación, distribución y reproducción) se han visto radicalmente alteradas por la tecnología. Antes del advenimiento de la informática –y, en especial, de la Internet– la publicación de una obra (por ejemplo, una novela) requería un proceso más o menos lento y costoso, en el que por lo general debía intervenir una casa editora. La posterior distribución de la obra resultante debía hacerse por medios igualmente lentos y costosos (transporte aéreo, marítimo y/o terrestre). Finalmente, la reproducción también implicaba una serie de factores de importancia: hacerlo legalmente quedaba a cargo de la misma casa editorial, con los consiguientes costos; hacerlo ilegalmente (por ejemplo, fotocopiando la novela) es lento, caro y conduce a copias de calidad notoriamente inferior.

Pero ahora, la ecuación ha cambiado de manera drástica. Prácticamente cualquiera puede publicar en Internet, prescindiendo de un editor. La distribución mundial es instantánea y tiene un costo ínfimo. La reproducción de obras en formato digital es rápida, de bajo costo y produce copias perfectas. De este modo,

*“... las barreras naturales a la infracción [de los derechos de propiedad intelectual] se han erosionado significativamente. (...) Las partes interesadas en todas las facetas del tema se preguntan si algo se puede hacer para restaurar el balance de fuerzas o si el mundo ha cambiado de manera fundamental y permanente.”<sup>6</sup>*

– 0 –

<sup>5</sup> Una excelente manifestación de este principio se encuentra en el artículo 1, inciso b), de nuestra Ley de Información No Divulgada (número 7975 del 4 de enero del 2000), que fija como uno de sus objetivos el de “Contribuir a promover la innovación tecnológica y la transferencia y difusión de la tecnología, en beneficio recíproco de productores y usuarios, de los conocimientos tecnológicos, de modo que favorezcan el bienestar socioeconómico, así como el equilibrio de derechos y obligaciones.”

<sup>6</sup> DAVIS, *op. cit.*

Está claro que todo lo dicho es profundamente atinente al software, que, una vez publicado, estará expuesto a los mismos problemas en lo que a distribución y reproducción se refiere. Y, consecuentemente, a la pregunta de por qué protegerlo, cabe agregar además la de cómo hacerlo.

## Conceptos básicos de propiedad intelectual

En general –es decir, incluyendo pero no referido exclusivamente al software– el régimen de la propiedad intelectual existe sobre la base de un reconocimiento que la sociedad hace a la inventiva individual o colectiva. Es una proclamación de que una determinada creación del intelecto reúne características de *creatividad* y *originalidad* que la hacen merecedora de tal reconocimiento.

*“Las simples ideas no tienen la protección de la ley, pero cuando las ideas se exteriorizan en forma original, poseyendo individualidad e integridad propia, se transforman en una obra intelectual, que merece entonces el amparo del ordenamiento jurídico.”<sup>7</sup>*

Como su nombre lo indica, hablamos aquí de una forma especial del derecho de propiedad,<sup>8</sup> lo cual implica, al igual que en la propiedad tradicional, que su titular es libre para utilizarla como desee –siempre que ese uso no infrinja la ley, la moral, el orden público o los derechos de terceros–, así como para impedir que otros invadan o utilicen indebidamente esa propiedad. Al mismo tiempo, sin embargo, difiere de la concepción clásica del dominio, entre otras razones, por su carácter temporal.<sup>9</sup>

Nuestra Constitución Política expresamente tutela la propiedad intelectual, al decir:

*“Artículo 47.- Todo autor, inventor, productor o comerciante gozará temporalmente de la propiedad exclusiva de su obra, invención, marca o nombre comercial, con arreglo a la ley.”*

<sup>7</sup> FERNÁNDEZ DELPECH, Horacio. *Protección jurídica del software*. Abeledo-Perrot, 1 edición, Buenos Aires, 2000, página 7.

<sup>8</sup> “No hay propiedad más particular ni más legítima al hombre que aquella que es producto de su trabajo y de su mente”, reza el preámbulo de la Ley Fundamental del estado de Massachusetts, Estados Unidos, 17 de marzo de 1789. Citada en: ZÚÑIGA CHAVES, Nuria. *Op. cit.*

<sup>9</sup> FERNÁNDEZ DELPECH, *op. cit.*, página 9.

También se refiere a este instituto el artículo 27, párrafo 2, de la Declaración Universal de Derechos Humanos, que indica:

*“Cada uno tiene el derecho a la protección de los intereses morales y materiales procedentes de toda producción científica, literaria o artística de la cual es autor.”<sup>10</sup>*

El marco institucional y normativo de ámbito internacional para la protección de la propiedad intelectual es bastante amplio. La Organización Mundial de la Propiedad Intelectual (OMPI), creada en 1967 como agencia especializada de la Organización de las Naciones Unidas, es la principal rectora de la materia en el mundo.<sup>11</sup> Costa Rica es parte de ella a partir de la vigencia de la ley número 6468 de 18 de setiembre de 1980, que aprobó la adhesión de nuestro país al instrumento de su creación.

La propiedad intelectual es, en sí, un concepto muy amplio, que incluye a “las obras literarias, artísticas y científicas; las interpretaciones de los artistas intérpretes y las ejecuciones de los artistas ejecutantes, los fonogramas y las emisiones de radiodifusión; las invenciones en todos los campos de la actividad humana; los descubrimientos científicos; los dibujos y modelos industriales; las marcas de fábrica, de comercio y de servicio, así como los nombres y denominaciones comerciales; la protección contra la competencia desleal; y todos los demás derechos relativos a la actividad intelectual en los terrenos industrial, científico, literario y artístico”.<sup>12</sup>

Dada esta amplitud, el estudio de esta materia suele desagregarse en dos grandes campos: el del derecho autoral (que incluyen los derechos de autor y los llamados “derechos conexos”<sup>13</sup>) y el de la propiedad industrial (que comprende a las patentes y dibujos industriales, las denominaciones de origen, los secretos comerciales o “información no divulgada”, las marcas y la competencia desleal).

<sup>10</sup> Es importante no perder de vista que la Constitución Política define a la propiedad intelectual como un derecho *individual* y que la Declaración lo tilda de derecho *humano*. Esto porque las grandes empresas comerciales suelen remitirse a estas disposiciones como parte de sus alegatos en contra de la piratería de obras protegidas, olvidando que ellas no son las beneficiarias de los referidos textos normativos. No debería ser necesario insistir en ello por evidente, pero en razón de lo explicado, valga reiterar aquí que el intelecto es un atributo de la persona y no de las entidades colectivas.

<sup>11</sup> Ver la dirección de su sitio web entre las direcciones relevantes, al final de esta obra.

<sup>12</sup> Convenio que establece la Organización Mundial de la Propiedad Intelectual, firmado en Estocolmo el 14 de julio de 1967; artículo 2, punto viii.

<sup>13</sup> Por medio de los cuales se otorga protección respecto de las contribuciones creativas de todos los que se ocupan de presentar obras al público, como los artistas intérpretes o ejecutantes, los productores de fonogramas y los organismos de radiodifusión.

## Protección del software

Es necesario comenzar destacando que el problema de cómo tutelar al software es una cuestión diferente a la de cómo hacerlo respecto del hardware, aunque ésta tampoco puede ser tenida como una afirmación absoluta e incontrovertible. En efecto, aunque está claro que el régimen jurídico aplicable al hardware es el usual de las patentes de invención y los modelos industriales, existe un área gris en tratándose de aquellos elementos físicos que almacenan software de manera integrada a sus circuitos, caso en el que el trazado mismo de estos últimos amerita considerarse una obra sometida a un régimen propio.

Este hecho ha dado pie a la suscripción de un “Tratado sobre la Propiedad Intelectual respecto de los Circuitos Integrados” (adoptado en Washington el 26 de mayo de 1989),<sup>14</sup> así como a la promulgación en el medio costarricense de la “Ley de protección a los sistemas de trazado de los circuitos integrados”, número 7961 del 17 de diciembre de 1999.

Consideremos, entonces, cuáles han sido los diversos mecanismos por los que se ha propuesto lograr una adecuada tutela legal del software propiamente dicho.

En esencia, los principales son los tres siguientes:

1. Derecho de autor
2. Patentes de invención
3. Sistemas de protección *sui generis*

Cada mecanismo tiene sus particulares ventajas y desventajas, que examinamos a continuación.<sup>15</sup>

---

<sup>14</sup> El texto se puede consultar en <http://www.wipo.org/spa/ipler/wo-top01.htm>

<sup>15</sup> Aunque en menor grado a mi juicio, el derecho marcario también es capaz de otorgar algún margen de protección al software. Después de todo, “Es difícil para los piratas vender ilegalmente copias de software y acceder a un público masivo sin usar su palabra distintiva, frase, logo o símbolo gráfico utilizado para identificar la fuente del producto y distinguir al fabricante.” MARESCA, Fernando. “¿Hacia el patentamiento del software?”, en *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 23, junio del 2000.



## Protección por medio del derecho de autor

### *Panorámica del derecho autoral*

El derecho autoral es el campo específico de la propiedad intelectual que se aplica a las obras literarias, artísticas y científicas, así como a las obras tecnológicas (como el software).<sup>16</sup>

Entre los principales textos jurídicos aplicables a esta materia encontramos:

- El *Convenio de Berna para la Protección de las Obras Literarias y Artísticas*, de 1866, revisado en 1971 y ratificado por nuestro país por ley número 6083 de 29 de agosto de 1977.<sup>17</sup>
- La *Convención Universal sobre Derechos de Autor*, de 6 de setiembre de 1952 y revisada en 1971, promulgada a iniciativa de la UNESCO y que Costa Rica incorporó a su ordenamiento interno mediante ley número 1680 de 6 de noviembre de 1953.
- El *Acuerdo sobre los Aspectos de los Derechos de Propiedad Intelectual relacionados con el Comercio* (ADPIC), que a su vez es fruto de la Ronda Uruguay de Negociaciones Comerciales del GATT y cuya administración actualmente incumbe a la Organización Mundial del Comercio (OMC).<sup>18</sup> Nuestro país aprobó el Acta Final respectiva mediante ley número 7475 de 20 de diciembre de 1994. El ADPIC constituye propiamente el Anexo 1C de esta legislación. Y,
- El *Tratado de la OMPI sobre derecho de autor*, adoptado por la Conferencia Diplomática de la OMPI sobre ciertas cuestiones de derecho de autor y derechos conexos, en Ginebra, el 20 de diciembre de 1996; que es complemento de la Convención de Berna y que fuera ratificado por Costa Rica mediante ley número 7968 de 16 de diciembre de 1999.<sup>19</sup> El “WCT” –como también se le conoce por sus siglas en inglés– es uno de los dos llamados “Tratados Internet” de la OMPI,

<sup>16</sup> En la legislación así como en la mayoría de la doctrina, los programas de cómputo y las bases de datos son consideradas obras literarias o artísticas. Lo evidentemente absurdo de esa concepción lleva a otros sectores a clasificarlas en la categoría propia de “obras tecnológicas”.

<sup>17</sup> El texto en español está disponible en <http://www.wipo.org/spa/iplex/wo-ber01.htm>

<sup>18</sup> El texto, en formato PDF, puede ser descargado desde [http://www.wto.int/spanish/docs\\_s/legal\\_s/27-trips.pdf](http://www.wto.int/spanish/docs_s/legal_s/27-trips.pdf)

<sup>19</sup> Entró en vigor el 6 de marzo del 2002, tres meses después de recibida por la OMPI la trigésima adhesión al tratado.

diseñados para establecer un marco jurídico que salvaguarde efectivamente los intereses de los creadores en el ciberespacio y abrir nuevos horizontes para que los compositores, artistas, escritores y demás personas puedan utilizar Internet con confianza para crear, distribuir y controlar la utilización de sus obras en el entorno digital.<sup>20</sup>

En el nivel interno, nuestro principal marco normativo en esta materia lo provee, en primer término, la Ley de Derechos de Autor y Derechos Conexos (número 6683 de 14 de octubre de 1982 y sus reformas). La LDADC cuenta a su vez con dos reglamentos, contenidos en los decretos ejecutivos número 23485-MP de 5 de julio de 1994 (que regula la aplicación del artículo 50 de la Ley) y 24611-J de 4 de setiembre de 1995.

En materia de propiedad intelectual, también contamos con un conjunto de leyes más recientes, promulgadas para cumplir con los compromisos adquiridos por el país como condición para su ingreso al GATT (hoy OMC) y que, a su vez, fueron fruto de la Ronda Uruguay de esa organización. Entre ellas y en lo que aquí nos concierne, destaca la *Ley de Procedimientos de Observancia de los Derechos de Propiedad Intelectual*, número 8039 de 12 de octubre del 2000.

– o –

En sí, la denominación “derecho de autor” es imprecisa, porque bajo ese término figura en realidad un complejo haz de derechos, todos los cuales convergen en cabeza del creador de la obra. Esos derechos son agrupados tradicionalmente en dos categorías: los *derechos patrimoniales* y los *derechos morales*.

La tutela que otorga el derecho de autor implica que determinadas utilidades de una obra protegida son legítimas sólo si cuentan con la autorización del titular. En este sentido, los llamados **derechos patrimoniales** de autor permiten a su titular obtener una remuneración derivada del uso de la obra protegida por otros (es decir, de la explotación económica de la obra), incluyendo:

- La publicación, copia o reproducción de cualquier forma de la obra. La importancia de este extremo es tal que, de hecho, da origen al término inglés *copyright* (“derecho de copia”), referido genéricamente al derecho de autor.

---

<sup>20</sup> El otro “Tratado Internet” es el “WPPT”, o “Tratado de la OMPI sobre Interpretación o Ejecución y Fonogramas”, al cual nos referiremos de nuevo más adelante. Este último entró en vigencia el 20 de mayo del 2002.

- La distribución de copias a terceros, gratuitamente o a cambio de un precio.
- El alquiler de copias de ciertas categorías de obras, incluyendo el software.
- La grabación de las interpretaciones o ejecuciones de obras literarias o musicales.
- La interpretación o ejecución en público, especialmente de obras musicales, dramáticas o audiovisuales.<sup>21</sup>
- La comunicación a terceros, de cualquier forma, de las interpretaciones o ejecuciones de dichas obras. Esto incluye transmitir las por radio, televisión o por otros medios inalámbricos.
- La traducción a otros idiomas.
- La adaptación de cualquier tipo de obra y, especialmente, el hecho de producir obras audiovisuales a partir de las mismas (por ejemplo, adaptar una novela a una cinta cinematográfica).

Es importante destacar que estos derechos patrimoniales distan mucho de ser absolutos o irrestrictos –como no lo es ningún otro derecho–, pudiendo sufrir ciertas atenuaciones o limitaciones, en atención de ciertas finalidades socialmente relevantes. En esencia, estas limitantes se manifiestan: a) en el carácter finito, en el tiempo, de los derechos patrimoniales; b) en las diversas excepciones y exclusiones previstas en la ley; y, c) en la noción de los “usos honestos” de una obra.

Bajo ese último concepto, (que es de gran interés para lo que más abajo analizaremos específicamente en cuanto al software), algunas utilidades estrictamente determinadas (por ejemplo, las citas bibliográficas, el uso de obras para ilustración en la enseñanza o la utilización de artículos sobre asuntos políticos o económicos publicados en otros diarios) son completamente libres, es decir, no requieren ni la autorización ni el pago de remuneración alguna al titular del derecho de autor. Se les conoce como “usos honestos” de una obra.<sup>22</sup>

---

<sup>21</sup> “Comunicación pública es todo acto por el que una pluralidad de personas tiene acceso al contenido de [una] obra sin necesidad de entregar a cada una de ellas un ejemplar.” CARRASCO, Juan. “Directiva de derechos de autor, ¿una verdadera armonización?”, en *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 98, noviembre del 2001.

– 0 –

Además de los derechos patrimoniales, los autores (sean o no titulares de los derechos patrimoniales) gozan de ciertos **derechos morales**, que les permiten tomar determinadas medidas para preservar el nexo personal existente entre autor y obra; es decir, para salvaguardar el aspecto inherente a la personalidad del autor que se refleja en su obra. En particular, este concepto lo legitima para:

- Reivindicar una obra como suya.
- Exigir que se indique su nombre en los ejemplares de la obra y en otros usos de la misma. Y,
- Oponerse a la mutilación o deformación de la obra.

Una diferencia esencial entre ambas categorías de derechos es que los patrimoniales son disponibles (es decir, cuantificables, transferibles y renunciables<sup>23</sup>) y temporales (sujetos a plazo). Por su parte, los derechos morales son inestimables, inalienables, irrenunciables y perpetuos.

Finalmente, una importante particularidad de este instituto es que la protección que otorga el derecho de autor es independiente de toda formalidad. Esto implica que esa tutela comienza desde el momento mismo de la creación de la obra,<sup>24</sup> sin más formalidad.<sup>25</sup> Por consiguiente, su eventual inscripción en los diversos registros de derechos de autor es optativa, poseyendo un carácter puramente declarativo y de certeza jurídica.

---

<sup>22</sup> Por “usos honestos” entendemos “los que no interfieren con la explotación normal de la obra ni causan un perjuicio irrazonable a los intereses legítimos del autor”. ZÚÑIGA CHAVES, *op. cit.*

<sup>23</sup> De interés para el caso del software y otras obras digitales es advertir que cualquier renuncia de derechos tiene que ser expresa. No se admite tal cosa como una renuncia implícita, surgida –por ejemplo– del hecho de que una obra haya sido copiada y distribuida repetidamente sin que el autor haya tomado acciones legales al respecto. Esto es frecuente en el contexto de la Internet, con relación a obras tales como las imágenes denominadas “*clip art*” y otras.

<sup>24</sup> En los países cuyo régimen jurídico es de *common law*, se suele exigir alguna forma de “fijación” de la obra, que implica hacerla constar en un soporte material más o menos permanente (imprimir un libro, grabar una canción, grabar un programa en disco, etcétera), tal que se pueda percibir, reproducir o comunicar. “Esta fijación ocurre cuando una expresión es suficientemente permanente o estable para permitir que sea percibida, reproducida o de otro modo comunicada por un período de una duración más que transitoria” (KOHN, Paul. “Understanding Copyright Law”, en *Revista Electrónica de Derecho e Informática* [<http://www.alfaredi.org>], número 98, noviembre del 2001).

<sup>25</sup> Ni siquiera es exigido que una obra haga advertencia expresa en el sentido de ser protegida, por medio del signo internacional “©” u otro mecanismo. KOHN, *op. cit.*

Para proteger el derecho así obtenido, las diversas legislaciones nacionales e internacionales conceden una serie de mecanismos, incluyendo las medidas precautorias o provisionales, recursos civiles, sanciones penales, adopción de medidas en las fronteras nacionales; y medidas, recursos y sanciones contra los abusos cometidos respecto de los medios técnicos.<sup>26</sup>

### **Aplicación al software**

La vía de la protección del software a través de los derechos de autor es la que han adoptado preferentemente hasta hoy la mayoría de los países, incluyendo al nuestro.

En efecto,

*“Tradicionalmente, y sobre todo a partir de los años 80, se ha discutido sobre cuál sería la disciplina jurídica más idónea para proteger el software, debatiéndose si constituían creaciones artísticas, a modo de obra, o invenciones susceptibles de ser objeto de patente. Desde un principio parecía claro que la protección debía venir del Derecho de la Propiedad Intelectual o Derecho de autor. La afinidad conceptual de los programas de ordenador con las obras científicas, la dificultad para encajar el concepto de programa de ordenador en las categorías del Derecho de patentes y la amplitud del radio de acción protector del Derecho de la Propiedad Intelectual constituían el inmejorable caldo de cultivo para la protección jurídica del software a través de dicha disciplina.”<sup>27</sup>*

Ahora bien, antes de proseguir, juzgamos necesario clarificar algunos aspectos relativos a los derechos de autor respecto de una aplicación informática.

En primer término, si bien nada impide (ni es infrecuente) que un programa de computación sea creado por una sola persona, lo más usual en la actualidad es que el software sea el fruto del trabajo de múltiples individuos, en cuyo caso es necesario diferenciar, para efectos de tutela, si en la obra resultante se hace posible o no distinguir el aporte concreto de cada cual.

Se llama *obra en colaboración* a aquella donde las distintas aportaciones de los autores son objetivamente identificables entre sí. En este caso, los derechos de

<sup>26</sup> Organización Mundial de la Propiedad Intelectual (OMPI). “Acerca de la propiedad intelectual”, publicación electrónica disponible en <http://www.wipo.int/about-ip/es/>.

<sup>27</sup> ERDOZAIN, José Carlos. “Un ensayo sobre la protección jurídica de los programas de ordenador”. En *Revista Electrónica de Derecho e Informática*, número 108, setiembre del 2002. Disponible en <http://www.alfa-redi.org/upload/revista/90302--12-50-Der.Infor%20julio01.pdf>

propiedad intelectual sobre la obra (el software, en este caso) pertenecen a todos ellos en la proporción que determinen, entendiéndose que en ausencia de otra fijación, lo harán por partes iguales. Además, en caso de fallecimiento de alguno de los coautores que no deje herederos, sus derechos acrecerán a los de los restantes (artículo 18 de la Ley de Derechos de Autor).

Cuando no es posible diferenciar en el todo la parte o partes aportadas por cada persona, estaremos frente a lo que en la terminología del derecho autoral se denomina *obra colectiva*. El Reglamento de la Ley de Derechos de Autor se refiere a esta clase de obra como la “creada por varios autores por iniciativa y bajo la responsabilidad de una persona natural o jurídica que la publica con su propio nombre, y en la cual las contribuciones de los autores participantes –por su elevado número o por el carácter indirecto de esas contribuciones–, se fusionan en la totalidad de la obra de modo que hace imposible identificar a los diversos autores ni atribuir, a cualquiera de ellos, una determinada participación.”<sup>28</sup>

La titularidad de las obras colectivas no corresponde a cada uno de los partícipes en su creación, sino a la persona física o jurídica que se ha encargado de impulsar, financiar y coordinar la publicación de la obra. En efecto,

*“La obra colectiva nace como respuesta funcional a la exigencia de que las personas jurídicas puedan ser consideradas como autor, venciendo de esa manera la tradicional oposición mostrada a que sujetos que no fuesen personas físicas pudieran atribuirse derechos creativos.”<sup>29</sup>*

En el caso del software, esto explica que los derechos de autor sobre la mayoría de las aplicaciones comerciales se atribuyan a una empresa determinada. En este sentido específico, expresamente agrega el citado Reglamento:

*“Artículo 7.- Salvo en los casos en que se trate de una obra individual, o que haya sido publicada con el nombre de los autores, se presume, salvo prueba en contrario, que el programa de cómputo es una obra colectiva, cuya titularidad corresponde, en los términos del artículo 6º de la Ley, al productor, quien además de los derechos de orden patrimonial, tiene la facultad de defender al derecho moral, en la medida que ello sea necesario para la explotación de la obra.”*

Este problema ya ha dado lugar a importantes pronunciamientos por parte de la jurisprudencia nacional. En efecto, la Sala Segunda de la Corte Suprema de

---

<sup>28</sup> Artículo 3.

<sup>29</sup> ERDOZAIN, *op. cit.*

Justicia, en sentencia número 94-415-LAB de las 9:00 horas del 22 de diciembre de 1994, analizó el caso de un trabajador que fue despedido con responsabilidad patronal. Cuando éste se presentó a retirar sus prestaciones, el pago le fue negado, porque la empleadora constató que el trabajador, antes de retirarse de la empresa, había despojado a los programas creados por él durante la relación laboral –y que se utilizaban como instrumentos de trabajo en la empresa– de los respectivos códigos fuente, dejándolos encriptados. Se examinó entonces el tema de la propiedad intelectual de estas aplicaciones, reafirmando claramente el derecho moral del autor a ser reconocido como su creador. Sin embargo, se precisó:

*“Pero el hecho de que el trabajador conserve la titularidad del derecho moral sobre los programas de ordenador creados por él con ocasión del contrato de trabajo, no puede permitirle, salvo pacto en contrario, que, al concluir la relación laboral, pueda despojar a esos programas de sus ‘códigos fuente’ e impedirle a su dueño patrimonial acceder a ellos, si tal cosa es indispensable para mantener su utilidad en la función o tarea para la que fueron hechos, pues, entonces, la característica intrínseca de adaptabilidad del programa de ordenador de acuerdo con la variación de los factores atinentes a esa tarea o función, es determinante de su valor como bien jurídico, razón por la cual, el conocimiento de las codificaciones que permita ir haciendo los cambios que se requieran, se torna en algo íntimamente ligado al aspecto patrimonial de la obra perteneciente al empleador. Al respecto, la doctrina enseña que cuando un trabajador es contratado para realizar una obra literaria, por ese hecho, aún implícitamente, queda obligado a transmitirle los derechos al empleador y a permitirle el uso o la explotación del bien por él creado, de acuerdo con la finalidad para la que se le creó en favor del patrono, sin perjuicio de su derecho moral, que no es transmisible (véase Krotoschin, op. cit. p. 414). Así las cosas, en ejercicio de ese derecho moral, el autor de la obra ya comunicada en aquella forma, sólo puede exigir del patrono, como titular del derecho de utilizarla, –salvo convenio en contrario, se insiste, que consigne reserva de derecho en favor del autor de la obra, como parte de sus derecho de explotación económica–, que se indique su nombre en cada uno de los programas y poner en práctica las demás facultades que enumera el artículo 14 de la Ley de Derechos de Autor y Derechos Conexos, ninguna de las cuales le reserva al autor la exclusividad de hacerle aquellos cambios de los que depende la utilidad de los programas como bienes y mal podría deducirse así de la aplicación ampliativa de esas reglas, pues ello equivaldría a dejar en manos del autor el decidir, ante los cambios sobrevinientes, la utilización del titular del derecho patrimonial, lo cual*

*es a todas luces inconveniente, pues de esa manera podrían resultar legitimadas las prácticas de la mala fe y, en general, el uso abusivo del derecho, en abierta contradicción con el ordenamiento (artículo 22 del Código Civil).”*

Y además:

*“Según se desprende del convenio escrito, que se firmó entre las partes al inicio de la relación, el actor fue contratado por la demandada para realizar en su beneficio el diseño y programación de sistemas computarizados (véase fotocopia de ese documento que se conserva en legajo aparte) y en cumplimiento de esa prestación hizo varios sistemas para ser utilizados en el Centro de Cómputo de la demandada, en diversas tareas, entre las que se pueden citar las relacionadas con planillas, inventario y facturación, cuya operación es dependiente de factores variables. En consecuencia, el señor [...] fue consciente, desde el primer momento, de que los programas por él creados para ser operados en dichos campos requerían ser modificados para ir adaptándolos al comportamiento de las variables, y si en el convenio no se dijo nada al respecto, de acuerdo con la doctrina comentada y la ley, debe entenderse como obligación a él correspondiente el traspaso a la empleadora de los respectivos instrumentos en todas sus partes necesarias para poder ejercitar el derecho patrimonial de utilización, con su presencia o sin ella, pues deben reputarse legítimamente adquiridos y pagados a través de las contraprestaciones salariales. La conducta expuesta por el actor, de despojar a los programas de sus ‘códigos fuente’, para hacerlos inaccesibles sin su intervención, y de pretender, a partir de esa situación, retribuciones económicas indebidas, es francamente contraria a la buena fe que debe caracterizar en todo momento las relaciones laborales, y justifica la conclusión del contrato sin responsabilidad para el empleador (doctrina del artículo 81, inciso d), del Código de Trabajo), sobre todo que en el caso concreto, el propósito dañoso del actor se hizo realidad, pues ante la negativa de satisfacer aquellas pretensiones indebidas, la empresa vio afectado su funcionamiento, ya que las primeras modificaciones relacionadas con las funciones para las que se crearon los programas, como por ejemplo el de ‘facturación’ con la variación del impuesto sobre las ventas, la obligaron a trabajos prolongados y onerosos para poner en práctica verdaderos nuevos programas.”<sup>30</sup>*

---

<sup>30</sup> Los paréntesis cuadrados no son del original.



– 0 –

Regresemos a la discusión sobre la protección jurídica.

Optar por aplicar al software el régimen de tutela del derecho autoral tiene ventajas indudables frente a la opción de hacerlo mediante patentes de invención (que analizaremos más adelante). Las podemos sintetizar así:

- Se elimina la necesidad de registrar un software nuevo para obtener la tutela jurídica. Mientras que la inscripción de una patente tiene eficacia constitutiva, la de una obra la posee meramente declarativa.
- Se evita tener que revelar el código fuente de la aplicación.<sup>31</sup>
- El plazo de protección es mayor.
- Desaparecen las exigencias de novedad y aplicabilidad industrial propias de las patentes.

Correlativamente, a la tutela vía derechos de autor se asocia una serie de desventajas o debilidades. En primer lugar, cabe preguntarnos si el software es verdaderamente una *obra*. Algunos autores cuestionan este aspecto –y, por ende, la aplicabilidad misma del régimen de derechos de autor–, en la medida en que afirman que el concepto clásico de obra (al menos en lo relativo a las artísticas y literarias) lleva implícito la posibilidad de pronunciar un juicio estético respecto de ella, cosa que difícilmente puede hacerse con relación a un programa informático. Desde este punto de vista, está claro que la posibilidad de calificar a éste como obra, en ese sentido tradicional, es viable solamente por medio de establecer una equiparación o analogía –ya de por sí forzada– con las científicas.

Por otra parte, algunos han destacado que los programas de cómputo son esencialmente medios para la obtención de un logro o resultado, pero no son el resultado en sí mismo. En efecto, según TROLLER,<sup>32</sup>

*“[Los] programas de ordenador constituyen una categoría especial, la de ‘métodos intelectuales’, los cuales no pueden incluirse en la categoría de obras artísticas y literarias, ya que el objeto de un programa de computación es el de dar instrucciones para una*

<sup>31</sup> Conforme al artículo 103, inciso 5), de la Ley de Derechos de Autor, “Cuando se trate de inscribir un programa de cómputo o una base de datos, la solicitud se presentará con cualquiera de los siguientes elementos: el programa, la descripción o el material auxiliar.” En consecuencia, la presentación del programa –su código fuente– es optativa.

<sup>32</sup> Citado por CORREA, *op. cit.*

*secuencia de operaciones destinadas a lograr un resultado; por su parte, el objetivo de las obras literarias y artísticas es su mera existencia.”*

Desde esta óptica, podríamos calificar de obra artística a una imagen producida a través de una aplicación de software gráfico, pero no el software mismo. En nuestro criterio, sin embargo, esto no es estrictamente cierto. El software no está constituido solamente por programas sino también por archivos de datos, o combinaciones inseparables de programas y datos, los cuales podrían constituir en sí una creación literaria, artística o científica (por ejemplo, las páginas web, las enciclopedias electrónicas, las imágenes o los archivos de música MIDI). Por otra parte, el concepto tradicional de programa como secuencia de instrucciones dirigidas a lograr un resultado no es válido en algunas categorías, como por ejemplo los juegos, donde la operación del programa (su despliegue visual y auditivo) es propiamente el objeto de disfrute, sin que califique como un “resultado”, al menos no en el sentido de las aplicaciones de negocios. Por este motivo, lo pertinente sería distinguir la *obra tecnológica*, que es la aplicación misma (el software), de las obras literarias, artísticas o científicas que podemos producir mediante aquélla. Todas son meritorias de tutela jurídica en su respectiva modalidad. Además de las obras mencionadas, el ADPIC incluye otro tipo de obras que puede ser protegida por el derecho de autor. Se trata de las producciones de multimedios y, si bien no existe una definición legal, por consenso, cualquier combinación de sonidos, textos e imágenes en formato digital, que se pone a disposición por medio de un programa informático, se considera como una expresión de autor original y, por consiguiente, entra dentro del ámbito del derecho de autor.

En adición a lo expresado *supra*, es posible apuntar algunas debilidades adicionales a la protección jurídica del software por medio del derecho autoral. Como lo expresa MARESCA,

*“Amén de las bondades brindadas por este cuerpo normativo, la utilización del Derecho de autor a estos fines es objeto de diversas críticas. Muchos sostienen que si bien el software se ha visto beneficiado por la tutela provista por el Derecho de autor, la cobertura todavía es insuficiente. En efecto, este régimen solo protege la expresión de un trabajo, y no las ideas, los conceptos y las funciones subyacentes. Un productor independiente puede arribar a través de su trabajo a idéntico resultado y, en la medida en que no haya tenido*

*acceso a la obra que ya gozaba de protección, tendrá derecho a que la suya esté también tutelada.”<sup>33</sup>*

El hecho de que la funcionalidad subyacente del software no esté protegida como tal se manifestó claramente en un célebre caso resuelto en EE.UU. en 1996. La empresa Lotus Development Corp. demandó judicialmente a Borland International, Inc., aduciendo que esta última había copiado la estructura de menús del programa de hoja de cálculo Lotus 1-2-3, para incorporarla en su producto competidor Quattro Pro y, de este modo, evitar que las personas que lo adoptaran después de haber sido usuarios de Lotus 1-2-3 tuvieran que aprender una nueva interface operativa. La demanda fue desestimada, bajo el argumento de que el menú de acceso a las funcionalidades de la aplicación es tan solo un “método de operación” no protegido por el derecho de autor. La decisión de la mayoría del tribunal dejó en claro que la disposición particular de los comandos del menú de acceso a la aplicación no constituye, en sí, una “obra literaria o artística” capaz de recibir protección jurídica.<sup>34</sup>

– 0 –

Con el tiempo se ha comprobado la necesidad de variar algunos de los conceptos e institutos tradicionales del derecho autoral para adaptarlos a las particulares características del software. Entre estas adecuaciones, destacan las siguientes:

- La necesidad de redefinir los parámetros para juzgar la *originalidad* de una obra de software. En el caso de las obras tradicionales, la determinación de si existe o no originalidad es comparativamente simple y un diferendo sobre el particular quizás se resuelva en la mayoría de los casos por medio de una simple confrontación. Esto no ocurre igual tratándose del software, debido sin duda al carácter complejo de este último. Por tal motivo, algún sector de la doctrina se inclina por no establecer la novedad de una aplicación meramente por la inexistencia de copia del respectivo código o de sus rasgos visibles más obvios, sino más bien atendiendo a que exista una funcionalidad claramente diferenciable entre el programa que se evalúa y otros similares.<sup>35</sup>

<sup>33</sup> MARESCA, Fernando. “Protección jurídica del software: un debate abierto”. En *Derecho y Tecnología*, revista del Centro de Investigaciones en Nuevas Tecnologías de la Universidad Católica del Táchira, Venezuela, N° 1/2002, página 149.

<sup>34</sup> VAVER, David (editor). “*Principles of copyright*”. Publicación de la Organización Mundial de la Propiedad Intelectual, Ginebra, 2002.

<sup>35</sup> ERDOZAIN llega incluso a estimar que un juicio definitivo sobre la originalidad de un software radica en su interface de usuario. En nuestro criterio, aunque ese factor es desde luego de gran importancia, no es el único ni puede ser el más importante, ya que confundiría la

- El clásico *derecho de reproducción (copyright)*, que otorgaba al autor la posibilidad de controlar ampliamente lo relativo a la reproducción de su obra, necesariamente adquiere ahora un nuevo cariz. En efecto, cuando un usuario obtiene una copia de una aplicación, probablemente deberá instalarla en su computador, lo cual implica generar una copia adicional en el disco fijo. Además, cuando ejecute el programa, éste deberá ser transferido total o parcialmente a la memoria principal, lo que implica generar una reproducción más, aunque sea transitoria. Por otra parte, si pensamos en la distribución de software vía Internet (sea desde un servidor web o FTP), necesariamente estamos anticipando la posibilidad de la creación y descarga de un número potencialmente ilimitado de copias de la aplicación. Conceptos tradicionales, como la cantidad de ejemplares de la edición o reimpresión de un libro, pierden aquí todo sentido.<sup>36</sup>
- Otra modificación al referido derecho tradicional está en la posibilidad aceptada de que el adquirente realice al menos una copia de respaldo de la aplicación. No hace mucho, el principal canal de distribución del software eran las cintas y disquetes, que por sus características se prestaban para una pérdida de la información ya fuera por el paso del tiempo o por la exposición a los elementos o a campos electromagnéticos. En consecuencia, se tornaba necesario proteger la inversión –frecuentemente fuerte– en el software de base o de aplicación, contra estas eventualidades. Y de allí la aceptación generalizada de la posibilidad de hacer y mantener un respaldo de seguridad del software. Si bien los medios de distribución actuales (CD-ROM, Internet, etcétera) son menos propensos a experimentar problemas como los relacionados, la creciente dependencia de la informática hace que una adecuada política de respaldo de aplicaciones y datos siga formando parte esencial de la correcta administración de los recursos computacionales de cualquier entidad.<sup>37</sup> En efecto:

---

protección que corresponde al derecho de autor con la que brindan las regulaciones sobre dibujos y modelos industriales, a los que nos referimos más adelante.

<sup>36</sup> Al respecto, la “Conferencia Diplomática Sobre Ciertas Cuestiones de Derecho de Autor y Derechos Conexos” de la OMPI, adoptó el 20 de diciembre de 1996 una “declaración concertada” respecto del Artículo 1.4 del Tratado sobre Derecho de Autor (WCT), en el sentido de que “El derecho de reproducción, tal como se establece en el Artículo 9 del Convenio de Berna, y las excepciones permitidas en virtud del mismo, son totalmente aplicables en el entorno digital, en particular a la utilización de obras en forma digital. Queda entendido que el almacenamiento en forma digital en un soporte electrónico de una obra protegida, constituye una reproducción en el sentido del Artículo 9 del Convenio de Berna.”

<sup>37</sup> Algunos autores han advertido que toda copia de respaldo debe estar debidamente identificada, con indicación de la persona autorizada que realizó la copia y la fecha de ésta, siempre respetando su finalidad de reemplazar el ejemplar original, en caso de pérdida o si llegara a devenir inútil. JEANGORGES, Gonzalo. “La protección legal de las obras de

*“Parece indudable que debe primar la interpretación flexible en virtud de la cual el carácter necesario de la copia parece en todo caso, puesto que los perjuicios económicos derivados de la ausencia de una copia de seguridad que utilizar en caso de fallo general del sistema licenciado serían mayores que el eventual perjuicio que el legítimo usuario podría causar al titular de derechos. Un examen funcional y finalista de los intereses en juego debe llevar a esa conclusión.”<sup>38</sup>*

- En el mismo sentido, la Directiva 2001/29 del Parlamento y del Consejo Europeos ha establecido una interesante excepción al derecho tradicional de reproducción de obras protegidas, relativa a la técnica informática conocida como *caching*<sup>39</sup>. Pensando en las páginas web (pero empleando un razonamiento perfectamente aplicable también al software transmitido por Internet), la directiva “viene a reconocer las actividades de ‘caching’ (guardar archivos en la memoria caché del ordenador), ya se realice en el mismo equipo del usuario (‘local caching’, en la memoria caché del ordenador del usuario que se conecta), ya se realice en el servidor de acceso a Internet (‘proxy caching’, en la memoria caché del proxy). Dicha excepción está justificada para el caso de que dichos actos de reproducción provisional sean transitorios o accesorios y formen parte integrante y esencial de un proceso tecnológico cuya única finalidad consista en la utilización o en facilitar la transmisión, a través de una red, de una obra o prestación protegidas. Un requisito imprescindible exigido por la Directiva es que dichos actos no tengan por sí mismos una significación económica independiente. Esta excepción está prevista, principalmente, para hacer posible el almacenamiento, en la memoria caché de un ordenador, de las páginas web visitadas, lo que facilita su posterior visualización y navegación a través de ellas, al permanecer una copia de las mismas en dicha memoria.”<sup>40</sup>

---

Informática en la Argentina”, en *Revista Electrónica de Derecho e Informática* (<http://www.alfaredi.org>), número 22, mayo del 2000.

<sup>38</sup> ERDOZAIN, *op. cit.*

<sup>39</sup> En el *caching*, determinada información que ha sido descargada desde un origen remoto (por ejemplo, un servidor de Internet) es mantenida temporalmente en un medio de almacenamiento local, bajo la presunción de que podría ser requerida nuevamente a corto plazo. Si así ocurriese, al estar la copia provisional disponible localmente, se puede poner a disposición de los usuarios de inmediato, en vez de tener que restablecer la conexión remota para repetir la descarga.

<sup>40</sup> CARRASCO, Juan. *Op. cit.*

- En cuanto al *principio de integridad de la obra* (que es un derecho moral), se ha señalado que el autor de una aplicación de software no se puede oponer a las adaptaciones que realicen usuarios autorizados. En efecto, el particular giro de una empresa u organización podría requerir que el programa en cuestión sea personalizado para un mejor desempeño.<sup>41</sup> No podría entenderse en estos casos, entonces, que el software ha sido mutilado o deformado en modo alguno.
  
- Salvo pacto expreso en contrario, tampoco se ve contradicho el principio de integridad de la obra por el hecho de introducir rectificaciones o enmiendas necesarias en los programas, por parte de los usuarios legítimos.<sup>42</sup> Puesto que se admite que la mayoría de las aplicaciones de productividad de hoy son altamente complejas y susceptibles de contener restricciones e incluso errores que escapan del proceso normal de depuración y prueba, los esfuerzos que posteriormente sean desplegados para superarlos no podrían ser considerados como una distorsión de la obra protegida. En este sentido, por ejemplo, el artículo 100.1 de la Ley de Propiedad Intelectual española de 1987 señala que “no necesitarán autorización del titular, salvo disposición contractual en contrario, la reproducción o transformación de un programa de ordenador incluida la corrección de errores, cuando dichos actos sean necesarios para la utilización del mismo por parte del usuario legítimo, con arreglo a su finalidad propuesta”. Eso sí,

*“La aplicación de la excepción al ius prohibendi del titular de derechos juega siempre y cuando se den determinadas circunstancias. Debe tratarse, en primer término, del usuario legítimo. Se considera como tal al legítimo titular de una copia del programa informático, al adquirente de la misma y a cualquier otra persona legalmente autorizada por cualquiera de los anteriores para realizar actos de mantenimiento o de administración en relación con el programa. Aparte de los anteriores será también considerado como usuario legítimo al así señalado por documento contractual (p. ej., imaginemos que en un contrato de distribución así se indica y acuerda por las partes, a fin de que el distribuidor pueda ejecutar el*

---

<sup>41</sup> Reconociendo esta realidad, muchas de las principales aplicaciones estándar de hoy de hecho proveen capacidades de adaptación a los requerimientos del usuario. Por ejemplo, los paquetes que permiten agregar, quitar o cambiar botones en las barras de herramientas suplidas de fábrica por la aplicación.

<sup>42</sup> Por ejemplo, una futura reforma legal que obligue a ajustar un software que realiza cálculos de impuestos, sin lo cual el programa perdería sentido o utilidad.

*programa a título de demostración o de exhibición). Las partes pueden del mismo modo delimitar negativamente la persona del legítimo usuario, estableciendo quién no tendrá dicha consideración pese a las circunstancias que rodeen el uso previsto del programa informático.”<sup>43</sup>*

- Desaparece el *derecho de retracto*, por el cual el autor podría decidir retirar una obra suya de la circulación. En efecto, tratándose del software, son evidentes las graves consecuencias y perjuicios económicos que podrían derivar de una facultad tal para las organizaciones públicas o privadas cuyo funcionamiento (y, con él, la prestación de servicios a ciudadanos o clientes, respectivamente) dependa del software que se pretenda retractar.

En una nota crítica, algunos sostienen que todas estas excepciones y adaptaciones al instituto del derecho de autor que se ha introducido a favor del software, en general pretenden favorecer a los fabricantes de software, nunca al consumidor. Esta sería entonces una razón más a favor de la tesis de tratar la protección del software a través de un régimen *sui generis* (al que nos referiremos más abajo), en vez de a través del derecho de autor.<sup>44</sup>

### **El cuadro normativo**

La protección del software a través del derecho autoral se concreta en una diversidad de normas de orden nacional e internacional.

Por ejemplo, del ya citado “Tratado de la OMPI sobre derecho de autor” extraemos las disposiciones de interés siguientes:

*“Artículo 4.- Programas de ordenador. 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.*

*Artículo 5.- Compilaciones de datos (bases de datos). Las compilaciones de datos o de otros materiales, en cualquier forma, que por razones de la selección o disposición de sus contenidos constituyan creaciones de carácter intelectual, están protegidas como*

<sup>43</sup> ERDOZAIN, *op. cit.*

<sup>44</sup> SUÑÉ LLINAS, Emilio. *El caso Microsoft y las aporías del viejo derecho*. En “ABC Economía”, Madrid, 9 de abril del 2000, página 7.

*tales. Esa protección no abarca los datos o materiales en sí mismos y se entiende sin perjuicio de cualquier derecho de autor que subsista respecto de los datos o materiales contenidos en la compilación.*

*Artículo 7.- Derecho de alquiler. 1.- Los autores de: i) programas de ordenador; (...) gozarán del derecho exclusivo de autorizar el alquiler comercial al público del original o de los ejemplares de sus obras. 2.- El párrafo 1) no será aplicable: i) en el caso de un programa de ordenador, cuando el programa propiamente dicho no sea el objeto esencial del alquiler; (...).*

*Artículo 11.- Obligaciones relativas a las medidas tecnológicas. 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 y que, respecto de sus obras, restrinjan actos que no estén autorizados por los autores concernidos o permitidos por la Ley.*

*Artículo 12.- Obligaciones relativas a la información sobre la gestión de derechos. 1) Las Partes Contratantes proporcionarán recursos jurídicos efectivos contra cualquier persona que, con conocimiento de causa, realice cualquiera de los siguientes actos sabiendo o, con respecto a recursos civiles, teniendo motivos razonables para saber que induce, permite, facilita u oculta una infracción de cualquiera de los derechos previstos en el presente Tratado o en el Convenio de Berna: i) suprima o altere sin autorización cualquier información electrónica sobre la gestión de derechos; ii) distribuya, importe para su distribución, emita, o comunique al público, sin autorización, ejemplares de obras sabiendo que la información electrónica sobre la gestión de derechos ha sido suprimida o alterada sin autorización. 2) A los fines del presente Artículo, se entenderá por 'información sobre la gestión de derechos' la información que identifica a la obra, al autor de la obra, al titular de cualquier derecho sobre la obra, o información sobre los términos y condiciones de utilización de la obras, y todo número o código que represente tal información, cuando cualquiera de estos elementos de información estén adjuntos a un ejemplar de una obra o figuren en relación con la comunicación al público de una obra."*



Sobre el tema general de la protección jurídica de los programas informáticos también existen directivas del Parlamento Europeo de los años 1991<sup>45</sup>, 1996<sup>46</sup> y 2001.

El “Acuerdo sobre los ADPIC” de la OMC incorpora al software como categoría protegida bajo la legislación de derechos de autor. El artículo 10.1 de dicho texto normativo dispone en este sentido que “Los programas de ordenador, sean programas fuente o programas objeto, serán protegidos como obras literarias en virtud del Convenio de Berna (1971).”

Entre algunas otras legislaciones nacionales de relevancia en la materia se encuentra la *Digital Millenium Copyright Act* de los Estados Unidos,<sup>47</sup> de la que hablaremos nuevamente en la sección destinada al tratamiento de los mecanismos tecnológicos de defensa de las obras digitales.

En el nivel de la legislación costarricense, nos conciernen en especial las modificaciones introducidas en la Ley de Derechos de Autor y Derechos Conexos mediante leyes número 7397 de 28 de abril de 1994 y 7979 de 22 de diciembre de 1999, que se refieren específicamente a los programas de cómputo y bases de datos.

Conforme a la actual redacción de la Ley, citamos los artículos relevantes, en lo conducente:

Artículo 1:

*“Por ‘obras literarias y artísticas’ deben entenderse todas las producciones en los campos literario y artístico, cualquiera que sea la forma de expresión, tales como: (...) los programas dentro de los cuales se incluyen sus versiones sucesivas y los programas derivados ...”*

Artículo 4: Definiciones:

<sup>45</sup> Directiva 91/250/CEE de 14 de mayo de 1991, sobre la protección jurídica de programas de ordenador.

<sup>46</sup> El texto en inglés de esta última está en <http://www.ispo.cec.be/infosoc/legreg/docs/969ec.html>. Consúltese también: ULL PONT, Eugenio. *Protección jurídica de las bases de datos conforme a la Directiva 96/9/CEE del Parlamento Europeo y del Consejo, de 11 de marzo de 1996*. Publicación electrónica disponible en <http://www.omdi.net/members/espanolmiembros/reivdi/pont.htm>, Madrid, 1998.

<sup>47</sup> Cuyo texto se puede consultar en <http://lcweb.loc.gov/copyright/fedreg/da59233.pdf>

*“l) Reproducción: copia de obra literaria o artística o de una fijación visual o sonora, en forma parcial o total, en cualquier forma tangible, incluso cualquier almacenamiento permanente o temporal por medios electrónicos, aunque se trate de la realización bidimensional de una obra tridimensional o viceversa.”*

*“n) Programa de cómputo: Conjunto de instrucciones expresadas mediante palabras, códigos, gráficos, diseño o en cualquier otra forma que, al ser incorporados en un dispositivo de lectura automatizada, es capaz de hacer que una computadora –un aparato electrónico o similar capaz de elaborar informaciones– ejecute determinada tarea u obtenga determinado resultado. También forma parte del programa su documentación técnica y sus manuales de uso.”*

Artículo 8:

*“Las bases de datos están protegidas como compilaciones.”*

*“Artículo 74.- También es libre la reproducción de una obra didáctica o científica, efectuada personal y exclusivamente por el interesado para su propio uso y sin ningún ánimo de lucro, directo o indirecto. Esa reproducción deberá realizarse en un solo ejemplar, mecanografiado o manuscrito.*

*Esta disposición no se aplicará a los programas de computación.”*

*“Artículo 103.- Para inscribir una producción, el interesado presentará, ante el Registrador, una solicitud escrita con los siguientes requisitos:*

*(...)*

*5) Cuando se trate de inscribir un programa de cómputo o una base de datos, la solicitud se presentará con cualquiera de los siguientes elementos: el programa, la descripción o el material auxiliar.”*

Nótese, en primer término, de la definición de “programa” contenida en el artículo 4 inciso n), que ésta es extensiva a la documentación técnica y a los manuales de uso. En principio, entonces, tendríamos que valorar al “programa de

cómputo” como un objeto complejo, comprensivo al menos de tres realidades fácticas distintas: el código fuente, el código objeto y los manuales técnicos.<sup>48</sup>

Por otra parte, el artículo 74 pareciera contradecir la posibilidad de efectuar copias de respaldo de seguridad del software. Sin embargo, el Reglamento de la ley, promulgado mediante decreto número 24611-J, viene a aclarar el punto. De este texto extraemos estos preceptos relevantes:

*“Artículo 6.- Se presume, salvo prueba en contrario, que es productor del programa de cómputo, la persona natural o jurídica que publique la obra bajo su responsabilidad o que aparezca indicada como tal en la misma de la manera acostumbrada.”*

*“Artículo 7.- Salvo en los casos en que se trate de una obra individual, o que haya sido publicada con el nombre de los autores, se presume, salvo prueba en contrario, que el programa de cómputo es una obra colectiva, cuya titularidad corresponde, en los términos del artículo 6º de la Ley, al productor, quien además de los derechos de orden patrimonial, tiene la facultad de defender al derecho moral, en la medida que ello sea necesario para la explotación de la obra.”*

*“Artículo 8.- Salvo cuando se trate de una obra individual, o publicada bajo el nombre de sus autores, el derecho sobre el programa de cómputo se extingue, en los términos del artículo 60 de la Ley, a los cincuenta años de su primera publicación, contados, de acuerdo al artículo 65, a partir del 31 de diciembre del año en que se dio inicio a dicha publicación.”*

*“Artículo 9.- A menos que el contrato de enajenación del soporte material que contiene el programa de cómputo o la licencia de uso expedida por el titular del derecho, disponga otra cosa, es permitida al adquirente o licenciataria, según los casos, la reproducción de una sola copia de la obra, exclusivamente con fines de resguardo o seguridad, así como la introducción del programa en la memoria interna del equipo, a los únicos efectos de su utilización por el usuario.”*

---

<sup>48</sup> ERDOZAIN, José Carlos, *op. cit.* Y agrega: “Debe tenerse presente que el ámbito de protección alcanza por igual a esas tres realidades sin que ninguna de ellas deba ser considerada más relevante que las otras. Por consiguiente, cuando se está hablando de protección jurídica se ha de pensar en que la protección se irradia necesaria e indistintamente sobre todas ellas.”

*“Artículo 10.- No constituye modificación de la obra, la adaptación de un programa de cómputo realizada por el propio usuario del ejemplar legítimo y para su utilización exclusivamente personal, salvo que se contemple otra cosa en el contrato de enajenación de dicho soporte material o en la licencia de uso expedida por el titular del derecho sobre la obra.”*

Finalmente, en cuanto al artículo 103, inciso 5) de la ley, es importante señalar que el hecho de presentar el programa al Registro no tiene la consecuencia de convertir su contenido en información de dominio público. Así se desprende de la aplicación armónica del artículo 4, párrafo último, de la Ley de Información No Divulgada (N° 7975), que al efecto explícitamente advierte que “No se considerará que entra al dominio público la información confidencial que (...) haya sido proporcionada a cualquier autoridad por quien la posea, cuando la haya revelado para obtener licencias, permisos, autorizaciones, registros o cualquier otro acto de autoridad, por constituir un requisito formal. En todo caso, las autoridades o entidades correspondientes deberán guardar confidencialidad.”

## **Protección por medio de patentes de invención**

### ***Nociones generales***

Una **patente** es un derecho exclusivo otorgado al creador de una *invención*, término por el que se entiende un producto o proceso que ofrezca una nueva manera de hacer algo, o una nueva solución técnica a un problema.

Nuestra Ley sobre Patentes de Invención, Dibujos y Modelos Industriales (número 6867 de 25 de abril de 1983 y sus reformas –en especial, la que se efectuó mediante ley número 7979 de 22 de diciembre de 1999–), señala al efecto y en lo conducente:

### ***“Artículo 1.- Invenciones***

*1. Invención es toda creación del Intelecto humano, capaz de ser aplicada en la industria, que cumpla las condiciones de patentabilidad previstas en esta ley. Podrá ser un producto, una máquina, una herramienta o un procedimiento de fabricación y estará protegida por la patente de invención.*

2. *Para los efectos de esta ley no se considerarán invenciones:*

49

a) *Los descubrimientos, las teorías científicas, los métodos matemáticos y los programas de ordenador considerados aisladamente.*<sup>50</sup>

b) *Las creaciones puramente estéticas, las obras literarias y artísticas.*

c) *Los planes, principios o métodos económicos de publicidad o de negocios y los referidos a actividades puramente mentales, intelectuales o a materia de juego.*

d) *La yuxtaposición de invenciones conocidas o mezclas de productos conocidos, su variación de forma o uso, dimensiones o materiales, salvo que se trate de una combinación o fusión tal que no puedan funcionar separadamente o que las cualidades o funciones características de ellas sean modificadas para obtener un resultado industrial no obvio para un técnico en la materia.*

3. *Las obtenciones vegetales tendrán protección mediante una ley especial.”*

Ahora bien, para poder obtener la protección de una patente, una invención debe satisfacer ciertas condiciones:

- Debe tener un uso práctico. Nuestra ley se refiere a este elemento como “aplicación industrial”, entendiéndose por tal que el objeto de la invención pueda ser producido o utilizado en la industria, entendida en su más amplio sentido, incluyendo entre otros, la artesanía, la agricultura, la minería, la pesca y los servicios.
- Debe poseer un elemento de novedad, es decir, alguna característica que no se conozca en el cuerpo de conocimiento existente en su respectivo ámbito técnico. Este cuerpo de conocimiento existente se llama “estado de la técnica”, que en nuestro país comprende “todo lo

<sup>49</sup> El texto de la norma, luego de su más reciente reforma, indicaba “*Para los efectos de esta ley se considerarán invenciones ...*”; es decir, en sentido afirmativo. Esto produjo gran confusión, reflejada incluso en la primera edición de esta obra. Una Fe de Erratas posterior enmendó el entuerto como se indica.

<sup>50</sup> Las negrillas son nuestras.

divulgado o hecho accesible al público en cualquier lugar del mundo y por cualquier medio, antes de la fecha de presentación de la solicitud de patente en Costa Rica o, en su caso, antes de la fecha de prioridad aplicable. También quedará comprendido en el estado de la técnica el contenido de otra solicitud de patente en trámite ante el mismo Registro de la Propiedad Industrial, cuya fecha de presentación o, en su caso, la prioridad, sea anterior a la de la solicitud en consideración; pero solo en la medida en que este contenido quede incluido en la solicitud de fecha anterior cuando sea publicada. El estado de la técnica no comprenderá lo divulgado dentro del año anterior a la fecha de presentación de la solicitud en Costa Rica o, en su caso, dentro del año anterior a la prioridad aplicable, siempre que tal divulgación resulte, directa o indirectamente, de actos realizados por el propio inventor o su causahabiente o del incumplimiento de contrato o acto ilícito cometido contra alguno de ellos.”<sup>51</sup>

- La creación debe además poseer un nivel inventivo, que se entiende que existe –según nuestro ordenamiento– cuando “para una persona de nivel medio versada en la materia correspondiente, la invención no resulta obvia ni se deriva de manera evidente del estado de la técnica pertinente”.

En adición a lo señalado, la invención debe ser considerada como “patentable”. En nuestro país, se excluyen de la patentabilidad, por mandato legal: a) las invenciones cuya explotación comercial deba impedirse objetiva y necesariamente para proteger el orden público, la moralidad, la salud o la vida de las personas o los animales o para preservar los vegetales o evitar daños graves al ambiente; b) los métodos de diagnóstico, terapéuticos y quirúrgicos para el tratamiento de personas o animales; c) las plantas y los animales; d) los procedimientos esencialmente biológicos para la producción de plantas o animales.<sup>52</sup>

Con esas excepciones, todos los demás productos o procedimientos que cumplan los mencionados requisitos pueden ser patentados, “sin discriminación por lugar de la invención, campo de tecnología o porque los productos sean importados o producidos en el país”, según advierte el numeral 2 de la pluricitada ley.

La protección de una patente concretamente implica que la invención no puede ser confeccionada, utilizada, distribuida o vendida comercialmente sin el

---

<sup>51</sup> Artículo 2, inciso 3, de la Ley sobre Patentes de Invención, Dibujos y Modelos Industriales.

<sup>52</sup> Artículo 1, inciso 4, *ibídem*.

asentimiento de su titular. Es decir, la patente confiere al titular el derecho a explotar, en forma exclusiva, la invención y a conceder licencias a terceros para la explotación.

Al respecto, la ley especifica que la patente otorga a su beneficiario los siguientes derechos exclusivos:

*“a) Cuando la materia de la patente sea un producto, el de impedir que terceros, sin su consentimiento, realicen actos de fabricación uso, oferta para la venta, venta o importación, para esos fines, del producto objeto de la patente.*

*b) Cuando la materia de la patente sea un procedimiento, el de impedir que terceros, sin su consentimiento, realicen el acto de utilización del procedimiento y los actos de uso, oferta para la venta, venta o importación para esos fines de al menos el producto obtenido directamente mediante dicho procedimiento.”<sup>53</sup>*

La protección se concede durante un período limitado, que en nuestro medio es “de veinte años improrrogables, contados a partir de a fecha de presentación de la solicitud en su país de origen”.<sup>54</sup> Al cabo de este lapso, expira la patente y la invención pasa al dominio público. Es decir, cesan los derechos exclusivos del titular sobre la invención, que pasará entonces a estar disponible para la explotación comercial por parte de terceros.

Ahora bien,

*“Todos los titulares de patentes deben, a cambio de la protección de la patente, publicar información sobre su invención, a fin de enriquecer el cuerpo total de conocimiento técnico del mundo. Este creciente volumen de conocimiento público promueve una mayor creatividad e innovación en otras personas. Así pues, las patentes proporcionan no sólo protección para el titular sino asimismo información e inspiración valiosa para las futuras generaciones de investigadores e inventores.”<sup>55</sup>*

La inscripción registral de la patente es requisito *sine qua non* de la tutela jurídica. Es decir, el registro tiene eficacia constitutiva, a diferencia de lo que ocurre con los derechos de autor, como veremos más adelante.

---

<sup>53</sup> Artículo 16.

<sup>54</sup> Artículo 17.

<sup>55</sup> Del sitio web de la OMPI, <http://www.wipo.int>.

### **Aplicación al software**

Como se comentó en el acápite anterior, nuestra legislación no considera invenciones –y por ende patentables– a “los programas de ordenador considerados aisladamente”. Pensamos que este texto legal conduce a estimar que un programa de software sí es patentable cuando la inscripción persigue tutelarlos en conjunto con algún otro elemento –presumiblemente un dispositivo de hardware– al cual se encuentra íntimamente asociado.

No obstante, lo cierto es que en otros ordenamientos existe todo un movimiento que persigue admitir, sin cortapisas, la patentabilidad del software. Esta tendencia obedece a dos influencias concretas:

- Una decisión dividida que adoptó en 1980 la Corte Suprema de Justicia de los EE.UU., permitiendo el patentado de bacterias artificiales, producto de la ingeniería genética. En su razonamiento jurídico, la mayoría de la Corte adoptó la tesis de que “cualquier cosa bajo el sol que sea obra de la mano del hombre” debería considerarse patentable. Naturalmente, esto se interpretó como que se debería admitir la posibilidad de patentar el software, opción que hasta entonces se descartaba por considerar a los programas de computación como algoritmos matemáticos y no como verdaderos inventos.<sup>56</sup> Al año siguiente, en el caso “Diamond v. Diehr”, la Corte –de nuevo por mayoría– expresamente tuteló por primera vez la patentabilidad de una aplicación de software destinada a dirigir un proceso industrial. En julio de 1998, otro tribunal federal acogió la patente de un software de negocios.<sup>57</sup>
- La adopción en su momento del “Acuerdo sobre los Aspectos de los Derechos de Propiedad Intelectual relacionados con el Comercio” (ADPIC), al que nos referimos ya en la sección precedente.

En efecto, aunque –como ya se explicó– el ADPIC estrictamente remite en su ordinal 10 al Convenio de Berna para efectos de que el software y las bases de datos sean protegidos como “obras literarias”, en rigor tampoco prohíbe que se establezca una protección por medio de las patentes. Por el contrario, teniendo en consideración lo dispuesto en su artículo 27,

<sup>56</sup> REBACK, Gary L., “Patently absurd”. Publicación electrónica de la revista *Forbes*, disponible en <http://www.forbes.com/asap/2002/0624/044.html>, 24 de junio del 2002.

<sup>57</sup> McORMOND, Russell, “A review of software patent issues”. Documento electrónico disponible en <http://www.flora.ca/russell/drafts/software-patent2003.shtml>, 2003.



*“[el ADPIC] exige a los países signatarios que tengan un marco legal adecuado para el otorgamiento de patentes a todo tipo de invento, sea producto o proceso, en todos los campos de la tecnología, sin discriminación alguna, sujetos sólo al tradicional test de la novedad, inventiva y aplicabilidad industrial. Asimismo establece que los derechos otorgados por una patente deben ser disfrutados sin importar el lugar de invención o el hecho de que el producto sea importado o producido localmente.”<sup>58</sup>*

Por ende,

*“... al no incluir el Acuerdo una definición de ‘invención’, debe entenderse que el apartado I del art. 27 consagra el principio de patentabilidad absoluta, por lo que no existiría razón para excluir al ‘software’ del término ‘invención’, máxime cuando el artículo comprende ‘cualquier campo de la tecnología’.”<sup>59</sup>*

Sin embargo, hay que advertir que la tesis de la patentabilidad del software no es pacífica en doctrina. De hecho, es un desarrollo mucho más reciente que aquellas orientaciones que postulan la tutela del software a través del derecho autorial y los defensores de este último lineamiento critican con fuerza la tesis contrapuesta. Para ello, parten de recordar que la patente de invención, por definición, es aplicable a productos y procesos, no a ideas, procedimientos, métodos de operación o conceptos matemáticos en sí.

En efecto,

*“El obstáculo para la protección del software como invento patentable reside en su naturaleza de método o esquema mental compuesto por una serie de sucesivas etapas y operaciones que cristalizan en instrucciones dadas a un ordenador, sin elemento material o tangible (...) ni aplicación industrial inmediata.”<sup>60</sup>*

Por otra parte, se ha sostenido también que el software –por ejemplo, un juego para computadora– no necesariamente llena a plenitud los requisitos tradicionales de las invenciones patentables, cuales son: novedad, carácter inventivo y aplicabilidad industrial.<sup>61</sup> En adición, esta vía crearía más problemas de los que resuelve, puesto que –por ejemplo– en la medida en que diversos

<sup>58</sup> MARESCA, “¿Hacia el patentamiento del software?”, *op. cit.* Los paréntesis cuadrados no son del original.

<sup>59</sup> MARESCA, “Protección jurídica del software”, *op. cit.* ERDOZAIN (*op. cit.*) coincide.

<sup>60</sup> CORREA, *op. cit.*

paquetes de procesamiento de textos realizan las mismas tareas esenciales, distinguiéndose más por su aspecto visual que por su funcionalidad interna, el patentamiento de una aplicación tal solamente vendría a bloquear la diversidad de ofertas en este terreno.<sup>62</sup>

ANTEQUERA PARILLI<sup>63</sup> resume la problemática de este modo:

*“1. El programa de computación, a diferencia de las invenciones industriales, no aporta ningún cambio a la materia ni a la energía en el mundo físico.*

*2. El software no es un aparato (como el hardware), ni tampoco un proceso industrial.*

*3. El derecho industrial dirige la protección al contenido mismo de la invención, mientras la tutela del software está referida a la forma de expresión, propia del derecho de autor.*

*4. Si bien los programas de computación tienen características de originalidad, en su sentido de individualidad (óptica del derecho de autor), muy pocos de ellos revisten novedad en el sentido del derecho invencional”.*

Otras desventajas de aplicar un mecanismo de patentes en este tema, según interpretamos los planteamientos de FUENTES PINZÓN<sup>64</sup>, serían las siguientes:

- El menor plazo de la protección de la patente (improrrogable, además), con relación al que brinda el derecho autoral.<sup>65</sup>
- La territorialidad de la patente, que obliga al titular de la invención a contar con una en cada país en que desee obtener tutela.<sup>66</sup>

<sup>61</sup> FUENTES PINZÓN, Fernando. “¿Por medio de cual rama jurídica debe protegerse al software?”. En *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 37, agosto del 2001.

<sup>62</sup> *Idem.*

<sup>63</sup> Citado por FUENTES, *op. cit.*

<sup>64</sup> *Idem.*

<sup>65</sup> No obstante, un sector de autores opina que, de toda suerte, los 20 años de protección que otorga la patente constituyen de por sí excesivamente largo tratándose de software, demorando la incorporación de la tecnología correspondiente en nuevos productos.

<sup>66</sup> Objeción que, sin embargo, pierde fuerza con el impulso que la OMPI viene dando al desarrollo de un sistema eficaz para el registro internacional de patentes. El primer paso consistió en la

- El carácter constitutivo del derecho derivado de la patente, frente al carácter meramente declarativo que tiene la inscripción de una obra. De aquí derivaría además la necesidad de garantizar el secreto de un nuevo programa de cómputo hasta que se solicite la patente, ya que de lo contrario, podría perderse la posibilidad de inscripción por ausencia de novedad.
- La patente exige acreditar que el producto supera el “estado de la técnica”, requisito que –como se explicó arriba– no satisfarían aplicaciones informáticas similares a otras previamente inscritas.
- Un invento patentable necesariamente debe aportar una solución o ventaja a la técnica o a la industria. Sin embargo, programas como los juegos de computación no aportan soluciones a problemas técnicos o industriales y, por consiguiente, quedarían excluidos de protección.

Sin embargo, el mismo autor nos advierte que la inconveniencia de patentar al software no debe llevarnos a excluir la protección que se puede otorgar por esta vía a diversas invenciones relacionadas indirectamente con el software. De esta manera, explica que la Comunidad Europea –por medio de la Oficina Europea de Patentes– ha establecido una serie de premisas, tales como las siguientes:

*“3.1. Un programa de computación en sí mismo (conjunto de instrucciones, por ejemplo en forma de un programa listado) si se reivindica por sí mismo, o como una grabación en un medio de soporte o cuando se carga en un computador conocido, no es patentable (...)*

*3.2. Un método o proceso para operar un computador de acuerdo con un programa determinado podría ser patentado, si puede demostrarse un efecto técnico (...)*

*3.3. Un computador conocido, si se programa de manera determinada, se considera patentable en tanto pueda identificarse un efecto técnico (...)*

---

promulgación de un “Tratado de cooperación en patentes”, adoptado en 1970 (y aprobado por nuestro país mediante ley N° 7836 del 22 de octubre de 1998), que permite solicitar protección para una invención simultáneamente en un gran número de países mediante la presentación de una “solicitud internacional”. Pueden presentar esa solicitud los nacionales o residentes de un Estado contratante. El Tratado reglamenta los requisitos formales que debe satisfacer toda solicitud internacional. Fuente: boletín electrónico “Actualidad” de la OMPI, N° 2004/216 del 16 de enero del 2004.

3.4. *Una fabricación controlada por programa o un proceso de control es patentable (...)*

3.5. *Un artículo de fabricación que comprenda un microprocesador o memoria en la que se encuentra almacenado un programa para llevar a cabo funciones específicas es patentable (...)*<sup>67</sup>.

Como se nota, estos lineamientos coinciden con la interpretación que podemos ofrecer al artículo 1 de la Ley costarricense y a la que ya se hizo alusión arriba.<sup>68</sup>

La discusión a favor o en contra del recurso a las patentes para proteger el software no se ha acabado. Por ejemplo, la *Patent and Trademark Office* de los EE.UU. anunció en 1995 que no rechazaría más solicitudes de patente por el sólo hecho de que se refieran a programas de computación, sino que simplemente exigiría que cumpliesen con los requisitos tradicionales. Se estudiará únicamente si el invento es útil a las artes tecnológicas en el “mundo real”, por oposición a algo que no representa más que una idea o concepto, lo cual no sería patentable.<sup>69</sup> Esta postura, no obstante, ha sido fuertemente criticada, sobre la base de que un exceso en el otorgamiento de patentes es tanto o más dañino aun que una inopia de ellas, dado que el componente monopolístico de la patente, en lo que se refiere a la exclusividad que ella otorga, eventualmente actúa como un obstáculo a la innovación y a la libre competencia.<sup>70</sup> Por ejemplo, por medio de una patente, sería posible impedir la incorporación de un determinado algoritmo (digamos, para compresión de imágenes) en productos competidores. Desde esta perspectiva, para el desarrollador de software deseoso de obtener la máxima rentabilidad de su producto y de la inversión que éste supone, no cabe duda de que el sistema de patentes ofrece claras ventajas respecto del de derechos de autor.<sup>71</sup>

De hecho, muchas empresas –conscientes de los privilegios competitivos que obtendrían– ahora procuran incentivar la generación de nuevas patentes por

<sup>67</sup> FUENTES PINZÓN, *op. cit.*

<sup>68</sup> Al momento de escribir estas líneas, nuestro país ha concluido la etapa de negociación de un Tratado de Libre Comercio (TLC) con los EE.UU. No obstante, aun no hay una redacción final del documento y, desde luego, no ha sido firmado ni aprobado por la Asamblea Legislativa. El capítulo 15 del Tratado está dedicado a temas de propiedad intelectual y, en concreto, la cláusula 15.9 del texto preliminar alude al compromiso que adquieren las partes de permitir la obtención de patentes “en todos los campos de la tecnología”. No está claro aun si esto conducirá a una reforma tácita o expresa de la legislación vigente en la materia de interés.

<sup>69</sup> MARESCA, “Hacia el patentamiento del software”, *op. cit.*

<sup>70</sup> REBACK, *op. cit.*

<sup>71</sup> McORMOND, *op. cit.*

medio de ofrecer a sus ingenieros una serie de beneficios tales como una porción de las regalías, acciones de la compañía o dinero efectivo; y realizan reuniones regulares de “lluvia de ideas” (“*brainstorming*”) para fomentar las ideas patentables.<sup>72</sup> También se ha comentado acerca del surgimiento de una nueva práctica viciada, de dimensiones notoriamente similares al conocido problema de la *ciberocupación* en el mundo de los nombres de dominio, que consiste en la actitud de ciertos empresarios que patentan calladamente determinadas tecnologías y luego esperan a que éstas sean incorporadas en nuevos productos, tan solo para demandar entonces cuantiosas sumas a cambio.<sup>73</sup> O considérese, en este sentido, la siguiente recomendación:

*“[Los gerentes de tecnología de la información] deberían desarrollar (o adquirir) un portafolios defensivo de patentes de software, para utilizarlo como herramienta de negociación cuando fueren confrontados con las patentes de otra empresa. Al ofrecer el licenciamiento de sus patentes –o por medio de la amenaza de reivindicarlas– se podría evitar tener que ceder a sus exigencias o pelear una larga y costosa batalla legal.”<sup>74</sup>*

Sobre este tema, la postura europea diverge claramente de la estadounidense. Como se explicó, las propuestas de la Comisión Europea se orientan hacia otorgar patentes solo a las aplicaciones de software que supongan un verdadero avance tecnológico (como podrían serlo los programas que cambien el funcionamiento de un aparato o los que hicieran trabajar a un dispositivo mecánico). No sería posible entonces conferir patentes de métodos de negocios como, por ejemplo, la que se registró en 1999 a favor de la empresa Amazon.com con respecto a su sistema “*one-click*” de compras por Internet. La postura europea busca situarse entonces en un punto medio entre la postura de los EE.UU. –que se considera demasiado liberal– y la de los partidarios del software *open source*, quienes se oponen a patentar cualquier producto de software, en la medida en que los términos de la patente resultan por lo general incompatibles con los esquemas de licenciamiento del software libre.<sup>75</sup> Se argumenta, en este sentido, que la

<sup>72</sup> TAM, Pui-Wing. “More Patents, Please!”, artículo publicado en el periódico *Wall Street Journal*, edición del 3 de octubre del 2002, pág. B1.

<sup>73</sup> KIM, Jonathan, “Patenting air or protecting property?”. Publicación electrónica del diario *Washington Post*, 11 de diciembre del 2003; disponible electrónicamente en <http://www.washingtonpost.com/ac2/wp-dyn?pagename=article&node=&contentId=A54548-2003Dec10&notFound=true>

<sup>74</sup> RADCLIFFE, Mark. “Patents: A Small Price to Pay for Progress”. En *CIO Magazine*, 1 de agosto del 2003, disponible electrónicamente en [http://www.cio.com/archive/080103/debate\\_pro.html](http://www.cio.com/archive/080103/debate_pro.html)

<sup>75</sup> Al respecto, véase el capítulo previo sobre clasificación del software.

existencia misma del movimiento *open source* demuestra la viabilidad de incentivar la innovación tecnológica dentro del marco de la protección del derecho autoral.

Al respecto, tendremos que esperar el trámite que se dé a la “Propuesta de directiva del Parlamento Europeo y del Consejo sobre la patentabilidad de las invenciones implementadas en computadora”, aprobada en Bruselas el 20 de febrero del 2002.<sup>76</sup>

En nuestro país, la suscripción de algunos instrumentos internacionales bilaterales y multilaterales, tales como el Tratado de Libre Comercio con los EE.UU.<sup>77</sup> o el Área de Libre Comercio de las Américas, también podrían venir a afectar el actual panorama normativo.

## Regímenes de protección *sui generis*

Diversas corrientes doctrinarias vienen afirmando con insistencia que así como la protección del software vía patente de invención es inidónea, también lo es la que se pretenda lograr a través del derecho autoral. Si se le considera cuidadosamente, el software es difícilmente adaptable a lo que en rigor se comprende bajo las leyes de derecho de autor, que han sido concebidas teniendo en mente las tradicionales obras literarias y artísticas, cuya naturaleza es muy diversa. Según estos puntos de vista, la aplicación al software de los conceptos de la propiedad intelectual clásica deviene necesariamente forzada y obliga a la introducción de una serie de excepciones y variantes puramente *ad hoc*, de las que ya hemos hablado más arriba.

En palabras de ERDOZAIN:

*“... la decisión de proteger un programa de ordenador como una obra intelectual se ha visto a la postre como un proceso llevado con rapidez y poca reflexión, acaso obligado por las circunstancias. Una de las razones que ha alimentado la crisis conceptual, no ha sido otra que la dificultad en considerar un programa de ordenador como una expresión formal de la que predicar una originalidad y su consiguiente división en facultades patrimoniales y morales. Como veremos (...), es dogmáticamente complicado establecer los criterios para considerar que un programa informático es original en el sentido del Derecho de autor; al mismo tiempo, la división entre facultades morales y patrimoniales da origen a dificultades prácticas insalvables, siendo las*

<sup>76</sup> TÉLLEZ, *op. cit.* Una decisión definitiva sobre esta propuesta ha sido reiteradamente postergada, precisamente a cuenta de su carácter controvertido.

<sup>77</sup> Cuyo texto final no se conoce aun a la fecha de redacción de estas líneas.

*primeras inalienables e irrenunciables, y estando expuesto el titular de derechos distinto al autor a una constante prohibición de modificación sensu stricto de la obra, consecuencia ésta que en el ámbito de los programas de ordenador es algo claramente impracticable.”<sup>78</sup>*

En principio, esta línea de razonamiento fue adoptada por la propia Organización Mundial de la Propiedad Intelectual (OMPI), cuando formuló en 1976 las “Disposiciones Tipo para la Protección del Soporte Lógico”.<sup>79</sup> Posteriormente, en algunos países (Japón, Brasil y Argentina entre otros) se ha propuesto también la necesidad de crear mecanismos alternativos de protección jurídica para el software. Hasta adonde sabemos, sin embargo, ninguno ha recibido una aceptación general o han posteriormente desechados. En palabras de FUENTES PINZÓN:

*“... sólo a nivel doctrinario, esta posición aun conserva defensores, aunque estos no realizan una elaboración sobre el tipo de protección que debería de imponerse, sino que se limitan a criticar a la protección por el Derecho de Autor y por las Patentes.”*

Las objeciones que han sido planteadas al lineamiento doctrinario de comentario han sido resumidas por el mismo autor así:

*“a) El destiempo. Esta posición no tiene justificación alguna, ya que la protección del software por medio del derecho de autor, es la solución acogida por casi la totalidad de las legislaciones del mundo.*

*b) Contribuye a la dispersión legislativa. El hecho de promulgar una ley especial, que contemple un régimen particular para la protección del software, no hace otra cosa que empujar el camino para que los productores de los programas de computación puedan disfrutar de sus derechos, ya que esta “ley” deberá de discutirse y aprobarse, además del tiempo que se requiere para su correcta implementación, por lo cual quedarían desprotegidas las casas productoras de ejercer algún derecho en el interin de este proceso (salvo las previstas por competencia desleal y por hecho ilícito).*

*c) La desprotección internacional. Contemplar un régimen paralelo a los existentes, significaría tener que elaborar Convenios y Tratados, tanto bilaterales como multilaterales, para poder otorgar una*

---

<sup>78</sup> *Idem.*

<sup>79</sup> FUENTES PINZÓN, *op. cit.* En el mismo sentido, TÉLLEZ VALDÉS, *op. cit.*

*protección medianamente uniforme del software y permitir su distribución internacional.*

*d) La carencia de objeto. Los autores que defiende esta tendencia, no establecen el verdadero motivo por el cual el software no puede ser considerado como una obra, sino que se limitan a mencionar que tiene ‘características particulares’, pero sin señalarlas, por lo cual a esta posición le falta un objeto específico que la mantenga. Tampoco establecen cuales deberían ser los elementos protegibles, ni las limitaciones, ni los derechos de los titulares y de los usuarios, la duración de la protección, el procedimiento correcto para el otorgamiento del título, tampoco si este título será constitutivo o mero declarativo de derechos, entre otras cuestiones que no llegan a plasmar, por lo que podemos sostener que esta posición doctrinaria es carente de un desarrollo real.*

*e) La falta de ventajas. No existe ventaja alguna para considerar como loable la promulgación de un sistema particular para la protección del software.*

*f) El desconocimiento sobre derecho de autor. Las críticas más comunes que le hacían los sostenedores de esta posición a la protección por el derecho de autor, evidencia que estos no tenían un manejo adecuado sobre la materia, en virtud de desconocer: sus principios básicos, el régimen para las obras complejas y las realizadas en colaboración, el concepto de obra intelectual y los derechos patrimoniales otorgados a favor de los titulares.”<sup>80</sup>*

No obstante estas críticas, la idea de la necesidad de un régimen especial no pierde su atractivo, a pesar de que algunos autores reconocen que esta solución ideal es difícilmente practicable:

*“El software no calza cómodamente en las categorías legales tradicionales de las obras funcionales ni de las obras expresivas. El derecho autoral, luego el de patentes y ahora el de la [libertad de expresión] han sido forzados hasta tornarse irreconocibles con el fin de acomodar esta tecnología. Quizás la más razonable solución a esta dificultad sería la de colocar al software dentro de su propia categoría –no la de las patentes, el derecho de autor y ni siquiera la de la libre expresión–, una especialmente diseñada para coincidir con las características de este inusual híbrido legal. Desafortunadamente,*

---

<sup>80</sup> *Idem.*



*probablemente sea demasiado tarde para devolver el reloj y tomar un camino tan razonable; así es que anticipamos que las instituciones jurídicas continuarán clasificando como ‘funcional’ o como ‘expresivo’ a una tecnología que podría ser ambas o ninguna.”<sup>81</sup>*

## Otros medios de tutela

En adición a los medios descritos en los apartes que preceden, otros comentaristas han señalado la existencia de diversos mecanismos complementarios que acuden a la tutela jurídica del software, incluyendo las regulaciones constitucionales sobre libertad de expresión, así como las que aplican al terreno comercial de los modelos y secretos industriales.<sup>82</sup>

### ***El software como expresión constitucionalmente protegida***

Las libertades de expresión, información y pensamiento se entrelazan en uno de los ciclos primordiales de la libertad: la expresión de unos crea la información que reciben otros, contribuyendo a moldear un pensamiento libre que, a su vez, tendrán la posibilidad de transmitir a otros más.

El derecho a la libre expresión “(se) traduce fundamentalmente en el derecho de dar a conocer el propio pensamiento a otros hombres”<sup>83</sup> y no se constriñe a la difusión de las ideas o pensamientos en forma de prosa escrita o hablada. En general, cualquier forma de transmisión de ideas de contenido expresivo –es decir, cuyo propósito sea la difusión de un pensamiento, opinión o recomendación a otras personas– es susceptible de obtener tutela en este contexto, independientemente de la manera en que se plasmen. Ya sea que nos refiramos a palabras, sonidos, señales o imágenes en general, la aprobación o desaprobación que nos produzcan o la valoración estética que puedan provocar no se tiene en cuenta para efectos de garantizar que gozarán de la protección constitucional.

Tal como lo señaló nuestra Sala Constitucional en una sentencia de 1991,

*“(El) artículo 29 de la Carta Suprema (...) se refiere al contenido de los pensamientos, ideas y opiniones y no a proveer un*

<sup>81</sup> BURK, Dan L. “Copyrightable functions and patentable speech”, en *Communications of the ACM*, volumen 44, número 2, febrero del 2001, página 69. Traducción libre del autor.

<sup>82</sup> MANGASSARIAN, Elina, “Copyright Protection of Software”. Documento electrónico disponible en <http://mishpat.net/articles/elina.doc>. Octubre del 2000.

<sup>83</sup> HERNÁNDEZ VALLE, Rubén. *Constitución Política de la República de Costa Rica. Comentada y Anotada*. Editorial Juricentro, primera edición, San José, 1998.

*determinado medio –que puede ser diverso– a través del cual las mismas pueden ser difundidas.”<sup>84</sup>*

Entonces, ¿una aplicación informática podría considerarse como una modalidad de manifestación de pensamientos e ideas, tal que la difusión que de ella haga su autor pueda gozar de tutela bajo el derecho de libre expresión?

La respuesta doctrinaria unánime es afirmativa, aunque únicamente con relación al software que está expresado en código fuente o incluso como pseudo código.<sup>85</sup> En efecto, cabe recordar, en primer término, que nuestro ordenamiento considera al software como una obra, por lo que –desde esa óptica– su comunicación a otros por cualquier medio debería recibir el mismo tratamiento jurídico que si se tratara de un libro o de una composición musical.

Pero incluso podemos hurgar más hondo: un programa informático siempre plasma uno o más *algoritmos*, que son secuencias de pasos que conducen a una solución o resultado útil. Desde esta óptica, la divulgación de un programa de software puede servir para que, por esa vía, el autor comunique a otras personas el mecanismo lógico mental por medio del cual el programa que creó logra arribar a dicha solución o resultado.

La cuestión podría parecer puramente académica, pero no lo es. De hecho, los tribunales estadounidenses ya han conocido de múltiples casos en que el punto cobra capital importancia. Por ejemplo, al establecer restricciones a la exportación de software para la encriptación de mensajes de correo electrónico, el gobierno de ese país ha sido acusado judicialmente de limitar la libre expresión. Del mismo modo, al sancionar la difusión de otro programa de cómputo cuyo resultado es desactivar la protección contra el copiado de los DVD (tema al que nos referiremos de nuevo más abajo), se ha aducido la inconstitucionalidad de la legislación de EE.UU. que así lo prohíbe, por el mismo motivo.<sup>86</sup>

No obstante, es vital recordar que la libertad de expresión –como cualquier otro derecho fundamental– no es absoluta e irrestricta. Por medio de la ley, es posible imponerle algunas limitaciones necesarias para asegurar a los terceros el legítimo disfrute de sus derechos. En este caso, la posibilidad de divulgar el código

<sup>84</sup> N° 2109-91 de las 9:20 horas del 18 de octubre de 1991. Los paréntesis no son del original.

<sup>85</sup> Cuando la lógica de un programa es expuesta empleando un lenguaje natural, se dice que está expresada en pseudo código (ejemplo: “Si  $a + b$  es mayor que 100, entonces multiplique el resultado por 3”). Por su parte, el acuerdo no es pacífico con respecto al código objeto, ya que, por su propia naturaleza, algunos le niegan contenido expresivo e idoneidad para fines comunicativos.

<sup>86</sup> BURK, Dan L. *Op. cit.* Véase también TOURETZKY, David S. “Free speech rights for programmers”, en *Communications of the ACM*. Volumen 44, número 8, agosto del 2001, página 23.

de una aplicación informática, bajo la protección del derecho a la libre expresión, podría entrar en conflicto con institutos jurídicos tales como el de la protección de los secretos empresariales: la llamada “información no divulgada”.

Le dedicamos seguidamente nuestra atención.

### **Secretos comerciales**

El concepto de “secreto comercial” se aplica a cualquier información confidencial que confiera o pueda conferir a una empresa una ventaja competitiva. Se comprende entonces, correlativamente, que el acceso y utilización no autorizada de dicha información puede causar un enorme perjuicio a la empresa afectada.

Diversas clases de información son susceptibles de ser consideradas secretos comerciales, incluyendo los métodos de venta y de distribución, perfiles del consumidor tipo, estrategias de publicidad, listas de proveedores y clientes y procesos de fabricación.<sup>87</sup>

En Costa Rica, las principales disposiciones sobre esta materia constan en la “Ley de Información No Divulgada”, número 7975 de 4 de enero del 2000, que protege

*“... la información no divulgada referente a los secretos comerciales e industriales que guarde, con carácter confidencial, una persona física o jurídica para impedir que información legítimamente bajo su control sea divulgada a terceros, adquirida o utilizada sin su consentimiento por terceros, de manera contraria a los usos comerciales honestos, siempre y cuando dicha información se ajuste a lo siguiente:*

*a) Sea secreta, en el sentido de que no sea, como cuerpo ni en la configuración y reunión precisas de sus componentes, generalmente conocida ni fácilmente accesible para las personas introducidas en los círculos donde normalmente se utiliza este tipo de información.*

---

<sup>87</sup> OMPI. “¿Cómo proteger los secretos industriales de su PYME?” Documento electrónico disponible en [http://www.wipo.int/sme/es/ip\\_business/trade\\_secrets/trade\\_secrets.htm](http://www.wipo.int/sme/es/ip_business/trade_secrets/trade_secrets.htm). Visto el 13 de marzo del 2002.

*b) Esté legalmente bajo el control de una persona que haya adoptado medidas razonables y proporcionales para mantenerla secreta.*

*c) Tenga un valor comercial por su carácter de secreta.*

*La información no divulgada se refiere, en especial, a la naturaleza, las características o finalidades de los productos y los métodos o procesos de producción.*

*(...)*

*La información que se considere como no divulgada deberá constar en documentos, medios electrónicos o magnéticos, discos ópticos, microfilmes, películas u otros elementos similares.”<sup>88</sup>*

Las sanciones penales aplicables por la infracción a estas disposiciones constan en la “Ley de Procedimientos de Observancia de los Derechos de Propiedad Intelectual”, que al respecto señala:

*“Artículo 49.- Divulgación de secretos comerciales o industriales*

*Será sancionado con prisión de uno a tres años quien divulgue, sin autorización del titular de secretos comerciales o industriales, información confidencial conocida por razón de su oficio, empleo, relación contractual o profesión, de modo que pueda causar perjuicio al titular.*

*Para los efectos del presente artículo, así como de los subsiguientes de esta sección, se utilizarán, para fines de interpretación, los conceptos de secreto comercial o industrial e información no divulgada, contenidos en la Ley de Información no divulgada, N° 7975, de 4 de enero de 2000.*

*Si un funcionario público es quien revela la información no divulgada, se le impondrá, además de la pena privativa de libertad, inhabilitación de uno a dos años para el ejercicio de cargos y oficios públicos.*

---

<sup>88</sup> Artículo 2.

*Artículo 50.- Obtención de información no divulgada por medios ilícitos*

*Será sancionado con prisión de uno a tres años quien, por medios ilícitos o desleales, obtenga información no divulgada.”*

El código fuente de una aplicación informática ha sido consistentemente considerado como parte de los secretos comerciales (es decir, la información no divulgada) de la respectiva empresa desarrolladora.<sup>89</sup> Esta perspectiva resulta provechosa, por ejemplo, para proteger a la empresa de eventuales acciones maliciosas por parte de sus competidores o bien incluso de los propios empleados, señalando la responsabilidad tanto civil como penal que cabría en tales casos. No obstante –como se explicará más adelante cuando analicemos el tema de los contratos de licenciamiento de software– esta faceta debe ser comprendida y aplicada en su correcta dimensión, evitando darle al régimen de información no divulgada una extensión mayor a la que realmente posee, en detrimento incluso del interés general.

El tema es relevante no solo desde la óptica del valor comercial del software como artículo de comercialización, sino precisamente en cuanto algunos autores han argumentado que, en definitiva, el valor que posee una determinada aplicación no está en ella misma, sino en el conocimiento que almacena. Es decir, bajo esta perspectiva, sería erróneo considerar al software meramente como un *producto*, ya que en realidad es solo un *medio* para almacenar y aplicar conocimiento.<sup>90</sup> En efecto, es sabido que las aplicaciones de negocios generalmente traducen reglas y procesos aplicados en el ámbito de la organización y, en ese tanto, actúan como repositorios de un conocimiento que es necesario conservar y proteger. Por ende, aun cuando el giro comercial de la empresa en cuestión no incluya propiamente el desarrollo y venta de software, los programas que utilice internamente –en especial si han sido desarrollados a la medida– tienen la virtud de retratar esos procesos de modo tal que su eventual divulgación no autorizada podría representar un sensible desmedro.

No obstante lo dicho, la protección que otorga el régimen de información no divulgada es comparativamente más débil que la que brindan los de patentes y de derechos de autor. En efecto,

<sup>89</sup> JURAN, Belinda y otros. “Trade Secret Protection for Source Code”, en *Internet Alert*. Revista electrónica publicada por Hale and Dorr, LLP (<http://www.haledorr.com>). Edición de 17 de julio del 2001.

<sup>90</sup> “... el valor del software no es el código, sino lo que el código hace. O, mejor, el conocimiento que el código encierra.” ARMOUR, Phillip G. “The case for a new business model”, en *Communications of the ACM*, volumen 43, número 8, agosto del 2000, página 20. Traducción del autor.

*“... los secretos comerciales son inherentemente fugaces, porque son susceptibles a la ingeniería reversa, el descubrimiento independiente o la revelación accidental, todos los cuales la legislación de información no divulgada considera como mecanismos legítimos para la obtención de secretos.”<sup>91</sup>*

Por ende, ante la eventual interrogante de si proteger una aplicación de software bajo la legislación de secretos comerciales, la de patentes (si satisface los criterios necesarios) o la de derechos de autor, sería necesario considerar las ventajas y desventajas relativas de cada sistema, que podríamos resumir así:<sup>92</sup>

Ventajas de los secretos comerciales:

- La protección no prescribe. Es decir, la tutela que confieren los secretos comerciales se prolonga indefinidamente, siempre que no sean divulgados al público.
- No se incurre en costos de registro (no obstante, el esfuerzo requerido para mantener confidencial la información ya podría, de por sí, requerir de alguna inversión más o menos considerable).
- La protección surge de pleno derecho.
- Para que opere la tutela, al no ser requerido un proceso de inscripción registral, no es necesario divulgar la información a una dependencia oficial.

Desventajas:

- Como ya se explicó, al plasmarse el secreto en un producto concreto, éste podrá ser inspeccionado y sometido a la ingeniería reversa por terceros que podrán, entonces, descubrirlo y utilizarlo. En este sentido, es oportuno reiterar que la protección por secreto comercial de una invención no confiere derechos exclusivos que impidan a terceros utilizarla comercialmente. Este tipo de protección es propio solamente de las patentes y los modelos de utilidad.

<sup>91</sup> SAMUELSON, Pamela. “Trade secrets vs. free speech”, en *Communications of the ACM*, volumen 46, número 6, junio del 2003, página 22. Traducción del autor.

<sup>92</sup> La enumeración proviene del documento “¿Patentes o secretos comerciales”, incluido en la publicación electrónica “La propiedad intelectual para las pequeñas y medianas empresas”, distribuido en formato de CD-ROM por la OMPI, 2003.

- Una vez que el secreto se divulga, todos tienen acceso a él y pueden utilizarlo como deseen, incluso patentándolo por medio de una invención que lo incorpore.
- Por lo general, es más difícil hacer respetar un secreto comercial que una patente.

### ***Dibujos y modelos industriales***

Los dibujos o modelos industriales aluden al aspecto ornamental o estético de un producto. Pueden consistir en rasgos tridimensionales, como la forma o la superficie de un artículo, o bien rasgos bidimensionales, como los diseños, las líneas o el color. Para recibir protección, un dibujo o modelo industrial no debe ser funcional. Lo relevante a estos efectos es el carácter estético del artículo, en vez de sus rasgos técnicos.<sup>93</sup>

Conforme a nuestra “Ley sobre Patentes de Invención, Dibujos y Modelos Industriales”, ya mencionada anteriormente, el titular de un dibujo o modelo industrial protegido tiene derecho a impedir que, sin su consentimiento, terceros fabriquen, vendan o importen artículos que ostenten o incorporen un dibujo o modelo que sea una copia o fundamentalmente una copia del dibujo o modelo protegido, cuando esos actos se realicen con fines comerciales.<sup>94</sup>

Ahora bien, a partir del desarrollo primero del sistema operativo de las computadoras Macintosh de la empresa Apple<sup>95</sup> y luego del Windows de Microsoft, la informática moderna ingresó de lleno en la era de las interfaces gráficas de usuario (GUI, por sus siglas en inglés). Como su nombre lo indica, una GUI consta de un conjunto de elementos gráficos que sirven de intermediario entre el usuario y las complejas instrucciones que determinan el funcionamiento de una computadora. De este modo, por ejemplo, es posible imprimir un documento pulsando un icono gráfico que representa a una impresora, en vez de tener que teclear secuencias de comandos, como se hacía hasta entonces.

Este avance fundamental dio pie a numerosas batallas en torno a la protección de los elementos gráficos del software. La más famosa de ellas se dio precisamente entre Apple y Microsoft, con resultados negativos para la primera,

<sup>93</sup> OMPI. “¿Qué es un dibujo o modelo industrial?”. Documento electrónico disponible en la dirección [http://www.wipo.int/about-ip/es/index.html?wipo\\_content\\_frame=/about-ip/es/industrial\\_designs.html](http://www.wipo.int/about-ip/es/index.html?wipo_content_frame=/about-ip/es/industrial_designs.html). Visitada el 13 de marzo del 2002.

<sup>94</sup> Artículo 25, inciso 4.

<sup>95</sup> Que a su vez se basa en el trabajo que la empresa Xerox venía desarrollando en su centro de investigaciones de Palo Alto, en California.

que pretendía obtener tutela frente a la segunda para lo que desde entonces se ha dado en llamar el *look and feel* (apariencia y sensación) de su software. En efecto,

*“... los programas que posean presentaciones en pantalla originales podrían recibir alguna protección legal contra el copiado de su ‘apariencia y sensación’ (‘look and feel’). [Sin embargo,] este campo está siendo arduamente disputado por las grandes empresas de la industria de la computación personal, de modo que la amplitud de los derechos sobre esa ‘apariencia y sensación’ no podría predecirse con certeza en estos momentos.”<sup>96</sup>*

Se ha dicho, con toda razón, que las dificultades que presenta esta temática estriban en qué debe entenderse exactamente por la “apariencia y sensación” de un software; qué tanta similitud debe haber entre la interface de una aplicación y otra para que se considere que existe una infracción y cómo superar los aspectos puramente subjetivos en la comparación de un programa con otro u otros.

## Otros temas puntuales relacionados

### ***Tutela de la propiedad intelectual de las bases de datos***

Durante la década pasada, surgió la discusión jurídica en torno a si una base de datos es susceptible o no de recibir protección bajo el derecho autoral, en forma separada de la que goza la aplicación empleada para gestionarla. En esta sección discutimos las razones por las que la respuesta debe ser afirmativa, con las condiciones y restricciones que también señalamos.<sup>97</sup>

Gran parte de la gestión informática en las empresas y organizaciones modernas gira hoy en torno a las bases de datos. En tiempos en que la información ha llegado a convertirse en uno de los activos más preciados, la tecnología de bases de datos se viene aplicando a la solución de problemas tanto viejos (manejo de inventarios, contabilidad, planillas, administración de recursos humanos, etcétera) como novedosos (en especial, en el campo del comercio electrónico).

Compilar información, verificarla, darle un formato apropiado y mantenerla al día puede ser una labor sumamente onerosa. Existe todo un mercado creado en torno a suplir esa información a quien la necesite. Diversas personas y empresas

<sup>96</sup> ROSE, Lance. *“Netlaw: your rights in the online world”*. Osborne McGraw-Hill. 1ª edición, 1995, página 100. Traducción libre del autor.

<sup>97</sup> HESS ARAYA, Christian. “Propiedad intelectual de las bases de datos”, en *Revista Electrónica de Derecho Informático* (<http://www.alfa-redi.org>), número 28, noviembre del 2000.



están dispuestas a pagar por un acceso oportuno a información de calidad. Pero es claro que muchos proveedores no estarían dispuestos a ingresar a esta importante y necesaria actividad, si no hubiera algún modo de proteger jurídicamente sus productos y servicios, ya que de lo contrario no habría modo de asegurar una adecuada recuperación de los costos involucrados.

Antes de avanzar más con el tratamiento del tema que nos ocupa, interesa aclarar algunas nociones técnicas fundamentales, en beneficio de los lectores sin formación específica en este campo. En particular, debe precisarse la distinción que existe entre *bases de datos* y *sistemas administradores de bases de datos*.

En síntesis:

- Una **base de datos** es una colección sistemática, estructurada, de datos (y a veces también de procedimientos asociados a ellos), almacenados electrónicamente y relativos a personas, objetos, eventos, etcétera.
- Un **sistema administrador de bases de datos** (en lo sucesivo “SABD”), por su parte, es el software de aplicación que se encarga de gestionar una base de datos, para incluir, modificar, suprimir o recuperar la información en ella contenida.

Formulemos una simple analogía para una mejor comprensión de la diferencia entre ambos conceptos:

Piénsese, en efecto, en una actividad rutinaria como ir a buscar un libro a una biblioteca. En este caso, posiblemente debamos ir hasta un mostrador, adonde indicaremos a una persona –el bibliotecario– cuál es el libro deseado. Esa persona irá hasta el sitio adonde se almacenan los libros, localizará el título solicitado y –asumiendo que esté disponible– nos lo presentará en el mostrador. Los aspectos destacables de esta sencilla transacción incluyen los siguientes (todos obvios, pero permítasenos aun así ponerlos de relieve para lo que sigue más abajo):

a.- El bibliotecario es capaz de encontrar la obra en cuestión, entre posiblemente varias centenas o miles de otros títulos, porque éstos se encuentran organizados de alguna manera racional que facilita su búsqueda.

b.- Como usuario de la biblioteca, desconozco –y probablemente no me interesa tampoco– cuál es ese esquema de distribución de los libros. Lo importante es que el bibliotecario lo domina a fondo y que por ello puede encontrar y entregarme la obra solicitada.

c.- No tengo que preocuparme en absoluto por cómo podré buscar en el futuro los nuevos libros que lleguen a la biblioteca, o por cuáles se encuentran prestados a otros usuarios, o cuántos hay en total, o cuáles sean eliminados por mal estado o cualquier otro motivo. De nuevo, para todo eso está el bibliotecario.

Pues bien, en esta sencilla analogía, la base de datos equivale a la colección de cientos o miles o más de libros, mientras que el SABD está representado por el bibliotecario. A través de un SABD, podemos almacenar electrónicamente los datos que nos interesa conservar para algún propósito dado, con la confianza de que por medio de él podremos incluir nuevos datos en el futuro, modificar los datos existentes o borrar los que ya no queramos conservar. De tanto o mayor importancia es el hecho de que, empleando el SABD, podemos formular preguntas<sup>98</sup> y extraer información útil de la colección de datos, tanto cualitativa (ejemplo: ¿cuál es el libro más solicitado por los usuarios de la biblioteca?) como cuantitativa (¿cuántas personas no han devuelto títulos prestados con fecha de entrega vencida?).

La tecnología de las bases de datos viene desarrollándose con mayor ímpetu aproximadamente a partir de los años sesenta, atravesando una serie de fases que se distinguen por el modelo dominante en cada cual. Así, inicialmente atravesamos por las etapas de las bases de datos jerárquicas y de redes. Durante los años setenta y a raíz del trabajo revolucionario que a nivel teórico desarrolló E.F. Codd, nació la era de las bases de datos relacionales, que en gran medida se prolonga hasta los últimos años. Actualmente nos encontramos en la fase de predominio de las llamadas bases de datos objeto-relacionales y que se espera que dé paso, a su vez, a una etapa de preponderancia de las bases de datos orientadas a objetos.<sup>99</sup>

Ahora bien, partiendo de que las aplicaciones informáticas –en cuanto secuencias de instrucciones ejecutables por un sistema informático– indudablemente merecen el amparo de la ley, no nos ofrece mayor dificultad lo referente a la defensa de los derechos e intereses de los autores de un SABD. En efecto, se trata en la especie de un software de aplicación que recibe el mismo tratamiento que cualquier otro programa informático. La cuestión que nos ocupa aquí radica, más bien, en si la colección misma de datos (o sea, la base de datos propiamente dicha) es o no susceptible de recibir un reconocimiento similar.

<sup>98</sup> “Consultas”, en el lenguaje de BD.

<sup>99</sup> Una discusión resumida acerca de qué son las BD orientadas a objetos y, en concreto, acerca de sus posibles aplicaciones en el ámbito judicial, aparece en: HESS ARAYA, Christian. *Creación de una base de datos de jurisprudencia constitucional, orientada a objetos*. Disponible electrónicamente en mi sitio web personal de <http://www.hess-cr.com>.

La respuesta de la doctrina es pacíficamente afirmativa, pero precisando que lo que se protege no son los datos en sí mismos, sino la *compilación* que representan, siempre que de ella pueda decirse que es *original y creativa*.<sup>100</sup> En otras palabras, el derecho de autor tutela únicamente la creatividad desplegada en la selección u ordenamiento de la información.<sup>101</sup> Esto es así porque los datos contenidos en la compilación –por mucho que el diseñador haya trabajado en su recolección y clasificación– no son originales suyos: “... un desarrollador de bases de datos no crea hechos; él o ella los *descubre* y nadie puede atribuirse derechos de autor sobre un descubrimiento.”<sup>102</sup>

Para comprenderlo, baste un sencillo ejemplo. Considérese, en efecto, la sección de “páginas blancas” de la guía telefónica. Ésta representa, sin duda, una típica base de datos, que incluye los nombres de todos los abonados del servicio y su respectivo número de teléfono. ¿Cabría entonces registrarlo como una obra protegida? La respuesta es no, porque en esta compilación no hay originalidad ni creatividad: se trata de la mera sumatoria de todos los datos pertinentes, siguiendo un orden alfabético que nada tiene de original o innovador.<sup>103</sup> Pero piénsese ahora en una base de datos que, luego de un largo y duro trabajo arqueológico, produzca un catálogo completo y detallado sobre las piezas arqueológicas de nuestro país, clasificadas en función del pueblo indígena que las creó e incluyendo datos sobre zona geográfica de localización, período histórico al que corresponden, comentarios personales, etcétera. En este caso, no nos cabe duda de que tal compilación sí merecería la tutela del derecho autoral, en reconocimiento a su originalidad y de su creatividad.

Nótese, entonces, la diferencia clara que existe entre una base de datos puramente inclusiva y otra que es el fruto de la selectividad en su construcción.

Ahora bien, el hecho de que lo tutelado sea la compilación y no los datos por sí solos es lógico, ya que –como se dijo– no sólo es frecuente que éstos incluyan

<sup>100</sup> Donde por “compilación” entendemos, en este contexto, el resultado de popular con información un determinado esquema de base de datos (conjunto de tablas, vistas, índices, etc.).

<sup>101</sup> SAMUELSON, *op. cit.*

<sup>102</sup> ROSE, Lance. *Op. cit.*, página 110, traducción libre del autor.

<sup>103</sup> La Corte Suprema de los Estados Unidos falló expresamente en este sentido en el caso “*Feist Publications, Inc. versus Rural Telephone Service Co., Inc.*” Sin embargo, cabe indicar que al menos la Corte Federal australiana ha seguido un lineamiento contrario en el caso “*Telstra Corporation Limited versus Desktop Marketing Systems Pty., Ltd.*”, indicando que los directorios de páginas blancas y amarillas de Telstra eran lo suficientemente originales como para ameritar tutela bajo el derecho de autor, reconociendo de este modo el “sudor de la frente” involucrado en su producción. MELCHIOR, Yedidya M. En *The Mishpat CyberLaw Informer*. Boletín electrónico que se puede consultar en la dirección <http://mishpat.net>. Número 71, junio del 2001.

información que no es ni original ni creativa, sino que puede resultar incluso de dominio público (como leyes, sentencias judiciales, etcétera).<sup>104</sup>

Sin embargo, si la base de datos no es de información pública y las personas que deseen aprovecharla deben suscribir primero un contrato que incorpore un compromiso de confidencialidad, entonces resultará viable pretender una protección jurídica, ya no a través del derecho autoral sino vía la legislación sobre secretos industriales (información no divulgada), tutela que en este caso alcanzaría incluso a los propios datos fácticos.<sup>105</sup>

– 0 –

Al igual que ocurre respecto del derecho autoral en general, la tutela jurídica de las bases de datos deriva de un cúmulo de disposiciones normativas internacionales, regionales y nacionales, que repasamos seguidamente.

#### **Plano internacional:**

En sí mismo, el Convenio de Berna para la Protección de las Obras Literarias y Artísticas no se refiere explícitamente a las bases de datos.<sup>106</sup> No obstante, el lenguaje amplio de algunos de sus preceptos (concretamente los artículos 2.1 y 2.5) permiten entenderlas como “colecciones”, susceptibles de recibir protección equiparable a la de las obras literarias y artísticas en general.<sup>107</sup> Así lo expresó el Comité de Expertos de la OMPI durante una sesión realizada en 1994.<sup>108</sup>

Ese mismo año, la OMC emitió el ya citado “*Acuerdo sobre los ADPIC*” (Acuerdo sobre los Aspectos de los Derechos de Propiedad Intelectual relacionados con el Comercio), cuyo numeral 10.2 dispone:

<sup>104</sup> Sobre la problemática jurídica de las bases de datos de textos legales, véase: GUASCH DÍAZ, Diego Manuel. “La extracción de Normas de Boletines Oficiales en papel y de ediciones en Internet. Algunas Consideraciones Jurídicas relevantes para las Empresas de Bases de Datos”. En *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), agosto del 2000.

<sup>105</sup> ROSE, Lance. *Op. cit.*, página 111.

<sup>106</sup> Organización Mundial de la Propiedad Intelectual (OMPI). *Legislación nacional y regional existente relativa a la propiedad intelectual en materia de bases de datos*. Memorandum preparado por la Oficina Internacional y disponible electrónicamente en [http://www.wipo.org/spa/meetings/infdat97/db\\_im\\_2.htm](http://www.wipo.org/spa/meetings/infdat97/db_im_2.htm). Junio de 1997.

<sup>107</sup> VILLEGAS, Ricardo. “La originalidad como condicionante para la protección internacional de las bases de datos”, en *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 38, setiembre del 2001.

<sup>108</sup> *Ídem*. Véase el documento BCP/CE/IV/3, párrafo 46, en el sitio web de la OMPI (<http://www.wipo.int>).

*“Las compilaciones de datos o de otros materiales, en forma legible por máquina o en otra forma, que por razones de la selección o disposición de sus contenidos constituyan creaciones de carácter intelectual, serán protegidas como tales. Esa protección, que no abarcará los datos o materiales en sí mismos, se entenderá sin perjuicio de cualquier derecho de autor que subsista respecto de los datos o materiales en sí mismos.”*

Este precepto –transcrito casi literalmente– dio paso al actual artículo 5 del “Tratado de la OMPI sobre derecho de autor” y que, en este sentido, constituye el precepto internacional de mayor interés en la materia. Establece dicha norma:

*“Artículo 5.- Compilaciones de datos (bases de datos). Las compilaciones de datos o de otros materiales, en cualquier forma, que por razones de la selección o disposición de sus contenidos constituyan creaciones de carácter intelectual, están protegidas como tales. Esa protección no abarca los datos o materiales en sí mismos y se entiende sin perjuicio de cualquier derecho de autor que subsista respecto de los datos o materiales contenidos en la compilación.”*

La Conferencia también adoptó en esa oportunidad una “declaración concertada”, que en cuanto nos interesa aclara que “El ámbito de la protección de las compilaciones de datos (bases de datos) en virtud del Artículo 5 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.”

### **Plano regional:**

A nivel regional, los preceptos aplicables a la materia obviamente dependerán del ámbito territorial relevante. Entre algunas disposiciones de interés destacan<sup>109</sup>:

El artículo 4 de la Decisión 351 del Acuerdo de Cartagena, que contiene el “Régimen común sobre derecho de autor y derechos conexos”, concertado el 17 de diciembre de 1993 entre Bolivia, Colombia, Ecuador, Perú y Venezuela.

El artículo 1705.1 del “Tratado de Libre Comercio de América del Norte”, alcanzado entre los gobiernos de Canadá, Estados Unidos y México el 8 de diciembre de 1993.

---

<sup>109</sup> *Idem.*

La Directiva de la Comunidad Europea sobre bases de datos (“Directiva 96/9/EC del Parlamento Europeo y del Consejo, de 11 de marzo de 1996, sobre la protección jurídica de las bases de datos”). Al respecto, valga destacar que este ordenamiento incluso va más allá, al contemplar un régimen *sui generis* de tutela de las bases de datos que extiende y complementa el marco general del derecho autoral. Los creadores de bases de datos están protegidos contra la extracción o la reutilización de la información (totalmente o en parte sustancial), durante un lapso de 15 años. A fin de estimular a otros países a promulgar una normativa similar, la protección que ofrece la Directiva se aplica solo a los nacionales de los países miembros de la Unión Europea y a los ciudadanos de otros estados que cuenten con una legislación equiparable.<sup>110</sup>

### **Plano nacional:**

La legislación interna de la mayoría de los Estados miembros de la OMPI contiene disposiciones que resultan directa o indirectamente aplicables a las bases de datos y, en este sentido, Costa Rica no es excepción.

Nuestro país, como ya se ha anticipado atrás, es signatario tanto de la Convención de Berna como del Tratado de la OMPI sobre Derecho de Autor. En el nivel de la legislación local, nuestro principal marco normativo en esta materia lo provee la Ley de Derechos de Autor y Derechos Conexos, en especial, a partir de las modificaciones introducidas en ese texto mediante las leyes número 7397 de 28 de abril de 1994 y 7979 de 22 de diciembre de 1999, que se refieren específicamente a los programas de cómputo y a las bases de datos. Es así que dispone, escuetamente, la citada ley:

*“Artículo 8.- (...) Las bases de datos están protegidas como compilaciones.”*

Más adelante, al señalar los procedimientos de inscripción de obras de esta naturaleza en el Registro de Derechos de Autor, se indica:

*“Artículo 103.- Para inscribir una producción, el interesado presentará, ante el Registrador, una solicitud escrita con los siguientes requisitos:*

*(...)*

*5) Cuando se trate de inscribir un programa de cómputo o una base de datos, la solicitud se presentará con cualquiera de los*

<sup>110</sup> SAMUELSON, *op. cit.*

*siguientes elementos: el programa, la descripción o el material auxiliar.”*

Técnicamente, aunque una base de datos puede contener segmentos de código ejecutable,<sup>111</sup> no pensamos que para lograr la indicada inscripción sea necesario aportar su código fuente. En efecto, puesto que lo que se protege es la compilación como tal y no el contenido de la base de datos, lo único que podría tener interés a efectos de registro sería el *esquema* o diseño lógico de sus componentes (tablas, relaciones, objetos, etcétera), debidamente documentado.

– 0 –

En conclusión, las bases de datos son indudablemente acreedoras de la protección jurídica otorgada a las obras tecnológicas en general, tutela que es distinta e independiente de la que se confiere a las aplicaciones empleadas para crearlas o administrarlas. Para ese efecto, no obstante, es necesario que la compilación no sea puramente inclusiva, sino que debe existir un criterio de selectividad que brinde a la colección de datos los indispensables atributos de originalidad y creatividad.

### **Medidas tecnológicas de defensa de las obras digitales**

*“Cada vez que escribo acerca de la imposibilidad de proteger eficazmente los archivos digitales en un computador de propósito general, obtengo respuestas de personas que rechazan la muerte del derecho de autor. ‘¿Cómo podrán los autores y artistas obtener remuneración por su trabajo?’ me preguntan. La verdad, no lo sé. Me siento casi como el físico que acaba de explicar la relatividad a un grupo de aspirantes a viajeros interestelares, tan sólo para que le pregunten: ‘¿Cómo espera usted que alcancemos las estrellas entonces?’ Lo siento, pero eso tampoco lo sé.”*

Bruce Schneier <sup>112</sup>

Ante el creciente fenómeno de la piratería de software y de otras obras digitales protegidas en general (incluyendo discos musicales, películas cinematográficas, transmisiones codificadas de televisión por cable, etcétera), la industria ha procurado responder aplicando diversas medidas tecnológicas de protección para impedir el acceso no autorizado a dichas obras y posibilitar la administración electrónica de los correspondientes derechos legales.<sup>113</sup>

<sup>111</sup> En forma de procedimientos almacenados (*stored procedures*) o métodos, según se trate de bases de datos relacionales u orientadas a objetos, respectivamente.

<sup>112</sup> SCHNEIER, Bruce. “Protecting Copyright in the Digital World”, en la revista electrónica *Crypto-Gram* (<http://www.counterpane.com/crypto-gram.html>). 15 de agosto del 2001.

<sup>113</sup> Por ejemplo, empleando “marcas de agua” digitales, “applets” Java que impiden la redistribución, etc. KOHN, Paul, *op. cit.*

En adición (y puesto que es de esperar que incluso las medidas más rigurosas resulten eventualmente vulnerables a los avances futuros tecnológicos), se ha procurado obtener protección jurídica complementaria contra acciones de terceros, orientadas a desactivar o de alguna manera burlar las defensas y mecanismos en cuestión.<sup>114</sup> Para los voceros de la industria, especialmente la del entretenimiento, esta última clase de acciones son el equivalente de forzar el ingreso a una vivienda, de manera que cualesquiera instrumentos o aplicaciones que de algún modo procuren o faciliten la inutilización de las defensas tecnológicas en cuestión son moralmente idénticos a las ganzúas o barras que emplean los delincuentes para robar casas.<sup>115</sup> Alegan, además, que estas defensas y prohibiciones son indispensables para hacer posible la circulación de obras protegidas en ambientes de redes, especialmente redes abiertas como la Internet, puesto que –de lo contrario– los titulares de derechos arriesgan perder el control sobre dichas obras.<sup>116</sup>

Nótese que estas disposiciones legales tienden a introducir lo que podría calificarse de un tercer estrato de protección a las obras tecnológicas:

- El primer nivel lo provee la legislación de propiedad intelectual, que tutela los derechos del autor respecto de la obra propiamente dicha.
- El segundo nivel corresponde a la protección adicional que brindan a las obras las medidas tecnológicas (empleo de contraseñas, marcas de agua digitales, encriptación, etcétera).

<sup>114</sup> Un caso notorio surgió en diciembre de 1999, cuando el programador noruego John Lech Johansen aplicó la ingeniería reversa a los DVD (discos de vídeo digital) y luego difundió en la Internet una aplicación llamada “DeCSS”, que permite extraer el contenido de los discos. El caso dio pie a una intensa batalla legal, en la que la defensa alegaba no sólo que la aplicación es necesaria para asegurar la portabilidad del formato DVD (por ejemplo, para posibilitar su empleo bajo el sistema operativo Linux), sino además que se trata de una cuestión de defensa de la libertad de expresión, tesis esta última que ha sido acogida por al menos un tribunal federal estadounidense (al respecto, véase [http://news.com.com/2100-1026\\_3-5166887.html](http://news.com.com/2100-1026_3-5166887.html)) y rechazada por otro (véase <http://news.cnet.com/news/0-1005-200-8011238.html>). El caso concluyó tres años después con la absolutoria de Johansen, sobre la base del argumento que éste tenía derecho a emplear el programa para acceder al contenido de un DVD adquirido para uso personal, aunque el tribunal reconoció que la tecnología podía ser empleada tanto para fines lícitos como ilícitos.

<sup>115</sup> SAMUELSON, Pamela. “Towards more sensible anti-circumvention regulations”. Documento electrónico disponible en la dirección <http://www.sims.berkeley.edu/~pam/papers/fincrypt2.doc>, visitado el 1 de noviembre del 2001.

<sup>116</sup> INDUSTRY CANADA / CANADIAN HERITAGE, “Consultation paper on digital copyright issues”. Documento electrónico disponible en <http://strategis.ic.gc.ca/SSG/rp01099e.html>. Junio del 2001. Página visitada el 27 de agosto del 2001.



- El tercer substrato consistiría, entonces, en la sanción normativa de las medidas tecnológicas en sí y la penalización de su eventual quebranto.

Un primer conjunto de regulaciones sobre el tema fue introducido, a instancias de los EE.UU., en el “Tratado sobre Derechos de Autor” de la OMPI. En efecto, dispone ese instrumento:

*“Artículo 11.- Obligaciones relativas a las medidas tecnológicas. 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 y que, respecto de sus obras, restrinjan actos que no estén autorizados por los autores concernidos o permitidos por la Ley.*

*Artículo 12.- Obligaciones relativas a la información sobre la gestión de derechos. 1) Las Partes Contratantes proporcionarán recursos jurídicos efectivos contra cualquier persona que, con conocimiento de causa, realice cualquiera de los siguientes actos sabiendo o, con respecto a recursos civiles, teniendo motivos razonables para saber que induce, permite, facilita u oculta una infracción de cualquiera de los derechos previstos en el presente Tratado o en el Convenio de Berna: i) suprima o altere sin autorización cualquier información electrónica sobre la gestión de derechos; ii) distribuya, importe para su distribución, emita, o comunique al público, sin autorización, ejemplares de obras sabiendo que la información electrónica sobre la gestión de derechos ha sido suprimida o alterada sin autorización. 2) A los fines del presente Artículo, se entenderá por ‘información sobre la gestión de derechos’ la información que identifica a la obra, al autor de la obra, al titular de cualquier derecho sobre la obra, o información sobre los términos y condiciones de utilización de la obras, y todo número o código que represente tal información, cuando cualquiera de estos elementos de información estén adjuntos a un ejemplar de una obra o figuren en relación con la comunicación al público de una obra.”*

De manera paralela, el “Tratado de la OMPI sobre interpretación o ejecución de fonogramas”, del mismo año,<sup>117</sup> establece también:

<sup>117</sup> Con fecha de entrada en vigor el 20 de mayo del 2002, tres meses después de que Honduras se convirtiera en el trigésimo país en dar su adhesión a este instrumento. Su propósito es actualizar y complementar el principal tratado sobre derechos conexos (a saber, la Convención de Roma de 1961), para proteger los intereses de los productores de fonogramas o grabaciones sonoras así como los de los artistas intérpretes o ejecutantes cuyas

*“Artículo 18.- Obligaciones relativas a las medidas tecnológicas*

*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.*

*Artículo 19.- Obligaciones relativas a la información sobre la gestión de derechos. 1º- Las partes Contratantes proporcionarán recursos jurídicos adecuados y efectivos contra cualquier persona que, con conocimiento de causa, realice cualquiera de los siguientes actos sabiendo o, con respecto a recursos civiles, teniendo motivos razonables para saber que induce, permite, facilita u oculta una infracción de cualquiera de los derechos previstos en el presente Tratado: i) suprima o altere sin autorización cualquier información electrónica sobre la gestión de derechos. ii) distribuya, importe para su distribución, emita, comunique o ponga a disposición del público, sin autorización, interpretaciones o ejecuciones, ejemplares de interpretaciones o ejecuciones fijadas o fonogramas sabiendo que la información electrónica sobre la gestión de derechos ha sido suprimida o alterada sin autorización. 2º- A los fines del presente artículo, se entenderá por 'información sobre la gestión de derechos' la información que identifica al artista intérprete o ejecutante, a la interpretación o ejecución del mismo, al productor del fonograma, al fonograma y al titular de cualquier derecho sobre interpretación o ejecución o el fonograma, o información sobre las cláusulas y condiciones de la utilización de la interpretación o ejecución o del fonograma, y todo número o código que represente tal información, cuando cualquiera de estos elementos de información esté adjunto a un ejemplar de una interpretación o ejecución fijada o a un fonograma o figuren en relación con la comunicación o puesta a disposición del público de una interpretación o ejecución fijada o de un fonograma.”*

El tema posteriormente llegó a despertar una intensa polémica en los propios Estados Unidos, a propósito de la promulgación, en octubre de 1998, de la llamada *Digital Millenium Copyright Act* (DMCA, Ley Pública 105-304/1998), uno de

---

interpretaciones o ejecuciones se fijan en fonogramas.

cuyos propósitos era el de implementar internamente las disposiciones del mencionado tratado sobre derechos de autor. La DMCA actualmente constituye la Sección 1201(a)(1) del Título 17 del *United States Code*.

A la DMCA se le atribuye el ir demasiado lejos en cuanto a la regulación de actividades legítimas relativas al desactivado de los sistemas técnicos de protección de obras digitales, así como de las tecnologías aplicables a ese propósito.<sup>118</sup> En líneas generales, se ha estimado que esas regulaciones son demasiado estrechas y ambiguas; que no incorporan ninguna excepción de propósito general que permitan a los tribunales de justicia exonerar de responsabilidad por ciertas actividades claramente justificables; y que se inclinan demasiado a favor de los intereses comerciales de la industria.<sup>119</sup> En concreto, pues, la discusión de fondo –válida, nos parece, tanto en el contexto estadounidense como el nuestro– es si el empleo de las discutidas defensas tecnológicas torna nugatorias o no las posibilidades de una persona de hacer “usos honestos” de obras digitales protegidas por el derecho de autor.

Por ejemplo, si he adquirido legítimamente un disco compacto de música, se considera un uso honesto copiarlo a un dispositivo portátil o bien a un cassette o disco compacto grabable (CD-R) con el propósito de escucharlo en el automóvil. Y si poseo una copia legal de una cinta cinematográfica, es igualmente permisible copiar pequeñas porciones y exhibirlas públicamente para efectos académicos o críticos. Sin embargo, si dichas obras están dotadas de mecanismos tecnológicos de protección, sería imposible ejercer esos derechos legítimos a menos que haga uso de algún mecanismo que desactive la protección incorporada. Peor aun, el empleo de restricciones tecnológicas puede incluso llevar a extremos para los que claramente no fueron pensadas, tales como restringir el acceso a obras que carecen de protección en lo que toca a los derechos patrimoniales de autor. Imaginemos, por ejemplo, una versión digital de la “Política”, de Aristóteles. La obra en sí se encuentra indudablemente en el dominio público y nadie podría pretender restringir el uso que se haga de ella. Pero, a través de los medios de protección incorporados en esta versión, su distribuidor podría impedir que la obra sea reproducida o impresa, al controlar el acceso que se haga del contenido. Y si se pretendiera emplear algún mecanismo para desactivar la protección y así ejercitar el derecho de uso honesto, se estaría infringiendo la ley.<sup>120</sup> Por estos motivos, los críticos del empleo de defensas tecnológicas frecuentemente le aplican la *metáfora de la gaveta con cerrojo*: aunque esté legitimado para tener

<sup>118</sup> SAMUELSON, Pamela. “Good news and bad news on the intellectual property front”, en *Communications of the ACM*, volumen 42, número 3, marzo de 1999. Página 19.

<sup>119</sup> SAMUELSON, “Towards ...” *Op. cit.*

<sup>120</sup> Este ejemplo es dado por el profesor Lawrence Lessig en WALKER, Jesse, “*Cyberspace’s Legal Visionary*”. Disponible electrónicamente en <http://www.reason.com/0206/fe.jw.cyberspaces.shtml>, junio del 2002.

acceso al contenido de la gaveta, cada vez que desee hacerlo debo pedir permiso al poseedor de la llave, quien tiene el poder de decidir si me la facilita o no.<sup>121</sup>

¿Qué tiene que ver esto en concreto con el software? Una prohibición amplia de la posibilidad de suprimir las defensas tecnológicas de una aplicación traería como consecuencia la imposibilidad de ejercer legalmente actividades como la ingeniería reversa (por ejemplo, cuando se requiera para depurar un programa cuyo código fuente no está disponible), pruebas de seguridad de un software e investigaciones en el área de la encriptación.<sup>122</sup> Tampoco sería posible efectuar ciertas investigaciones académicas que requieran, por ejemplo, el empleo de herramientas informáticas para hacer búsquedas dentro de la obra digital protegida tecnológicamente.<sup>123</sup> De mayor interés para los usuarios finales, las pluricitadas defensas impedirían la realización de copias de respaldo legítimas de una aplicación, un “uso honesto” claramente reconocido.

En Estados Unidos, la Biblioteca del Congreso –bajo mandato de la DMCA– estableció en octubre del 2000 que existe una excepción relativa al software respecto de las prohibiciones contenidas en la ley: el desactivado de las defensas tecnológicas de “obras literarias, incluyendo los programas de computador y las bases de datos, protegidas por mecanismos de control de acceso que impidan dicho acceso debido a mal funcionamiento, daño u obsolescencia”. Es decir, sería permitido tomar medidas para suprimir un mecanismo de control defectuoso que, en razón de dicho desperfecto, impida utilizar la obra del modo previsto (por ejemplo, ejecutar la aplicación).<sup>124</sup> Tres años después se agregó los supuestos de los programas de cómputo y los videojuegos en formatos obsoletos, así como los libros electrónicos que fuesen inaccesibles para personas con discapacidades.<sup>125</sup>

Sin embargo, la discusión pasó de lo académico a lo dramático a mediados del año 2001, con el llamado “caso Sklyarov”.<sup>126</sup> En efecto, un programador ruso llamado Dmitry Sklyarov, fue arrestado en Estados Unidos por el FBI, que lo acusó de violar la DMCA al publicar un software llamado “Advanced eBook Processor”, que suprime las defensas tecnológicas de los libros electrónicos producidos por la conocida empresa Adobe. La compañía rusa ElcomSoft, para la que Sklyarov

<sup>121</sup> INDUSTRY CANADA, *op. cit.*

<sup>122</sup> SAMUELSON, Pamela. “Why the anticircumvention regulations need revision”, en *Communications of the ACM*, volumen 42, número 9, setiembre de 1999. Página 17.

<sup>123</sup> APPEL, Andrew W. y FELTEN, Edward W. “Technological access control interferes with non-infringing scholarship”, en *Communications of the ACM*, volumen 43, número 9, setiembre del 2000, página 21.

<sup>124</sup> Véase la declaración completa en <http://www.loc.gov/copyright/1201/anticirc.html>

<sup>125</sup> Véase <http://www.copyright.gov/fedreg/2003/68fr2011.pdf>

<sup>126</sup> Reportado –entre otros– por MELCHIOR, Yedidya M., en *The Mishpat Cyberlaw Informer* (<http://www.cyberlawinformer.com>), agosto del 2001.

labora, alegó en su defensa que la supresión de dichas defensas es necesaria para efectuar los respaldos que exige la legislación de ese país. Grandes protestas siguieron al arresto, al punto de que la propia Adobe se manifestó a favor de la liberación del programador. A finales del 2002, tanto Sklyarov como ElcomSoft han sido exonerados de responsabilidad legal.<sup>127</sup>

La directiva 2001/29/EC del Parlamento Europeo y del Consejo de Europa, de 22 de mayo del 2001 (“Directiva de los Derechos de Autor y Derechos Afines en la Sociedad de la Información”), se refiere a esta temática en su capítulo III, artículos 6 y 7. Prohíbe tanto la desactivación de las “medidas efectivas de defensa” que haya implantado el titular de los derechos respectivos, como la elaboración y distribución de mecanismos que conduzcan a esa finalidad. Sin embargo, la directiva indica que dichos titulares deben tomar medidas para asegurar que las personas legitimadas puedan hacer usos honestos de las obras protegidas, de conformidad con las respectivas legislaciones nacionales.

En otros países, las soluciones han sido diversas. Japón, por ejemplo, ha implementado una normativa altamente restrictiva. Australia ha preferido no prohibir el acto mismo de desactivar las defensas tecnológicas, pero sí prohíbe la elaboración o intercambio de mecanismos para facilitárselo a otros, o prestarles asistencia para ese propósito.<sup>128</sup> En EE.UU., algunas propuestas legislativas tienden a trasladar el problema de los proveedores de contenido a los fabricantes de hardware, quienes se verían legalmente obligados a agregar sistemas mecánicos de gestión de los derechos de propiedad intelectual a sus productos (computadoras, unidades de CD-R, etcétera). Los fabricantes, por su parte, se han opuesto a estas iniciativas, no solo alegando el costo y esfuerzo adicional que ello representaría, sino además que –de todos modos– estos sistemas de protección tendrían poca o ninguna eficacia para eliminar ciertos actos de piratería, por ejemplo, de quienes sustraen obras digitales durante el proceso de producción.<sup>129</sup>

¿Cuál es la situación en Costa Rica? Se debe recordar, en primer término, que nuestro país ratificó los tratados de la OMPI sobre fonogramas y derechos de autor –respectivamente– mediante leyes número 7967 de 2 de diciembre y número

<sup>127</sup> Otro caso similar involucra al investigador Edward Felten, quien debió abstenerse de publicar un trabajo académico sobre encriptación, bajo amenaza de acciones legales conforme a la DMCA. Felten ha establecido una demanda civil para defender sus derechos y los de sus colegas en general. El caso se encuentra pendiente de resolución a la fecha de escribir estas líneas.

<sup>128</sup> INDUSTRY CANADA, *op. cit.*

<sup>129</sup> Véase <http://ww1.infoworld.com/cgi-bin/fixup.pl?story=http://www.infoworld.com/articles/hn/xml/02/06/13/020613hnclash.xml&dctag=contentmanagement> (17 de junio del 2002).

7968 de 16 de diciembre, ambas de 1999. De manera que los artículos citados *supra* resultan enteramente aplicables en nuestro medio.

Más recientemente, la Ley de Procedimientos de Observancia de los Derechos de Propiedad Intelectual, número 8039 de 5 de octubre del 2000, incorporó a nuestro ordenamiento las disposiciones siguientes:

*“Artículo 62.- Alteración, supresión, modificación o deterioro de las defensas tecnológicas contra la reproducción de obras o la puesta a disposición del público.*

*Será sancionado con prisión de uno a tres años quien, en cualquier forma, altere, suprima, modifique o deteriore los mecanismos de protección electrónica o las señales codificadas de cualquier naturaleza que los titulares de derechos de autor, artistas, intérpretes o ejecutantes, o productores de fonogramas hayan introducido en las copias de sus obras, interpretaciones o fonogramas, con la finalidad de restringir su comunicación al público, reproducción o puesta a disposición del público.”*

*“Artículo 63.- Alteración de información electrónica colocada para proteger derechos patrimoniales del titular.*

*Será sancionado con prisión de uno a tres años quien altere o suprima, sin autorización, la información electrónica colocada por los titulares de los derechos de autor o conexos, para posibilitar la gestión de sus derechos patrimoniales y morales, de modo que puedan perjudicarse estos derechos.*

*La misma pena se aplicará a quien distribuya, importe con fines de distribución, emita o comunique al público, sin autorización, ejemplares de obras, interpretaciones o fonogramas, sabiendo que la información electrónica, colocada por los titulares de derechos de autor o conexos, ha sido suprimida o alterada sin autorización.”*

### **Notas críticas sobre el problema de la piratería de software**

Se libra una batalla mundial contra la piratería de obras protegidas por la propiedad intelectual, que alcanza a obras de diversas clases, entre ellas el software. Las sumas que deja de percibir la industria se estiman en unos \$11.000 millones al año y ningún país se exime de responsabilidad, incluyendo a los más desarrollados. Por ejemplo, se estima que al menos un 25 por ciento del nuevo

software instalado por las empresas en los EE. UU. es producto de la piratería, cifra que representa además una quinta parte del total mundial. Las tasas más altas parecen concentrarse en la Europa del Este, donde alcanzan el 67 por ciento.<sup>130</sup>

¿Qué estimula el fenómeno? Aunque la respuesta variará de un entorno a otro, ciertas hipótesis toman fuerza, al menos respecto de naciones en vías de desarrollo como la nuestra.

Es claro que la tutela que brinda la propiedad intelectual brinda un obvio incentivo a la investigación y al desarrollo de productos nuevos, a la vez que estimula la inversión extranjera en los países que la practiquen eficazmente. Pero, al propio tiempo, se ha destacado que los regímenes estrictos en materia de propiedad intelectual dificultan la transferencia tecnológica y, por ende, amenazan el bienestar social de las naciones menos desarrolladas.<sup>131</sup>

Frente a la opción de adquirir un producto legítimo o una copia pirata, un comprador sopesará varios factores. Se puede ofrecer la hipótesis de que –excluyendo al delincuente compulsivo– es muy probable que la elección del producto original se facilitará por la percepción que tenga el potencial adquirente de que la transacción es esencialmente justa y, además, razonable; es decir, que el producto que va a comprar realmente vale lo que piden por él y que este precio es aceptable y accesible. Por el contrario, si existe en el consumidor la convicción de que el trato no es justo o razonable, es probable que lo rechace e incluso que considere moralmente aceptable optar por una alternativa que, en rigor legal, no sea lícita.

En última instancia, se debe comprender que el fenómeno de la piratería de software trasciende lo puramente jurídico para involucrar cuestiones socioeconómicas esenciales. En efecto, existen estudios que claramente identifican la existencia de una relación inversa entre los precios del software y la piratería.<sup>132</sup> No solo –pero especialmente– en los países en vías de desarrollo, los niveles de ingreso pueden afectar la capacidad de los consumidores para adquirir

<sup>130</sup> HESS ARAYA, Christian. “Piratería no, pero ...”, en el periódico *La Nación*, edición del 12 de noviembre del 2001, San José, Costa Rica. Véase también la noticia “Study: Software piracy up for second straight year”, difundida por la agencia Reuters el 10 de junio del 2002 y disponible electrónicamente en la dirección [http://www.reuters.com/news\\_article.jhtml?type=technologynews&StoryID=1068362](http://www.reuters.com/news_article.jhtml?type=technologynews&StoryID=1068362).

<sup>131</sup> (Autor no especificado). “Fin de la falsificación y la piratería”, en revista *Enlace Mundial*. Promotora del Comercio Exterior de Costa Rica (PROCOMER), año 3, número 2. San José, setiembre de 1999.

<sup>132</sup> GOPAL, Ram D. y SANDERS, G. Lawrence. “Global software piracy: you can't get blood out of a turnip”, en *Communications of the ACM*, volumen 43, número 9, setiembre del 2000, página 83.

software: el elevado precio de éste es entonces lo que impulsa a muchas personas a optar por copias ilegales.

*“La motivación para la piratería es el significativo diferencial de precios entre las versiones legales y las piratas, lo cual se ve exacerbado por los relativamente bajos niveles de ingreso en los países en vías de desarrollo. El problema clave es que los precios son fijados a niveles de Estados Unidos, que son significativamente más altos que lo que pueden pagar las personas en la mayoría de los países. En una gran cantidad de éstos, la adquisición legítima de software no es una opción que la mayoría de los individuos considere seriamente.”*<sup>133</sup>

Bajo esta óptica, la estricta aplicación de las leyes contra la piratería de software a veces no es vista favorablemente ni siquiera por las propias autoridades gubernamentales, que saben que esto solamente restringiría el acceso al software de una mayoría de la población.<sup>134</sup>

*“Entonces, se torna aparente que concentrar esfuerzos solamente en la observancia de los derechos de propiedad intelectual tendrá un éxito limitado. En la lucha contra la piratería, las armas legislativas y educativas podrán ganar unas cuantas batallas, pero la guerra contra la piratería no podrá ser ganada sin considerar las actuales políticas draconianas de precios. Argumentamos que los principios económicos de diferenciación de precios deben ser aplicados por la industria fabricante de software para lograr reducciones significativas en las tasas globales de piratería.”*<sup>135</sup>

Quizás en ello radique el meollo del problema.

---

<sup>133</sup> GOPAL y SANDERS, *op. cit.*, página 86; traducción libre del autor.

<sup>134</sup> *Idem*. No obstante, en Costa Rica, es justo destacar que el Gobierno de la República ha promulgado una serie de directrices orientadas a combatir el empleo ilegal de programas de cómputo en las oficinas del Estado. Véase el texto completo en el Anexo II de esta obra. Igual han hecho algunas otras dependencias de la Administración pública descentralizada.

<sup>135</sup> *Idem*; traducción libre del autor.



## Capítulo III - Contratos relativos al software

Puesto que la doctrina mayoritaria y la ley consideran al software como una obra (ya fuere literaria o tecnológica), las figuras negociales que pueden estar asociadas a él serán todas las pertinentes a esa especial clase de bienes jurídicos, por lo cual no haremos aquí un examen especialmente exhaustivo al respecto. Más bien dedicaremos nuestra atención a los tipos principales de convenios observados en la práctica: el *contrato de licencia de uso de software*, la *cesión de derechos*, el *desarrollo de software a la medida* y el arrendamiento de software en línea (ASP).

### Contrato de licencia de uso de software

Es la principal figura contractual aplicable en la actualidad al software. En palabras de la OMPI, “Un acuerdo de licencia es una asociación entre un titular de derechos de propiedad intelectual (licenciante) y otra persona que recibe la autorización de utilizar dichos derechos (licenciataria) a cambio de un pago convenido de antemano (tasa o regalía).”<sup>1</sup>

Nuestro ordenamiento la contempla escuetamente, referida a toda clase de obras (incluyendo, desde luego, al software), en el Reglamento a la Ley de Derechos de Autor:

*“Artículo 43.- El titular de derechos de autor o conexos, o la entidad de gestión colectiva que lo represente, podrá sustituir la enajenación total o parcial del derecho patrimonial, por una simple concesión a terceros de una licencia o autorización de uso de la obra o producción intelectual, no exclusiva e intransferible, la cual constará por escrito, y que se registrará por las estipulaciones del contrato respectivo y por las atinentes a las enajenaciones o cesiones de derechos, en cuanto sea pertinente.”*

De la caracterización normativa anterior podemos rescatar y comentar los elementos siguientes:

---

<sup>1</sup> OMPI, “La concesión de licencias sobre derechos de propiedad intelectual: un componente vital de la estrategia comercial de su PYME”. En la publicación “La propiedad intelectual para las pequeñas y medianas empresas”, *op. cit.*

- El otorgamiento de licencias de uso es conceptualizado como una opción alternativa (“... *podrá sustituir* ...”) a la cesión total o parcial de los derechos patrimoniales del autor sobre la obra. Sin embargo, esta alternativa es, en la práctica, la regla y no la excepción. La popularidad del licenciamiento respecto de la cesión de derechos obedece precisamente a que el primero permite a los desarrolladores proteger eficazmente la propiedad intelectual de sus productos, limitar su responsabilidad por los daños que puedan surgir de aplicaciones posiblemente defectuosas y restringir el uso del software para maximizar sus oportunidades de lucro.<sup>2</sup>
- Como regla de principio, la licencia que se confiera debe entenderse como no exclusiva e intransferible. Decimos “como regla de principio” porque, al encontrarnos en el terreno del derecho privado, nada impediría que convencionalmente se acuerde otra cosa, por ejemplo, en cuanto a la transmisibilidad de la licencia. La no exclusividad evidentemente implica que, si la naturaleza de la obra lo permite (y decididamente así es en el caso del software), el titular puede otorgar licencias de uso múltiples sin que el o los licenciarios previos puedan oponerse a ello.

En la licencia –o *licenciamiento*, como también se le conoce– de uso de software, su creador o fabricante retiene en todo momento la titularidad dominical sobre la aplicación y solamente autoriza al licenciario para usar una copia.

La licencia ordinariamente otorga derecho al uso de una o unas pocas copias del programa.<sup>3</sup> Si nada se dijere al respecto en la licencia, necesariamente tendremos que interpretar que solo se autorizó el empleo de una copia individual.

Las empresas proveedoras de software acostumbran brindar facilidades para la adquisición de licencias múltiples, por ejemplo, para fines corporativos. Pero, en todo caso, la instalación<sup>4</sup> de un número de copias de la aplicación en

<sup>2</sup> LEVINE, Jeffrey A. “*Shrink-wrap Licenses - Shrinking Protection*”, 1996. Documento electrónico disponible en <http://www.swiggartagin.com/aigc/tic64.html>.

<sup>3</sup> Aunque cabe reiterar aquí las advertencias que ya hicimos en el capítulo precedente, al discutir la aplicación de los principios tradicionales del derecho autoral al software, en cuanto a que – en la práctica– el empleo de aplicaciones informáticas necesariamente implica la generación, aunque fuere transitoriamente, de múltiples instancias del programa (en disco y en la memoria principal).

<sup>4</sup> La “instalación” de un software es un proceso requerido por virtualmente la totalidad del software distribuido en la actualidad. En el pasado, era frecuente que una aplicación pudiese ser ejecutada directamente desde un medio de almacenamiento removible (por ejemplo, un disquete). Hoy, por el contrario, esto no suele ser viable, no solo debido a que el tamaño de los archivos respectivos puede exceder la capacidad del medio de distribución, sino que –

exceso de las permitidas por la licencia –o bien la instalación de aunque fuere una sola copia no amparada a una licencia válida– conllevará el quebranto de las respectivas leyes sobre propiedad intelectual. Como ya se ha indicado *supra*, nuestro ordenamiento (como la mayoría en la actualidad) considera a esta conducta como sancionable penalmente. Al respecto, la “Ley de Procedimientos de Observancia de los Derechos de Propiedad Intelectual” indica, refiriéndose a las obras tuteladas por el derecho de autor en general:

*“Artículo 54.- Reproducción no autorizada de obras literarias o artísticas, fonogramas o videogramas.*

*“Será sancionado con prisión de uno a tres años quien fije y reproduzca obras literarias o artísticas, fonogramas o videogramas protegidos, sin autorización del autor, el titular o el representante del derecho, de modo que pueda resultar perjuicio.”*

– 0 –

Desde el punto de vista formal, identificamos tres vías comunes para el perfeccionamiento del contrato de licencia de uso de software:

- Licencia convencional: es la que se concierta y formaliza por escrito, como cualquier otro convenio, normalmente en el contexto de un contrato de desarrollo de software a la medida en el que no hay transferencia de derechos al usuario.<sup>5</sup>
- Licencia *shrink-wrap*: en esta modalidad, típica del software comercial, el clausulado viene estampado, bien en la caja de distribución del producto o –más frecuentemente aun– en un documento que se agrega al contenido, ya sea al inicio del manual del usuario o en hoja aparte. Se le denomina de esta manera por asociación con el plástico adhesivo mediante el cual se sella el paquete y cuyo rasgado es interpretado

---

además– la norma es que la correcta operación del programa probablemente requiera efectuar ciertas operaciones de configuración del equipo, junto con la descompresión y copiado en el disco duro del usuario de los archivos del programa y de sus librerías (aplicaciones complementarias) asociadas. Se acostumbra automatizar este proceso mediante la ejecución de un programa independiente, llamado *instalador*. La ejecución de este programa y la exitosa realización de esas diversas actividades preparatorias es lo que se denomina “instalar” un software.

<sup>5</sup> Caso contrario, nos remitiremos a lo que se indica en la sección siguiente.

como una aceptación tácita de los términos de la licencia.<sup>6</sup> Su creación, a finales de la década de 1970, se atribuye a Seymour Rubinstein.<sup>7</sup>

- Licencia *click-on* o *click-wrap*: cuando el software es recibido electrónicamente (por ejemplo, descargándolo de Internet), lo usual es que el clausulado de la licencia sea desplegado al usuario del mismo modo. Una posibilidad es que aparezca en forma de página web; otra, que se muestre en pantalla durante el proceso de instalación. En ambos casos, el usuario deberá hacer clic en un botón como señal de aceptación de los términos de la licencia, antes de que el proceso de descarga o de instalación continúe. El nombre de “click-on” o “click-wrap” deriva precisamente de ese gesto de consentimiento simbólico.

Por su propia naturaleza, los contratos de licenciamiento de software concertados bajo las últimas dos modalidades citadas poseen un carácter evidentemente adhesivo, lo cual se presta en ocasiones para la creación de un notorio estado de desequilibrio entre los derechos del proveedor y los del consumidor. La situación es especialmente grave en el caso del software que se distribuye ya instalado al momento de comprar una computadora nueva, puesto que –como es obvio– en estos casos, el consumidor ni siquiera ha tenido oportunidad de leer la licencia y aceptar sus términos antes de recibir el producto. La naturaleza adhesiva del contrato es puesto aquí de relieve en el hecho de que, en la mayoría de estas hipótesis, el consumidor tiene poco margen de elección, ya que la preinstalación de ciertos paquetes de software suele establecerse como componente integral de equipos preconfigurados y vendidos como unidad.

Como señala correctamente MARESCA,

*“Las condiciones monopolísticas del mercado, en donde determinadas compañías dominan en forma absoluta su área, ha tenido como resultado la creación de contratos en donde el usuario ha perdido prácticamente todo su poder para negociar, debiéndose adherirse a los términos predispuestos por la otra parte. Esto es lo que ocurre con las licencias de uso, las cuales son redactadas*

<sup>6</sup> En estos casos, la licencia o alguna etiqueta visible externamente suele especificar algo como lo siguiente: “Por el hecho de abrir este paquete, rompiendo la envoltura plástica e instalando el software en su computadora, usted manifiesta que ha leído y que comprende los términos y condiciones fijados en esta licencia y que consiente sujetarse a todas ellas”.

<sup>7</sup> DVORAK, John C. “The software protection racket”, en la revista *PC Magazine*, 20 de agosto de 1998. Versión electrónica disponible en [http://www.zdnet.com/pcmag/insites/dvorak\\_print/jd980806.htm](http://www.zdnet.com/pcmag/insites/dvorak_print/jd980806.htm).

*unilateralmente e importan un verdadero desequilibrio en la relación contractual.*<sup>8</sup>

Como es de esperar, esta circunstancia ha dado pie a diversas batallas judiciales, algunas de las cuales han sido resueltas a favor del usuario consumidor, particularmente en los casos en que los términos de la licencia no eran suficientemente precisos en su literalidad o consecuencias.

– o –

Es importante recordar en este punto que, conforme a lo que disponen la mayoría de las licencias de software (y que está reafirmado en nuestro ordenamiento por el artículo 9 del Reglamento a la Ley de Derechos de Autor, como se comentó en el capítulo correspondiente), el adquirente o licenciataria puede producir al menos una copia de respaldo del software, exclusivamente para fines de seguridad, sin costo adicional. Esta práctica se origina en las épocas en que el software se distribuía en disquetes o cintas magnéticas que se dañaban rápida o fácilmente, pero que se continúa dando hoy por razones de seguridad informática básica.

– o –

El licenciataria de software comercial como regla no tiene acceso al código fuente y, por ende, no tiene derecho a modificarlo en modo alguno, dependiendo por completo del proveedor para el mantenimiento y futuras mejoras del programa. De hecho, las licencias normalmente incorporan una prohibición expresa de descompilación o desensamblaje (“ingeniería reversa”) del producto.

Ahora bien, es necesario detenernos momentáneamente y examinar con cuidado esta última restricción, a fin de no otorgarle una extensión inapropiada. En efecto, la ingeniería reversa es una técnica que persigue restituir el código fuente de una aplicación, a partir de su código objeto (que, como se explicó en su momento, consiste de ceros y unos). Esta operación se realiza, por ejemplo, para rescatar el código fuente en caso de que éste se hubiere perdido; para analizar la forma en que el programa realiza ciertas operaciones; para mejorar su desempeño; para corregir un defecto; para identificar la posible existencia de un virus informático en el programa, o bien para adaptar una aplicación escrita para un determinado microprocesador a otro.<sup>9</sup>

<sup>8</sup> MARESCA, Fernando. “Reflexiones sobre la comercialización del software”, en *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 20, marzo del 2000.

<sup>9</sup> Definición traducida de la original en inglés de WhatIs.com (<http://www.whatis.com>).

Pero además, bajo un régimen de protección jurídica del software basado en el derecho autoral –donde lo que se protege es la expresión de la obra y no sus particularidades de diseño y funcionamiento interno– la ingeniería reversa constituye una modalidad legítima de acceso al código fuente para fines de estudiarlo y proceder luego al diseño y creación de productos competitivos. Se trata sin duda de una interesante facultad, ya que, en la práctica, constituye una forma legítima de acceder a lo que, de otro modo, se consideraría y protegería como un secreto comercial.<sup>10</sup> En efecto, conforme a la legislación de la materia, existiría una infracción al secreto comercial solo si una persona o empresa le da un uso ilegítimo o divulga la información en violación de un pacto de confidencialidad; o si la obtiene por medios antijurídicos (soborno, espionaje industrial, etcétera); o si la recibe ilegalmente de un tercero. Por ende, examinar el funcionamiento interno de un software por medio de la ingeniería reversa, a fin de comprenderlo y poder visualizar así la manera de crear un producto competidor superior (evidentemente que no basado en el código fuente del original), constituye una técnica legítima, que garantiza un adecuado equilibrio de intereses y el beneficio de los consumidores. Entenderlo de otro implicaría otorgar a los regímenes de derechos de autor y de información no divulgada una extensión propia del sistema de patentes.<sup>11</sup> En este sentido, dispone por ejemplo el artículo 100 de la Ley de Propiedad Intelectual española que

*“el usuario legítimo de la copia de un programa 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 durante cualquiera de las operaciones de carga, visualización, ejecución, transmisión o almacenamiento del programa que tiene derecho a hacer”*<sup>12</sup>

Entonces, lo que en definitiva resulta prohibido bajo la legislación de propiedad intelectual –y es así como se manifiesta en los respectivos contratos de licencia– es la ingeniería reversa que se practique meramente con el propósito de alterar o copiar un programa, por ejemplo, para introducir parte de su código en otra aplicación.

– 0 –

<sup>10</sup> SAMUELSON, Pamela. “Reverse engineering under siege”, en *Communications of the ACM*, volumen 45, número 10, octubre del 2002, página 15.

<sup>11</sup> *Idem*. La autora cita al efecto un interesante fallo de la Corte Suprema de Justicia de los EE. UU., donde se reafirma que “en tanto un producto en circulación general no esté protegido por una patente, la reproducción de un atributo funcional suyo [por parte de otro producto] constituye una actividad competitiva legítima”.

<sup>12</sup> Citada por ERDOZAIN, *op. cit.*

En adición a las disposiciones anteriormente examinadas, las licencias de uso de software generalmente contienen cláusulas relativas a la legislación y jurisdicción aplicable en caso de conflicto, así como a si se permite o no su transferencia a terceros.

A este último respecto, ya señalamos que, en nuestro ordenamiento se parte de la noción de que la licencia es intransferible como regla. En la práctica, sin embargo, se observa que la mayoría de las licencias estándar en la industria del software admiten la cesión a terceros, bajo determinadas condiciones que allí mismo se regula y que, en resumen, suelen incluir las siguientes:

- Sólo es posible la cesión como tal. Es decir, no se admite arrendarlo o prestarlo.
- La transferencia debe incluir todas las copias, los disquetes o discos de instalación y la documentación escrita del software, incluyendo todos los manuales y la propia licencia impresa, si la hubiere.
- El cedente debe eliminar todas las copias del software que haya instalado.

No obstante lo anterior, existe una importante restricción a la transmisibilidad de la licencia, que corresponde a los casos en que lo que se adquiere es una actualización<sup>13</sup> de un producto de software. En estos casos no estamos ante una distribución completa del paquete, sino solo de los archivos necesarios para actualizar una versión previa. En estas hipótesis, se considera que el producto original y la versión de actualización conforman en conjunto una única unidad de software, por lo que no es posible transferirlos separadamente.

## **Cesión de derechos patrimoniales del software**

En materia de propiedad intelectual, la cesión de derechos es el convenio por medio del cual se traslada a terceros –a título gratuito u oneroso– alguno, algunos o bien la totalidad de los derechos patrimoniales del autor de una obra protegida.<sup>14</sup>

La cesión de derechos no es muy frecuente respecto de las aplicaciones comerciales de productividad, salvo en la hipótesis de compra o fusión de empresas completas (en cuyo caso las aplicaciones propiedad de la empresa adquirida o refundida presumiblemente se transmitirán a la adquirente o

<sup>13</sup> “Upgrade”, en inglés.

<sup>14</sup> Recuérdese que los derechos morales son intransferibles.

superviviente junto con el resto de los activos) o bien en las mega adquisiciones de productos de software de una empresa pequeña por parte de otra más grande que desea incorporarlos a su portafolios comercial. La cesión se puede encontrar más frecuentemente como una cláusula accesoria en el desarrollo externo de software hecho a la medida, figura a la que nos referiremos en la sección siguiente.

A diferencia de lo que acontece con respecto al licenciamiento de software, la cesión –por su propia naturaleza– carece del carácter adhesivo que normalmente tiene aquél.

Es importante advertir que, en la actualidad, el software es frecuentemente producido a partir del uso de ciertas herramientas de desarrollo que, en sí mismas, podrían constituir aplicaciones protegidas por otras personas o empresas;<sup>15</sup> o bien emplear *componentes reutilizables* desarrollados internamente o adquiridos de terceras partes. Un componente es, en síntesis, un segmento de código ejecutable que cuenta con una interface conocida.<sup>16</sup> En estos casos, la cesión del software desarrollado a partir de esas herramientas o componentes puede o no requerir el pago de regalías por la incorporación al producto final de las librerías (conocidas también como “archivos *runtime*”) o componentes requeridos. Estos aspectos deben clarificarse al momento de negociar los términos de la transferencia del software. En efecto,

*“Los desarrolladores y los terceros comercializadores [de componentes] deben diseñar contratos de licenciamiento apropiados que documenten el uso correcto y la responsabilidad del desarrollador en caso de que los componentes no operen conforme a la especificación acordada. En el caso de clientes específicos, el desarrollador y el cliente, o el ensamblador, deberían convenir formalmente a quién pertenecen los componentes y demás productos. En el escenario del mercado abierto, los desarrolladores de componentes confrontan algunos de los mismos problemas de propiedad intelectual que los desarrolladores de aplicaciones convencionales.”<sup>17</sup>*

Cabe subrayar que, así como no es posible admitir una renuncia implícita de los derechos patrimoniales del autor de un software protegido (tema que

<sup>15</sup> Los llamados “lenguajes de cuarta generación” o “herramientas CASE”.

<sup>16</sup> VITHARANA, Padmal. “Risks and challenges of component-based software development”. En *Communications of the ACM*, volumen 46, número 8, agosto del 2003, pág. 67. Véase también RAVICHANDRAN, T. y ROTHENBERGER, Marcus A., “Software reuse strategies and component markets”. En *Communications of the ACM*, volumen 46, número 8, agosto 2003, página 109.

<sup>17</sup> Idem, págs. 69-70.



discutimos en el capítulo pertinente), tampoco procede pensar en una transferencia de derechos que no haya sido pactada explícitamente. Por ende, en caso de duda, lo correcto será interpretar los términos del respectivo convenio en el sentido de que no hay transmisión de derechos, ya que ésta no fue expresamente acordada.

Algunas obligaciones importantes del cedente, en esta materia, incluirán:

a.- Entregar el código fuente, junto con todas las especificaciones de diseño lógico y físico del software. Sin el código fuente y las especificaciones de diseño, las actualizaciones o correcciones a la aplicación que desee hacer el cesionario en el futuro resultarán muy difíciles. Sin embargo, ya advertimos anteriormente que el software moderno suele incorporar librerías y componentes que pueden pertenecer a terceras partes, por lo que es oportuno aclarar que el código que deberá entregar el cedente se refiere solamente al desarrollado por sí mismo. Aun así, también es posible que alguno o algunos de esos componentes hayan sido creados por el propio cedente (si es, a la vez, el desarrollador), pero sin el propósito de que transferencia del producto final los incluya a ellos también, posiblemente, por ejemplo, porque se trate de herramientas reutilizables que desee continuar empleando en otros proyectos futuros. Incluso podría ocurrir que dichos componentes sean valorados como un secreto comercial de la empresa desarrolladora.<sup>18</sup> En tales casos, es perfectamente viable que se acuerde que la empresa programadora se limitará a documentar y entregar al cliente solamente el detalle de la *interface pública*<sup>19</sup> del componente. Claro está, deberá establecerse entonces alguna regulación contractual para comprometer a la empresa a depurar la aplicación en el evento de que el componente –debido a su programación interna– dé lugar a una incorrecta operación del programa.

b.- Garantizar al cesionario contra reclamos de terceros por la eventual violación de derechos de propiedad intelectual. Si en el desarrollo del software cedido se ha hecho uso ilícito de rutinas o herramientas protegidas por terceros, es concebible que surjan disputas futuras en torno al empleo de la aplicación adquirida. En este caso, es menester reconocer la existencia de una especie de garantía de evicción, por virtud de la cual el desarrollador deberá responder por los posibles daños y perjuicios que deriven de ese hecho.

Por su parte, incumbirá esencialmente al cesionario:

<sup>18</sup> Por ejemplo, si incorporan un algoritmo novedoso, haya sido patentado expresamente o no.

<sup>19</sup> En la terminología de la moderna programación orientada a objetos, se denomina “interface pública” a la especificación de los métodos que el componente expone al programador, con su respectiva sintaxis y argumentos.

a.- El pago del precio convenido. Obviamente es el principal deber del adquirente del software.

b.- Respetar los derechos morales del desarrollador, no intentando ocultar o desvirtuar su autoría original del software.

## Desarrollo de software a la medida

Al ensayar una clasificación jurídica del software, se estableció en el capítulo primero que si atendemos como criterio de distinción lo relativo al grado de estandarización o uniformidad del software, podemos diferenciar el *software estándar* del que es *hecho a la medida*. Dijimos que el segundo es aquel que es producido específicamente contra pedido y para las necesidades concretas de un usuario o usuarios, que por cualquier motivo no encuentran en el software estándar la solución a sus necesidades particulares.

En efecto,

*“La celebración de un contrato de desarrollo de un programa de computación parte del presupuesto de que una persona o empresa se halla ante la necesidad de automatizar el tratamiento de datos en un modo o con un objetivo particular, y que los programas de computación que conoce como que están disponibles en el mercado no satisfacen dicha necesidad, o no pueden ser utilizados en el sistema informático de esa persona o empresa, o son accesibles a un costo demasiado elevado o su calidad no alcanza los standards pretendidos.”<sup>20</sup>*

Un software hecho a la medida puede ser desarrollado, naturalmente, por el propio personal de planta de una empresa, en el contexto de una relación laboral asalariada. No obstante, nos interesa aquí más bien el caso de aquellas aplicaciones cuya preparación es contratada más bien a un tercero o empresa independiente.<sup>21</sup>

<sup>20</sup> MARTORELL, Ernesto Eduardo. Citado por VARGAS VÁSQUEZ, Elizabeth, en *“Contratos informáticos de desarrollo de software a la medida”*. Tesis para optar al grado de licenciada en Derecho, Universidad Internacional de las Américas, San José, 2002, pág. 68.

<sup>21</sup> Lo que en la jerga se suele denominar *“outsourcing”*. Véase al respecto ROMERO PÉREZ, Jorge Enrique, *“La externalización de actividades laborales (outsourcing)”*. En *Revista de Ciencias Jurídicas*, número 102 (setiembre – diciembre 2003). Universidad de Costa Rica – Colegio de Abogados de Costa Rica, página 119.

En este sentido, el desarrollo a la medida de software se caracteriza claramente como un contrato de obra, que puede conllevar o no la cesión de los derechos patrimoniales del producto resultante. Caso afirmativo, se aplicará lo dicho en la sección precedente. Pero en caso de que no lo haga, estaremos más bien ante un licenciamiento de uso de software, en cuya hipótesis nos remitiremos más bien a lo detallado en el acápite trasanterior.

En cuanto a las principales obligaciones de las partes (el desarrollador de una parte y la persona o empresa comitente, de otra), tendremos:

#### Obligaciones del desarrollador:

a.- Informar y asesorar al comitente sobre los requerimientos técnicos del software desarrollado. Claramente, nadie conoce mejor las especificaciones técnicas de la aplicación o aplicaciones que son objeto del convenio, que su creador. Por ende, a éste incumbe dar a conocer qué clase de plataforma de hardware y software se requiere para la adecuada operación de su programa (disponibilidad de memoria RAM, espacio libre en disco, versión de sistema operativo, velocidad de procesador, etcétera). Este es, desde luego, una aplicación concreta del principio general de “informar, prevenir y aconsejar” que recae sobre la parte técnicamente más versada y que es propio de los diversos contratos informáticos en general. En efecto,

*“El proveedor –prestador, dador– tiene la obligación de carácter general de información y consejo a su potencial cliente, usuario o tomador, quien como dijimos está en un plano inferior en cuanto a la posibilidad de apreciar adecuadamente sus necesidades a nivel técnico. Esto significa que el proveedor de bienes y servicios informáticos está obligado a informar y advertir al cliente todo cuanto sea posible en relación a la calidad, prestaciones, rendimiento, utilización riesgos, ventajas y desventajas de aquello que va a adquirir. Sobre esta base, deberá aconsejar al cliente en la correcta elección de los equipos, programas y accesorios, teniendo siempre como principio rector a la buena fe. Cabe aclarar, sin embargo, que cuanto mayores sean los conocimientos técnicos del cliente, menor será la responsabilidad del proveedor, ya que éste en cierta medida ‘obedecerá’ las indicaciones de aquél.”<sup>22</sup>*

<sup>22</sup> GHERSI, Carlos Alberto. *Contratos Civiles y Comerciales*, Parte General y Especial, Tomo II, segunda edición, Editorial Astrea, 1992, pg. 309-310). Este principio aparece expresamente incorporado como tal en la jurisprudencia de nuestra Sala Primera de la Corte Suprema de Justicia. Véase, por ejemplo, la sentencia número 117-F-98 de las 14:50 horas del 11 de noviembre de 1998.

Además,

*“[El consejo] debe ser objetivo y profesional, y no parcial o tendencioso. El cliente no debe ser inducido a tomar una acción, pues como resultado de ese asesoramiento se pueden determinar las acciones futuras del cliente. Debe quedar claro que asesorar significa brindar alternativas técnico económico–jurídicas, adecuadas a las necesidades del cliente.”*<sup>23</sup>

b.- Instalación, pruebas, migración de datos y capacitación. Si el software ha sido desarrollado a la medida, será necesario instalarlo en la computadora o computadoras del comitente y correr las pruebas preestablecidas. Si, además, la aplicación fue diseñada para sustituir a otra que se encuentra en producción, probablemente se querrá trasladar los datos contenidos en la última al nuevo sistema, de manera que no se pierda información existente que representa un importante activo organizacional. En ese mismo sentido, habrá que brindar al adquirente un determinado nivel de capacitación en el funcionamiento de la aplicación cedida, ya sea en forma de manuales de usuario, instrucción al personal o ambas cosas.

c.- Entregar el código fuente y las especificaciones de diseño lógico y físico del software, como se indicó en cuanto a la cesión de derechos y por las mismas razones.

d.- Otorgar garantía de (razonable) buen funcionamiento. Por motivos sobre los cuales ahondaremos en el capítulo siguiente, el complejo software que se suele desarrollar y distribuir en la actualidad está sujeto a contener defectos que escapan a la detección durante las etapas de diseño y depuración. Desde esta óptica y mientras las condiciones no cambien, es irreal pretender garantizar un desempeño ciento por ciento libre de errores al momento de la entrega del software. Por ese motivo, es usual convenir que el desarrollador deberá brindar servicios de mantenimiento correctivo de la aplicación durante un período estipulado. La extensión de ese plazo dependerá de las necesidades de las partes. Ahora bien, se podría discutir si por “buen funcionamiento” se entiende nada más que la aplicación satisfaga la funcionalidad pactada o si además debe hacerlo con ciertas características de rendimiento mínimo (por ejemplo, ejecutar un determinado número de transacciones por unidad de tiempo o satisfacer las exigencias de alguna prueba estándar de la industria<sup>24</sup>). Nuestra recomendación es que si el rendimiento mínimo es crítico para la parte que recibirá y usará el software, lo

<sup>23</sup> VARGAS VÁSQUEZ, *op. cit.*, pág. 80.

<sup>24</sup> Conocidos en inglés como *benchmarks*.

mejor será preverlo explícitamente así en el convenio, estipulando los mecanismos de verificación más objetivos posibles.

e.- Proteger los secretos industriales o comerciales a los cuales llegue a tener acceso durante la realización del trabajo, incluyendo listas de clientes, planes de expansión, cantidad de empleados, datos privados de los mismos, etcétera.<sup>25</sup>

f.- Garantizar al adquirente contra las reclamaciones de terceros en caso de infracción de sus derechos de propiedad intelectual. En efecto,

*“... el proveedor debe garantizar a través del contrato, que el ‘software’ que distribuye es un trabajo original suyo, con el objeto de salvaguardar la responsabilidad del cliente en el uso de la aplicación, de modo que no se estén infringiendo derechos de autor o patentes de terceras personas. El proveedor no debe incorporar al ‘software’ que distribuye, partes de otras aplicaciones para las que no cuenta con la autorización que corresponde. Si lo hace, debe asumir las consecuencias legales que esto pueda generar, y el cliente debe tener el derecho de suspender los pagos si llega a conocer que ha recibido un sistema o partes de éste, sin que el proveedor original haya dado su autorización.”<sup>26</sup>*

De parte del comitente y, desde luego, en adición al pago oportuno del precio en las condiciones pactadas, se deberá:

a.- Disponer de una plataforma de hardware y software apropiada para el sistema desarrollado. Habiendo informado y asesorado previamente el desarrollador sobre los requerimientos técnicos del software, es de esperar que al momento de la entrega, el comitente satisfaga los requerimientos expuestos. Lo contrario exonerará al primero contra reclamos por una posible incompatibilidad o bajo rendimiento de la aplicación, en la medida de que ha sido el segundo quien no habrá posibilitado el correcto cumplimiento de las obligaciones contraídas.<sup>27</sup>

b.- Extender constancia de aceptación. En el desarrollo de software hecho a la medida, es importante establecer un momento en que se tiene por aceptado a

<sup>25</sup> FUENTES PINZÓN, Fernando, “Contratación para la elaboración del software individualizado. Estudio comparado de la legislación iberoamericana.” En *Revista de Ciencias Jurídicas*, número 102 (setiembre – diciembre 2003). Universidad de Costa Rica – Colegio de Abogados de Costa Rica, página 91.

<sup>26</sup> VARGAS VÁSQUEZ, *op. cit.*, pág. 81.

<sup>27</sup> Este principio ha sido recogido por la jurisprudencia nacional, por ejemplo, en la sentencia número 376 de las 9:00 horas del 20 de setiembre del 2000, dictada por el Tribunal Segundo Civil de San José, Sección Segunda.

satisfacción el producto. Aparte, desde luego, de una manifestación expresa en ese sentido, la aceptación también puede darse tácitamente, por ejemplo, por el transcurso de un determinado plazo contractualmente previsto, sin que el comitente formule reparos o reclamos a la aplicación; o bien por el mero hecho de ponerla en producción, lo cual implica destinar el software al uso normal previsto, trabajando con datos reales y ya no puramente de prueba.

– o –

De acuerdo con FUENTES PINZÓN,<sup>28</sup> es importante que al momento de concertar un contrato de desarrollo de software a la medida, se convenga oportunamente sobre algunos aspectos relevantes del proceso, incluyendo el lugar donde se realizará el trabajo, las veces que puede trabajar el o los programadores con el hardware del comitente antes de las primeras pruebas y el tiempo estimado que demorará la instalación del producto terminado.

## **ASP: arrendamiento en línea de aplicaciones**

Las contrataciones ASP constituyen una modalidad comercial surgida de la mano del crecimiento de la Internet y del perfeccionamiento de las tecnologías telemáticas, por un lado, así como de la necesidad de las empresas de hacer un uso más eficiente de sus recursos, por otro.

El término ASP deriva del inglés *application service provider* (proveedor de servicios de aplicaciones).<sup>29</sup> En lo medular, la prestación que es objeto del contrato consiste en ofrecer el acceso por medio de la Internet (o de una red privada) a aplicaciones y servicios relacionados, que de otro modo tendrían que residir en el equipo propio del arrendatario. En el modelo ASP, el proveedor hospeda aplicaciones de escritorio desde una “granja de servidores” (*server farm*) localizada en su centro de datos (*data center*). Los usuarios de una o más organizaciones cliente acceden a esas aplicaciones ya sea por medio de una conexión de red privada de área ancha (WAN) o de la Internet.<sup>30</sup>

Es decir, el proveedor ASP se presenta como una empresa que ofrece la puesta a disposición y la administración de aplicaciones por medio de la red, a

<sup>28</sup> FUENTES PINZÓN, *op. cit.*, página 97.

<sup>29</sup> No confundir con la tecnología ASP (*active server page*) de la empresa Microsoft, utilizada para la generación dinámica de páginas web; ni con “ISP”, que significa *Internet service provider* o proveedor de servicios de acceso a la Internet. Ejemplo: RACSA en Costa Rica.

<sup>30</sup> WALSH, Kenneth R., “Analyzing the Application ASP Concept: Technologies, Economies, and Strategies”. En *Communications of the ACM*, volumen 46, número 8, agosto 2003, página 103.

cambio de una tarifa calculada por unidad de tiempo, por el número de usuarios u otro parámetro similar.<sup>31</sup>

*“Ello supone reducir enormemente los costes de administración de (los) sistemas informáticos, permitiendo al empresario dedicar la práctica totalidad de sus recursos, humanos y económicos, a la actividad que le es propia, lo que a la larga se traduce en una mayor rentabilidad y capacidad competitiva de su empresa.”<sup>32</sup>*

*“Entre los beneficios clave del modelo ASP están las economías de escala que permiten al ASP operar un centro de datos seguro y confiable a un menor costo por usuario. Para las pequeñas o medianas organizaciones, el ASP puede ofrecer mayores niveles de seguridad y confiabilidad que una organización cliente. El ASP obtiene esta ventaja por medio de distribuir los costos de soluciones innovadoras entre muchos clientes. Los ASP también se benefician al contratar expertos en tecnología talentosos que muchas compañías pequeñas no podrían costear o a los que subutilizarían. La empresa cliente también se beneficia de la relación ASP en cuanto el ASP la mantiene actualizada con las últimas tecnologías.”<sup>33</sup>*

Existen varias implementaciones de servicios ASP. En la práctica, podemos identificar al menos dos modalidades preponderantes:

- En su concepto originario (y que es el que propiamente corresponde con un arrendamiento de software en línea), el ASP se ofrece como alternativa al licenciamiento tradicional de aplicaciones, ya que es el proveedor y no los clientes quien obtiene las licencias y luego cobra a los segundos únicamente por el uso efectivo que se haga del software (o sea, por lapsos determinados o por la cantidad de usuarios). Esto adquiere mucho sentido cuando al cliente le resulte impráctica la alternativa del licenciamiento, ya sea por su alto costo o del uso escaso que se haga del producto en cuestión (por ejemplo, una herramienta avanzada de diseño y desarrollo de aplicaciones).

<sup>31</sup> LEE, Jae-Nam y otros. “IT outsourcing evolution – past, present, and future”. En *Communications of the ACM*, volumen 46, número 5, mayo del 2003, página 88.

<sup>32</sup> CARRASCO, Juan. “ASP, una compleja relación jurídica”, en *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 35, junio del 2001. Los paréntesis no son del original.

<sup>33</sup> WALSH, *op. cit.*

- Bajo la modalidad de *data center*, el cliente adquiere las licencias respectivas del software, pero recurre al proveedor ASP para que las instale, corra y mantenga, reduciendo así sus costos administrativos.<sup>34</sup>

Sea cual fuere la opción, este convenio se presenta entonces como una figura híbrida entre los contratos de software y los de servicios telemáticos. Precisamente, una de las áreas actuales de mayor fortaleza se encuentra en la prestación de servicios de software vía la web.<sup>35</sup>

El proveedor ASP crea un centro de servicios en línea, desde el cual pone a disposición de sus clientes un conjunto de aplicaciones informáticas, así como la infraestructura necesaria para la administración y gestión de los servicios de tecnología de información de sus empresas.<sup>36</sup> En el paquete más completo, una organización solo retiene aplicaciones cliente mínimas (*thin clients*) y servicios de red local, mientras que el ASP provee aplicaciones y conexiones de Internet.

*“El ASP consiste en alojar, actualizar y mantener, por una empresa ‘A’ (la empresa que presta el servicio ASP), determinadas aplicaciones informáticas de una empresa ‘B’ (la empresa cliente). Dicho alojamiento se realiza en un servidor o centro hosting, encontrándose las mismas a disposición de los usuarios, los cuales podrán conectarse remotamente desde cualquier equipo terminal que se encuentre conectado a Internet o a una red privada. De esta forma, a través del almacenamiento de las aplicaciones por parte del ASP, se alivia el trabajo a los administradores de sistemas informáticos.”*<sup>37</sup>

El carácter bilateral de la contratación ASP no impide la eventual inclusión de terceras partes, tales como otras empresas o personas desarrolladoras de software, a quienes se abre así nuevos canales para la distribución de sus productos.

*“En este supuesto, bastante habitual, la relación a tres bandas se establece, por un lado, entre la empresa desarrolladora del software y la empresa que presta el servicio ASP (A) y, por otro, entre la empresa que presta el servicio ASP y la empresa cliente (B), de modo que, indirectamente, surge una relación entre la empresa desarrolladora del software y la empresa cliente (C). Y es esta última*

<sup>34</sup> GRIMES, Brad. “The At-Your-Service ASP”, en *PC Magazine*, volumen 22, número 21, 25 de noviembre del 2003, página 70.

<sup>35</sup> *Idem.*

<sup>36</sup> CARRASCO, *op. cit.*

<sup>37</sup> *Idem.*



*relación la que debe quedar perfectamente reflejada en la Licencia de Uso ASP, ya que la empresa cliente es la que va a hacer uso de las aplicaciones informáticas de la empresa desarrolladora, aplicaciones que están instaladas en el servidor del ASP. En este supuesto, el ASP actuaría como canal o medio de distribución de las aplicaciones informáticas, llevando a cabo todas las labores de publicidad, marketing y contacto con las empresas clientes, limitándose las empresas desarrolladoras de software a las labores que les son propias.”<sup>38</sup>*

Debido a la complejidad de esta clase de acuerdos, es importante que el contrato respectivo clarifique diversos aspectos que, de otro modo, podrían dar pie a futuras discusiones y conflictos. Entre estos factores, conviene detallar todo lo relativo a los equipos empleados (en lo que a su titularidad y obligaciones se refiere), los derechos de autor respectivos y –si corresponde– el mantenimiento de las aplicaciones informáticas, ya fuere preventivo, correctivo o evolutivo. Conviene asimismo estipular las obligaciones del proveedor ASP en lo concerniente a disponibilidad del servicio, que debe ser alta frente a las eventuales caídas que pueda experimentar en sus sistemas y que, evidentemente, podrían dar lugar a una fuerte disminución de la productividad de sus clientes. En efecto,

*“La seguridad y la confiabilidad son probablemente las más importantes cualidades de desempeño de un ASP. (...) Una organización dependiente de un ASP será responsabilizada por sus clientes en el evento de un fallo de sistema y, por ende, necesita confiar fuertemente en el ASP.*

*Estos niveles de confianza pueden ser cimentados por medio de la certificación otorgada por expertos independientes. Algunas firmas de contaduría y consultoría están ofreciendo auditorías de seguridad para ASPs y otorgan certificación a aquellas que las aprueban.”<sup>39</sup>*

Algunos analistas estiman que los servicios ASP son la solución para los clásicos problemas de defectos en los programas (“pulgas”), obsolescencia e instalación, contribuyendo a reducir significativamente los costos de las empresas desarrolladoras. Al mantener las aplicaciones en sus propios servidores, los productores de software no tendrían que distribuir costosas actualizaciones en caso de que se requiera la rectificación de alguna deficiencia en el código.

---

<sup>38</sup> *Idem.*

<sup>39</sup> WALSH, *op. cit.*

Adicionalmente, se elimina la cadena de intermediación que actualmente es necesaria para distribuir las aplicaciones y se minimizan los problemas de piratería.

No obstante, los críticos sostienen que con el arrendamiento ASP, los consumidores podrían terminar pagando más dinero por cargos de conexión que el que invertirían en adquirir el software a través del licenciamiento tradicional. Por otra parte, señalan que la mayoría de los usuarios difícilmente tendrán acceso a los requerimientos de hardware y al ancho de banda necesarios para el acceso remoto de software (generándose así lo que los informáticos denominan una *latencia* excesiva en la transmisión de los datos), ni sentirán ninguna seguridad en el hecho de que sus datos dependan de aplicaciones que no tienen instaladas localmente.<sup>40</sup> En adición, si no se toman las previsiones necesarias por adelantado, podrían presentarse problemas de integración de sistemas entre el cliente y el ASP.

Por todo lo anterior, es importante tener claro que el modelo de contratación ASP en realidad es una síntesis de varios productos y servicios, incluyendo el software, la conectividad, el hospedaje web, el hardware, la integración de los sistemas, la supervisión de la red y de las aplicaciones, así como servicios ampliados de soporte y asistencia técnica.<sup>41</sup>

Finalmente, una última alternativa interesante de ASP a la que se refiere WALSH,

*“... es el outsourcing de sistemas interorganizacionales. Así como un ASP puede fungir como un distribuidor de aplicaciones, también puede serlo de datos, proveyendo una interface entre los sistemas de socios de negocios. A medida en que los ASP usan y desarrollan estándares para la computación interorganizacional, pueden reducir el tiempo y el costo de configurar tales sistemas, liberando a sus clientes para que establezcan negocios entre ellas. Al diseñar nuevos estándares tecnológicos de e-comercio B2B, el ASP transforma el proceso a uno en el que él se encarga de los estándares técnicos de múltiples organizaciones, dejando a ellas la cuestión de determinar con quiénes quiere compartir sus datos.”<sup>42</sup>*

<sup>40</sup> SANDBERG, Jared. “Goodbye, Shrink-Wrap: Software for Rent”. En la revista *Newsweek*, edición del 18 de setiembre del 2000, volumen 136, número 12, página 74C. Véase <http://www.msnbc.com/news/457651.asp>.

<sup>41</sup> LEE y otros, *op. cit.*

<sup>42</sup> WALSH, *op. cit.*

# Capítulo IV - Responsabilidad civil originada en el diseño y uso de software

## Introducción

La informática no opera en el vacío. La influencia que ejerce trasciende los dispositivos de hardware que ejecutan los programas, para alcanzar a personas y entidades reales. Y aunque desearíamos que ese efecto fuera siempre positivo, la realidad demuestra que puede ser todo lo contrario:

- En 1962, la misión espacial de la nave Mariner 1 al planeta Venus se malogró debido tan solo a una coma incorrecta en un programa en lenguaje Fortran.<sup>1</sup>
- Se atribuye a un defecto de software la caída de un avión, en la que murieron sus 159 ocupantes. El problema era conocido de antemano por los fabricantes, quienes no solo hicieron caso omiso sino que –peor aun– contribuyeron a magnificar las consecuencias del error al dotar al software de una mala interfaz de operación. Un jurado en la Florida, Estados Unidos, condenó a la empresa desarrolladora a pagar la suma de \$75 millones.<sup>2</sup>
- En 1997, una firma constructora estadounidense empleó un software especial para la preparación de ofertas en procesos licitatorios a fin de someter su cotización para el levantamiento de un nuevo centro médico en la ciudad de Seattle. Ganaron el contrato, tan sólo para descubrir –ya tarde– que el software había subestimado el precio a cotizar en \$1.95 millones.<sup>3</sup>

---

<sup>1</sup> LIEBERMAN, Henry y FRY, Christopher. "Will software ever work?", en *Communications of the ACM*, volumen 44, número 3, marzo del 2001, página 122.

<sup>2</sup> KAPPELMAN, Leon A. "Killer Apps & Dead Bodies", en *Information Week* de 26 de junio del 2000; disponible electrónicamente en <http://www.informationweek.com/792/92uwlk.htm>

<sup>3</sup> Véase el artículo noticioso escrito por HENNIGH, Scott E., en [http://www.constructionweblinks.com/Resources/Industry\\_Reports\\_\\_Newsletters/Sept\\_18\\_2000/defective\\_software.htm](http://www.constructionweblinks.com/Resources/Industry_Reports__Newsletters/Sept_18_2000/defective_software.htm), visitado el 20 de febrero del 2002.

- En algunos países, se ha popularizado el uso de paquetes de software de asesoría jurídica. Estas aplicaciones guían a los usuarios en la preparación de algunas clases de documentos jurídicos (por ejemplo, testamentos o convenios de divorcio), sin necesidad de solicitar los servicios de un abogado. A principios de 1999, un juez federal en Estados Unidos determinó que un paquete de esta clase, denominado *Quicken Family Lawyer*, caía en el ejercicio ilegítimo de la abogacía, ya que iba más allá de meramente proveer formularios legales para que los usuarios los llenaran, aventurándose a brindar consejos jurídicos que podían ocasionarles graves consecuencias legales. Aunque los fabricantes de esta clase de tecnología se amparan en la libertad de expresión y en la necesidad de disminuir el costo de acceso a la justicia como sustento de sus productos, sus críticos señalan que estas aplicaciones sobre simplifican la aplicación de la ley, al punto de poner en peligro a los consumidores. Eventualmente se aprobó una legislación en el estado de Texas que determina que la venta de estos productos no puede ser restringida, siempre que ellos adviertan prominentemente que su utilización no pretende sustituir el criterio de un abogado.<sup>4</sup>
  
- En el verano del 2003, un gran apagón eléctrico afectó a la costa este de los Estados Unidos y algunas áreas de Canadá. Los investigadores del incidente concluyeron luego que un defecto en el software de monitoreo y control del operador FirstEnergy provocó la caída de los sistemas de alerta que, de otro modo, hubieran permitido tomar acciones oportunas para evitar el posterior efecto en cascada que se dio. Los pobladores de ciudades tan importantes como Nueva York estuvieron largas horas sin fluido eléctrico, dejando a numerosas personas atrapadas en ascensores.<sup>5</sup>

Un estudio publicado en junio del 2002 por el Departamento de Comercio de los EE.UU. estima en \$59,5 mil millones al año el costo para la economía estadounidense por los defectos del software.<sup>6</sup> Y a medida que el comercio electrónico y los mercados globalizados se desarrollan, los problemas probablemente se intensificarán con la multiplicación de las fallas a lo largo de

<sup>4</sup> NELSON, Sharon D. y SIMEK, John W. (editores). *Bytes in Brief*. Revista electrónica publicada por Sensei Enterprises, Inc. y el bufete Nelson & Wolfe, disponible en Internet (<http://www.senseient.com>). Edición de julio de 1999.

<sup>5</sup> JESDANUN, Anick. “*Software Bug Blamed for Blackout Alarm Failure*”, noticia divulgada el 12 de febrero del 2004 por la agencia AP y disponible electrónicamente en <http://www.siliconvalley.com/mld/siliconvalley/news/editorial/7941550.htm>

<sup>6</sup> “*Software errors cost billions*”. Despacho noticioso de la agencia Reuters, disponible electrónicamente en [http://story.news.yahoo.com/news?tmpl=story&cid=581&ncid=581&e=7&u=/nm/20020628/tc\\_nm/tech\\_software\\_dc\\_1](http://story.news.yahoo.com/news?tmpl=story&cid=581&ncid=581&e=7&u=/nm/20020628/tc_nm/tech_software_dc_1).

sistemas complejos y crecientemente interrelacionados. Estos errores pueden provenir de múltiples fuentes, incluyendo errores humanos o de software, hardware o redes inestables.

En los últimos tiempos, los defectos del software incluso han dado pie a preocupaciones en campos tan sensibles como la seguridad nacional de algunos países. En efecto, la amenaza del ciberterrorismo se incrementa en la medida en que son precisamente los defectos y vulnerabilidades del software lo que lo hace posible.<sup>7</sup>

## ¿Por qué falla el software?

*“... la mayoría de nuestros sistemas son más complejos de lo que podría considerarse saludable y son demasiado desordenados y caóticos para ser empleados con confort y confianza. El consumidor promedio de la industria informática ha sido servido tan pobremente que da por un hecho que su sistema se caerá todo el tiempo. Somos testigos de una distribución mundial masiva de software plagado de defectos, por el cual deberíamos avergonzarnos profundamente.”*

Edsger W. Dijkstra<sup>8</sup>

Desde la óptica de un programador, existen tres clases de errores que pueden afectar la correcta operación de un software:

- Los *errores de sintaxis* son, por lo general, los más simples de detectar y corregir. Ocurren cuando el desarrollador escribe una instrucción de un modo que contraviene las convenciones propias del lenguaje de programación empleado. En esencia, equivalen a los errores ortográficos en el lenguaje natural y pueden ser descubiertos automáticamente al momento de la compilación del código fuente, o incluso antes si el ambiente de desarrollo lo permite. Por ejemplo, escribir “prnt” en vez de “print”.
- Los *errores en tiempo de corrida* son más difíciles de evidenciar, porque surgen hasta el momento en que la aplicación es ejecutada. Bajo determinadas condiciones, es posible que el programa pretenda llevar a cabo una instrucción que es sintácticamente correcta, pero que conduce a una situación que hace imposible continuar o que incluso puede provocar la “caída” del software, cuando no del sistema informático como un todo. Podría ser que una sección específica de un programa sea ejecutada solo bajo de manera infrecuente (por ejemplo, cuando el

<sup>7</sup> Véase <http://www.newsfactor.com/perl/story/19104.html> (página visitada el 26/8/2002).

<sup>8</sup> DIJKSTRA, Edsger W. “The end of computing science?”, en *Communications of the ACM*, volumen 44, número 3, marzo del 2001, página 92. Traducción libre del autor.

año es bisiesto), o que la condición que produce el error sea externa (por ejemplo, cuando el programa intenta acceder a un archivo que normalmente existe pero que un momento posterior haya sido eliminado), o que se produzca una circunstancia inesperada (por ejemplo, un valor ingresado de manera incorrecta conduce a que se intente efectuar una división por cero). Es a esta categoría de errores la que suelen corresponder los diversos casos en que una aplicación se bloquea o se provoca su terminación abrupta.

- Finalmente, los *errores lógicos* son los más problemáticos de todos. En efecto, mientras que los errores sintácticos se evidencian fácilmente y los de tiempo de corrida tarde o temprano darán lugar a alguna conducta o mensaje por parte del software o de la computadora que los pondrá en descubierto, los errores lógicos actúan silenciosamente, produciendo resultados incorrectos para el usuario pero que, desde el punto de vista del software, no son ni más ni menos que los que éste fue programado para generar. Dicho de otro modo, los errores lógicos no son errores para el software, pero sí para el usuario. Supongamos un programa de cálculo de impuestos, en el que el programador introduce una tasa incorrecta para la determinación del monto del tributo. Digamos, 10% en vez de 15%, que es lo pertinente. El software posiblemente correrá a la perfección y, en lo que a él concierne, el resultado que genere será exacto. Pero obviamente no lo será para el usuario (y, sobra decirlo, mucho menos para la oficina de impuestos). Esta clase de errores son la pesadilla de los programadores, porque para descubrirlos no es posible emplear ninguna herramienta tecnológica: solo una exhaustiva revisión del código fuente, a veces línea por línea, dará finalmente con la fuente del problema.

¿Cómo pueden llegar a introducirse errores de estas clases en el software? Algunas de las causas están, sin duda, dentro de la esfera de control y previsión de los programadores y empresas desarrolladoras. Pero es igualmente cierto que otras escapan a los mejores esfuerzos en ese sentido.

Dentro de la categoría de los errores imputables a las personas se encuentra el deficiente *análisis de sistemas*, previo a la programación. En aquellas aplicaciones en que se busca automatizar procesos de la vida real, los analistas deben realizar un trabajo meticuloso para identificar toda la información relevante y los distintos procesos que el software deberá tener en cuenta, a partir del entorno real en que será empleado (por ejemplo, los procedimientos para el control de inventarios de una empresa). No agotar adecuadamente esta etapa de investigación y análisis es una de las principales razones para la introducción de errores lógicos en el software.

Los errores en tiempo de corrida suelen derivar de que se efectúen malas o insuficientes pruebas del software, antes de ponerlo en manos de los usuarios. La etapa de *depuración* que normalmente sigue al momento en que la primera versión de un software está lista,<sup>9</sup> debe estar orientada a someter a la aplicación a las pruebas más intensivas y fidedignas posibles, con respecto al entorno en que será eventualmente utilizada. Aparte de poner en evidencia los errores, no es infrecuente que este proceso revele la necesidad de introducir cambios diversos en el producto, que a la postre lo mejorarán.<sup>10</sup> Sin embargo, está claro que solo se puede hacer pruebas hasta un límite razonable; donde lo “razonable” no se puede predefinir para todas las clases de aplicaciones.<sup>11</sup>

Pero como ya lo señalamos, otras situaciones que pueden afectar el desempeño de una aplicación están fuera del control o de la razonable previsión del desarrollador. Entre ellas, la creciente complejidad del hardware, los cambios en el software de sistema subyacente, así como la imposibilidad de prever todas las situaciones a las que el programa será sometido, o el tiempo durante el cual será usado.<sup>12</sup> Las innumerables combinaciones posibles de hardware y software, así como las eventuales interacciones entre aplicaciones, escapan a las posibilidades de los procesos más rigurosos de depuración.

En este complicado panorama entran en juego también las inevitables presiones de mercado, que pueden tentar a una empresa desarrolladora a acortar artificialmente las etapas de creación y pruebas, en aras de asegurar su competitividad. Del mismo modo, se ha destacado la ausencia de estándares de desarrollo generalmente aceptados en la industria, como parámetro para fijar un

<sup>9</sup> La industria informática suele aplicar el calificativo de “beta” para referirse a un software que entra en proceso de pruebas, antes de su comercialización o distribución definitivas.

<sup>10</sup> Un interesante y exhaustivo estudio acerca del costo de no mantener una adecuada infraestructura para la depuración se encuentra en el documento “*The Economic Impacts of Inadequate Infrastructure for Software Testing*”, publicado por el Instituto Nacional de Estándares y Tecnología (NIST) de los EE.UU. en mayo del 2002. Está disponible electrónicamente en <http://www.nist.gov/director/prog-ofc/report02-3.pdf>

<sup>11</sup> Al menos un autor ha propuesto como estándar el considerar que un software de calidad razonablemente aceptable es aquél que no contiene más de un defecto por cada millón de líneas de código fuente. Véase <http://www.newsfactor.com/perl/story/19104.html> (página visitada el 26/8/2002).

<sup>12</sup> El mal llamado “error del milenio” o “Y2K” surgió fundamentalmente como resultado de ese último supuesto: aplicaciones desarrolladas tan atrás incluso como la década de 1960 hacían uso de convenciones de programación (la representación del año mediante sus últimos dos dígitos, en este caso) no adecuadas a la transición al año 2000. Muy posiblemente, al momento de escribirlas, no se preveía o imaginaba que pudieran encontrarse en uso activo treinta o más años después. Véase HESS ARAYA, Christian. “¿Qué aprendimos del Y2K?”, enero del 2000. Disponible electrónicamente en <http://www.hess-cr.com/publicaciones/dereinfo/y2k.html>

control de calidad mínimo en la producción del software.<sup>13</sup> Finalmente, debemos considerar igualmente las implicaciones de los procesos de “globalización” –que exigen a las empresas crear productos que tengan en cuenta las distintas regulaciones e idiosincrasias internacionales– como un factor más que acrecienta las probabilidades de error.<sup>14</sup>

## Condiciones para la existencia de una responsabilidad y sus eventuales eximentes

Dejando de lado las situaciones que, como explicamos, trascienden el control y previsión del programador, podemos encontrar sin duda ciertas condiciones en las que cabe establecer la existencia de un claro gravamen a cargo suyo (o de la empresa o entidad correspondiente), como fruto del incorrecto desempeño de un software y de los daños o perjuicios que éste pueda irrogar. Siguiendo las reglas clásicas de la responsabilidad subjetiva,<sup>15</sup> estas condiciones se reducirán a que pueda demostrarse el *dolo* o la *culpa*, ya fuere en la etapa de desarrollo y pruebas de la aplicación o incluso con posterioridad a su distribución a los usuarios.

### ***El software malicioso***

Habrá dolo, naturalmente, cuando el software haya sido deliberadamente diseñado para producir el resultado defectuoso o dañoso. En el mundo de la informática, a esta clase de aplicaciones se les conoce como *software malicioso*.<sup>16</sup> Los ejemplos abundan y podemos encontrarlos fundamentalmente en el submundo de los creadores de *virus informáticos*, *caballos de Troya*, *bombas lógicas* y *gusanos*, así como en la categoría del llamado *spyware*.

<sup>13</sup> Importantes esfuerzos están siendo desarrollados para suplir esta crítica omisión. Por ejemplo, en el terreno de la programación orientada a objetos, el *lenguaje unificado de modelaje* (UML, por sus siglas en inglés) gana terreno como el estándar más moderno y aceptado para el desarrollo cualitativo de software. Por otra parte, las empresas desarrolladoras cada vez más buscan someter sus procesos de control de calidad a estándares tales como ISO/IEC 9126, así como aplicar herramientas como el “Capability Maturity Model” (CMM) del Software Engineering Institute de la Universidad Carnegie Mellon de EE.UU. Véase al respecto <http://www.iso.org> y <http://www.sei.cmu.edu/cmm/cmm.html>, respectivamente.

<sup>14</sup> TOMPKINS, Tim. *Hardware and Software Liability*. Publicación electrónica disponible en la dirección <http://www.cs.rpi.edu/courses/fall00/ethics/papers/tompkt.html>, visitada el 20 de febrero del 2002.

<sup>15</sup> Ningún autor de que tengamos noticia ha propuesto la existencia de una responsabilidad sin culpa (objetiva) en esta materia.

<sup>16</sup> Aunque, sobra decirlo, el malicioso en realidad es el programador y no el software. Cualquier tecnología es, en sí, moralmente neutra.



- Los **virus informáticos** son programas que se “adhieren” a otros archivos, desde los cuales pueden lanzar sus ataques e infectar a otros sistemas informáticos. Normalmente incorporan tres componentes característicos: una *carga*, un *mecanismo de replicación* y otro de *elusión*. La carga<sup>17</sup> es el resultado, generalmente dañino o destructivo, que el virus está diseñado para ocasionar. Puede ir desde mostrar un mensaje determinado en pantalla, hasta modificar o eliminar archivos en el computador. El mecanismo de replicación, por medio del cual el virus “infecta” a otras aplicaciones o computadoras, está constituido por un conjunto de instrucciones cuyo propósito es diseminar el virus por medio de disquetes, mensajes de correo electrónico y otras vías. Finalmente, el mecanismo de elusión es el que procura lograr que el virus pueda actuar sin ser detectado.

La posibilidad teórica de que un programa de software se pudiese replicar por sí mismo fue propuesta por primera vez en 1949 por el gran matemático y precursor de la computación moderna, John von Neumann.<sup>18</sup> Pero no fue sino hasta noviembre de 1983, cuando el estudiante universitario Fred Cohen creó el primer virus, con el fin de apoyar sus investigaciones en torno al concepto.

- Un **caballo de Troya** también es un programa cuyo propósito es provocar un resultado perjudicial. Esta clase de software no actúa adhiriéndose a otro programa o documento, como lo hacen los virus, sino disfrazándose de otra cosa, tal como un juego, para inducir al usuario a ejecutarlo. Dicho expediente engañoso es lo que le da su nombre, que, desde luego, se inspira en la conocida leyenda homónima de la antigua Grecia.
- Una **bomba lógica** también incorpora una determinada carga, usualmente dañina, pero además incluye un *detonante*: solamente cuando se dé una condición prevista en el programa (como el arribo de una fecha o que el usuario presione cierta combinación de teclas), se activará la carga. A modo de ejemplo, en 1996, un programador que había sido descendido de su puesto en la empresa norteamericana *Omega Engineering Corp.*, programó una bomba lógica que “detonó” el 31 de julio, causando una pérdida a la compañía de más de \$10 millones en ventas y contratos perdidos. Fue procesado y sentenciado a

---

<sup>17</sup> *Payload* en la terminología inglesa.

<sup>18</sup> MICROSOFT CORP., “Virus (computer)”, artículo de la *Microsoft Encarta Enciclopedia 2002*.

tres años de prisión, en el primer caso fallado bajo la nueva *Fraud and Related Activities In Connection With Computers Act* de ese país.<sup>19</sup>

- Finalmente, se denomina **gusano** a un programa autocontenido (autónomo) cuya única o principal funcionalidad consiste en replicarse numerosas veces, ya sea en una sola computadora o a través de redes informáticas. Aun cuando no contenga una carga dañina específica, un gusano puede replicarse tantas veces como para finalmente provocar la caída del sistema o red, o por lo menos, ocasionar un rendimiento degradado. Típicamente residen en la memoria principal del computador, no en el disco duro, lo cual puede dificultar su detección. John Schoch y Jon Hupp fueron los primeros en realizar un análisis académico sobre los gusanos informáticos.<sup>20</sup>

Dichas categorías no son necesariamente excluyentes. Hoy en día es común encontrar formas híbridas de ellas.

La producción de virus informáticos y programas maliciosos similares ha ido en incremento tal, junto con sus consecuencias perniciosas, que los diversos ordenamientos han comenzado a reaccionar sancionando ya no solo civilmente sino además penalmente a sus creadores. En nuestro país, mediante ley N° 8148 del 24 de octubre del 2001, se adicionó en este sentido un artículo 229 bis al Código Penal para prever y sancionar la alteración de datos y sabotaje informático. Lamentablemente, debido a una inadvertencia legislativa, el tipo penal fue abrogado pocos meses después.<sup>21</sup>

Por su parte, por **spyware** se entiende una categoría de software malicioso, similar pero separada de la de los caballos de Troya, cuya funcionalidad consiste en recopilar información subrepticamente sobre el usuario o usuarios de un sistema informático, o bien supervisar sus patrones de conducta en línea, con el fin de reportar los resultados de vuelta a su creador. El propósito más común de estos datos es el de crear perfiles de consumo para efectos de mercadeo.

El mecanismo empleado más frecuentemente para diseminar *spyware* es el de la distribución de ciertos productos de software gratuito, los cuales presumiblemente tienen un propósito inocuo (como buscar y compartir música o

<sup>19</sup> NELSON, Sharon D. y SIMEK, John W. (editores). *Bytes in Brief. Op. cit.*, edición de abril del 2002. Hay más información también en la página <http://www.cybercrime.gov/lloydSent.htm>.

<sup>20</sup> BERGHEL, Hal. "The Code Red Worm". En *Communications of the ACM*, volumen 44, número 12, diciembre 2001, página 16.

<sup>21</sup> En efecto, como resultado de un conjunto de reformas al Código Penal efectuado mediante ley N° 8250 de 17 de abril del 2002, se promulgó un nuevo artículo 229 bis con el título de "Abandono dañino de animales".

vídeo en Internet), pero que una vez instalados comienzan también a vigilar clandestinamente al usuario y a generar los datos de interés. De allí la indicada semejanza con los caballos de Troya.

Aunque no son propiamente aplicaciones informáticas, las llamadas *cookies* (particularmente las *cookies remotas*) pueden ser empleadas en Internet como parte de un proceso de *spyware*, en la medida en que recopilen información de un navegante de la red, sin su conocimiento.<sup>22</sup>

### **Diseño imprudente o negligente de aplicaciones**

En materia de responsabilidad, tradicionalmente se caracterizan como imprudentes las acciones de una persona que exceden los parámetros normalmente asociados a la situación concreta en que se actúa y a sus consecuencias previsibles. Conlleva la ausencia de la diligencia y precaución debidas en atención a las circunstancias. Por el contrario, es negligente la conducta que no llega a alcanzar el estándar mínimo de comportamiento que ordinariamente se esperaría frente a una determinada situación. Es la “omisión de la diligencia o cuidado que debe ponerse en los negocios, en las relaciones con las personas y en el manejo o custodia de las cosas” y que equivale a la desidia, la falta de atención o el olvido de órdenes.<sup>23</sup> En términos simples, mientras que en la imprudencia se peca por exceso (hacer), en la negligencia se peca por defecto (no hacer).

Estos conceptos son aplicables al diseño de software. Aunque la imprudencia es un poco más difícil de visualizar en este contexto, sin duda tendríamos que caracterizar como tal –por ejemplo– a la conducta de quien asume un proyecto de programación para el cual no está preparado. Hay imprudencia precisamente porque se pretende hacer más de lo que se está entrenado para hacer y, consecuentemente, se deberá cargar con las consecuencias dañosas que de ello deriven.

En el terreno de la informática, lo que vamos a encontrar con mayor frecuencia serán más bien las hipótesis de negligencia.

<sup>22</sup> HESS ARAYA, Christian. “Derecho a la privacidad y *cookies*”. En *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 24, julio del 2000. También aparece en el libro *“Derecho informático y comercio electrónico. Doctrina y legislación.”* Publicado por la Facultad de Derecho y Ciencias Políticas de la Universidad Inca Garcilaso de la Vega, Lima, Perú, 2002.

<sup>23</sup> CABANELLAS, Guillermo. *“Diccionario de derecho usual”*. Cuarta edición, tomo III, Bibliográfica Omeba, Buenos Aires, 1962.

*“La negligencia ordinaria se aplica cuando un desarrollador de software no emplea el grado de cuidado que una persona razonablemente prudente usaría cuando desarrolla software (...). En los casos judiciales relacionados con la negligencia, se comprobará o descartará que el desarrollador de un producto ejercitó su ‘deber de cuidado’ en la manufactura y venta del indicado producto. Si se llegase a determinar que hay algo que un desarrollador de software debió hacer y que razonablemente habría sido esperado de él por todos los involucrados en el uso y distribución del software, entonces podrá ser hallado culpable de negligencia y ser condenado a pagar daños y perjuicios al demandante.”<sup>24</sup>*

Como se advirtió arriba, no existen lineamientos de programación universalmente aceptados, de manera que la apreciación de qué es y qué no es una conducta negligente en este sentido puede resultar harto subjetiva. Además, qué tan rigurosamente juzguemos a un programador o empresa de software probablemente deberá tener en cuenta también el contexto en que un producto sea desarrollado, distribuido y usado. Claramente no se podría aplicar los mismos parámetros a una pequeña aplicación creada para fines recreativos por un programador ocasional, que la distribuye gratuitamente y sin ninguna pretensión de calidad o rendimiento, que a un sistema operativo sofisticado que una experimentada empresa dedicada a ese giro comercializa ampliamente y que por sus características debería desempeñarse de un modo razonablemente a prueba de fallos.

Aun así, en cualquier manual o curso de programación podemos encontrar algunas recomendaciones generales, cuya inobservancia –siempre teniendo en cuenta las cambiantes circunstancias apuntadas arriba– probablemente sería interpretada como una injustificable omisión por un programador experimentado. Por citar tan solo un ejemplo, es de esperar que un software que reciba y procese datos ingresados por usuarios no especializados durante su ejecución, necesariamente contenga lo que en esta materia se conoce como *rutinas de validación*, por medio de las cuales se verifique si los datos suministrados son del tipo y rango esperados<sup>25</sup> y, caso contrario, se alerte al usuario del error para que lo corrija antes de que el programa continúe con su trabajo. Omitir esta clase de verificaciones será, en la mayoría de los casos, un claro indicio de un trabajo de programación negligente.

<sup>24</sup> TOMPKINS, Tim. *Op. cit.* Traducción libre del autor. Los paréntesis no son del original.

<sup>25</sup> Por ejemplo, que no se ingrese un número donde corresponda una fecha, o que, a pesar de que lo ingresado sí sea una fecha, ésta sea inválida, tal como “31 de febrero”; etcétera.

No obstante la ausencia de estándares de programación aplicables a la industria del software como un todo, eso no significa que no podamos encontrar lineamientos de esa índole en un caso concreto de examen. Por ejemplo, muchas empresas desarrolladoras de software tienen regulaciones internas que sus programadores deben conocer y aplicar. La comprobada inobservancia de esas directrices perfectamente podría conducir entonces al diagnóstico de una inexcusable negligencia y al establecimiento de las consiguientes responsabilidades, en caso de que un producto de software resulte defectuoso por tal motivo.

Otra posible valiosa fuente de análisis son los códigos de ética profesional que existen en el ámbito de las ciencias de la computación y que promulgan las entidades gremiales especializadas. A nivel internacional, destacan especialmente las regulaciones de la ACM y de la IEEE.<sup>26</sup> A nivel nacional, el Colegio de Profesionales en Informática y Computación (CPIC<sup>27</sup>) también posee su código de moral, cuya infracción podría emplearse eventualmente como parámetro para juzgar la existencia de una imprudencia o negligencia profesional.

### ***La (tristemente célebre) cláusula “as is” y otros mecanismos de exoneración de responsabilidad***

El carácter adhesivo de las licencias de uso de software, a que hemos hecho alusión en el capítulo precedente, ha dado la posibilidad a los proveedores de software de manipular abiertamente las condiciones que determinan su responsabilidad eventual por productos defectuosos. En particular, esto ha sido posible a través de lo que la doctrina anglosajona conoce como la *cláusula “as is”*,<sup>28</sup> que las empresas normalmente introducen como parte de la letra menuda de sus licencias.

El texto típico de una de estas cláusulas dirá más o menos lo siguiente (donde “XX” corresponde al nombre de la respectiva compañía):

*“Este producto es suministrado tal cual, sin garantías de ninguna especie. De la manera más amplia que permita la ley, XX niega cualquier clase de garantía implícita en el sentido de que este producto sea apropiado para algún propósito determinado. Todo el riesgo derivado del uso o rendimiento del producto corresponde al usuario. En ningún caso será XX responsable de cualquier daño directo o indirecto (incluyendo ganancias no percibidas, interrupción*

<sup>26</sup> Véase sus sitios web en <http://www.acm.org> y <http://www.ieee.org>, respectivamente.

<sup>27</sup> Véase <http://www.cpic.or.cr>.

<sup>28</sup> “Tal cual” o “así como está”.

*de los negocios, pérdida de información comercial o cualquier otra pérdida pecuniaria) que derive del empleo o de la incapacidad de emplear este producto o su documentación, aun cuando XX haya sido puesto en conocimiento de la posibilidad de esos daños.”*

Es obvio que esta clase de disposiciones ha sido siempre fuertemente censurada por su evidente carácter inequitativo y perjudicial para el usuario consumidor. No obstante, lo cierto es que las cláusulas “as is” han recibido un amplio respaldo judicial, al menos en la jurisprudencia estadounidense.

En efecto y a modo de ejemplo, al inicio de este capítulo se mencionó el caso de *M.A. Mortenson Co., Inc. versus Timberline Software Corp.*, en el que la primera presentó una cotización errónea, por causa de un error en el software distribuido por la segunda, sufriendo una pérdida de \$1.95 millones. Según se ha explicado,<sup>29</sup> Mortenson adquirió el software a través de una orden de compra girada a un distribuidor autorizado de la empresa desarrolladora Timberline. La licencia de uso, de tipo *shrink-wrap*, venía impresa en los sobres sellados que contenían los disquetes de instalación, así como en la contraportada del manual del producto. En ella, una cláusula “as is” limitaba la responsabilidad del fabricante al precio de adquisición del software y advertía a los usuarios que si no estaban de acuerdo con los términos del convenio, podían devolver el producto y recibir un reintegro.

El día en que se preparó y presentó la oferta utilizando el software de Timberline, la aplicación generó 19 mensajes de error, a pesar de los cuales Mortenson sometió su cotización. Posteriormente se logró determinar que Timberline estaba previamente enterada del defecto que provocaba los mensajes de error, pero había considerado que éste no era un gran problema y, además, que era poco probable que surgiera durante el empleo normal del producto.

Mortenson demandó a Timberline por los \$1.95 millones ofertados de menos. Sin embargo, Timberline ganó el caso tanto a nivel del tribunal de juicio, como del tribunal de apelación y la Suprema Corte del estado de Washington.

En su demanda, Mortenson alegó que la orden de compra, que no contenía limitación alguna de responsabilidad, había definido los términos de la relación contractual entre las partes, en vez de la licencia de uso, que de todos modos debía considerarse lesiva e inválida. Sin embargo, los diversos pronunciamientos judiciales estimaron que el licenciamiento tipo *shrink-wrap* resultaba una modalidad de contratación válida tanto bajo la legislación estatal como la federal. Se subrayó que, en el caso concreto, los términos de la licencia habían sido claramente

<sup>29</sup> HENNIGH, Scott E. *Op. cit.*

visibles y en ellos se dio oportunidad a Mortenson de no aceptarlos, devolviendo el software a cambio del reintegro del precio pagado.<sup>30</sup>

Como resultado de este caso, se ha comentado que los usuarios de software deben tomar muy en serio las licencias *shrink-wrap* y sus correlativas cláusulas “*as is*”. Deben leerlas y comprender sus alcances. Si no las consideran aceptables, deberían devolver el producto y adquirir otro o bien intentar negociar unos términos más favorables, opción por lo general irreal. Máxime, agregamos nosotros, si –como suele ocurrir en nuestro medio– los proveedores de software son empresas extranjeras, no sometidas a la legislación o jurisdicción nacionales.

– 0 –

El esfuerzo legislativo más importante –y controversial– que conocemos en materia de regulación de la responsabilidad de los fabricantes de software es la *Uniform Computer Information Transaction Act* (UCITA), de Estados Unidos. La UCITA surgió como una propuesta orientada a que los estados que integran ese país armonizaran sus legislaciones locales con relación al tema del licenciamiento del software y de toda clase de información digital. Originalmente estaba prevista para ser promulgada a nivel federal, a manera de un artículo 2B del *Uniform Commercial Code*. Sin embargo, la controversia que la ha rodeado desde sus inicios claramente imposibilitó esta posibilidad, por lo que en agosto de 1999 se varió el planteamiento inicial, dejando a opción de cada estado incorporar la nueva legislación o no. A la fecha, solo los estados de Maryland y Virginia lo han hecho.

La resistencia a la UCITA ha sido considerable. Contra ella incluso se han formado organizaciones especiales.<sup>31</sup> Se le atribuye en particular el inclinar la balanza injusta y desproporcionadamente a favor de los fabricantes de software y en contra de los consumidores. Por ejemplo, la UCITA expresamente autorizaba al fabricante, en una disputa sobre derechos de licenciamiento, a desactivar remotamente el software utilizado por una persona u organización, sin previa aprobación judicial, reduciendo o eliminando en algunos casos la responsabilidad por los daños y perjuicios que con ello se ocasionara.<sup>32</sup> También dificulta las posibilidades del consumidor de impugnar disposiciones irrazonables o injustas contenidas en contratos de licencia de tipo *shrink-wrap* y *click-on*.

<sup>30</sup> Esta cláusula es usual en contratos de licencia, aunque debe tenerse en cuenta que en tales casos el vendedor suele disponer que el reintegro monetario no incluirá los gastos de envío ni los impuestos de ventas.

<sup>31</sup> Por ejemplo, la “*Americans for Fair Electronic Commerce Transactions*” (AFFECT, <http://www.4cite.org>).

<sup>32</sup> Posteriores revisiones al texto han suprimido esta posibilidad, conocida como “*self-help provision*”.

En particular, la UCITA formaliza la modalidad de licenciamiento “as is”, de modo que –en la práctica– los fabricantes de software no tendrían que responder por reclamos de que su producto es defectuoso o no opera como debiera. En el peor de los casos, un reclamo estaría limitado al precio de adquisición del producto.

*“Esto es como decirle a las empresas expendedoras de alimentos que si a sabiendas venden productos contaminados, solamente estarán obligadas a reembolsar el precio pagado a los consumidores que se enfermen al ingerirlos.”<sup>33</sup>*

La UCITA brinda además al fabricante la posibilidad de fijar la jurisdicción territorial aplicable en caso de disputas judiciales, la cual podría incluso centrarse en otro país. No hay obligación de verificar que el software distribuido esté libre de virus y, si no lo estuviera, una simple cláusula contractual –que el adquirente no vería sino hasta después de pagar por el producto– eximiría al proveedor de toda responsabilidad.<sup>34</sup> Para colmo, la publicación de los resultados de las pruebas de rendimiento realizadas a un software en particular sólo sería posible con el consentimiento del fabricante. Lo cual lógicamente nos lleva a preguntarnos: ¿quién estará de acuerdo en autorizar esto si sus resultados fueran negativos?

La *American Bar Association* (Colegio de Abogados de EE.UU.) oportunamente integró un comité especial para examinar las implicaciones de la UCITA. El panel concluyó que la ley solamente conduciría a un incremento en la litigiosidad relativa al licenciamiento de software. Igualmente criticó el lenguaje ambiguo que eventualmente daría a los vendedores la posibilidad de actuar prácticamente del modo que quisieran en caso de disputa con un usuario.<sup>35</sup> Eventualmente, la ABA declinó dar su crucial respaldo a la ley.<sup>36</sup> Esto sin duda contribuyó a que en agosto del 2003, la *National Conference of Commissioners on Uniform State Laws*, entidad patrocinadora de la UCITA, finalmente decidiera suspender sus esfuerzos por lograr que las legislaturas estatales aprueben la ley, lo cual, aunque no equivale a desistir de ella, en la práctica le resta viabilidad política.<sup>37</sup>

<sup>33</sup> SIMONS, Barbara. “Melissa’s Message”, en *Communications of the ACM*, volumen 42, número 6, junio de 1999, página 25.

<sup>34</sup> KANER, Cem y PELS, David L. “UCITA: a bad law that protects bad software”. Publicación electrónica disponible en <http://www.badssoftware.com/networld.htm>. Julio de 1999.

<sup>35</sup> THIBODEAU, Patrick. “ABA: UCITA Law Confusing, Susceptible to Litigation”, en *Computerworld Online* (<http://www.computerworld.com>), edición del 4 de febrero del 2002.

<sup>36</sup> Véase <http://www.nwfusion.com/news/2003/0212ucitahits.html>, 12 de febrero del 2003.

<sup>37</sup> Véase <http://www.sciam.com/article.cfm?SID=mail&articleID=000AFFF8-B887-1F80-B57583414B7F0103>, noviembre del 2003.



## Soluciones propuestas

Cada día, más y más especialistas de la materia se expresan en el sentido de que el software debería ser juzgado bajo los mismos estándares de responsabilidad que cualquier otro producto. Señalan, por ejemplo, que carece de sentido que un fabricante de neumáticos para automóvil que comercialice un producto defectuoso pueda ser demandado y sancionado, mientras que los grandes fabricantes de software se salen con la suya, distribuyendo aplicaciones a las que frecuentemente se les descubren errores y otros defectos, sin que se les pueda responsabilizar debido a los términos de sus licencias.<sup>38</sup>

Recientemente, la Academia Nacional de Ciencias de EE.UU. se ha manifestado señalando que los desarrolladores de software deberían ser legalmente responsables por los defectos de seguridad de sus productos que den lugar a intrusiones en los sistemas informáticos de sus usuarios. Pide además la promulgación de leyes que obliguen a los primeros a reportar los problemas de seguridad que subsecuentemente descubran en las aplicaciones distribuidas.

En la actualidad, cada vez que un *hacker* logra aprovechar los defectos de seguridad de un programa de software, se desata una controversia y un señalamiento de acusaciones mutuas que van desde el propio intruso hasta el investigador que descubre y denuncia el problema. Usualmente, las sanciones penales y civiles recaen solamente sobre el *hacker*, mientras que la empresa desarrolladora, cuando mucho, no experimenta más que una humillación pública. Las compañías frecuentemente niegan la existencia de problemas, alegando que no se conoce ningún caso real en que la falla haya sido explotada por un usuario malicioso. En otros casos se limitan a afirmar que el problema es puramente teórico e imposible de reproducir en el mundo real.<sup>39</sup>

Por otra parte, nadie propone que los fabricantes de software sean tenidos como ilimitadamente responsables por los defectos de sus productos. Después de todo, es de humanos errar, especialmente si esos humanos son programadores de software. Pero las reglas para limitar o eliminar la responsabilidad deben ser restrictivas y claras.

---

<sup>38</sup> SCHNEIER, Bruce. Revista electrónica *Crypto-Gram* (<http://www.counterpane.com/crypto-gram.html>), edición del 15 de febrero del 2002.

<sup>39</sup> BOWMAN, Lisa M. “*Companies on the hook for security*”, publicación electrónica de la empresa CNET News.com, 23 de enero del 2002 (disponible en <http://news.com.com/2100-1023-821266.html>).

KANER y PAGLIA,<sup>40</sup> juristas dedicados a la temática del control de calidad de software, consideran que los productores de software únicamente deberían quedar exentos de responsabilidad por daños causados por un defecto que,

- Fuere desconocido al momento de distribuir el producto, siempre que ese desconocimiento no sea el resultado de prácticas negligentes de desarrollo o depuración; o,
- Fuere claramente advertido al adquirente como un limitación conocida de la aplicación en la literatura que la acompaña, expresada en términos fácilmente comprensibles al usuario promedio del segmento de mercado a que el producto se oriente.

Esta última propuesta pone de relieve la importancia del deber de “*full disclosure*”, en particular para el caso del software cuyos defectos conduzcan a comprometer la seguridad de los sistemas informáticos. Por tal se conoce a la práctica de divulgar públicamente la información relativa a un defecto de seguridad de un software, así como a un conjunto de lineamientos acerca de cómo hacerlo. La persona u organización que descubre la vulnerabilidad, o bien una entidad representativa tal como el “Computer Emergency Response Team” (CERT),<sup>41</sup> efectúa la divulgación generalmente después de alertar al desarrollador y de otorgarle un determinado lapso de tiempo para corregir el problema antes de publicarlo.<sup>42</sup>

La cuestión de exactamente cuánta información divulgar y en qué momento hacerla pública es altamente polémica. Algunos postulan el principio de divulgación completa e inmediata, incluyendo los detalles técnicos específicos que indiquen de qué manera podría ser explotada la vulnerabilidad. Otros estiman que una cantidad limitada de información debería ser sometida a un grupo selecto, cierto tiempo después del hallazgo. Finalmente, otros opinan que no se debería publicar del todo la información relativa a vulnerabilidades. La política del CERT, por ejemplo, incluye: informar al desarrollador tan pronto sea posible, luego de recibida la denuncia; mantener informado al denunciante acerca de las acciones tomadas; y, en la mayoría de los casos, revelar la información al público cuarenta y cinco días después del reporte inicial, ya sea que la empresa desarrolladora haya actuado al respecto o no.<sup>43</sup>

<sup>40</sup> Citados por SIMONS, *op. cit.*, página 26.

<sup>41</sup> Véase <http://www.cert.org>

<sup>42</sup> Definición de la enciclopedia electrónica WhatIs.com, disponible en <http://whatis.techtarget.com/dice/0,294542,sid9,00.html?track=NL-34&Offer=dicewhtssur1>

<sup>43</sup> Idem. Para más información, véase <http://www.vulnwatch.org/disclosure.html>

## Situación nacional

No existe en nuestro país una legislación específicamente relativa a la responsabilidad de los fabricantes de software. Por ende, los casos concretos relativos a esta materia tendrían que ser examinados con aplicación de las normas generales sobre responsabilidad contractual y extracontractual.

El Colegio de Profesionales en Informática y Computación (CPIC), conforme a su Ley Orgánica,<sup>44</sup> tiene entre sus objetivos el de “Tutelar los derechos e intereses legítimos de quienes contraten los servicios de los profesionales miembros del Colegio, por las actividades, actos u omisiones que realicen o dejen de realizar en el ejercicio de su profesión, sin perjuicio de las responsabilidades civiles o penales en que puedan incurrir.” (artículo 2, inciso g).

Nótese que este ámbito de tutela está explícitamente limitado –como sería de esperar tocante a la naturaleza del ente gremial en cuestión– a las personas físicas que integren el CPIC, no a las empresas desarrolladoras o comercializadoras como tales.<sup>45</sup> En adición, la ley no contempla ninguna previsión más allá de la responsabilidad disciplinaria de los informáticos por faltas a la ética profesional.

¿Pero existirá tal cosa como una malpraxis profesional en tratándose de los profesionales en computación? El tema ha planteado duda en otros ordenamientos:

*“La malpraxis es la omisión de aplicar el alto estándar de cuidado que un miembro de una profesión debería emplear (...) No se ha determinado que la malpraxis pueda ser siempre aplicada en casos relativos a la computación. Esto se debe simplemente a que no se ha llegado a concluir inequívocamente si los profesionales en informática son, de hecho, profesionales. Algunos poseen las características apropiadas (...) Sin embargo, ciertos profesionales en computación poseen poca autonomía y no existe ninguna organización profesional legalmente reconocida (...) En algunos casos, los tribunales se han negado a señalar semejante gravamen, sobre la base de que simplemente porque una actividad sea técnicamente más compleja e importante no significa que debe asociársele un mayor grado de responsabilidad. (...) Sin embargo, existen ejemplos de casos en que el veredicto ha favorecido al*

<sup>44</sup> Número 7537 de 22 de agosto de 1995.

<sup>45</sup> De todos modos, la membresía en el CPIC es facultativa para los profesionales en informática, a diferencia de lo que ocurre con otras actividades como la abogacía o la medicina.

*accionante que reclama una ‘malpraxis informática’, cuando se determinó que la empresa consultora ‘no actuó razonablemente, a la luz de su conocimiento superior y experticia en el campo de los sistemas de computación’. (...) Observamos aquí el inicio de una tendencia: la información y las decisiones relativas a la responsabilidad informática son confusas y frecuentemente conflictivas.”<sup>46</sup>*

Nuestra situación es, desde luego, otra. En Costa Rica, por disposición legal, no existe duda acerca del *status* profesional de los informáticos. Pero, como ya se expresó arriba, las regulaciones normativas aplicables no van más allá de los aspectos disciplinarios.

En conclusión,

*“La responsabilidad por hardware y software es un tema nuevo en la sociedad y en la ley, de modo que las reacciones han sido lentas. No está claro cómo aplicar la legislación existente a estos productos y servicios, o bien cuales nuevas leyes son necesarias. En cuanto se trata de elementos que están presentes cada vez más en nuestras vidas diarias, la sociedad deberá afrontar esto y espero que nos preparemos debidamente, en vez de apresurar políticas que no enfrenten eficazmente la situación. En términos simples, ‘... el objeto de la normativa sobre responsabilidad es inducir un comportamiento apropiado tanto de parte del consumidor como del productor’ (...) La pregunta que debemos hacernos acerca de quién deba ser responsable es: ¿quién tiene el poder de impedir los daños? (...) Aquéllos que lo tengan deben ser impulsados a ejercerlo.”<sup>47</sup>*

---

<sup>46</sup> TOMPKINS, Tim. *Op. cit.*; traducción libre del autor. Los paréntesis no son del original.

<sup>47</sup> *Idem.*

## Bibliografía

APPEL, Andrew W. y FELTEN, Edward W. "Technological access control interferes with noninfringing scholarship", en *Communications of the ACM*, volumen 43, número 9, setiembre del 2000.

ARMOUR, Phillip G. "The case for a new business model", en *Communications of the ACM*, volumen 43, número 8, agosto del 2000.

(Autor no especificado). "Fin de la falsificación y la piratería", en revista *Enlace Mundial*. Promotora del Comercio Exterior de Costa Rica (PROCOMER), año 3, número 2. San José, setiembre de 1999.

BERGHEL, Hal. "The Code Red Worm". En *Communications of the ACM*, volumen 44, número 12, diciembre 2001.

BOWMAN, Lisa M. "Companies on the hook for security", publicación electrónica de la empresa CNET News.com, 23 de enero del 2002 (disponible en <http://news.com.com/2100-1023-821266.html>).

BURK, Dan L. "Copyrightable functions and patentable speech", en *Communications of the ACM*, volumen 44, número 2, febrero del 2001.

CABANELLAS, Guillermo. "Diccionario de derecho usual". Cuarta edición, tomo III, Bibliográfica Omeba, Buenos Aires, 1962.

CARRASCO, Juan. "ASP, una compleja relación jurídica", en *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 35, junio del 2001.

CARRASCO, Juan. "Directiva de derechos de autor, ¿una verdadera armonización?", en *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 98, noviembre del 2001.

CARRASCOSA LÓPEZ, Valentín y otros. *El comercio informático*. En "Ponencias del VI Congreso Iberoamericano de Derecho e Informática de la FIADI", publicación del comité organizador. Montevideo, 1998.

CORREA, Carlos M. y otros. *Derecho informático*. Ediciones Depalma, Buenos Aires, primera reimpresión, 1994.

DAVIS, Randall. “The digital dilemma”, en *Communications of the ACM*, volumen 44, número 2, febrero del 2001.

DEMPSEY, Bert J. y otros. “Who is an open source software developer?”, en *Communications of the ACM*, volumen 45, número 2, febrero del 2002.

DIJKSTRA, Edsger W. “The end of computing science?”, en *Communications of the ACM*, volumen 44, número 3, marzo del 2001.

DVORAK, John C. “The software protection racket”, en la revista *PC Magazine*, 20 de agosto de 1998. Versión electrónica disponible en [http://www.zdnet.com/pcmag/insites/dvorak\\_print/jd980806.htm](http://www.zdnet.com/pcmag/insites/dvorak_print/jd980806.htm).

ERDOZAIN, José Carlos. “Un ensayo sobre la protección jurídica de los programas de ordenador”. En *Revista Electrónica de Derecho e Informática*, número 108, setiembre del 2002. Disponible en <http://www.alfa-redi.org/upload/revista/90302--12-50-Der.Infor%20julio01.pdf>

FERNÁNDEZ DELPECH, Horacio. *Protección jurídica del software*. Abeledo-Perrot, 1 edición, Buenos Aires, 2000.

FUENTES PINZÓN, Fernando, “Contratación para la elaboración del software individualizado. Estudio comparado de la legislación iberoamericana.” En *Revista de Ciencias Jurídicas*, número 102 (setiembre – diciembre 2003). Universidad de Costa Rica – Colegio de Abogados de Costa Rica.

FUENTES PINZÓN, Fernando. “¿Por medio de cual rama jurídica debe protegerse al software?”. En *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 37, agosto del 2001.

GHERSI, Carlos Alberto. *Contratos Civiles y Comerciales*, Parte General y Especial, Tomo II, segunda edición, Editorial Astrea, 1992.

GOPAL, Ram D. y SANDERS, G. Lawrence. “Global software piracy: you can't get blood out of a turnip”, en *Communications of the ACM*, volumen 43, número 9, setiembre del 2000.

GRIMES, Brad. “The At-Your-Service ASP”, en *PC Magazine*, volumen 22, número 21, 25 de noviembre del 2003.

GUASCH DÍAZ, Diego Manuel. “La extracción de Normas de Boletines Oficiales en papel y de ediciones en Internet. Algunas Consideraciones Jurídicas relevantes para las Empresas de Bases de Datos”. En *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), agosto del 2000.

HERNÁNDEZ VALLE, Rubén. *Constitución Política de la República de Costa Rica. Comentada y Anotada*. Editorial Juricentro, primera edición, San José, 1998.

HESS ARAYA, Christian. “Apuntes en torno al software *shareware* o *trialware*”, en *Revista Electrónica de Derecho e Informática*, número 37, agosto del 2001.

HESS ARAYA, Christian. *Creación de una base de datos de jurisprudencia constitucional, orientada a objetos*. Disponible electrónicamente en mi sitio web personal de <http://www.hess-cr.com>, junio de 1997.

HESS ARAYA, Christian. “Derecho a la privacidad y *cookies*”. En *Revista Electrónica de Derecho e Informática*, número 24, julio del 2000. También aparece en el libro “*Derecho informático y comercio electrónico. Doctrina y legislación.*” Publicado por la Facultad de Derecho y Ciencias Políticas de la Universidad Inca Garcilaso de la Vega, Lima, Perú, 2002.

HESS ARAYA, Christian. “Piratería no, pero ...”, en el periódico *La Nación*, edición del 12 de noviembre del 2001, San José, Costa Rica.

HESS ARAYA, Christian. “Propiedad intelectual de las bases de datos”, en *Revista Electrónica de Derecho Informático* (<http://www.alfa-redi.org>), número 28, noviembre del 2000.

HESS ARAYA, Christian. “¿Qué aprendimos del Y2K?”, enero del 2000. Disponible electrónicamente en <http://www.hess-cr.com/publicaciones/dereinfo/y2k.html>.

INDUSTRY CANADA / CANADIAN HERITAGE, “*Consultation paper on digital copyright issues*”. Documento electrónico disponible en <http://strategis.ic.gc.ca/SSG/rp01099e.html>. Junio del 2001.

JEANGORGES, Gonzalo. “La protección legal de las obras de Informática en la Argentina”, en *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 22, mayo del 2000.

JESDANUN, Anick. “*Software Bug Blamed for Blackout Alarm Failure*”, noticia divulgada el 12 de febrero del 2004 por la agencia AP y disponible electrónicamente en <http://www.siliconvalley.com/mld/siliconvalley/news/editorial/7941550.htm>.

JIJENA LEIVA, Renato Javier. “La protección legal del software en Chile. Análisis de la ley 17336 y sus modificaciones.”, en *Ponencias del VI Congreso Iberoamericano de Derecho e Informática de la FIADI*. Publicación del Comité Organizador. Montevideo, 1998.

JURAN, Belinda y otros. “Trade Secret Protection for Source Code”, en *Internet Alert*. Revista electrónica publicada por Hale and Dorr, LLP (<http://www.haledorr.com>). Edición de 17 de julio del 2001.

KANER, Cem y PELS, David L. “UCITA: a bad law that protects bad software”. Publicación electrónica disponible en <http://www.badsoftware.com/networld.htm>. Julio de 1999.

KAPPELMAN, Leon A. “Killer Apps & Dead Bodies”, en *Information Week* de 26 de junio del 2000; disponible electrónicamente en <http://www.informationweek.com/792/92uwlk.htm>.

KIM, Jonathan, “Patenting air or protecting property?”. Publicación electrónica del diario *Washington Post*, 11 de diciembre del 2003; disponible electrónicamente en <http://www.washingtonpost.com/ac2/wp-dyn?pagename=article&node=&contentId=A54548-2003Dec10&notFound=true>.

KOHN, Paul. “Understanding Copyright Law”, en *Revista Electrónica de Derecho e Informática* [<http://www.alfa-redi.org>], número 98, noviembre del 2001.

LEVINE, Jeffrey A. “*Shrink-wrap Licenses - Shrinking Protection*”, 1996. Documento electrónico disponible en <http://www.swiggartagin.com/aigc/tic64.html>.

LEE, Jae-Nam y otros. “IT outsourcing evolution – past, present, and future”. En *Communications of the ACM*, volumen 46, número 5, mayo del 2003.

LIEBERMAN, Henry y FRY, Christopher. “Will software ever work?”, en *Communications of the ACM*, volumen 44, número 3, marzo del 2001.

MANGASSARIAN, Elina, “Copyright Protection of Software”. Documento electrónico disponible en <http://mishpat.net/articles/elina.doc>. Octubre del 2000.



MARESCA, Fernando. “¿Hacia el patentamiento del software?”, en *Revista Electrónica de Derecho e Informática*, número 23, junio del 2000.

MARESCA, Fernando. “Protección jurídica del software: un debate abierto”. En *Derecho y Tecnología*, revista del Centro de Investigaciones en Nuevas Tecnologías de la Universidad Católica del Táchira, Venezuela, N° 1/2002.

MARESCA, Fernando. “Reflexiones sobre la comercialización del software”, en *Revista Electrónica de Derecho e Informática*, número 20, marzo del 2000.

McORMOND, Russell, “A review of software patent issues”. Documento electrónico disponible en <http://www.flora.ca/russell/drafts/software-patent2003.shtml>, 2003.

MICROSOFT CORP., “Virus (computer)”, artículo de la *Microsoft Encarta Enciclopedia 2002*.

NELSON, Sharon D. y SIMEK, John W. (editores). *Bytes in Brief*. Revista electrónica publicada por Sensei Enterprises, Inc. y el bufete Nelson & Wolfe, disponible en Internet (<http://www.senseient.com>). Edición de julio de 1999.

NELSON, Sharon D. y SIMEK, John W. (editores). *Bytes in Brief. Op. cit.*, edición de abril del 2002.

ORGANIZACIÓN MUNDIAL DE LA PROPIEDAD INTELECTUAL (OMPI). “¿Cómo proteger los secretos industriales de su PYME?”. Documento electrónico disponible en [http://www.wipo.int/sme/es/ip\\_business/trade\\_secrets/trade\\_secrets.htm](http://www.wipo.int/sme/es/ip_business/trade_secrets/trade_secrets.htm).

OMPI. “La propiedad intelectual para las pequeñas y medianas empresas”. Publicación electrónica distribuida en formato de CD-ROM, 2003.

OMPI. “Legislación nacional y regional existente relativa a la propiedad intelectual en materia de bases de datos”. Memorándum preparado por la Oficina Internacional y disponible electrónicamente en [http://www.wipo.org/spa/meetings/infdtat97/db\\_im\\_2.htm](http://www.wipo.org/spa/meetings/infdtat97/db_im_2.htm). Junio de 1997.

OMPI. “¿Qué es un dibujo o modelo industrial?”. Documento electrónico disponible en la dirección [http://www.wipo.int/about-ip/es/index.html?wipo\\_content\\_frame=/about-ip/es/industrial\\_designs.html](http://www.wipo.int/about-ip/es/index.html?wipo_content_frame=/about-ip/es/industrial_designs.html).

REBACK, Gary L., "Patently absurd". Publicación electrónica de la revista *Forbes*, disponible en <http://www.forbes.com/asap/2002/0624/044.html>, 24 de junio del 2002.

RÍOS, Wilson Rafael. "Aspectos Legales del Software Libre (Open Source)", en *Revista Electrónica de Derecho Informático*, N° 117 - Junio 2003, disponible en <http://www.alfa-redi.org/revista/data/61-1.asp>.

ROMERO PÉREZ, Jorge Enrique, "La externalización de actividades laborales (outsourcing)". En *Revista de Ciencias Jurídicas*, número 102 (setiembre – diciembre 2003). Universidad de Costa Rica – Colegio de Abogados de Costa Rica.

ROSE, Lance. "*Netlaw: your rights in the online world*". Osborne McGraw-Hill. 1ª edición, 1995.

SAMUELSON, Pamela. "Good news and bad news on the intellectual property front", en *Communications of the ACM*, volumen 42, número 3, marzo de 1999.

SAMUELSON, Pamela. "Reverse engineering under siege", en *Communications of the ACM*, volumen 45, número 10, octubre del 2002.

SAMUELSON, Pamela. "*Towards more sensible anti-circumvention regulations*". Documento electrónico disponible en la dirección <http://www.sims.berkeley.edu/~pam/papers/fincrypt2.doc>.

SAMUELSON, Pamela. "Trade secrets vs. free speech", en *Communications of the ACM*, volumen 46, número 6, junio del 2003.

SAMUELSON, Pamela. "Why the anticircumvention regulations need revision", en *Communications of the ACM*, volumen 42, número 9, setiembre de 1999.

SANDBERG, Jared. "Goodbye, Shrink-Wrap: Software for Rent". En la revista *Newsweek*, edición del 18 de setiembre del 2000, volumen 136, número 12.

SCHNEIER, Bruce. "Protecting Copyright in the Digital World", en la revista electrónica *Crypto-Gram* (<http://www.counterpane.com/crypto-gram.html>). 15 de agosto del 2001.

SCHNEIER, Bruce. Revista electrónica *Crypto-Gram* (<http://www.counterpane.com/crypto-gram.html>), edición del 15 de febrero del 2002.

SIMONS, Barbara. "Melissa's Message", en *Communications of the ACM*, volumen 42, número 6, junio de 1999.

SUÑÉ LLINAS, Emilio. *El caso Microsoft y las aporías del viejo derecho*. En "ABC Economía", Madrid, 9 de abril del 2000.

TÉLLEZ VALDÉS, Julio. "La patentabilidad del software. El debate que continúa.", en *Ponencias del IX Congreso Iberoamericano de Derecho e Informática de la FIADI*. Publicación en CD-ROM del Comité Organizador. San José, abril del 2002.

THIBODEAU, Patrick. "ABA: UCITA Law Confusing, Susceptible to Litigation", en Computerworld Online (<http://www.computerworld.com>), edición del 4 de febrero del 2002.

TOMPKINS, Tim. *Hardware and Software Liability*. Publicación electrónica disponible en la dirección <http://www.cs.rpi.edu/courses/fall00/ethics/papers/tompkt.html>, visitada el 20 de febrero del 2002.

TOURETZKY, David S. "Free speech rights for programmers", en *Communications of the ACM*. Volumen 44, número 8, agosto del 2001.

ULL PONT, Eugenio. *Protección jurídica de las bases de datos conforme a la Directiva 96/9/CEE del Parlamento Europeo y del Consejo, de 11 de marzo de 1996*. Publicación electrónica disponible en <http://www.omdi.net/members/espanolmiembros/reivdi/pont.htm>, Madrid, 1998.

VARGAS VÁSQUEZ, Elizabeth. "*Contratos informáticos de desarrollo de software a la medida*". Tesis para optar al grado de licenciada en Derecho, Universidad Internacional de las Américas, San José, 2002.

VILLEGAS, Ricardo. "La originalidad como condicionante para la protección internacional de las bases de datos", en *Revista Electrónica de Derecho e Informática* (<http://www.alfa-redi.org>), número 38, setiembre del 2001.

VITHARANA, Padmal. "Risks and challenges of component-based software development". En *Communications of the ACM*, volumen 46, número 8, agosto del 2003.

WALKER, Jesse, “*Cyberspace’s Legal Visionary*”. Disponible electrónicamente en <http://www.reason.com/0206/fe.jw.cyberspaces.shtml>, junio del 2002.

ZÚÑIGA CHAVES, Nuria. *Manual Práctico sobre Derechos de Autor*. Editado por la Asociación de Compositores y Autores Musicales de Costa Rica (ACAM), San José, 1998.

## Anexo I: Sitios relevantes en Internet

La siguiente es una breve lista de algunos sitios web de interés para obtener mayor información acerca de los temas tratados en esta obra. Aunque todas las direcciones brindadas eran correctas a la fecha de la publicación, es necesario tener en cuenta que el carácter dinámico y cambiante de la Internet podría determinar que, posteriormente, algunas de ellas cambien o desaparezcan. Agradeceré informarme de cualquier modificación (o bien sugerir otros sitios relevantes) a través de mi página personal, que aparece citada de primera:

### En castellano

- **Hess Araya, Christian** (sitio web personal), <http://www.hess-cr.com>
- **Asociación Costarricense de Derecho Informático** (ACDI), <http://www.hess-cr.com/~acdi/>
- **Organización Mundial de la Propiedad Intelectual** (OMPI), <http://www.wipo.int/index.html.es>
- **Organización Mundial del Comercio** (OMC), <http://www.wto.int/indexsp.htm>
- **Revista Electrónica de Derecho e Informática**, <http://www.alfa-redi.org>

### En inglés

- **Business Software Alliance** (BSA), <http://www.bsa.org>
- **Bytes In Brief**, <http://www.senseient.com>
- **Computer Professionals for Social Responsibility**, <http://www.cp-sr.org>
- **GNU Project / Free Software Foundation**, <http://www.gnu.org>

- **Intellectual Property Rights (IPR) Helpdesk**, <http://www.ipr-helpdesk.org>
- **The Mishpat Cyberlaw Informer**, <http://www.cyberlawinformer.com>
- **The Free Software Foundation**, <http://www.fsf.org/>
- **The Open Source Initiative**, <http://www.opensource.org/>
- **U.S. Copyright Office**, <http://www.loc.gov/copyright/>
- **Cybercrime**, <http://www.cybercrime.gov>
- **The Twenty Most Critical Internet Security Vulnerabilities**, <http://www.sans.org/top20/>
- **Copyright and Fair Use**, <http://fairuse.stanford.edu/>

## **Anexo II: Directrices relativas al empleo ilegal de software en las oficinas gubernamentales**

Decreto ejecutivo número 30.151-J del 1 de febrero del 2002, publicado en el diario oficial "La Gaceta" del día 21 siguiente:

EL PRESIDENTE DE LA REPÚBLICA  
Y LA MINISTRA DE JUSTICIA Y GRACIA

En uso de las facultades conferidas en el Artículo 140, incisos 3) y 18) de la Constitución Política,

### **Considerando:**

1°—Que la Ley N° 6683 de Derechos de Autor y Derechos Conexos del 14 de octubre de 1982 y sus reformas contiene los principios generales que tutelan los derechos de los creadores de las obras del ingenio y de los artistas intérpretes o ejecutantes, los productores de fonogramas y videogramas y de los organismos de radiodifusión.

2°—Que los programas de cómputo están protegidos por la Ley N° 6683 anteriormente citada y se han convertido en un factor importante en la gestión y funcionamiento de las instituciones gubernamentales, siendo el Gobierno Central uno de los mayores usuarios de la tecnología de la informática.

3°—Que la Ley N° 8039 sobre Procedimientos de Observancia de los Derechos de Propiedad Intelectual del 12 de octubre del 2000 establece los procedimientos aplicables en materia civil y penal para la protección de los derechos intelectuales, incluidos los derechos de autor.

4°—Que el Gobierno Central debe asegurar el mejor mantenimiento del equipo de cómputo al servicio del Estado, lograr la seguridad de la información, prevenir virus y asegurar la eficiente custodia de los archivos informáticos de las instituciones públicas.

5°—Que la reproducción, distribución y uso no autorizado de programas de cómputo constituye una actividad ilícita, perjudica gravemente las oportunidades de empleo y el ingreso tributario generado por la industria de programas de cómputo, incluyendo fabricantes, productores y distribuidores.

6°—Que el Gobierno debe ser ejemplo para la empresa privada, en cuanto al manejo adecuado de los programas de cómputo, cumpliendo con lo que prescribe la normativa vigente tanto nacional como internacional.

### **Decreta:**

Artículo 1°—Se ordena que todo el Gobierno Central se proponga diligentemente prevenir y combatir el uso ilegal de programas de cómputo, con el fin de cumplir con las disposiciones sobre derecho de autor que establece la Ley N° 6683 y sus reformas y la Ley N° 8039, acatando las provisiones pertinentes de los acuerdos internacionales, incluyendo el Acuerdo sobre los Aspectos de los Derechos de Propiedad Intelectual relacionados con el Comercio y también las otras disposiciones de la normativa nacional vigente.

Artículo 2°—Cada Ministerio deberá realizar un inventario inicial de los equipos existentes y de los programas que tengan las computadoras y el número de copias autorizadas de cada programa, determinando la fecha de instalación y versión de cada uno. El término para el cumplimiento de este inventario inicial no será mayor de diez (10) meses a partir de la entrada en vigencia de este decreto.

Artículo 3°—Posterior al levantamiento del inventario, cada Ministerio del Gobierno Central deberá establecer sistemas y controles dentro de un término de dieciocho (18) meses para garantizar la utilización en sus computadoras única y exclusivamente de aquellos programas que cumplan con los derechos de autor correspondientes. Cualquier programa que exceda el número autorizado o que no cuente con la licencia correspondiente deberá removerse inmediatamente.

El Ministro designará a una persona como responsable, entre otras cosas, de presentar un informe anual ante el Registro Nacional de Derechos de Autor y Derechos Conexos, en el que hará constar que el respectivo Ministerio cumple con la protección de los derechos de autor relativos a los programas de cómputo.

Artículo 4°—El Registro Nacional de Derechos de Autor y Derechos Conexos en coordinación con cada Ministerio deberá desarrollar y mantener un sistema de información que registre los resultados del inventario inicial de equipos y programas y que de allí en adelante registre las adquisiciones de las autorizaciones adicionales e instalaciones o el uso de copias adicionales permitido bajo tales autorizaciones. Cada Ministerio deberá garantizar que se tengan



suficientes autorizaciones para cubrir todos los equipos y los programas en uso, guardándose la documentación correspondiente en un solo lugar con la custodia necesaria.

Artículo 5º—El Registro Nacional de Derechos de Autor y Derechos Conexos obligatoriamente llevará a cabo auditorías, para lo cual coordinará con cada Ministerio del Gobierno Central, debiendo realizar inventarios periódicos de las computadoras, al menos una vez al año, para determinar la fidelidad del sistema de información y el acatamiento a las normas sobre derecho de autor. Para esto, dicho registro contará con el auxilio y colaboración del Departamento de Informática del Registro Nacional, y podrá además, en caso necesario solicitar a la Junta Administrativa del mismo Registro, aprobación para la contratación de personal externo adecuado para este cumplimiento.

Artículo 6º—A partir de la entrada en vigencia de este Decreto no podrán instalarse programas de cómputo que no cuenten con la respectiva licencia de uso en ninguna oficina del Gobierno Central. En la etapa de evaluación para la contratación, todas las solicitudes de compra para programas de cómputo deberán ser consultadas con el experto en informática que ellos tengan designado en sus instituciones.

Artículo 7º—Para los efectos de las adquisiciones y la utilización de programas de cómputo de acuerdo con las disposiciones de este Decreto, el experto en informática de cada Ministerio del Gobierno Central deberá cumplir con las siguientes reglas:

- a. Establecer y mantener una política comprensiva de manejo de programas de cómputo y un sistema efectivo para garantizar la adquisición y uso adecuado de todos los programas de cómputo.
- b. Garantizar que el respectivo Ministerio cumpla con la protección del derecho de autor de los programas de cómputo.
- c. Establecer medidas para evaluar el cumplimiento en respectivo Ministerio de las disposiciones en materia de derecho de autor en lo concerniente a la adquisición de programas de cómputo, utilizados por el referido Ministerio de conformidad con las disposiciones de este decreto;
- d. Dirigir y ofrecer apoyo institucional al entrenamiento apropiado del personal de servicio público en materia de derecho de autor relacionado con los programas de cómputo y las políticas y procedimientos adoptados para cumplir con ellos.

e. Exhortar a todos los contratistas y proveedores del Gobierno Central a cumplir con las normas sobre derecho de autor, a adquirir y utilizar programas de cómputo con sus respectivas licencias de uso.

Artículo 8°—Cada Ministerio del Gobierno Central elaborará manuales para el uso e instalación de programas de ordenador y velarán por el entrenamiento de todos los funcionarios públicos de acuerdo con las necesidades y el uso legal de los programas de cómputo, incluyendo la expedición de notas de advertencia, el establecimiento y la aplicación de medidas disciplinarias por incumplimiento de las disposiciones del presente decreto.

Artículo 9°—Los Ministerios deberán adoptar las medidas necesarias para asegurarse que los proyectos de presupuestos para programas de cómputo y los requerimientos para el procesamiento de datos incluyan recursos adecuados para la adquisición de las licencias correspondientes de los programas de cómputo que necesiten.

Artículo 10.—Cada Ministerio cooperará ampliamente en la aplicación de este Decreto, compartiendo información que pudiera ser de utilidad para combatir el uso ilícito de programas de cómputo.

Artículo 11. —Las Instituciones del Estado, en los casos que sea posible, podrán utilizar software de código abierto en sus diferentes aplicaciones, como una alternativa útil; garantizando el respeto a los derechos de la propiedad intelectual.

*(Así reformado por decreto ejecutivo N° 30236 de 22 de febrero del 2002.)*

Artículo 12.—Este decreto rige a partir de su publicación.

Dado en la Presidencia de la República.—San José, a las nueve horas del día primero de febrero del año dos mil dos.

*MIGUEL ÁNGEL RODRÍGUEZ ECHEVERRÍA.—La Ministra de Justicia y Gracia, Mónica Nagel Berger.*

## **Sobre el autor**

Christian Hess Araya es costarricense; licenciado en Derecho y máster en Informática. Letrado de la Sala Constitucional de la Corte Suprema de Justicia de Costa Rica desde 1996. Fue integrante de la Junta Directiva del Colegio de Abogados 2002-2003. Ha sido profesor de Derecho Informático de la Universidad de La Salle y de Derecho Público en la Universidad Latina de Costa Rica, así como de la Maestría en Ciencias de la Computación e Informática de la Universidad de Costa Rica en San José. Miembro fundador de la Asociación Costarricense de Propiedad Intelectual y de la Asociación Costarricense de Derecho Informático. Miembro de la “*Association for Computing Machinery*” (ACM). Sitio web: [www.hess-cr.com](http://www.hess-cr.com)